# Some JS concepts

By Prakash Shrestha

## 1. Concatenation

There are three ways to concatenate strings in JavaScript

a. The + Operator
b. String#concat()
c. Array#join()

# The + operator

```
let a = "hello"
let b = 5;
let c = 10;
let d = "11"
console.log(a+b)      //hello5
console.log(b+c)       //15
console.log(c+d)       //1011
console.log(a+d)       //hello11
```

# String#concat()

```
let a = "hello"

let b = 5;

const e = a.concat(' ola ',  'hi')

console.log(e);  //hello ola
```

# Array#join()

```
let f = ['Hello', ' ', 'World'].join(''); // 'Hello World'

console.log(f)
```

# 2. Given, a = 5; b = 6; c = “chaturdev”; sum = b - a;

```
var earn = 5;
var expense = 6;
var c = "Chaturdev";
var sum = expense-earn;

var isProfit = function(){
  if(expense < earn){
    var loss = expense -earn;
    return loss;
  }
 else{
    var profit = earn - expense;
    return profit;
  }
}

console.log(`${c} made a profit of
${earn} and a loss of ${expense} with
a loss of ${isProfit()}. We calculated
sum of ${sum} which is wrong.`)
```

output

```
Chaturdev made a profit of 5 and a loss of 6 with a loss of -1. We calculated sum of 1 which is wrong.
```

# 3. Reverse String

```
let str = "pallindrome"
let newString = "";
for (let i = str.length - 1; i >= 0; i--) {
   newString += str[i];
}

console.log(newString)     //emordnillap
```

# 4. (education) → no of consonants, no of variables

```
let str = "education";
var countConsonants = 0;
var variable = 0;

for (var i = 0; i < str.length;
i++) {

  if (str[i] == "a" || str[i] ==
"e" || str[i] == "i" ||
   str[i] == "o" || str[i] ==
"u") {
      variable++;
   }
   else{
     countConsonants ++;
  }
}
console.log( "no of
consonants in string is " +
countConsonants);
console.log("no of variable
in string " + variable);
```

Output

```
no of consonants in string
is 4

no of variable in string 5
```

# Return and Break

break is used to exit (escape) the for-loop, while-loop, switch-statement that you are currently executing.

return will exit the entire method you are currently executing (and possibly return a value to the caller, optional).

# Scope

```
var a = 1;
let b = 2;
const c = 3;

function sum(){
  var a = 10;
   b = 20;
   // c = 30;   //not possible

   console.log(a, b, c);
}
var a = 5;
// let b = 6;   // not possible
console.log(a, b)

sum();
```

## 5. Array mapping

map() creates a new array from calling a function for every array element.

map() calls a function once for each element in an array.

map() does not execute the function for empty elements.

map() does not change the original array.

# 6. Array Function

```
let n = 5;
let string = "";
// External loop
for (let i = 1; i <= 5; i++) {

 for (let j = 1; j <= 5 - i; j++) {
  string += " ";
 }
 // printing star
 for (let k = 0; k < 2 * i - 1; k++) {
  string += "*";
 }
 string += "\n";
}
console.log(string);
```

# 7. Array Check

[2, 17, 6, 3, 12, 9, 8]

a. Ascending
b. Odd
c. Even

# a. Ascending

```
let myarray = [2, 17, 6, 3, 12, 9, 8];
console.log(
   myarray.sort(
      function (a, b) {
         return a - b;
      }
   )
)
```

# B. odd

```
console.log("even numbers are:")
myarray.forEach((el) =>{
   if(el%2 == 0){
      console.log(el)
   }
})
```

# C. Even

```
console.log("odd numbers are:")
myarray.forEach((el) =>{
  if(el%2 != 0){
    console.log(el)
  }
})
```

# 9. Null and Undefined

a. Undefined means a variable has been declared but has not yet been assigned a value
b. Null is an assignment value. It can be assigned to a variable as a representation of no value

```
var testVar;

alert(testVar); //shows undefined

alert(typeof testVar); //shows undefined
```

```
var testVar = null;

alert(testVar); //shows null

alert(typeof testVar); //shows object
```