JS CheatSheet

Hide comments

Ads

If - Else

```
if ((age >= 14) && (age < 19)) {        // logical condition
status = "Eligible.";                 // executed if condition is true
} else {                                // else block is optional
status = "Not eligible.";             // executed if condition is false
}
```

Switch Statement

```
switch (new Date().getDay()) {      // input is current day
case 6:                           // if (day == 6)
        text = "Saturday";
        break;
case 0:                           // if (day == 0)
        text = "Sunday";
        break;
default:                          // else...
        text = "Whatever";
}
```

Data Types

```
var age = 18;                           // number
var name = "Jane";                      // string
var name = {first:"Jane", last:"Doe"};  // object
var truth = false;                      // boolean
var sheets = ["HTML","CSS","JS"];       // array
var a; typeof a;                        // undefined
var a = null;                           // value null
```

Objects

```
var student = {                 // object name
firstName:"Jane",             // list of properties and values
lastName:"Doe",
age:18,
height:170,
fullName : function() {     // object function
   return this.firstName + " " + this.lastName;
}
};
student.age = 19;             // setting value
student[age]++;               // incrementing
name = student.fullName();  // call object function
```

Strings

```
var abc = "abcdefghijklmnopqrstuvwxyz";
var esc = 'I don\'t \n know';   // \n new line
var len = abc.length;           // string length
abc.indexOf("lmno");            // find substring, -1 if doesn't contain
abc.lastIndexOf("lmno");        // last occurance
abc.slice(3, 6);                // cuts out "def", negative values count f
abc.replace("abc","123");       // find and replace, takes regular express
abc.toUpperCase();              // convert to upper case
abc.toLowerCase();              // convert to lower case
abc.concat(" ", str2);          // abc + " " + str2
abc.charAt(2);                  // character at index: "c"
abc[2];                         // unsafe, abc[2] = "C" doesn't work
abc.charCodeAt(2);              // character code at index: "c" -> 99
abc.split(",");                 // splitting a string on commas gives an a
abc.split("");                  // splitting on characters
128.toString(16);               // number to hex(16), octal (8) or binary
```

Numbers and Math

Basics

On page script

```
<script type="text/javascript">  ...
</script>
```

Include external JS file

```
<script src="filename.js"></script>
```

Delay - 1 second timeout

```
setTimeout(function () {

}, 1000);
```

Functions

```
function addNumbers(a, b) {
return a + b; ;
}
x = addNumbers(1, 2);
```

Edit DOM element

```
document.getElementById("elementID").innerHTML = "Hello World!";
```

Output

```
console.log(a);             // write to the browser console
document.write(a);          // write to the HTML
alert(a);                   // output in an alert box
confirm("Really?");         // yes/no dialog, returns true/false depending
prompt("Your age?","0");    // input dialog. Second argument is the initia
```

Comments

```
/* Multi line
comment */
// One line
```

Variables x

```
var a;                      // variable
var b = "init";             // string
var c = "Hi" + " " + "Joe"; // = "Hi Joe"
var d = 1 + 2 + "3";        // = "33"
var e = [2,3,5,8];          // array
var f = false;              // boolean
var g = /()/;               // RegEx
var h = function(){};       // function object
const PI = 3.14;            // constant
var a = 1, b = 2, c = a + b;  // one line
let z = 'zzz';              // block scope local variable
```

Strict mode

```
"use strict";   // Use strict mode to write secure code
x = 1;          // Throws an error because variable is not declared
```

Values

```
false, true                 // boolean
18, 3.14, 0b10011, 0xF6, NaN  // number
"flower", 'John'            // string
undefined, null , Infinity  // special
```

Operators

```
a = b + c - d;      // addition, substraction
a = b * (c / d);    // multiplication, division
x = 100 % 48;       // modulo. 100 / 48 remainder = 4
a++; b--;           // postfix increment and decrement
```

Bitwise operators

&	AND	5 & 1 (0101 & 0001)	1 (1)
|	OR	5 | 1 (0101 | 0001)	5 (101)
~	NOT	~ 5 (~0101)	10 (1010)
^	XOR	5 ^ 1 (0101 ^ 0001)	4 (100)
<<	left shift	5 << 1 (0101 << 1)	10 (1010)
>>	right shift	5 >> 1 (0101 >> 1)	2 (10)
>>>	zero fill right shift	5 >>> 1 (0101 >>> 1)	2 (10)

Arithmetic

```
a * (b + c)         // grouping
person.age          // member
person[age]         // member
!(a == b)           // logical not
a != b              // not equal
typeof a            // type (number, object, function...)
x << 2  x >> 3      // minary shifting
a = b               // assignment
a == b              // equals
a != b              // unequal
a === b             // strict equal
a !== b             // strict unequal
a < b   a > b       // less and greater than
a <= b  a >= b      // less or equal, greater or eq
```

