Enabler Test Specification for

ECMA Script Mobile Profile Test Code

Candidate Version 1.0 – 09 Oct 2007

Open Mobile Alliance

OMA-ETS-ESMP_Test_Code-V1_0-20071009-C

Contents

1. Scope

This version of the test plan does not include test procedures for the following parts of ESMP specification and may include them in the future versions:

- Link object
- Image object
- Button object
- Screen object

Currently, there are 948 tests in this testsuite. Total estimated time to complete one test pass is around 16 - 20 hours depending on the tester's familiarity with the test suite. This estimate does not include time required for error reporting.

2. References

2.1 Normative References

[RFC2119]	"Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, March 1997, URL: http://www.ietf.org/rfc/rfc2119.txt
[ESMP]	ECMAScript Mobile Profile Specification, Version 11 Aug 2003, Open Mobile Alliance™, URL http://www.openmobilealliance.org/
[ECMACR]	Crypto Object for ECMAScript Mobile Profile Specification, Open Mobile Alliance™, URL http://www.openmobilealliance.org/
[ECMA-262]	ECMAScript Language Specification, W3C Recommendation, 19 December 2000, URL http://www.ecma.ch/ecma1/STAND/ECMA-262.HTM
[JSBIBLE]	JavaScript Bible, 4th Edition, Author: Danny Goodman, Published by: Hungry Minds, Inc.
[XHTMLMP11]	XHTML Mobile Profile 1.1 Specification, Open Mobile Alliance™, URL http://www.openmobilealliance.org/

2.2 Informative References

[OMADICT]	"Dictionary for OMA Specifications", Version x.y, Open Mobile Alliance™, OMA-ORG-Dictionary-Vx_y, URL:http://www.openmobilealliance.org/

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope”, are normative, unless they are explicitly indicated to be informative.

The following numbering scheme is used:

xxx-y.z-con-number where:

xxx	Name of enabler, e.g. MMS or Browsing
y.z	Version of enabler release, e.g. 1.2 or 1.2.1
’con’	Indicating this test is a conformance test case
number	Leap number for the test case

Or

xxx-y.z-int-number where:

xxx	Name of enabler, e.g. MMS or Browsing
y.z	Version of enabler release, e.g. 1.2 or 1.2.1
’int’	Indicating this test is a interoperability test case
number	Leap number for the test case

3.2 Definitions

None

3.3 Abbreviations

OMA	Open Mobile Alliance

4. Introduction

"ECMAScript – Mobile Profile" [ESMP] is the wireless markup scripting language specified by the Open Mobile Alliance (OMA). The ESMP is based on the ECMAScript Release 3 [ECMA-262], which in turn is a standardized form of traditional javascript scripting language.

The [ECMA-262] contains the specification for the core language and core objects of the scripting language. [ESMP] is a superset of this specification and also includes browser host objects, XHTML DOM objects and the DOM2 Core objects.

The Crypto Object as specified in the "Crypto Object for the ECMAScript Mobile Profile" specification [ESMPCR], provides the access to cryptographic features of the user agent.

This test plan specifies the test procedures for testing the user-agent's support to the above three specifications.

5. Testing Procedure Template

Sections 5 to 12 of this document contain the testing procedures. Following is an example test case (given in italic font) and the description about its format.

49.xx.xx

Requirement: Mandatory
SCR: ESMP -C-039
Test path: esmp_Sec9/objLocation/assign1a.xhtml

***Assertion:** When the **assign()** method of the **location** object is called, where a string representing an URL is supplied as an argument, then the browser MUST load the document pointed by the supplied URL. The history stack MUST be pushed with this URL.*

Traceability:

[ESMP]	*Section 9.4.3.1*

Verification:
Step 1: Verify that page titled "Assign 1A" is loaded.
Step 1: Verify that page titled "Assign 1A" is loaded.
Step 2: Verify that page titled "Assign 1B" is loaded again.
Step 3: Test passed. Post results.

Example:
var x = location.assign("page2.xhtml"); //must navigate to page2.xhtml

- Test case ID: A unique id of the testcase within the testsuite in the format 49.xx.xx
- Requirement: Whether passing this test is a mandatory or optional requirement, if the user-agent claims conformance to the ESMP specification.
- SCR: The Static Conformance Number of ESMP specification, if any.
- Test path: The path of the test file relative to the root directory of the testsuite.
- Assertion: One or more assertions covered in this test.
- Traceability: Traceability of the assertions to the specifications from which they are derived.
- Verification: If the test needs manual verification, then the steps to verify.
- Example: For some test cases, example is also shown for clarify.

6. LANGUAGE SYNTAX

6.1 Lexical Conventions

49.01.01

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-003

Test path: esmp_Sec5/5.1/5.1.1/utf8.xhtml

Assertion: At a minimum, ECMAScript Mobile Profile implementations MUST support the UTF-8 encoding of code points into bit patterns for the source text.

Traceability:

[ESMP]	Section 5.1.1
[ECMA262]	Section 6

49.01.02

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-002

Test path: esmp_Sec5/5.1/5.1.1/utf16.xhtml

Assertion: At a minimum, ECMAScript Mobile Profile implementations MUST support the UTF-16 encoding of code points into bit patterns for the source text.

Traceability:

[ESMP]	Section 5.1.1
[ECMA262]	Section 6

49.01.03

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.1/case.xhtml

Assertion: ECMAScript Mobile Profile is a case-sensitive language. All language keywords, variables, objects, methods and properties MUST use the proper capitalization.

Traceability:

[ESMP]	Section 5.1.1
[ECMA262]	Section 6

49.01.04

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.2/tabcharacter1.xhtml

Assertion: The white space characters can occur within a string, and MUST be treated as part of the string.

Assertion: The white space characters can occur between any two tokens, and MUST be ignored.

Traceability:

[ESMP]	Section 5.1.2
[ECMA262]	Section 7.2

49.01.05

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.2/tabcharacter2.xhtml

Assertion: The white space characters cannot occur in any other kind of token, other than a literal string.

Traceability:

[ESMP]	Section 5.1.2
[ECMA262]	Section 7.2

49.01.06

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.2/linefeed.xhtml

Assertion: The line feed character (\u000A) cannot occur within any token including literal strings.

Traceability:

[ESMP]	Section 5.1.2
[ECMA262]	Section 7.2

49.01.07

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.2/carriagereturn.xhtml

Assertion: The carriage return character (\u000D) cannot occur within any token including literal strings.

Traceability:

[ESMP]	Section 5.1.2
[ECMA262]	Section 7.2

49.01.08

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.2/ulineseparator.xhtml

Assertion: The line separator character (\u2028) cannot occur within any token including literal strings.

Traceability:

[ESMP]	Section 5.1.2
[ECMA262]	Section 7.2

49.01.09

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.2/uparagseparator.xhtml

Assertion: The paragraph separator character (\u2029) cannot occur within any token including literal strings.

Traceability:

[ESMP]	Section 5.1.2
[ECMA262]	Section 7.2

49.01.10

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-0005

Test path: esmp_Sec5/5.1/5.1.3/withsemicolon.xhtml

Assertion: Semicolons MUST be included as statement separators, and are required following ECMAScript Mobile Profile language statements.

Traceability:

[ESMP]	Section 5.1.3

49.01.11

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-0005

Test path: esmp_Sec5/5.1/5.1.3/withoutsemicolon.xhtml

Assertion: Semicolons MUST be included as statement separators, and are required following ECMAScript Mobile Profile language statements. It is incorrect to end a statement without semicolon.

Traceability:

[ESMP]	Section 5.1.3

49.01.12

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.4/linecomment.xhtml

Assertion 1: Single line comments contain all characters starting with character “//” upto the end of the line. Compiler MUST ignore these comments.

Assertion 2: Single line comments MUST not contain line terminator characters.

Traceability:

[ESMP]	Section 5.1.4
[ECMA262]	Section 7.4

49.01.13

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.4/blockcomment.xhtml

Assertion: Multi line comments contain all characters starting with “/*” and ending with “*/”. Compiler MUST ignore these comments.

Assertion: Multi line comments can also contain line terminator characters.

Traceability:

[ESMP]	Section 5.1.4
[ECMA262]	Section 7.4

49.01.14

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-0005

Test path: esmp_Sec5/5.1/5.1.5/reservedwordbreak.xhtml

Assertion: The ECMAScript keywords cannot be used as identifiers.

Traceability:

[ESMP]	Section 5.1.5
[ECMA262]	Section 7.5.2

49.01.15

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-0005

Test path: esmp_Sec5/5.1/5.1.5/futurereservedwordsuper.xhtml

Assertion: The ECMAScript future reserved words cannot be used as identifiers.

Traceability:

[ESMP]	Section 5.1.5
[ECMA262]	Section 7.5.3

49.01.16

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-0005

Test path: esmp_Sec5/5.1/5.1.5/nullidentifier.xhtml

Assertion: The Null literal (null keyword) cannot be used as identifier.

Traceability:

[ESMP]	Section 5.1.5
[ECMA262]	Section 7.5.1

49.01.17

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-0005

Test path: esmp_Sec5/5.1/5.1.5/booleanidentifier.xhtml

Assertion: The Boolean literals (true, false keywords) cannot be used as identifiers.

Traceability:

[ESMP]	Section 5.1.5
[ECMA262]	Section 7.5.1

49.01.18

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier01.xhtml

Assertion: An identifier name can start with a capital letter.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.19

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier02.xhtml

Assertion: An identifier name can have a capital letter in the middle.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.20

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier03.xhtml
Assertion: An identifier name can have a capital letter as the last character.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.21

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier04.xhtml

Assertion: An identifier name can start with dollar sign character ($).

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.22

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier05.xhtml

Assertion: An identifier name can have dollar sign character ($) in the middle.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.23

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier06.xhtml

Assertion: An identifier name can end with dollar sign character ($).

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.24

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier07.xhtml

Assertion: An identifier name can start with two consecutive dollar sign characters.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.25

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier08.xhtml

Assertion: An identifier name can have two consecutive dollar sign characters in the middle.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.26

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier09.xhtml

Assertion: An identifier name can end with two consecutive dollar sign characters.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.27

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier10.xhtml

Assertion: An identifier name can start with underscore character (_).

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.28

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier11.xhtml

Assertion: An identifier name can have underscore character (_) in the middle.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.29

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier12.xhtml

Assertion: An identifier name can end with underscore character (_).

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

49.01.30

Requirement: Mandatory
SCR: None

Test path: esmp_Sec5/5.1/5.1.6/legalidentifier13.xhtml

Assertion: An identifier or string literal can have Unicode escaped characters in any position.

Traceability:

[ESMP]	Section 5.1.6
[ECMA262]	Section 7.6

6.2 Execution Context

49.04.01

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context1.xhtml

Assertion 1: The global code of a particular program MUST not include any source text that is parsed as part of a function body.

Assertion 2: The function code of a particular function body MUST not include any source text that is parsed as part of a nested function body.

Assertion 3: When the control enters the global context, then **this** value MUST be global object, it self.

Traceability:

[ESMP]	Section 5.4
[ECMA262]	Section 10.2.1

49.04.02

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context2.xhtml

Assertion 1: A variable declared inside the function MUST have only local scope, and has no effect on a global variable with the same name.

Assertion 2: A variable declared inside the function MUST have only local scope, and cannot be accessed from out side the function.

Assertion 3: Irrespective of the actual position of declaration of variables in the source code in a particular scope and their assigned values, all of the declared variables MUST be available in the beginning of the scope with initial value **undefined**.

Traceability:

[ESMP]	Section 5.4
[ECMA262]	Section 10

49.04.03

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context3.xhtml

Assertion 1: When a function with a formal parameter list is called, then variables in the function context MUST be set with the same names as of the formal parameters.

Assertion 2: When a function with a formal parameter list is called, then variables in the function context MUST be set with the same names as of the formal parameters. The values of these variables MUST be set with the values supplied in the function call and in the same order.

Assertion 3: When a function call supplies fewer parameter values, than required by the formal parameter list of the function, then the extra formal parameters MUST be set with value undefined.

Traceability:

[ESMP]	Section 5.4.1
[ECMA262]	Section 10.1.3

49.04.04

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context4.xhtml

Assertion: When two or more formal parameters of a function share the same name, then each of these parameters MUST be set with the value supplied in the function call, for the last of parameter with this name.

Traceability:

[ESMP]	Section 5.4.1
[ECMA262]	Section 10.1.3

49.04.05

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context5.xhtml

Assertion: When two or more formal parameters of a function share the same name, and if the function call does not supply the value for the last of these parameters, then each of these parameters with the same name MUST be set with value 'undefined'.

Traceability:

[ESMP]	Section 5.4.1
[ECMA262]	Section 10.1.3

49.04.06

Test Type: No
Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context6.xhtml

Assertion: When a function call supplies additional parameter values than required by the formal parameter list of the function, then those additional values MUST be ignored.

Traceability:

[ESMP]	Section 5.4.1
[ECMA262]	Section 10.1.3

49.04.07

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context7.xhtml

Assertion 1: When the control enters the execution context for a function code, an **arguments** object MUST be created, whose **length** property is set to the number of actual parameter values supplied to by the caller to this function.

Assertion 2: The **arguments** object available in a function scope, MUST contain the actual parameter values supplied by the caller as its properties. Each of these values can be accessed by the array syntax **arguments[**n**]**, where n is 0 to (**arguments.length** – 1).

Traceability:

[ESMP]	Section 5.4.1
[ECMA262]	Section 10.1.8

49.04.08

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context8.xhtml

Assertion: A property named **callee** of the **arguments** object MUST be available within a function scope, which is a reference to the same function.

Traceability:

[ESMP]	Section 5.4.1
[ECMA262]	Section 10.1.8

49.04.09

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context9.xhtml

Assertion: When the control enters an execution context of a function code, where the **this** value is not provided by the caller, then the **this** value MUST be set as the global object.

Traceability:

[ESMP]	Section 5.4
[ECMA262]	Section 10.2.3

49.04.10

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context10.xhtml

Assertion: When the control enters an execution context of a function code, where the **this** value provided by the caller is not an object, then the **this** value MUST be set as the global object.

Traceability:

[ESMP]	Section 5.4
[ECMA262]	Section 10.2.3

49.04.11

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context11.xhtml

Assertion: When the control enters an execution context of a function code, where the caller provides an object as the **this**, then the **this** value MUST be set as the object provided by the caller.

Traceability:

[ESMP]	Section 5.4
[ECMA262]	Section 10.2.3

49.04.12

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-xxx

Test path: esmp_Sec5/5.4/context12.xhtml

Assertion: Variables declared in side any block { }, which is not a function body, MUST not have the scope limited to the block.

Traceability:

[ESMP]	Section 5.4
[ECMA262]	Section 10

6.3 Language Syntax and Semantics

6.3.1 Expressions

49.05.01

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/object_native1.xhtml

Assertion: The property of an object can be accessed by dot notation using the syntax: objectName**.**propertyName

Assertion: The property of an object can be accessed by bracket notation using the syntax: objectName**["**propertyName**"]**

Traceability:

[ESMP]	Section 5.5.1
[ECMA262]	Section 11.2.1

49.05.02

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/object_custom1.xhtml

Assertion: The property of an object can be accessed by dot notation using the syntax: objectName**.**propertyName

Assertion: The property of an object can be accessed by bracket notation using the syntax: objectName**["**propertyName**"]**

Traceability:

[ESMP]	Section 5.5.1
[ECMA262]	Section 11.2.1

Assertion: When the **new** operator is applied on an non-native object type, which supports constructor, then an instance of that object MUST be created.

Traceability:

[ESMP]	Section 5.5.1
[ECMA262]	Section 11.2.2

Example:

```
function square(side1, side2) { this.area = side1*side2;  }

var smallSquare  = new square(100, 100);     //smallSquare is an object of type 'square'.
```

49.05.03

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/non_function1.xhtml

Assertion: When a function call is made, where no function with that name exists, then a TypeError exception MUST be thrown.

Traceability:

[ESMP]	Section 5.5.1
[ECMA262]	Section 11.2.3

6.3.2 Operators

49.06.01

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/delete1.xhtml

Assertion: When the unary operator **delete** is applied on an expression, then the expression MUST be evaluated, and if the evaluated expression results in a primitive data type, then a Boolean true MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.1

49.06.02

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/delete2.xhtml

Assertion: When the unary operator **delete** is applied on an expression, then the expression MUST be evaluated, and if the evaluated expression results as a property P of an object O, where the property P of the object O is deletable, then the property P MUST be removed from the object O, and a Boolean true MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.1

49.06.03

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/delete3.xhtml

Assertion: When the unary operator **delete** is applied on an expression, then the expression MUST be evaluated, and if the evaluated expression results as a property P of an object O, where the property P does not exist in the object O, a Boolean true MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.1

49.06.04

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/delete4.xhtml

Assertion: When the unary operator **delete** is applied on an expression, then the expression MUST be evaluated, and if the evaluated expression results as a property P of an object O, where the property P of the object O is not deletable, then the property P MUST not be removed from the object O, and a Boolean false MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.1

49.06.05

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/void1.xhtml

Assertion: When the unary operator **void** is applied on an expression, then the expression MUST be evaluated, and the return value MUST be undefined.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.2

49.06.06

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/typeof1.xhtml

Assertion: When the unary operator **typeof** is applied on an expression, then the expression MUST be evaluated, and if the result of the expression is a property P of an object O, then the return value MUST be a string as per the following table:

Data type of value V	Return value
Undefined	"undefined"
null	"undefined"
boolean	"boolean"
string	"string"
number	"number"

Object	"object"

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.3

49.06.07

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/typeof2.xhtml

Assertion: When the unary operator **typeof** is applied on an expression, then the expression MUST be evaluated, and if the result of the expression is value V, then the return value MUST be a string as per the following table:

Data type of value V	Return value
Undefined	"undefined"
null	"undefined"
boolean	"boolean"
string	"string"
number	"number"
Object (other than Function)	"object"
Function object	"function"
Host object	Implementation-specific

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.3

49.06.08

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/increment1.xhtml

Assertion: When the prefix increment unary operator '++' is applied on an expression E, then the expression E MUST be evaluated into a numeric value N, as per number conversion rules and MUST be incremented by 1. The return value MUST be the incremented number.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.4

49.06.09

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/increment2.xhtml

Assertion: When the postfix increment unary operator '++' is applied on an expression E, then the expression MUST be evaluated into a numeric value N, as per number conversion rules and MUST be returned as a return value. Then, the numeric value N of the expression E MUST be incremented by 1.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.3.1

49.06.10

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/decrement1.xhtml

Assertion: When the prefix decrement unary operator '--' is applied on an expression E, then the expression E MUST be evaluated into a numeric value N, as per number conversion rules and MUST be decremented by 1. The return value MUST be the decremented number.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.5

49.06.11

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/decrement2.xhtml

Assertion: When the postfix decrement unary operator '--' is applied on an expression E, then the expression MUST be evaluated into a numeric value N, as per number conversion rules and MUST be returned as a return value. Then, the numeric value N of the expression E MUST be decremented by 1.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.3.2

49.06.12

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/unary_plus.xhtml

Assertion: When the unary operator '+' is applied on an expression, then the expression MUST be evaluated into a numeric value N, as per number conversion rules and return the resulting number.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.6

49.06.13

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/unary_minus.xhtml

Assertion When the unary operator '-' is applied on an expression, then the expression MUST be evaluated into a numeric value N, as per number conversion rules and return the resulting number with its sign negated.

Assertion When the unary operator '-' is applied on an expression, where the expression evaluates to a non-numeric value, then **NaN** MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.7

49.06.14

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/bitwise_not.xhtml

Assertion: When the bitwise NOT unary operator '~' is applied on an expression, then the expression MUST be evaluated into a an integer value N, as per integer conversion rules. Then the bitwise complement MUST be applied to this integer and the resulting integer MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.8

49.06.15

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/logical_not.xhtml

Assertion: When the logical NOT unary operator '!' is applied on an expression, then the expression MUST be evaluated into a boolean value B, as per boolean conversion rules. Then the opposite boolean value of the value B MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.4.9

49.06.16

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/multiplication1.xhtml

Assertion: When the multiplication operator '*' is applied on two operands A and B, then both the operands MUST be first converted into numeric values M and N as per the conversion rules. Then the operator MUST return a value as per the following table:

Operands M and N	**Return value**
Either of them is NaN	NaN
One is infinity and other is zero	NaN
Both are of same sign	Value with positive sign
Each has different sign	Value with negative sign.
One of them is infinity	Signed Infinity
Both are infinities	Signed nfinity
If their product is too large	Signed infinity
If their product is too small	Signed zero.
All other cases	Product rounded to nearest representable value.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.5.1

49.06.17

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/division1.xhtml

Assertion: When the division operator '/' is applied on two operands A and B, then both the operands MUST be first converted into numeric values M and N as per the conversion rules. Then the operator MUST return a value as per the following table:

Operands M and N	**Return value**
Both are of same sign	Value with positive sign
Each has different sign	Value with negative sign.
Either of them is NaN	NaN
Both are infinities	NaN
Both are zero's.	NaN
M is non-zero finite value, N is infinity	Signed zero.
M is non-zero finite value, N is zero	Signed infinity
M is infinity, N is non-zero finite value.	Signed infinity.
M is infinity and N is zero	Signed infinity
If their quotient is too large	Signed infinity
If their quotient is too small	Signed zero.
All other cases	Quotient rounded to nearest representable value.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.5.2

49.06.18

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/remainder1.xhtml

Assertion: When the remainder operator '%' is applied on two operands A and B, then both the operands MUST be first converted into numeric values M and N as per the conversion rules. Then the operator MUST return a value as per the following table:

Operands M and N	**Return value**
Sign of N is positive	Value with positive sign
Sign of N is negative	Value with negative sign.
Either of them is NaN	NaN
Both are infinities	NaN
Both are zero's.	NaN
M is infinity	NaN
N is zero.	NaN
M is finite, N is infinity	Same as N
M is zero, N is finite	Same as N
Other cases	Floating-point remainder whose value is: M – (N * q) Where q is an integer.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.5.2

49.06.19

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/addition1.xhtml

Assertion: When the addition operator '+' is applied on two number type operands M and N, then the return value MUST be as per following table:

Operands M and N	Return value
Either of them is NaN	NaN
Infinities of opposite signs	NaN
Infinities of same sign	Infinity of that same sign.
One is infinity and other is finite.	Same as infinite operand
Both are negative zero's	-0
Both are positive zero's	+0
One is positive zero, other negative zero.	+0
One is a zero, other is non-zero finite.	Same as non-zero operand
Both have same magnitude non-zero finite value but of opposite signs	+0
If their sum is too large	Signed infinity
If their sum is too small	Signed zero.
All other cases	Sum rounded to nearest representable value.

Assertion: When the addition operator '+' is applied on two operands A and B, where either of them are non-number type, then both MUST be converted into strings as per string conversion rules and a concatenated string of these two strings MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.6.1

49.06.20

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/subtraction1.xhtml

Assertion: When the substraction operator '-' is applied on two number type operands M and N, then the return value MUST be as per following table:

Operands M and N	Return value
Either of them is NaN	NaN
Infinities of same signs	NaN
Infinities of opposite signs	Same as M.
M is infinity and N is finite.	Same as M.
M is finite and N is infinity.	Infinity with opposite sign of N.
M positive zero, N negative zero	+0
M negative zero, N positive zero	-0
Both are positve zero's.	+0
Both are negative zero's	+0
M is a zero, N is non-zero finite.	Same as N, but with opposite sign.
M is a non-zero finite, N is zero	Same as M.
Both have same magnitude non-zero finite value and of same sign	+0
If their difference is too large	Signed infinity
If their difference is too small	Signed zero.
All other cases	Difference rounded to nearest representable value.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.6.2

49.06.21

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/bitwise_leftshift.xhtml

Assertion: When the bitwise left shift operator '<<' is applied on two operands A and B, then the operands MUST be converted into 32 bit integers M and N as per conversion rules. Then the bit pattern of the integer M MUST be left shifted by the amount of bits specified by number N, and the resulting signed 32 bit integer MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.7.1

49.06.22

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/bitwise_signed_rightshift.xhtml

Assertion: When the bitwise signed right shift operator '>>' is applied on two operands A and B, then the operands MUST be converted into 32 bit integers M and N as per conversion rules. Then a sign-extending right shift of the bit pattern of the integer M MUST be performed by the amount of bits specified by number N, and the resulting signed 32 bit integer MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.7.2

49.06.23

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/bitwise_unsigned_rightshift.xhtml

Assertion: When the bitwise unsigned right shift operator '>>>' is applied on two operands A and B, then the operands MUST be converted into 32 bit integers M and N as per conversion rules. Then a zero-filling right shift of the bit pattern of the integer M MUST be performed by the amount of bits specified by number N, and the resulting signed 32 bit integer MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.7.3

49.06.24

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/less_than1.xhtml

Assertion: When the relational operator '<' is applied on two operands A and B, then the operands MUST be compared and the return value MUST be as per following table:

Operands M and N	Return value
Either of them is NaN	Boolean false
Both are of same number value	Boolean false
Zeros of opposite signs.	Boolean false
M is positive infinity	Boolean false
N is positive infinity	Boolean true
N is negative infinity	Boolean false
M is a negative infinity	Boolean true
M & N are finite numbers, where M is greater than N	Boolean true
M & N are finite numbers, where M is greater than N	Boolean false
M & N are strings, where M is a prefix of N	Boolean true
M & N are strings, where N is a prefix of M	Boolean false
M & N are strings	Boolean value depending on the string value.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.1

49.06.25

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/greater_than1.xhtml

Assertion: When the relational operator '>' is applied on two operands A and B, then the operands MUST be compared and the return value MUST be boolean true if A is greater than B, or boolean false, otherwise.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.2

49.06.26

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/less_equal1.xhtml

Assertion: When the relational operator '<=' is applied on two operands A and B, then the operands MUST be compared and the return value MUST be boolean true if A is less than or equal to B, or boolean false, otherwise.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.4

49.06.27

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/greater_equal1.xhtml

Assertion: When the relational operator '>=' is applied on two operands A and B, then the operands MUST be compared and the return value MUST be boolean true if A is greater than or equal to B, or boolean false, otherwise.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.5

49.06.28

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/instanceof1.xhtml

Assertion: When the operator **instanceof** is applied on two operands A and B, where A is an object of type B, then a boolean true MUST be returned.

Assertion: When the operator **instanceof** is applied on two operands A and B, where A is not an object of type B, then a boolean false MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.6

49.06.29

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/instanceof2.xhtml

Assertion: When the operator **instanceof** is applied on two operands A and B, where B is not an object type, then the exception TypeError MUST be thrown.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.6

49.06.30

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/instanceof3.xhtml

Assertion: When the operator **instanceof** is applied on two operands A and B, where B is non-instantiable object (eg: Math object), then the exception TypeError MUST be thrown.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.6

49.06.31

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/in_operator1.xhtml

Assertion: When the operator **in** is applied on two operands A and B, where A is a property name of the object B, then the return value MUST be boolean true.

Assertion: When the operator **in** is applied on two operands A and B, where A is not a property name of the object B, then the return value MUST be boolean false.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.7

49.06.32

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/in_operator2.xhtml

Assertion: When the operator **in** is applied on two operands A and B, where B is not an object type, then the exception TypeError MUST be thrown.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.8.7

49.06.33

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/equals1.xhtml

Assertion: When the equality operator '==' is applied on two operands M and N, then a comparision of values of M and N MUST be performed, and if they are equal, then a boolean value MUST be returned as per following table.

Operands M and N	Return value
Type of M is undefined	Boolean true
Type of M is null	Boolean true
M is null, N is undefined	Boolean true
M is undefined, N is null	Boolean true
M is NaN	Boolean false
N is NaN	Boolean false
M and N have same number values	Boolean true
M and N are zero's with different signs	Boolean true
M and N have any other number values	Boolean false
M is a string	Boolean true, if M and N have same exact sequence of characters. Else, boolean false
M is boolean	Boolean true, if both M and N are true or false, Else, boolean false
M & N both refer to same object	Boolean true
M & N each refers to different objects	Boolean false
One is of number type and other is a string or boolean type	Convert the non-number type into number type, compare and return a boolean
If one of them is a string or number type and the other is an object type.	Convert the object type to its primitive value and compare.
All other cases	Boolean false

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.9.1

49.06.34

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/not_equals1.xhtml

Assertion: When the does-not-equals operator '!=' is applied on two operands A and B, then a comparision of two operands MUST be done as per equality rules, and if they are equal, a boolean false MUST be returned.

Assertion: When the does-not-equals operator '!=' is applied on two operands A and B, then a comparision of two operands MUST be done as per equality rules, and if they are not equal, a boolean true MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.9.2

49.06.35

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/strict_equals1.xhtml

Assertion: When the strict-equals operator '===' is applied on two operands A and B, then a boolean value MUST be returned as per following table.

Operands M and N	Return value
M and N are of different types	Boolean false
M is of undefined type	Boolean true
M is of Null type	Boolean true
M is NaN	Boolean false
N is NaN	Boolean false
M and N have same number values	Boolean true
M and N are zero's with different signs	Boolean true
M and N have any other number values	Boolean false
M is a string	Boolean true, if M and N have same exact sequence of characters. Else, boolean false
M is boolean	Boolean true, if both M and N are true or false, Else, boolean false
M & N both refer to same object	Boolean true
M & N each refers to different objects	Boolean false

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.9.4

49.06.36

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/strict_not_equals1.xhtml

Assertion: When the strict-does-not-equals operator '!==' is applied on two operands A and B, then a comparision of the operands MUST be performed as per strict equality rules, and if they are equal, boolean false MUST be returned.

Assertion: When the strict-does-not-equals operator '!==' is applied on two operands A and B, then a comparision of the operands MUST be performed as per strict equality rules, and if they are not equal, boolean true MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.9.4

49.06.37

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/bitwise_AND.xhtml

Assertion: When the binary bitwise AND operator '&' is applied on two operands A and B, then a bitwise comparision MUST be done on each corresponding bit position of the integer forms of A and B. For each bit position, if the corresponding bits in both operands are 1, a binary 1 MUST be returned. If not, a binary 0 MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.10

49.06.38

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/bitwise_XOR.xhtml

Assertion: When the binary bitwise XOR operator ' ^ ' is applied on two operands A and B, then a bitwise comparision must be done on each corresponding bit position of the integer forms of A and B. For each bit position, if only one of the corresponding bits in the two operands is 1, a binary 1 MUST be returned. If not, a binary 0 MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.10

49.06.39

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/bitwise_OR.xhtml

Assertion: When the binary bitwise OR operator '!' is applied on two operands A and B then a bitwise comparision must be done on each corresponding bit position of the integer forms of A and B. For each bit position, if either one or both of the corresponding bits in the two operands is 1, a binary 1 MUST be returned. If not, a binary 0 MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.10

49.06.40

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/logical_AND1.xhtml

Assertion: When logical AND operator '&&' is applied on two expressions M and N, where expression M evaluates to a boolean false, then the return value MUST be boolean false.

Assertion: When logical AND operator '&&' is applied on two expressions M and N, where expression M evaluates to a boolean true and expression N evaluates to a boolean value, then the return value MUST be boolean value of the expression N.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.11

49.06.41

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/logical_AND2.xhtml

Assertion: When logical AND operator '&&' is applied on two expressions M and N, where expression M evaluates to a boolean true and expression N evaluates to a non-boolean value, then the return value MUST be the value of expression N.

Assertion: When logical AND operator '&&' is applied on two expressions M and N, where both expressions M and N evaluate to non-boolean values, then the return value MUST be the value of expression N.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.11

49.06.42

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/logical_OR1.xhtml

Assertion: When logical OR operator '||' is applied on two expressions M and N, where expression M evaluates to a boolean true, then the return value MUST be boolean true.

Assertion: When logical OR operator '||' is applied on two expressions M and N, where expression M evaluates to a non-boolean value, then the return value MUST be the value of the expression M.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.11

49.06.43

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/logical_OR2.xhtml

Assertion: When logical OR operator '||' is applied on two expressions M and N, where expression M evaluates to a boolean false and the expression N evaluates to a boolean value, then the return value MUST be boolean value of the expression N.

Assertion: When logical OR operator '||' is applied on two expressions M and N, where expression M evaluates to a boolean false and the expression N evaluates to a non-boolean value, then the return value MUST be the value of the expression N.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.11

49.06.44

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/ternary.xhtml

Assertion: When the conditional expression **(** C **?** M **:** N**)** is used, where the condition C evaluates to boolean true, then the return value MUST be the result of evaluating expression M.

Assertion: When the conditional expression **(** C **?** M **:** N**)** is used, where the condition C evaluates to boolean false, then the return value MUST be the result of evaluating expression N.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.11

49.06.45

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/simple_assign.xhtml

Assertion: When the assignment operator '**=**' is used with two operands M and N (eg: M = N), the the result of evaluating expression N MUST be assigned as value to the operand M.

Assertion: When the assignment operator '**=**' is used with more than two operands (eg: M = N = ……= P = R), the the result of evaluating right most expression R MUST be assigned as value to each of the variables M, N, P,…

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.13.1

49.06.46

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/comma_operator.xhtml

Assertion: When the comma operator ',' is used with more than one expression, then each of the expression starting from left to right MUST be evaluated and the evaluated value of the last expression MUST be returned.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.14

49.06.47

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/compound_assign1.xhtml

Assertion: When the compound assignment operator 'op=' is used with two operands M and N (where 'op' is any one of the: * / % + -), then the result of evaluating the operands M and N with the operator 'op' MUST be assigned to the operand M.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.13.2

49.06.48

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/compound_assign2.xhtml

Assertion: When the compound assignment operator 'op=' is used with two operands M and N (where 'op' is any one of the: << >> >>> & ^ |), then the result of evaluating the operands M and N with the operator 'op' MUST be assigned to the operand M.

Traceability:

[ESMP]	Section 5.5.2
[ECMA262]	Section 11.13.2

49.06.49

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/precedence1.xhtml

Assertion: When an expression contains more than one operator, then the operator precedence MUST be as given in following table. The operators listed within a row have same precedence. The operators given in each row have higher precedence than the operators given in the next row.

Operator	Description
. [] ()	Dot notation, array indexing, function calls and expression grouping
++ -- + - ~ ! delete new typeof void	Unary operators
* / %	Multiplication, division, remainder
+ - +	Addition, substraction, string concatenation
<< >> >>>	Bit wise shift operators
< <= > => instanceof	Relational operators
== != === !==	Equality operators
&	Bitwise AND
^	Bitwise XOR
|	Bitwise OR
&&	Logical AND
||	Logical OR
?:	Conditional operator
= op=	Assignment operators
,	Comma operator

Assertion: When an expression contains more than one operator with equal precedence, then they MUST be evaluated from left to right, in the order they appear.

Traceability:

[ESMP]	Section 5.5.2

6.3.3 Grammer and Syntax

49.07.01

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/if_else1.xhtml

Assertion: When the “**if** (expression) statements;” is encountered, where ‘expression’ evaluates a Boolean false, then the "statements" MUST not be executed.

Assertion: When the “**if** (expression) statements;” is encountered, where ‘expression’ evaluates a Boolean true, then the ‘statements’ MUST be executed .

Assertion: When the “**if** (expression) statements1 **else** statements2;” is encountered, where ‘expression’ evaluates a Boolean true, then the ‘statements1’ MUST be executed.

Assertion: When the “**if** (expression) statements1 **else** statements2;” is encountered, where ‘expression’ evaluates a Boolean false then the ‘statements2’ MUST be executed.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.5

49.07.02

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/do_while1.xhtml

Assertion: When the "**do** Statements **while** (Expression);" is encountered, then the "Statements" MUST be executed first, then "Expression" MUST be evaluated. If the "Expression" evaluates to Boolean true, then the "Statements" MUST be executed again, and this cycle MUST be repeated as long as "Expression" evaluates to true.

Assertion: When the "**do** Statements **while** (Expression);" is encountered, then the "Statements" MUST be executed first, then "Expression" MUST be evaluated. If the "Expression" evaluates to Boolean false, then the control MUST be returned to the current scope.

Assertion: When the "**do** Statements **while** (Expression);" is encountered, then the "Statements" MUST be executed first, then "Expression" MUST be evaluated. If the "Expression" evaluates to Boolean true, then the "Statements" MUST be executed again, and this cycle MUST be repeated as long as "Expression" evaluates to true. When the "Expression" evaluates to false, then the control MUST be returned to the current scope.

Assertion: In a "**do** Statements **while** (Expression);" block, while the Statements are being executed, if a **continue** statement occurs without a label, then the rest of the statements in the block MUST be skipped in the current loop. The "Expression" MUST be evaluated, and if it evaluates to Boolean true, the execution of "Statements" MUST start again.

Assertion: In a "**do** Statements **while** (Expression);" block, while the Statements are being executed, if a **break** statement occurs without a label, then the control MUST be passed to the statement immediately after the current do-while block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.1

49.07.03

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/do_while_label1.xhtml

Assertion: In a "**do** Statements **while** (Expression);" block, while the Statements are being executed, if a **continue** statement occurs with a label, then the execution of the labeled statements MUST start.

Assertion: In a "**do** Statements **while** (Expression);" block, while the Statements are being executed, if a **break** statement occurs with a label, then the control MUST be passed to the statement immediately after the labeled block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.1

49.07.04

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/while1.xhtml

Assertion: When a "**while** (Expression) Statements;" block is encountered, the "Expression" MUST be evaluated first, and if the evaluation results in Boolean true, then the "Statements" MUST be executed. Again the "Expression" MUST be evaluated and "Statements" MUST be executed. This cycle MUST continue, till the "Expression" evaluates to Boolean false, at which stage, the control MUST be passed to the statement immediately after the current **while** block.

Assertion: When a "**while** (Expression) Statements;" block is encountered, the "Expression" MUST be evaluated first, and if the evaluation results in boolean false, then the control MUST be passed to the statement immediately after the current **while** block.

Assertion: In a "**while** (Expression) Statements;" block, while the "Statements" are being executed, if a **continue** statement without a label occurs, then the rest of the statements in the block MUST be skipped in the current loop. The "Expression" MUST be evaluated, and if it evaluates to Boolean true, the execution of "Statements" MUST start again.

Assertion: In a "**while** (Expression) Statements;" block, while the "Statements" are being executed, if a **break** statement without a label occurs, then the control MUST be passed to the statement immediately after the current **while** block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.2

49.07.05

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/while_label1.xhtml

Assertion: In a "**while** (Expression) Statements;" block, while the "Statements" are being executed, if a **continue** statement with a label occurs, then the control MUST be passed to the labeled block.

Assertion: In a "**while** (Expression) Statements;" block, while the "Statements" are being executed, if a **break** statement with a label occurs, then the control MUST be passed to the statement immediately after the labeled block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.2

49.07.06

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008
Test path: esmp_Sec5/5.5/for1.xhtml

Assertion: When the "**for** (iniExprn; Condn; incrExprn) Statements;" block is encountered, then the expression "iniExprn" MUST be executed first, then "Condn" MUST be evaluated. If the "Condn" evaluates to Boolean true, then the "Statements" MUST be executed. Then the expression "incrExprn" MUST be executed. Then if the "Condn" evaluates to true, again the "Statements" and the "incrExprn" MUST be executed. This cycle of evaluating the "Condn" and executing the "Statements" and the "incrExprn" MUST be repeated, until the "Condn" evaluates to false. Then the control MUST be passed to the statement, immediately after the current **for** block.

Assertion: When the "**for** (iniExprn; Condn; incrExprn) Statements;" block is being executed, whenever the "Condn" evaluates to Boolean false, the control MUST be passed to the statement immediately after the **for** block.

Assertion: While the "Statements" of a **for** loop are being executed, if a **continue** statement without a label occurs, then the rest of the statements in the current loop MUST be skipped and control MUST be passed to the "condition" part of the **for** loop.

Assertion: While the "Statements" of a **for** loop are being executed, if a **break** statement without a label occurs, then the control MUST be passed to the statement immediately after the current **for** block.

Assertion: When the "**for** (**var** varDeclrn; Condn; incrExprn) Statements;" block is encountered, then the expression "varDeclrn" MUST be evaluated first as per variables declaration rules. Then "Condn" MUST be evaluated. If the "Condn" evaluates to Boolean true, then the "Statements" MUST be executed. Then the expression "incrExprn" MUST be executed. Then if the "Condn" evaluates to true, again the "Statements" and the "incrExprn" MUST be executed. This cycle of evaluating the "Condn" and executing the "Statements" and the "incrExprn" MUST be repeated, until the "Condn" evaluates to false. Then the control MUST be passed to the statement, immediately after the current **for** block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.3

49.07.07

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for2.xhtml

Assertion: When the "**for** (; Condn; incrExprn) Statements;" block is encountered, then "Condn" MUST be evaluated. If the "Condn" evaluates to Boolean true, then the "Statements" MUST be executed. Then the expression "incrExprn" MUST be executed. Then if the "Condn" evaluates to true, again the "Statements" and the "incrExprn" MUST be executed. This cycle of evaluating the "Condn" and executing the "Statements" and the "incrExprn" MUST be repeated, until the "Condn" evaluates to false. Then the control MUST be passed to the statement, immediately after the current **for** block.

Assertion: When the "**for** (iniExprn; ; incrExprn) Statements;" block is encountered, then the expression "iniExprn" MUST be executed first, then the "Statements" MUST be executed. Then the expression "incrExprn" MUST be executed. Then again, the "Statements" and the "incrExprn" MUST be executed. This cycle of evaluating the "Condn" and executing the "Statements" and the "incrExprn" MUST be repeated until some exit condition is met.

Assertion: When the "**for** (iniExprn; Condn;) Statements;" is encountered, then the expression "iniExprn" MUST be executed first, then "Condn" MUST be evaluated. If the "Condn" evaluates to Boolean true, then the "Statements" MUST be executed. Then if the "Condn" evaluates to true, again the "Statements" MUST be executed. This cycle of evaluating the "Condn" and executing the "Statements" MUST be repeated, until the "Condn" evaluates to false. Then the control MUST be passed to the statement, immediately after the current **for** block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.3

49.07.08

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for3.xhtml

Assertion: When the "**for** (iniExprn; ;) Statements;" block is encountered, then the expression "iniExprn" MUST be executed first. Then the "Statements" block MUST be executed repeatedly until some exit condition is met.

