Next Gen JS

Study Guide

Author
Srinivas Dande

ES5/ES6 New Features

1) arguments object (ES5)
2) Rest parameter (...) (ES6)
3) JSON conversions (ES5)

    JSON.parse()
    JSON.stringify()

4) for…of (ES6)
5) Template literals `` (ES6)
6) Arrow Functions (ES6)
7) Default function parameters (ES6)
8) let and const (ES6)
9) Array extensions (ES5)

    Array.forEach()
    Array.map()
    Array.filter()
    Array.reduce()
    Array.reduceRight()
    Array.indexOf()
    Array.lastIndexOf()

10) Array extensions (ES6)
    find()
    findIndex()

11) String extensions (ES5)
    String.trim()

12) String extensions (ES6)
    startsWith()
    endsWith()
    includes()

13) Date extensions (ES5)
    Date.now()

14) Spread operator (...) (ES6)

15) DE-Structuring Assignment : (ES6)

16) Classes and Objects (ES6)

17) Inheritance (ES6)

18) Modules (ES6)

demo1.html

```
<!DOCTYPE html>
<html lang="en">
<body>
<h3> arguments object</h3>
<script>
function show() {
console.log("---show()---");
console.log("Length : ", arguments.length);
for (let myvalue of arguments) {
console.log(myvalue);
}
}

show();
show(10);
show(10, 20);
show(10, 20, 30);

function show2() {
console.log("---show2()---");
console.log("Length : ", arguments.length);
}

function show1() {
console.log("---show1()---");
console.log("Length : ", arguments.length);
show2(arguments)  //[10,20]
}
```

```
show1(10, 20);

console.log("Done!!!");
</script>
</body>
</html>
```

demo2.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Rest Operator</h3>
  <script>

    function show(...myargs) {
      console.log("---show()---");
      console.log("Length : ", myargs.length);
      for (let myvalue of myargs) {
        console.log(myvalue);
      }
    }

    show();
    show(10);
    show(10, 20);
    show(10, 20, 30);
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo3.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> JSON.parse() Demo </h3>
```

```
  <script>
    let mydata = '{"courseId":101, "courseName":"Java", "trainer":"Srinivas"}';
    console.log(typeof (mydata)); //string

    let mycourse = JSON.parse(mydata);
    console.log(typeof (mycourse)); //object

    console.log(mycourse.courseId);
    console.log(mycourse.courseName);
    console.log(mycourse.trainer);

    //let x= "Hello Guys";
    //let y = JSON.parse(x);
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo4.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> JSON.stringify Demo </h3>
  <script>

    let mystudent = {
      sname: "Srinivas",
      email: "Sri@jlc.com",
      phone: 12345,
      courseId: 101,
    };
    console.log(typeof (mystudent));

    var mystring = JSON.stringify(mystudent);
     console.log(typeof (mystring));
    console.log(mystring);
```

```
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo5.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Ways to Access Arrays </h3>

  <script>
    let arr = ["Angular", "NodeJS", "ReactJS", "ExpressJS"];

    console.log(arr);
    console.log("------Regular for---------");
    for (let i = 0; i < arr.length; i++) {
      console.log(arr[i]);
    }

    console.log("-----for in----------");
    for (let index in arr) {
      console.log(arr[index]);
    }

    console.log("-----for of----------");
    for (let value of arr) {
      console.log(value);
    }


    console.log("------forEach---------");

    function showArray1(myvalue, myindex) {
      console.log(myindex + "      " + myvalue);
    }
    arr.forEach(showArray1); //Higher-Order Functions
```

