Javascript Notes

->JavaScript was invented by Brendan Eich in 1995.
->It was developed for Netscape 2, and became the ECMA-262 standard in 1997.
->After Netscape handed JavaScript over to ECMA, the Mozilla foundation continued to develop JavaScript for the Firefox browser.
->JavaScript" was originally trademarked by Sun Microsystem, the company that developed Java.
->Later, Oracle aquired the company Sun Microsystem and hence, all trademarks owned by them were transfered to Oracle. Thus, currently, Oracle owns the trademark of JavaScript.

Types of JS
1)Internal JS [using script tag in head & body tag]
2)External JS [create .js file]

Javascript comments
// – single line
/*   */ - multiline

Output /Printing Any statements

- Writing into an HTML element, using `innerHTML`.
- Writing into the HTML output using `document.write()`.
- Writing into an alert box, using `window.alert()`.
- Writing into the browser console, using `console.log()`.

Declaring Variables

★ Var [use in older browser 1995 to 2015]
★ Let [after 2015]
★ Const
★ Nothing

What is difference b/w var & let?

→Once you fixed value of var,It will be redeclare & reinitialize.

Var a=”Hi”;

Var a;

→You can’t redeclare variable & value using let & const keyword.

Let a=”Hi”;

Let a;

==
JS Function

1)Normal function without arguments

```
//Take nothing[No arguments] Return nothing[No return type]

        function disp(){

            alert("Hiii........");

```

}

[redacted]

```
function name(){


}
name();
```

2)function with arguments

```
function name(a,b){


}
name(10,20);
name("hii","hello");
name();
```

3)Return function without arguments

```
function getNumber() {
    return 10;
};
```

4)Return function with arguments

```
function Sum(num1, num2) {
    return num1 + num2;
};
```

5)Nested function

```
function multiple(x) {

    function fn(y)
    {
        return x * y;
    }

    return fn;
}

var triple = multiple(3);
triple(2); // returns 6
triple(3); // returns 9
```

=================================

Function Expression

```
var add = function (num1, num2) {
    return num1 + num2;
};
```

```
var result = add(10, 20);//returns 30
```

Javascript String Methods

1)Length - count the length - str.length

2)slice() - str.slice(start,end)- position/index start with 0.ending -1

Ex.. str.slice(3/-3); str.slice(3,6); str.slice(-6,-3)

3)substring() - `substring()` is similar to `slice()`.

4)substr() - It is same as slice() but last argument is different. [start,length]

5)replace() - case sensitive, only one word match

Replace multiple words - `text.replace(/Rinkal/g, "Rinks");`

Insensitive - `text.replace(/RInkal/i, "Rinks");`

6)replaceAll() - Global & Insensitive - `text.replaceAll(/RInkal/gi, "Rinks");`

7)toUpperCase() - str.toUpperCase()

8)toLowerCase() - str.toLowerCase()

9)concat()

```
Ex:  let text1 = "Rinkal";
     let text2 = "Adesara";
     let text3 = text1.concat("@", text2); //Rinkal@Adesara
```