Assertion: When the "**for** (; Condn;) Statements;" block is encountered, then "Condn" MUST be evaluated first. If the "Condn" evaluates to Boolean true, then the "Statements" MUST be executed. This cycle of evaluating the "Condn" and executing the "Statements" MUST be repeated, until the "Condn" evaluates to false. Then the control MUST be passed to the statement, immediately after the current **for** block.

Assertion: When the "**for** (;; incrExprn) Statements;" is encountered, then "Statements" MUST be executed. Then the expression "incrExprn" MUST be executed. Then again, the "Statements" and the "incrExprn" MUST be executed. This cycle of executing the "Statements" and the "incrExprn" MUST be repeated, until some exit condition is met.

Assertion: When the "**for** (; ;) Statements;" block is encountered, then "Statements" block MUST be executed repeatedly until some exit condition is met.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.3

49.07.09

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_label1.xhtml

Assertion: While the "Statements" of a **for** loop are being executed, if a **continue** statement with a label occurs, then control MUST be passed to the labeled block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.3

49.07.10

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_label2.xhtml

Assertion: While the "Statements" of a **for** loop are being executed, if a **break** statement with a label occurs, then control MUST be passed to the statement immediately after the labeled block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.3

49.07.11

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in0.xhtml

Assertion: When the "**for** (someVariable **in** someObject) Statements;" block is encountered, where there are no enumerable properties in the 'someObject' , then the control MUST be returned to the statement immediately after the current **for** block, with a empty return value.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.12

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in1.xhtml

Assertion: When the "**for** (someVariable **in** someObject) Statements;" block is encountered, where there are N number of enumerable properties in the object 'someObject, then the 'Statements' MUST be executed N number of times. Every time, before starting the loop (say, n^{th} loop where n= 1 to N), the variable 'someVariable' MUST be set with the name of the n^{th} enumerable property of the object 'someObject'.

Assertion: When the "**for** (someVariable **in** someObject) Statements;" block is encountered, where there are N number of enumerable properties in the object 'someObject, then the 'Statements' MUST be executed N number of times. Before starting the n^{th} loop (where n= 1 to N), the variable 'someVariable' MUST be assigned the name of the n^{th} enumerable property of the object 'someObject'. At the end of the N loops, the control MUST be passed to the statement immediately after the **for** loop.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.13

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in2.xhtml

Assertion: While executing a loop of the "**for** (someVariable **in** someObject) Statements;" if a **continue** statement without a label is encountered, and if there is at least one more enumerable property in the 'someObject', then the rest of the statements in the current loop MUST be skipped, and the next loop MUST start with the value of the 'someVariable' reset to the name of the next enumerable property.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.14

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in3.xhtml

Assertion: While executing a loop of the "**for** (someVariable **in** someObject) Statements;" if a **continue** statement without a label is encountered, and if there are no more enumerable properties in the 'someObject', then the control MUST be passed to the statement immediately after the **for** loop.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.15

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in4.xhtml

Assertion: While executing a loop of the "**for** (someVariable **in** someObject) Statements;", if a **break** statement without a label occurs, then the control MUST be passed to the statement immediately after the current **for** block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.16

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in5.xhtml

Assertion: When the "**for** (**var** varDecln **in** someObject) Statements;" block is encountered, where there are N number of enumerable properties in the object 'someObject, then the 'Statements' MUST be executed N number of times. Every time, before starting the loop (say, n^th^ loop where n= 1 to N), the 'varDecln' MUST be evaluated as per variable declaration rules and MUST be set with the name of the n^th^ enumerable property of the object 'someObject'.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.17

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in6.xhtml

Assertion: While executing a loop of the "**for** (someVariable **in** someObject) Statements;" block, if any of the unvisited properties is deleted from the current object, then the deleted property MUST not be visited in the remaining loops.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.18

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in7.xhtml

Assertion: While executing a loop of the "**for** (someVariable **in** someObject) Statements;" block, if a new property is added to the current object, then the new property is not guaranteed to be visited in the remaining loops of the active enumeration.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.19

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in_label1.xhtml

Assertion: While executing a loop of the "**for** (someVariable **in** someObject) Statements;" if a **continue** statement with a label is encountered, then the control MUST be passed to the labeled block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.20

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/for_in_label2.xhtml

Assertion: While executing a loop of the "**for** (someVariable **in** someObject) Statements if a **break** statement with a label occurs, then control MUST be passed to the statement immediately after the labeled block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.6.4

49.07.21

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/continue1.xhtml

Assertion: It is syntactically incorrect to use a **continue** statement without a label at any other place in a program, other than nesting directly or indirectly in a **for** loop or in a **do-while** loop or in a **while** loop.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.7

49.07.22

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/continue2.xhtml

Assertion: It is syntactically incorrect to use a **continue** statement with a label in a **for** loop or **do-while** loop or in **while** loop, if the label does not appear in the label set of the enclosing iteration statement.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.7

49.07.23

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/break1.xhtml

Assertion: It is syntactically incorrect to use a **break** statement without a label at any other place in a program, other than nesting directly or indirectly in a **for** loop or in a **do-while** loop or in a **while** loop.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.8

49.07.24

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/break2.xhtml

Assertion: It is syntactically incorrect to use a **break** statement with a label in a **for** loop or **do-while** loop or in **while** loop, if the label does not appear in the label set of the enclosing iteration statement.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.8

49.07.25

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/return1.xhtml

Assertion: The **return;** statement in side a function body MUST cause the function to cease execution and return the value **undefined** to the caller.

Assertion: The **return Expression;** statement in side a function body MUST cause the function to cease execution and return the value of the Expression to the caller.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.9

49.07.26

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/return2.xhtml

Assertion: In a **return Expression;** statement, line-break character MUST not appear between the token **return** and the expression following it.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.9

49.07.27

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/return3.xhtml

Assertion: It is syntactically incorrect to use a **return** statement out side a function body.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.9

49.07.28

Requirement: Optional
SCR: ESMP-LANGUAGE-C-012

Test path: esmp_Sec5/5.5/with_support1.xhtml

Assertion: The 'with' statement MUST add a computed object to the front of the scope chain of the current execution context, then MUST execute a statement with this augmented scope chain, and then MUST restore the scope chain.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.10

49.07.29

Requirement: Mandatory.
SCR: ESMP-LANGUAGE-C-012

Test path: esmp_Sec5/5.5/with_no_support1.xhtml

Assertion: When the implementation does not support the 'with' statement, then using 'with' statement MUST result in syntax error.

Traceability:

[ESMP]	Section 5.5.3
[ECMA327]	Section 5.3

49.07.30

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/switch1.xhtml

Assertion: When the "**switch** (Expression) CaseBlock;" block is encountered, then the "Expression" MUST be evaluated first and its value MUST be passed as parameter to "CaseBlock". When the CaseBlock contains one or more CaseClauses and a DefaultClause, then the CaseClauses MUST be evaluated in the order they appear.

Assertion: When a CaseClause of a **switch** block is evaluated, the associated expression of the **case** statement MUST be evaluated and its value MUST be compared to the parameter P. When the comparison results in Boolean false, then the statements associated with this CaseClause MUST not be executed.

Assertion: When a CaseClause of a **switch** block is evaluated, the associated expression of the **case** statement MUST be evaluated and its value MUST be compared to the parameter P. When the comparison results in Boolean true, then the statements associated with this CaseClause MUST be executed.

Assertion: After evaluating the CaseBlock of a **switch** block, the control MUST be passed to the statement immediately after the **switch** block.

Assertion: While the statements associated with a CaseClause or defaultClause of a **switch** block are being executed, if a **break** statement occurs, then the control MUST be passed to the statement immediately after the **switch** block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.11

49.07.31

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/switch2.xhtml

Assertion: When the "**switch** (Expression) CaseBlock;" block is encountered, then the "Expression" MUST be evaluated first and its value MUST be passed as parameter to "CaseBlock". When the CaseBlock contains only a DefaultClause, then the statements in the DefaultClause MUST be executed, and then control MUST be passed to the statement after the **switch** block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.11

49.07.32

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/switch3.xhtml

Assertion: When the "**switch** (Expression) CaseBlock;" block is encountered, then the "Expression" MUST be evaluated first and its value MUST be passed as parameter to "CaseBlock". When the CaseBlock contains one or more CaseClauses and a DefaultClause, then the CaseClauses MUST be evaluated in the order they appear. But, the DefaultClause MUST be evaluated, only after evaluating all of the CaseClauses, irrespective of the position of the DefaultClause in the source code.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.11

49.07.33

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/label1.xhtml

Assertion: Any statement can be prefixed by a label using the syntax "label_identifier **:** statement"

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.12

49.07.34

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/label2.xhtml

Assertion: It is syntactically incorrect to use a labeled statement, which is enclosed by another labeled statement, where both the labeled statements have same label.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.12

49.07.35

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/label3.xhtml

Assertion: A labeled statement inside a function body can have same label, as the another labeled statement outside the function body, with which it is directly or indirectly nested.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.12

49.07.36

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/try1.xhtml

Assertion: When a "**throw** Expression**;"** statement appears in a block, then the Expression MUST be evaluated, and its value MUST be returned as an exception.

Assertion: When a block of code enclosed in a **try** statement is evaluated, where the evaluation generates an exception (either by a run-time error or by a **throw** statement), then the control MUST be passed to a **catch** block, if it exists, with the exception as the return value.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.14

49.07.37

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/try2.xhtml

Assertion: When a block of code enclosed in a **try** statement is evaluated, where the evaluation does not generate any exceptions (either by a run-time error or by a **throw** statement), then the control MUST be passed to a **finally** block, if it exists.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.14

49.07.38

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/try3.xhtml

Assertion: When the statements of a "**catch(**identifier**)** Block" are evaluated, and if no error occurs in this evaluation, then control MUST be passed to **finally** block, if it exists.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.14

49.07.39

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/try4.xhtml

Assertion: When an exception is generated in a **catch** block or **finally** block, then the exception MUST be propagated to the outer **try** block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.14

49.07.40

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/try5.xhtml

Assertion: When any exception thrown in an inner **try** block is not caught in the inner **catch** block, then the exception MUST be propagated to the outer **try** block.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 12.14

49.07.41

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/function1.xhtml

Assertion: A function can be declared any where in the source code, irrespective of the location of its calling statements.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 13

49.07.42

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/function2.xhtml

Assertion: A function can be declared any where in the source code, irrespective of the location of its calling statements.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 13

49.07.43

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/function3.xhtml

Assertion: A function body can contain a call to the same function (making it a recursive function).

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 13

49.07.44

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/function4.xhtml

Assertion: A function can be nested within another function's body.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 13

49.07.45

Requirement: Mandatory
SCR: ESMP-LANGUAGE-C-008

Test path: esmp_Sec5/5.5/function5.xhtml

Assertion: When an array or object is passed to a function as an argument, then it MUST be passed only by reference.

Assertion: When an object is passed to a function as an argument, and if any of the properties of are modified in side the function, then the change MUST reflect in the object outside the function.

Traceability:

[ESMP]	Section 5.5.3
[ECMA262]	Section 13

7. NATIVE (BUILT-IN) OBJECTS

7.1 Global Object (Parent Object)

7.1.1 Version History

49.08.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-LANGUAGE-C-014

Test path: esmp_Sec6/6.3_global/verison.xhtml

Assertion: The Global object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.3.1, Sec 6.2.1

Verification: Load the page. Verify that the version of the Global object is displayed in the form “M.m.I.i”.

7.1.2 Properties

49.08.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/no_constructor.xhtml

Assertion: There is no constructor property for the global object, hence the call to the constructor "new Global()" MUST throw the **TypeError** exception.

Traceability:

[ESMP]	Section 6.3.2

49.08.03

Requirement: Mandatory

SCR: -013

Test path: esmp_Sec6/6.3_global/global1.xhtml

Assertion 1: The unique **global** object MUST be instantiated prior to entering any execution context.

Assertion 2: All objects MUST be the properties of **global** object, which can be referred as 'self'.

Assertion 3: All objects MUST be the properties of **global** object, which can be referred as 'top'.

Assertion 4: All objects MUST be the properties of **global** object, which can be referred as 'parent'.

Assertion 5: All objects MUST be the properties of **global** object, which can be referred as 'window'.

Traceability:

[ESMP]	Section 6.1
[ECMA262]	Section 10.1.5, Section 15.1

7.1.3 Methods

7.1.3.1 encodeURI()

49.08.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/encodeURI1.xhtml

Assertion: When the method **encodeURI()** of the **global** object is called, where a complete URI is passed as an argument, then the method MUST return a URI string, such that the delimiting reserved characters in the input URI are not escaped.

Traceability:

[ESMP]	Section 6.3.3.1
[ECMA262]	Section 15.1.3.3
[RFC2396]	Section 2.2

49.08.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/encodeURI2.xhtml

Assertion: When the method **encodeURI()** of the **global** object is called, where a complete URI is passed as an argument, then the method MUST return a URI string, such that the delimiting reserved characters in the input URI are not escaped.

Traceability:

[ESMP]	Section 6.3.3.1
[ECMA262]	Section 15.1.3.3
[RFC2396]	Section 2.2

49.08.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/encodeURI3.xhtml

Assertion: When the method **encodeURI()** of the **global** object is called, where a complete URI is passed as an argument, then the method MUST return a URI string, such that the un- reserved punctuation characters (hyphen, underscore, dot, exclamation, tilde, asterisk, single quote, parentheses characters) in the input URI are not escaped.

Traceability:

[ESMP]	Section 6.3.3.1
[ECMA262]	Section 15.1.3.3
[RFC2396]	Section 2.3

49.08.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/encodeURI4.xhtml

Assertion: When the method **encodeURI()** of the **global** object is called, where a complete URI is passed as an argument, then the method MUST return a URI string, such that the un-safe characters (angle brackets, hash, double quote, percent characters) in the input URI are escaped.

Traceability:

[ESMP]	Section 6.3.3.1
[ECMA262]	Section 15.1.3.3
[RFC2396]	Section 2.4.3

49.08.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/encodeURI5.xhtml

Assertion: When the method **encodeURI()** of the **global** object is called, where a complete URI is passed as an argument, then the method MUST return a URI string, such that the un-wise delimiting characters (curly braces, pipe, backward slash, carat, square brackets and back-apostrophe character) in the input URI are escaped.

Traceability:

[ESMP]	Section 6.3.3.1
[ECMA262]	Section 15.1.3.3
[RFC2396]	Section 2.4.3

7.1.3.2 encodeURIComponent()

49.08.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/encodeURIComponent1.xhtml

Assertion: When the **encodeURIComponent()** method of the global object is called, where a URI fragment is passed as an argument, then the method MUST return an UTF-8 encoded string, with appropriate characters encoded into escaped sequences as defined in the RFC2396 including the URI separator characters.

Traceability:

[ESMP]	Section 6.3.3.2
[ECMA262]	Section 15.1.3.4
[RFC2396]	Section 2

49.08.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/encodeURIComponent2.xhtml

Assertion: When the **encodeURIComponent()** method of the global object is called, where a URI fragment is passed as an argument, then the method MUST return an UTF-8 encoded string, with appropriate characters encoded into escaped sequences as defined in the RFC2396 including the URI separator characters.

Traceability:

[ESMP]	Section 6.3.3.2
[ECMA262]	Section 15.1.3.4
[RFC2396]	Section 2

7.1.3.3 decodeURI()

49.08.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/decodeURI1.xhtml

Assertion: When the method **decodeURI()** of the global object is called, where an encoded URI string is passed as an argument, then the method MUST return the decoded URI string, such that the delimiting reserve characters in the input URI are not decoded.

Traceability:

[ESMP]	Section 6.3.3.3
[ECMA262]	Section 15.1.3.1
[RFC2396]	Section 2.2

49.08.12

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/decodeURI2.xhtml

Assertion: When the method **decodeURI()** of the global object is called, where an encoded URI string is passed as an argument, then the method MUST return the decoded URI string, such that the unreserved puctuation marks in the input URI are not decoded into any other character.

Traceability:

[ESMP]	Section 6.3.3.3
[ECMA262]	Section 15.1.3.1
[RFC2396]	Section 2.4.3

49.08.13

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/decodeURI3.xhtml

Assertion: When the method **decodeURI()** of the global object is called, where an encoded URI string is passed as an argument, then the method MUST return the decoded URI string, such that the escaped delimiter characters and space characters in the input URI are decoded into their original form.

Traceability:

[ESMP]	Section 6.3.3.3
[ECMA262]	Section 15.1.3.1
[RFC2396]	Section 2.4.3

49.08.14

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/decodeURI4.xhtml

Assertion: When the method **decodeURI()** of the global object is called, where an encoded URI string is passed as an argument, then the method MUST return the decoded URI string, such that the escaped unwise characters in the input URI are decoded into their original form.

Traceability:

[ESMP]	Section 6.3.3.3
[ECMA262]	Section 15.1.3.1
[RFC2396]	Section 2.4.3

49.08.15

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/decodeURI_Error1.xhtml

Assertion: When the method **decodeURI()** of the global object is called, where an encoded URI string is passed as an argument and if the code points defined by %XXX are out of the bounds, then the compiler MUST throw the URIError exception.

Traceability:

[ESMP]	Section 6.3.3.3
[ECMA262]	Section 15.1.3.1

7.1.3.4 decodeURIComponent()

49.08.16

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/decodeURIComponent1.xhtml

Assertion: When the method **decodeURIComponent()** of the global object is called, where a string containing UTF-8 encoded octets is passed as an argument, then the method MUST return the string after re-storing the UTF-8 encoded octets with the characters they represent.

Traceability:

[ESMP]	Section 6.3.3.4
[ECMA262]	Section 15.1.3.2

49.08.17

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/decodeURIComponent2.xhtml

Assertion: When the method **decodeURIComponent()** of the global object is called, where a string containing UTF-8 encoded octets is passed as an argument, then the method MUST return the string after re-storing the UTF-8 encoded octets with the characters they represent.

Traceability:

[ESMP]	Section 6.3.3.4
[ECMA262]	Section 15.1.3.2

49.08.18

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/decodeURIComponent_Error1.xhtml

Assertion: When the method **decodeURIComponent()** of the global object is called, where a string containing UTF-8 encoded octets is passed as an argument and if the code points defined by %XXX are out of the bounds, then the compiler MUST throw the URIError exception.

Traceability:

[ESMP]	Section 6.3.3.4
[ECMA262]	Section 15.1.3.2

7.1.3.5 isFinite()

49.08.19

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/isFinite1.xhtml

Assertion: When the method **isFinite()** of the global object is called, where the argument passed is a valid number, then the method MUST return Boolean true.

Traceability:

[ESMP]	Section 6.3.3.5
[ECMA262]	Section 15.1.2.5

49.08.20

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/isFinite2.xhtml

Assertion: When the method **isFinite()** of the global object is called, where the argument passed is a value which can be converted into a valid number, then the method MUST return Boolean true.

Traceability:

[ESMP]	Section 6.3.3.5
[ECMA262]	Section 15.1.2.5

49.08.21

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/isFinite3.xhtml

Assertion: When the method **isFinite()** of the global object is called, where the argument passed is a not a number, then the method MUST return Boolean false.

Traceability:

[ESMP]	Section 6.3.3.5
[ECMA262]	Section 15.1.2.5

49.08.22

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/isFinite4.xhtml

Assertion 1: When the method **isFinite()** of the global object is called, where the argument passed is a positive infinity number, then the method MUST return Boolean false.

Assertion 2: When the method **isFinite()** of the global object is called, where the argument passed is a negative infinity number, then the method MUST return Boolean false.

Traceability:

[ESMP]	Section 6.3.3.5
[ECMA262]	Section 15.1.2.5

7.1.3.6 isNaN()

49.08.23

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/isNaN1.xhtml

Assertion 1: When the method **isNaN()** of the global object is called, where the argument passed is not a number, then the method MUST return Boolean true.

Assertion 2: When the method **isNaN()** of the global object is called, where the argument passed is not a number primitive, then the method MUST return Boolean true.

Assertion 3: When the method **isNaN()** of the global object is called, where the argument passed is undefined, then the method MUST return Boolean true.

Traceability:

[ESMP]	Section 6.3.3.6
[ECMA262]	Section 15.1.2.4

49.08.24

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/isNaN2.xhtml

Assertion 1: When the method **isNaN()** of the global object is called, where the argument passed is a valid number, then the method MUST return Boolean false.

Assertion 2: When the method **isNaN()** of the global object is called, where the argument passed is a positive inifinity number, then the method MUST return Boolean false.

Assertion 3: When the method **isNaN()** of the global object is called, where the argument passed is a negative infinity number, then the method MUST return Boolean false.

Assertion 4: When the method **isNaN()** of the global object is called, where the argument passed is a Boolean true, then the method MUST return Boolean false.

Assertion 5: When the method **isNaN()** of the global object is called, where the argument passed is a Boolean false, then the method MUST return Boolean false.

Assertion 6: When the method **isNaN()** of the global object is called, where the argument passed is null, then the method MUST return Boolean false.

Traceability:

[ESMP]	Section 6.3.3.6
[ECMA262]	Section 15.1.2.4

7.1.3.7 parseInt()

49.08.25

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt1.xhtml

Assertion 1: When the method **parseInt()** of the global object is called, where an integer number is passed as the argument, then the method MUST return the same integer value.

Assertion 2: When the method **parseInt()** of the global object is called, where a string containing an integer number is passed as the argument, then the method MUST parse the string S into the integer and return this value.

Assertion 3: When the method **parseInt()** of the global object is called, where a float number is passed as the argument, then the method MUST return only the integer portion of the float number as an integer.

Assertion 4: When the method **parseInt()** of the global object is called, where a string containing float number is passed as the argument, then the method MUST return only the integer portion of the float number as an integer.

Assertion 5: When the method **parseInt()** of the global object is called, where a string containing leading whitespace followed by digit 1-9 and any other digits is passed as the argument, then the method MUST ignore the leading white space and return only the integer portion of the digits.

Assertion 6: When the method **parseInt()** of the global object is called, where a string-containing digit 1-9 followed by any other digits followed by any other characters is passed as the argument, then the method MUST return only the first occurred integer characters ignoring the rest of the characters including white space or dot character.

Assertion 7: When the method **parseInt()** of the global object is called, where a string S containing a minus sign immediately followed by digits and then any other characters is passed as the argument, then the method MUST return only the first occurred integer characters as a negative integer number.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Examples:

```
anInt = parseInt(23);   // anInt = 23

anInt = parseInt("23");   // anInt = 23

anInt = parseInt(23.345);   // anInt = 23

anInt = parseInt("23.345");   // anInt = 23

anInt = parseInt("    23.345");   // anInt = 23

anInt = parseInt("23 . def 34 xy5 zh ");   // anInt = 23

anInt = parseInt(" -23 def 34 xy5 zh ");   // anInt = -23
```

49.08.26

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Assertion: When the method **parseInt()** of the global object is called, where a string containing a plus sign immediately followed by digits and then any other characters is passed as the argument, then the method MUST return only the first occurred integer characters as an unsigned integer number.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Example:

anInt = parseInt(" +23 def 34 xy5 zh "); // anInt = 23

49.08.27

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt2.xhtml

Assertion : When the method **parseInt()** of the global object is called, where a string containing digit 1-9 preceded by any character other than white space, 0, 0x, 0X or plus or minus sign is passed as the argument, then the method MUST return **NaN**

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Examples:

anInt = parseInt(" a23 "); // anInt = NaN

anInt = parseInt(NaN); // anInt = NaN

49.08.28

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt3.xhtml

Assertion: When the method **parseInt()** of the global object is called, where a string containing leading 0x or 0X followed by an hexadecimal number is passed as the argument, then the method MUST return the decimal integer value of the hexadecimal character.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Examples:

anInt = parseInt(" 0x9A4 "); // anInt = 2468

anInt = parseInt(" 0X9A4 "); // anInt = 2468

anInt = parseInt(" 0x9A4M93 "); // 'M' is not a hex char. So, anInt = 2468

49.08.29

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt4.xhtml

Assertion: When the method **parseInt()** of the global object is called, where a string containing leading 0 followed by an octal number is passed as the argument, then the method MUST return the decimal integer value of the octal number.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Example:

```
anInt = parseInt("  04644 ");   // anInt =  2468

anInt = parseInt("  046449");   // '9' not an octal digit, so, anInt =  2468

anInt = parseInt("  0000004644 ");   // anInt =  2468
```

49.08.30

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt5.xhtml

Assertion 1: When the method **parseInt()** of the global object is called, where a string containing leading 0x or 0X followed by an hexadecimal number is passed as the first argument and a radix of value 16 as the second argument, then the method MUST return the integer value of the hexadecimal character.

Assertion 2: When the method **parseInt()** of the global object is called, where a string containing a hexadecimal number without leading 0x or 0X is passed as the first argument and a radix of value 16 as the second argument, then the method MUST return the integer value of the hexadecimal character.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Examples:

```
anInt = parseInt("  0x9A4 ", 16);   // anInt =  2468
anInt = parseInt("  9A4 ", 16);      // anInt =  2468
```

49.08.31

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt6.xhtml

Assertion: When the method **parseInt()** of the global object is called, where a hexadecimal number is passed as the first argument and a radix of value 16 as the second argument, then the method MUST return the integer value of the hexadecimal character.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Examples:

anInt = parseInt(0x9A4, 16);   // anInt = 2468

49.08.32

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt7.xhtml

Assertion: When the method **parseInt()** of the global object is called, where a string containing an octal number is passed as the first argument and a radix of value 8 is passed as the second argument, then the method MUST return the decimal integer value of the octal number.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Example:

anInt = parseInt(" 5230899mdc ", 8); // anInt = 2712

anInt = parseInt(52308, 8); // anInt = 2712

anInt = parseInt(0052308, 8); // anInt = 2712

49.08.33

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt8.xhtml

Assertion: When the method **parseInt()** of the global object is called, where any value is passed as the first argument and a radix of value less than 2 or greater than 32 is passed as the second argument, then the method MUST return **NaN**.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

Example:

```
anInt = parseInt("  83.25 ",  37);   // anInt =  NaN
```

49.08.34

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseInt9.xhtml

Assertion: When the method **parseInt()** of the global object is called, where a string containing any number is passed as the first argument and a radix of value 2 to 36 is passed as the second argument, then the method MUST return the decimal integer value of the parsed string based on the radix supplied.

Traceability:

[ESMP]	Section 6.3.3.7
[ECMA262]	Section 15.1.2.2

7.1.3.8 parseFloat()

49.08.35

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseFloat1.xhtml

Assertion 1: When the method **parseFloat()** of the global object is called, where a float number is passed as an argument, then the method MUST return the same number.

Assertion 2: When the method **parseFloat()** of the global object is called, where a string S containing a float number is passed as an argument, then the method MUST parse the string S into a float value and return this value.

Assertion 3: When the method **parseFloat()** of the global object is called, where a string S containing leading whitespace characters followed by a float number is passed as an argument, then the method MUST parse the string S into a float value and return this value ignoring all the leading whitespace characters.

Assertion 4: When the method **parseFloat()** of the global object is called, where a string S containing a float number followed by any other non-numeric characters is passed as an argument, then the method MUST parse the string S into a float value and return this value ignoring all the non-numeric characters following the float number.

Assertion 5: When the method **parseFloat()** of the global object is called, where a string S containing a number preceded by minus sign is passed as an argument, then the method MUST parse the string S into a float value and return this as a negative float value.

Assertion 6: When the method **parseFloat()** of the global object is called, where a string S containing a number preceded by plus sign is passed as an argument, then the method MUST parse the string S into a float value and return this as an unsigned float value.

Assertion 7: When the method **parseFloat()** of the global object is called, where a string S containing a minus sign, decimal symbol and numbers is passed as an argument, then the method MUST parse the string S into a float value and return this as a negative float value.

Assertion 8: When the method **parseFloat()** of the global object is called, where a string S containing a number in exponential format is passed as an argument, then the method MUST parse the string S into a float value and return this value either as a float value.

Traceability:

[ESMP]	Section 6.3.3.8
[ECMA262]	Section 15.1.2.3

Examples:

afloat = parseFloat(23.54); // afloat = 23.54

afloat = parseFloat("23.54"); // afloat = 23.54

afloat = parseFloat("   23.54"); // afloat = 23.54

afloat = parseFloat(" 23.54xyz123 "); // afloat = 23.54

afloat = parseFloat("   -23.54abc"); // afloat = -23.54

afloat = parseFloat("   +23.54abc"); // afloat = 23.54

afloat = parseFloat("   -.54abc"); // afloat = -0.54

afloat = parseFloat("   23.54e02abc"); // afloat = 2354.0

49.08.36

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-017

Test path: esmp_Sec6/6.3_global/parseFloat2.xhtml

Assertion 1: When the method **parseFloat()** of the global object is called, where a string S containing any leading non-numeric characters is passed as an argument, then the method MUST return NaN.

Assertion 2: When the method **parseFloat()** of the global object is called, where an undefined value is passed as an argument, then the method MUST return NaN.

Assertion 3: When the method **parseFloat()** of the global object is called, where a Null value is passed as an argument, then the method MUST return NaN.

Assertion 4: When the method **parseFloat()** of the global object is called, where a NaN is passed as an argument, then the method MUST return NaN.

Traceability:

[ESMP]	Section 6.3.3.8
[ECMA262]	Section 15.1.2.3

Example:

```
afloat = parseFloat("  adfb23.54 ");   // afloat =  NaN
```

7.1.3.9 toString()

The **toString()** method is inherited by all native object types. The assertions for each of these object types are covered in Section 5.3.3 and in other sections of this document.

7.1.3.10 eval()

49.08.37

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-010, ESMP-LANGUAGE-C-009

Test path: esmp_Sec6/6.3_global/eval_noSupport.xhtml

Assertion: When the method **eval()** of the global object is called, and if the implementation does not support this method, then the compiler MUST return **EvalError** exception.

Traceability:

[ESMP]	Section 6.3.3.10, Section 5.4.2
[ECMA327]	Section 5.1

49.08.38

Requirement: Optional

SCR: ESMP-LANGUAGE-C-017, ESMP-LANGUAGE-C-009

Test path: esmp_Sec6/6.3_global/eval_support.xhtml

Assertion: When the **eval()** method of the global object is called, where a string S is passed as an argument, and if the implementation supports **eval()** method and if the string S contains valid ECMAScript code, then the method MUST evaluate and execute that code.

Traceability:

[ESMP]	Section 6.3.3.10, Section 5.4.2
[ECMA327]	Section 5.1
[ECMA262]	Section 10.1.2

7.2 Array Object

49.09.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConstructor1.xhtml

Assertion: When the constructor **new Array()** is used, without passing any arguments, then an empty array object MUST be created and returned.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.2.1

49.09.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConstructor2.xhtml

Assertion: When the constructor **new Array()** is used, with more than one value are passed as arguments , then an array object MUST be created and returned whose elements are the arguments supplied.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.2.1

49.09.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConstructor3.xhtml

Assertion: When the constructor **new Array()** is used, with one non-numeric value is passed as argument , then an array object MUST be created whose only element is the argument supplied.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.2.1

49.09.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConstructor4.xhtml

Assertion: When the constructor **new Array()** is used, with one integer value is passed as argument , then an array object MUST be created whose length is the argument supplied and whose elements are **undefined**.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.2.2

49.09.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConstructor5.xhtml

Assertion: When the constructor **new Array()** is used, with one numeric value N is passed as argument, where N is not an unsigned integer value, then the implementation **must** raise the **RangeError** exception.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.2.2

49.09.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayFunction1.xhtml

Assertion : When the function **Array()** is called, without passing any arguments, then an empty array object MUST be created and returned.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.1

49.09.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayFunction2.xhtml

Assertion : When the function **Array()** is called, with more than one value are passed as arguments , then an array object MUST be created and returned whose elements are the arguments supplied.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.1

49.09.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayFunction3.xhtml

Assertion : When the function **Array()** is called, with one non-numeric value is passed as argument , then an array object MUST be created whose only element is the argument supplied.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.1

49.09.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayFunction4.xhtml

Assertion : When the function **Array()** is called, with one integer value is passed as argument , then an array object MUST be created whose length is the argument supplied and whose elements are **undefined**.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.1

49.09.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayFunction5.xhtml

Assertion : When the function **Array()** is called, with one numeric value N is passed as argument, where N is not an unsigned integer value, then the implementation **must** raise the **RangeError** exception.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 15.4.1

49.09.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayAccess1.xhtml

Assertion: A property in an array can be accessed by using bracket notation, if its name is the array index. The index MUST be used inside the bracket without enclosing quotation marks.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 11.2.1

49.09.12

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayAccess2.xhtml

Assertion : If the name of a property in an array is not an array index, then its value can be accessed by using either dot notation or bracket notation.

Traceability:

[ESMP]	Section 6.4
[ECMA262]	Section 11.2.1

7.2.1 Version History

49.09.13

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-014

Test path: esmp_Sec6/6.4_array/version.xhtml

Assertion: The static **Array** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.4.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

7.2.2 Properties

7.2.2.1 length

49.09.14

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayLength1.xhtml

Assertion: The **length** property of an array object is an integer number representing the number of properties in that array, whose name is an array index.

Traceability:

[ESMP]	Section 6.4.2.1
[ECMA262]	Section 15.4.3, Sec 15.4.5.2

49.09.15

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayLength2.xhtml

Assertion 1: When a new element with index M is added to an array of length N, where M >= N, then the **length** value of the array MUST be set to (M +1).

Assertion 2: When a new element with index M is added to the an array of length N, where M >= N, then the elements with index N to (M-1) MUST be added to array and each of them MUST be set to value **undefined**.

Traceability:

[ESMP]	Section 6.4.2.1
[ECMA262]	Section 15.4

49.09.16

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayLength3.xhtml

Assertion: When the script assigns an integer value M, to the **length** property of an array of N elements, where M < N, then the elements with indices M to (N-1) MUST be automatically deleted from the array.

Traceability:

[ESMP]	Section 6.4.2.1
[ECMA262]	Section 15.4

49.09.17

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayLength4.xhtml

Assertion: When the script assigns an integer value M, to the **length** property of an array of N elements, where M > N, , then the elements with index N to (M-1) MUST be added to array and each of them MUST be set to value **undefined**.

Traceability:

[ESMP]	Section 6.4.2.1
[ECMA262]	Section 15.4

49.09.18

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayLength5.xhtml

Assertion: The **length** property of an empty array MUST be equal to 0.

Traceability:

[ESMP]	Section 6.4.2.1
[ECMA262]	Section 15.4

49.09.19

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayLength6.xhtml

Assertion: The **length** property of an array is not affected by adding or deleting properties, whose names are not array indices.

Traceability:

[ESMP]	Section 6.4.2.1
[ECMA262]	Section 15.4

7.2.3 Methods

7.2.3.1 concat()

49.09.20

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConcat1.xhtml

Assertion: When the **concat()** method of an array object A1 is called, where one or more array objects A2, A3 etc, are passed as arguments, then method MUST return an array R which is a concatenation of the arrays A1, A2, A3 …

Traceability:

[ESMP]	Section 6.4.3.1
[ECMA262]	Section 15.4.4.4

49.09.21

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConcat2.xhtml

Assertion: When the **concat()** method of an array object A1 is called, where one or more array objects A2, A3 etc, are passed as arguments, then method MUST return an array R which is a concatenation of the arrays A1, A2, A3 …

Traceability:

[ESMP]	Section 6.4.3.1
[ECMA262]	Section 15.4.4.4

49.09.22

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConcat3.xhtml

Assertion: When the **concat()** method of an array object A1 is called, where one or more array objects A2, A3 etc, are passed as arguments, then method MUST not affect either the original array object A1 or the arrays passed as arguments .

Traceability:

[ESMP]	Section 6.4.3.1
[ECMA262]	Section 15.4.4.4

49.09.23

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayConcat4.xhtml

Assertion: When the **concat()** method of an array object A1 is called, where no arguments are passed, then method MUST return an array R that is a duplicate of array A1.

Traceability:

[ESMP]	Section 6.4.3.1
[ECMA262]	Section 15.4.4.4

7.2.3.2 join()

49.09.24

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayJoin1.xhtml

Assertion: When the **join()** method of an array object A is called and no arguments are passed, then the method MUST join the contents of the array entries into a single string, delimited by a comma separator and MUST return the resulting string.

Traceability:

[ESMP]	Section 6.4.3.2
[ECMA262]	Section 15.4.4.5

49.09.25

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayJoin2.xhtml

Assertion: When the **join()** method of an array object A is called and no arguments are passed, then the method MUST join the contents of the array entries into a single string, delimited by a comma separator and MUST return the resulting string.

Traceability:

[ESMP]	Section 6.4.3.2
[ECMA262]	Section 15.4.4.5

49.09.26

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayJoin3.xhtml

Assertion: When the **join()** method of an array object A is called and a delimiter string is passed as an argument, then the method MUST join the contents of the array entries into a single string, delimited by the delimiter supplied and MUST return the resulting string.

Traceability:

[ESMP]	Section 6.4.3.2
[ECMA262]	Section 15.4.4.5

49.09.27

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/arrayJoin4.xhtml

Assertion: When the **join()** method of an empty array object A is called, then the method **MUST** return an empty string.

Traceability:

[ESMP]	Section 6.4.3.2
[ECMA262]	Section 15.4.4.5

7.2.3.3 toString()

49.09.28

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_toString1.xhtml

Assertion: When the **toString()** method of an array object A is called, the method MUST join the contents of the array entries into a single string each delimited by a comma separator and MUST return the resulting string.

Traceability:

[ESMP]	Section 6.4.3.3
[ECMA262]	Section 15.4.4.5

49.09.29

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_toString2.xhtml

Assertion: When the **toString()** method of an array object A is called, the method MUST join the contents of the array entries into a single string each delimited by a comma separator and MUST return the resulting string.

Traceability:

[ESMP]	Section 6.4.3.3
[ECMA262]	Section 15.4.4.5

49.09.30

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_toString3.xhtml

Assertion: When the **toString()** method of an array object A is called, where a delimiter string is passed as an argument, then the method MUST join the contents of the array entries into a single string, delimited by the delimiter supplied and MUST return the resulting string.

Traceability:

[ESMP]	Section 6.4.3.3
[ECMA262]	Section 15.4.4.5

49.09.31

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_toString4.xhtml

Assertion: When the **toString()** method of an empty array object A is called, then the method **MUST** return an empty string.

Traceability:

[ESMP]	Section 6.4.3.3
[ECMA262]	Section 15.4.4.5

7.2.3.4 pop()

49.09.32

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_pop1.xhtml

Assertion 1: When the **pop()** method of an array object A is called, the method MUST return the last entry in the array A.

Assertion 2: When the **pop()** method of an array object A is called, the method MUST delete the last entry in the array A, there by reducing its length by 1.

Traceability:

[ESMP]	Section 6.4.3.4
[ECMA262]	Section 15.4.4.6

49.09.33

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_pop2.xhtml

Assertion: When the **pop()** method of an empty array object A is called, then the method MUST return **undefined**.

Traceability:

[ESMP]	Section 6.4.3.4
[ECMA262]	Section 15.4.4.6

7.2.3.5 push()

49.09.34

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_push1.xhtml

Assertion 1: When the **push()** method of an array object A is called, where a value V is passed as an argument, then the value V MUST be appended to the array A as the last entry, thereby increasing the array.length by 1.

Assertion 2: When the **push()** method of an array object A is called, where a value V is passed as an argument, then the value V MUST be appended to the array A as the last entry and MUST return the new length of the array.

Traceability:

[ESMP]	Section 6.4.3.5
[ECMA262]	Section 15.4.4.7

7.2.3.6 reverse()

49.09.35

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_reverse1.xhtml

Assertion: When the **reverse()** method of an array object A is called, then the return value **must** be an array whose elements are exactly in the reverse order of the elements in the original array A.

Traceability:

[ESMP]	Section 6.4.3.6
[ECMA262]	Section 15.4.4.8

49.09.36

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_reverse2.xhtml

Assertion: When the **reverse()** method of an array object A is called, then its elements **must** be re-arranged in the reverse order.

Traceability:

[ESMP]	Section 6.4.3.6
[ECMA262]	Section 15.4.4.8

49.09.37

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_reverse3.xhtml

Assertion: When the **reverse()** method of an empty array object A is called, the method MUST return an empty array.

Traceability:

[ESMP]	Section 6.4.3.6
[ECMA262]	Section 15.4.4.8

7.2.3.7 shift()

49.09.38

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_shift1.xhtml

Assertion 1: When the **shift()** method of an array object A is called, the first value of the array A, MUST be removed from the array, shifting down all the indexes.

Assertion 2: When the **shift()** method of an array object A is called, the method MUST return the first value of the array.

Traceability:

[ESMP]	Section 6.4.3.7
[ECMA262]	Section 15.4.4.9

49.09.39

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_shift2.xhtml

Assertion: When the **shift()** method of an empty array object A is called, the method MUST return **undefined**.

Traceability:

[ESMP]	Section 6.4.3.7
[ECMA262]	Section 15.4.4.9

7.2.3.8 slice()

49.09.40

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice1.xhtml

Assertion: When the **slice()** method of an array object A is called, where an integer N is passed as an argument, then the method MUST return a new array containing elements from array A starting from $(N+1)^{st}$ item to the last item.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.41

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice2.xhtml

Assertion: When the **slice()** method of an array object A is called, where two integers M and N are passed as an argument, then the method MUST return a new array containing elements from array A starting from $(M+1)^{th}$ item upto N^{th} item.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.42

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice3.xhtml

Assertion: When the **slice()** method of an array object A is called, the original array A remains unaffected.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.43

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice4.xhtml

Assertion: When the **slice()** method of an array object A of length L is called, where the start number M passed is a negative number and if (M+L > 0) , then the start number must be taken as (M+L).

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.44

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice5.xhtml

Assertion: When the **slice()** method of an array object A of length L is called, where the start number M passed is a negative number and if (M+L <= 0) , then the start number must be taken as 0.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.45

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice6.xhtml

Assertion: When the **slice()** method of an array object A is called, where two integers M and N are passed as arguments and if N is greater than the array length, then the method MUST return a new array containing elements from array A starting from (M+1)th item to last item.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.46

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice7.xhtml

Assertion: When the **slice()** method of an array object A of length L is called, where the start number M passed is greater than the length L , then an empty array MUST be returned.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.47

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice8.xhtml

Assertion: When the **slice()** method of an array object A of length L is called, where the start number M passed is equal to the length L , then an empty array MUST be returned.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.48

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice9.xhtml

Assertion: When the **slice()** method of an array object A is called, where two integers M and N of equal value are passed as arguments, then an empty array MUST be returned.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.49

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice10.xhtml

Assertion: When the **slice()** method of an array object A is called, where two integers M and N passed as arguments and if M is greater than N, then an empty array MUST be returned.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.50

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice11.xhtml

Assertion: When the **slice()** method of an array object A of length L is called, where two integers M and N are passed as arguments and if the end number N is a negative number such that (N+L > 0) , then the end number N must be taken as (N+L).