```
    console.log("----------------");

    function showArray2(myvalue) {
      console.log(myvalue);
    }
    arr.forEach(showArray2); //Higher-Order Functions
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo6.html

```
  <!DOCTYPE html>
<html lang="en">
<body>
  <h3> Arrow Functions Demo </h3>
  <script>
    show1 = () => {
      console.log("I am show 1");
    }
    const show2 = (x) => {
      console.log("I am show 2", x);
    }
    const findSum = (x, y) => {
      return x + y;
    }

    show1();
    show2(55);

    let sum = findSum(100, 200);
    console.log("Sum : ", sum)

    const showStudent = (sname, email, phone) => {
      //let msg="Name : "+sname +", Email : "+email +", Phone : "+phone;
      let msg = `Name is ${sname} , Email is ${email} and Phone is ${phone}`;
      console.log(msg);
    }

    showStudent("Sri", "Sri@jlc", 12345);
```

```
    const showArray = (arr) => {
      for (let value of arr) {
        console.log(value);
      }
    }

    let myarr = [10, 20, 30, 40, 50];
    showArray(myarr)

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo7.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Default Parameters </h3>
  <script>
    function findSum(a, b = 10) {
      return a + b;
    }

    let result = findSum(10);
    console.log("Sum :  ", result);

    result = findSum(10, 20);
    console.log("Sum : ", result);

    const sum = (a, b = 10) => {
      return a + b;
    }

    result = sum(10);
    console.log("Sum : ", result);
```

```
    result = sum(10, 20);
    console.log("Sum : ", result);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo8.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> var and let </h3>
  <script>
    console.log("Variable Scope")
    var a = 10;
    console.log(a);
    var a = "Hello Guys";
    console.log(a);

    let b = 20;
    console.log(b);
    // let b = "Hai Guys";
    b = "Hai Guys";
    console.log(b);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo9.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> var and let </h3>
```

```
  <script>
    console.log("var and let ")

    var x = 10;

    function show() { //Function Definition
      var y = 20;
      if (true) {
        var z = 30;
        console.log("Inside If- x = " + x);
        console.log("Inside If- y = " + y);
        console.log("Inside If- z = " + z);
      }
      console.log("After If- x = " + x);
      console.log("After If- y = " + y);
      console.log("After If- z = " + z);
    }
    show(); //Function Call
    console.log("Outside show()- x = " + x);
    //console.log("Outside show()- y = "+y);
    //console.log("Outside show()- z = "+z);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo10.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> let and const </h3>

  <script>
    console.log("let and const")
```

```
    let x = 10;
    console.log(x);
    x = 20;
    console.log(x);

    const y = 10;
    console.log(y);
    // y = 20;
    // console.log(y);
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo11.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Primities and References</h3>
  <script>
    console.log("Primities and References");

    let x = 10;
    let y = x;
    x = 55;
    console.log(x);
    console.log(y);

    let arr1 = [1, 2, 3];
    let arr2 = arr1;
    arr1.push(5);
    console.log(arr1);
    console.log(arr2);

    const arr3 = [1, 2, 3];
    const arr4 = arr3;
    arr3.push(55);
    arr4.push(99);
```

```
    console.log(arr3);
    console.log(arr4);
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo12.html

```
<!DOCTYPE html>
<html lang="en">
<body>
 <h3> Array – Extensions - 1</h3>
 <script>
  function get(x) {
   return x;
  }
  function square(x) {
   return x * x;
  }
  function cube(x) {
   return x * x * x;
  }
  function cubeString(x) {
   let cube = x * x * x;
   return `Cube of ${x} => ${cube}`;
  }

  let numArray = [1, 2, 3, 4, 5];
  console.log(numArray);

  let outputArr = numArray.map(get);
  console.log(outputArr)

  outputArr = numArray.map(square);
  console.log(outputArr)

  outputArr = numArray.map(cube);
  console.log(outputArr)
```

```
  outputArr = numArray.map(cubeString);
  console.log(outputArr)

  console.log("-----------------------------");
  outputArr = numArray.map(x => x);
  console.log(outputArr)

  outputArr = numArray.map(x => x * x);
  console.log(outputArr)

  outputArr = numArray.map(x => x * x * x);
  console.log(outputArr)