10)trim() - space remove

11)trimStart() - space remove from starting

12)trimEnd() - space remove from ending

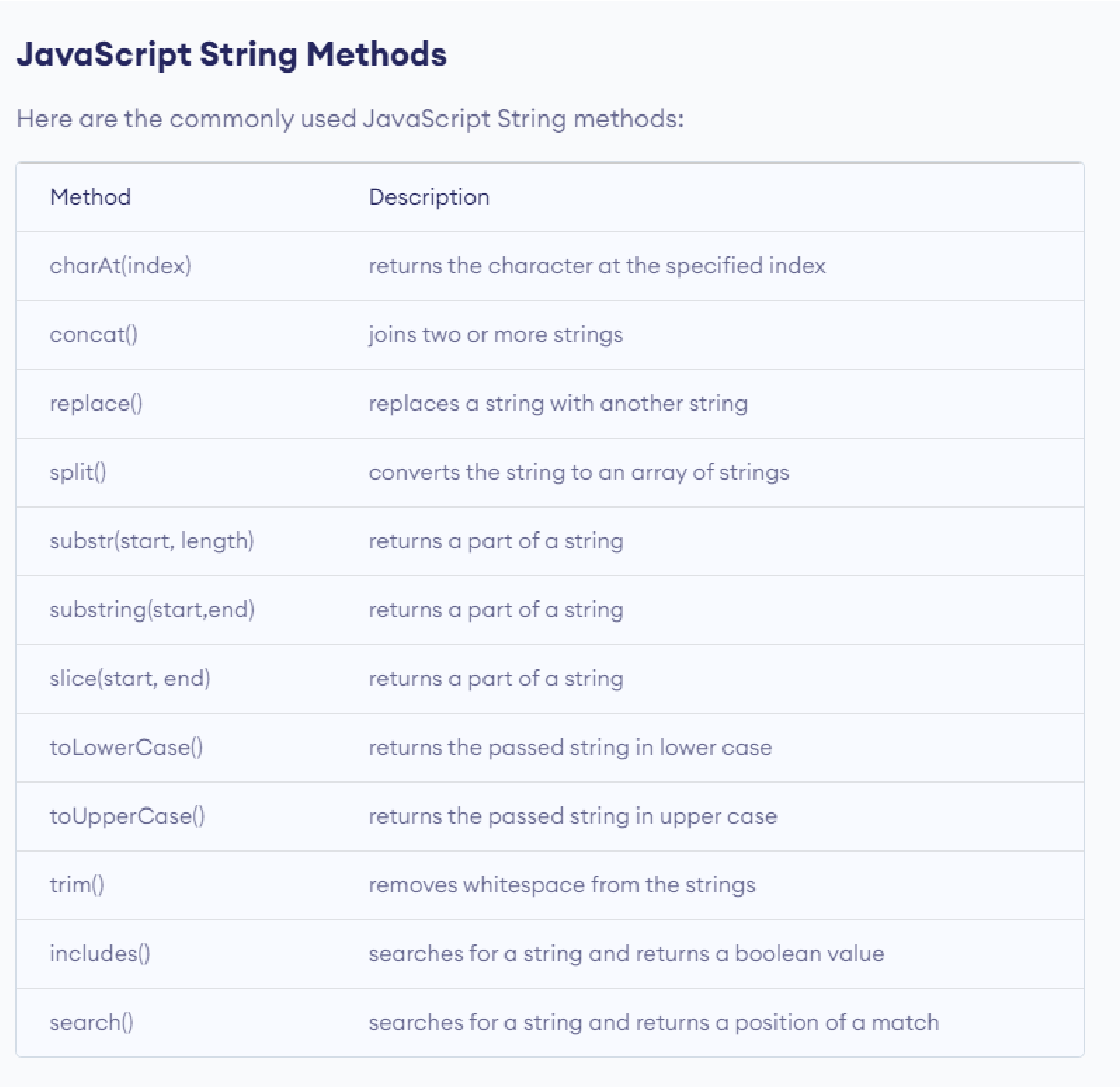
13)padStart() - append at starting

padStart(total length,append string or character)