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.51

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice12.xhtml

Assertion: When the **slice()** method of an array object A of length L is called, , where two integers M and N are passed as arguments and if the end number N is a negative number such that (N+L < 0) , then an empty array must be returned.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

49.09.52

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_slice13.xhtml

Assertion: When the **slice()** method of an array object A of length L is called, where two integers M and N are passed as arguments and if the end number N is a negative number such that (N+L = 0) , then an empty array must be returned.

Traceability:

[ESMP]	Section 6.4.3.8
[ECMA262]	Section 15.4.4.10

7.2.3.9 sort()

49.09.53

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_sort1.xhtml

Assertion: When the **sort()** method of an array object A is called without passing any arguments, then the method MUST return the initial array A with its elements sorted in the ascending order of their character string values.

Traceability:

[ESMP]	Section 6.4.3.9
[ECMA262]	Section 15.4.4.11

49.09.54

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_sort2.xhtml

Assertion: When the **sort()** method of an array object A is called without passing any arguments, then the method MUST return the initial array A with its elements sorted in the ascending order of their character string values.

Traceability:

[ESMP]	Section 6.4.3.9
[ECMA262]	Section 15.4.4.11

49.09.55

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_sort3.xhtml

Assertion: When the **sort()** method of an array object A is called, where a non-existing function name is passed as an arguments, then the exception **ReferenceError** MUST be thrown.

Traceability:

[ESMP]	Section 6.4.3.9
[ECMA262]	Section 15.4.4.11

49.09.56

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_sort4.xhtml

Assertion: When the **sort()** method of an array object A is called, where a compare function F is passed as argument, where as the compare function F returns a either a negative, zero or positive value when comparing two elements of the source array A, then the **sort()** method MUST return the original array A with its elements sorted in the order defined in the compare function F.

Traceability:

[ESMP]	Section 6.4.3.9
[ECMA262]	Section 15.4.4.11

49.09.57

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_sort5.xhtml

Assertion: When the **sort()** method of an array object A is called, where an incompatible compare function F is passed as argument, then an exception MUST be thrown by the compiler.

Traceability:

[ESMP]	Section 6.4.3.9
[ECMA262]	Section 15.4.4.11

7.2.3.10 splice()

49.09.58

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice1.xhtml

Assertion: When the **splice()** method of an array object A is called, where two numbers M and N representing startIndex and deleteCount are passed as arguments, then the N number of items starting from the (M+1)th item from the original array A must be deleted.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.59

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice2.xhtml

Assertion: When the **splice()** method of an array object A is called, where two numbers M and N representing startIndex, deleteCount are passed as arguments, then the method MUST return a new array object which contains the N number of items starting from the (M+1)th item from the original array A.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.60

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice3.xhtml

Assertion: When the **splice()** method of an array object A is called, where two numbers M and N representing startIndex and deleteCount and one or more additional items are passed as arguments, then the N number of items starting from the (M+1)th item from the original array A must be replaced with the additional items passed as the arguments.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.61

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice4.xhtml

Assertion: When the **splice()** method of an array object A is called, where two numbers M and N representing startIndex and deleteCount and one or more additional items are passed as arguments, then the method MUST return a new array object which contains the N number of items starting from the (M+1)th item from the original array A.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.62

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice5.xhtml

Assertion: When the **splice()** method of an array object A is called, where two numbers M and N representing startIndex and deleteCount and one or more additional items are passed as arguments, then the N number of items starting from the (M+1)th item from the original array A must be replaced with the additional items passed as the arguments.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.63

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice6.xhtml

Assertion: When the **splice()** method of an array object A is called, where two numbers M and N representing startIndex and deleteCount and one or more additional items are passed as arguments, then the method MUST return a new array object which contains the N number of items starting from the $(M+1)^{th}$ item from the original array A.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.64

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice7.xhtml

Assertion: When the **splice()** method of an array object A is called, where only one number M is passed as argument, then array A must remain unaffected.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.65

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice8.xhtml

Assertion: When the **splice()** method of an array object A is called, where only one number M is passed as argument, then the method must return an empty array.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.66

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice9.xhtml

Assertion: When the **splice()** method of an array object A is called, where two values M and N representing startIndex and deleteCount and one or more additional items are passed as arguments, and if N is not convertible into an integer, then the third to last of the arguments supplied must be inserted in the array A, starting at index M.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

49.09.67

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_splice10.xhtml

Assertion: When the **splice()** method of an array object A is called, where two values M and N representing startIndex and deleteCount and one or more additional items are passed as arguments, and if N is convertible into an integer, then the N number of items starting from the $(M+1)^{th}$ item from the original array A must be replaced with the additional items passed as the arguments.

Traceability:

[ESMP]	Section 6.4.3.10
[ECMA262]	Section 15.4.4.12

7.2.3.11 unshift()

49.09.68

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_unshift1..xhtml

Assertion: When the **unshift()** method of an array object A is called, where one or more items are passed as arguments, then the method MUST prepend the items passed to the start of array A in the same order they are listed as the arguments.

Traceability:

[ESMP]	Section 6.4.3.11
[ECMA262]	Section 15.4.4.13

49.09.69

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_unshift2..xhtml

Assertion: When the **unshift()** method of an array object A is called, where one or more items are passed as arguments, then the method MUST prepend the items passed to the start of array A and MUST return the count of the items prepended.

Traceability:

[ESMP]	Section 6.4.3.11
[ECMA262]	Section 15.4.4.13

49.09.70

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_unshift3..xhtml

Assertion: When the **unshift()** method of an array object A is called without passing any arguments, then the original array A MUST remain unaffected.

Traceability:

[ESMP]	Section 6.4.3.11
[ECMA262]	Section 15.4.4.13

49.09.71

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-018

Test path: esmp_Sec6/6.4_array/array_unshift4..xhtml

Assertion: When the **unshift()** method of an array object A is called without passing any arguments, then the method MUST return 0.

Traceability:

[ESMP]	Section 6.4.3.11
[ECMA262]	Section 15.4.4.13

7.3 String Object

49.10.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_function1.xhtml

Assertion: When the **String()** is called as a function, where no argument is passed, then it MUST return an empty string.

Traceability:

[ESMP]	Section 6.5
[ECMA262]	Section 15.5.2

49.10.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_function 2.xhtml

Assertion: When the **String()** is called as a function, where a primitive value is passed as argument, then it MUST perform a type conversion and return a string primitive value as per the following table.

Argument Type	Return string
Undefined	"undefined"
Null	"null"
Boolean true	"true"
Boolean false	"false"
Number +0 or -0	"0"
Number NaN	"NaN"
Number +Infinity	"Infinity"
Number -Infinity	"-Infinity"

Number +/- infinity	"Infinity"
Any other number	String representation of the number.
String	Return the input string, no conversion.

Traceability:

[ESMP]	Section 6.5
[ECMA262]	Section 15.5.2

49.10.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_function3.xhtml

Assertion: When the **String()** is called as a function, where an object is passed as argument, then it MUST return a string primitive value based on the input object as per the following table.

Argument	Return string
Boolean object with value true	"true"
Boolean object with value false	"false"
Number object with value +0 or -0	"0"
Number object with value NaN	"NaN"
Number object with value +Infinity	"Infinity"
Number object with value -Infinity	"-Infinity"
Number object with value +/- infinity	"Infinity"
Any other number object	String representation of the number.
String object	Return the input string, no conversion.

Traceability:

[ESMP]	Section 6.5
[ECMA262]	Section 15.5.2

49.10.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_constructor1.xhtml

Assertion: When the string constructor **new String()** is called, where no arguments are passed, then it MUST instantiate a new String object and set the value property this string object to empty string.

Traceability:

[ESMP]	Section 6.5
[ECMA262]	Section 15.5.2

49.10.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_constructor2.xhtml

Assertion: When the string constructor **new String()** is called, where a primitive value is passed as argument, then it MUST instantiate a new String object and set the value property this string object as per the following table.

Argument Type	Value of the created String object
Undefined	"undefined"
Null	"null"
Boolean true	"true"
Boolean false	"false"
Number +0 or -0	"0"
Number NaN	"NaN"
Number +Infinity	"Infinity"
Number -Infinity	"-Infinity"
Any other number	String representation of the number.
String	Return the input string, no conversion.

Traceability:

[ESMP]	Section 6.5
[ECMA262]	Section 15.5.2

49.10.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_constructor3.xhtml

Assertion: When the string constructor **new String()** is called, where an object is passed as argument, then it MUST instantiate a new String object and set the value property this string object as per the following table.

Argument Type	Value of the created String object
Boolean object with value true	"true"
Boolean object with value false	"false"
Number object with value +0 or -0	"0"
Number object with value NaN	"NaN"
Number object with value +Infinity	"Infinity"
Number object with value -Infinity	"-Infinity"
Any other number object	String representation of the number.
String object	Return the input string, no conversion.

Traceability:

[ESMP]	Section 6.5
[ECMA262]	Section 15.5.2

7.3.1 Version History

49.10.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-014

Test path: esmp_Sec6/6.5_string/ version.xhtml

Assertion: The string object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.5.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

7.3.2 Properties

7.3.2.1 length

49.10.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_length1.xhtml

Assertion: The **length** property of a string object MUST be an integer representing the number of characters in the string object.

Traceability:

[ESMP]	Section 6.5.2.1
[ECMA262]	Section 15.5.4

49.10.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_length2.xhtml

Assertion: Any attempt to write to the read-only property **length** of a string object MUST be ignored.

Traceability:

[ESMP]	Section 6.5.2.1
[ECMA262]	Section 8.6.1

7.3.3 Methods

7.3.3.1 toString()

49.10.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_toString1.xhtml

Assertion: When the **toString()** method of a string object is called, the method MUST return a primitive string value of the string object.

Traceability:

[ESMP]	Section 6.5.3.1
[ECMA262]	Section 15.5.4.2

49.10.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_toString2.xhtml

Assertion: When the **toString()** method of a non-string object is called, the method MUST return a TypeError.

Traceability:

[ESMP]	Section 6.5.3.1
[ECMA262]	Section 15.5.4.2

7.3.3.2 valueOf()

49.10.12

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_valueOf1.xhtml

Assertion 1: When the **valueOf()** method of a string object is called, the method MUST return a primitive string value of the string object.

Assertion 2: When the **charAt()** method of a string object S1 is called, the original string object S1 MUST remain unaffected.

Traceability:

[ESMP]	Section 6.5.3.2
[ECMA262]	Section 15.5.4.3

7.3.3.3 charAt()

49.10.13

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_charAt1.xhtml

Assertion: When the **charAt()** method of a string object S1 is called, where a number N representing an index is passed as argument, then the method MUST return a string containing the $(N+1)^{Th}$ character of the string object S1.

Traceability:

[ESMP]	Section 6.5.3.3
[ECMA262]	Section 15.5.4.4

49.10.14

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_charAt2.xhtml

Assertion: When the **charAt()** method of a string object S1 is called, where a number N whose value is less than 0 or greater the string length of S1 is passed as argument, then the method MUST return an empty string.

Traceability:

[ESMP]	Section 6.5.3.3
[ECMA262]	Section 15.5.4.4

7.3.3.4 charCodeAt()

49.10.15

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_charCodeAt1.xhtml

Assertion: When the **charCodeAt()** method of a string object S1 is called, where a number N representing an index is passed as argument, then the method MUST return an integer which is the UNICODE value of the (N+1)th character of string object S1.

Traceability:

[ESMP]	Section 6.5.3.4
[ECMA262]	Section 15.5.4.5

49.10.16

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_charCodeAt2.xhtml

Assertion: When the **charCodeAt()** method of a string object S1 is called, where a number N whose value is less than 0 or greater than length of S2 is passed as argument, then the method MUST return **NaN**.

Traceability:

[ESMP]	Section 6.5.3.3
[ECMA262]	Section 15.5.4.4

7.3.3.5 concat()

49.10.17

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_concatt1.xhtml

Assertion 1: When the **concat()** method of a string object S1 is called, where one or more string objects S2, S3, … are passed as arguments, then the method MUST return the initial string S1 with the argument strings appended to it in the same order in which they are listed as the arguments.

Assertion 2: When the **concat()** method of a string object S1 is called, where no arguments are passed, then the method MUST return the initial string S1 unaffected.

Traceability:

[ESMP]	Section 6.5.3.5
[ECMA262]	Section 15.5.4.6

7.3.3.6 indexOf()

49.10.18

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_indexOf1.xhtml

Assertion 1: When the **indexOf()** method of a string object S1 is called, where a string S2 is passed as argument, then the method MUST make a search for the substring S2 within string S1 and if the search is successful, then MUST return the zero based index N at which the first match is found.

Assertion 2: When the **indexOf()** method of a string object S1 is called, where a string S2 is passed as argument, then the method MUST make a search for the substring S2 within string S1 and if no match is found, then MUST return -1.

Traceability:

[ESMP]	Section 6.5.3.6
[ECMA262]	Section 15.5.4.7

49.10.19

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_indexOf2.xhtml

Assertion 1: When the **indexOf()** method of a string object S1 is called, where a string S2 and an integer P are passed as arguments, then the method MUST make a search for the substring S2 within string S1 starting at the Pth position and if the search is successful, then MUST return the zero based index N at which the first match is found.

Assertion 2: When the **indexOf()** method of a string object S1 is called, where a string S2 and an integer P are passed as arguments, then the method MUST make a search for the substring S2 within string S1 starting at the Pth position and if no match is found, then MUST return -1.

Traceability:

[ESMP]	Section 6.5.3.6
[ECMA262]	Section 15.5.4.7

49.10.20

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_indexOf3.xhtml

Assertion 1: When the **indexOf()** method of a string object S1 is called, where a string S2 and an integer P are passed as arguments and if P <= 0, then the method MUST make a search for the substring S2 within string S1 and if the search is successful, then MUST return the zero based index N at which the first match is found.

Assertion 2: When the **indexOf()** method of a string object S1 is called, where a string S2 and an integer P are passed as arguments and if P <= 0, then the method MUST make a search for the substring S2 within string S1 and if no match is found, then MUST return -1.

Traceability:

[ESMP]	Section 6.5.3.6
[ECMA262]	Section 15.5.4.7

49.10.21

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_indexOf4.xhtml

Assertion: When the **indexOf()** method of a string object S1 is called, where a string S2 and an integer P are passed as arguments, and if P is greater than or equal to string length of S1, then the method MUST return -1 irrespective of the match.

Traceability:

[ESMP]	Section 6.5.3.6
[ECMA262]	Section 15.5.4.7

7.3.3.7 lastIndexOf()

49.10.22

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_lastIndexOf1.xhtml

Assertion 1: When the **lastIndexOf()** method of a string object S1 is called, where a string S2 is passed as argument, then the method MUST make a back-ward search for the substring S2 starting from the end of the string S1 and if the search is successful, then MUST return the zero-based index N at which the first match is found.

Assertion 2: When the **lastIndexOf()** method of a string object S1 is called, where a string S2 is passed as argument, then the method MUST make a back-ward search for the substring S2 starting from the end of the string S1 and if no match is found, MUST return -1.

Traceability:

[ESMP]	Section 6.5.3.7
[ECMA262]	Section 15.5.4.8

49.10.23

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_lastIndexOf2.xhtml

Assertion 1: When the **lastIndexOf()** method of a string object S1 is called, where a string S2 and an integer M are passed as arguments, then the method MUST make a back-ward search for the substring S2 starting from Mth position of the string S1 and if the search is successful, then MUST return the zero-based index N at which the first match is found.

Assertion 2: When the **lastIndexOf()** method of a string object S1 is called, where a string S2 and an integer M are passed as arguments, then the method MUST make a back-ward search for the substring S2 starting from Mth position of the string S1 and if no match is found, MUST return -1.

Traceability:

[ESMP]	Section 6.5.3.7
[ECMA262]	Section 15.5.4.8

49.10.24

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_lastIndexOf3.xhtml

Assertion: When the **lastIndexOf()** method of a string object S1 is called, where a string S2 and an integer M are passed as arguments and if M is a negative number, then the method return -1 irrespective of the matching.

Traceability:

[ESMP]	Section 6.5.3.7
[ECMA262]	Section 15.5.4.8

49.10.25

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_lastIndexOf4.xhtml

Assertion 1: When the **lastIndexOf()** method of a string object S1 is called, where a string S2 and an integer M are passed as arguments and if M >= length of string S1, then the method MUST make a back-ward search for the substring S2 starting from the end of the string S1 and if the search is successful, then MUST return the zero-based index N at which the first match is found.

Assertion 2: When the **lastIndexOf()** method of a string object S1 is called, where a string S2 and an integer M are passed as arguments and if M >= length of string S1, then the method MUST make a back-ward search for the substring S2 starting from the end of the string S1 and if no match is found, MUST return -1.

Traceability:

[ESMP]	Section 6.5.3.7
[ECMA262]	Section 15.5.4.8

7.3.3.8 localeCompare()

49.10.26

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_localeCompare1.xhtml

Assertion 1: When the **localeCompare()** method of a string object S1 is called, where another string S2 is passed as argument, then the method MUST compare both the strings S1 and S2 in locale dependent way and if they are equivalent, then the method MUST return 0.

Assertion 2: When the **localeCompare()** method of a string object S1 is called, where another string S2 is passed as argument, then the method MUST compare both the strings S1 and S2 in locale dependent way and if S1 comes before S2 in the sort order, then the method MUST return a negative value.

Assertion 3: When the **localeCompare()** method of a string object S1 is called, where another string S2 is passed as argument, then the method MUST compare both the strings S1 and S2 in locale dependent way and if S1 comes after S2 in the sort order, then the method MUST return a positive value.

Traceability:

[ESMP]	Section 6.5.3.8
[ECMA262]	Section 15.5.4.9

7.3.3.9 match()

7.3.3.10 replace()

7.3.3.11 search()

7.3.3.12 slice()

49.10.27

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_slice1.xhtml

Assertion 1: When the **slice()** method of a string object S1 is called, where a start number M is passed as argument, then the method MUST return the substring of S1 starting at index M and ending at the end of string S1.

Assertion 2: When the **slice()** method of a string object S1 is called, where a negative start number M is passed as argument, then the method MUST return the substring of S1 starting at index "string S1 length - M" and ending at the end of string S1.

Traceability:

[ESMP]	Section 6.5.3.12
[ECMA262]	Section 15.5.4.13

49.10.28

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_slice2.xhtml

Assertion 1: When the **slice()** method of a string object S1 is called, where a start number M and end number N are passed as arguments, then the method MUST return the substring of S1 starting at index M and ending at the index N .

Assertion 2: When the **slice()** method of a string object S1 is called, where a start number M and a negative end number N are passed as arguments, then the method MUST return the substring of S1 starting at index M and ending at the index (string S1 length – N) .

Assertion 3: When the **slice()** method of a string object S1 is called, where a negative start number M and end number N are passed as arguments, then the method MUST return the substring of S1 starting at index "string S1 length - M" and ending at the index "string S1 length - N" .

Assertion 4: When the **slice()** method of a string object S1 is called, where a negative start number M and a negative end number N are passed as arguments, then the method MUST return the substring of S1 starting at index "string S1 length - M" and ending at the index N .

Traceability:

[ESMP]	Section 6.5.3.12
[ECMA262]	Section 15.5.4.13

7.3.3.13 split()

49.10.29

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_split1.xhtml

Assertion 1: When the **split()** method of a string object S1 is called, where a string separator S2 is passed as argument, then the method MUST return an array whose contents are the sub strings from string S1 split at the each occurrence of separator S2 . The array elements MUST not include the separator character.

Assertion 2: When the **split()** method of a string object S1 is called, the original string object S1 MUST remain unaffected.

Traceability:

[ESMP]	Section 6.5.3.13
[ECMA262]	Section 15.5.4.14

49.10.30

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_split2.xhtml

Assertion 1: When the **split()** method of a string object S1 is called, where a string separator S2 and an integer M are passed as arguments, then the method MUST return an array whose contents are the first M number of sub strings from string S1 split at the each occurrence of separator S2 .

Assertion 2: When the **split()** method of a string object S1 is called, where a string separator S2 and 0 are passed as arguments, then the method MUST return an empty array.

Traceability:

[ESMP]	Section 6.5.3.13

[ECMA262]	Section 15.5.4.14

7.3.3.14 substring()

49.10.31

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_substring1.xhtml

Assertion 1: When the **substring()** method of a string object S1 is called, where a start number M is passed as argument, then the method MUST return the substring of S1 starting at index M and ending at the end of string S1.

Assertion 2: When the **substring()** method of a string object S1 is called, where a start number M is passed as argument, and if M is zero or negative or NaN, then the method MUST return the substring of S1 starting at index 0 and ending at the end of string S1.

Assertion 3: When the **substring()** method of a string object S1 is called, without passing any arguments, then the method MUST return the substring of S1 starting at index 0 and ending at the end of string S1.

Assertion 4: When the **substring()** method of a string object S1 is called, the target string S1 remains unchanged.

Traceability:

[ESMP]	Section 6.5.3.14
[ECMA262]	Section 15.5.4.15

49.10.32

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_substring2.xhtml

Assertion 1: When the **substring()** method of a string object S1 is called, where a start number M and end number N are passed as arguments, and if N > M, then the method MUST return the substring of S1 starting at index M and ending at the index N .

Assertion 2: When the **substring()** method of a string object S1 is called, where a start number M and end number N are passed as arguments, and if N < M, then the method MUST return the substring of S1 starting at index N and ending at the index M .

Assertion 3: When the **substring()** method of a string object S1 is called, where a start number M and end number N are passed as arguments, and if N is 0 or negative or NaN, then the method MUST return the substring of S1 starting at index 0 and ending at the index M .

Assertion 4: When the **substring()** method of a string object S1 is called, where a start number M and end number N are passed as arguments, and if M = N, then the method MUST return the single character string pointed by the index M in S1.

Traceability:

[ESMP]	Section 6.5.3.14
[ECMA262]	Section 15.5.4.15

7.3.3.15 toLowerCase()

49.10.33

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_toLowerCase1.xhtml

Assertion 1: When the **toLowerCase()** method of a string object S1 is called, then the method MUST return a string, which is the string of the S1 converted to all lowercase characters as defined by the Unicode Character Database in a locale independent way.

Assertion 2: When the **toLowerCase()** method of a string object S1 is called, then the initial string object S1 MUST remain unaffected.

Traceability:

[ESMP]	Section 6.5.3.15
[ECMA262]	Section 15.5.4.16

49.10.34

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_toLocaleLowerCase1.xhtml

Assertion 1: When the **toLocaleLowerCase()** method of a string object S1 is called, then the method MUST return a string, which is the string of the S1 converted to all lowercase characters as defined by the Unicode Character Database for the host environment's current locale.

Assertion 2: When the **toLocaleLowerCase()** method of a string object S1 is called, then the initial string object S1 MUST remain unaffected.

Traceability:

[ESMP]	Section 6.5.3.15

[ECMA262]	Section 15.5.4.16

7.3.3.16 toUpperCase()

49.10.35

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_toUpperCase1.xhtml

Assertion 1: When the **toUpperCase()** method of a string object S1 is called, then the method MUST return a string, which is the string of the S1 converted to all uppercase characters as defined by the Unicode Character Database in a locale independent way.

Assertion 2: When the **toUpperCase()** method of a string object S1 is called, then the initial string object S1 MUST remain unaffected.

Traceability:

[ESMP]	Section 6.5.3.16
[ECMA262]	Section 15.5.4.18

49.10.36

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/6.5_string/string_toLocaleUpperCase1.xhtml

Assertion 1: When the **toLocaleUpperCase()** method of a string object S1 is called, then the method MUST return a string, which is the string of the S1 converted to all uppercase characters as defined by the Unicode Character Database for the host environment's current locale.

Assertion 2: When the **toLocaleUpperCase()** method of a string object S1 is called, then the initial string object S1 MUST remain unaffected.

Traceability:

[ESMP]	Section 6.5.3.16

[ECMA262]	Section 15.5.4.18

7.4 RegExp Object

7.4.1 Version History

49.11.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-014

Test path: esmp_Sec6/ 6.6_regExp/regExp_version.xhtml

Assertion: The string object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.6.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

7.4.2 Properties

7.4.2.1 source

49.11.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_source1.xhtml

Assertion: A regular expression object can be instantiated by using either **RegExp(pattern, flags)** constructor syntax or by **/pattern/flags** syntax.

Assertion: The **source** property of a regular expression object is the original pattern string of that object.

Traceability:

[ESMP]	Section 6.6.4.1
[ECMA262]	Section 15.10.7.1
[JSBIBLE]	Chapter 38

49.11.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_source_readonly1.xhtml

Assertion: The **source** property of a RegExp object is a read-only property.

Traceability:

[ESMP]	Section 6.6.4.1
[ECMA262]	Section 15.10.7.1

7.4.2.2 global

49.11.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_global1.xhtml

Assertion: The **global** property of a regular expression object must be Boolean true, if the global modifier character **g** was used as a flag to create that object.

Assertion: The **global** property of a regular expression object must be Boolean false, if the global modifier character **g** was not used as a flag to create that object.

Traceability:

[ESMP]	Section 6.6.4.2
[ECMA262]	Section 15.10.7.2

49.11.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_global_readonly1.xhtml

Assertion: The **global** property of a regular expression object is a read-only property.

Traceability:

[ESMP]	Section 6.6.4.2
[ECMA262]	Section 15.10.7.2

7.4.2.3 ignoreCase

49.11.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ignoreCase1.xhtml

Assertion: The **ignoreCase** property of a regular expression object is Boolean true, if the ignoreCase modifier is used to create that object.

Assertion: The **ignoreCase** property of a regular expression object is Boolean false, if the ignoreCase modifier is not used to create that object.

Traceability:

[ESMP]	Section 6.6.4.3
[ECMA262]	Section 15.10.7.3

49.11.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ignoreCase_readonly1.xhtml

Assertion: The **ignoreCase** property of a regular expression object is a read-only property.

Traceability:

[ESMP]	Section 6.6.4.3
[ECMA262]	Section 15.10.7.3

7.4.2.4 lastIndex

49.11.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_lastIndex1.xhtml

Assertion: The **lastIndex** property of a newly constructed regular expression object must be set to the number 0.

Traceability:

[ESMP]	Section 6.6.4.4
[ECMA262]	Section 15.10.4.1

49.11.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_lastIndex2.xhtml

Assertion: The **lastIndex** property of a regular expression object specifies the index in the target string, at which to start the next match.

Traceability:

[ESMP]	Section 6.6.4.4
[ECMA262]	Section 15.10.7.5

7.4.2.5 multiline

49.11.10

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_multiline1.xhtml

Assertion: The **multiline** property of a regular expression object must be Boolean true, if the global modifier character **m** was used as a flag to create that object.

Assertion: The **multiline** property of a regular expression object must be Boolean false, if the global modifier character **m** was not used as a flag to create that object.

Traceability:

[ECMA262]	Section 15.10.7.5
[JSBIBLE]	CD-301

49.11.11

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_multiline2.xhtml

Assertion: The **multiline** property of the static RegExp object must be Boolean true, if the global modifier character **m** was used as a flag to create the most recent regular expression object within the scope of current document.

Assertion: The **multiline** property of the static RegExp object must be Boolean false, if the global modifier character **m** was not used as a flag to create the most recent regular expression object within the scope of current document.

Traceability:

[ECMA262]	Section 15.10.7.5
[JSBIBLE]	CD-306

7.4.2.6 input

49.11.12

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_input1.xhtml

Assertion: The **input** property of the static **RegExp** object is the string against which a regular expression is compared in search of a match.

Traceability:

[JSBIBLE]	CD-305

7.4.2.7 lastMatch

49.11.13

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_lastMatch1.xhtml

Assertion: The **lastMatch** property of the static **RegExp** object is matching string of most recently executed regular expression.

Assertion: The value of the **lastMatch** property of the static **RegExp** object must be same as that of first element of return array of the most recently executed regular expression **exec()** method.

Traceability:

[JSBIBLE]	CD-307

7.4.2.8 leftContext

49.11.14

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_leftContext1.xhtml

Assertion: The **leftContext** property of the static **RegExp** object must contain the part of the main string used in the most recently executed regular expression matching operation, starting from the point at which the most recent search began and upto but not including the matched string.

Traceability:

[JSBIBLE]	CD-308

7.4.2.9 rightContext

49.11.15

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_rightContext1.xhtml

Assertion: The **rightContext** property of the static **RegExp** object must contain the part of the main string used in the most recently executed regular expression matching operation, starting immediately from the current match and upto the end of the main string.

Traceability:

[JSBIBLE]	CD-308

7.4.2.10 Backreferences aka $1..$9

49.11.16

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_backref1.xhtml

Assertion: When a regular expression method is executed, any parenthesized results upto nine values must be stored in static **RegExp**'s nine properties **$1..$9**.

Traceability:

[JSBIBLE]	CD-308

7.4.2.11 lastParen

49.11.17

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_lastParen1.xhtml

Assertion: When a regular expression method is executed, any parenthesized results upto nine values must be stored in static **RegExp**'s nine properties **$1..$9**.

Traceability:

[JSBIBLE]	CD-308

7.4.2.12 returnArray properties

49.11.18

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_returnArray1.xhtml

Assertion: When a regular expression method is executed, the first element of the return array must be the string segment from target string, found to be matched with the regular expression.

Assertion: When a regular expression method is executed, the **index** property of the return array must get the index of the start of the matched string found in the target string.

Assertion: When a regular expression method is executed, the **input** property of the return array must be the entire target string.

Traceability:

[JSBIBLE]	CD-291

7.4.3 Methods

7.4.3.1 exec()

49.11.19

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_exec1.xhtml

Assertion: When the **exec** method of a regular expression object is called, where a string (or String object) S is passed as an argument, then the method must perform a regular expression match on the string and must return **null** if the string did not match.

Traceability:

[ESMP]	Section 6.6.5.1
[ECMA262]	Section 15.10.6.2

49.11.20

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_exec2.xhtml

Assertion: When the **exec** method of a regular expression object is called, where a string (or String object) S is passed as an argument, then the method must perform a regular expression match on the string and must return an Array object containing the results of the match.

Assertion: The index [0] of the return array must be the string of most recent matched characters.

Traceability:

[ESMP]	Section 6.6.5.1
[ECMA262]	Section 15.10.6.2

49.11.21

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_exec3.xhtml

Assertion: When the pattern of a regular expression object contains parenthesized components, and if the **exec()** method of that object is called by passing a string S as argument, then the indexes [1] to [n] of the return array must contain the results of each of the component match.

Traceability:

[ESMP]	Section 6.6.5.1
[ECMA262]	Section 15.10.6.2

49.11.22

Requirement: Optional

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_exec_shortcut1.xhtml

Assertion: The shortcut method of calling an exec method of a regular expression object **re** to match against string **s1** is to pass string **s1** directly as argument to regular expression object. In other words, the syntax **re.exec(s1)** is equivalent to syntax **re(s1)**.

Traceability:

[JSBIBLE]	CD-290

49.11.23

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_exec_global1.xhtml

Assertion: When the global identifier **g** is used in the pattern of a regular expression object, then the search for the pattern can be performed throughout a target string, rather than just for the first match.

Assertion: When the global identifier **g** is not used in the pattern of a regular expression object, then the search for the pattern can be performed only for the first match in a target string.

Traceability:

[ESMP]	Section 6.6.5.1
[ECMA262]	Section 15.10.6.2

49.11.24

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_exec_ignoreCase11.xhtml

Assertion: When the ignoreCase identifier **i** is used in the pattern of a regular expression object, then a case-insensitive search for the pattern must be made in the target string with any pattern matching operation.

Assertion: When the ignoreCase identifier **i** is not used in the pattern of a regular expression object, then only a case sensitive search for the pattern must be made in the target string with any pattern matching operation.

Traceability:

[ESMP]	Section 6.6.5.1
[ECMA262]	Section 15.10.6.2

49.11.25

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_exec_multilinel1.xhtml

Assertion: When the multiline identifier **m** is used in the pattern of a regular expression object, then a multi-line search for the pattern must be made in the target string in any pattern matching operations.

Assertion: When the multiline identifier **m** is not used in the pattern of a regular expression object, then the search for the pattern must be made in the target string only in its first line, with any pattern matching operations.

Traceability:

[ESMP]	Section 6.6.5.1
[ECMA262]	Section 15.10.6.2

49.11.26

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_wordboundary1.xhtml

Assertion: The meta character **\b** in the pattern restricts the matching string bounded to a word boundary.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-284

49.11.27

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_word_nonboundary1.xhtml

Assertion: The meta character **\B** in the pattern restricts the matching string not to be bounded to a word boundary.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-284

49.11.28

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_numeral1.xhtml

Assertion: The meta character **\d** in the pattern matches only with a numeral character

Assertion: The meta character **\D** in the pattern matches only with a non-numeral character.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-284

49.11.29

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_whitespace1.xhtml

Assertion: The meta character **\s** in the pattern matches with a white space character.

Assertion: The meta character **\S** in the pattern matches with any non-white space character.

Traceability:

[ESMP]	Section 6.6.2

[JSBIBLE]	CD-284

49.11.30

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_charnum1.xhtml

Assertion: The meta character **\w** in the pattern matches any letter, numeral or underscore character.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.31

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_charnum2.xhtml

Assertion: The meta character **\W** in the pattern matches any character other than letter, numeral or underscore characters.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.32

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_dotchar1.xhtml

Assertion: The meta character **.** in the pattern matches any character other than newline character.

Assertion: When the character backslash preceeds a meta character, then that meta character must be treated as an ordinary character in the search operation.

Eg: \. Matches the dot character in the target string.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.33

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_dollarchar1.xhtml

Assertion: When the meta character **$** is used at the end of the pattern string, then the match will be found only if the pattern string matches a string segment located at the end of the target string.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.34

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_caratchar1.xhtml

Assertion: When the meta character ^ is used at the beginning of the pattern string, then the match will be found only if the pattern string matches a string segment located at the beginning of the target string.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.35

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ charset1.xhtml

Assertion: The set pattern such as **[xyz]** specifies that any one of the characters indicated in the set (ie., x, y or z) must match with the target character.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.36

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ charset_range1.xhtml

Assertion: The range pattern such as **[a-d]** specifies that any one of the characters indicated in the range (ie., a,b,c or d) must match with the target character.

.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.37

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_negcharset1.xhtml

Assertion: The negated set pattern such as **[^xyz]** specifies that none of the characters indicated in the set (i.e., x, y or z) must match with the target character.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.38

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ charset_negrange1.xhtml

Assertion: The negated range pattern such as **[^m-p]** indicates none of the characters indicated in the range (i.e., m,n,o,p) must match with the target character.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.39

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ 0orMore1.xhtml

Assertion: When the meta character * follows any character in a pattern string, then in any of the matching operations, the preceding character must occur zero or more times for the match to be successful.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-286

49.11.40

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ 0orOne1.xhtml

Assertion: When the meta character **?** follows any character in a pattern string, then in any of the matching operations, the preceding character must occur zero or one time for the match to be successful.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-286

49.11.41

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ 1orMore1.xhtml

Assertion: When the meta character **+** follows any character in a pattern string, then in any of the matching operations, the preceding character must occur one or more times successively for the match to be successful.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-286

49.11.42

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ exactNtimes1.xhtml

Assertion: When the range modifer **{n}** follows any character in a pattern string, where n is an integer, then in any of the matching operations, the preceding character must occur exactly **n** times successively for the match to be successful.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-286

49.11.43

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ NorMore1.xhtml

Assertion: When the range modifer **{n,}** follows any character in a pattern string, where n is an integer, then in any of the matching operations, the preceding character must occur either **n** or more times successively for the match to be successful.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-286

49.11.44

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ NtoM1.xhtml

Assertion: When the range modifer **{n,m}** follows any character in a pattern string, where n and m are integers and **n<m**, then in any of the matching operations, the preceding character must occur at least **n** times, but not more than **m** times successively for the match to be successful.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-286

49.11.45

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ NtoM2.xhtml

Assertion: When the range modifer **{n,m}** follows any character in a pattern string, where n and m are integers and **n>m**, then it is an invalid syntax.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-286

49.11.46

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ NtoM3.xhtml

Assertion: When the range modifer **{n,m}** follows any character in a pattern string, where n and m are integers and **n=m**, then in any of the matching operations, the preceding character must occur exactly **n** times successively for the match to be successful.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-286

49.11.47

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ pipeChar1.xhtml

Assertion: The pipe character in the pattern such as **[x|y]** specifies that either of the characters delimited by pipe symbol can match with the target character.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

49.11.48

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_ pipeChar2.xhtml

Assertion: The pipe character in the pattern such as **(x|y)** specifies that either of the characters delimited by pipe symbol can match with the target character.

Traceability:

[ESMP]	Section 6.6.2
[JSBIBLE]	CD-285

7.4.3.2 test()

49.11.49

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_test.xhtml

Assertion: When the **test()** method of a regular expression object is executed against a target string, then the method must return boolean **true**, if a match is found as per the pattern.

Assertion: When the **test()** method of a regular expression object is executed against a target string, then the method must return boolean **false**, if no match is found as per the pattern.

Traceability:

[ESMP]	Section 6.6.5.2
[ECMA262]	Section 15.10.6.3

7.4.3.3 toString ()

49.11.50

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_toString1.xhtml

Assertion: When a regular expression object R is created using **new RegExp(pattern, flags)** construnctor syntax, and when the **toString()** method of that regular expression object R is called, then the method must return a string S, which is "/" chars concatenated to the beginning and end of the pattern string used for creating that object.

Assertion: If the **global** property of the object R is true, then the character **g** must be appended to the string S.

Assertion: If the **ignoreCase** property of the object R is true, then the character **i** must be appended to the string S.

Traceability:

[ESMP]	Section 6.6.5.3
[ECMA262]	Section 15.10.6.4

49.11.51

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-020

Test path: esmp_Sec6/ 6.6_regExp/regExp_toString2.xhtml

Assertion: When a regular expression object R is created using **/pattern/flags** syntax, and when the **toString()** method of that regular expression object R is called, then the method must return the exact string used in creating the object.

Traceability:

[ESMP]	Section 6.6.5.3

[ECMA262]	Section 15.10.6.4

7.4.3.4 compile()

49.11.52

Requirement: Mandatory

SCR:

Test path: esmp_Sec6/ 6.6_regExp/regExp_compile1.xhtml

Assertion: The **compile()** method of a regular expression object compiles the expression into a regular expression object for use with any associated methods.

Traceability:

[JSBIBLE]	CD-300

7.4.3.5 String.match()

49.11.53

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_match1.xhtml

Assertion: When the **match** method of string object is called, where a regular expression object R is passed as an argument, then the method must perform a regular expression match on the string and must return **null** if the no match is found.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

49.11.54

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_match2.xhtml

Assertion: When the **exec** method of a regular expression object is called, where a string (or String object) S is passed as an argument, then the method must perform a regular expression match on the string and must return an Array object containing the results of the match.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

49.11.55

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_match3.xhtml

Assertion: When the **match** method of an empty string object is called, where a regular expression object R is passed as an argument, then the method must return **null**.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

7.4.3.6 String.replace()

49.11.56

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_replace.xhtml

Assertion: When the **replace(searchRegExp, replaceStr)** method of string object S is called, where the arguments **searchRegExp** is a regular expression object and **replaceStr** is any string, then the method must return a string which is the string S, with its matching strings replaced by the **replaceStr.**

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

7.4.3.7 String.search()

49.11.57

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_search1.xhtml

Assertion: When the **search** method of string object is called, where a regular expression object R is passed as an argument, then the method must perform a regular expression search on the string and must return the index of the matching string.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

49.11.58

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_search2.xhtml

Assertion: When the **search** method of string object is called, where a regular expression object R is passed as an argument, then the method must perform a regular expression search on the string and must return **-1** if no match is found.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

7.4.3.8 String.split()

49.11.59

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_split1.xhtml

Assertion: When the **split** method of string object S is called, where a regular expression object R is passed as an argument (which is to be treated as a separator), then the method must split the string S starting from left to right, at each matching separator and return an array with the stored split strings.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

49.11.60

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_split2.xhtml

Assertion: When the **split** method of string object S is called, where a regular expression object R is passed as an argument (which is to be treated as a separator), then the method must return the entire string S as the array of length 1.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

49.11.61

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_split3.xhtml

Assertion: When the **split** method of string object S is called, where a regular expression object R and an integer N are passed as arguments, then the method must split the string S starting from left to right, at each matching separator and return the first N split strings, as an array.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

49.11.62

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-019

Test path: esmp_Sec6/ 6.6_regExp/regExp_string_split4.xhtml

Assertion: When the **split** method of string object S is called, where a regular expression object R and the integer 0 are passed as arguments, then the method must return an empty array.

Traceability:

[ESMP]	Section 6.5.3.9
[ECMA262]	Section 15.5.4.10

7.5 Boolean Object

49.12.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_function1.xhtml

Assertion: When the Boolean() is called as a function, where a value V is passed as an argument, then it MUST return a Boolean value as per the following table.

Argument Type	Return value
Undefined	false
Null	false
Boolean true	true
Boolean false	false

Traceability:

[ESMP]	Section 6.7
[ECMA262]	Section 15.6.1, Sec 9.2

Example:

var aValue ; //undefined variable

var bValue = Boolean(aValue) // bValue = false

49.12.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_function2.xhtml

Assertion: When the Boolean() is called as a function, where a value V is passed as an argument, then it MUST return a Boolean value as per the following table.

Argument Type	Return value

+0	false
-0	false
0	false
NaN	false
Positive Infinity	true
Negative Infinity	true
Any other number	true

Traceability:

[ESMP]	Section 6.7
[ECMA262]	Section 15.6.1, Sec 9.2

Example:

```
var aValue = 23 ;
var bValue = Boolean(aValue)    //  bValue =  true
```

49.12.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_function3.xhtml

Assertion: When the Boolean() is called as a function, where a value V is passed as an argument, then it MUST return a Boolean value as per the following table.