  outputArr = numArray.map(d => `Cube of ${d} => ${d*d*d}`);
  console.log(outputArr)

  console.log("-----------------------------");

  console.log("Done!!!");
 </script>
</body>
</html>
```

demo13.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Array – Extensions - 2</h3>
  <script>
    let numArray = [1, 2, 3, 4, 5, 6, 7];
    console.log(numArray);

    function findEven(num) {
      if (num % 2 == 0)
        return true;
      else
        return false;
```

```
    }

    function findOdd(num) {
      if (num % 2 != 0)
        return true;
      else
        return false;
    }

    let resultArray = numArray.map(findEven);
    console.log(resultArray)

    resultArray = numArray.filter(findEven);
    console.log(resultArray)
    console.log("-----------------------");

    resultArray = numArray.map(findOdd);
    console.log(resultArray)

    resultArray = numArray.filter(findOdd);
    console.log(resultArray)
    console.log("-----------------------");

    resultArray = numArray.map(x => x % 2 == 0);
    console.log(resultArray)
    resultArray = numArray.filter(x => x % 2 == 0);
    console.log(resultArray)

    console.log("-----------------------");
    resultArray = numArray.map(x => x % 2 != 0);
    console.log(resultArray)
    resultArray = numArray.filter(x => x % 2 != 0);
    console.log(resultArray)

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo14.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Array - Extensions</h3>

  <script>
    let numArray = [1, 2, 3, 4, 5];
    console.log(numArray);

    const sumNumbers = (result, value) => {
      return result + value;
    }
    const mulNumbers = (result, value) => {
      return result * value;
    }
    let result = numArray.reduce(sumNumbers);
    console.log(result);
    result = numArray.reduce(mulNumbers);
    console.log(result);
    console.log("-----------------------------------");
    result = numArray.reduce(sumNumbers, 10);
    console.log(result);
    result = numArray.reduce(mulNumbers, 10);
    console.log(result);
    console.log("-----------------------------------");
    result = numArray.reduceRight(sumNumbers);
    console.log(result);
    result = numArray.reduceRight(mulNumbers);
    console.log(result);
    console.log("-----------------------------------");
    result = numArray.reduceRight(sumNumbers, 10);
    console.log(result);
    result = numArray.reduceRight(mulNumbers, 10);
    console.log(result);
    console.log("-----------------------------------");
    console.log("Done!!!");
```

```
  </script>
</body>
</html>
```

demo15.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Array - Extensions</h3>
  <script>
    let numArray = [1, 2, 3, 4, 5];
    console.log(numArray);

    result = numArray.reduce((x, y) => x + y);
    console.log(result);
    result = numArray.reduce((x, y) => x * y);
    console.log(result);
    console.log("-----------------------------------");
    result = numArray.reduce((x, y) => x + y, 10);
    console.log(result);
    result = numArray.reduce((x, y) => x * y, 10);
    console.log(result);
    console.log("-----------------------------------");
    result = numArray.reduceRight((x, y) => x + y, 0);
    console.log(result);
    result = numArray.reduceRight((x, y) => x * y, 1);
    console.log(result);
    console.log("-----------------------------------");
    result = numArray.reduce((x, y) => (x > y) ? x : y, 1);
    console.log(result);
    result = numArray.reduce((x, y) => (x < y) ? x : y, 1);
    console.log(result);
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo16.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Array - Extensions</h3>

  <script>
    let myarray = [10, 20, 30, 20, 30, 40, 50];

    let myindex = myarray.indexOf(20); //1
    console.log(myindex)
    myindex = myarray.indexOf(30);//2
    console.log(myindex)
    myindex = myarray.lastIndexOf(20); //3
    console.log(myindex)
    myindex = myarray.lastIndexOf(30); //4
    console.log(myindex)
    myindex = myarray.indexOf(50); //6
    console.log(myindex)
    myindex = myarray.lastIndexOf(50); //6
    console.log(myindex)
    myindex = myarray.indexOf(99);
    console.log(myindex)
    myindex = myarray.lastIndexOf(99);
    console.log(myindex)
    console.log("---------------");

