ES6:

ES6 or ECMAScript 6 is a scripting language specification which is standardized by ECMAScript International. This specification governs some languages such as [JavaScript](https://www.javatpoint.com/javascript-tutorial), ActionScript, and Jscript. ECMAScript is generally used for client-side scripting, and it is also used for writing server applications and services by using [Node.js](https://www.javatpoint.com/nodejs-tutorial).

ES6 allows you to write the code in such a way that makes your code more modern and readable. By using ES6 features, we write less and do more, so the term 'Write less, do more' suits ES6.

ES6 or ECMA6(European Computer Manufacturers Association) is a standard specification for a scripting language.

To work with ES6 you need to install VS code editor and also node.js – which is a platform that facilitates write the scripting code. Along with node.js you get npm (Node Package Manager) which helps you install other required softwares/modules.

Like any other language, ES6 is a Javascript scripting standardised language which also contains Variables, constants, loops, if-else conditions, arrays, classes etc.

The ES6 files are saved as .js files and are executed as

node filename

The below are the features of ES6: [features means we learn about constants, variables, operators, arrays etc]

1. Spread operator: newly introduced operator in ES6 and is denoted by 3 dots. […]

let colors = ['Red', 'Yellow'];

let newColors = [...colors, 'Violet', 'Orange', 'Green'];

console.log(newColors);

1. Rest Parameter: newly introduced as part of ES6, The rest parameter allows us to represent an indefinite number of arguments as an array. By using the rest parameter, a function can be called with any number of arguments. Denoted by […]

function show(...args) {

  let sum = 0;

  for (let i of args) {

      sum += i;

  }

  console.log("Sum = "+sum);

}

show(10, 20, 30);

1. Variables:

var x=100;

var y=200;

console.log(x+"  "+y);

var $var12 = 200;

function example() {

    var $var12 = 300;

    console.log("Inside example() function = "

               + $var12);

}

console.log("Outside example() function = "

               + $var12);

example();

1. Operators: Arithmetic, Conditional, Logical, bitwise all are available.

var x = 30;

var y = 80;

console.log("Value of x = " + x );

console.log("Value of y = " + y );

var result = ((x < 40) && (y <= 90));

console.log("(x < 40) && (y <= 90): ", result);

var result = ((x == 50) || (y > 80));

console.log("(x == 50) || (y > 80): ", result);

var result = !((x > 20) && (y >= 80));

console.log("!((x > 20) && (y >= 80)): ", result);

1. Loops: for, while, do-while as usual.

var fruits = ['Apple', 'Banana', 'Mango', 'Orange'];

for(let value of fruits)

{

  console.log(value);

}

var n = 1;

while(n<=7)

{

   console.log("n="+n);

   if(n==4)

   {

       break;

   }

   n++;

}

6)

Decision Making:

var x = 40, y=20;

if (x < y)

{

   console.log("y is greater");

}

else

{

   console.log("x is greater");

}

var num = 5;

switch(num) {

    case 0 : {

      console.log("Sunday");

      break;

   }

   case 1 : {

    console.log("Monday");

    break;

 }

   case 2 : {

    console.log("Tuesday");

    break;

 }

default:{

  console.log("Holiday");

    break;

 }

1. Functions, default functions, return, Anonymous functions:

function add( n1,n2) {

var sum = n1 + n2

console.log("The sum of the values entered "+sum)

}

add(12,13)

function show (num1, num2=200)

{

    console.log("num1 = " +num1);

    console.log("num2 = " +num2);

}

show(100);

function show(a, ...args)

{

    console.log(a + " " + args);

}

show(50,60,70,80,90,100);

1. Arrow function: Similar to Lambda functions.

// function expression

var myfun1 = function show() {

 console.log("It is a Function Expression");

}

// Anonymous function

var myfun2 = function () {

    console.log("It is an Anonymous Function");

   }

//Arrow function

var myfun3 = () => {

    console.log("It is an Arrow Function");

   };

myfun1();

myfun2();

myfun3();

1. Arrays:

var num;

num = [2,4,6,8];

console.log(num[0]);

console.log(num[1]);

var num = new Array(1,2,3,4,5);

var i;

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

console.log(num[i]);

}

var rainbow = new Array["Violet", "Indigo", "Blue", "Green", "Yellow", "Orange", "Red"];

function show(rainbow) {

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

      console.log(rainbow[i])

   }

}

show(rainbow)

Activity: Study several in built methods of Arrays when time permits.

10)

ES6 Objects:

Object in ES6 can be created in 2 ways:

By using object literal and By using object constructor

var user =  {};  // 'object literal' syntax

var name = new Object();  //'object constructor' syntax

An object can be created by using curly braces **{...}** along with an optional properties list. The property is a **"key:value"** pair, where the key is a string or a property name, and the value can be anything.

let fname = 'Anil',

    lname = 'kumar';

let user = {

   fname,

   lname

};

console.log(user.fname);

console.log(user.lname);

var department = 'dep\_name';

var emp = {

    id : 102,

    name : 'Anil',

    [department]:'Production'

}

console.log(emp);

11)

ES6 Classes:

class Student {

   constructor(name, age) {

      this.n = name;

      this.a = age;

   }

   stu() {

      console.log("The Name of the student is: ", this.n)

      console.log("The Age of the student is: ",this. a)

   }

}

var stuObj = new Student('Peter',20);

stuObj.stu();

12)

String: Same rules apply as like Java here also!

let str = "hello world"

console.log(str.includes('world',5));

console.log(str.includes('World', 11))