Javascript Basics

Including JavaScript in an HTML Page

```
<script type="text/javascript">
  //JS code goes here
</script>
```

Call an External JavaScript File

```
<script src="myscript.js"></script><code></code>
```

Including Comments

```
//
```
Single line comments

```
/* comment here */
```
Multi-line comments

Variables

var, const, let

`var`

The most common variable. Can be reassigned but only accessed within a function. Variables defined with var move to the top when code is executed.

`const`

Cannot be reassigned and not accessible before they appear within the code.

`let`

Similar to const, however, let variable can be reassigned but not re-declared.

Data Types

`var age = 23`

Numbers

`var x`

Variables

```
var a = "init"
```
Text (strings)

```
var b = 1 + 2 + 3
```
Operations

```
var c = true
```
True or false statements

```
const PI = 3.14
```
Constant numbers

```
var name = {firstName:"John", lastName:"Doe"}
```
Objects

Objects
```
var person = {
  firstName:"John",
  lastName:"Doe",
  age:20,
  nationality:"German"
};
```

Arrays
```
var fruit = ["Banana", "Apple", "Pear"];
```

Operators

Basic Operators

```
+      Addition
-      Subtraction
*      Multiplication
/      Division
(..)   Grouping operator
%      Modulus (remainder)
++     Increment numbers
--     Decrement numbers
```

Comparison Operators

```
==    Equal to
===   Equal value and equal type
!=    Not equal
!==   Not equal value or not equal type
>     Greater than
<     Less than
>=    Greater than or equal to
<=    Less than or equal to
?     Ternary operator
```

Logical Operators

```
&&    Logical and
||    Logical or
!     Logical not
```

Bitwise Operators

```
&     AND statement
|     OR statement
~     NOT
^     XOR
<<    Left shift
>>    Right shift
>>>   Zero fill right shift
```

Functions

```
function name(parameter1, parameter2, parameter3) {
  // what the function does
}
```

Loops

```
for (before loop; condition for loop; execute after loop) {
  // what to do during the loop
}
for
```

The most common way to create a loop in Javascript

```
while
```

Sets up conditions under which a loop executes

```
do while
```

Similar to the while loop, however, it executes at least once and performs a check at the end to see if the condition is met to execute again

```
break
```

Used to stop and exit the cycle at certain conditions

```
continue
```

Skip parts of the cycle if certain conditions are met

If - Else Statements

```
if (condition) {
  // what to do if condition is met
} else {
  // what to do if condition is not met
}
```

Strings

```
var person = "John Doe";
```

```
Escape Characters
\'    — Single quote
\"    — Double quote
\\    — Backslash
\b    — Backspace
\f    — Form feed
\n    — New line
\r    — Carriage return
\t    — Horizontal tabulator
```

Prutor.aig - Beginner's Javascript Cheat Sheet

\v   — Vertical tabulator

String Methods
charAt()

Returns a character at a specified position inside a string

charCodeAt()

Gives you the unicode of character at that position

concat()

Concatenates (joins) two or more strings into one

fromCharCode()

Returns a string created from the specified sequence of UTF-16 code units

indexOf()

Provides the position of the first occurrence of a specified text within a string

lastIndexOf()

Same as indexOf() but with the last occurrence, searching backwards

match()

Retrieves the matches of a string against a search pattern

replace()

Find and replace specific text in a string

search()

Executes a search for a matching text and returns its position

slice()

Extracts a section of a string and returns it as a new string

split()

Splits a string object into an array of strings at a specified position

substr()

Similar to slice() but extracts a substring depended on a specified number of characters

substring()

Also similar to slice() but can't accept negative indices

toLowerCase()

Convert strings to lowercase

toUpperCase()

Convert strings to uppercase

valueOf()

Returns the primitive value (that has no properties or methods) of a string object

```
[abc]      Find any of the characters between the brackets
[^abc]     Find any character not in the brackets
[0-9]      Used to find any digit from 0 to 9
[A-z]      Find any character from uppercase A to lowercase z
(a|b|c)    Find any of the alternatives separated with |
```