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo17.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Array - Extensions</h3>

  <script>
    let myarray = [10, 20, 30, 20, 30, 40, 50];

    //try these numbers one by one
    let searchNum = 20;
    // let searchNum = 30;
    //let searchNum = 99;

    const searchMyNumber = (element) => {
      if (element == searchNum) {
        return true;
      } else {
        return false;
      }
    }

    let result = myarray.find(searchMyNumber);
    console.log(result)

    let myindex = myarray.findIndex(searchMyNumber);
    console.log(myindex)

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo18.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Array - Extensions</h3>

  <script>
    let myarray = [10, 20, 30, 20, 30, 40, 50];

    //try these numbers one by one
    let searchNum = 20;
    // let searchNum = 30;
    //let searchNum = 99;

    result = myarray.find((element) => {

      if (element == searchNum) {
        return true;
      } else {
        return false;
      }
    });

    console.log(result)

    myindex = myarray.findIndex((element) => {
      if (element == searchNum) {
        return true;
      } else {
        return false;
      }
    });
    console.log(myindex)
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo19.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Array - Extensions</h3>

  <script>
    let myarray = [10, 20, 30, 20, 30, 40, 50];

    //try these numbers one by one
    let searchNum = 20;
    // let searchNum = 30;
    //let searchNum = 99;

    result = myarray.find((element) => element == searchNum);
    console.log(result)

    myindex = myarray.findIndex((element) => element == searchNum);
    console.log(myindex)

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo20.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> String Extensions </h3>

  <script>
    let mystr = "   Srinivas  Dande   ";
    console.log(mystr);
    console.log(mystr.length);
```

```
    let mystr1 = mystr.trim();
    console.log(mystr);
    console.log(mystr.length);
    console.log(mystr1);
    console.log(mystr1.length);

    console.log("---------------");
    let str = "Hello JLC Guys";

    console.log(str.startsWith("Hello"))
    console.log(str.startsWith("Guys"))
    console.log(str.endsWith("Hello"))
    console.log(str.endsWith("Guys"))
    console.log("------------------");

    console.log(str.includes("Hello"));
    console.log(str.includes("JLC"));
    console.log(str.includes("Guys"));
    console.log(str.includes("Hai"));

    console.log("---------------");
    console.log(Date.now());
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo21.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Spread Operator -1 </h3>

  <script>
    let arr = [1, 2, 3];
    let arr1 = [arr, 4, 5];
    console.log(arr1);
```

```
    let arr2 = [...arr, 4, 5]; //[1,2,3,4,5]
    console.log(arr2);

    let arr3 = [4, ...arr, 5]; //4,1,2,3,5
    console.log(arr3);

    let arr4 = [4, 5, ...arr]; //4,5,1,2,3
    console.log(arr4);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo22.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Spread Operator -2 </h3>
  <script>
    const stu = {
      sid: 101,
      sname: "Sri",
      email: 'sri@jlc',
    };

    const mystu1 = {
      stu,
      phone: 12345,
      city: "Blore",
    }

    console.log(stu);
    console.log(mystu1);
```

```
    console.log(mystu1.stu.sid);
    console.log(mystu1.stu.sname);
    console.log(mystu1.stu.email);
    console.log(mystu1.phone);
    console.log(mystu1.city);

    const mystu2 = {
      ...stu,
      phone: 12345,
      city: "Blore",
    }

    console.log(mystu2);
    console.log(mystu2.sid);
    console.log(mystu2.sname);
    console.log(mystu2.email);
    console.log(mystu2.phone);
    console.log(mystu2.city);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo23.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Spread Operator- 3</h3>

  <script>
    // Creating Copy of Object
    const mystu = {
      sid: 101,
      sname: "Sri",
      email: 'sri@jlc',
      status: true,
      courses: ['Java', "Angular", "React JS"],
```

