JavaScript cheat sheet

Variables

Name	Description
"var" keyword	"var" is a keyword used to tell JavaScript to make a new variable.
"let" keyword	An alternative to "var," "let" is another keyword for creating variables in JavaScript.
"const" keyword	"const" is similar to "let," but declares a fixed variable.

Operators

Arithmetic operators

Name	Symbol	Description
"Addition" operator	+	The "addition" operator adds numbers together.
"Subtraction" operator	-	The "subtraction" operator subtracts the right-hand value from the left-hand value.
"Multiplication" operator	*	The "multiplication" operator multiplies numbers together.
"Division" operator	/	The "division" operator divides the left-hand number by the right-hand number.
"Modulus" operator	%	The "modulus" operator returns a remainder after division.

Comparison operators

Name	Symbol	Description
"Equality" operator	==	Compares if two values are equal, regardless of data type. For example, "5 == 5.0" would return "true" even though the first value is an integer and the other is a floating-point number (a numeric value with decimal places) with the same numeric value.
"Strict equality" operator	===	Compares if two values are equal, including the data type. For example, "5 === 5.0" would return "false" because the first value is an integer and the other is a floating-point number, which is a different data type.
"Inequality" operator	!=	Checks if two values are not equal. It doesn't matter what type of values they are. For example, "5 != 10" would return "true" because 5 does not equal 10.
"Strict inequality" operator	!==	Checks if two values are not equal, including the data type. For example, "5!== 5.0" would return "true" because the first value is an integer and the other is a floating-point number, which is a different data type.
"Greater than" operator	>	Checks if the left value is greater than the right value. For example, "10 > 5" returns "true."
"Less than" operator	<	Checks if the left value is less than the right value. For example, "5 < 10" returns "true."
"Greater than or equal to" operator	>=	Checks if the left value is greater than or equal to the right value. For example, "10 >= 5" returns "true."
"Less than or equal to" operator	<=	Checks if the left value is less than or equal to the right value. For example, "5 <= 10" returns "true."

Logical operators

Name	Symbol	Description
"Logical AND" operator	&&	The "logical AND" operator is used to combine two or more conditions. It returns "true" only if all the conditions are true.
"Logical OR" operator	||	The "logical OR" operator is used to combine multiple conditions. And it returns "true" if at least one of the conditions is true. If all conditions are false, the result will be "false."
"Logical NOT" operator	!	The "logical NOT" operator is used to reverse the logical state of a single condition. If a condition is true, "!" makes it "false." And if a condition is false, "!" makes it "true."

Assignment operators

Name	Symbol	Description
"Basic assignment" operator	=	The "basic assignment" operator is used to assign a value to a variable.
"Addition assignment" operator	+=	This operator adds a value to the variable's current value and assigns the result to the variable.
"Subtraction assignment" operator	-=	This operator subtracts a value from the variable's current value and assigns the result to the variable.
"Multiplication assignment" operator	*=	This operator multiplies the variable's current value by a specified value and assigns the result to the variable.
"Division assignment" operator	/=	This operator divides the variable's current value by a specified value and assigns the result to the variable.

If-else statement

Name	Description
"if-else" statement	The "if-else" statement is a conditional statement that allows you to execute different blocks of code based on a condition.

Loops

Name	Description
"for" loop	A "for" loop is a loop that specifies "do this a specific number of times".
"while" loop	A "while" loop is a loop that indicates "keep doing this as long as something is true".
"do...while" loop	A "do...while" loop works similarly to "for" and "while" loops, but it has different syntax.
"for...in" loop	The "for...in" loop is used to iterate over the properties of an object (a data structure that holds key-value pairs).

Functions

Name	Description
"alert()" function	This function displays a message in a pop-up dialog box in the browser. It's often used for simple notifications, error messages, or getting the user's attention.
"prompt()" function	This function displays a dialog box where the user can enter an input. The input is returned as a string.
"confirm()" function	This function shows a confirmation dialog box with "OK" and "Cancel" buttons. It returns "true" if the user clicks "OK" and "false" if they click "Cancel."
"console.log()" function	This function is used to output messages and data to the browser's console.
"parseInt()" function	This function extracts and returns an integer from a string.
"parseFloat()" function	This function extracts and returns a floating-point number (a numeric value with decimal places).

Strings

Name	Description
"toUpperCase()" method	This method converts all characters in a string to uppercase.
"toLowerCase()" method	This method converts all characters in a string to lowercase.
"concat()" method	The "concat()" method is used to combine two or more strings and create a new string that contains the merged text.
"match()" method	The "match()" method is used to search a string for a specified pattern and return the matches as an array (a data structure that holds a collection of values—like matched substrings or patterns).
"charAt()" method	The "charAt()" method is used to retrieve the character at a specified index (position) within a string.
"replace()" method	The "replace()" method is used to search for a specified substring (a part within a string) and replace it with another substring.
"substr()" method	The "substr()" method is used to extract a portion of a string, starting from a specified index (position) and extending for a specified number of characters.

Events

Name	Description
"onclick" event	The "onclick" event executes a function or script when an HTML element (such as a button or a link) is clicked by a user.
"onmouseover" event	The "onmouseover" event occurs when a user moves the mouse pointer over an HTML element, such as an image, a button, or a hyperlink.
"onkeyup" event	The "onkeyup" is an event that occurs when a user releases a key on their keyboard after pressing it.
"onmouseout" event	The "onmouseout" event occurs when a user moves the mouse pointer out of the area occupied by an HTML element like an image, a button, or a hyperlink.
"onload" event	The "onload" event executes a function or script when a webpage or a specific element within the page (such as an image or a frame) has finished loading.
"onfocus" event	The "onfocus" event triggers when an HTML element like an input field receives focus or becomes the active element of a user's input or interaction.
"onsubmit" event	The "onsubmit" event triggers when a user submits an HTML form. Typically by clicking a "Submit" button or pressing the "Enter" key within a form field.

Numbers & math

Name	Description
"Math.abs()" method	This method returns the absolute value of a number, ensuring the result is positive.
"Math.round()" method	This method rounds a number up to the nearest integer.
"Math.max()" method	This method returns the largest value among the provided numbers or values.
"Math.min()" method	The "Math.min()" method returns the smallest value among the provided numbers or values.
"Math.random()" Method	This method generates a random floating-point number between 0 (inclusive) and 1 (exclusive).
"Math.pow()" Method	This method calculates the value of a base raised to the power of an exponent.
"Math.sqrt()" Method	This method computes the square root of a number.
"Number.isInteger()" Method	This method checks whether a given value is an integer. It returns true if the value is an integer and false if not.

Date objects

Name	Description
"getDate()" method	This method retrieves the current day of the month. The day is returned as an integer, ranging from 1 to 31.
"getDay()" method	This method retrieves the current day of the week. The day is returned as an integer, with Sunday being 0, Monday being 1, and so on. Up to Saturday being 6.
"getMinutes()" method	This method retrieves the minutes portion from the present date and time. The minutes will be an integer value, ranging from 0 to 59.
"getFullYear()" method	This method retrieves the current year. It'll be a four-digit integer.
"setDate()" method	This method sets the day of the month. By changing the day of the month value within the "date" object.