Argument Type	Return value
Empty String	false
Non-empty string	true
Object	true

Traceability:

[ESMP]	Section 6.7
[ECMA262]	Section 15.6.1, Sec 9.2

Example:

var aValue = ""; //empty string

var bValue = Boolean(aValue) // bValue = false

49.12.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_constructor1.xhtml

Assertion: When the Boolean() is called as a constructor, where a value V is passed as an argument, then it MUST instantiate and return a Boolean object whose value is as per the following table.

Argument Type	Return Boolean Object value
Undefined	false
Null	false

Boolean true	true
Boolean false	false

Traceability:

[ESMP]	Section 6.7
[ECMA262]	Section 15.6.2.1, Sec 9.2

Example:

```
var aValue ; //undefined variable
var bValue = new Boolean(aValue)    //  bValue =  false
```

49.12.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_ constructor2.xhtml

Assertion: When the Boolean() is called as a constructor, where a value V is passed as an argument, then it MUST instantiate and return a Boolean object whose value is as per the following table.

Argument Type	Return Boolean Object value
+0	false
-0	false
0	false
NaN	false
Positive Infinity	true
Negative Infinity	true
Any other number	true

Traceability:

[ESMP]	Section 6.7
[ECMA262]	Section 15.6.2.1, Sec 9.2

Example:

var aValue = 23 ;

var bValue = new Boolean(aValue) // bValue = true

49.12.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_ constructor3.xhtml

Assertion: When the Boolean() is called as a constructor, where a value V is passed as an argument, then it MUST instantiate and return a Boolean object whose value is as per the following table.

Argument Type	**Return Boolean Object value**
Empty String	false
Non-empty string	true
Object	true

Traceability:

[ESMP]	Section 6.7
[ECMA262]	Section 15.6.2.1, Sec 9.2

Example:

var aValue = ""; //empty string

var bValue = new Boolean(aValue)   // bValue = false

7.5.1 Version History

49.12.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-014

Test path: esmp_Sec6/6.7_boolean/version.xhtml

Assertion: The Boolean object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.7.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

7.5.2 Methods

7.5.2.1 toString()

49.12.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_ toString1.xhtml

Assertion: When the **toString()** method of a Boolean object is called, a string "true" MUST be returned, if the Boolean object's value is true.

Traceability:

[ESMP]	Section 6.7.2.1
[ECMA262]	Section 15.6.4.2

49.12.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_ toString2.xhtml

Assertion: When the **toString()** method of a Boolean object is called, a string "false" MUST be returned, if the Boolean object's value is false.

Traceability:

[ESMP]	Section 6.7.2.1
[ECMA262]	Section 15.6.4.2

7.5.2.2 valueOf()

49.12.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_valueOf1.xhtml

Assertion: When the **valueOf()** method of a Boolean object is called, a Boolean true MUST be returned, if the Boolean object's value is true.

Traceability:

[ESMP]	Section 6.7.2.2
[ECMA262]	Section 15.6.4.2

49.12.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-021

Test path: esmp_Sec6/6.7_boolean/boolean_valueOf2..xhtml

Assertion: When the **valueOf()** method of a Boolean object is called, a Boolean false MUST be returned, if the Boolean object's value is false.

Traceability:

[ESMP]	Section 6.7.2.2
[ECMA262]	Section 15.6.4.2

7.6 Number Object

49.13.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_function1..xhtml

Assertion: When the **Number()** is called as a function, where a value X is passed as an argument, then it MUST return a primitive number type, whose value is as per following table.

Argument Type	Return primitive value
Undefined	NaN
Null	+0
Boolean true	1
Boolean false	+0
No argument	+0

Traceability:

[ESMP]	Section 6.8.
[ECMA262]	Section 15.7.1

Example:

var aValue ;

var bValue = Number(aValue) // bValue = NaN

49.13.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_function2..xhtml

Assertion: When the **Number()** is called as a function, where a number value X is passed as an argument, then it MUST return a primitive number type, whose value is the same as number X.

Traceability:

[ESMP]	Section 6.8.
[ECMA262]	Section 15.7.1

Example:

```
var aValue =  "20";

var bValue = Number(aValue)    //  bValue =  20
```

49.13.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_function3.xhtml

Assertion: When the **Number()** is called as a function, where a value X is passed as an argument, then it MUST return a primitive number type, whose value is as per following table.

Argument Type	Return primitive value
String (string numeric literal)	A number, which is the rounded value of the input's mathematical value.
Empty string	+0
String with only white-space	+0
String (non-string numeric literal)	NaN

Traceability:

[ESMP]	Section 6.8.
[ECMA262]	Section 15.7.1

Example:

```
var aValue =  "abcd";
var bValue = Number(aValue)    //  bValue =  NaN
```

49.13.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_function4.xhtml

Assertion: When the **Number()** is called as a function, where an object O is passed as an argument, then it MUST return a primitive number type, whose value depends on the primitive value of the object passed.

Traceability:

[ESMP]	Section 6.8.
[ECMA262]	Section 15.7.1

Example:

var aValue = new Boolean(false); // a boolean object

var bValue = Number(aValue)   // bValue = +0

49.13.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_constructor1.xhtml

Assertion: When the constructor **new Number()** is called, where a value X is passed as an argument, then it MUST instantiate and return a Number object, whose value is set, as per following table.

Argument Type	Value of the created Number object
Undefined	NaN
Null	+0
Boolean true	1
Boolean false	+0
No argument passed	+0

Traceability:

[ESMP]	Section 6.8.
[ECMA262]	Section 15.7.2.1

Example:

var aValue = "20";

var bValue = new Number(aValue) // bValue is a Number object, with value 20

49.13.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_constructor2..xhtml

Assertion: When the constructor **new Number()** is called, where a string X is passed as an argument, then it MUST instantiate and return a Number object, whose value is set, as per following table.

Argument Type	Value of the created Number object
String (string numeric literal)	A number, which is the rounded value of the input's mathematical value.
Empty string	+0
String with only white-space	+0
String (non-string numeric literal)	NaN

Traceability:

[ESMP]	Section 6.8.
[ECMA262]	Section 15.7.2.1

Example:

var aValue = ""; // empty string

var bValue = new Number(aValue) // bValue is a Number object, with value +0

49.13.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_constructor3.xhtml

Assertion: When the constructor **new Number()** is called, where a number X is passed as an argument, then it MUST instantiate and return a Number object, whose value is equal to the number passed.

Traceability:

[ESMP]	Section 6.8.
[ECMA262]	Section 15.7.2.1

Example:

```
var aValue =  20 ;
var bValue = new Number(aValue)    //  bValue  is a Number object, with value  20
```

49.13.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_constructor4.xhtml

Assertion: When the constructor **new Number()** is called, where an object O is passed as an argument, then it MUST instantiate and return a Number object, whose value depends on the primitive value of the object passed.

Traceability:

[ESMP]	Section 6.8.
[ECMA262]	Section 15.7.2.1

Example:

```
var aValue =  new Boolean(false); // a boolean object
var bValue = new Number(aValue)    //  bValue  is a Number object, with value  0
```

7.6.1 Version History

49.13.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-014

Test path: esmp_Sec6/6.8_number/version.xhtml

Assertion: The number object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.8.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

7.6.2 Constants

7.6.2.1 MAX_VALUE

49.13.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_maxvalue1.xhtml

Assertion: The value of the MAX_VALUE property of a Number object is the largest positive finite value of the number type supported by the implementation and should be approximately equal to 1.797693134862316e+308

Traceability:

[ESMP]	Section 6.8.2.1
[ECMA262]	Section 15.7.3.2

7.6.2.2 MIN_VALUE

49.13.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_minvalue1.xhtml

Assertion: The value of the MIN_VALUE property of a Number object is the smallest positive finite value of the number type supported by the implementation and should be approximately equal to 5.562684646268e-309

Traceability:

[ESMP]	Section 6.8.2.2
[ECMA262]	Section 15.7.3.3

7.6.2.3 NaN

49.13.12

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_NaN.xhtml

Assertion: The value of the **NaN** property of a Number object MUST be **NaN**.

Traceability:

[ESMP]	Section 6.8.2.3
[ECMA262]	Section 15.7.3.4

7.6.2.4 NEGATIVE_INFINITY

49.13.13

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_negInfinity.xhtml

Assertion: The value of the NEGATIVE_INFINITY property of a Number object MUST be any value, which is less than the implementation supported MIN_VALUE .

Traceability:

[ESMP]	Section 6.8.2.4
[ECMA262]	Section 15.7.3.5

7.6.2.5 POSITIVE_INFINITY

49.13.14

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_posInfinity.xhtml

Assertion: The value of the POSITIVE_INFINITY property of a Number object MUST be MUST be any value, which is greater than the implementation supported MAX_VALUE .

Traceability:

[ESMP]	Section 6.8.2.4
[ECMA262]	Section 15.7.3.5

7.6.3 Methods

7.6.3.1 toExponential()

49.13.15

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toExponential1.xhtml

Assertion: When the method **toExponential()** of a number object N is called, without passing any arguments, then the method MUST return the string representation of number N in exponential format .

Traceability:

[ESMP]	Section 6.8.3.1
[ECMA262]	Section 15.7.4.6

Example:

var aNumber = 38.7;

var bNumber = aNumber.exponential()   // bNumber = "3.87e+1"

49.13.16

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toExponential2.xhtml

Assertion: When the method **toExponential()** of a number object N is called, where an integer M is passed as argument, then the method MUST return the string representation of number N in exponential format with M digits to the right of the decimal point. Any missing digits MUST be zero filled.

Traceability:

[ESMP]	Section 6.8.3.1
[ECMA262]	Section 15.7.4.6

Example:

```
var aNumber =  38.7;

var bNumber = aNumber.exponential(5)    //  bNumber =  "3.87000e+1"
```

49.13.17

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toExponential3.xhtml

Assertion: When the method **toExponential()** of a number object N is called, where an integer M is passed as argument, and if M < 0, then the compiler MUST throw **RangeError** exception.

Traceability:

[ESMP]	Section 6.8.3.1
[ECMA262]	Section 15.7.4.6

49.13.18

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toExponential4.xhtml

Assertion: When the method **toExponential()** of a NaN is called, then method MUST return the string NaN.

Traceability:

[ESMP]	Section 6.8.3.1
[ECMA262]	Section 15.7.4.6

49.13.19

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toExponential5.xhtml

Assertion: When the method **toExponential()** of a number object N is called, without passing any arguments, then the method MUST return the string representation of number N in exponential format with enough digits to the right of the decimal point to uniquely specify the number.

Traceability:

[ESMP]	Section 6.8.3.1
[ECMA262]	Section 15.7.4.6

Example:

var aNumber = 0.00000387;

var bNumber = aNumber.exponential()   // bNumber = "3.87e-6"

49.13.20

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toExponential6.xhtml

Assertion: The implementation MUST support at least 14 digits of accuracy, with the **toExponential()** method of the number object.

Assertion: The implementation MUST support **toExponential()** method of Number object, with an integer argument M, where M is at least 13.

Traceability:

[ESMP]	Section 6.8.3.1
[ECMA262]	Section 15.7.4.6

Example:

var aNumber = 11234567890123456789;

var bNumber = aNumber.exponential(15) // bNumber = "1.123456789012300e+15"

49.13.21

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toExponential7.xhtml

Assertion: The implementation MUST support at least 14 digits of accuracy, with the **toExponential()** method of the number object.

Traceability:

[ESMP]	Section 6.8.3.1

[ECMA262]	Section 15.7.4.6

49.13.22

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toExponential8.xhtml

Assertion: When the method **toExponential()** of a number object N is called, where an integer M is passed as argument, and if M is greater than the maximum value supported by the implementation, then the compiler MUST throw **RangeError** exception.

Traceability:

[ESMP]	Section 6.8.3.1
[ECMA262]	Section 15.7.4.6

7.6.3.2 toFixed()

49.13.23

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toFixed1.xhtml

Assertion: When the method **toFixed()** of a number object N is called, where an integer M is passed as argument, and if the number N contains less than M digits after decimal point, then the method MUST return the string representation of number N in fixed format with M digits to the right of the decimal point filling all the missing digits to the right of the decimal point with zeros.

Traceability:

[ESMP]	Section 6.8.3.2
[ECMA262]	Section 15.7.4.5

Example:

var aNumber = 38.7;

var bNumber = aNumber.toFixed(5) // bNumber = "38.70000"

49.13.24

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toFixed2.xhtml

Assertion: When the method **toFixed()** of a number object N is called, where an integer M is passed as argument, and if the number N contains more than M digits after decimal point, then the method MUST return the string representation of number N in fixed format with M digits to the right of the decimal point rounding the additional digits.

Traceability:

[ESMP]	Section 6.8.3.2
[ECMA262]	Section 15.7.4.5

Example:

```
var aNumber =  38.75213;

var bNumber = aNumber.toFixed(2)    //  bNumber =  "38.75"

var cNumber =  38.75824;

var dNumber = aNumber.toFixed(2)    //  dNumber =  "38.76"
```

49.13.25

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toFixed3.xhtml

Assertion: When the method **toFixed()** of a number object N is called, without passing any arguments, then the method MUST return the string representation of the number N after rounding it to a whole number.

Traceability:

[ESMP]	Section 6.8.3.2
[ECMA262]	Section 15.7.4.5

Example:

```
var aNumber =  38.28;
var bNumber = aNumber.toFixed()    //  bNumber =  "38"
var cNumber =  38.75824;
var dNumber = aNumber.toFixed()    //  dNumber =  "39"
```

49.13.26

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toFixed4.xhtml

Assertion: The implementation MUST support at least 14 decimal digits of output accuracy, with the **toFixed()** method of the number object.

Assertion: The implementation MUST support **toFixed()** method of Number object, with an integer argument M, where M is at least 13.

Traceability:

[ESMP]	Section 6.8.4.2
[ECMA262]	Section 15.7.4.5

49.13.27

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toFixed5.xhtml

Assertion: When the method **toFixed()** of a number object N is called, where an integer M is passed as argument, and if M is greater than the maximum value supported by the implementation, then the compiler MUST throw **RangeError** exception.

Traceability:

[ESMP]	Section 6.8.3.2
[ECMA262]	Section 15.7.4.5

49.13.28

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toFixed6.xhtml

Assertion: When the method **toFixed()** of a number object N is called, where an integer M is passed as argument, and if M is less than zero, then the compiler MUST throw **RangeError** exception.

Traceability:

[ESMP]	Section 6.8.3.2
[ECMA262]	Section 15.7.4.5

49.13.29

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/number_toFixed7.xhtml

Assertion: When the method **toFixed()** of a **NaN** is called, then method MUST return the string NaN.

Traceability:

[ESMP]	Section 6.8.3.2
[ECMA262]	Section 15.7.4.5

7.6.3.3 toLocaleString()

49.13.30

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toLocaleString1.xhtml

Assertion: In the absence of a true locale specific implementation, the method **toLocaleString()** of a number object N MUST return the string representation of number N in fixed format with 2 digits to the right of the decimal point, filling any missing digits with zeros.

Traceability:

[ESMP]	Section 6.8.3.3
[ECMA262]	Section 15.7.4.3

Example:

```
var aNumber =  38.7;
var bNumber = aNumber.toLocaleString()    //  bNumber =  "38.70"
```

7.6.3.4 toString()

49.13.31

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toString1.xhtml

Assertion 1: When the **toString()** method of a number object N is called, without passing any arguments, then the method MUST return the string representation of number N in decimal format.

Assertion 2: When the **toString()** method of a number object N is called, where a radix R is passed as an argument, and if the R is equal to 10, then the method MUST return the string representation of number N in decimal format.

Traceability:

[ESMP]	Section 6.8.3.4
[ECMA262]	Section 15.7.4.2

Example:

var aNumber = 38.7;

var bNumber = aNumber.toString() // bNumber = "38.7"

49.13.32

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toString2.xhtml

Assertion: When the **toString()** method of a number object N is called, where a radix R is passed as an argument, and if the R is any integer in the range 2 to 36, then the method MUST return the string representation of number N in format based on the radix value.

Traceability:

[ESMP]	Section 6.8.3.4
[ECMA262]	Section 15.7.4.2

Example:

```
var aNumber =  164;
var bNumber = aNumber. toString(16)    //  bNumber =  "A4"
```

49.13.33

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toString3.xhtml

Assertion: When the **toString()** method of a number object N is called, where a radix R is passed as an argument, and if the R is any integer in the range 2 to 36, then the method MUST return the string representation of number N in format based on the radix value.

Traceability:

[ESMP]	Section 6.8.3.4
[ECMA262]	Section 15.7.4.2

Example:

```
var aNumber =  164;
var bNumber = aNumber. toString(16)    //  bNumber =  "A4"
```

49.13.34

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toString4.xhtml

Assertion: Implementation MUST ensure at least 14 digits of accuracy when **toString()** method of a number object is called.

Traceability:

[ESMP]	Section 6.8.3.4
[ECMA262]	Section 15.7.4.2

49.13.35

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number /toString5.xhtml

Assertion: When the method **toString()** of a **NaN** is called, then the method MUST return the string NaN.

Traceability:

[ESMP]	Section 6.8.3.4
[ECMA262]	Section 15.7.4.2

49.13.36

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toString6.xhtml

Assertion: When the method **toString()** of a number object N is called, where an integer M is passed as argument, and if M is less than 2, then the compiler MUST throw **RangeError** exception.

Traceability:

[ESMP]	Section 6.8.3.2
[ECMA262]	Section 15.7.4.5

49.13.37

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toString7.xhtml

Assertion: When the method **toString()** of a number object N is called, where an integer M is passed as argument, and if M is greater than 32, then the compiler MUST throw **RangeError** exception.

Traceability:

[ESMP]	Section 6.8.3.2
[ECMA262]	Section 15.7.4.5

7.6.3.5 valueOf()

49.13.38

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/valueOf1.xhtml

Assertion: When the method **valueOf()** of a number object N is called, then the method MUST return the number value of the object N.

Traceability:

[ESMP]	Section 6.8.4.5
[ECMA262]	Section 15.7.4.4

Example:

var numObject = new Number(164);

var bNumber = numObject. valueOf() // bNumber = 164

49.13.39

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/valueOf2.xhtml

Assertion: When the method **valueOf()** of a number object N is called, then the method MUST return the number value of the object N.

Traceability:

[ESMP]	Section 6.8.4.5
[ECMA262]	Section 15.7.4.4

Example:

```
var numObject = new Number(null);
var bNumber = numObject. valueOf()    //  bNumber =  0
```

7.6.3.6 toPrecision()

49.13.40

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision1.xhtml

Assertion: When the method **toPrecision()** of a number object N is called, where no arguments are supplied, then a string representing the number N MUST be returned.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

49.13.41

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision2.xhtml

Assertion When the method **toPrecision()** of an object N is called, where the number value of N is NaN, then the string "NaN" must be returned.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

49.13.42

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision3.xhtml

Assertion: When the method **toPrecision()** of an object N is called, where the number value of N is positive infinity, then the string "Infinity" must be returned.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

49.13.43

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision4.xhtml

Assertion: When the method **toPrecision()** of an object N is called, where an integer M is passed as argument, and if M < 1, then compiler MUST throw a RangeError exception.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

49.13.44

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision5.xhtml

Assertion: When the method **toPrecision()** of an object N is called, where an integer M is passed as argument, and if M >21, then compiler MUST throw a RangeError exception.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

49.13.45

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision6.xhtml

Assertion: When the method **toPrecision()** of an object N is called, where the number value of N is negative infinity, then the string "-Infinity" must be returned.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

49.13.46

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision7.xhtml

Assertion: When the method **toPrecision()** of an object N is called with an integer P as argument, where the number value of N is 0, then a string of P number of 0's must be returned with a decimal point after first zero.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

Example:

```
var aNumber = 0;
var bNumber = aNumber. toPrecision(4)    //  bNumber =  "0.000"
```

49.13.47

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision8.xhtml

Assertion: When the method **toPrecision()** of a number object N is called, where an integer P is passed as argument, and if P is greater than 0 and less than 15, then the method MUST return a string representing the number N in fixed format with P number of total digits.

Traceability:

[ESMP]	Section 6.8.4.6

[ECMA262]	Section 15.7.4.7

Example:

var aNumber = 689.463;

var bNumber = aNumber. toPrecision(2) // bNumber = "689.46"

49.13.48

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision9.xhtml

Assertion: When the method **toPrecision()** of a number object N is called, where an integer P is passed as argument, and if P is greater than 0 and less than 15, then the method MUST return a string representing the number N in exponential notation with one digit before significand's decimal point and P number of digits total digits.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

Example:

var aNumber = -123.456;

var bNumber = aNumber. toPrecision(1) // bNumber = " -1e+2"

49.13.49

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-022

Test path: esmp_Sec6/6.8_number/toPrecision10.xhtml

Assertion: The implementation MUST support 14 digits of accuracy when using **toPrecision()** method of number object.

Traceability:

[ESMP]	Section 6.8.4.6
[ECMA262]	Section 15.7.4.7

7.7 Math Object

49.14.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_function1.xhtml

Assertion: When the function **Math()** is called, ReferenceError exception MUST be thrown.

Traceability:

[ESMP]	Section 6.9
[ECMA262]	Section 15.8

Example:

```
var aValue =  "20";
var bValue = Math(aValue)    //  MUST generate ReferenceError exception
```

49.14.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_constructor1.xhtml

Assertion: When the constructor **new Math()** is called, ReferenceError or TypeError exception MUST be thrown.

Traceability:

[ESMP]	Section 6.9
[ECMA262]	Section 15.8

Example:

```
var bValue = new Math()    //  MUST generate an error.
```

7.7.1 Version History

49.14.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-014

Test path: esmp_Sec6/6.9_math/version.xhtml

Assertion: The **Math** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.9.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

7.7.2 Properties and Constants

49.14.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_E.xhtml

Assertion: The constant **Math.E** MUST have the value 2.7182818284590452354 accurate up to at least 14 digits.

Traceability:

[ESMP]	Section 6.9.2
[ECMA262]	Section 15.8

49.14.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_LN2.xhtml

Assertion: The constant **Math.LN2** MUST have the value 0.6931471805599453 accurate up to at least 14 digits.

Traceability:

[ESMP]	Section 6.9.2
[ECMA262]	Section 15.8

49.14.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_LN10.xhtml

Assertion: The constant **Math.LN10** MUST have the value 2.302585092994046 accurate up to at least 14 digits.

Traceability:

[ESMP]	Section 6.9.2
[ECMA262]	Section 15.8

49.14.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_LOG10E.xhtml

Assertion: The constant **Math.LOG10E** MUST have the value 0.4342944819032518 accurate up to at least 14 digits.

Traceability:

[ESMP]	Section 6.9.2
[ECMA262]	Section 15.8

49.14.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_LOG2E.xhtml

Assertion: The constant **Math.LOG2E** MUST have the value 1.4426950408889634 accurate up to at least 14 digits.

Traceability:

[ESMP]	Section 6.9.2
[ECMA262]	Section 15.8

49.14.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_PI.xhtml

Assertion: The constant **Math.PI** MUST have the value 3.1415926535897932 accurate up to at least 14 digits.

Traceability:

[ESMP]	Section 6.9.2
[ECMA262]	Section 15.8

49.14.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_SQRT1_2.xhtml

Assertion: The constant **Math.SQRT1_2** MUST have the value 0.7071067811865476 accurate up to at least 14 digits.

Traceability:

[ESMP]	Section 6.9.2
[ECMA262]	Section 15.8

49.14.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_SQRT2.xhtml

Assertion: The constant **Math.SQRT2** MUST have the value 1.4142135623730951 accurate up to at least 14 digits.

Traceability:

[ESMP]	Section 6.9.2
[ECMA262]	Section 15.8

7.7.3 Methods

7.7.3.1 Integerizing Methods

49.14.12

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_round1.xhtml

Assertion 1: When the method Math.round(x) is called, where x is any number, then the method MUST return an integer, whose number value is closest to the given value x.

Assertion 2: When the method Math.round(x) is called, where x is already an integer, then the method MUST return the integer x.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.15

Example:

var aNumber = Math.round(38.27) // aNumber = 38

var bNumber = Math.round(38.72) // bNumber = 39

var cNumber = Math.round(-38.72) // cNumber = -38

var aNumber = Math.round(38) // aNumber = 38

49.14.13

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_round2.xhtml

Assertion : When the method Math.round(x) is called, where x is any number, and if two integers M and N are equally close to x, then the method must return the largest of M and N as the result.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.15

Example:

var aNumber = Math.round(38.5) // 38.5 is equally close to 38 and 39, so aNumber = 39

var bNumber = Math.round(-38.5) // -38.5 is equally close to -38 and -39, so bNumber = -38

49.14.14

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_round3.xhtml

Assertion 1: When the method Math.round(x) is called, where x is NaN, then the method MUST return NaN.

Assertion 2: When the method Math.round(x) is called, where x is +0, then the method MUST return +0.

Assertion 3: When the method Math.round(x) is called, where x is -0, then the method MUST return -0.

Assertion 4: When the method Math.round(x) is called, where x is +∞, then the method MUST return +∞.

Assertion 5: When the method Math.round(x) is called, where x is -∞, then the method MUST return -∞.

Assertion 6: When the method Math.round(x) is called, where x is greater than 0 but less than 0.5, then the method MUST return +0.

Assertion 7: When the method Math.round(x) is called, where x is less than 0 but greater or equal to –0.5, then the method MUST return –0.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.15

Example:

var aNumber = Math.round("abcd")   // aNumber = NaN

49.14.15

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_ceil1.xhtml

Assertion 1: When the method Math.ceil(x) is called, where x is any number, then the method MUST return an integer, which is the smallest number that is not less than x.

Assertion 2: When the method Math.ceil(x) is called, where x is already an integer, then the method MUST return the integer x.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.6

Example:

var aNumber = Math.ceil(38.27) // aNumber = 39

var bNumber = Math.ceil(-38.27) // bNumber = -38

49.14.16

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_ceil2.xhtml

Assertion 1: When the method Math.ceil(x) is called, where x is NaN, then the method MUST return NaN.

Assertion 2: When the method Math.ceil(x) is called, where x is +0, then the method MUST return +0.

Assertion 3: When the method Math.ceil(x) is called, where x is -0, then the method MUST return -0.

Assertion 4: When the method Math.ceil(x) is called, where x is +∞, then the method MUST return +∞.

Assertion 5: When the method Math.ceil(x) is called, where x is -∞, then the method MUST return -∞.

Assertion 6: When the method Math.ceil(x) is called, where x is less than 0 but greater than -1, then the method MUST return -0.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.6

Example:

var aNumber = Math.ceil("abc") // aNumber = NaN

49.14.17

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_floor1.xhtml

Assertion 1: When the method Math.floor(x) is called, where x is any number, then the method MUST return an integer, which is the greatest number that is not greater than x.

Assertion 2: When the method Math.floor(x) is called, where x is already an integer, then the method MUST return the integer x.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.9

Example:

var aNumber = Math. floor(38.72) // aNumber = 38

var bNumber = Math. floor(-38.27) // bNumber = -39

49.14.18

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_floor2.xhtml

Assertion 1: When the method Math.ceil(x) is called, where x is NaN, then the method MUST return NaN.

Assertion 2: When the method Math.ceil(x) is called, where x is +0, then the method MUST return +0.

Assertion 3: When the method Math.ceil(x) is called, where x is -0, then the method MUST return -0.

Assertion 4: When the method Math.ceil(x) is called, where x is +∞, then the method MUST return +∞.

Assertion 5: When the method Math.ceil(x) is called, where x is -∞, then the method MUST return -∞.

Assertion 6: When the method Math.ceil(x) is called, where x is greater than 0 but less than +1, then the method MUST return +0.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.9

Example:

```
var aNumber = Math.floor("abc")    //  aNumber =  NaN
```

49.14.19

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_floor3.xhtml

Assertion 1: The value of **Math.floor(x)** MUST be the same as the value of **–Math.ceil(-x)**.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.9

Assertion 2: The value of **Math.ceil(x)** MUST be the same as the value of **–Math.floor(-x)**.

Traceability:

[ESMP]	Section 6.9.3.1
[ECMA262]	Section 15.8.2.6

Example:

```
var aNumber = Math. floor(38.72);
        var bNumber = -Math.ceil(-38.72);    //  aNumber = bNumber
```

7.7.3.2 General Mathematical Methods

49.14.20

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_abs1.xhtml

Assertion: When the method Math.abs(x) is called, where x is any number, then the method MUST return the absolute value of 'x'. (i.e., same magnitude but with positive sign).

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.2

Example:

var aNumber = Math. abs(38.72) // aNumber = +38.72

var bNumber = Math. abs(-38.72) // bNumber = +38.72

49.14.21

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_abs2.xhtml

Assertion 1: When the method Math.abs(x) is called, where x is NaN, then the method MUST return NaN.

Assertion 2: When the method Math.abs(x) is called, where x is -0, then the method MUST return +0.

Assertion 3: When the method Math.abs(x) is called, where x is -∞, then the method MUST return +∞.

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.2

Example:

var aNumber = Math. abs("abcd")   // aNumber = NaN

49.14.22

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_exp1.xhtml

Assertion: When the method Math.exp(x) is called, where x is any number, then the method MUST return the exponential function of 'x'. (i.e., 'e' raised to the power of 'x', where 'e' is the base of the natural logarithms).

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.8

Example:

var aNumber = Math. exp(3) // aNumber = 20.085536… (which is e^3)

49.14.23

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_exp2.xhtml

Assertion 1: When the method Math.exp(x) is called, where x is NaN, then the method MUST return NaN.

Assertion 2: When the method Math.exp(x) is called, where x is +0, then the method MUST return 1.

Assertion 3: When the method Math.exp(x) is called, where x is -0, then the method MUST return 1.

Assertion 4: When the method Math.exp(x) is called, where x is +∞, then the method MUST return +∞.

Assertion 5: When the method Math.exp(x) is called, where x is -∞, then the method MUST return +0.

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.8

Example:

var aNumber = Math. exp("abcd") // aNumber = NaN

49.14.24

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_log1.xhtml

Assertion: When the method Math.log(x) is called, where x is any number, then the method MUST return the natural logarithm of 'x' (i.e., to base 'e').

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.10

Example:

var aNumber = Math. log(3) // aNumber = 1.09861…

49.14.25

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_log2.xhtml

Assertion 1: When the method Math.log(x) is called, where x is NaN, then the method MUST return NaN.

Assertion 2: When the method Math.log(x) is called, where x is less than 0, then the method MUST return NaN.

Assertion 3: When the method Math.log(x) is called, where x is +0, then the method MUST return -∞.

Assertion 4: When the method Math.log(x) is called, where x is -0, then the method MUST return -∞.

Assertion 5: When the method Math.log(x) is called, where x is 1, then the method MUST return +0.

Assertion 6: When the method Math.log(x) is called, where x is +∞, then the method MUST return +∞.

Assertion 7: When the method Math.log(x) is called, where x is -∞, then the method MUST return NaN.

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.10

Example:

var aNumber = Math. log("abcd") // aNumber = NaN

49.14.26

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_pow1.xhtml

Assertion: When the method Math.pow(x, y) is called, where x and y are any numbers, then the method MUST return the result of raising 'x' to the power of 'y'.

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.13

Example:

var aNumber = Math. pow(2,3) // aNumber = 8

49.14.27

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_pow2..xhtml

Assertion: When the method **Math.pow(x, y)** is called, then the method must return a value based on values of x and y as per following table.

x	y	Return Value
any value	NaN	NaN
NaN	non-zero	NaN
any value	+0 or -0	1
NaN	+0 or -0	1
abs(x) > 1	+infinity	+infinity
abs(x) > 1	-infinity	+0
abs(x) = 1	+infinity	NaN
abs(x) = 1	-infinity	NaN
abs(x) < 1	+infinity	+0
abs(x) < 1	-infinity	+infinity
+infinity	>0	+infinity
+infinity	<0	+0
-infinity	>0 and odd number	-infinity
-infinity	>0 and even number	+infinity
-infinity	<0 and odd number	-0
-infinity	<0 and even number	+0
+0	>0	+0
+0	<0	+infinity
-0	>0 and odd number	-0
-0	>0 and even number	+0

-0	<0 and odd number	-infinity
-0	<0 and even number	+infinity
<0, finite	finite, not integer	NaN

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.13

49.14.28

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_sqrt1.xhtml

Assertion: When the method Math.sqrt(x) is called, where x is any number, then the method MUST return the square root value of the number.

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.17

Example:

```
var aNumber = Math. sqrt(9)    //  aNumber =  3
```

49.14.29

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_sqrt2_method.xhtml

Assertion 1: When the method Math.sqrt(x) is called, where x is NaN, then the method MUST return NaN.

Assertion 2: When the method Math.sqrt(x) is called, where x is less than 0, then the method MUST return NaN.

Assertion 3: When the method Math.sqrt(x) is called, where x is +0, then the method MUST return +0.

Assertion 4: When the method Math.sqrt(x) is called, where x is -0, then the method MUST return -0.

Assertion 5: When the method Math.sqrt(x) is called, where x is +∞, then the method MUST return +∞.

Traceability:

[ESMP]	Section 6.9.3.2
[ECMA262]	Section 15.8.2.17

Example:

```
var aNumber = Math. sqrt(9)    //  aNumber =  3
```

7.7.3.3 Trigonometric Methods

49.14.30

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_acos1.xhtml

Assertion: When the method **Math.acos(x)** is called, where x is a number in the range: -1 to +1, then the method MUST return the arc cosine of the number 'x' in radians.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.2

Example:

var aNumber = Math. acos(0.5) // aNumber = 1.04719… (which is +π/3)

49.14.31

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_acos2.xhtml

Assertion : When the method **Math.acos(x)** is called, where x is NaN, then the method must return NaN.

Assertion : When the method **Math.acos(x)** is called, where x is greater than 1, then the method must return NaN.

Assertion : When the method **Math.acos(x)** is called, where x is less than -1, then the method must return NaN.

Assertion : When the method **Math.acos(x)** is called, where x exactly 1 then the method must return +0.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.2

49.14.32

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_asin1.xhtml

Assertion: When the method **Math.asin(x)** is called, where x is a number in the range: -1 to +1, then the method MUST return the arc sine of the number 'x'.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.8

Example:

var aNumber = Math. asin(0.5) // aNumber = 0.523598… (which is +π/6)

49.14.33

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_asin2.xhtml

Assertion 1: When the method **Math.asin(x)** is called, where x is NaN, then the method must return NaN.

Assertion 2: When the method **Math.asin(x)** is called, where x is greater than 1, then the method must return NaN.

Assertion 3: When the method **Math.asin(x)** is called, where x is less than -1, then the method must return NaN.

Assertion 4: When the method **Math.asin(x)** is called, where x is +0, then the method must return +0.

Assertion 5: When the method **Math.asin(x)** is called, where x is -0, then the method must return -0.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.8

49.14.34

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_atan1.xhtml

Assertion: When the method **Math.atan(x)** is called, where x is any number, then the method MUST return the arc-tangent of the number 'x'.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

Example:

```
var aNumber = Math. atan(1)   //  aNumber =  0.785398…  (which is  +π/4  radians )
var bNumber = Math. atan(+∞)   //  bNumber =  1.570796…  (which is  +π/2  radians )
```

49.14.35

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_atan2.xhtml

Assertion 1: When the method **Math.atan(x)** is called, where x is NaN, then the method must return NaN.

Assertion 2: When the method **Math.atan(x)** is called, where x is positive infinity, then the method must return approximately +π/2 .

Assertion 3: When the method **Math.atan(x)** is called, where x is negative infinity, then the method must return approximately -π/2 .

Assertion 4: When the method **Math.atan(x)** is called, where x is +0, then the method must return +0.

Assertion 5: When the method **Math.atan(x)** is called, where x is -0, then the method must return -0.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

49.14.36

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_atan2_yx1.xhtml

Assertion: When the method **Math.atan2(y,x)** is called, where x and y are any two numbers, then the method MUST return the arc tangent of the quotient y, x .

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

Example:

var aNumber = Math. atan2(1,1) // aNumber = 0.785398…. (which is π/4 radians, i.e. 45º)

49.14.37

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_atan2_yx2.xhtml

Assertion: When the method **Math.atan2(y,x)** is called, then the method must return a value based on values of x and y as per following table.

Y	x	Return Value
any value	NaN	NaN
NaN	any value	NaN
NaN	NaN	NaN
>0	+0	+π/2
>0	-0	+π/2
+0	>0	+0
+0	+0	+0
+0	-0	+π
+0	<0	+π
-0	>0	-0
-0	+0	-0
-0	-0	-π
-0	<0	-π
<0	+0	-π/2
<0	-0	-π/2
>0 and finite	+infinity	+0
>0 and finite	-infinity	+π
<0 and finite	+infinity	-0
<0 and finite	-infinity	-π
+infinity	finite	+π/2

-infinity	finite	-π/2
+infinity	+infinity	+π/4
+infinity	-infinity	+3π/4
-infinity	+infinity	-π/4
-infinity	-infinity	-3π/4

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

49.14.38

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_cos1.xhtml

Assertion: When the method **Math.cos(x)** is called, where x is a number representing radians, then the method MUST return cosine value of the number x.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

Example:

var aNumber = Math. cos(+π/2) // aNumber = 0

49.14.39

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_cos2.xhtml

Assertion 1: When the method **Math.cos(x)** is called, where x is NaN, then the method must return NaN.

Assertion 2: When the method **Math.cos(x)** is called, where x is positive infinity, then the method must return NaN .

Assertion 3: When the method **Math.cos(x)** is called, where x is negative infinity, then the method must return NaN .

Assertion 4: When the method **Math.cos(x)** is called, where x is +0, then the method must return 1.

Assertion 5: When the method **Math.cos(x)** is called, where x is -0, then the method must return 1.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

49.14.40

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_sin1.xhtml

Assertion: When the method **Math.sin(x)** is called, where x is a number representing radians, then the method MUST return sine value of number x.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

Example:

```
var aNumber = Math. sin(+π/2)    //  aNumber =  1
```

49.14.41

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_sin2.xhtml

Assertion 1: When the method **Math.sin(x)** is called, where x is NaN, then the method must return NaN.

Assertion 2: When the method **Math.sin(x)** is called, where x is positive infinity, then the method must return NaN .

Assertion 3: When the method **Math.sin(x)** is called, where x is negative infinity, then the method must return NaN .

Assertion 4: When the method **Math.sin(x)** is called, where x is +0, then the method must return +0.

Assertion 5: When the method **Math.sin(x)** is called, where x is -0, then the method must return -0.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

49.14.42

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_tan1.xhtml

Assertion: When the method **Math.tan(x)** is called, where x is a number representing radians, then the method MUST return tangent value of number x.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

Example:

var aNumber = Math. tan(+π/4) // aNumber = 1

49.14.43

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_tan2.xhtml

Assertion 1: When the method **Math.tan(x)** is called, where x is NaN, then the method must return NaN.

Assertion 2: When the method **Math.tan(x)** is called, where x is positive infinity, then the method must return NaN .

Assertion 3: When the method **Math.tan(x)** is called, where x is negative infinity, then the method must return NaN .

Assertion 4: When the method **Math.tan(x)** is called, where x is +0, then the method must return +0.

Assertion 5: When the method **Math.tan(x)** is called, where x is -0, then the method must return -0.

Traceability:

[ESMP]	Section 6.9.3.3
[ECMA262]	Section 15.8.2.4

7.7.3.4 Max()

49.14.44

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_max1.xhtml

Assertion: When the method **Math.max()** is called, where two or more values are passed as arguments, and if the arguments supplied are convertible into number values, then the method MUST return largest number of the resulting values.

Traceability:

[ESMP]	Section 6.9.3.4
[ECMA262]	Section 15.8.2.11

Example:

```
var aNumber = Math. max("38.1E-2", 400.23, 0.0003, 76)    //  aNumber =  400.23
```

49.14.45

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_max2.xhtml

Assertion 1: When the method **Math.max()** is called, where only one value is passed as arguments, and if the argument supplied is convertible into number value, then the method MUST return that value as a number.

Assertion 2: When the method **Math.max()** is called, without passing any arguments, then the method MUST return negative infinity as the result.

Assertion 3: When the method **Math.max()** is called, where any of the arguments supplied is NaN, then NaN MUST be returned as the result.

Assertion 4: When the method **Math.max()** is called, where any of the arguments supplied is positive infinity, then positive infinity MUST be returned as the result.

Assertion 5: The method **Math.max()** MUST treat +0 as greater than –0.

Traceability:

[ESMP]	Section 6.9.3.4
[ECMA262]	Section 15.8.2.11

Example:

```
var aNumber = Math. max(400.23)    //  aNumber =  400.23
var bNumber = Math. max()    //  bNumber =  -∞
```

7.7.3.5 min()

49.14.46

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_min1.xhtml

Assertion: When the method **Math.min()** is called, where two or more values are passed as arguments, and if the arguments supplied are convertible into number values, then the method MUST return the smallest number of the resulting values.

Traceability:

[ESMP]	Section 6.9.3.5
[ECMA262]	Section 15.8.2.11

Example:

```
var aNumber = Math.min("38.1E-2", 400.23, 0.0003, 76)    //  aNumber =  0.0003
```

49.14.47

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_min2.xhtml

Assertion 1: When the method **Math.min()** is called, where only one value is passed as arguments, and if the argument supplied is convertible into number value, then the method MUST return that value as a number.

Assertion 2: When the method **Math.min()** is called, without passing any arguments, then the method MUST return positive infinity as the result.

Assertion 3: When the method **Math.min()** is called, where any of the arguments supplied is a NaN, then NaN MUST be returned as the result.

Assertion 4: When the method **Math.min()** is called, where any of the arguments supplied is negative infinity, then negative infinity MUST be returned as the result.

Assertion 5: The method **Math.min()** MUST treat -0 as lesser than +0.

Traceability:

[ESMP]	Section 6.9.3.5
[ECMA262]	Section 15.8.2.11

Example:

```
var aNumber = Math.min(400.23)    //  aNumber =  400.23
```

7.7.3.6 random()

49.14.48

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_random1.xhtml

Assertion: When the method **Math.random()** is called, then it MUST return a pseudo-random number R, where R is a floating point number whose value is >= 0 and < 1.

Traceability:

[ESMP]	Section 6.9.3.6
[ECMA262]	Section 15.8.2.14

Example:

```
var aNumber = Math.random()    //  aNumber  can be  0.385623    (which is >=0, <1)
```

49.14.49

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_random2.xhtml

Assertion: When the method **Math.random(**X**)** is called, where X is an integer number, then it MUST return a pseudo-random number R, where R is an integer whose value is >= 0 and < X.