```
    address: {
      city: "Blore",
      state: "KA"
    }
  };

  let mystu1 = {
    ...mystu
  };

  console.log(mystu);
  console.log(mystu1);
  console.log(mystu == mystu1);
  console.log(mystu === mystu1);

  mystu.sname = "Srinivas";
  mystu.email = "srinivas@jlc.com";

  console.log(mystu);
  console.log(mystu1);
  console.log(mystu == mystu1);
  console.log(mystu === mystu1);

  /*
  let mystu2 = mystu;
  console.log(mystu == mystu2);
  console.log(mystu === mystu2);
  */
  console.log("Done!!!");
 </script>
</body>
</html>
```

demo24.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Destructuring Demo -1</h3>

  <script>
    const myarr1 = [55, 99];
    const [a, b] = myarr1;
    console.log(a);
    console.log(b);

    const mycourses = ['Java Script', 'Angular', 'React'];
    const [C1, C2, C3] = mycourses;
    console.log(C1);
    console.log(C2);
    console.log(C3);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo25.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Destructuring Demo -2</h3>

  <script>
    let myarr1 = [55, 66, 77, 88, 99];
    const [x, y, ...remArr] = myarr1;
    console.log("x= ", x);
    console.log("y= ", y);
    console.log("remArr= ", remArr);
```

```
    let myarr2 = [55, 66, 77, 88, 99];
    const [p, q, r] = myarr2;
    console.log("p= ", p);
    console.log("q= ", q);
    console.log("r= ", r);

    /*
    let myarr3 = [55, 66, 77, 88, 99];
    const [...a, b, c] = myarr3;
    console.log("a= ", a);
    console.log("b= ", b);
    console.log("c= ", c);
    */
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo26.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Destructuring Demo -3 </h3>

  <script>
    let a, b;

    [a, b] = [55];
    console.log(a);
    console.log(b);

    [a = 10, b = 20] = [];
    console.log(a);
    console.log(b);

    [a = 10, b = 20] = [55];
    console.log(a);
    console.log(b);
```

```
    [a = 10, b = 20] = [55, 99];
    console.log(a);
    console.log(b);

    let x = 10;
    let y = 20;
    console.log(x);
    console.log(y);

    [x, y] = [y, x];
    console.log(x);
    console.log(y);

    const arr = [10, 20, 30];
    console.log(arr);
    [arr[2], arr[1]] = [arr[1], arr[2]];
    console.log(arr);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo27.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Destructuring Demo - 4 </h3>

  <script>
    const mystu = {
      sid: 101,
      sname: "Sri",
      email: 'sri@jlc',
      phone: 12345,
    };
```

```
    console.log(mystu.sid);
    console.log(mystu.sname);
    console.log(mystu.email);
    console.log(mystu.phone);

    mysid = mystu.sid;
    mysname = mystu.sname;
    myemail = mystu.email;
    myphone = mystu.phone;

    console.log(mysid);
    console.log(mysname);
    console.log(myemail);
    console.log(myphone);

    const {  sid, sname,  email, phone  } = mystu;

    console.log(sid);
    console.log(sname);
    console.log(email);
    console.log(phone);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo28.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Destructuring Demo - 5</h3>

  <script>
    const mystu = {
      sid: 101,
      sname: "Sri",
      contactInfo: {
        email: 'sri@jlc',
        phone: 12345,
      },
      address: {
        city: "Blore",
        state: "KA"
      },
      courses: [{
          cid: "C-501",
          cname: "Java Script",
          trainer: "Srinivas"
        },
        {
          cid: "C-502",
          cname: "Angular",
          trainer: "Srinivas"
        },
        {
          cid: "C-503",
          cname: "React",
          trainer: "Srinivas"
        },
      ]
    };
```

```
    const {  sid,  sname  } = mystu;
    console.log(sid);
    console.log(sname);

    console.log("--------------");
    console.log(mystu.contactInfo.email);
    console.log(mystu.contactInfo.phone);
    console.log("--------------");

    const {  email,  phone } = mystu;
    console.log(email);
    console.log(phone);
    console.log("--------------");

    const {  email,  phone } = mystu.contactInfo
    console.log(email);
    console.log(phone);
    console.log("--------------");

    const {  contactInfo  } = mystu;
    console.log(contactInfo.email);
    console.log(contactInfo.phone);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo29.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Destructuring Demo - 6</h3>

  <script>
    const mystu = {
      sid: 101,
      sname: "Sri",
      contactInfo: {
        email: 'sri@jlc',
        phone: 12345,
      },
      address: {
        city: "Blore",
        state: "KA"
      },
      courses: [{
          cid: "C-501",
          cname: "Java Script",
          trainer: "Srinivas"
        },
        {
          cid: "C-502",
          cname: "Angular",
          trainer: "Srinivas"
        },
        {
          cid: "C-503",
          cname: "React",
          trainer: "Srinivas"
        },
      ]
    };
```

```
    console.log("----1--------");
    console.log(mystu.courses);

    const [myjs, myangular, myreact] = mystu.courses;
    console.log(1, myjs);
    console.log(2, myangular);
    console.log(3, myreact);

    console.log("----2--------");
    console.log(4, myjs.cid);
    console.log(5, myjs.cname);
    console.log(6, myjs.trainer);

    console.log("--------3-------");
    console.log(myjs.cid);
    console.log(mystu.courses);
    myjs.cid = 9001;
