JavaScript Basics

String Functions

Strings

- Sequence of characters with in quotes
- Quotes can be single or double quotes or backticks (`)

```
let single = 'single-quoted';
let double = "double-quoted";

let backticks = `backticks`; //Added in ES6
```

Backticks allow you to embed any expresssions including function calls using syntex ${ expression }

```
function sum(a, b) {
  return a + b;
}

alert(`1 + 2 = ${sum(1, 2)}.`); // 1 + 2 = 3.
```

String length

The length property has the string length:

```
console.log( `varma`.length ); // 5
```

Accessing charecters

To get a character at position **pos** use bracket notation or charAt method

```
let str = `Hello`;

// the first character
alert( str[0] ); // H
alert( str.charAt(0) ); // H

//How do you access last charecter
console.log(str[str.length-1]);
```

Strings are immutable

Strings can't be changed in JavaScript. It is impossible to change a character.

```
let str = 'Hi';

str[0] = 'h'; // error
```

Change case

Methods toLowerCase() and toUpperCase() change the case:

```
console.log( 'Varma'.toUpperCase() ); // VARMA
console.log( 'Varma'.toLowerCase() ); // varma
```

str.split([seperator, [limit]])

splits a string into an array of strings by separating the string into substrings

```
var str = 'hello I am john, I am a real human being';
var words = str.split(' '); // splits by every space
var lines = str.split(','); // splits by every ,
var allChar = str.split(''); // splits by every character
```

str.replace(regex, newstring)

replace() method returns a new string with some or all matches of a pattern replaced by a replacement

```
var str = 'Twas the night before Xmas...';
var newstr = str.replace(/xmas/i, 'Christmas');
console.log(newstr);  // Twas the night before Christmas...
```

Slice - Getting a substring

str.slice(start [, end])

Returns the part of the string from start to (but not including) end.

```
let str = "stringify";
alert( str.slice(0, 5) ); // 'strin'
alert( str.slice(0, 1) ); // 's'
```

If there is no second argument, then slice goes till the end of the stri

```
let str = "stringify";
alert( str.slice(2) ); // ringify, from the 2nd till the end
```

Negative values for start/end are also possible. They mean the position is counted from the string end:

```
let str = "stringify";

// start at the 4th position from the right, end at the 1st f
alert( str.slice(-4, -1) ); // gif
```

Searching for a substring

There are multiple ways to look for a substring within a string.

indexOf and **lastIndexOf** methods return position of searched string

```
let str = "Varma teaching JavaScript Basics and Advanced JavaScri

str.indexOf("JavaScript") //15
str.lastIndexOf("JavaScript") //46
```

includes, startsWith and **endsWith** methods are modern way to check existence of substring within.

It's the right choice if we need to test for the match, but don't need its position:

```
str.includes("JavaScript") //true
str.startsWith("JavaScript") //false
str.endsWith("JavaScript") //true
```

trim

removes (“trims”) spaces from the beginning and end of the string.

```
console.log("  varma  ".trim()) //varma
```