Metacharacters

```
.    — Find a single character, except newline or line terminator
\w   — Word character
\W   — Non-word character
\d   — A digit
\D   — A non-digit character
\s   — Whitespace character
\S   — Non-whitespace character
\b   — Find a match at the beginning/end of a word
\B   — A match not at the beginning/end of a word
\0   — NUL character
\n   — A new line character
\f   — Form feed character
\r   — Carriage return character
\t   — Tab character
\v   — Vertical tab character
```

Math Methods

abs(x)

Returns the absolute (positive) value of x

acos(x)

The arccosine of x, in radians

asin(x)

Arcsine of x, in radians

atan(x)

The arctangent of x as a numeric value

atan2(y,x)

Arctangent of the quotient of its arguments

ceil(x)

Value of x rounded up to its nearest integer

cos(x)

The cosine of x (x is in radians)

`exp(x)`

Value of Ex

`floor(x)`

The value of x rounded down to its nearest integer

`log(x)`

The natural logarithm (base E) of x

`max(x,y,z,...,n)`

Returns the number with the highest value

`min(x,y,z,...,n)`

Same for the number with the lowest value

`pow(x,y)`

X to the power of y

`random()`

Returns a random number between 0 and 1

`round(x)`

The value of x rounded to its nearest integer

`sin(x)`

The sine of x (x is in radians)

`sqrt(x)`

Square root of x

`tan(x)`

The tangent of an angle

Dealing with Dates

Setting Dates

`Date()`

Creates a new date object with the current date and time

`Date(2017, 5, 21, 3, 23, 10, 0)`

Create a custom date object. The numbers represent year, month, day, hour, minutes, seconds, milliseconds. You can omit anything you want except for year and month.

`Date("2017-06-23")`

Date declaration as a string

Pulling Date and Time Values

`getDate()`

Get the day of the month as a number (1-31)

`getDay()`

The weekday as a number (0-6)

`getFullYear()`

Year as a four digit number (yyyy)

`getHours()`

Get the hour (0-23)

`getMilliseconds()`

The millisecond (0-999)

`getMinutes()`

Get the minute (0-59)

`getMonth()`

Month as a number (0-11)

`getSeconds()`

Get the second (0-59)

`getTime()`

Get the milliseconds since January 1, 1970

`getUTCDate()`

The day (date) of the month in the specified date according to universal time (also available for day, month, fullyear, hours, minutes etc.)

`parse`

Parses a string representation of a date, and returns the number of milliseconds since January 1, 1970

Set Part of a Date

setDate()

Set the day as a number (1-31)

setFullYear()

Sets the year (optionally month and day)

setHours()

Set the hour (0-23)

setMilliseconds()

Set milliseconds (0-999)

setMinutes()

Sets the minutes (0-59)

setMonth()

Set the month (0-11)

setSeconds()

Sets the seconds (0-59)

setTime()

Set the time (milliseconds since January 1, 1970)

setUTCDate()

Sets the day of the month for a specified date according to universal time (also available for day, month, fullyear, hours, minutes etc.)

Element Methods

getAttributeNode()

Gets the specified attribute node

getAttributeNodeNS()

Returns the attribute node for the attribute with the given namespace and name

getElementsByTagName()

Provides a collection of all child elements with the specified tag name

getElementsByTagNameNS()

Returns a live HTMLCollection of elements with a certain tag name belonging to the given namespace

hasAttribute()

Returns true if an element has any attributes, otherwise false

hasAttributeNS()

Provides a true/false value indicating whether the current element in a given namespace has the specified attribute

removeAttribute()

Removes a specified attribute from an element

removeAttributeNS()

Removes the specified attribute from an element within a certain namespace

removeAttributeNode()

Takes away a specified attribute node and returns the removed node

setAttribute()

Sets or changes the specified attribute to a specified value

setAttributeNS()

Adds a new attribute or changes the value of an attribute with the given namespace and name

setAttributeNode()

Sets or changes the specified attribute node

setAttributeNodeNS()