Traceability:

[ESMP]	Section 6.9.3.6
[ECMA262]	Section 15.8.2.14

Example:

```
var aNumber = Math.random(50)    //  aNumber  can be  38    (which is >0, <50)
```

49.14.50

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_random3.xhtml

Assertion: When the method **Math.random()** is called, the implementation MUST generate a truly randomly distributed numbers.

Traceability:

[ESMP]	Section 6.9.3.6
[ECMA262]	Section 15.8.2.14

49.14.51

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_random4.xhtml

Assertion: When the method **Math.random(X)** is called, where X is an integer number, the implementation MUST generate a truly randomly distributed numbers.

Traceability:

[ESMP]	Section 6.9.3.6
[ECMA262]	Section 15.8.2.14

49.14.52

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_random5.xhtml

Assertion: When the implementation supports only integers, then calling the method **Math.random()** without any arguments, MUST return NaN.

Traceability:

[ESMP]	Section 6.9.3.6
[ECMA262]	Section 15.8.2.14

49.14.53

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-023

Test path: esmp_Sec6/6.9_math/math_random6.xhtml

Assertion: When the method **Math.random(**X**)** is called, where X is a negative number, then the compiler MUST throw an exception.

Traceability:

[ESMP]	Section 6.9.3.6
[ECMA262]	Section 15.8.2.14

Example:

```
var aNumber = Math.random(-50)    //  MUST throw exception
```

7.8 Date Object

49.15.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor1.xhtml

Assertion 1: When the constructor "new Date()" is called, where no arguments are passed, then it MUST instantiate and return a Date object.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

Assertion 2: When the function "Date()" is called, where no arguments are passed, then it MUST return a string primitive.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.2

Example:

```
var aDate = new Date()    //  aDate is an object
var bDate = Date()    //  bDate is string
```

49.15.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor2.xhtml

Assertion 1: When the constructor "new Date()" is called, where no arguments are passed, then it MUST instantiate and return a Date object, whose value is set to current UTC time.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

Assertion 2: When the function "Date()" is called, where no arguments are passed, then it MUST return a string, whose value is the current UTC time.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.2

Verification: Verify that current time is displayed twice on the page. Verify that display shows both date and time parts.

49.15.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor3.xhtml

Assertion: When the constructor "new Date()" is called, where two numeric values are passed as arguments, then it MUST instantiate and return a Date object, whose year and month values are equal to the arguments passed. The values of the date of the month, hours, minutes, seconds and the milliseconds must be set to 1,0,0,0,0 respectively.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

49.15.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor4.xhtml

Assertion: When the constructor "new Date()" is called, where three numeric values are passed as arguments, then it MUST instantiate and return a Date object, whose year, month and the date of the month values are equal to the arguments passed. The hours, minutes, seconds and the milliseconds must be set to 0,0,0,0 respectively.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

49.15.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor5.xhtml

Assertion: When the constructor "new Date()" is called, where four numeric values are passed as arguments, then it MUST instantiate and return a Date object, whose year, month, the date of the month and hours values are equal to the arguments passed. The minutes, seconds and the milliseconds must be set to 0,0,0 respectively.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

49.15.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor6.xhtml

Assertion: When the constructor "new Date()" is called, where five numeric values are passed as arguments, then it MUST instantiate and return a Date object, whose year, month, the date of the month, hours and minutes values are equal to the arguments passed. The seconds and the milliseconds must be set to 0,0 respectively.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

49.15.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor7.xhtml

Assertion: When the constructor "new Date()" is called, where six numeric values are passed as arguments, then it MUST instantiate and return a Date object, whose year, month, the date of the month, hours, minutes and seconds values are equal to the arguments passed. The milliseconds must be set to 0.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

49.15.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor8.xhtml

Assertion: When the constructor "new Date()" is called, where seven numeric values are passed as arguments, then it MUST instantiate and return a Date object, whose year, month, the date of the month, hours, minutes, seconds and milliseconds values are equal to the arguments passed.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

49.15.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_constructor9.xhtml

Assertion: When the constructor "new Date(X)" is called, where toNumber(X) is valid number, then it MUST instantiate and return a Date object, whose value is a UTC date represented by toNumber(X).

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.2

Example:

var aDate = new Date("0") // aDate is a Date object.

49.15.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_NaN1.xhtml

Assertion: When the constructor "new Date(X)" is called, where toNumber(X) is not finite, then no object MUST be created and a NaN MUST be returned.

Assertion: When the constructor "new Date(X)" is called, where toNumber(X) is greater than 8.64×10^{15}, then no object MUST be created and a NaN MUST be returned.

Assertion: When the constructor "new Date(X)" is called, where toNumber(X) is less than

-8.64×10^{15}, then no object MUST be created and a NaN MUST be returned.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.2, 15.9.1.14

Example:

var aDate = new Date("9 x 10^{15}")   // aDate = NaN.

49.15.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_NaN2.xhtml

Assertion: When any of the **get** methods of a Date object is called (such as **getHours()**), where the date is NaN, then the method must return NaN.

Traceability:

[ESMP]	Section 6.10.5.2 to 6.10.5.10
[ECMA262]	Section 15.9.5.10 to 15.9.5.26

49.15.12

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_function1.xhtml

Assertion: When the function "Date()" is called, where any number of any type of arguments are supplied, then the function must ignore the arguments and return a string representing current UTC time.

Traceability:

[ESMP]	Section 6.10
[ECMA262]	Section 15.9.3.3

Verification: Verify that current time is displayed twice on the page. Verify that display shows both date and time parts.

Example:

```
var aDate = Date(2013, 02, 10, 5, 12, 12, 259)    //  aDate is string of current time.
```

7.8.1 Version History

49.15.13

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-014

Test path: esmp_Sec6/6.10_date/version.xhtml

Assertion: The date object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.10.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

7.8.2 Time Range

49.15.14

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_timerange1.xhtml

Assertion: The exact moment of midnight at the beginning of 01 January, 1970 UTC MUST be represented by the value +0.

Traceability:

[ESMP]	Section 6.10.2
[ECMA262]	Section 15.9.1.1

49.15.15

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_timerange2.xhtml

Assertion: The implementation MUST support time value with a precision of 1 millisecond.

Traceability:

[ESMP]	Section 6.10.2
[ECMA262]	Section 15.9.1.1

49.15.16

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_timerange3.xhtml

Assertion: The implementation MUST support time range of exactly ± 8.64e+15 milliseconds, from 01 January 1970 UTC.

Traceability:

[ESMP]	Section 6.10.2
[ECMA262]	Section 15.9.1.1

7.8.3 Day Number and Time within Day

7.8.4 Properties

No properties are specified for the Date object.

7.8.5 Methods

7.8.5.1 getTime()

49.15.17

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getTime1.xhtml

Assertion: When the **getTime()** method of a date object is called, then the current time value of the object must be returned.

Traceability:

[ESMP]	Section 6.10.5.1
[ECMA262]	Section 15.9.5.9

49.15.18

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getTime2.xhtml

Assertion: When the **getTime()** method is applied to a non-date object, then the implementation MUST throw the TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.1
[ECMA262]	Section 15.9.5.9

7.8.5.2 getFullYear()

49.15.19

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getFullYear1.xhtml

Assertion: When the **getFullYear()** method of a date object is called, then the local time zone based year contained in the date object MUST be returned as integer number.

Traceability:

[ESMP]	Section 6.10.5.2
[ECMA262]	Section 15.9.5.10

49.15.20

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getFullYear2.xhtml

Assertion: When the **getFullYear()** method is applied to a non-date object, then the compiler MUST throw a TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.2
[ECMA262]	Section 15.9.5.10

49.15.21

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getFullYear3.xhtml

Assertion: The **getFullYear()** method of a date object MUST be able to return the integer number in the range 1970 ± ~238, 000

Traceability:

[ESMP]	Section 6.10.5.2
[ECMA262]	Section 15.9.5.10

49.15.22

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCFullYear1.xhtml

Assertion: When the **getUTCFullYear()** method of a date object is called, then the UTC time zone based year contained in the date object MUST be returned as integer number.

Traceability:

[ESMP]	Section 6.10.5.2
[ECMA262]	Section 15.9.5.11

49.15.23

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCFullYear2.xhtml

Assertion: When the **getUTCFullYear()** method is applied to a non-date object, then the compiler MUST throw a TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.2
[ECMA262]	Section 15.9.5.11

49.15.24

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCFullYear3.xhtml

Assertion: The **getUTCFullYear()** method of a date object MUST be able to return the integer number in the range 1978 ± ~238, 000

Traceability:

[ESMP]	Section 6.10.5.2
[ECMA262]	Section 15.9.5.11

7.8.5.3 getMonth()

49.15.25

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getMonth1.xhtml

Assertion: When the **getMonth()** method of a date object is called, then the method MUST return the local time zone based month contained in the date object as an integer number in the range 0 to 11, where 0 is January and 11 is the December.

Assertion: When the **getUTCMonth()** method of a date object is called, then the method MUST return the UTC time zone based month contained in the date object as an integer number in the range 0 to 11, where 0 is January and 11 is the December.

Traceability:

[ESMP]	Section 6.10.5.3
[ECMA262]	Section 15.9.5.12, 15.9.5.13

49.15.26

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getMonth2.xhtml

Assertion: When the **getMonth()** method of a date object is called, then the method MUST return the local time zone based month contained in the date object as an integer number in the range 0 to 11, where 0 is January and 11 is the December.

Assertion: When the **getUTCMonth()** method of a date object is called, then the method MUST return the UTC time zone based month contained in the date object as an integer number in the range 0 to 11, where 0 is January and 11 is the December.

Traceability:

[ESMP]	Section 6.10.5.3
[ECMA262]	Section 15.9.5.12, 15.9.5.13

49.15.27

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getMonth3.xhtml

Assertion: When the **getMonth()** method is applied to a non-date object, then the compiler MUST throw a TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.3
[ECMA262]	Section 15.9.5.12

49.15.28

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCMonth2.xhtml

Assertion: When the **getUTCMonth()** method is applied to a non-date object, then the compiler MUST throw a TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.3
[ECMA262]	Section 15.9.5.13

7.8.5.4 getDate()

49.15.29

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getDate1.xhtml

Assertion: When the **getDate()** method of a date object is called, then the method MUST return the local time zone based date of the month contained in the date object as an integer number in the range 1 to 31.

Assertion: When the **getUTCDate()** method of a date object is called, then the method MUST return the UTC time zone based date of the month contained in the date object as an integer number in the range 1 to 31.

Traceability:

[ESMP]	Section 6.10.5.4
[ECMA262]	Section 15.9.5.14, Section 15.9.5.15

49.15.30

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getDate2.xhtml

Assertion: When the **getDate()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.4
[ECMA262]	Section 15.9.5.14

49.15.31

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCDate2.xhtml

Assertion: When the **getUTCDate()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.4
[ECMA262]	Section 15.9.5.15

7.8.5.5 getDay()

49.15.32

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getDay1.xhtml

Assertion: When the **getDay()** method of a date object is called, then the method MUST return the local time zone based day of the week contained in the date object as an integer number in the range 0 to 6, where 0 represents Sunday and 6 represents Saturday.

Assertion: When the **getUTCDay()** method of a date object is called, then the method MUST return the UTC time zone based day of the week contained in the date object as an integer number in the range 0 to 6, where 0 represents Sunday and 6 represents Saturday.

Traceability:

[ESMP]	Section 6.10.5.5
[ECMA262]	Section 15.9.5.16, Section 15.9.5.17

49.15.33

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getDay2.xhtml

Assertion: When the **getDay()** method is applied to a non-date object, then the compiler MUST throw a TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.5
[ECMA262]	Section 15.9.5.16

49.15.34

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCDay2.xhtml

Assertion: When the **getUTCDay()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.5
[ECMA262]	Section 15.9.5.17

7.8.5.6 getHours()

49.15.35

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getHours1.xhtml

Assertion: When the **get Hours()** method of a date object is called, then the method MUST return the local time zone based hour of the day contained in the date object as an integer number in the range 0 to 23.

Assertion: When the **getUTCHours()** method of a date object is called, then the method MUST return the UTC time zone based hour of the day contained in the date object as an integer number in the range 0 to 23.

Traceability:

[ESMP]	Section 6.10.5.6
[ECMA262]	Section 15.9.5.18, Section 15.9.5.19

49.15.36

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getHours2.xhtml

Assertion: When the **getHours()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.6
[ECMA262]	Section 15.9.5.18

49.15.37

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCHours2.xhtml

Assertion: When the **getUTCHours()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.6
[ECMA262]	Section 15.9.5.19

7.8.5.7 getMinutes()

49.15.38

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getMinutes1.xhtml

Assertion: When the **get Minutes()** method of a date object is called, then the method MUST return the local time zone based minute within the hour contained in the date object as an integer number in the range 0 to 59.

Assertion: When the **getUTCMinutes()** method of a date object is called, then the method MUST return the UTC time zone based minute within the hour contained in the date object as an integer number in the range 0 to 59.

Traceability:

[ESMP]	Section 6.10.5.7
[ECMA262]	Section 15.9.5.20, Section 15.9.5.21

49.15.39

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getMinutes2.xhtml

Assertion: When the **getMinutes()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.7
[ECMA262]	Section 15.9.5.20

49.15.40

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCMinutes2.xhtml

Assertion: When the **getUTCMinutes()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.7
[ECMA262]	Section 15.9.5.21

7.8.5.8 getSeconds()

49.15.41

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getSeconds1.xhtml

Assertion: When the **getSeconds()** method of a date object is called, then the method MUST return the local time zone based second within the minute contained in the date object as an integer number in the range 0 to 59.

Assertion: When the **getUTCSeconds()** method of a date object is called, then the method MUST return the UTC time zone based second within the minute contained in the date object as an integer number in the range 0 to 59.

Traceability:

[ESMP]	Section 6.10.5.8
[ECMA262]	Section 15.9.5.22, Section 15.9.5.23

49.15.42

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getSeconds2.xhtml

Assertion: When the **getSeconds()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.8
[ECMA262]	Section 15.9.5.22

49.15.43

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCSeconds2.xhtml

Assertion: When the **getUTCSeconds()** method is applied to a non-date object, then the compiler MUST throw a **TypeError** exception.

Traceability:

[ESMP]	Section 6.10.5.8
[ECMA262]	Section 15.9.5.23

7.8.5.9 getMilliseconds()

49.15.44

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getMilliseconds1.xhtml

Assertion: When the **getMilliSeconds()** method of a date object is called, then the method MUST return the local time zone based millisecond within the second contained in the date object as an integer number in the range 0 to 999.

Assertion: When the **getUTCMilliSeconds()** method of a date object is called, then the method MUST return the UTC time zone based millisecond within the second contained in the date object as an integer number in the range 0 to 999.

Traceability:

[ESMP]	Section 6.10.5.9
[ECMA262]	Section 15.9.5.24, Section 15.9.5.25

49.15.45

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getMilliseconds2.xhtml

Assertion: When the **getMilliSeconds()** method is applied to a non-date object, then the compiler MUST throw a TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.9
[ECMA262]	Section 15.9.5.24

49.15.46

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getUTCMilliseconds2.xhtml

Assertion: When the **getUTCMilliSeconds()** method is applied to a non-date object, then the compiler MUST throw a TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.9
[ECMA262]	Section 15.9.5.25

7.8.5.10 getTimezoneOffset()

49.15.47

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getTimezoneOffset1.xhtml

Assertion: When the **getTimezoneOffset()** method of a date object is called, then the method MUST return the value in minutes, which is the difference between the current time zone and UTC, as an integer number in the range 0 to 1410.

Traceability:

[ESMP]	Section 6.10.5.10
[ECMA262]	Section 15.9.5.26

49.15.48

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_getTimezoneOffset2.xhtml

Assertion: When the target of the **getTimezoneOffset()** method is not a date object, then the compiler MUST throw a TypeError exception.

Traceability:

[ESMP]	Section 6.10.5.10
[ECMA262]	Section 15.9.5.26

7.8.5.11 parse()

The **parse()** method of date object is not supported by [ESMP].

7.8.5.12 UTC()

49.15.49

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_utc1.xhtml

Assertion: When the function **Date.UTC(**year, month, date, hours, minutes, seconds, ms**)** is called, where year is a number with full year value, month is a number in the range 0-11, date is a number in the range 0-31, hours is a number in the range 0-23, minutes is a number in the range 0-59 and ms is a number in the range 0-999, then the function MUST compute a UTC date from the arguments supplied. The function MUST return an integer number, which corresponds to the UTC date value D, in milliseconds from the epoch.

Assertion: When the function **Date.UTC(**year, month**)** is called, where year is a number with full year value and month is a number in the range 0-11, then the function MUST compute a UTC date D from the year, month supplied and using 1 for date, 0 for hours, 0 for minutes, 0 for seconds, 0 for ms. The function MUST return an integer number which corresponds to the UTC date value D, in milliseconds from the epoch.

Traceability:

[ESMP]	Section 6.10.5.12
[ECMA262]	Section 15.9.4.3

7.8.5.13 setTime()

49.15.50

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setTime1.xhtml

Assertion: When the **setTime()** method of a Date object D is called, where a signed integer N is passed as argument, then the method MUST assign the time represented by the input value, as the time for target Date object D.

Traceability:

[ESMP]	Section 6.10.5.13
[ECMA262]	Section 15.9.5.27

49.15.51

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setTime2.xhtml

Assertion: When the **setTime()** method is applied to non-date object, then a TypeError exception MUST be generated by the implementation.

Traceability:

[ESMP]	Section 6.10.5.13
[ECMA262]	Section 15.9.5.27

7.8.5.14 setFullYear()

49.15.52

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setFullYear1.xhtml

Assertion: When the **setFullYear()** method of a Date object D is called, where an integer number Y is supplied as argument, then the method MUST set the year of the target Date object D to value Y, using the local time.

Assertion: When the **setFullYear()** method of a Date object D is called, then the method MUST set the year of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.14
[ECMA262]	Section 15.9.5.40

49.15.53

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setFullYear2.xhtml

Assertion: When the **setFullYear()** method of a Date object D is called, where an integer number Y and integer number M are supplied as arguments, then the method MUST set the year of the target Date object D to value Y and month to value M, using the local time.

Traceability:

[ESMP]	Section 6.10.5.14
[ECMA262]	Section 15.9.5.40

49.15.54

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setFullYear3.xhtml

Assertion: When the **setFullYear()** method of a Date object D is called, where an integer number Y, integer number M and an integer number N are supplied as arguments, then the method MUST set the year of the target Date object D to value Y, month to value M, and date of the month to value N, using the local time.

Traceability:

[ESMP]	Section 6.10.5.14
[ECMA262]	Section 15.9.5.40

49.15.55

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_NaN3.xhtml

Assertion: When the **setFullYear()** method of a Date object D is called, where any of the arguments are out of range of a signed 64 bit integer, then the date of the Date object D MUST be set to NaN.

Assertion: When the **setFullYear()** method of a Date object D is called, where the resultant date can not be expressed within a signed 64 bit integer, then the date of the Date object D MUST be set to NaN.

Note: Above two assertions apply similarly for these methods also: setUTCFullYear(), setMonth(), setUTCMonth(), setDate(), setUTCDate().

Traceability:

[ESMP]	Section 6.10.5.14
[ECMA262]	Section 15.9.5.40

49.15.56

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCFullYear1.xhtml

Assertion: When the **setUTCFullYear()** method of a Date object D is called, where an integer number Y is supplied as argument, then the method MUST set the year of the target Date object D to value Y, using Universal Coordinated Time.

Assertion: When the **setUTCFullYear()** method of a Date object D is called, then the method MUST set the year of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.14
[ECMA262]	Section 15.9.5.41

49.15.57

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCFullYear2.xhtml

Assertion: When the **setUTCFullYear()** method of a Date object D is called, where an integer number Y and integer number M are supplied as arguments, then the method MUST set the year of the target Date object D to value Y and month to value M, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.14
[ECMA262]	Section 15.9.5.41

49.15.58

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCFullYear3.xhtml

Assertion: When the **setUTCFullYear()** method of a Date object D is called, where an integer number Y, integer number M and an integer number N are supplied as arguments, then the method MUST set the year of the target Date object D to value Y, month to value M, and date of the month to value N, using the Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.14
[ECMA262]	Section 15.9.5.41

7.8.5.15 setMonth()

49.15.59

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setMonth1.xhtml

Assertion: When the **setMonth()** method of a Date object D is called, where a positive integer number M is supplied as argument, then the method MUST set the month of the target Date object D to value M, using the local time.

Assertion: When the **setMonth()** method of a Date object D is called, where a negative integer number M is supplied as argument, then the method MUST set the month of the target Date object D to value '12-M', using the local time.

Assertion: When the **setMonth()** method of a Date object D is called, then the method MUST set the month of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.15
[ECMA262]	Section 15.9.5.38

49.15.60

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setMonth2.xhtml

Assertion: When the **setMonth()** method of a Date object D is called, where an integer number M and an integer number N are supplied as arguments, then the method MUST set the month of the target Date object D to value M, and date of the month to value N, using the local time.

Traceability:

[ESMP]	Section 6.10.5.15
[ECMA262]	Section 15.9.5.38

49.15.61

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCMonth1.xhtml

Assertion: When the **setUTCMonth()** method of a Date object D is called, where an integer number M is supplied as argument, then the method MUST set the month of the target Date object D to value M, using the Universal Coordinated Time.

Assertion: When the **setUTCMonth()** method of a Date object D is called, then the method MUST set the month of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.15
[ECMA262]	Section 15.9.5.39

49.15.62

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCMonth2.xhtml

Assertion: When the **setUTCMonth()** method of a Date object D is called, where an integer number M and an integer number N are supplied as arguments, then the method MUST set the month of the target Date object D to value M, and date of the month to value N, using the Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.15
[ECMA262]	Section 15.9.5.39

7.8.5.16 setDate()

49.15.63

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setDate1.xhtml

Assertion: When the **setDate()** method of a Date object D is called, where an integer number N is supplied as argument, then the method MUST set the 'date of the month' of the target Date object D to value N, using the local time.

Assertion: When the **setDate()** method of a Date object D is called, then the method MUST set the 'date of the month' of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.16
[ECMA262]	Section 15.9.5.36

49.15.64

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCDate1.xhtml

Assertion: When the **setUTCDate()** method of a Date object D is called, where an integer number N is supplied as argument, then the method MUST set the 'date of the month' of the target Date object D to value N, using Universal Coordinated Time.

Assertion: When the **setUTCDate()** method of a Date object D is called, then the method MUST set the 'date of the month' of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.16
[ECMA262]	Section 15.9.5.37

7.8.5.17 setHours()

49.15.65

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setHours1.xhtml

Assertion: When the **setHours()** method of a Date object D is called, where an integer number H is supplied as argument, then the method MUST set the hour of the target Date object D to value H, using the local time.

Assertion: When the **setHours()** method of a Date object D is called, then the method MUST set the hour of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.34

49.15.66

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setHours2.xhtml

Assertion: When the **setHours()** method of a Date object D is called, where an integer number H is supplied as argument whose value is 24, then the method MUST set the hour of the target Date object D to value 0 and increment the day count, using the local time.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.34

49.15.67

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setHours3.xhtml

Assertion: When the **setHours()** method of a Date object D is called, where an integer number H and integer number M are supplied as arguments, then the method MUST set the hour of the target Date object D to value H and minutes to value M, using the local time.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.34

49.15.68

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setHours4.xhtml

Assertion: When the **setHours()** method of a Date object D is called, where an integer number H, integer number M and an integer number S are supplied as arguments, then the method MUST set the hour of the target Date object D to value H, minutes to value M, and seconds to value S, using the local time.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.34

49.15.69

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setHours5.xhtml

Assertion: When the **setHours()** method of a Date object D is called, where an integer number H, integer number M, integer number S and an integer number mS are supplied as arguments, then the method MUST set the hour of the target Date object D to value H, minutes to value M, seconds to value S, and milliseconds to value mS, using the local time.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.34

49.15.70

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_NaN4.xhtml

Assertion: When the **setHours()** method of a Date object D is called, where any of the arguments are out of range of a signed 64 bit integer, then the date of the Date object D MUST be set to NaN.

Assertion: When the **setHours()** method of a Date object D is called, where the resultant date can not be expressed within a signed 64 bit integer, then the date of the Date object D MUST be set to NaN.

Note: Above two assertions apply similarly for these methods also: setUTCHours(), setMinutes(), setUTCMinutes(), setSeconds(), setUTCSeconds(), setMilliSeconds(), setUTCMilliSeconds().

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.34

49.15.71

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCHours1.xhtml

Assertion: When the **setUTCHours()** method of a Date object D is called, where an integer number H is supplied as argument, then the method MUST set the hour of the target Date object D to value H, using Universal Coordinated Time.

Assertion: When the **setUTCHours()** method of a Date object D is called, then the method MUST set the hour of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.35

49.15.72

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCHours2.xhtml

Assertion: When the **setUTCHours()** method of a Date object D is called, where an integer number H is supplied as argument whose value is 24, then the method MUST set the hour of the target Date object D to value 0 and increment the day count, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.35

49.15.73

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCHours3.xhtml

Assertion: When the **setUTCHours()** method of a Date object D is called, where an integer number H and integer number M are supplied as arguments, then the method MUST set the hour of the target Date object D to value H and minutes to value M, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.35

49.15.74

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCHours4.xhtml

Assertion: When the **setUTCHours()** method of a Date object D is called, where an integer number H, integer number M and an integer number S are supplied as arguments, then the method MUST set the hour of the target Date object D to value H, minutes to value M, and seconds to value S, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.35

49.15.75

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCHours5.xhtml

Assertion: When the **setUTCHours()** method of a Date object D is called, where an integer number H, integer number M, integer number S and an integer number mS are supplied as arguments, then the method MUST set the hour of the target Date object D to value H, minutes to value M, seconds to value S, and milliseconds to value mS, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.17
[ECMA262]	Section 15.9.5.35

7.8.5.18 setMinutes()

49.15.76

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setMinutes1.xhtml

Assertion: When the **setMinutes()** method of a Date object D is called, where an integer number M is supplied as argument, then the method MUST set the minutes of the target Date object D to value M, using the local time.

Assertion: When the **setMinutes()** method of a Date object D is called, then the method MUST set the minutes of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.18
[ECMA262]	Section 15.9.5.32

49.15.77

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setMinutes2.xhtml

Assertion: When the **setMinutes()** method of a Date object D is called, where a negative integer number M is supplied as argument, then the method MUST set the hour of the target Date object D to value 60-M and decrement the hour, using the local time.

Traceability:

[ESMP]	Section 6.10.5.18
[ECMA262]	Section 15.9.5.32

49.15.78

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setMinutes3.xhtml

Assertion: When the **setMinutes()** method of a Date object D is called, where an integer number M and an integer number S are supplied as arguments, then the method MUST set the minutes of the target Date object D to value M and seconds to value S, using the local time.

Traceability:

[ESMP]	Section 6.10.5.18
[ECMA262]	Section 15.9.5.32

49.15.79

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setMinutes4.xhtml

Assertion: When the **setMinutes()** method of a Date object D is called, where an integer number M, integer number S and an integer number mS are supplied as arguments, then the method MUST set the minutes of the target Date object D to value M, seconds to value S and milli-seconds to value mS, using the local time.

Traceability:

[ESMP]	Section 6.10.5.18
[ECMA262]	Section 15.9.5.32

49.15.80

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCMinutes1.xhtml

Assertion: When the **setUTCMinutes()** method of a Date object D is called, where an integer number M is supplied as argument, then the method MUST set the minutes of the target Date object D to value M, using Universal Coordinated Time.

Assertion: When the **setUTCMinutes()** method of a Date object D is called, then the method MUST set the minutes of the target Date object D and MUST return the value of the newly calculated value.

Traceability:

[ESMP]	Section 6.10.5.18
[ECMA262]	Section 15.9.5.33

49.15.81

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCMinutes2.xhtml

Assertion: When the **setUTCMinutes()** method of a Date object D is called, where a negative integer number M is supplied as argument, then the method MUST set the hour of the target Date object D to value 60-M and decrement the hour, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.18
[ECMA262]	Section 15.9.5.33

49.15.82

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCMinutes3.xhtml

Assertion: When the **setUTCMinutes()** method of a Date object D is called, where an integer number M and an integer number S are supplied as arguments, then the method MUST set the minutes of the target Date object D to value M and seconds to value S, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.18
[ECMA262]	Section 15.9.5.33

49.15.83

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCMinutes4.xhtml

Assertion: When the **setUTCMinutes()** method of a Date object D is called, where an integer number M, integer number S and an integer number mS are supplied as arguments, then the method MUST set the minutes of the target Date object D to value M, seconds to value S and milliseconds to value mS, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.18
[ECMA262]	Section 15.9.5.33

7.8.5.19 setSeconds()

49.15.84

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setSeconds1.xhtml

Assertion: When the **setSeconds()** method of a Date object D is called, where an integer number S is supplied as argument, then the method MUST set the seconds of the target Date object D to value S, using the local time.

Assertion: When the **setSeconds()** method of a Date object D is called, then the method MUST set the seconds of the target Date object D to value S, using the local time and MUST return the newly calculated date.

Traceability:

[ESMP]	Section 6.10.5.19
[ECMA262]	Section 15.9.5.30

49.15.85

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setSeconds2.xhtml

Assertion: When the **setSeconds()** method of a Date object D is called, where an integer number S and an integer number mS are supplied as arguments, then the method MUST set the seconds of the target Date object D to value S and milliseconds to value mS, using the local time.

Traceability:

[ESMP]	Section 6.10.5.19
[ECMA262]	Section 15.9.5.30

49.15.86

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCSeconds1.xhtml

Assertion: When the **setUTCSeconds()** method of a Date object D is called, where an integer number S is supplied as argument, then the method MUST set the seconds of the target Date object D to value S, using Universal Coordinated Time.

Assertion: When the **setUTCSeconds()** method of a Date object D is called, then the method MUST set the seconds of the target Date object D to value S, using Universal Coordinated Time and MUST return the newly calculated date.

Traceability:

[ESMP]	Section 6.10.5.19
[ECMA262]	Section 15.9.5.31

49.15.87

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setUTCSeconds2.xhtml

Assertion: When the **setUTCSeconds()** method of a Date object D is called, where an integer number S and an integer number mS are supplied as arguments, then the method MUST set the seconds of the target Date object D to value S and milliseconds to value mS, using Universal Coordinated Time.

Traceability:

[ESMP]	Section 6.10.5.19
[ECMA262]	Section 15.9.5.31

7.8.5.20 setMilliseconds()

49.15.88

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_setMilliseconds1.xhtml

Assertion: When the **setMilliSeconds()** method of a Date object D is called, where an integer number S is supplied as argument, then the method MUST set the milliseconds of the target Date object D to value S, using the local time.

Assertion: When the **setMilliSeconds()** method of a Date object D is called, then the method MUST set the milliseconds of the target Date object D to value S, using the local time and MUST return the newly calculated date.

Traceability:

[ESMP]	Section 6.10.5.20
[ECMA262]	Section 15.9.5.28

7.8.5.21 toString()

49.15.89

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_toString1.xhtml

Assertion: When the **toString()** method of a Date object D is called, then the method MUST return the full date of the date object D, in a human readable string form in the local time zone.

Traceability:

[ESMP]	Section 6.10.5.21
[ECMA262]	Section 15.9.5.2

Verification: Verify that current time based on local time zone is displayed. Verify that both date and time portions are displayed.

49.15.90

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_toLocaleString1.xhtml

Assertion: When the **toLocaleString()** method of a Date object D is called, then the method MUST return the full date of the date object D, in a human readable string form in the current time zone in the current locale format.

Traceability:

[ESMP]	Section 6.10.5.21
[ECMA262]	Section 15.9.5.2

Verification: Verify that current time based on local time zone is displayed. Verify that both date and time portions are displayed.

49.15.91

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_toUTCString1.xhtml

Assertion: When the **toUTCString()** method of a Date object D is called, then the method MUST return the full date of the date object D, in a human readable string form in UTC convention.

Traceability:

[ESMP]	Section 6.10.5.21
[ECMA262]	Section 15.9.5.42

Verification: Verify that current UTC time is displayed. Verify that both date and time portions are displayed.

49.15.92

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_NaN5.xhtml

Assertion: When the **toString()** method of a NaN Date object is called, then the method MUST return NaN.

Note: Above two assertions apply similarly for these methods also: toLocaleString(), toUTCString(), toDateString(), toLocaleDateString(),toTimeString(),toLocaleTimeString().

Traceability:

[ESMP]	Section 6.10.5.21
[ECMA262]	Section 15.9.5.2

7.8.5.22 toDateString()

49.15.93

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_toDateString1.xhtml

Assertion: When the **toDateString()** method of a Date object D is called, then the method MUST return only the date part of the date object D, in a human readable string form in the local time zone.

Traceability:

[ESMP]	Section 6.10.5.22
[ECMA262]	Section 15.9.5.3

Verification: Verify that only date portion of the current local time is displayed.

49.15.94

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_toLocaleDateString1.xhtml

Assertion: When the **toLocaleDateString()** method of a Date object D is called, then the method MUST return only the date part of the date object D, in a human readable string form in the current time zone in the current locale format.

Traceability:

[ESMP]	Section 6.10.5.22
[ECMA262]	Section 15.9.5.6

Verification: Verify that only date portion of the current local time is displayed.

7.8.5.23 toTimeString()

49.15.95

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_toTimeString1.xhtml

Assertion: When the **toTimeString()** method of a Date object D is called, then the method MUST return time part of the date object D, in a human readable string form in the local time zone.

Traceability:

[ESMP]	Section 6.10.5.23
[ECMA262]	Section 15.9.5.4

Verification: Verify that only time portion of the current local time is displayed.

49.15.96

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_toLocaleTimeString1.xhtml

Assertion: When the **toLocaleTimeString()** method of a Date object D is called, then the method MUST return the time part of the date object D, in a human readable string form in the current time zone in the current locale format.

Traceability:

[ESMP]	Section 6.10.5.23
[ECMA262]	Section 15.9.5.7

Verification: Verify that only time portion of the current local time is displayed.

7.8.5.24 valueOf()

49.15.97

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-024

Test path: esmp_Sec6/6.10_date/date_valueOf1.xhtml

Assertion: When the **valueOf()** method of a Date object is called, then the method MUST return a signed integer, which represents the date contained in the date object.

Assertion: When the **valueOf()** method of a Date object is called, where the date contained in the date object is not prior to "01 January, 1970 UTC" then the method MUST return a positive integer, which represents this date.

Assertion: When the **valueOf()** method of a Date object is called, where the date contained in the date object is prior to "01 January, 1970 UTC" then the method MUST return a negative integer, which represents this date.

Assertion: When the **valueOf()** method of a Date object is called, where the date contained in the date object is The exact moment of midnight at the beginning of 01 January, 1970 UTC, then the method MUST return a +0.

Traceability:

[ESMP]	Section 6.10.5.24
[ECMA262]	Section 15.9.5.8, 15.9.1.1

7.9 Error (Exception) Object

7.9.1 Version History

49.16.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-014

Test path: esmp_Sec6/6.11_error/version.xhtml

Assertion: The **error** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 6.11.1, Sec 6.2.1

7.9.2 Constructor

49.16.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/error_constructor1.xhtml

Assertion 1: The “new Error();” constructor MUST create and return an instance of error object

Traceability:

[ESMP]	Section 6.11.2
[ECMA262]	Section 15.11.1

Assertion 2: The initial value of a newly constructed Error object **name** MUST be the string “Error”

Traceability:

[ESMP]	Section 6.11.3.1

[ECMA262]	Section 15.11.7

Assertion 3: The **message** property of a newly constructed error object MUST contain a string whose value is implementation specific.

Traceability:

[ESMP]	Section 6.11.3.2
[ECMA262]	Section 15.11.7

Assertion 4: The initial value of the **code** property of a newly constructed error object MUST be NaN.

Traceability:

[ESMP]	Section 6.11.3.3
[ECMA262]	Section 15.11.7

7.9.3 Properties

7.9.3.1 name

49.16.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/error_name1.xhtml

Assertion: It MUST be possible to re-set the **name** property of an Error object with a one-word string value.

Traceability:

[ESMP]	Section 6.11.3.1
[ECMA262]	Section 15.11.7

7.9.3.2 message

49.16.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/error_message1.xhtml

Assertion: When a string M representing a message is passed as argument with the “new Error();” constructor, then an instance of error object MUST be created and the **message** property of the created error object MUST be set to the supplied string message M.

Traceability:

[ESMP]	Section 6.11.2
[ECMA262]	Section 15.11.1

7.9.3.3 code

49.16.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/error_code1.xhtml

Assertion: The **code** property of an error object MUST be an integer, which is the value of the exception thrown.

Traceability:

[ESMP]	Section 6.11.3.3
[ECMA262]	Section 15.11.7

7.9.4 Native Error Types (Constants)

49.16.06

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/RangeError.xhtml

Assertion 1: When a numeric value exceeds its range, then compiler MUST throw the native error object, whose name is “RangeError” and code value is 101.

Assertion 2: The implementation MUST be able to throw instantiation of any of the native Error objects.

Assertion 3: It MUST be possible to catch and process any of the native error objects thrown by the implementation, using try/catch {} clause. The catch parameter MUST be set with the reference to the error object thrown by the implementation.

Traceability:

[ESMP]	Section 6.11.4
[ECMA262]	Section 15.11.7

49.16.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/ReferenceError.xhtml

Assertion 1: When an invalid reference is detected, then compiler MUST throw the native error object, whose name is “ReferenceError” and code value is 102.

Assertion 2: The implementation MUST be able to throw instantiation of any of the native Error objects.

Assertion 3: It MUST be possible to catch and process any of the native error objects thrown by the implementation, using try/catch {} clause. The catch parameter MUST be set with the reference to the error object thrown by the implementation.

Traceability:

[ESMP]	Section 6.11.4
[ECMA262]	Section 15.11.7

49.16.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/SyntaxError.xhtml

Assertion 1: When a parsing error occurs, then compiler MUST throw the native error object, whose name is “SyntaxError” and code value is 103.

Assertion 2: The implementation MUST be able to throw instantiation of any of the native Error objects.

Assertion 3: It MUST be possible to catch and process any of the native error objects thrown by the implementation, using try/catch {} clause. The catch parameter MUST be set with the reference to the error object thrown by the implementation.

Traceability:

[ESMP]	Section 6.11.4
[ECMA262]	Section 15.11.7

49.16.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/TypeError.xhtml

Assertion 1: When actual type of an operand is different than expected type, then compiler MUST throw the native error object, whose name is “TypeError” and code value is 104.

Assertion 2: The implementation MUST be able to throw instantiation of any of the native Error objects.

Assertion 3: It MUST be possible to catch and process any of the native error objects thrown by the implementation, using try/catch {} clause. The catch parameter MUST be set with the reference to the error object thrown by the implementation.

Traceability:

[ESMP]	Section 6.11.4
[ECMA262]	Section 15.11.7

49.16.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/URIError.xhtml

Assertion 1: When a global URI function is used in an inconsistent or incompatible way, then compiler MUST throw the native error object, whose name is “URIError” and code value is 105.

Assertion 2: The implementation MUST be able to throw instantiation of any of the native Error objects.

Assertion 3: It MUST be possible to catch and process any of the native error objects thrown by the implementation, using try/catch {} clause. The catch parameter MUST be set with the reference to the error object thrown by the implementation.

Traceability:

[ESMP]	Section 6.11.4
[ECMA262]	Section 15.11.7

49.16.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-025

Test path: esmp_Sec6/6.11_error/EvalError.xhtml

Assertion 1: When the implementation does not support global eval() function and an attempt is made to use this function, then compiler MUST throw the native error object, whose name is “EvalError” and code value is 106.

Assertion 2: The implementation MUST be able to throw instantiation of any of the native Error objects.

Assertion 3: It MUST be possible to catch and process any of the native error objects thrown by the implementation, using try/catch {} clause. The catch parameter MUST be set with the reference to the error object thrown by the implementation.

Traceability:

[ESMP]	Section 6.11.4
[ECMA262]	Section 15.11.7

7.10 Unsupported Native Objects

7.10.1 Object Object

49.17.01

Requirement: Optional

SCR: ESMP-LANGUAGE-C-026

Test path: esmp_Sec6/6.12_unsupported/Object_support1.xhtml

Assertion: When the Object constructor is called, where a value is passed as the argument, then the result MUST be as per the following table.

Input	Result
Undefined	Create an object, without any value.
Null	Create an object, without any value.
Boolean true	Create a Boolean object, whose value is true.
Boolean false	Create a Boolean object, whose value is false.
Number	Create a Number object, whose value is the value of the input number.
String	Create a String object, whose value is value of the input string.
Object	Same as input, no conversion.

Traceability:

[ESMP]	Section 6.12.1
[ECMA262]	Section 15.2

49.17.02

Requirement: Optional

SCR: ESMP-LANGUAGE-C-013

Test path: esmp_Sec6/6.12_unsupported/Object_no_modi.xhtml

Assertion: If the implementation does not support modification to the properties of built-in objects, then it MUST throw a ReferenceError exception, when an attempt is made to do so.

Traceability:

[ESMP]	Section 6
[ECMA327]	Section 5.2

7.10.2 Function Object

49.17.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-028

Test path: esmp_Sec6/6.12_unsupported/Function_no_support1.xhtml

Assertion: When the **Function** constructor is called, and if the implementation does not support dynamic creation of functions, then compiler MUST throw the EvalError exception.

Traceability:

[ESMP]	Section 6.12.2
[ECMA327]	Section 5.1

49.17.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-028

Test path: esmp_Sec6/6.12_unsupported/Function_no_support2.xhtml

Assertion: When the **Function** function is called, and if the implementation does not support dynamic creation of functions, then compiler MUST throw the EvalError exception.

Traceability:

[ESMP]	Section 6.12.2
[ECMA327]	Section 5.1

49.17.05

Requirement: Optional

SCR: ESMP-LANGUAGE-C-027

Test path: esmp_Sec6/6.12_unsupported/Function_support1.xhtml

Assertion: When the **Function** constructor is called, then a **Function** object MUST be created and initialized.

Traceability:

[ESMP]	Section 6.12.2
[ECMA262]	Section 15.3

49.17.06

Requirement: Optional

SCR: ESMP-LANGUAGE-C-027

Test path: esmp_Sec6/6.12_unsupported/Function_support2.xhtml

Assertion: When the function **Function** is called, then a **Function** object MUST be created and initialized.

