

# JavaScript Cheat Sheet

JavaScript is a a loosely typed and dynamic scripting language developed for the Internet, and a core technology of the the World Wide Web.

## Variables

Variable names can contain uppercase or lowercase letters (Aa to Zz), or numbers (0 to 9), or an underscore (\_). They cannot start with a number.

Data type	Global scope	Block scope
number	<code>var foo = 10;</code>	<code>let foo = 10;</code>
string	<code>var foo = "example";</code>	<code>let foo = "example";</code>
Boolean	<code>var foo = true;</code> <code>var foo = false;</code>	<code>let foo = true;</code>

## Binary operators

## Assignment shortcuts

<code>a &amp; b</code>	Bitwise AND (1 if both bits are 1)	<code>a += b;</code>	Addition	<code>a = a + b;</code>
<code>a   b</code>	Bitwise OR (1 if either bits are 1)	<code>a -= b;</code>	Subtraction	<code>a = a - b;</code>
<code>a ^ b</code>	Bitwise XOR (1 if bits differ)	<code>a *= b;</code>	Multiplication	<code>a = a * b;</code>
<code>a &lt;&lt; n</code>	Shift bits to the left	<code>a /= b;</code>	Division	<code>a = a / b;</code>
<code>a &gt;&gt; n</code>	Shift bits to the right	<code>a %= b;</code>	Modulo	<code>a = a % b;</code>

## Interacting with HTML

Set contents of any HTML element with ID **demo** to the contents of variable **foo**

```
document.getElementById("demo").innerHTML = foo;
```

Set contents of **button** element to the result of the **Date()** function

```
<button onclick="this.innerHTML = Date()">Timestamp</button>
```

## Common HTML events

click	drag / dragover / dragend	drop
focus	keydown / keypress / keyup	input
load	mousedown / mouseover / mouseup	submit



## Objects

An object can contain many properties (key value pairs), and is defined with the **const** keyword.

```
const penguin =  
{genus:"Eudyptula",  
specific:"minor" };
```

Retrieve property

```
penguin.genus;
```

Retrieve property (alternate syntax)

```
penguin["genus"];
```

Set property

```
Penguin.genus = "Linux"
```

## Functions

Provide a function name followed by required variables in parentheses, and function statements in curly braces.

```
function myFunction(arg1, arg2) {  
  return arg1 + arg2;  
}
```

## Methods

A function can be stored as object property. In this example, the **binomial** property is a method.

```
const penguin = {  
  genus: "Eudyptula",  
  specific: "Minor",  
  binomial: function() {  
    return this.genus + " " +  
    this.specific;  
  }  
};
```

## Arrays

Create an array

```
const foo = ["Linux","BSD"];
```

Get length of array

```
foo.length; // returns 2
```

Retrieve value

```
foo[0]; // returns "Linux"
```

Set value

```
foo[0] = "Fedora";
```

Add value

```
foo.push("illumos")
```

Get last element of array

```
foo[foo.length -1]; // returns "BSD"
```

Iterate over array

```
for (let i = 0; i < foo.length; i++) {  
  foo[i];  
}
```