14)padEnd() - append at ending

15)charAt() - position wise data return

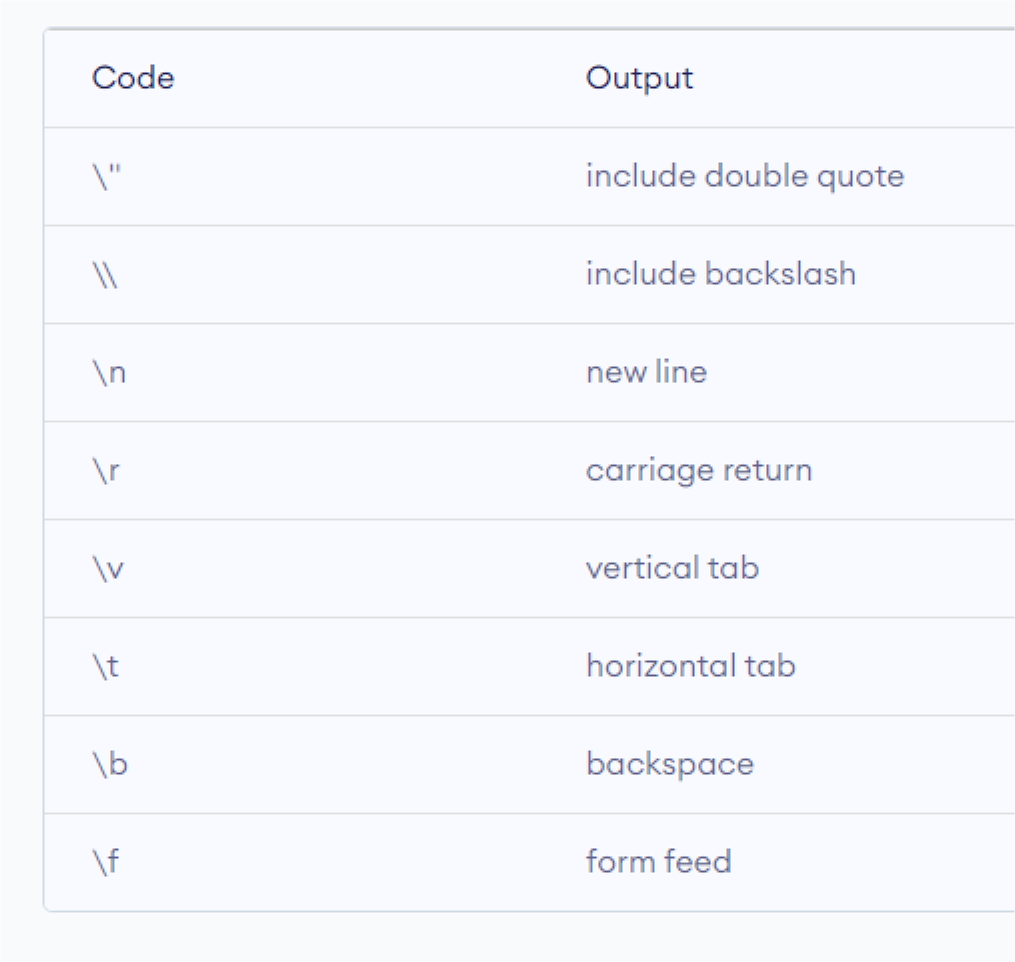
16)charCodeAt() - unicode value return



JavaScript String Methods

Here are the commonly used JavaScript String methods:

Method	Description
charAt(index)	returns the character at the specified index
concat()	joins two or more strings
replace()	replaces a string with another string
split()	converts the string to an array of strings
substr(start, length)	returns a part of a string
substring(start,end)	returns a part of a string
slice(start, end)	returns a part of a string
toLowerCase()	returns the passed string in lower case
toUpperCase()	returns the passed string in upper case
trim()	removes whitespace from the strings
includes()	searches for a string and returns a boolean value
search()	searches for a string and returns a position of a match



Code	Output
\"	include double quote
\\	include backslash
\n	new line
\r	carriage return
\v	vertical tab
\t	horizontal tab
\b	backspace
\f	form feed

Examples Of Functions

https://www.tutorialstonight.com/js/javascript-string-methods

String to Array Conversion

Let str = “Hi hello How are you?”;

Let str2 = str.split(“ “); [str2[0] = hi str2[1] = hello]

JS Array

let arrayName = [value1, value2, ...]; // Method 1

let arrayName = new Array(); // Method 2

Array Methods

1)toString() - Convert Array into String

By default it takes comma separate [string conversion]

    Let a =”hi,hello,how,are,you?”;

2)Push() - add one at last - a.push(“fg”)

3)Pop() - remove from last - a.pop()

4)length - count length of array - a.length

5)shift() - remove from first

6)unshift() - add from first

7)update element using index/position -

    a = [“hi”,”how”,”are”,”you”];

    a[2] = “Hello”;

8)delete array element using index/position

    Delete a[2];

9)Merge 2 or 3 or multiple array.

    concat()

    a = [“hi”,”how”,”are”,”you”]; b=[“hello”]; c = [“abc”,”xyz”];

    a.concat(b);   a.concat(b,c);

10)Splice - [position,removed,new element]

    Note: You can add & remove using splice function.

    splice(0,1) [0 means no add, 1 means remove first element]

11)Slice - create a new array. Does not change original array.

Let a = [“hi”,”how”,”are”,”you”];

Let b = a.slice(1); [”how”,”are”,”you”];

Let c = a.slice(1,3); [”how”,”are”]

ARRAY SORT

sort() - Ascending order

reverse() - Descending Order

Compare Function

```
function(a, b){return a - b} ascending

function(a, b){return b-a}
```

Array Iteration

1)**forEach()**

```
const numbers = [45, 4, 9, 16, 25];

let txt = "";

numbers.forEach(myFun);

function myFun(value) {

  txt += value + "<br>";

}
```

2)map() - create new array & does not change original array

3)flatMap() - first map & create new array
4)filter() - search from array

5)reduce() - 4 arguments (prev,next value get)

6)reduceRight - takes from reverse order

Example

```
const numbers = [45, 4, 9, 16, 25];

let sum = numbers.reduce(myFunction);

function myFunction(total, value) {

  return total + value;
}
```

7)indexOf - Find position of element in array

8)includes() - check value is exist or not from array [true,false]