Traceability:

[ESMP]	Section 6.12.2
[ECMA262]	Section 15.3

8. THE LANGUAGE ENVIRONMENT

8.1 Script Context

The script context is the aggregation of the current browser context and the document context. The browser context is composed of the history, location and event state associated with the currently executing document. The document context is accessed via the browser Document object. The tests included in the sections 9 and 10 of this test plan covers the browser context and document context testing indirectly.

8.2 Script Invocation Mechanisms

49.19.01

Requirement: Mandatory

SCR: None

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/NoInvoke1.xhtml

Assertion: The script invocation by directly navigating to a script document MUST not be supported.

Traceability:

[ESMP]	Section 7.2.1

Verification: Load the page. Activate the link "link". Test is failed, if an alert message saying "Error, if you get this alert message" appears. It is OK, if the script source code is displayed as plain text.

49.19.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-029

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/noscript1.xhtml

Assertion: When the scripting language specified in the <script> element is not supported by the implementation and if one or more **<noscript>** blocks are present in the document, then all of the **<noscript>** content MUST be processed.

Assertion: The user-agent must not process **<noscript>** elements, until it has encountered a **<script>** element that specifies an unsupported scripting language.

Traceability:

[ESMP]	Section 7.2.2
[XHTMLMP11]	Section 9.1.2

Verification: Verify that the word “Success” is displayed twice. Verify that the word “ERROR” is not displayed.

49.19.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-029

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/noscript2.xhtml

Assertion: The user-agent must continue to process **<script>** elements, even after it has found a **<script>** element that specifies an unsupported scripting language.

Traceability:

[ESMP]	Section 7.2.2
[XHTMLMP11]	Section 9.1.2

Verification: Verify that the word “Success” is displayed three times.

49.19.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-029

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/noscript3.xhtml

NOTE: This test applies to only to those user-agents, which do not support scripting, or whose script support is disabled. Turn the script support to OFF before running this test.

Assertion: When the user-agent does not support script, then it must process all of the **<noscript>** blocks present in the document.

Traceability:

[ESMP]	Section 7.2.2
[XHTMLMP11]	Section 9.1.2

Verification: Verify that the word “Success” is displayed twice. Verify that the word “ERROR” is not displayed.

49.19.05

Requirement: Mandatory

SCR:

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/html_comment.xhtml

Assertion: While executing script, the user-agent must ignore an HTML comment tag "<!--" that occurs in the first line after <script> tag.

Traceability:

[ESMP]	Section 7.2.2
[XHTMLMP11]	Section 9.1.2

Verification: Verify that the word "Success" is displayed.

49.19.06

SCR:

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/script_src1.xhtml

Assertion: When the external script obtained via the “src” attribute is of any of the following three media types, then that script content must be processed.

- text/javascript
- text/ecmascript

Traceability:

49.19.07

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-029

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/InlineExec.xhtml

Assertion 1: The directly embedded inline script segments into the mark-up document without using a function body MUST be directly invoked.

Traceability:

[ESMP]	Section 7.2.2.1

Assertion 2: The script invocation MUST occur during the loading of the mark-up document.

Traceability:

[ESMP]	Section 7.2

Verification: Load the page. Verify that the text “Test passed” is displayed.

49.19.08

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-029

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/MultiInlineExec.xhtml

Assertion: Multiple direct embedded script segments MUST be supported.

Traceability:

[ESMP]	Section 7.2.2.1

Verification: Load the page. Verify that the text “SUCCESS” is displayed in three consecutive lines.

49.19.09

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-029

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/OrderInvoke.xhtml

Assertion: Directly invoked script segments MUST be executed in the order in which they appear.

Traceability:

[ESMP]	Section 7.2.2.1

Verification: Load the page. Verify that the text "SUCCESS" is displayed in the page.

49.19.10

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-029

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/DirectVsEventInvoke.xhtml

Assertion: Directly invoked script segments MUST be executed prior to any event invoked script segments.

Traceability:

[ESMP]	Section 7.2.2.1

Assertion: The script engine MUST allow deferred execution scripts to be located in the <head> of a document.

Traceability:

[ESMP]	Section 7.2.2.2

Verification: Load the page. Verify that the text "Default" is displayed in the input box. Activate the link "link". Verify that the text in the input box changes to "SUCCESS".

49.19.11

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-030

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/BodyEventBasedExec.xhtml

Assertion: The script engine MUST allow deferred execution scripts to be located in the <body> of a document.

Traceability:

[ESMP]	Section 7.2.2.2

Verification: Load the page. Verify that the text “Default” is displayed in the input box. Activate the link “link”. Verify that the text in the input box changes to “SUCCESS”.

49.19.12

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-031

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/FileBasedExec1.xhtml

Assertion: Defining the script content by referencing a URL that contains the source for the script to execute MUST be supported. This means remotely referenced script files MUST be supported.

Traceability:

[ESMP]	Section 7.2.2.3

Verification: Load the page. Move the focus onto input field. Verify that the text “SUCCESS” is displayed in the input field.

49.19.13

Requirement: Mandatory

SCR:

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/FileBasedExec2.xhtml

Assertion: The implementation must support the execution of external scripts of the media type “application/x-javascript”.

Traceability:

Verification: Load the page. Move the focus onto input field. Verify that the text “SUCCESS” is displayed in the input field.

49.19.14

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-031

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/FileBasedExec3.xhtml

Assertion: The implementation must support the execution of external scripts of the media type “text/javascript”.

Traceability:

[ESMP]	Section 7.2.2.3

Verification: Load the page. Move the focus onto input field. Verify that the text “SUCCESS” is displayed in the input field.

49.19.15

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-031

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/FileBasedExec4.xhtml

Assertion: The implementation must support the execution of external scripts of the media type “text/ecmascript”.

Traceability:

[ESMP]	Section 7.2.2.3

Verification: Load the page. Move the focus onto input field. Verify that the text “SUCCESS” is displayed in the input field.

49.19.16

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-031

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/remoteScript.xhtml

Assertion: The script engine MUST allow script elements representing references to remote scripts to be located in either the <body> or the <head> of a document.

Traceability:

[ESMP]	Section 7.2.2.3

Verification: Load the page. Verify that the text “This test passed." is displayed.

49.19.17

Requirement: Mandatory

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/schemeBased1.xhtml

Assertion: The supported script functions assigned directly as values of the supported event attributes must be executed on triggering those events.

Verification: Load the page. Activate the “link”. Verify that alert message “SUCCESS” appears.

49.19.18

Requirement: Mandatory

Test path: esmp_Sec7/7.2_Scr_Inv_Mech/schemeBased2.xhtml

Assertion: Scheme-based invocation, (using a scheme syntax such as href=”ecmascript: <immediate statements>“) MUST be supported.

Verification: Load the page. Activate the “link”. Verify that alert message “SUCCESS” appears.

8.3 Script Completion Mechanisms

49.20.01

Requirement: Mandatory

SCR: None

Test path: esmp_Sec7/7.3_Scr_Comp_Mech/cancelNav1.xhtml

Assertion: When the script requests navigation, completes and returns to parent document context, and if the user invokes navigation cancel command, then the script's navigation request MUST be cancelled.

Traceability:

[ESMP]	Section 7.3.1

Verification: Load the page. Activate the "Button". Verify that this starts navigation to another page. Immediately cancel the navigation. Verify that "Page 1" does not get loaded.

49.20.02

Requirement: Mandatory

SCR: None

Test path: esmp_Sec7/7.3_Scr_Comp_Mech/cancelNav2.xhtml

Assertion: When the script requests navigation, completes and returns to parent document context, and if the user immediately selects another navigation object, then the script's navigation request MUST be cancelled.

Traceability:

[ESMP]	Section 7.3.1

Verification: Load the page. Activate the "Button". Verify that this starts navigation to another page. Immediately activate the link "Link". Verify that "Page 1" does not get loaded and "Page 2" gets loaded.

49.20.03

Requirement: Optional

SCR: ESMP-LANGUAGE-C-032

Test path: esmp_Sec7/7.3_Scr_Comp_Mech/abnormalTerm1.xhtml

Assertion: The user agent MAY report to the user, errors from the abnormal termination of scripts.

Traceability:

[ESMP]	Section 7.3.1

Verification: Load the page. There is an error in scripting in the page. Verify that user-agent indicates that there is an error with the script.

49.20.04

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-033

Test path: esmp_Sec7/7.3_Scr_Comp_Mech/abort_feature1.xhtml

Assertion 1: There MUST be a user input mechanism, such as a key or button, available to provide access to the abort functionality to the end user.

Assertion 2: Completion of the abort processing MUST guarantee that the browser is capable of continuing, or returning to a known state.

Traceability:

[ESMP]	Section 7.3.2

Verification: Load the page. Activate the “StartLoop” link. This starts an infinite loop of script execution. Verify that the script execution can be aborted by using the script abort feature of your browser. Then verify that activating “Display” link displays a “Success!” message.

49.20.05

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-033

Test path: esmp_Sec7/7.3_Scr_Comp_Mech/abort_event1.xhtml

Assertion 1: The script engine MUST provide an abort event to allow the initiation of abnormal script execution.

Assertion 2: Completion of the abort processing MUST guarantee that the browser is capable of continuing, or returning to a known state.

Traceability:

[ESMP]	Section 7.3.2

Verification: Load the page. Cancel the loading of image and verify that an alert message

"Loading aborted" is displayed. Then verify that activating “Display” link displays a “Success!” message.

9. EVENTS

9.1 XHTML Event Types

9.2 Event Binding

49.22.01

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-035

Evt_2

Test path: esmp_Sec8/event_binding/binding1.xhtml

Assertion: The user agent MUST operate as if the value of the event handler attribute is the body of an anonymous function that is registered as an event handler.

Traceability:

[ESMP]	Section 8.2

Assertion: The “onload” event on the “body” element must be supported.

Traceability:

[XHTMLMP11]	Section 10.2

Verification: Verify that an alert message "Test passed" appears upon loading of the page.

49.22.02

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-035

Evt_2

Test path: esmp_Sec8/event_binding/binding2.xhtml

Assertion: Only a single event handler is supported for a given event type on a given target element.

Traceability:

[ESMP]	Section 8.2

Verification: Verify that only one alert message appears upon loading of the page, which says "Test passed".

49.22.03

Requirement: Mandatory

SCR: ESMP-LANGUAGE-C-035

Test path: esmp_Sec8/event_binding/binding3.xhtml

Assertion: Any attempt to modify the value of an event handler attribute via the DOM interfaces MUST result in a deregistration of any existing event handler.

Traceability:

[ESMP]	Section 8.2

Verification: Verify that an alert message "Hello!" appears when the "Button-1" is activated. Verify that no message appears when "Button-2" is activated.

9.3 Event Object

49.23.01

Requirement: Optional

SCR: None

Test path: esmp_Sec8/objEvent/eventObject1.xhtml

Assertion: When the user agent handles an event, then it MUST make available to the event listener an Event object.

Traceability:

[ESMP]	Section 8.3

49.23.02

Requirement: Mandatory

SCR: None

Test path: esmp_Sec8/objEvent/keyCode1.xhtml

Assertion: When user presses and releases a keyboard key and triggers a **KeyPress** event, where the key has a Unicode equivalent mapping, then the keyCode property of the event MUST be set with that Unicode integer value of the character of the key.

Traceability:

[ESMP]	Section 8.3.2.1

Verification: Load the page. Activate the input field. Enter the character "A" (the capital A) in the input field.

49.23.03

Requirement: Mandatory

SCR: None

Test path: esmp_Sec8/objEvent/target1.xhtml

Assertion: When an event is generated, the **target** property of the event object MUST return a reference to the element object, which was the target of the event.

Traceability:

[ESMP]	Section 8.3.2.2

49.23.04

Requirement: Mandatory

SCR: None

Test path: esmp_Sec8/objEvent/timestamp1.xhtml

Assertion: When an event is generated, the **timestamp** property of the event object MUST return an integer number, which represents the time the event was generated.

Assertion: The **timestamp** property of the event object MUST return the time the event was created in a format compatible with **Date** object.

Traceability:

[ESMP]	Section 8.3.2.3

Verification: Load the page. Verify that current time stamp is displayed as an alert message.

49.23.05

Requirement: Mandatory

SCR: None

Test path: esmp_Sec8/objEvent/eventType1.xhtml

Assertion: When an event is generated due to event type “onload”, then the type property of the event object MUST be set to the exact string “load”.

Traceability:

[ESMP]	Section 8.3.2.4

Assertion 2: The script invocation MUST occur as a result of events.

Traceability:

[ESMP]	Section 7.2

Assertion 3: When an event supported by a user agent occurs, the user agent MUST create a read-only instance of an Event object.

Traceability:

[ESMP]	Section 8.4

Assertion 4: Script context MUST include the event state, which is the set of events (0 or more) and their associated attribute values, which are pending at the time the script is invoked.

Traceability:

[ESMP]	Section 7.1.1

Assertion 5: Script context MUST include the event state associated with the currently executing document.

Traceability:

[ESMP]	Section 7.1.2

Assertion 6: When an event supported by a user agent occurs, the user agent MUST create a read-only instance of an Event object and then MUST invoke the event handler associated with that event, passing the Event object instance to the handler.

Traceability:

[ESMP]	Section 8.4

Assertion 7: The XHTML Mobile Profile user agent MUST support XHTML Events as a prerequisite for script support.

Traceability:

[ESMP]	Section 8.1
[XHTMLMP11]	Section 10

10. BROWSER HOST OBJECTS

10.1 Global Object (Window Object)

49.25.01

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-036

Test path: esmp_Sec9/objWindow/global_enum.xhtml

Assertion: The global (window) object MUST be enumerable.

Traceability:

[ESMP]	Section 9.1, Sec 6.2.2

Verification: Verify that any implementation specific extensions to the Global object are listed out.

10.1.1 Properties

49.25.02

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_2, Host_2

Test path: esmp_Sec9/objWindow/history_prop1.xhtml

Assertion: The “history” property of the window object is the history object.

Traceability:

[ESMP]	Section 9.1.1.1

Assertion: The global (window) object may also be referred to as “parent” .

Traceability:

[ESMP]	Section 9.1

Verification:

Step 1: Verify that page titled "History Page 1" is loaded. Activate the hyperlink "link".

Step 2: Verify that page titled "History Page 2" is loaded. Activate the hyperlink "link".

Step 3: Verify that page titled "History Page 1" is loaded. If so, post success using post results link.

49.25.03

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_3, Host_2

Test path: esmp_Sec9/objWindow/navigator_prop.xhtml

Assertion: The “navigator” property of the window object is the navigator object.

Traceability:

[ESMP]	Section 9.1.1.1

Assertion: The global (window) object may also be referred to as “top”.

Traceability:

[ESMP]	Section 9.1

Verification: Verify that the user-agent identifier string of browser application is displayed. Verify that it contains user-agent name first and any optional qualifier name in parenthesis in the end.

49.25.04

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_1

Test path: esmp_Sec9/objWindow/location_prop.asp

Assertion: The “location” property of the window object is the location object.

Traceability:

[ESMP]	Section 9.1.1.3

49.25.05

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_4, Host_2

Test path: esmp_Sec9/objWindow/document_prop.xhtml

Assertion: The “document” property of the window object is the document object.

Traceability:

[ESMP]	Section 9.1.1.4

Assertion: The global (window) object may also be referred to as “self”.

Traceability:

[ESMP]	Section 9.1

49.25.06

Requirement: Mandatory

SCR:

Host_1_1

Test path: esmp_Sec9/objWindow/winLocn1a.xhtml

Assertion: When the **window.location** property is assigned with a string value representing an URL, then the browser MUST load the document pointed by the assigned URL.

Traceability:

[ESMP]	Section 9.4

10.1.2 Methods

10.1.2.1 prompt()

49.25.07

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_7

Test path: esmp_Sec9/objWindow/ prompt_nochange1.xhtml

Assertion: When the prompt() method is called, where a message M is supplied as a string argument, then the user-agent MUST display the given message M as a dialog with the default reply "undefined" and prompt for the user-input. The dialog MUST be a modal dialog.

Assertion: If the user accepts the message without changing the default reply, then “undefined” MUST be returned as return value.

Traceability:

[ESMP]	Section 9.1.2.1

49.25.08

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_7

Test path: esmp_Sec9/objWindow/ prompt_change1.xhtml

Assertion: When the prompt() method is called, where a message M is supplied as a string argument, then the user-agent MUST display the given message M as a dialog and prompt for the user-input. The dialog MUST be a modal dialog. If the user enters a reply R, then the string R MUST be returned as return value.

Traceability:

[ESMP]	Section 9.1.2.1

49.25.09

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_7

Test path: esmp_Sec9/objWindow/ prompt_nochange2.xhtml

Assertion: When the prompt() method is called, where a message M and default reply D are supplied as the two string arguments, then the user-agent MUST display the given message M as a dialog along with the string D as the default reply and prompt for the user-input. The dialog MUST be a modal dialog.

Assertion: If the user accepts the message without changing the default reply, then the default reply string D MUST be returned as the return value.

Traceability:

[ESMP]	Section 9.1.2.1

49.25.10

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_7

Test path: esmp_Sec9/objWindow/ prompt_change2.xhtml

Assertion: When the prompt() method is called, where a message M and default reply D are supplied as the two string arguments, then the user-agent MUST display the given message M as a dialog along with the string D as the default reply and prompt for the user-input. The dialog MUST be a modal dialog.

Assertion: If the user enters a reply R, then the string R MUST be returned as return value

Traceability:

[ESMP]	Section 9.1.2.1

10.1.2.2 confirm()

49.25.11

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_6

Test path: esmp_Sec9/objWindow/confirm_ok1.xhtml

Assertion: When the confirm() method is called, where a message M is supplied as a string argument, then the user-agent MUST display the given message M as a dialog and wait for the user to select positive or negative selection choice. The method MUST return Boolean true if the user selects the positive selection choice.

Traceability:

[ESMP]	Section 9.1.2.2

49.25.12

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_6

Test path: esmp_Sec9/objWindow/confirm_cancel1.xhtml

Assertion: When the confirm() method is called, where a message M is supplied as a string argument, then the user-agent MUST display the given message M as a dialog and wait for the user to select positive or negative selection choice. The method MUST return Boolean false if the user selects the negative selection choice.

Traceability:

[ESMP]	Section 9.1.2.2

Verification: Verify that a confirm dialog is displayed. Verify that "Yeah" is displayed as label for positive choice. Select either of the choices to close confirm dialog. Post results.

49.25.13

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_6

Test path: esmp_Sec9/objWindow/confirm_ok2.xhtml

Assertion: When the confirm() method is called, where a message M , an OKReply O are supplied as two string arguments, then the user-agent MUST display the given message M as a dialog along with the OKReply O as the label for positive selection choice.

Traceability:

[ESMP]	Section 9.1.2.2

49.25.14

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_6

Test path: esmp_Sec9/objWindow/confirm_cancel2.xhtml

Assertion: When the confirm() method is called, where a message M , an OKReply O and a CancelReply C are supplied as the three string arguments, then the user-agent MUST display the given message M as a dialog along with the OKReply O as the label for positive selection choice and the CancelReply C as the label for the negative selection choice.

Traceability:

[ESMP]	Section 9.1.2.2

Verification: Verify that a confirm dialog is displayed. Verify that "Yeah" is displayed as label for positive choice and "Nope" as label for negative choice. Select either of the choices to close confirm dialog. Post results.

49.25.15

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_6

Test path: esmp_Sec9/objWindow/ confirm_msg1.xhtml

Assertion: When the confirm() method is called, where no arguments are supplied, then the user-agent MUST display a dialog to the user without any message, but still with a positive and negative selection choices for the user to select.

Traceability:

[ESMP]	Section 9.1.2.2

Verification: Verify that a confirm dialog is displayed. Verify that two choices are available for selection. Select either of the choices to close confirm dialog. Post results.

10.1.2.3 alert()

49.25.16

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_5

Test path: esmp_Sec9/objWindow/alert_msg1.xhtml

Assertion: When the alert() method is called, where a message M is supplied as a string argument, then the user-agent MUST display the given message M as a dialog along and wait for the user confirmation. The dialog MUST be a modal dialog.

Assertion: When the alert() method is called, where a message M is supplied as a string argument, then the user-agent MUST display the given message M as a dialog along and wait for the user confirmation. When the user confirms the message, the method MUST not return any value.

Traceability:

[ESMP]	Section 9.1.2.3

Verification: Verify that an alert message "Hello!" is displayed. Acknowledge the alert message. Post results.

49.25.17

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Host_1_5

Test path: esmp_Sec9/objWindow/alert_no_msg1.xhtml

Assertion: When the alert() method is called without supplying any argument, then the user-agent MUST a dialog without any message and wait for the user confirmation. The dialog MUST be a modal dialog.

Traceability:

[ESMP]	Section 9.1.2.3

Verification: Verify that a blank alert message is displayed. Acknowledge the alert message. Post results.

10.1.2.4 setTimeout()

49.25.18

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Test path: esmp_Sec9/objWindow/setTimeout1.xhtml

Assertion: When the setTimeout() method is called, where a script function name F and a number N representing duration in milliseconds are passed as the two arguments, then the requested script function F MUST be called after a duration of N milliseconds.

Assertion: The execution of the current script MUST not stop during this waiting time of N milliseconds.

Assertion: The implementation MUST support a minimum of a single timeout.

Traceability:

[ESMP]	Section 9.1.2.4

Verification: Load the page. Wait for 10 seconds. Post failure, if results are not automatically posted.

49.25.19

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Test path: esmp_Sec9/objWindow/setTimeout2.xhtml

Assertion: When the script function name supplied as an argument for the setTimeout() method has parentheses at the end of its name, then the function call will be immediately made without waiting for the timer to end.

Traceability:

[ESMP]	Section 9.1.2.4

Verification: Load the page. Wait for 10 seconds. Post failure, if results are not automatically posted.

49.25.20

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Test path: esmp_Sec9/objWindow/setTimeout3.xhtml

Assertion: When the setTimeout() method is called, where a script function name F and a number N representing duration in milliseconds and Ax number of function arguments are supplied as arguments, then the requested script function F MUST be called after a duration of N milliseconds and the arguments must be passed to the function called.

Traceability:

[ESMP]	Section 9.1.2.4

Verification: Load the page. Wait for 10 seconds. Post failure, if results are not automatically posted.

49.25.21

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Test path: esmp_Sec9/objWindow/setTimeout4.xhtml

Assertion: When the setTimeout() method is called, where a script function name F and a number N representing duration in milliseconds and Ax number of function arguments are supplied as arguments, then the requested script function F MUST be called after a duration of N milliseconds. If the requested function F requires more than Ax number of parameters, then all the unspecified arguments MUST be set to “undefined”.

Traceability:

[ESMP]	Section 9.1.2.4

Verification: Load the page. Wait for 10 seconds. Post failure, if results are not automatically posted.

10.1.2.5 clearTimeout()

49.25.22

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-036

Test path: esmp_Sec9/objWindow/clearTimeout1.xhtml

Assertion: When the clearTimeout() method is called, where a number representing a timeoutID is supplied as argument, then any existing time out represented by the given ID MUST be unset and cleared.

Assertion: The return value of the clearTimeout() method MUST be none.

Assertion: When the setTimeout() method is called, then the method MUST return a number as an opaque handle representing the timer.

Traceability:

[ESMP]	Section 9.1.2.4

Verification: Load the page. Wait for 10 seconds. Verify that an alert message "SUCCESS" is displayed. If so, test passed. Post results.

10.2 Navigator Object

10.2.1 Version

49.26.01

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-014

Test path: esmp_Sec9/objNavigator/version.xhtml

Assertion: The **navigator** object MUST maintain an enumerable, read-only version property in the format "M.m.I.i".

Traceability:

[ESMP]	Section 9.2.1, Sec 6.2.1

Verification: Verify that the version number is displayed in the form "M.m.I.i".

49.26.02

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-037

Test path: esmp_Sec9/objNavigator/navigator_enum.xhtml

Assertion: The **navigator** object MUST be enumerable.

Traceability:

[ESMP]	Section 9.2.1, Sec 6.2.2

Verification: Verify that any implementation specific extensions to the Navigator object are listed out.

10.2.2 Properties

10.2.2.1 appName

49.26.03

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-037

Host_3_1

Test path: esmp_Sec9/objNavigator/appName.xhtml

Assertion: The **appName** property of the **navigator** object MUST be a string identifying the application manufacturer first, a string identifying the browser application next and any optional additional qualifiers inside parentheses in the end.

Traceability:

[ESMP]	Section 9.2.2.1

Verification: Verify that a descriptor of browser application and device is displayed. Verify that it contains the following three parts in the same order:

- browser application manufacturer
- browser application name
- any optional qualifier in parenthesis.

example: Nokia Mbrowser (test version)

49.26.04

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-037

Host_3_1

Test path: esmp_Sec9/objNavigator/appName_readonly.xhtml

Assertion: Any attempt to modify the read-only **appName** property of the **navigator** object, MUST be ignored.

Traceability:

[ESMP]	Section 9.2.2.1
[ECMA262]	Section 8.6.1

10.2.2.2 appVersion

49.26.05

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-037

Host_3_2

Test path: esmp_Sec9/objNavigator/appVersion.xhtml

Assertion: The **appVersion** property of the **navigator** object MUST be a string identifying the application version number first and any optional additional qualifiers inside parentheses next.

Traceability:

[ESMP]	Section 9.2.2.2

Verification: Verify that the version number of browser application and device is displayed. Verify that it contains the following two parts in the same order:

- Manufacturer's application version number
- any optional qualifier in parenthesis.

example: 4.0 (test version)

49.26.06

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-037

Host_3_2

Test path: esmp_Sec9/objNavigator/appVersion_readonly.xhtml

Assertion: Any attempt to modify the read-only **appVersion** property of the **navigator** object, MUST be ignored.

Traceability:

[ESMP]	Section 9.2.2.2
[ECMA262]	Section 8.6.1

10.2.2.3 userAgent

49.26.07

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-037

Host_3_3

Test path: esmp_Sec9/objNavigator/userAgent.xhtml

Assertion: The **userAgen**t property of the **navigator** object MUST have a string identifying the user agent name first and any optional additional qualifiers inside parentheses next.

Traceability:

[ESMP]	Section 9.2.2.3

Verification: Verify that a descriptor of the user-agent is displayed. Verify that it contains the following two parts in the same order:

- User-agent name
- any optional qualifier in parenthesis.

example: Titanix/4.0 (test version)

49.26.08

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-037

Host_3_3

Test path: esmp_Sec9/objNavigator/userAgent_readonly.xhtml

Assertion: Any attempt to modify the read-only **userAgent** property of the navigator object, MUST be ignored.

Traceability:

[ESMP]	Section 9.2.2.3
[ECMA262]	Section 8.6.1

10.3 History Object

10.3.1 Version

49.27.01

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-014

Test path: esmp_Sec9/objHistory/version.xhtml

Assertion: The **history** object MUST maintain an enumerable, read-only version property in the format "M.m.I.i".

Traceability:

[ESMP]	Section 9.3.1, Sec 6.2.1

Verification: Verify that the version number is displayed in the form "M.m.I.i".

49.27.02

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-038

Test path: esmp_Sec9/objHistory/history_enum.xhtml

Assertion: The **history** object MUST be enumerable.

Traceability:

[ESMP]	Section 9.3.1, Sec 6.2.2

Verification: Verify that any implementation specific extensions to the History object are listed out.

10.3.2 Properties

10.3.2.1 length

49.27.03

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_1

Test path: esmp_Sec9/objHistory/length1a.xhtml

Assertion: The **length** property of **history** object MUST be an integer number, which is equal to the number of history entries currently in the history list.

Traceability:

[ESMP]	Section 9.3.2.1

Verification:

Step 1: Note the value of L1. Click the "link" to load page titled "Length 1b".

Step 2: Note the value of L2. Verify that L2 = L1 + 1.

49.27.04

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_1

Test path: esmp_Sec9/objHistory/length_number.xhtml

Assertion: The **length** property of **history** object MUST be an integer number, which is equal to the number of history entries currently in the history list.

Traceability:

[ESMP]	Section 9.3.2.1

49.27.05

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_1

Test path: esmp_Sec9/objHistory/length_readonly.xhtml

Assertion: Any attempt to modify the read-only **length** property of the **history** object, MUST be ignored.

Traceability:

[ESMP]	Section 9.3.2.1
[ECMA262]	Section 8.6.1

10.3.3 Methods

10.3.3.1 back()

49.27.06

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_2

Test path: esmp_Sec9/objHistory/back1a.xhtml

Assertion 1: When the **back()** method of the **history** object is called and if a previous URL entry exists in the history list, then the browser MUST navigate to that previous entry.

Traceability:

[ESMP]	Section 9.3.3.1

Assertion 2: Script context MUST be inherited at the time of script invocation, which is an aggregation of the current browser context and the document context from which the script was called.

Traceability:

[ESMP]	Section 7.1.1

Assertion: Script context MUST include the history state associated with the currently executing document.

Traceability:

[ESMP]	Section 7.1.2

Verification:

Step 1: Verify that page titled "Back 1a" is loaded. Click the "link".

Step 2: Verify that page titled "Back 1b" is loaded. Click the "link". If the link does not work, test is failed.

Step 3: Verify that page titled "Back 1a" is loaded again. If so, test passed. Post the results.

49.27.07

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_2

Test path: esmp_Sec9/objHistory/back2a.xhtml

Assertion: When the **back()** method of the **history** object is called, and if there is no previous URL entry in the history list, then the method MUST not do anything.

Traceability:

[ESMP]	Section 9.3.3.1

Verification:

Step 1: Verify that page titled "Back 2a" is loaded.

Step 2: Clear the history list of your browser.

Step 3: Activate the hyperlink "link".

Step 4: Verify that nothing happens.

10.3.3.2 forward()

49.27.08

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_3

Test path: esmp_Sec9/objHistory/forward1a.xhtml

Assertion: When the **forward()** method of the **history** object is called and if a next URL entry exists in the history list, then the browser MUST navigate to that previous entry.

Traceability:

[ESMP]	Section 9.3.3.2

Verification:

Step 1: Verify that page titled "Forward 1a" is loaded.

Step 2: Activate the hyperlink "link 1".

Step 3: Verify that page titled "Forward 1b" is loaded.

Step 4: Use the Back button of the browser to navigate back

Step 5: Verify that page titled "Forward 1a" is loaded again.

Step 6: Activate the hyperlink "link 2".

Step 7: Verify that page titled "Forward 1b" is loaded again.

Step 8: Test passed. Post the results.

49.27.09

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_3

Test path: esmp_Sec9/objHistory/forward2a.xhtml

Assertion: When the **forward()** method of the **history** object is called, and if there is no next URL entry in the history list, then the method MUST not do anything.

Traceability:

[ESMP]	Section 9.3.3.2

Verification:

Step 1: Verify that page titled "Forward 2a" is loaded.

Step 2: Activate the hyperlink "link".

Step 3: Verify that nothing happens.

Step 4: Test passed. Post the results.

10.3.3.3 go()

49.27.10

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_4

Test path: esmp_Sec9/objHistory/go_n1.xhtml

Assertion: When the **go()** method of the **history** object is called, where a positive integer N is supplied as an argument, and if there is an Nth URL entry in the subsequent history list of the browser, then the browser MUST navigate to that URL.

Assertion: When the **go()** method of the **history** object is called, where a negative integer N is supplied as an argument, and if there is an Nth URL entry in the previous history list of the browser, then the browser MUST navigate to that URL.

Traceability:

[ESMP]	Section 9.3.3.3

Verification:

Step 1: Verify that page titled "Page-1" is loaded.

Step 2: Activate the hyperlink "Next" to load "Page-2".

Step 3: In "Page-2", activate the hyperlink "Next" to load "Page-3".

Step 4: In "Page-3", activate the hyperlink "Next" to load "Page-4".

Step 5: In "Page-4", activate the hyperlink "Next" to load "Page-5".

Step 6: In "Page-5", activate the hyperlink "Next" to load "Page-6".

Step 7: In "Page-6", activate the hyperlink "Next" to load "Page-7".

Step 8: In "Page-7", activate the hyperlink "Next" to load "Page-8".

Step 9: In "Page-8", activate the hyperlink "Go to Page-5".

Step 10: Verify that page titled "Page-5" is loaded.

Step 11: Activate the hyperlink "Go to Page-7".

Step 12: Verify that page titled "Page-7" is loaded.

Step 13: Activate the hyperlink "Go to Page-2".

Step 14: Verify that page titled "Page-2" is loaded.

Step 15: Activate the hyperlink "Go to Page-4".

Step 16: Verify that page titled "Page-4" is loaded.

Step 17: Activate the hyperlink "Go to Page-6".

Step 18: Verify that page titled "Page-6" is loaded.

Step 19: Activate the hyperlink "Go to Page-3".

Step 20: Verify that page titled "Page-3" is loaded.

Step 21: Activate the hyperlink "Go to Page-1".

Step 22: Verify that page titled "Page-1" is loaded.

Step 23: Test passed. Post the results.

49.27.11

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_4

Test path: esmp_Sec9/objHistory/ reload1.asp.xhtml

Assertion: When the **go()** method of the **history** object is called, where the number zero is supplied as an argument, then the browser MUST reload the current URL.

Traceability:

[ESMP]	Section 9.3.3.3

Verification:

Step 1: Verify that page titled "Reload 1" is loaded.

Step 2: Note the value of N and also timestamp.

Step 3: Activate the hyperlink "link" .

Step 4: Verify that same page gets reloaded.

Step 5: Verify that N remains same and timestamp changes.

Step 6: Test passed. Post results.

49.27.12

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_4

Test path: esmp_Sec9/objHistory/goplus1a.xhtml

Assertion: When the **go()** method of the **history** object is called, where a positive integer N is supplied as an argument, and if there is an Nth URL entry in the subsequent history list of the browser, then the browser MUST navigate to that URL.

Traceability:

[ESMP]	Section 9.3.3.3

49.27.13

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-038

Host_5_4

Test path: esmp_Sec9/objHistory/gominus1a.xhtml

Assertion: When the **go()** method of the **history** object is called, where a negative integer N is supplied as an argument, and if there is an Nth URL entry in the previous history list of the browser, then the browser MUST navigate to that URL

Traceability:

[ESMP]	Section 9.3.3.3

10.4 Location Object

10.4.1 Version

49.28.01

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-014

Test path: esmp_Sec9/objLocation/version.xhtml

Assertion: The **location** object MUST maintain an enumerable, read-only version property in the format "M.m.I.i".

Traceability:

[ESMP]	Section 9.4.1, Sec 6.2.1

Verification: Verify that the version number is displayed in the form "M.m.I.i".

49.28.02

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/location_enum.xhtml

Assertion: The **location** object MUST be enumerable.

Traceability:

[ESMP]	Section 9.4.1, Sec 6.2.2

Verification: Verify that any implementation specific extensions to the Location object are listed out.

10.4.2 Properties

10.4.2.1 hash

49.28.03

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/hash_get1a.asp

Assertion: The **hash** property of the **location** object MUST be a string, which is fragment (anchor) part of the current URL including the hash symbol.

Traceability:

[ESMP]	Section 9.4.2.1

49.28.04

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/hash_none.xhtml

Assertion: The **hash** property of the **location** object MUST be an empty string if there is no fragment (anchor) part for the current URL.

Traceability:

[ESMP]	Section 9.4.2.1

49.28.05

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/hash_set1.xhtml

Assertion: When the **hash** property of the **location** object is assigned with a string representing a named anchor (which is available in the current document), irrespective of the presence of hash symbol in the assigned value, navigation to that named anchor MUST take place.

Traceability:

[ESMP]	Section 9.4.2.1

49.28.06

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/hash_set2.xhtml

Assertion: When the **hash** property of the **location** object is assigned with a string representing a named anchor (which is available in the current document), irrespective of the presence of hash symbol in the assigned value, navigation to that named anchor MUST take place.

Traceability:

[ESMP]	Section 9.4.2.1

49.28.07

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/hash_invalid1a.xhtml

Assertion: When the **hash** property of the **location** object is assigned with an unavailable anchor name, then an HTTP error may be generated.

Traceability:

[ESMP]	Section 9.4.2.1

49.28.08

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/hash_illegalURI.xhtml

Assertion: When the **hash** property of the **location** object is assigned with an illegal anchor name, then the compiler MUST throw URIError.

Traceability:

[ESMP]	Section 9.4.2.1

10.4.2.2 host

49.28.09

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/host_prop1a.asp

Assertion: The **host** property of the **location** object MUST be a string, which is the concatenation of the hostname and port of the current URL, separated by a colon.

Assertion: When the **host** property of the **location** object is assigned with a value representing a host name with/without port number, then the browser may navigate to the new URL, as set by the assignment.

Traceability:

[ESMP]	Section 9.4.2.2

49.28.10

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/host_prop2a.asp

Assertion: When the port number of current URL is null, then the **host** property of the **location** object MUST be a string, which is the hostname of the current URL.

Traceability:

[ESMP]	Section 9.4.2.2

10.4.2.3 href

49.28.11

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_1

Test path: esmp_Sec9/objLocation/href_prop.asp

Assertion 1: The **href** property of the **location** object MUST be a string specifying the entire current URL.

Traceability:

[ESMP]	Section 9.4.2.3

Assertion 2: Script context MUST be inherited at the time of script invocation, which is an aggregation of the current browser context and the document context from which the script was called.

Traceability:

[ESMP]	Section 7.1.1

Assertion: Script context MUST include the location state associated with the currently executing document.

Traceability:

[ESMP]	Section 7.1.2

49.28.12

Requirement: Optional.

SCR: ESMP-HOSTOBJECT-C-039

Host_4_1

Test path: esmp_Sec9/objLocation/href_set1a.asp

Assertion: When the **href** property of the **location** object is assigned with a value representing an URL, then the browser may navigate to this new URL, as set by the assignment.

Traceability:

[ESMP]	Section 9.4.2.3

10.4.2.4 hostname

49.28.13

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_2

Test path: esmp_Sec9/objLocation/hostname_prop.asp

Assertion: The **hostname** property of the **location** object MUST be a string specifying the hostname part of current URL.

Traceability:

[ESMP]	Section 9.4.2.4

49.28.14

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-039

Host_4_2

Test path: esmp_Sec9/objLocation/hostname_set.xhtml

Assertion: When the **hostname** property of the **location** object is assigned with a hostname value, then the browser may navigate to this new resulting URL.

Traceability:

[ESMP]	Section 9.4.2.4

10.4.2.5 pathname

49.28.15

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_4

Test path: esmp_Sec9/objLocation/pathname_prop.asp

Assertion: The **pathname** property of the **location** object MUST be a string specifying the pathname component of current URL.

Traceability:

[ESMP]	Section 9.4.2.5

49.28.16

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-039

Host_4_4

Test path: esmp_Sec9/objLocation/pathname_set1a.xhtml

Assertion: When the **pathname** property of the **location** object is assigned with a pathname value, then the browser may navigate to this new resulting URL.

Traceability:

[ESMP]	Section 9.4.2.5

10.4.2.6 port

49.28.17

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Test path: esmp_Sec9/objLocation/port_prop1a.asp

Assertion: The **port** property of the **location** object MUST be a string whose value is the port number component of current URL. This value MUST not include the colon that separates the hostname from the port number.

Traceability:

[ESMP]	Section 9.4.2.6

10.4.2.7 protocol

49.28.18

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_5

Test path: esmp_Sec9/objLocation/protocol_prop1a.asp

Assertion: The **protocol** property of the **location** object MUST be a string whose value is the protocol part of the current URL including the colon at the end.

Traceability:

[ESMP]	Section 9.4.2.7

49.28.19

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_5

Test path: esmp_Sec9/objLocation/protocol_readonly.xhtml

Assertion: Any attempt to write to the read-only **protocol** property of the **location** object MUST be ignored.

Traceability:

[ESMP]	Section 9.4.2.7
[ECMA262]	Section 8.6.1

10.4.2.8 search

49.28.20

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_3

Test path: esmp_Sec9/objLocation/search_prop.asp?a=1&b=2

Assertion: The **search** property of the **location** object MUST be a string specifying the substring of the current URL that comes after the first question mark URI delimiter. This value MUST include the question mark also.

Traceability:

[ESMP]	Section 9.4.2.8

49.28.21

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-039

Host_4_3

Test path: esmp_Sec9/objLocation/search_set.asp?a=1&b=2

Assertion: When the **search** property of the **location** object is assigned with a new search string value, then the browser may navigate to this new resulting URL.

Traceability:

[ESMP]	Section 9.4.2.8

10.4.3 Methods

10.4.3.1 assign()

49.28.22

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_6

Test path: esmp_Sec9/objLocation/assign1a.xhtml

Assertion: When the **assign()** method of the **location** object is called, where a string representing an URL is supplied as an argument, then the browser MUST load the document pointed by the supplied URL. The history stack MUST be pushed with this URL.

Traceability:

[ESMP]	Section 9.4.3.1

Verification:

Step 1: Verify that page titled "Assign 1A" is loaded.

Step 2: Activate the hyperlink "Next" to load page titled "Assign 1B".

Step 3: Activate the hyperlink "link".

Step 4: Verify that page titled "Assign 1C" is loaded.

Step 5: Navigate back using browser's back functionality.

Step 6: Verify that page titled "Assign 1B" is loaded again.

Step 7: Test passed. Post results.

49.28.23

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-039

Host_4_6

Test path: esmp_Sec9/objLocation/assign_exit1a.xhtml

Assertion: When the **assign()** method of the **location** object is called, where a string representing an URL is supplied as an argument, then the actual loading of the document pointed by this URL, is not guaranteed till the script exit.

Traceability:

[ESMP]	Section 9.4.3.1

Verification:

Step 1: Verify that page titled "Assign exit 1A" is loaded.

Step 2: Verify that an alert message "Hello!" is displayed.

Step 3: Accept the alert message. Page titled "Assign exit 1B" must be loaded automatically.

Step 4: Post success using post results link.

49.28.24

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-039

Host_4_6

Test path: esmp_Sec9/objLocation/assign_error1a.xhtml

Assertion: When the **assign()** method of the **location** object is called, where a non-existent URL is passed as argument, then standard HTTP error must be generated.

Traceability:

[ESMP]	Section 9.4.3.1

Verification:

Step 1: Verify that page titled "Page 1A" is loaded.

Step 2: Click the link "Next".

Step 3: Verify that an HTTP error is displayed.

