THE FLATIRON SCHOOL

/*-------------------------- Javascript Cheat Sheet -------------------------*/

Comments

// foo	single line
/* foo */	multi line

Data Types

var i = 123;	*number (integer), (whole)*
var f = 0.2;	*number (floating), (real)*
var t = "text";	*string (text)*
var b = true;	*boolean (true or false)*
var a = [1, "deux", 'trois'] ;	object *(array)*
var o = {1:"un", "deux":2}	*object (object literal)*
var z = function x(){ }	*function (object)*

Arithmetic Operators

(+, -, *, /, %, ++, --, unary -, unary +)

Assignment Operators

(=, *=, /=, %=, +=, -=, <<=, >>=, >>>=, &=, ^=, |=)

Bitwise Operators

(&, |, ^, ~, <<, >>, >>>)

Comparison Operators

(==, !=, ===, !==, >, >=, <, <=)

Logical Operators

(&&, ||, !)

String Operators

(+ and +=)

Event handlers

onAbort	*loading stopped*
onBlur	*focus lost*
onChange	*content modified*
onClick	*clicked*
onDblClick	*clicked twice*
onDragDrop	*moved*
onError	*not loaded*
onFocus	*focus entered*
onKeyDown	*key depressed*
onKeyPress	*key pressed*
onKeyUp	*key released*
onLoad	*just after loading*
onMouseDown	*mouse button depressed*
onMouseMove	*mouse moved*
onMouseOut	*mouse exited*
onMouseOver	*mouse on the element*
onMouseUp	*mouse button released*
onReset	*reset form button clicked*
onResize	*size of page changed*
onSelect	*element selected*
onSubmit	*submit form button clicked*
onUnload	*page exited*

Methods of objects (inherited by all objects)

toString()	*convert to a string*
toLocaleString()	*convert to a localized string*
valueOf()	*get the value*

Date methods

new Date()	*constructor, arguments: milliseconds, string, list*
getDate()	*day of the month*
getDay()	*day of the week*
getTime()	*number of milliseconds since 1/1/1970*
getYear()	*and getMonth/Hour/Minutes/Seconds*

String methods

charAt()	*character at the given position*
charCodeAt()	*code of a character*
concat()	*concatenate with the argument*
indexOf()	*position of a character*
lastIndexOf()	*position from the end*
localeCompare()	*localized comparison*
match()	*apply a regular expression*
replace()	*replace a substring*
search()	*search a substring*
slice()	*extract a part*
split()	*cut to build an array with parts*
substring()	*extract a part*
toLowerCase()	*convert to lowercase*
toUpperCase()	*convert to uppercase*
toLocaleLowerCase()	*localized lowercase*
toLocaleUpperCase()	*localized uppercase*

Array, index and methods

a["one"]=1	*assignment by indice*
a.one=1	*assignment by attribute*
delete a["one"]	*deletion by indice*
delete a.one	*deletion by attribute*
for(var k in a) {}	*iteration on the content*
concat()	*add a second array*
join()	*concatenate the elements into a string*
push()	*add an element*
pop()	*get and remove the last element*
reverse()	*invert the order of elements*
shift()	*insert an element at start*
slice()	*extract a sub-array*
splice()	*insert an array*
sort()	*sort the elements*
toString()	*return the array as a string*
unshift()	*get and remove the first element*

Number methods

new Number()	*constructor with a decimal/hexa/string argument*
toString()	*convert to a string*
toExponential()	*exponential form*
toPrecision()	*convert to a given number of decimals*

Functions

function x(a, b) { return y; }	*declaration*
y = x(1, "two")	*call*
var y = new x(1, "two")	*declaring a instance*
x.prototype.methodx = function() { }	*adding a method*

Built-in functions

eval()	*evaluate an expression*
parseInt()	*convert a string to an integer*
parseFloat()	*convert a string to a floating number*
isNaN()	*check if the content of a variable is valid*
ifFinite()	*check for overflow*
decodeURI()	*convert to a string*
decodeURIComponent()	*decode a component of the URL*
encodeURI()	*convert to file name*
encodeURIComponent()	*encode a component to URL*
escape()	*convert to URL parameters*
unescape()	*convert parameters to normal string*

Regular expressions, suffixes

g	*global*
i	*case-insensitive*
s	*single line*
m	*multi-lines*

Regular expressions, masks

^	*start of string*
$	*end of string*
(...)	*grouping*
!()	*but this group*
.	*any character*
(x|y)	*either x or y*
[xyz]	*among x y or z*
[^xyz]	*any but x y or z*
a?	*may holds a once*
a+	*at least a once*
a*	*zero or several times a*
a{5}	*five times a*
a{5,}	*at least five times a*
a{1, 4}	*a between 1 and 4 times*