**JAVASCRIPT**

**BROWSER OBJECT MODEL: THE WINDOW OBJECT**

**Properties**

* window.innerHeight: \*height of window (px)
* window.innerWidth: \*width of window (px)
* window.pageXOffset: \*distance document has been scrolled horizontally (px)
* window.pageYOffset: \*distance document has been scrolled vertically (px)
* window.screenX: \*X-coordinate of pointer, relative to top left corner of screen (px)
* window.screenY: \*Y-coordinate of pointer, relative to top left corner of screen (px)
* window.location: \*current URL of window object
* window.document: \*reference to document object
* window.history: \*reference to history object (contains details of the pages that have been viewed in that window or tab)
* window.history.length: \*number of items for history object
* window.screen: \*reference to screen object
* window.screen.width: \*accesses screen object and finds value of its width (px)
* window.screen.height: \*accesses screen object and finds value of its height (px)

**Methods**

* window.alert():\*creates dialog box with message (user must click OK to close it)
* window.open():\*opens new browser window with URL specified as parameter
* window.print():\*tells browser that the user wants to print content of current page

**THE DOCUMENT OBJECT MODEL: THE DOCUMENT OBJECT**

**Properties**

* document.title: \*title of current document
* document.lastModified: \*date on which document was last modified
* document.URL: \*returns string containing URL of current document
* document.domain: \*returns domain of current document

**Methods**

* document.write(): \*writes text to document
* document.getElementById():\*returns element with the matching Id
* document.querySelectorAll():\*returns list of elements that match a CSS selector, which is specified as a parameter
* document.createElement():\*creater new element
* document.createTextNode():\*creates new text node

**GLOBAL OBJECTS**

**STRING OBJECT**

**Properties**

* length \*returns number of characters in the string

**Methods**

* toUpperCase(): \*changes string to uppercase characters
* toLowerCase(): \*changes string to lowercase characters
* charAt(): \*takes an index number as a parameter, and returns the character found at that position
* indexOf(): \*returns index number of the first time a character is found within a string
* lastIndexOf(): \*returns index number of the last time a character is found within a string
* substring(): \*returns characters found within two index numbers, the character of the first index number is included but the character of the last one is not
* split(): \*when a character is specified, it splits the string each time it is found, then stores each individual part in an array
* trim(): \*removes whitespace from start and end of string
* replace(): \*takes one value that should be found, and another to replace it; it only replaces the first match that finds

**NUMBER OBJECT**

**Methods**

* isNaN(): \*checks if the value is not a number
* toFixed(): \*rounds to specified number of decimal places (returns a string)
* toPrecision(): \*rounds to total number of places (returns a string)
* toExponential(): \*returns a string representing the number in exponential notation.

**MATH OBJECT**

**Properties**

* Math.PI: \*returns pi

**Methods**

* Math.round(): \* rounds number to the nearest integer
* Math.sqrt(n): \*returns square root of positive number
* Math.ceil(): \*rounds number up to the nearest integer
* Math.floor():\*rounds number down to the nearest integer
* Math.random():\*generates a random number between 0 (inclusive) and 1 (not inclusive)

**DATE OBJECT**

**Methods**

* getDate() / setDate(): \*returns/sets day of month
* getDay() / setDay():\*returns/sets the day of the week
* getFullYear() / setFullYear():\*returns/sets the year
* getHours() / setHours():\*returns/sets the hour
* getMilliseconds() / setMilliseconds (): \*returns/sets the milliseconds
* getMinutes() / setMinutes():\*returns/sets the minutes
* getMonth() / setMonth():\*returns/sets the month
* getSeconds() / setSeconds():\*returns/sets the seconds
* getTime() / setTime(): \*number of milliseconds since January 1, 1970
* getTimezoneOffset():\*returns time zone offset in mins for locale
* toDateString():\*returns date as a human-readable string
* toTimeString():\*returns time as a human-readable string
* toString():\*returns a string representing the specified date