Step 4: Post results from page "Page 1A"

49.28.25

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_6

Test path: esmp_Sec9/objLocation/locnDef1a.xhtml

Assertion: When the **location** object is assigned to a value representing an URL, then the browser MUST load the document pointed by the assigned URL.

Traceability:

[ESMP]	Section 9.4

10.4.3.2 reload()

49.28.26

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_8

Test path: esmp_Sec9/objLocation/reload_formData1.asp

Assertion: When the **reload()** method of the **location** object is called, where a Boolean true is supplied as an argument, then the browser MUST reload the current document from the origin server, bypassing the client cache. All form variable data MUST be re-initialized when the document is reloaded.

Traceability:

[ESMP]	Section 9.4.3.2

Verification:

Step 1: Verify that page titled "Reload formData 1" is loaded.

Step 2: Note the timestamp and the value displayed in input field. Note that checkbox is checked by default.

Step 3: Modify the values of input field and de-select the checkbox.

Step 4: Activate the link "link" to reload the page.

Step 5: Verify that page is reloaded.

Step 6: Verify that timestamp is changed now.

Step 7: Verify that input field and check box have default values again.

Step 8: Test passed. Post results.

49.28.27

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_8

Test path: esmp_Sec9/objLocation/reload2.asp

Assertion: When the **reload()** method of the **location** is called, where a Boolean false is supplied as an argument, and if the current document is available in the client cache, then the browser MUST reload the current document from the client cache.

Traceability:

[ESMP]	Section 9.4.3.2

Verification:

Step 1: Make sure the browser cache is enabled before loading the test.

Step 2: Verify that page titled "Reload 2" is loaded

Step 3: Note the timestamp.

Step 4: Activate the link "link" to reload the page, within 5 minutes of first loading.

Step 5: Verify that page is reloaded.

Step 6: Verify that timestamp remained same.

Step 7: Post results.

49.28.28

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_8

Test path: esmp_Sec9/objLocation/reload3.asp

Assertion: When the **reload()** method of the **location** object is called without passing any arguments, and if the current document is available in the client cache, then the browser MUST reload the current document from the client cache.

Traceability:

[ESMP]	Section 9.4.3.2

Verification:

Step 1: Make sure the browser cache is enabled before loading the test.

Step 2: Verify that page titled "Reload 3" is loaded

Step 3: Note the timestamp.

Step 4: Activate the link "link" to reload the page, within 5 minutes of first loading.

Step 5: Verify that page is reloaded.

Step 6: Verify that timestamp remained same.

Step 7: Post results.

10.4.3.3 replace()

49.28.29

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-039

Host_4_7

Test path: esmp_Sec9/objLocation/replace1a.xhtml

Assertion: When the **replace()** method of the **location** object is called, where a string representing an URL is supplied as an argument, then the browser MUST load the document pointed by the supplied URL. The current entry in the history stack MUST be replaced with this new URL.

Traceability:

[ESMP]	Section 9.4.3.3

Verification:

Step 1: Verify that page titled "Replace 1A" is loaded.

Step 2: Activate the hyperlink "link" to load page titled " Replace 1B".

Step 3: Activate the hyperlink "link".

Step 4: Verify that page titled " Replace 1C" is loaded.

Step 5: Navigate back using browser's back functionality.

Step 6: Verify that page titled " Replace 1A" is loaded and not the page "Replace 1B".

Step 7: Test passed. Post results.

10.5 Basic Document Object

10.5.1 Version

49.29.01

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-014

Test path: esmp_Sec9/objDocument/version.xhtml

Assertion: The **document** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 9.5.1, Sec 6.2.1

Verification: Verify that the version number is displayed in the form “M.m.I.i”.

49.29.02

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-040

Test path: esmp_Sec9/objDocument/doc_enum.xhtml

Assertion: The **document** object MUST be enumerable.

Traceability:

[ESMP]	Section 9.5.1, Sec 6.2.2

Verification: Verify that any implementation specific extensions to the Document object are listed out.

10.5.2 Properties

10.5.2.1 cookie

49.29.03

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_1

Test path: esmp_Sec9/objDocument/readCookie1.asp

Assertion: The **cookie** property of the document object MUST provide access to the cookies associated with the current document domain.

Traceability:

[ESMP]	Section 9.5.2.1

49.29.04

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_1

Test path: esmp_Sec9/objDocument/readCookie2.asp

Assertion: The **document.cookie** MUST return a semi-colon-separated list of the cookies for the domain of the current document from the client cookie store.

Traceability:

[ESMP]	Section 9.5.2.1

Verification: Verify that a string of semi-colon separated name, value pairs is displayed.

49.29.05

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_1

Test path: esmp_Sec9/objDocument/setCookie1.xhtml

Assertion: When the **document.cookie** property is assigned with a string S representing a single name value pair, and if the string S matches the cookie production rules from [RFC2109], then a cookie MUST be created in client cookie store with the given name value pair.

Assertion: When the **document.cookie** property is assigned with a string S representing a single name value pair with acceptable attribute value pairs, and if the string S matches the cookie production rules from [RFC2109], then a cookie MUST be created in client cookie store with the given name value pair and the attribute value pairs.

Traceability:

[ESMP]	Section 9.5.2.1

49.29.06

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_1

Test path: esmp_Sec9/objDocument/setCookie2.xhtml

Assertion: When the **document.cookie** property is assigned with a string S representing a single name value pair, and if the string S does not match the cookie production rules from [RFC2109], then cookie MUST not be created.

Traceability:

[ESMP]	Section 9.5.2.1

10.5.2.2 domain

49.29.07

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Test path: esmp_Sec9/objDocument/domain_prop.asp

Assertion: The **domain** property of the **document** object MUST be a string containing the current document domain.

Traceability:

[ESMP]	Section 9.5.2.2

49.29.08

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Test path: esmp_Sec9/objDocument/domain_invalidChange.xhtml

Assertion: Attempting to change the origin domain by assigning new value to the **domain** property of the **document** object MUST not be allowed and an exception MUST be raised in such situations.

Traceability:

[ESMP]	Section 9.5.2.2

10.5.2.3 referrer

49.29.09

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_4

Test path: esmp_Sec9/objDocument/referrer_prop.xhtml

Assertion: The **referrer** property of the **document** object MUST be the URL of the document that linked to the current document.

Traceability:

[ESMP]	Section 9.5.2.3

49.29.10

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_4

Test path: esmp_Sec9/objDocument/referrer_empty1a.asp

Assertion: When the user reaches the current document, not by activating a link in some other document, then the **referrer** property of the **document** object MUST be set to empty string.

Traceability:

[ESMP]	Section 9.5.2.3

49.29.11

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_4

Test path: esmp_Sec9/objDocument/referrer_readonly.xhtml

Assertion: Any attempt by the script to modify the value of the **referrer** property MUST be ignored by the implementation.

Traceability:

[ESMP]	Section 9.5.2.3

10.5.2.4 title

49.29.12

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_5

Test path: esmp_Sec9/objDocument/title_prop.xhtml

Assertion: The **title** property of the **document** object MUST be the text in the <title> element of the document, if present.

Traceability:

[ESMP]	Section 9.5.2.4

49.29.13

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_5

Test path: esmp_Sec9/objDocument/title_empty.xhtml

Assertion: The **title** property of the **document** object MUST be set to empty string, when there is no <title> element present in the document.

Traceability:

[ESMP]	Section 9.5.2.4

49.29.14

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_5

Test path: esmp_Sec9/objDocument/title_readonly.xhtml

Assertion: Any attempt by the script to modify the read-only property **title** of the **document** object MUST be ignored.

Traceability:

[ESMP]	Section 9.5.2.4

10.5.3 Methods

10.5.3.1 clear()

49.29.15

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Test path: esmp_Sec9/objDocument/clear1a.xhtml

Assertion: When the **clear()** method of the **document** object is called, the current document MUST be cleared from the browser window.

Assertion: When current document is cleared from the browser window using the **clear()** method of **document** object, the history list MUST not be affected.

Assertion: When current document is cleared from the browser window using the **clear()** method of **documen**t object, the cached version of the current document MUST not be affected.

Traceability:

[ESMP]	Section 9.5.3.1

Verification:

Step 1: Load the page. Verify that page titled "Clear 1a" is loaded.

Step 2: Activate the link "Next".

Step 3: Verify that page titled "Clear 1b" is loaded.

Step 4: Activate the link "Clear Me".

Step 5: Verify that document is cleared from the display. If so, test passed.

Step 6: Use the back button to go back to page titled "Clear 1a" .

Step 7: Post results.

10.5.3.2 open()

49.29.16

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Test path: esmp_Sec9/objDocument/open1a.xhtml

Assertion: When the **open()** method of the **document** object is called, a path to the current document MUST be opened for writing.

Assertion: When the **open()** method of the **documen**t object is called, a path to the current document MUST be opened for writing and the current document DOM tree MUST be cleared.

Assertion: When the **close()** method of the **document** object is called, the write path to the current document MUST be closed and all the output MUST be flushed to the document.

Traceability:

[ESMP]	Section 9.5.3.2

Verification: Load the page. Verify that page titled "Open 1B" is loaded. Verify that the text "Test passed" is displayed .

49.29.17

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Test path: esmp_Sec9/objDocument/open2a.xhtml

Assertion: When the **open()** method of the **document** object is called, a path to the current document MUST be opened for writing.

Assertion: When the **open()** method of the **documen**t object is called, a path to the current document MUST be opened for writing and the current document DOM tree MUST be cleared.

Assertion: When the **close()** method of the **document** object is called, the write path to the current document MUST be closed and all the output MUST be flushed to the document.

Traceability:

[ESMP]	Section 9.5.3.2

Verification: Load the page. Verify that page titled "Open 2C" is loaded. Verify that the text "Test passed" is displayed .

49.29.18

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Test path: esmp_Sec9/objDocument/open3a.xhtml

Assertion: When the **open()** method of the **document** object is called, where an optional mime type M is supplied as argument, then the supplied mime type M MUST have no effect on the script compiler.

Traceability:

[ESMP]	Section 9.5.3.2

Verification: Load the page. Verify that page titled "Open 3B" is loaded. Verify that the text "Test passed" is displayed .

10.5.3.3 write()

49.29.19

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_7

Test path: esmp_Sec9/objDocument/write1.xhtml

Assertion: When the **write()** method of the **document** object is called, where a string S is passed as an argument, then the string S MUST be written to the current document.

Traceability:

[ESMP]	Section 9.5.3.1

Verification: Verify that the number "12345" is displayed five times, each in a different line.

49.29.20

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_7

Test path: esmp_Sec9/objDocument/write2.xhtml

Assertion: When the **write()** method of the **document** object is called, where a string S is passed as an argument, and if the document is already closed, then the previously written content MUST be discarded and the string S MUST be written to the current document.

Traceability:

[ESMP]	Section 9.5.3.1

Verification: Verify that the number "12345" is displayed five times, each in a different line.

49.29.21

Requirement: Optional

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_7

Test path: esmp_Sec9/objDocument/write3.xhtml

Assertion: When the **document.write()** statements are used in intrinsic event handler, then when the event is triggered, the **document.write()** MUST create and write to a reset document rather than modifying the current document.

Traceability:

[ESMP]	Section 9.5.3.1

Verification: Verify that the text "Test passed" is displayed.

49.29.22

Requirement: Mandatory

SCR: ESMP-HOSTOBJECT-C-040

Dom_1_8

Test path: esmp_Sec9/objDocument/writeln1.xhtml

Assertion: When the **writeln()** method of the document object is called, where a string S is passed as an argument, then the string S MUST be written to the current document and a newline MUST be appended to the end of the written string.

Traceability:

[ESMP]	Section 9.5.3.1

Verification: Verify that the number "12345" is displayed five times, each in a different line.

11. BROWSER XHTML DOM OBJECTS

11.1 XHTML Document Object

49.30.01

Requirement: Mandatory

Dom_1_6, Dom2_1, Dom_13

Test path: esmp_Sec10/formObject/forms_collection1.xhtml

Assertion: The **document.forms** returns an array containing all the form objects in the document.

Assertion: The **document.forms.length** returns the number of forms present in the document.

Assertion: The **n**th form object appearing in the document can be referred using the notation **document.forms[n-1]**

Traceability:

[S60KM2.5REQ]	Req ID: Dom_1_6, Dom_13

Assertion: The **name** property of a form object MUST return the value of the **name** attribute of that form.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_2_1

11.2 Link Element Object

To be included in future versions of the document.

11.3 Image Element Object

To be included in future versions of the document.

11.4 Form Object

49.33.01

Requirement: Mandatory

Dom_2_2, Dom_12

Test path: esmp_Sec10/formObject/forms_collection2.xhtml

Assertion: The **document.<formName>** MUST return the form object from the document, whose name attribute is set to value <**formName>.**

Traceability:

[S60KM2.5REQ]	Req ID: Dom_12

Assertion: The **action** property of a form object MUST return the value of the **action** attribute of that form.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_2_2

49.33.02

Requirement: Mandatory

Dom_2_1, Dom_2_3

Test path: esmp_Sec10/formObject/form_name1.xhtml

Assertion: The **name** property of a form object can be set with a value by assigning the value to the property using script.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_2_1

Assertion: The **elements** property of a form object MUST return an array of the control elements within that form.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_2_3

49.33.03

Requirement: Mandatory

Dom_2_2, Dom_2_4

Test path: esmp_Sec10/formObject/form_action1.xhtml

Assertion: The **action** property of a form object can be set with a value by assigning the value to the property using script.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_2_2

Assertion: When the **submit()** method of a form object is called, then the form data MUST be submitted.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_2_4

49.33.04

Requirement: Mandatory

??

Test path: esmp_Sec10/formObject/form_length1.xhtml

Assertion: The **length** property of a form object MUST return an integer which is equal to the number of control elements within that form.

Assertion: The **elements** property of a form object MUST return an array of the control elements within that form.

Traceability:

[S60KM2.5REQ]	Req ID: ??

49.33.05

Requirement: Mandatory

Evt_3

Test path: esmp_Sec10/formObject/form_onsubmit_true.xhtml

Assertion: When a **onSubmit** event is defined on a form with an event-handler, then activating the **submit** button of that form, MUST call that event-handler.

Assertion: When the event handler activated by **onSubmit** event of a form returns boolean true, then the form data MUST be submitted.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_3

Verification: Activate the "Submit" button. This must post the SUCCESS automatically. If nothing happens after activating "Submit" button, then test is failed. Post failure using the "Post Results" link.

49.33.07

Requirement: Mandatory

Dom_2_4

Test path: esmp_Sec10/formObject/form_submit.xhtml

Assertion: When the **submit()** method of a form object is called, then the form data MUST be submitted without activating any **onSubmit** event.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_2_4

49.33.08

Requirement: Mandatory

Evt_4

Test path: esmp_Sec10/formObject/form_onReset.xhtml

Assertion: When a **onReset** event is defined on a form with an event-handler E, then activating the **reset** button of that form, MUST call the event-handler E.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_4

Verification: Activate the "Reset" button. Verify that an alert message "SUCCESS" is displayed.

49.33.09

Requirement: Mandatory

Dom_2_5

Test path: esmp_Sec10/formObject/form_reset.xhtml

Assertion: When the **reset()** method of a form object is called, then all the fields of that form MUST be reset to the initial values.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_2_5

Verification: Note the initial values of the form fields. Modify the input values. Activate the "Reset" button. Verify that the values are reset to the initial values.

49.33.10

Requirement: Mandatory

Test path: esmp_Sec10/formObject/form_this.xhtml

Assertion: When an event is triggered, as a result of event defined on a form, the current form can be passed to the event handling function, by passing **this** as a parameter from the calling statement.

Traceability:

[S60KM2.5REQ]	Req ID:

Verification: Activate the submit button. Verify that an alert message "test passed" appears.

49.33.11

Requirement: Mandatory

Evt_1, Evt_6, DOM_3_5, DOM_7_6

Test path: esmp_Sec10/formObject/form_field_this.xhtml

Assertion: When an event is triggered, as a result of event defined on a child element of a form, the current form can be passed to the event handling function, by passing **this.form** as a parameter from the calling statement.

Assertion: When a **onFocus** event is defined on a text or select element with an event-handler E, then focussing onto that element, MUST call the event-handler E.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_6, DOM_3_5, DOM_7_6

Assertion: When a **onClick** event is defined on a button object (input types submit, reset, checkbox or radio) with an event-handler E, then focussing onto that element, MUST call the event-handler E.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_1

Verification: Move the focus onto each of the text input field, text area field and select list. Activate each of the radio buttons, check box and the submit button. Verify that an alert message "SUCCESS" appears every time.

49.33.12

Requirement: Mandatory

Dom_10

Test path: esmp_Sec10/formObject/form_fieldby_name.xhtml

Assertion: Any child element of a form object **formObj** can be referred as **formObj.elemName**, where **elemName** is the value of the **name** attribute of that child element.

Traceability:

[S60KM2.5REQ] Req ID: Dom_10

49.33.13

Requirement: Mandatory

Dom_11

Test path: esmp_Sec10/formObject/form_fieldby_array.xhtml

Assertion: The **elements** property of a form object MUST return an array of the control elements within that form.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_11

11.5 Input:Text/password/hidden & textarea Objects

49.34.01

Requirement: Mandatory

Dom_3_1

Test path: esmp_Sec10/inputObject/textInput_name1.xhtml

Assertion: The **name** property of a text input object is a string representing the name of input element. This assertion applies to input elements of types " text, password, hidden and text-area ".

Traceability:

[S60KM2.5REQ]	Req ID: Dom_3_1

49.34.02

Requirement: Mandatory

Dom_3_2

Test path: esmp_Sec10/inputObject/textInput_type1.xhtml

Assertion: The **type** property of a text input object is a string representing the type of input element. This assertion applies to input elements of types " text, password, hidden and text-area ".

Traceability:

[S60KM2.5REQ]	Req ID: Dom_3_2

49.34.03

Requirement: Mandatory

Dom_3_3

Test path: esmp_Sec10/inputObject/text_value1.xhtml

Assertion: The **value** property of an input object returns the string currently displayed in the field. This assertion applies to input elements of types " text, password and text-area "

Traceability:

[S60KM2.5REQ]	Req ID: Dom_3_3

Verification: Modify the values in the three input fields. Activate each of the links “User name”, “Password” and “Comments”. Verify that each time, the content in the corresponding input field is displayed as an alert message.

49.34.04

Requirement: Mandatory

Dom_3_3

Test path: esmp_Sec10/inputObject/text_value2.xhtml

Assertion: The displayed text in an input field can be modified through script, by assigning a value to the **value** property of that input object. This assertion applies to input elements of types " text, password and text-area ".

Traceability:

[S60KM2.5REQ]	Req ID: Dom_3_3

49.34.05

Requirement: Mandatory

Dom_3_4

Test path: esmp_Sec10/inputObject/text_dvalue1.xhtml

Assertion: The **defaultValue** property of an input object returns the string value set in the VALUE attribute on that **input** element in the mark-up for any of the input elements of types "text, password, hidden and text-area".

Traceability:

[S60KM2.5REQ]	Req ID: Dom_3_4

49.34.06

Requirement: Mandatory

Dom_3_4

Test path: esmp_Sec10/inputObject/text_dvalue2.xhtml

Assertion: The **defaultValue** property of an input object returns the string value set in the VALUE attribute on that **input** element in the mark-up for any of the input elements of types "text, password, hidden and text-area".

Traceability:

[S60KM2.5REQ] Req ID: Dom_3_4

49.34.07

Requirement: Mandatory

Dom_3_4

Test path: esmp_Sec10/inputObject/text_dvalue3.xhtml

Assertion: The **defaultValue** property of an input object can not be modified through script. This assertion applies to input elements of types " text, password, hidden and text-area ".

Traceability:

[S60KM2.5REQ] Req ID: Dom_3_4

49.34.08

Requirement: Mandatory

Dom_3_5

Test path: esmp_Sec10/inputObject/text_focus1.xhtml

Assertion: When the **focus()** method of an input object is called, then that input field in the display MUST receive focus. This assertion applies to input elements of types " text, password and text-area ".

Traceability:

[S60KM2.5REQ] Req ID: Dom_3_5

Verification: Verify that the "Comments" field is under focus, when the page is loaded.

49.34.09

Requirement: Mandatory

Evt_5

Test path: esmp_Sec10/inputObject/input_onchange1.xhtml

Assertion: When a **onChange** event is defined on a text input element with an event-handler E, then changing the content of that field, MUST call the event-handler E.This assertion applies to input elements of types " text, password and text-area ".

Traceability:

[S60KM2.5REQ]	Req ID: Evt_5

Verification: Activate each of the text input, password and the textarea fileds. Change the content of the field and end the input. Verify that each time, the new content of the field is displayed as alert message.

11.6 Radio Input Object

49.35.01

Requirement: Mandatory

Dom_6_1

Test path: esmp_Sec10/radioButtonObject/radioButton_name1.xhtml

Assertion: The **name** property of input object of "radio" type is a string representing the name of that input element.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_6_1

49.35.02

Requirement: Mandatory

Dom_6_2

Test path: esmp_Sec10/radioButtonObject/radioButton_type1.xhtml

Assertion: The **type** property of input object of "radio" type is the string "radio".

Traceability:

[S60KM2.5REQ]	Req ID: Dom_6_2

49.35.03

Requirement: Mandatory

Dom_6_3

Test path: esmp_Sec10/radioButtonObject/radioButton_value1.xhtml

Assertion: The **value** property of input object of "radio" type is the string representing the **value** attribute of that input element.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_6_3

49.35.04

Requirement: Mandatory

Dom_6_4

Test path: esmp_Sec10/radioButtonObject/radio_checked1.xhtml

Assertion: When the **checked** property of an input object is assigned boolean **true** through script, then that element MUST be displayed in a checked state.

Assertion: When the **checked** property of an input object is assigned boolean **false** through script, then that element MUST be displayed in un-checked state.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_6_4

Verification: Verify that the the two radio button R1 and R2 are displayed in unchecked state and the check box R3 in a pre-checked state.

49.35.05

Requirement: Mandatory

Dom_6_4

Test path: esmp_Sec10/radioButtonObject/radio_checked2.xhtml

Assertion: The **checked** property of an input object returns boolean **true**, if that element is currently in checked state.

Assertion: The **checked** property of an input object returns boolean **false**, if that element is currently in a unchecked state.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_6_4

49.35.06

Requirement: Mandatory

Evt_1, Dom_6_5

Test path: esmp_Sec10/radioButtonObject/radio_click1.xhtml

Assertion: When the **click()** method of an input object is called, then the user action of clicking on that element MUST be replicated.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_1, Dom_6_5

Verification: Click each of the links “radio 1” to “radio 4”. Verify that everytime, only the radio button next to the link gets checked.

11.7 Checkbox Input Object

49.36.01

Requirement: Mandatory

Dom_5_1

Test path: esmp_Sec10/checkBoxObject/checkBox_name1.xhtml

Assertion: The **name** property of input object of checkbox type is a string representing the name of that input element.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_5_1

49.36.02

Requirement: Mandatory

Dom_5_2

Test path: esmp_Sec10/checkBoxObject/checkBox_type1.xhtml

Assertion: The **type** property of input object of checkbox type is the string “checkbox”.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_5_2

49.36.03

Requirement: Mandatory

Dom_5_3

Test path: esmp_Sec10/checkBoxObject/checkBox_value1.xhtml

Assertion: The **value** property of input object of checkbox type is the string representing the **value** attribute of that input element.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_5_3

49.36.04

Requirement: Mandatory

Dom_5_4

Test path: esmp_Sec10/checkBoxObject/input_checked1.xhtml

Assertion: When the **checked** property of an input object is assigned boolean **true** through script, then that element MUST be displayed in a checked state.

Assertion: When the **checked** property of an input object is assigned boolean **false** through script, then that element MUST be displayed in un-checked state.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_5_4

Verification: Verify that the the two check boxes C1 and C2 are displayed in unchecked state and the check box C2 in a pre-checked state.

49.36.05

Requirement: Mandatory

Dom_5_4

Test path: esmp_Sec10/checkBoxObject/input_checked2.xhtml

Assertion: The **checked** property of an input object returns boolean **true**, if that element is currently in checked state.

Assertion: The **checked** property of an input object returns boolean **false**, if that element is currently in a unchecked state.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_5_4

49.36.06

Requirement: Mandatory

Evt_1, Dom_5_5

Test path: esmp_Sec10/checkBoxObject/input_click1.xhtml

Assertion: When the **click()** method of an input object is called, then the user action of clicking on that element MUST be replicated.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_1, Dom_5_5

Verification: Click each of the links "CheckBox 1" to "CheckBox 4". Verify that everytime, the check box next to the link gets checked or dechecked.

11.8 Submit and Reset Objects

49.37.01

Requirement: Mandatory

Dom_4_1

Test path: esmp_Sec10/submitResett/button_name1.xhtml

Assertion: The **name** property of a text input object is a string representing the name of input element.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_4_1

49.37.02

Requirement: Mandatory

Dom_4_2

Test path: esmp_Sec10/submitResett/button_type1.xhtml

Assertion: The **type** property of a text input object is a string representing the type of input element. This assertion applies to input elements of types " text, password, hidden and text-area ".

Traceability:

[S60KM2.5REQ]	Req ID: Dom_4_2

49.37.03

Requirement: Mandatory

Dom_4_3

Test path: esmp_Sec10/submitResett/button_value1.xhtml

Assertion: The **value** property of an input object returns the string currently displayed in the field. This assertion applies to input elements of types " text, password and text-area "

Traceability:

[S60KM2.5REQ]	Req ID: Dom_4_3

49.37.04

Requirement: Mandatory

Dom_4_3

Test path: esmp_Sec10/submitResett/buttont_value2.xhtml

Assertion: The displayed text in an input field can be modified through script, by assigning a value to the **value** property of that input object.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_4_3

Verification: Verify that the four buttons on the page are labeled “Button 1” to “Button 4” in that order.

49.37.05

Requirement: Mandatory

Evt_1, Dom_4_4

Test path: esmp_Sec10/radioButtonObject/submit_click1.xhtml

Assertion: When the **click()** method of an input object of type “submit” is called, then the user action of clicking on that element MUST be replicated. This means the associated form must be submitted.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_1, Dom_4_4

Verification: Click the hyperlink “Submit link” to submit the form.

49.37.06

Requirement: Mandatory

Evt_1, Dom_4_4

Test path: esmp_Sec10/radioButtonObject/reset_click1.xhtml

Assertion: When the **click()** method of an input object of type “reset” is called, then the user action of clicking on that element MUST be replicated. This means the associated form controls must be reset to their default values.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_1, Dom_4_4

Verification: Note the default values of the form fields. Modify the values in the form fields. Click the hyperlink “Reset Link”. Verify that all of the form fields are reset to their default values.

11.9 Select Element Object

49.38.01

Requirement: Mandatory

Dom_7_1

Test path: esmp_Sec10/selectObject/select_length1.xhtml

Assertion: The **name** property of a **select** object MUST return a string value of the NAME attribute on that SELECT element in the mark-up.

Assertion: The **length** property of a **select** object MUST return an integer, which is equal to the number of **option**s in that **select** list.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_7_1

49.38.02

Requirement: Mandatory

Dom_7_2

Test path: esmp_Sec10/selectObject/select_type1.xhtml

Assertion: The **type** property of a **select** object MUST return the string "select-one", if that SELECT element in the mark-up has no MULTIPLE attribute set on it.

Assertion: The **type** property of a **select** object MUST return the string "select-multiple", if that SELECT element in the mark-up has MULTIPLE attribute set on it.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_7_2

49.38.03

Requirement: Mandatory

Dom_7_4

Test path: esmp_Sec10/selectObject/selected_index1.xhml

Assertion: The **selectedIndex** property of a **select** object MUST return an integer, which is equal to the index of currently selected option of that select list.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_7_4

49.38.04

Requirement: Mandatory

Dom_7_4

Test path: esmp_Sec10/selectObject/selected_index2.xhml

Assertion: When an integer value N is assigned to the **selectedIndex** property of a **select** object in the script, where N is <= number of options, then the (N+1)th option of that select list MUST be displayed selected.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_7_4

Verification: Verify that the option 1 is pre-selected in the select list.

49.38.05

Requirement: Mandatory

Dom_7_6

Test path: esmp_Sec10/selectObject/select_focus1.xhml

Assertion: When the **focus()** method of an **select** object is called, then that select list in the display MUST receive focus.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_7_6

Verification: Verify that the focus in on select list, by default.

49.38.06

Requirement: Mandatory

Dom_7_5, Dom_9

Test path: esmp_Sec10/selectObject/options_length1.xhml

Assertion: The **options** property of a **select** object MUST return an array of the **option** objects of that select object.

Assertion: The **options.length** property of a **select** object MUST return an integer, which is equal to the number of options present in that select list currently.

Assertion: The **options[n]** property of a **select** object MUST return the option object with index **n** of that select list, where n < **options.length** .

Traceability:

[S60KM2.5REQ]	Req ID: Dom_7_5, Dom_9

49.38.07

Requirement: Mandatory

Evt_5

Test path: esmp_Sec10/selectObject/select_onchange1.xhml

Assertion: When a **onChange** event is defined on a select element with an event-handler E, then changing the selection of that select list, MUST call the event-handler E.

Traceability:

[S60KM2.5REQ]	Req ID: Evt_5

Verification: Change the selection in the select list. Verify that an alert message appears, which says “Option changed”.

11.10 Option Element Object

49.38.08

Requirement: Mandatory

Test path: esmp_Sec10/selectObject/option_selected1.xhml

Assertion: The **options[n].selected** property of a **select** object MUST return boolean true, if the (n+1)th option of that select list is currently in selected state.

Assertion: The **options[n]. selected** property of a **select** object MUST return boolean false, if the (n+1)th option of that select list is currently not in a selected state.

Traceability:

[S60KM2.5REQ]	Req ID:

49.38.09

Requirement: Mandatory

Test path: esmp_Sec10/selectObject/option_selected2.xhml

Assertion: When the **options[n].selected** property of a **select** object is assigned boolean true using the script, then the status of the (n+1)th option of that select list MUST be changed to selected.

Assertion: When the **options[n].selected** property of a **select** object is assigned boolean false using the script, then the status of the (n+1)th option of that select list MUST be changed to de-selected.

Traceability:

[S60KM2.5REQ]	Req ID:

49.38.10

Requirement: Mandatory

Dom_8_1, Dom_8_2

Test path: esmp_Sec10/selectObject/option_value1.xhml

Assertion: The **options[n].value** property of a **select** object MUST return the string value of the VALUE attribute of the (n+1}th option of that select list .

Assertion: The **options[n].text** property of a **select** object MUST return the text of the (n+1)th option of that select list.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_8_1, Dom_8_2

49.38.11

Requirement: Mandatory

Dom_7_3

Test path: esmp_Sec10/selectObject/selectSingle_value1.xhml

Assertion: The **value** property of a **select** object MUST return the value of currently selected option, in case of select element of type single.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_7_3

Verification:

Step 1: Verify that page titled "selectSingle value 1" is loaded.

Step 2: Verify that page contains a single selectable select list and a hyperlink “Link”.

Step 3: Activate the select list and modify the selection.

Step 4: Activate the hyperlink “Link”.

Step 5: Verify that an alert message appears displaying the option currently selected.

Step 6: Repeat steps 3,4 and 5 above two or three times.

49.38.12

Requirement: Mandatory

Dom_7_3

Test path: esmp_Sec10/selectObject/selectMultiple_value1.xhml

Assertion: The **value** property of a **select** object MUST return the value of first option among the currently selected options, in case of select element of type multiple.

Traceability:

[S60KM2.5REQ]	Req ID: Dom_7_3

Verification:

Step 1: Verify that page titled "selectMultiple value 1" is loaded.

Step 2: Verify that page contains a multiple selectable select list and a hyperlink “Link”.

Step 3: Activate the select list and modify the selection.

Step 4: Activate the hyperlink “Link”.

Step 5: Verify that an alert message appears displaying the first of the options among the currently selected options.

Step 6: Repeat steps 3,4 and 5 above two or three times.

11.11 Button Element Object

To be included in future versions of the document.

11.12 Screen Object

To be included in future versions of the document.

12. BROWSER DOM2 CORE OBJECTS

12.1 DOMException Object

12.1.1 Version History

49.41.01

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: esmp_Sec11/objDOMException/DOMExc_version.xhtml

Assertion: The **DOMException** object MUST maintain an enumerable, read-only version property in the format "M.m.I.i".

Traceability:

[ESMP]	Section 11.1.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

12.1.2 Properties

49.41.02 Test Type:

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: esmp_Sec11/objDOMException/DOMExc_properties.xhtml

Assertion: The **message** property of the **DOMException** object MUST contain a string whose value is implementation specific.

Traceability:

[ESMP]	Section 11.1.1, Sec 6.11

49.41.03 Test Type:

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: esmp_Sec11/objDOMException/DOMExc_properties_2.xhtml

Assertion: The **code** property of the **DOMException** object MUST be an integer, which is the value of the exception thrown.

Traceability:

[ESMP]	Section 11.1.1, Sec 6.11

12.1.3 Constants

49.41.04

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: When referring to a DOM2 Core object, if the index or size is negative, or greater than the allowed value, then the compiler MUST throw the DOMException object, whose name is “INDEX_SIZE_ERR” and code value is 1.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.05

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to insert any node in a place where it doesn't belong, then the compiler MUST throw the DOMException object, whose name is "HIERARCHY_REQUEST_ ERR” and code value is 3.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.06

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to use a node in a different document, than the one in which it is created, then the compiler MUST throw the DOMException object, whose name is “WRONG_ DOCUMENT_ERR” and code value is 4.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.07

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to specify an invalid or illegal character, such as in a name, then the compiler MUST throw the DOMException object, whose name is “INVALID_ CHARACTER_ ERR” and code value is 5.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.08

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to specify data for a node, which does not support data, then the compiler MUST throw the DOMException object, whose name is “NO_DATA_ALLOWED_ ERR” and code value is 6.

Traceability:

[ESMP]	Section 11.1.3

[DOM2CORE]	Section 1.2

49.41.09

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to modify an object, where modifications are not allowed, then the compiler MUST throw the DOMException object, whose name is "NO_MODIFICATION_ ALLOWED_ERR" and code value is 7.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.10

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to reference a node which does not exist, then the compiler MUST throw the DOMException object, whose name is "NOT_FOUND_ERR" and code value is 8.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.11

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If the implementation does not support the requested object or operation, then the compiler MUST throw the DOMException object, whose name is “NOT_SUPPORTED_ERR” and code value is 9.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.12

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to add an attribute that is already in use elsewhere, then the compiler MUST throw the DOMException object, whose name is “INUSE_ATTRIBUTE_ERR” and code value is 10.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.13

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to use an object that is not or is no longer usable, then the compiler MUST throw the DOMException object, whose name is “INVALID_STATE_ERR” and code value is 11.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.14

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to specify an invalid or illegal string, then the compiler MUST throw the DOMException object, whose name is “SYNTAX_ERR” and code value is 12.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.15

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to modify the type of underlying object, then the compiler MUST throw the DOMException object, whose name is “INVALID_MODIFICATION_ERR” and code value is 13.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

49.41.16

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made to create or change an object in a way which is incorrect with regard to namespaces, then the compiler MUST throw the DOMException object, whose name is “NAMESPACE_ERR” and code value is 14.

Traceability:

[ESMP]	Section 11.1.3

[DOM2CORE]	Section 1.2

49.41.17

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-042

Test path: NA

Assertion: If an attempt is made for an operation or parameter, which is not supported by underlying object, then the compiler MUST throw the DOMException object, whose name is “INVALID_ACCESS_ERR” and code value is 15.

Traceability:

[ESMP]	Section 11.1.3
[DOM2CORE]	Section 1.2

12.2 Node Object

12.2.1 Version History

49.42.01 Test Type: Manual

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/node_version.xhtml

Assertion: The **node** object MUST maintain an enumerable, read-only version property in the format "M.m.I.i".

Traceability:

[ESMP]	Section 11.2.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

49.42.02 Test Type: Manual

Requirement Type: Optional

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/node_enum.xhtml

Assertion: The **node** object MUST be enumerable.

Traceability:

[ESMP]	Section 11.2.1, Sec 6.2.1

Verification: Verify that any implementation specific extensions to the Node object are listed out.

12.2.2 Properties

12.2.2.1 nodeName

This property is not supported by [ESMP].

12.2.2.2 nodeValue

This property is not supported by [ESMP].

12.2.2.3 nodeType

49.42.03

Requirement Type: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/node_type_num.xhtml

Assertion: The **nodeType** property of a node object is of type ‘number’.

Traceability:

[ESMP]	Section 11.2.2.3

49.42.04

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/node_type_readOnly.xhtml

Assertion: The **nodeType** property of a node object is a read-only property.

Traceability:

[ESMP]	Section 11.2.2.3

49.42.05

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/node_type1.xhtml

Assertion: The value of the **nodeType** property of an element type node is 1.

Traceability:

[ESMP]	Section 11.2.2.3

49.42.06

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/node_type3.xhtml

Assertion: The value of the **nodeType** property of a text type node is 3.

Traceability:

[ESMP]	Section 11.2.2.3

49.42.07

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/node_type9.xhtml

Assertion: The value of the **nodeType** property of a document type node is 9.

Traceability:

[ESMP]	Section 11.2.2.3

49.42.08

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/node_type0.xhtml

Assertion: The value of the **nodeType** property of any un-supported node is 0.

Traceability:

[ESMP]	Section 11.2.2.3

12.2.2.4 parentNode

49.42.09

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/parentNode.xhtml

Assertion: The **parentNode** property of a node object is a reference to its parent node.

Traceability:

[ESMP]	Section 11.2.2.4

49.42.10

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/parentNode_null.xhtml

Assertion: The **parentNode** property of an unconnected node or a node without any parent node is set to null.

Traceability:

[ESMP]	Section 11.2.2.4

49.42.11

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/parentNode_readonly.xhtml

Assertion: The **parentNode** property of a node object is a read-only property.

Traceability:

[ESMP]	Section 11.2.2.4

12.2.2.5 childNodes

49.42.12

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/childNodes1.xhtml

Assertion: When a node object has one or more child nodes, then its **childNodes** property is an array of its child nodes.

Traceability:

[ESMP]	Section 11.2.2.5

49.42.13

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/childNodes2.xhtml

Assertion: When a node object has one or more child nodes, then its **childNodes** property is an array of its child nodes.

Traceability:

[ESMP]	Section 11.2.2.5

49.42.14

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/childNodes_null1.xhtml

Assertion: When a node object has no child nodes, then its **childNodes** property is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.5

49.42.15

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/childNodes_null2.xhtml

Assertion: The **childNodes** property of an unconnected node object is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.5

49.42.16

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/childNodes_readonly.xhtml

Assertion: The **childNodes** property of a node object is a read-only property.

Traceability:

[ESMP]	Section 11.2.2.5

12.2.2.6 firstChild

49.42.17

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/firstChild1.xhtml

Assertion: When a node object has more than one child node, then its **firstChild** property is a reference to its first child node.

Traceability:

[ESMP]	Section 11.2.2.6

49.42.18

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/firstChild2.xhtml

Assertion: When a node object has only one child node, then its **firstChild** property is a reference to that child node.

Traceability:

[ESMP]	Section 11.2.2.6

49.42.19

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/firstChild_null1.xhtml

Assertion: When a node object has no child nodes, then its **firstChild** property is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.6

49.42.20

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/firstChild_null2.xhtml

Assertion: The **firstChild** property of an unconnected node is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.6

49.42.21

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/firstChild_readonly.xhtml

Assertion: The **firstChild** property of a node object is a read-only property.

Traceability:

[ESMP]	Section 11.2.2.6

12.2.2.7 lastChild

49.42.22

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/lastChild1.xhtml

Assertion: When a node object has more than one child node, then its **lastChild** property is a reference to its last child node.

Traceability:

[ESMP]	Section 11.2.2.7

49.42.23

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/lastChild2.xhtml

Assertion: When a node object has only one child node, then its **lastChild** property is a reference to that child node.

Traceability:

[ESMP]	Section 11.2.2.7

49.42.24

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/lastChild_null1.xhtml

Assertion: When a node object has no child nodes, then its **lastChild** property is set to null.

Traceability:

[ESMP]	Section 11.2.2.7

49.42.25

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/lastChild_null2.xhtml

Assertion: The **lastChild** property of an unconnected node is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.7

49.42.26

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/lastChild_readonly.xhtml

Assertion: The **lastChild** property of a node object is a read-only property.

Traceability:

[ESMP]	Section 11.2.2.7

49.42.27

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/firstChild_is_lastChild.xhtml

Assertion: When a node object has only one child node, then both of its **firstChild** and **lastChild** properties refer to that same child node.

Traceability:

[ESMP]	Section 11.2.2.6, Section 11.2.2.7

12.2.2.8 previousSibling

49.42.28

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/prevSibling.xhtml

Assertion: When a node object has any preceding nodes, then its **previousSibling** property is a reference to the most immediately preceding node.

Traceability:

[ESMP]	Section 11.2.2.8

49.42.29

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/prevSibling_null1.xhtml

Assertion: When a node object has no preceding nodes, then its **previousSibling** property is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.8

49.42.30

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/prevSibling_null2.xhtml

Assertion: The **previousSibling** property of an unconnected node is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.8

49.42.31

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/prevSibling_readonly.xhtml

Assertion: Any attempt to modify the read-only property **previousSibling** MUST be ignored.

Traceability:

[ESMP]	Section 11.2.2.8
[ECMA262]	Section 8.6.1

12.2.2.9 nextSibling

49.42.32

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/nextSibling1.xhtml

Assertion: When a node object has any following nodes, then its **nextSibling** property is a reference to the most immediately following node.

Traceability:

[ESMP]	Section 11.2.2.9

49.42.33

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/nextSibling_null1.xhtml

Assertion: When a node object has no following nodes, then its **nextSibling** property is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.9

49.42.34

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/nextSibling_null2.xhtml

Assertion: The **nextSibling** property of an unconnected node is set to **null**.

Traceability:

[ESMP]	Section 11.2.2.9

49.42.35

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/nextSibling_readonly.xhtml

Assertion: The **nextSibling** property of a node object is a read-only property.

Traceability:

[ESMP]	Section 11.2.2.9

49.42.36

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-043

Test path: esmp_Sec11/objNode/nextSibling_is_prevSibling.xhtml

Assertion: When node B is preceded by node A and followed by node C, then both the **nextSibling** property of node A and the **previousSibling** property of node C refer to the same node which is node B.

