1. **javascript string**

charAt, It access an individual character in a string

return 'cat'.charAt(1) // returns "a"

concat, It concatenates the strings into a new string.

let hello = 'Hello, '

console.log(hello.concat('Kevin', '. Have a nice day.'))

// Hello, Kevin. Have a nice day.

Includes ,It is case sensitive

const str = 'To be, or not to be, that is the question.'

console.log(str.includes('To be')) // true

console.log(str.includes('question')) // true

console.log(str.includes('nonexistent')) // false

console.log(str.includes('To be', 1)) // false

console.log(str.includes('TO BE')) // false

console.log(str.includes('')) // true

, endsWith, It determines whether or not a string ends with another string. It is case-sensitive.

let str = 'To be, or not to be, that is the question.'

console.log(str.endsWith('question.')) // true

console.log(str.endsWith('to be')) // false

console.log(str.endsWith('to be', 19)) // true

, indexOf

Characters in a string are indexed from left to right. The index of the first character is 0, and the index of the last character of a string called stringName is stringName.length - 1.

'Blue Whale'.indexOf('Blue') // returns 0

'Blue Whale'.indexOf('Blute') // returns -1

'Blue Whale'.indexOf('Whale', 0) // returns 5

'Blue Whale'.indexOf('Whale', 5) // returns 5

'Blue Whale'.indexOf('Whale', 7) // returns -1

'Blue Whale'.indexOf('') // returns 0

'Blue Whale'.indexOf('', 9) // returns 9

'Blue Whale'.indexOf('', 10) // returns 10

'Blue Whale'.indexOf('', 11) // returns 10

, lastIndexOf,

The **lastIndexOf()** method returns the index within the calling [String](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String) object of the last occurrence of the specified value, searching backwards from fromIndex. Returns -1 if the value is not found.

The following example

let anyString = 'Brave new world';

console.log('The index of the first w from the end is ' + anyString.lastIndexOf('w'));

// logs 10

console.log('The index of "new" from the end is ' + anyString.lastIndexOf('new'));

// logs 6

Replace It returns a new string with some or all matches of a pattern replaced by a replacement

example

let str = 'Twas the night before Xmas...';

let newstr = str.replace(/xmas/i, 'Christmas');

console.log(newstr); // Twas the night before Christmas...

,

slice, It extracts a section of a string and returns it as a new string, without modifying the original string.

examples

const str = 'The quick brown fox jumps over the lazy dog.';

console.log(str.slice(31));

// expected output: "the lazy dog."

console.log(str.slice(4, 19));

// expected output: "quick brown fox"

console.log(str.slice(-4));

// expected output: "dog."

console.log(str.slice(-9, -5));

// expected output: "lazy"

Split

It divides a [String](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String) into an ordered list of substrings, puts these substrings into an array, and returns the array

Ex:

const str = 'The quick brown fox jumps over the lazy dog.';

const words = str.split(' ');

console.log(words[3]);

// expected output: "fox"

, startsWith,

It determines whether a string begins with the characters of a specified string, returning true or false as appropriate.

Substr It extracts length characters from a str, counting from the start index.

, toLowercase

It is the value of the string converted to lower case.

, toUppercase,

It is the value of the string converted to uppercase

trim,

It removes whitespace from both ends of a string

trimStart,

It removes whitespace from the beginning of a string

trimEnd

It  removes whitespace from the end of a string.

2**) javascript Number**

 isNaN, It  determines whether the passed value is NaN and its type is number

parseFloat It   parses an argument and returns a floating point number

, parseInt

It  parses a string argument and returns an integer of the specified radix or base.

3) **javascript Math**

Abs, *The*Math*.abs() function returns the absolute value of a number*

ceil, *The*Math*.ceil() function always rounds a number up to the next largest integer.*

floor, *The*Math*.floor() function returns the largest integer less than or equal to a given number.*

min, *The static function*Math*.min() returns the lowest-valued number passed into it, or NaN if any parameter*

max, *The*Math*.max() function returns the largest of the zero or more numbers given as input parameters.*

random, *The*Math*.random() function returns a floating-point, pseudo-random number in the range 0 to less than*

round, *The*Math*.round() function returns the value of a number rounded to the nearest integer*

sqrt *The*Math*.sqrt() function returns the square root of a number,*

4)  **javascript array**

concat, The **concat()** method is used to merge two or more arrays. This method does not change the existing arrays, but instead returns a new array.

every The **every()** method tests whether all elements in the array pass the test implemented by the provided function. It returns a Boolean value.

filter

filter() calls a provided callback function once for each element in an array, and constructs a new array of all the values for which callback returns [a value that coerces to true](https://developer.mozilla.org/en-US/docs/Glossary/Truthy).

The find method executes the callback function once for each index of the array until the callback returns a [truthy](https://developer.mozilla.org/en-US/docs/Glossary/truthy) value. If so, find immediately returns the value of that element. Otherwise, find returns [undefined](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/undefined).

, findIndex,

forEach

, indexOf,

join,

map,

astaIndexOf,

pop,

push,

reduce

, reverse

, shift,

slice,

sort,

splice,

unshift,