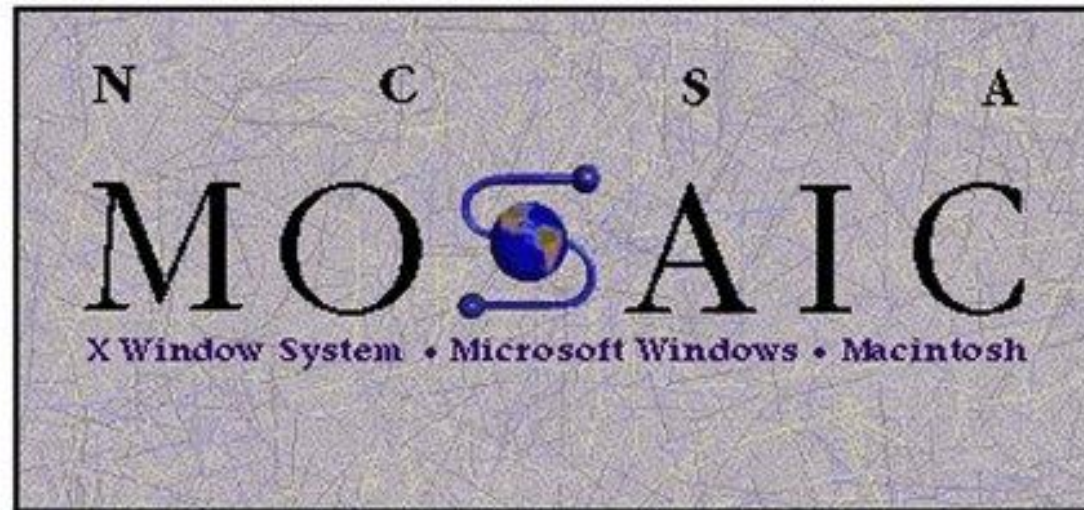




JAVASCRIPT |

MOSAIC



NETSCAPE



Netscape®

BRENDAN EICH



Java

Scheme

Self

LiveScript



Java

Scheme

Self

~~LiveScript~~
JavaScript



```
graph TD; Java --- Node(( )); Scheme --- Node; Self --- Node; Node --- LiveScript; LiveScript --- JavaScript;
```

Java

Scheme

Self

~~LiveScript~~

~~JavaScript~~

ECMAScript

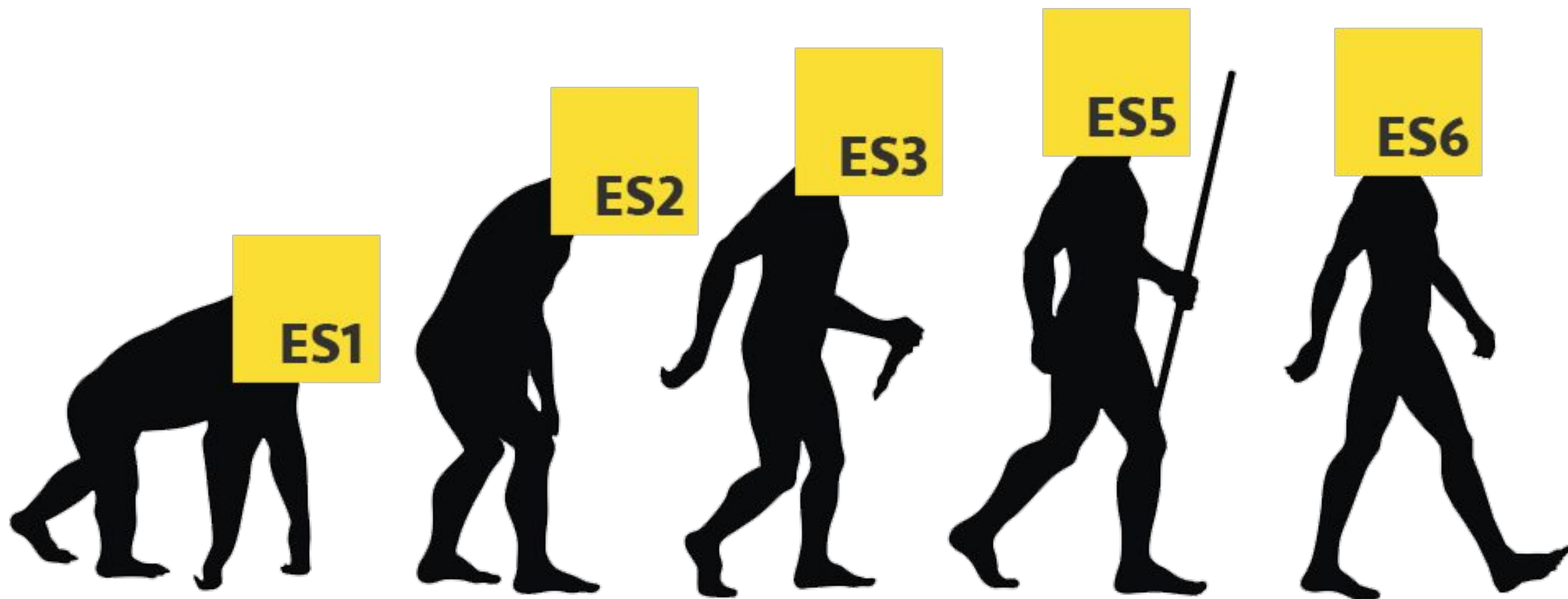
1997

1998

1999

2009

2015



WHAT IS FIRST CLASS OBJECT

A first class object is an entity that can be dynamically created, destroyed, passed to a function, returned as a value, and have all the rights as other variables in the programming language have.