Traceability:

[ESMP]	Section 11.2.2.8, Sec 11.2.2.9

12.2.2.10 attributes

The attributes property is not supported in [ESMP].

12.2.2.11 ownerDocument

The ownerDocument property is not supported in [ESMP].

12.2.2.12 namespaceURI

The namespaceURI property is not supported in [ESMP].

12.2.2.13 prefix

The prefix property is not supported in [ESMP].

12.2.2.14 localName

The localName property is not supported in [ESMP].

12.2.3 Methods

12.2.3.1 hasAttributes()

49.42.37

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-044

Test path: esmp_Sec11/objNode/hasAttributes1.xhtml

Assertion 1: When a node has one or more attributes associated with it, then its **hasAttributes()** method returns Boolean true.

Traceability:

[ESMP]	Section 11.2.3.1

Assertion 2: The document context as defined by [DOM2CORE] MUST be accessible to the script, via the Browser Document Object.

Traceability:

[ESMP]	Section 7.1.3

49.42.38

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-044

Test path: esmp_Sec11/objNode/hasAttributes2.xhtml

Assertion: When a node has no attributes associated with it, then its **hasAttributes()** method returns Boolean false.

Traceability:

[ESMP]	Section 11.2.3.1

12.2.3.2 hasChildNodes()

49.42.39

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-044

Test path: esmp_Sec11/objNode/hasChildNodes1.xhtml

Assertion: When a node has one or more child nodes, then its **hasChildNodes()** method returns Boolean true.

Traceability:

[ESMP]	Section 11.2.3.2

49.42.40

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-044

Test path: esmp_Sec11/objNode/hasChildNodes2.xhtml

Assertion: When a node has no child nodes, then its **hasChildNodes()** method returns Boolean false.

Traceability:

[ESMP]	Section 11.2.3.2

49.42.41

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-044

Test path: esmp_Sec11/objNode/hasChildNodes_error1.xhtml

Assertion: When the **hasChildNodes()** method is used without any reference node, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.2.3.2

12.2.3.3 insertBefore()

49.42.42

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/insertBefore_append.xhtml

Assertion: When **insertBefore()** method of a node object A is called, where the argument passed to the method is another node object B, then node B MUST be appended as the last child of node A.

Traceability:

[ESMP]	Section 11.2.3.3

49.42.43

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/insertBefore_insert.xhtml

Assertion: When **insertBefore()** method of a node object A is called, where the arguments passed to the method are a node object B and an existing child object C of node A, then node B MUST be inserted as a child node of node A, before the existing child node C.

Traceability:

[ESMP]	Section 11.2.3.3

49.42.44

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/insertBefore_increase.xhtml

Assertion: When **insertBefore()** method of a node object A is called, where the argument passed to the method is another node object B, then node B MUST be appended as a child node of node A. Thus the number of child nodes of node A gets incremented by 1.

Traceability:

[ESMP]	Section 11.2.3.3

49.42.45

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/insertBefore_error1.xhtml

Assertion: When **insertBefore()** method of a node object A is called, where the arguments passed to the method are a node object B and a non-existing child object C of node A, then **NOT_FOUND _ERR** exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.2.3.3

49.42.46

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/insertBefore_error2.xhtml

Assertion: When **insertBefore()** method of a read-only node object A is called, where the arguments passed to the method are a node object B and a child object C of node A, then **NO_MODIFICATION_ALLOWED _ERR** exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.2.3.3

49.42.47

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/insertBefore_return.xhtml

Assertion: When **insertBefore()** method of a node object A is called, where the argument passed to the method is another node object B, then node B MUST be appended as a child node of node A. Then the method MUST return a reference to the newly inserted child node B.

Traceability:

[ESMP]	Section 11.2.3.3

12.2.3.4 replaceChild()

49.42.48

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/replaceChild1.xhtml

Assertion: When the **replaceChild()** method of a node object A is called, where the arguments passed to the method are a node object B and an existing child object C of node A, then the existing child node C MUST be replaced with the node B.

Traceability:

[ESMP]	Section 11.2.3.4

49.42.49

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/replaceChild2.xhtml

Assertion: When the **replaceChild()** method of a node object A is called, where the arguments passed to the method are a node object B and an existing child object C of node A, then the existing child node C MUST be replaced with the node B. Thus the number of child nodes of node A remains unchanged.

Traceability:

[ESMP]	Section 11.2.3.4

49.42.50

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/replaceChild_error1.xhtml

Assertion: When the **replaceChild()** method of a node object A is called, where only one node object is passed as the argument, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.2.3.4

49.42.51

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/replaceChild_error2.xhtml

Assertion: When the **replaceChild()** method of a node object A is called, where the arguments passed to the method are a node object B and a non-existing child object C of node A, then **NOT_FOUND_ERR** exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.2.3.4

49.42.52

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/replaceChild_error3.xhtml

Assertion: When the **replaceChild()** method of a read-only node object A is called, where the arguments passed to the method are a node object B and a child object C of node A, then **NO_MODIFICATION_ALLOWED_ERR** exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.2.3.4

49.42.53

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/replaceChild_return.xhtml

Assertion: When the **replaceChild()** method of a node object A is called, where the arguments passed to the method are a node object B and an existing child node C of node A, then the existing child node C MUST be replaced with the node B and the method MUST return a reference to the new child node B.

Traceability:

[ESMP]	Section 11.2.3.4

12.2.3.5 removeChild()

49.42.54

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/removeChild1.xhtml

Assertion: When the **removeChild()** method of a node object A is called, where the argument passed to the method is an existing child object C of node A, then node C MUST be removed as being child node of parent node A.

Traceability:

[ESMP]	Section 11.2.3.5

49.42.55

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/removeChild2.xhtml

Assertion: When the **removeChild()** method of a node object A is called, where the argument passed to the method is an existing child object C of node A, then node C MUST be removed as being child node of parent node A. Thus the number of child nodes of node A decreases by 1.

Traceability:

[ESMP]	Section 11.2.3.5

49.42.56

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/removeChild_error1.xhtml

Assertion: When the **removeChild()** method of a node object A is called, where the argument passed to the method is a non-existing child node C of node A, then **NOT_FOUND_ERR** exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.2.3.5

49.42.57

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/removeChild_error2.xhtml

Assertion: When the **removeChild()** method of a read-only node object A is called, where the argument passed to the method is a child node C of node A, then **NO_MODIFICATION_ALLOWED_ERR** exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.2.3.5

49.42.58

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/removeChild_error3.xhtml

Assertion: When the **removeChild()** method of a node object A is called, where no arguments are passed to the method , then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.2.3.5

49.42.59

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/removeChild_return.xhtml

Assertion: When the **removeChild()** method of a node object A is called, where the argument passed to the method is an existing child object C of node A, then node C MUST be removed as being child node of parent node A and the method MUST return a reference to the removed node C.

Traceability:

[ESMP]	Section 11.2.3.5

12.2.3.6 appendChild()

49.42.60

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/appendChild1.xhtml

Assertion: When the **appendChild()** method of a node object A is called, where the argument passed to the method is another node object B, then node B MUST be appended as the last child of node A.

Traceability:

[ESMP]	Section 11.2.3.6

49.42.61

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/appendChild2.xhtml

Assertion: When the **appendChild()** method of a node object A is called, where the argument passed to the method is another node object B, then node B MUST be appended as the last child of node A. Thus the number of child nodes of node A increases by 1.

Traceability:

[ESMP]	Section 11.2.3.6

49.42.62

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/appendChild_return.xhtml

Assertion: When the **appendChild()** method of a node object A is called, where the argument passed to the method is another node object B, then node B MUST be appended as the last child of node A and the method MUST return a reference to the newly appended node B.

Traceability:

[ESMP]	Section 11.2.3.6

49.42.63

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/appendChild_error1.xhtml

Assertion: When the **appendChild()** method of a read-only node object A is called, where the argument passed to the method is another node object B, then **NO_MODIFICATION_ALLOWED_ERR** exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.2.3.6

49.42.64

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/appendChild_error2.xhtml

Assertion: When the **appendChild()** method of a node object A is called, where no arguments are passed to the method, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.2.3.6

12.2.3.7 cloneNode()

49.42.65

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode1.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean false, then a copy of the node A is returned by the method.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.66

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode2.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean false and the node A does have one or more child nodes, then a copy of the node A alone is returned by the method, without any child nodes.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.67

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode3.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean false and the node A does have one or more child nodes, then a copy of the node A alone is returned by the method, without any child nodes. This copy is created but not attached to the document.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.68

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode4.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean false and the node A does have one or more child nodes, then a copy of the node A alone is returned by the method, without any child nodes. This copy is not attached to the document unless it is explicitly connected to the document later.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.69

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode5.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean false and the node A does have one or more child nodes, then a copy of the node A alone is returned by the method, without any child nodes. This copy without child nodes is attached to the document only when it is explicitly connected to the document later.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.70

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode_error6.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where no argument is passed, then a copy of the node A alone is returned by the method, without any child nodes.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.71

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode_error1.xhtml

Assertion: When the **cloneNode()** method is called without any reference node, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.72

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode_tree1.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean true, then a copy of the node A is returned by the method.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.73

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode_tree2.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean true and the node A does have one or more child nodes, then a copy of the node tree A (along with its child nodes) is returned by the method.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.74

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode_tree3.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean true and the node A does have one or more child nodes, then a copy of the node tree A (along with its child nodes) is returned by the method. This copy is created but not attached to the document.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.75

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode_tree4.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean true and the node A does have one or more child nodes, then a copy of the node tree A (along with its child nodes) is returned by the method. This copy is not attached to the document unless it is explicitly connected to the document later.

Traceability:

[ESMP]	Section 11.2.3.7

49.42.76

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objNode/cloneNode_tree5.xhtml

Assertion: When the **cloneNode()** method of a node object A is called, where the argument passed to the method is Boolean true and the node A does have one or more child nodes, then a copy of the node tree A (along with its child nodes) is returned by the method. This copy without child nodes is attached to the document only when it is explicitly connected to the document later.

Traceability:

[ESMP]	Section 11.2.3.7

12.3 DOM2 Document Object

12.3.1 Version History

49.43.01

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-045

Test path: esmp_Sec11/objDocument/document_version.xhtml

Assertion: The **document** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 11.3.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

12.3.2 Properties

No properties supported.

12.3.3 Methods

12.3.3.1 createElement()

49.43.02

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-048

Test path: esmp_Sec11/objDocument/createElement1.xhtml

Assertion: When **createElement()** method of the document object is called, where a tag name string is passed as argument, then a new element node is created which is of the type specified by the tag name. The method returns a reference to the newly created element object.

Traceability:

[ESMP]	Section 11.3.3.1

49.43.03

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-048

Test path: esmp_Sec11/objDocument/createElement_error1.xhtml

Assertion: When **createElement()** method of the document object is called, where a tag name string with invalid characters is passed as the argument, then the compiler MUST raise **INVALID_CHARACTER_ERR** exception.

Traceability:

[ESMP]	Section 11.3.3.1

49.43.04

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-048

Test path: esmp_Sec11/objDocument/createElement_error2.xhtml

Assertion: When **createElement()** method is called without any reference, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.3.3.1

12.3.3.2 createTextNode()

49.43.05

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-048

Test path: esmp_Sec11/objDocument/createTextNode1.xhtml

Assertion: When **createTextNode()** method of the document object is called, where a string is passed as argument, then a new Text node is created with the specified string as its value. The method returns a reference to this newly created text node.

Traceability:

[ESMP]	Section 11.3.3.2

49.43.06

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-048

Test path: esmp_Sec11/objDocument/createTextNode2.xhtml

Assertion: When **createTextNode()** method of the document object is called, where a string is passed as argument, then a new Text node is created with the specified string as its value. The method returns a reference to this newly created text node.

Traceability:

[ESMP]	Section 11.3.3.2

49.43.07

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-048

Test path: esmp_Sec11/objDocument/createTextNode_error1.xhtml

Assertion: When **createTextNode()** method is called without any reference, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.3.3.2

12.3.3.3 getElementsByTagName ()

49.43.08

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-047

Test path: esmp_Sec11/objDocument/getElementsByTagName1.xhtml

Assertion: When **getElementsByTagName()** method of the document object is called, where a string specifying an element name is passed as argument, then a reference to a nodeList, which contains the list of the elements in the document with the specified element name, is returned.

Traceability:

[ESMP]	Section 11.3.3.3

49.43.09

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-047

Test path: esmp_Sec11/objDocument/getElementsByTagName_error1.xhtml

Assertion: When **getElementsByTagName()** method of the document object is called, where a string specifying an element name is passed as argument and when there are no elements in the document, which matches this element name, then a Null is returned.

Traceability:

[ESMP]	Section 11.3.3.3

49.43.10

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-047

Test path: esmp_Sec11/objDocument/getElementsByTagName_error2.xhtml

Assertion: When **getElementsByTagName()** method is called without any reference, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.3.3.3

12.3.3.4 getElementById()

49.43.11

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-047

Test path: esmp_Sec11/objDocument/getElementById1.xhtml

Assertion: When **getElementById()** method of the document object is called, where a string specifying the value of an XML id attribute is passed as argument and if the document contains any element with matching id attribute, then a reference to that element is returned.

Traceability:

[ESMP]	Section 11.3.3.4

49.43.12

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-047

Test path: esmp_Sec11/objDocument/getElementById2.xhtml

Assertion: When **getElementById()** method of the document object is called, where a string specifying the value of an XML id attribute is passed as argument and if the document does not contain any element with matching id attribute, then null is returned.

Traceability:

[ESMP]	Section 11.3.3.4

49.43.13

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-047

Test path: esmp_Sec11/objDocument/getElementById_error1.xhtml

Assertion: When **getElementById()** method of the document object is called, where a string specifying the value of an XML id attribute is passed as argument and if the document contains more than one element with matching id attribute, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.3.3.4

49.43.14

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-047

Test path: esmp_Sec11/objDocument/getElementById_error2.xhtml

Assertion: When **getElementById()** method is called without any reference, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.3.3.4

12.4 NodeList Object

12.4.1 Version History

49.44.01

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList_version.xhtml

Assertion: The **nodeList** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 11.4.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

49.44.02

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList_enum.xhtml

Assertion: The **nodeList** object should be enumerable.

Traceability:

[ESMP]	Section 11.4.1, Sec 6.2.2

Verification: Verify that any implementation specific extensions to the nodeList object are listed out.

12.4.2 Properties

12.4.2.1 length

49.44.03

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList_length.xhtml

Assertion: The **length** property of a **nodeList** object is a number equal to the number of nodes in that nodeList.

Traceability:

[ESMP]	Section 11.4.2.1

49.44.04

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList_length_readonly.xhtml

Assertion: The **length** property of a **nodeList** object is a read-only property. Any attempt to modify its value MUST be ignored.

Traceability:

[ESMP]	Section 11.4.2.1
[ECMA262]	Section 8.6.1

12.4.3 Methods

12.4.3.1 Item()

49.44.05

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList1.xhtml

Assertion: When the **item()** method of a nodeList object is called using the syntax ***<nodeList>.item(N)***, where N is non-negative integer, then a reference to the (N+1)th node in the node list MUST be returned.

Traceability:

[ESMP]	Section 11.4.3.1

49.44.06

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList2.xhtml

Assertion: When the **item()** method of a nodeList object is called using the syntax ***<nodeList>.item(N)***, where N is non-negative integer, and if the node list contains equal to or less than N nodes, then **null** MUST be returned.

Traceability:

[ESMP]	Section 11.4.3.1

49.44.07

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList3.xhtml

Assertion: When the **item()** method of a nodeList object is called using the syntax ***<nodeList>.item(N)***, where N is any invalid value, then **null** MUST be returned.

Traceability:

[ESMP]	Section 11.4.3.1

49.44.08

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList4.xhtml

Assertion: When the **item()** method of a nodeList object is called using the syntax ***<nodeList>[N]***, where N is non-negative integer, then a reference to the (N+1)th node in the node list MUST be returned.

Traceability:

[ESMP]	Section 11.4.3.1

49.44.09

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList5.xhtml

Assertion: When the **item()** method of a nodeList object is called using the syntax ***<nodeList>[N]***, where N is non-negative integer, and if the node list contains equal to or less than N nodes, then **null** MUST be returned.

Traceability:

[ESMP]	Section 11.4.3.1

49.44.10

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList6.xhtml

Assertion: When the **item()** method of a nodeList object is called using the syntax *<nodeList>[N]*, where N is any invalid value, then **null** MUST be returned.

Traceability:

[ESMP]	Section 11.4.3.1

49.44.11

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-049

Test path: esmp_Sec11/objNodeList/nodeList_error1.xhtml

Assertion: When **item()** method is called without any reference, then an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.4.3.1

12.5 Element Object

12.5.1 Version History

49.45.01

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-050

Test path: esmp_Sec11/objElement/version.xhtml

Assertion: The **Element** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 11.5.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

49.45.02

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-050

Test path: esmp_Sec11/objElement/element_enum.xhtml

Assertion: The **Element** object MUST be enumerable.

Traceability:

[ESMP]	Section 11.5.1, Sec 6.2.2

Verification: Verify that any implementation specific extensions to the Element object are listed out.

12.5.2 Properties

12.5.2.1 tagName

49.45.03

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-050

Test path: esmp_Sec11/objElement/tagName1.xhtml

Assertion: The **tagName** property of an element object is the string value of the tag name of that element.

Traceability:

[ESMP]	Section 11.5.2.1

49.45.04

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-050

Test path: esmp_Sec11/objElement/tagName_error1.xhtml

Assertion: Attempt to modify the read-only **tagName** property MUST be ignored.

Traceability:

[ESMP]	Section 11.5.2.1
[ECMA262]	Section 8.6.1

49.45.05

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-050

Test path: esmp_Sec11/objElement/tagName_error2.xhtml

Assertion: When the **tagName** property is used without reference, an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.5.2.1

12.5.3 Methods

12.5.3.1 getAttribute()

49.45.06

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute1.xhtml

Assertion: When the **getAttribute()** method of an element object is called, where an attribute name is passed as argument and if the element object has a matching attribute, then the method MUST return the current value of the matching attribute as a string. This means, if the value of the requested attribute is not changed since the document is loaded, then the return value MUST be the initial document value.

Traceability:

[ESMP]	Section 11.5.3.1

49.45.07

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute2.xhtml

Assertion: When the **getAttribute()** method of an element object is called, where an attribute name is passed as argument and if the element object has a matching attribute, then the method MUST return the current value of the matching attribute as a string. This means, if the value of the requested attribute is changed since the document is loaded, then the return value MUST be the current value.

Traceability:

[ESMP]	Section 11.5.3.1

49.45.08

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute3.xhtml

Assertion: When the **getAttribute()** method of an element object is called, where an attribute name is passed as argument, then the method MUST make a case-insensitive search and return the current value of the matching attribute as a string.

Traceability:

[ESMP]	Section 11.5.3.1

49.45.09

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute4.xhtml

Assertion: When the **getAttribute()** method of an element object is called, where an attribute name is passed as the first argument and the Boolean false as the second argument, then the method MUST make an case-insensitive search and return the current value of the matching attribute as a string.

Traceability:

[ESMP]	Section 11.5.3.1

49.45.10

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute5.xhtml

Assertion: When the **getAttribute()** method of an element object is called, where an attribute name is passed as the first argument and the Boolean true as the second argument, then the method MUST make an case-sensitive search and return the current value of the matching attribute as a string.

Traceability:

[ESMP]	Section 11.5.3.1

49.45.11

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute6.xhtml

Assertion: When the **getAttribute()** method of an element object is called, where an attribute name is passed as the first argument , then the method MUST make an case-insensitive search and MUST return an empty string, if no matching attribute is found.

Traceability:

[ESMP]	Section 11.5.3.1

49.45.12

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute7.xhtml

Assertion: When the **getAttribute()** method of an element object is called, where an attribute name is passed as the first argument and the Boolean false as the second argument , then the method MUST make an case-insensitive search and MUST return an empty string, if no matching attribute is found.

Traceability:

[ESMP]	Section 11.5.3.1

49.45.13

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute8.xhtml

Assertion: When the **getAttribute()** method of an element object is called, where an attribute name is passed as the first argument and the Boolean true as the second argument, then the method MUST make an case-sensitive search and MUST return an empty string, if no matching attribute is found.

Traceability:

[ESMP]	Section 11.5.3.1

49.45.14

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getAttribute_error1.xhtml

Assertion: When the **getAttribute()** property is used without reference, an exception MUST be thrown.

Traceability:

[ESMP]	Section 11.5.3.1

12.5.3.2 setAttribute()

49.45.15

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/setAttribute1.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name A and an attribute value V are passed as arguments, then the method MUST make an case-insensitive search for the requested attribute A of the element E and if found, change its current value with the requested value V.

Traceability:

[ESMP]	Section 11.5.3.2

49.45.16

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/setAttribute2.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name A and an attribute value V and Boolean false are passed as arguments, then the method MUST make an case-insensitive search for the requested attribute A of the element E and if found, change its current value with the requested value V.

Traceability:

[ESMP]	Section 11.5.3.2

49.45.17

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/setAttribute3.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name A and an attribute value V and Boolean true are passed as arguments, then the method MUST make an case-sensitive search for the requested attribute A of the element E and if found, change its current value with the requested value V.

Traceability:

[ESMP]	Section 11.5.3.2

49.45.18

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/setAttribute_return.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name A and an attribute value V are passed as arguments, then the return value of the method MUST be none.

Traceability:

[ESMP]	Section 11.5.3.2

49.45.19

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-053

Test path: esmp_Sec11/objElement/setAttribute_new1.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name A and an attribute value V are passed as arguments, then the method MUST make an case-insensitive search for the requested attribute A of the element E and if not found, MUST create the attribute A with value V for the element E.

Traceability:

[ESMP]	Section 11.5.3.2

49.45.20

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-053

Test path: esmp_Sec11/objElement/setAttribute_new2.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name A and an attribute value V and Boolean false are passed as arguments, then the method MUST make an case-insensitive search for the requested attribute A of the element E and if not found, MUST create the attribute A with value V for the element E.

Traceability:

[ESMP] Section 11.5.3.2

49.45.21

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-053

Test path: esmp_Sec11/objElement/setAttribute_new3.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name A and an attribute value V and Boolean true are passed as arguments, then the method MUST make an case-sensitive search for the requested attribute A of the element E and if not found, MUST create the attribute A with value V for the element E.

Traceability:

[ESMP] Section 11.5.3.2

49.45.22

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/setAttribute_error1.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name containing illegal characters and an attribute value are passed as arguments, then the compiler MUST raise INVALID_CHARACTER_ERR exception.

Traceability:

[ESMP] Section 11.5.3.2

49.45.23

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/setAttribute_error2.xhtml

Assertion: When the **setAttribute()** method of an element object E is called, where an attribute name A and an attribute value V are passed as arguments, and if the attribute A of the element E is a read-only attribute, then NO_MODIFICATION_ALLOWED_ERR exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.5.3.2

12.5.3.3 removeAttribute()

49.45.24

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/removeAttribute1.xhtml

Assertion: When the **removeAttribute()** method of an element object E is called, where an attribute name A is passed as argument, then the method MUST make an case-insensitive search for the requested attribute A of the element E and if found, remove it from the referenced element E.

Traceability:

[ESMP]	Section 11.5.3.3

49.45.25

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/removeAttribute2.xhtml

Assertion: When the **removeAttribute()** method of an element object E is called, where an attribute name A is passed as argument, then the method MUST make an case-insensitive search for the requested attribute A of the element E and if found, remove it from the referenced element E. If thus removed attribute is required by DTD, then a new instance of the same attribute MUST be instantiated whose value is the default set by the DTD.

Traceability:

[ESMP]	Section 11.5.3.3

49.45.26

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/removeAttribute3.xhtml

Assertion: When the **removeAttribute()** method of an element object E is called, where an attribute name A and a Boolean false are passed as arguments, then the method MUST make an case-insensitive search for the requested attribute A of the element E and if found, remove it from the referenced element E.

Traceability:

[ESMP]	Section 11.5.3.3

49.45.27

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/removeAttribute4.xhtml

Assertion: When the **removeAttribute()** method of an element object E is called, where an attribute name A and a Boolean true are passed as arguments, then the method MUST make an case-sensitive search for the requested attribute A of the element E and if found, remove it from the referenced element E.

Traceability:

[ESMP]	Section 11.5.3.3

49.45.28

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/removeAttribute_return.xhtml

Assertion: When the **removeAttribute()** method of an element object E is called, then the return value of the method is none.

Traceability:

[ESMP]	Section 11.5.3.3

49.45.29

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-052

Test path: esmp_Sec11/objElement/removeAttribute_error1.xhtml

Assertion: When the **removeAttribute()** method of an element object E is called, where an attribute name A is passed as argument, and if the attribute A of the element E is a read-only attribute, then compiler MUST raise NO_MODIFICATION_ALLOWED_ERR exception.

Traceability:

[ESMP]	Section 11.5.3.3

12.5.3.4 getElementsByTagName()

49.45.30

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getElementsByTagName1.xhtml

Assertion: When the **getElementsByTagName()** method of an element object E is called, where a tag name T is passed as argument, then the method MUST return a node list of all child elements of element E, whose tag names match with the requested tag name.

Traceability:

[ESMP]	Section 11.5.3.4

49.45.31

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getElementsByTagName2.xhtml

Assertion: When the **getElementsByTagName()** method of an element object E is called, where a tag name T is passed as argument, then the method MUST return a node list of all child elements of element E, whose tag names match with the requested tag name irrespective of the case-sensitivity of the tag name.

Traceability:

[ESMP]	Section 11.5.3.4

49.45.32

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/getElementsByTagName_error1.xhtml

Assertion: When the **getElementsByTagName()** method of an element object E is called, where a tag name T is passed as argument and if none of the child elements of the element E has matching tag name, then the method MUST return a Null.

Traceability:

[ESMP]	Section 11.5.3.4

12.5.3.5 hasAttribute()

49.45.33

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/hasAttribute1.xhtml

Assertion: When the **hasAttributes()** method of an element object E is called, then the method MUST return a Boolean true if the referenced element E has any attributes associated with it.

Traceability:

[ESMP]	Section 11.5.3.5

49.45.34

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/hasAttribute2.xhtml

Assertion: When the **hasAttributes()** method of an element object E is called, then the method MUST return a Boolean false if the referenced element E has no attributes associated with it.

Traceability:

[ESMP]	Section 11.5.3.5

49.45.35

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-051

Test path: esmp_Sec11/objElement/hasAttribute_error1.xhtml

Assertion: When the **hasAttribute()** property is used without reference, an exception MUST NOT be thrown.

Traceability:

[ESMP]	Section 11.5.3.5

12.6 Text (Character Data) Object

12.6.1 Version History

49.46.01

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/version.xhtml

Assertion: The **text** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Section 11.6.1, Sec 6.2.1

Verification: Verify the version number in the format "M.m.I.i" is displayed.

49.46.02

Requirement: Optional

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/text_enum.xhtml

Assertion: The **text** object MUST be enumerable.

Traceability:

[ESMP]	Section 11.6.1, Sec 6.2.2

Verification: Verify that any implementation specific extensions to the text object are listed out.

12.6.2 Properties

12.6.2.1 data

49.46.03

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/data1.xhtml

Assertion: The **data** property of a text node is the current value of the text in the text node.

Traceability:

[ESMP]	Section 11.6.2.1

49.46.04

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/data2.xhtml

Assertion: By using the **data** property of a text node as a left-side assignment, its value can be re-assigned.

Traceability:

[ESMP]	Section 11.6.2.1

49.46.05

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/data3.xhtml

Assertion: The **data** property of a text node is the current value of the text in the text node. A minimum of 32-character data string MUST be supported as the value.

Traceability:

[ESMP]	Section 11.6.2.1

49.46.06

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/data4.xhtml

Assertion: By using the **data** property of a text node as a left-side assignment, its value can be re-assigned. A minimum of 32-character data string MUST be supported for re-assignment.

Traceability:

[ESMP]	Section 11.6.2.1

12.6.2.2 length

49.46.07

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/length1.xhtml

Assertion: The **length** property of a text node T is the count of characters available in the text of the text node.

Traceability:

[ESMP]	Section 11.6.2.2

49.46.08

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/length2.xhtml

Assertion: Any attempt to write to the read-only property **length** of a text-node object, MUST be ignored.

Traceability:

[ESMP]	Section 11.6.2.2
[ECMA262]	Section 8.6.1

12.6.3 Methods

12.6.3.1 appendData()

49.46.09

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/appendData1.xhtml

Assertion: When the **appendData()** method of a node object N is called, where a string S is passed as argument, then the string S MUST be appended to the end of any existing character data associated with the referenced node N.

Traceability:

[ESMP]	Section 11.6.3.1

49.46.10

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/appendData2.xhtml

Assertion: When the **appendData()** method of a node object N is called, where a string S is passed as argument, then the string S MUST be appended to the end of any existing character data associated with the referenced node N. The return value of the method MUST be none.

Traceability:

[ESMP]	Section 11.6.3.1

12.6.3.2 deleteData()

49.46.11

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/deleteData1.xhtml

Assertion: When the **deleteData()** method of a node object N is called, where two numbers representing offset X and count Y are passed as arguments, then Y number of the characters MUST be removed from the character data of the node N, starting from the X+1st character. The data property of this node object N MUST reflect this change.

Traceability:

[ESMP]	Section 11.6.3.2

49.46.12

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/deleteData2.xhtml

Assertion: When the **deleteData()** method of a node object N is called, where two numbers representing offset X and count Y are passed as arguments, then Y number of the characters MUST be removed from the character data of the node N, starting from the X+1st character. The length property of this node object N MUST reflect the change.

Traceability:

[ESMP]	Section 11.6.3.2

49.46.13

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/deleteData3.xhtml

Assertion: When the **deleteData()** method of a node object N is called, it does not return any value.

Traceability:

[ESMP]	Section 11.6.3.2

49.46.14

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/deleteData4.xhtml

Assertion: When the **deleteData()** method of a node object N is called, where two numbers representing offset X and count Y are passed as arguments and if the offset X is greater than the number of characters available in the character data of node N, then INDEX_SIZE_ERR exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.6.3.2

12.6.3.3 insertData()

49.46.15

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/insertData1.xhtml

Assertion: When the **insertData()** method of a node object N is called, where a number representing offset X and a string S are passed as arguments, then the supplied string S MUST be inserted into the character data of the node N, after first X existing characters. The data property of this node object N MUST reflect this change.

Traceability:

[ESMP]	Section 11.6.3.3

49.46.16

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/insertData2.xhtml

Assertion: When the **insertData()** method of a node object N is called, where a number representing offset X and a string S are passed as arguments, then the supplied string S MUST be inserted into the character data of the node N, after first X existing characters. The length property of this node object N MUST reflect this change.

Traceability:

[ESMP]	Section 11.6.3.3

49.46.17

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/deleteData3.xhtml

Assertion: When the **insertData()** method of a node object N is called, it does not return any value.

Traceability:

[ESMP]	Section 11.6.3.3

49.46.18

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/deleteData4.xhtml

Assertion: When the **insertData()** method of a node object N is called, where a number representing offset X and a string S are passed as arguments and if the offset X is greater than the number of characters available in the character data of node N, then INDEX_SIZE_ERR exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.6.3.3

12.6.3.4 replaceData()

49.46.19

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/replaceData1.xhtml

Assertion: When the **replaceData()** method of a node object N is called, where two numbers representing offset X and count Y and a string S are passed as arguments, then Y number of the characters of the node N, starting from the X+1st character MUST be replaced with the supplied string parameter S. The data property of the node object N MUST reflect this change.

Traceability:

[ESMP]	Section 11.6.3.4

49.46.20

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/replaceData2.xhtml

Assertion: When the **replaceData()** method of a node object N is called, where two numbers representing offset X and count Y and a string S are passed as arguments, then Y number of the characters of the node N, starting from the X+1st character MUST be replaced with the supplied string parameter S. The length property of the node object N MUST reflect this change.

Traceability:

[ESMP]	Section 11.6.3.4

49.46.21

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/replaceData3.xhtml

Assertion: When the **replaceData()** method of a node object N is called, its return value MUST be none.

Traceability:

[ESMP]	Section 11.6.3.4

12.6.3.5 substringData()

49.46.22

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/substringData1.xhtml

Assertion: When the **substringData()** method of a node object N is called, where two numbers representing offset X and count Y are passed as arguments, then Y number of the characters of the node N, starting from the X+1st character MUST be extracted and returned as a string.

Traceability:

[ESMP]	Section 11.6.3.5

49.46.23

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/substringData2.xhtml

Assertion: When the **substringData()** method of a node object N is called, where two numbers representing offset X and count Y are passed as arguments, then Y number of the characters of the node N, starting from the X+1st character MUST be extracted and returned as a string. The original character data of the node N MUST not be affected.

Traceability:

[ESMP]	Section 11.6.3.5

49.46.24

Requirement: Mandatory

SCR: ESMP-XMLDOMOBJECT-C-054

Test path: esmp_Sec11/objText/substringData3.xhtml

Assertion: When the **substringData()** method of a node object N is called, where two numbers representing offset X and count Y are passed as arguments, and if the offset X is greater than the number of characters available in the character data of node N, then INDEX_SIZE_ERR exception MUST be raised by the compiler.

Traceability:

[ESMP]	Section 11.6.3.5

13. CRYPTO OBJECT

13.1 Crypto Object

49.47.01

Requirement: Optional

SCR: None

Test Path: esmp_Crypto\version.xhtml

Assertion: The **crypto** object MUST maintain an enumerable, read-only version property in the format “M.m.I.i”.

Traceability:

[ESMP]	Sec 6.2.2

Verification: Verify that the version number is displayed in the form “M.m.I.i”.

49.47.02

Requirement: Mandatory

SCR: ECMACR-C-001

Sign_1

Test Path: esmp_Crypto\crypto03.xhtml

Assertion: The User Agent SHOULD use special signature keys that are distinct from authentication keys used for transport layer security.

Traceability:

[ECMACR]	Section 5.1.2.1.1

Verification: Only NR certificates associated with NR private keys are listed to user.

49.47.03

Requirement: Mandatory

SCR: None

Sign_1

Test Path: esmp_Crypto\crypto04.xhtml

Assertion: **StringToSign** is presented to the user and it MUST be human-readable.

Traceability:

[ECMACR]	Section 5.1.2.1.2

Verification: User is able to read **StringToSign**

49.47.04

Requirement: None

SCR: None

Sign_1

Test Path: esmp_Crypto\crypto05.xhtml

Assertion: StringToSign MUST be gracefully handled in the following cases:

- Empty string
- String with the length of 1024 characters.
- String that contains numeric and special character such as “!@#$%^&*”

Traceability: None

Verification: The user agent handles the cases given above without any crash or malfunctioning.

49.47.05

Requirement: Mandatory

SCR: None

Sign_1

Test Path: esmp_Crypto\crypto06.xhtml

Assertion: The user agent MUST ignore options it does not recognize.

Traceability:

[ECMACR Section 5.1.2.1.3

49.47.06

Requirement: Optional

SCR: ECMACR-C-006

Sign_1

Test Path: esmp_Crypto\crypto07.xhtml

Assertion: “nocert" Certicates option indicates that the certificate(s) should not be included in the result.

Traceability:

[ECMACR]	Section 5.1.2.1.3

Verification: The user agent does not show the list of certificates to user for selection.

49.47.07

Requirement: Optional

SCR: ECMACR-C-004

Sign_1

Test Path: esmp_Crypto\crypto08.xhtml

Assertion: If the CA option is set to “auto” and, no caNameString parameters are provided, the User Agent selects a certificate from the entire set of available certificates that signText can use for signing.

Traceability:

[ECMACR]	Section 5.1.2.1.3

Verification: The user agent does not show the list of certificates to user for selection although there are available certificates signed by different CAs.

49.47.08

Requirement: Mandatory

SCR: ECMACR-C-005

Sign_1

Test Path: esmp_Crypto\crypto09.xhtml

Assertion: If the CA option is set to "auto", signText automatically selects a certificate signed by a CA specified by one of the caNameString parameters. Crypto SignText operation succeeds.

Traceability:

[ECMACR]	Section 5.1.2.1.4

Verification: The user agent does not show the list of certificates to user for selection although there are available certificates signed by same CA. Crypto SignText operation succeeds.

49.47.09

Requirement: Mandatory

SCR: ECMACR-C-003

Sign_1

Test Path: esmp_Crypto\crypto10.xhtml

Assertion: If the CA option is set to "ask", signText displays all certificates in the certificate database that are signed by a CA identified by one of the caNameString parameters and invites the user to select one of them.

Traceability:

[ECMACR]	Section 5.1.2.1.4

Verification: The user agent shows the list of certificates signed by same CA to user for selection. Crypto SignText operation succeeds.

49.47.10

Requirement: Mandatory

SCR: ECMACR-C-003

Sign_1

Test Path: esmp_Crypto\crypto11.xhtml

Assertion: If the CA option is set to "ask" but no caNameString parameters are provided, signText displays all the certificates in the certificate database that can be used for signing.

Traceability:

[ECMACR]	Section 5.1.2.1.4

Verification: The user agent shows the list of certificates signed by different CAs to user for selection. Crypto SignText operation succeeds.

49.47.11

Requirement: Mandatory

SCR: ECMACR-C-010

Sign_1

Test Path: esmp_Crypto\crypto12.xhtml

Assertion: The value of “version” component of signedData is “1”.

Traceability:

[ECMACR]	Section 5.1.2.1.5

Verification: The application running on the server parses the information and sends back a response containing “Test Passed”.

49.47.12

Requirement: Mandatory

SCR: ECMACR-C-010

Sign_1

Test Path: esmp_Crypto\crypto13.xhtml

Assertion: The value of “digestAlgorithms” component of signedData is “sha-1”.

Traceability:

[ECMACR]	Section 5.1.2.1.5

Verification: The application running on the server parses the information and sends back a response containing “Test Passed”.

49.47.13

Requirement: Mandatory

SCR: ECMACR-C-010

Sign_1

Test Path: esmp_Crypto\crypto14.xhtml

Assertion: The value of “signerInfos.digestAlgorithm” component of signedData is “sha-1”.

Traceability:

[ECMACR]	Section 5.1.2.1.5

Verification: The application running on the server parses the information and sends back a response containing “Test Passed”.

49.47.14

Requirement: Mandatory

SCR: ECMACR-C-010

Sign_1

Test Path: esmp_Crypto\crypto15.xhtml

Assertion: The value of “encapContentInfo.eContentType” component of signedData is “id-data”.

Traceability:

[ECMACR]	Section 5.1.2.1.5

Verification: The application running on the server parses the information and sends back a response containing “Test Passed”.

49.47.15

Requirement: Mandatory

SCR: ECMACR-C-010

Sign_1

Test Path: esmp_Crypto\crypto16.xhtml

Assertion: The value of the content type attribute is “id-data”.

Traceability:

[ECMACR] Section 5.1.2.1.5

Verification: The application running on the server parses the information and sends back a response containing “Test Passed”.

49.47.16

Requirement: Optional

SCR: ECMACR-C-011, ECMACR-C-012

Sign_1

Test Path: esmp_Crypto\crypto17.xhtml

Assertion: The value the signing time attribute is the time that object was signed or random nonce.

Traceability:

[ECMACR]	Section 5.1.2.1.5

49.47.17

Requirement: Mandatory

SCR: ECMACR-C-031, ECMACR-C-032

Sign_1

Test Path: esmp_Crypto\crypto18.xhtml

Assertion: Either RSA or ECDSA MUST be supported by the client.

Traceability:

[ECMACR] Section 5.1.2.1.5

Verification: The application running on the server parses the information and sends back a response containing "Test Passed".

49.47.18

Requirement: Optional

SCR: ECMACR-C-013

Sign_1

Test Path: esmp_Crypto\crypto19.xhtml

Assertion: The signing certificate attribute SHOULD be present in case the signing certificate was indicated to the user.

Traceability:

[ECMACR]	Section 5.1.2.1.5

Verification: The application running on the server parses the information and sends back a response containing "Test Passed".

49.47.19

Requirement: Optional

SCR: ECMACR-C-014

Sign_1

Test Path: esmp_Crypto\crypto20.xhtml

Assertion: The signing certificate attribute should use the hash of the certificate and optionally the issuerSerial attribute.

Traceability:

[ECMACR]	Section 5.1.2.1.5

Verification: The application running on the server parses the information and sends back a response containing “Test Passed”.

49.47.20

Requirement: Optional

SCR: ECMACR-C-015

Sign_1

Test Path: esmp_Crypto\crypto21.xhtml

Assertion: Client can gracefully handle Certificate URLs by using Certificate URL attribute (the value of “SignerInfos.UnsignedAttrs”).

Traceability:

[ECMACR]	Section 5.1.2.1.6

Verification: The user agent shows the list of Certificate URLs. The application running on the server parses the information and sends back a response containing “Test Passed”.

49.47.21

Requirement: Optional

SCR: ECMACR-C-041

Sign_1

Test Path: esmp_Crypto\crypto22.xhtml

Assertion: The PIN MUST be entered separately for each signature operation.

Traceability:

[ECMACR]	Section 5.1.2.1.7

Verification: Signature keys are PIN protected and user is asked to enter PIN-NR before signature operation.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version –or- No previous version within OMA

A.2 Draft/Candidate Version 1.0 History

Document Identifier	Date	Sections	Description
Draft Versions OMA-ETS-ESMP_Test_Code-V1_0	28 Sep 2006	n/a	First draft
	24 Aug 2007	all	Incorporated CRs: OMA-IOP-BR0-2007-0173, OMA-IOP-BR0-2007-0174, OMA-IOP-BR0-2007-0190, OMA-IOP-BR0-2007-0191, OMA-IOP-BR0-2007-0211 IOP WG agreed Applied ETS template and prepared for TP approval as doc. # OMA-TP-2007-0356INP_ETS_ESMP_V1_0_and_ETS_ESMP_Test_Code_V1_0_for_Approval
Candidate Version OMA-ETS-ESMP_Test_Code-V1_0	09 Oct 2007	n/a	Status changed to candidate, TP R&A from 2007-10-03 to 2007-10-09. TP doc # OMA-TP-2007-0356R02-INP_ETS_ESMP_V1_0_and_ETS_ESMP_Test_Code_V1_0_for_candidate_Approval