    console.log(myjs.cid);
    console.log(mystu.courses);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo30.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Classes and Objects -1 </h3>
  <script>
    class Customer {
      //Instance Variables
      cid = 101;
      cname = "Sri";
      email = "sri@jlc";

      //constructor
      constructor() {
        console.log(" I am 0-arg Con ");
      }

      //Instance Method
      showCustomer = () => {
        console.log(" showCustomer ");
        let sid=111111;
        console.log(this.cid);
        console.log(this.cname);
        console.log(this.email);
      }
     }

    let c1 = new Customer();
    c1.showCustomer();
    let c2 = new Customer();
    c2.showCustomer();
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo31.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Classes and Objects -2 </h3>
  <script>
    class Student {
      /*
      constructor() {
        console.log(" I am D.C ");
      }
      */
      constructor(sid, sname, email) {
        this.sid = sid;
        this.sname = sname;
        this.email = email;
      }

      showStudent = () => {
        console.log(" showStudent ");
        console.log(this.sid);
        console.log(this.sname);
        console.log(this.email);
      }
    }

    let stu1 = new Student(101, "Sri", "sri@jlc");
    stu1.showStudent();

    console.log(stu1.sid);
    console.log(stu1.sname);
    console.log(stu1.email);
    console.log("------------------");

    let stu2 = new Student(102, "Vas", "vas@jlc");
    stu2.showStudent();
```

```
    console.log(stu2.sid);
    console.log(stu2.sname);
    console.log(stu2.email);

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo32.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Classes and Objects -3 </h3>

  <script>
    class Customer {

      constructor() {
        console.log(" I am D.C ");
        this.cid = 0;
        this.cname = null;
        this.phone = 0;
      }

      showCustomer = () => {
        console.log(" showCustomer ");
        console.log(this.cid);
        console.log(this.cname);
        console.log(this.phone);
      }

      get customerId() {
        return this.cid;
      }
```

```
    get customerName() {
      return this.cname;
    }
    get customerPhone() {
      return this.phone;
    }
    set customerId(cid) {
      this.cid = cid;
    }
    set customerName(cname) {
      this.cname = cname;
    }
    set customerPhone(phone) {
      this.phone = phone;
    }
}

let mycust = new Customer();
mycust.showCustomer();
console.log("---way to access Propertues---");
console.log(mycust.cid);
console.log(mycust.cname);
console.log(mycust.phone);

console.log("---way to access getters---");
console.log(mycust.customerId);
console.log(mycust.customerName);
console.log(mycust.customerPhone);

console.log("---way to access setters---");
mycust.customerId = 101;
mycust.customerName = "Srinivas";
mycust, customerName = 12345;

console.log("---way to access getters---");
console.log(mycust.customerId);
console.log(mycust.customerName);
console.log(mycust.customerPhone);
```

```
    console.log("---calling showCustomer()---");
    mycust.showCustomer();
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo33.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Classes and Objects -4</h3>

  <script>
    class Hello {
      a = 10;
      static b = 20;
      constructor() {
        console.log(" I am D.C ");
        this.c = 30;
      }

      showHello = () => {
        console.log(" showHello ");
        console.log("a = ", this.a);
        console.log("b = ", this.b);
        console.log("b = ", Hello.b);
        console.log("c = ", this.c);
      }
      static myshow() {
        console.log("i am static method- myshow()");
        console.log("a = ", this.a);
        console.log("b = ", this.b);
        console.log("b = ", Hello.b);
        console.log("c = ", this.c);
      }

    }
```

```
    let hello = new Hello();

    console.log("---calling showHello()---");
    hello.showHello();

    console.log("---way to access Properties---");
    console.log("a = ", hello.a);
    console.log("b = ", hello.b);
    console.log("b = ", Hello.b);
    console.log("c = ", hello.c);
    console.log("---calling static methods()---");
    // hello.myshow();
    Hello.myshow();
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo34.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Classes and Objects -5</h3>
  <script>
    class Customer {
      constructor(sid, sname) {
        console.log(" Customer -2 arg ");

        this.sid = sid;
        this.sname = sname;
      }
      showCustomer = () => {
        console.log(" showCustomer ");
        console.log(this.sid);
        console.log(this.sname);
      }
    }
```

```
    class GoldCustomer extends Customer {
      constructor(sid, sname, email, phone) {
        console.log(" GoldCustomer -4 arg ");
        super(sid, sname);
        this.email = email;
        this.phone = phone;
      }

      showGoldCustomer = () => {
        console.log(" showGoldCustomer ");
        console.log(this.sid);
        console.log(this.sname);
        console.log(this.email);
        console.log(this.phone);
      }
    }

    let mycust = new GoldCustomer(101, "Sri", "sri@jlc", 12345);
    mycust.showCustomer();
    mycust.showGoldCustomer();

    console.log("Done!!!");
  </script>
</body>
</html>
```