Numbers and Math 2

```
var pi = 3.141;
pi.toFixed(0);          // returns 3
pi.toFixed(2);          // returns 3.14 - for working with money
pi.toPrecision(2)       // returns 3.1
pi.valueOf();           // returns number
Number(true);           // converts to number
Number(new Date())      // number of milliseconds since 1970
parseInt("3 months");   // returns the first number: 3
parseFloat("3.5 days"); // returns 3.5
Number.MAX_VALUE        // largest possible JS number
Number.MIN_VALUE        // smallest possible JS number
Number.NEGATIVE_INFINITY// -Infinity
Number.POSITIVE_INFINITY// Infinity
```

Math.

```
var pi = Math.PI;       // 3.141592653589793
Math.round(4.4);        // = 4 - rounded
Math.round(4.5);        // = 5
Math.pow(2,8);          // = 256 - 2 to the power of 8
Math.sqrt(49);          // = 7 - square root
Math.abs(-3.14);        // = 3.14 - absolute, positive value
Math.ceil(3.14);        // = 4 - rounded up
Math.floor(3.99);       // = 3 - rounded down
Math.sin(0);            // = 0 - sine
Math.cos(Math.PI);      // OTHERS: tan,atan,asin,acos,
Math.min(0, 3, -2, 2);  // = -2 - the lowest value
Math.max(0, 3, -2, 2);  // = 3 - the highest value
Math.log(1);            // = 0 natural logarithm
Math.exp(1);            // = 2.7182pow(E,x)
Math.random();          // random number between 0 and 1
Math.floor(Math.random() * 5) + 1;  // random integer, from 1 to 5
```

Constants like Math.PI:

E, PI, SQRT2, SQRT1_2, LN2, LN10, LOG2E, Log10E

Global Functions ()

```
eval();                 // executes a string as if it was script code
String(23);             // return string from number
(23).toString();        // return string from number
Number("23");           // return number from string
decodeURI(enc);         // decode URI. Result: "my page.asp"
encodeURI(uri);         // encode URI. Result: "my%page.asp"
decodeURIComponent(enc);// decode a URI component
encodeURIComponent(uri);// encode a URI component
isFinite();             // is variable a finite, legal number
isNaN();                // is variable an illegal number
parseFloat();           // returns floating point number of string
parseInt();             // parses a string and returns an integer
```

Ads

```
a += b              // a = a + b (works with - * %...)
a && b              // logical and
a || b              // logical or
```

Dates

Wed Sep 14 2022 22:06:39 GMT+0100 (British Summer Time)

```
var d = new Date();
```

1663189599255 miliseconds passed since 1970

```
Number(d)
```

```
Date("2017-06-23");                 // date declaration
Date("2017");                       // is set to Jan 01
Date("2017-06-23T12:00:00-09:45");  // date - time YYYY-MM-DDTHH:MM:SSZ
Date("June 23 2017");               // long date format
Date("Jun 23 2017 07:45:00 GMT+0100 (Tokyo Time)"); // time zone
```

Get Times

```
var d = new Date();
a = d.getDay();     // getting the weekday

getDate();          // day as a number (1-31)
getDay();           // weekday as a number (0-6)
getFullYear();      // four digit year (yyyy)
getHours();         // hour (0-23)
getMilliseconds();  // milliseconds (0-999)
getMinutes();       // minutes (0-59)
getMonth();         // month (0-11)
getSeconds();       // seconds (0-59)
getTime();          // milliseconds since 1970
```

Setting part of a date

```
var d = new Date();
d.setDate(d.getDate() + 7); // adds a week to a date

setDate();          // day as a number (1-31)
setFullYear();      // year (optionally month and day)
setHours();         // hour (0-23)
setMilliseconds();  // milliseconds (0-999)
setMinutes();       // minutes (0-59)
setMonth();         // month (0-11)
setSeconds();       // seconds (0-59)
setTime();          // milliseconds since 1970)
```

Regular Expressions \n

```
var a = str.search(/CheatSheet/i);
```

Modifiers

i	*perform case-insensitive matching*
g	*perform a global match*
m	*perform multiline matching*

Patterns

****	*Escape character*
\d	*find a digit*
\s	*find a whitespace character*
\b	*find match at beginning or end of a word*
n+	*contains at least one n*
n*	*contains zero or more occurrences of n*
n?	*contains zero or one occurrences of n*
^	*Start of string*
$	*End of string*
\uxxxx	*find the Unicode character*
	Any single character

Errors

```
try {                       // block of code to try
undefinedFunction();
}
catch(err) {                // block to handle errors
console.log(err.message);
}
```

Throw error

```
throw "My error message";    // throw a text
```

Input validation

```
var x = document.getElementById("mynum").value; // get input value
try {
if(x == "")  throw "empty";                 // error cases
if(isNaN(x)) throw "not a number";
x = Number(x);
if(x > 10)   throw "too high";
}
catch(err) {                                    // if there's an error
document.write("Input is " + err);          // output error
console.error(err);                         // write the error in console
}
finally {
document.write("</br />Done");              // executed regardless of the
}
```

Error name values

RangeError	*A number is "out of range"*
ReferenceError	*An illegal reference has occurred*
SyntaxError	*A syntax error has occurred*
TypeError	*A type error has occurred*
URIError	*An encodeURI() error has occurred*

```
s
}
})
}
va
my
do
re
})
})
co
})
```

St
pe

Pr
Pr

Me
Pr
Pr