- can be assigned to variable
- Can be passed as parameter
- Can be included as property from object
- Can be returned from functions

TYPES

Type	Value example	Value category
Number	1 , 5 , 6	Primitive value
String	“ahmed” , “hi ”	
Boolean	True , false	
Null	Null	
undefined	undefined	
Object	[1,2,3,4] Function() {} {id:1 } Regular expressions	Object values

NAN

Special number : Not a number

Any arithmetic operation with nan as input will have Nan as a result

NaN is not equal anything including NaN

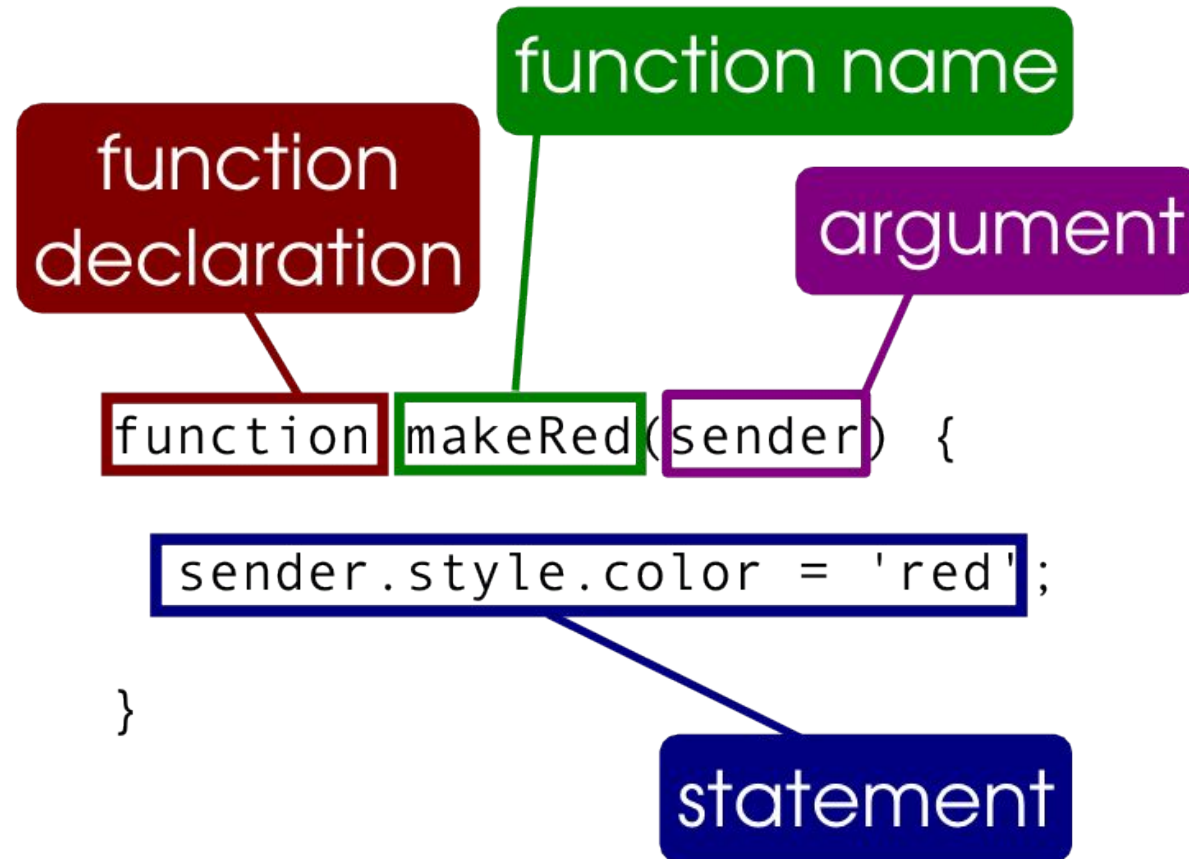
`NaN === NaN` is false

`NaN !== NaN` is true

`parseInt('ss') === NaN` !!

`isNaN(parseInt('ss'))`

FUNCTIONS



FUNCTIONS

- **Functions are first class object**
 - May be passed as an argument to a function
 - May be returned from a function
 - May be assigned to a variable
 - May be store in an object or array

VAR STATEMENT

It gets split into two parts :

- The declaration part is get **hoisted** to the top of the function initializing to undefined

```
var x = 5 ,y=4;
```

```
var x ,y;
```

```
x=5;
```

```
y=4
```

FUNCTIONS

Function statement

Is just a short hand for var statement with a function value

```
Function foo(){} 
```

Expand to

```
Var foo;
```

```
Foo = function (){} 
```

- * **the assignment of the function is also hoisted**

- * name is mandatory

Function expression

```
var foo = function() {}
```

Expand to

```
Var foo ;
```

```
Foo = function() {}
```

- * **function declaration is only hoisted**

- * name is optional

FUNCTIONS

```
Function foo1() { }
```

```
Var f = function foo2() { }
```

```
(function foo3(){ })
```

```
!function foo4(){ }
```


WHAT WILL BE ALERTED ?

```
function foo()  
{  
    function bar(){ return 3;}  
    return bar();  
    function bar(){return 8;}  
}  
alert(foo())
```

```
function foo()  
{  
    var bar = function() { return 3;}  
    return bar();  
    var bar = function() { return 8;}  
}  
alert(foo())
```

WHAT WILL BE ALERTED

```
function foo()  
{  
    return bar();  
    function bar() { return 3;}  
    var bar = function() { return 8;}  
}  
alert(foo())
```

```
console.log(foo())  
var foo = function()  
{  
    return bar();  
    function bar() { return 3;}  
    var bar = function() { return 8;}  
}
```