Adds a new namespaced attribute node to an element

Working with the Browser

Window Properties

closed

Checks whether a window has been closed or not and returns true or false

defaultStatus

Sets or returns the default text in the statusbar of a window

document

Returns the document object for the window

frames

Returns all <iframe> elements in the current window

history

Provides the History object for the window

innerHeight

The inner height of a window's content area

innerWidth

The inner width of the content area

length

Find out the number of <iframe> elements in the window

location

Returns the location object for the window

name

Sets or returns the name of a window

navigator

Returns the Navigator object for the window

opener

Returns a reference to the window that created the window

outerHeight

The outer height of a window, including toolbars/ scrollbars

outerWidth

The outer width of a window, including toolbars/ scrollbars

pageXOffset

Number of pixels the current document has been scrolled horizontally

pageYOffset

Number of pixels the document has been scrolled vertically

parent

The parent window of the current window

screen

Returns the Screen object for the window

screenLeft

The horizontal coordinate of the window (relative to screen)

screenTop

The vertical coordinate of the window

screenX

Same as screenLeft but needed for some browsers

screenY

Same as screenTop but needed for some browsers

self

Returns the current window

status

Sets or returns the text in the statusbar of a window

top

Returns the topmost browser window

Window Methods

alert()

Displays an alert box with a message and an OK button

blur()

Removes focus from the current window

clearInterval()

Clears a timer set with setInterval()

clearTimeout()

Clears a timer set with setTimeout()

close()

Closes the current window

confirm()

Displays a dialogue box with a message and an OK and Cancel button

focus()

Sets focus to the current window

moveBy()

Moves a window relative to its current position

moveTo()

Moves a window to a specified position

open()

Opens a new browser window

print()

Prints the content of the current window

prompt()

Displays a dialogue box that prompts the visitor for input

resizeBy()

Resizes the window by the specified number of pixels

resizeTo()

Resizes the window to a specified width and height

scrollBy()

Scrolls the document by a specified number of pixels

scrollTo()

Scrolls the document to specific coordinates

setInterval()

Calls a function or evaluates an expression at specified intervals

setTimeout()

Calls a function or evaluates an expression after a specified interval

stop()

Stops the window from loading

Screen Properties

availHeight

Returns the height of the screen (excluding the Windows Taskbar)

availWidth

Returns the width of the screen (excluding the Windows Taskbar)

colorDepth

Returns the bit depth of the color palette for displaying images

height

The total height of the screen

pixelDepth

The color resolution of the screen in bits per pixel

width

The total width of the screen

Events

Mouse

onclick

The event occurs when the user clicks on an element

oncontextmenu

User right-clicks on an element to open a context menu

ondblclick

The user double-clicks on an element

onmousedown

User presses a mouse button over an element

onmouseenter

The pointer moves onto an element

onmouseleave

Pointer moves out of an element

onmousemove

The pointer is moving while it is over an element

onmouseover

When the pointer is moved onto an element or one of its children

onmouseout

User moves the mouse pointer out of an element or one of its children

onmouseup

The user releases a mouse button while over an element

Keyboard

onkeydown

When the user is pressing a key down

onkeypress

The moment the user starts pressing a key

onkeyup

The user releases a key

Form

onblur

When an element loses focus

onchange

The content of a form element changes (for <input>, <select>and <textarea>)

onfocus

An element gets focus

onfocusin

When an element is about to get focus

onfocusout

The element is about to lose focus

oninput

User input on an element

oninvalid

An element is invalid

onreset

A form is reset

onsearch

The user writes something in a search field (for <input="search">)

onselect

The user selects some text (for <input> and <textarea>)

onsubmit

A form is submitted

Drag

ondrag

An element is dragged

ondragend

The user has finished dragging the element

ondragenter

The dragged element enters a drop target

ondragleave

A dragged element leaves the drop target

ondragover

The dragged element is on top of the drop target

ondragstart

User starts to drag an element

ondrop

Dragged element is dropped on the drop target

Clipboard

oncopy

User copies the content of an element

oncut

The user cuts an element's content