**String Function**

| **Function** | Syntax Description | **Sample** | **Output** |
| --- | --- | --- | --- |
| charAt() | str.charAt(index) Returns the character at the specified index. | "hello".charAt(1) | "e" |
| concat() | str1.concat(str2, ..., strN) Combines the text of two or more strings and returns a new string. | "hello".concat(" ", "world") | "hello world" |
| includes() | str.includes(searchString) Determines whether one string may be found within another string. | "hello".includes("ell") | True |
| indexOf() | str.indexOf(searchValue) Returns the index within the calling String object of the first occurrence of the specified value. | "hello".indexOf("e") | 1 |
| lastIndexOf() | str.lastIndexOf(searchValue) Returns the index within the calling String object of the last occurrence of the specified value. | "hello".lastIndexOf("l") | 3 |
| replace() | str.replace(searchValue, newValue) Returns a new string with some or all matches of a pattern replaced by a replacement. | "hello".replace("e", "a") | "hallo" |
| slice() | str.slice(beginIndex, endIndex) Extracts a section of a string and returns it as a new string. | "hello".slice(1, 4) | "ell" |
| split() | str.split(separator) Splits a String object into an array of strings by separating the string into substrings. | "hello world".split(" ") | ["hello", "world"] |
| substring() | str.substring(indexStart, indexEnd) Returns the part of the string between the start and end indexes, or to the end of the string. | "hello".substring(1, 4) | "ell" |
| toLowerCase() | str.toLowerCase() Returns the calling string value converted to lower case. | "HELLO".toLowerCase() | "hello" |
| toUpperCase() | str.toUpperCase() Returns the calling string value converted to uppercase. | "hello".toUpperCase() | "HELLO" |
| trim() | str.trim() Trims whitespace from both ends of a string. | " hello ".trim() | "hello" |

Sure! Here are some tables for different JavaScript APIs, including Array, Date, and Math functions:

**Array Functions**

| **Function** | Syntax Description | **Sample** | **Output** |
| --- | --- | --- | --- |
| push() | arr.push(element1, ..., elementN) Adds one or more elements to the end of an array and returns the new length of the array. | let arr = [1, 2]; arr.push(3); | [1, 2, 3] |
| pop() | arr.pop() Removes the last element from an array and returns that element. | let arr = [1, 2, 3]; arr.pop(); | [1, 2] |
| shift() | arr.shift() Removes the first element from an array and returns that element. | let arr = [1, 2, 3]; arr.shift(); | [2, 3] |
| unshift() | arr.unshift(element1, ..., elementN) Adds one or more elements to the beginning of an array and returns the new length of the array. | let arr = [1, 2]; arr.unshift(0); | [0, 1, 2] |
| slice() | arr.slice(begin, end) Extracts a section of a string and returns it as a new string. | let arr = [1, 2, 3, 4]; arr.slice(1, 3); | [2, 3] |
| splice() | arr.splice(start, deleteCount, item1, ..., itemN) | let arr = [1, 2, 3]; arr.splice(1, 1, 4); | [1, 4, 3] |
| forEach() | arr.forEach(callback) Executes a provided function once for each array element. | let arr = [1, 2, 3]; arr.forEach(x => console.log(x)); | 1 2 3 (logged to console) |
| map() | arr.map(callback) Creates a new array with the results of calling a provided function on every element in this array. | let arr = [1, 2, 3]; arr.map(x => x \* 2); | [2, 4, 6] |
| filter() | arr.filter(callback) Creates a new array with all elements that pass the test implemented by the provided function. | let arr = [1, 2, 3]; arr.filter(x => x > 1); | [2, 3] |
| reduce() | arr.reduce(callback, initialValue) Applies a function against an accumulator and each element in the array to reduce it to a single value. | let arr = [1, 2, 3]; arr.reduce((a, b) => a + b, 0); | 6 |

**Date Functions**

| **Function** | Syntax Description | **Sample** | **Output** |
| --- | --- | --- | --- |
| getDate() | date.getDate() Returns the day of the month for the specified date according to local time. | let date = new Date(); date.getDate(); | 4 (if today is the 4th) |
| getDay() | date.getDay() Returns the day of the week for the specified date according to local time. | let date = new Date(); date.getDay(); | 5 (if today is Friday) |
| getFullYear() | date.getFullYear() Returns the year of the specified date according to local time. | let date = new Date(); date.getFullYear(); | 2025 |
| getMonth() | date.getMonth() Returns the month in the specified date according to local time. | let date = new Date(); date.getMonth(); | 3 (April, as months are 0-indexed) |
| getHours() | date.getHours() Returns the hour for the specified date according to local time. | let date = new Date(); date.getHours(); | 11 (if current hour is 11) |
| getMinutes() | date.getMinutes() Returns the minutes in the specified date according to local time. | let date = new Date(); date.getMinutes(); | 54 (if current minute is 54) |
| getSeconds() | date.getSeconds() Returns the seconds in the specified date according to local time. | let date = new Date(); date.getSeconds(); | 9 (if current second is 9) |
| setDate() | date.setDate(day) Sets the day of the month for a specified date according to local time. | let date = new Date(); date.setDate(15); | Date object set to 15th |
| setFullYear() | date.setFullYear(year) Sets the full year for a specified date according to local time. | let date = new Date(); date.setFullYear(2023); | Date object set to 2023 |
| setMonth() | date.setMonth(month) Sets the month for a specified date according to local time. | let date = new Date(); date.setMonth(5); | Date object set to June |

**Math Functions**

| **Function** | Syntax Description | **Sample** | **Output** |
| --- | --- | --- | --- |
| abs() | Math.abs(x) Returns the absolute value of a number. | Math.abs(-5) | 5 |
| ceil() | Math.ceil(x) Returns the smallest integer greater than or equal to a number. | Math.ceil(4.2) | 5 |
| floor() | Math.floor(x) Returns the largest integer less than or equal to a number. | Math.floor(4.8) | 4 |
| max() | Math.max(x1, x2, ..., xn) | Math.max(1, 2, 3) | 3 |
| min() | Math.min(x1, x2, ..., xn) | Math.min(1, 2, 3) | 1 |
| pow() | Math.pow(base, exponent) | Math.pow(2, 3) | 8 |
| random() | Math.random() | Math.random() | 0.123456789 (random number) |
| round() | Math.round(x) | Math.round(4.5) | 5 |
| sqrt() | Math.sqrt(x) | Math.sqrt(16) | 4 |
| trunc() | Math.trunc(x) | Math.trunc(4.9) | 4 |