demo35.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Map object</h3>

  <script>
    let mymap = new Map()
    console.log(mymap);
    console.log(mymap.size);
```

```
    mymap.set('sid', 101);
    mymap.set('sname', "Sri");
    mymap.set('email', "Sri@jlc");
    mymap.set('phone', 5599);

    console.log(mymap);
    console.log(mymap.size);
    console.log("-------------");
    console.log(mymap.has('sname'));
    console.log(mymap.has('phone'));
    console.log(mymap.has('city'));

    console.log(mymap.get("sname"));
    console.log(mymap.get("phone"));
    console.log(mymap.get("city"));
    console.log("-------------");

    console.log(mymap);
    console.log(mymap.size);

    console.log(mymap.delete('sname'));
    console.log( mymap.delete('city'));

    console.log(mymap);
    console.log(mymap.size);


    console.log("Done!!!");
  </script>
</body>
</html>
```

demo36.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Map object</h3>

  <script>
    let mymap = new Map()
    console.log(mymap);
    console.log(mymap.size);

    mymap.set('sid', 101);
    mymap.set('sname', "Sri");
    mymap.set('email', "Sri@jlc");
    mymap.set('phone', 5599);

    console.log(mymap);
    console.log(mymap.size);

    console.log("-------------");
    for (let [key, value] of mymap) {
      console.log(key + ' = ' + value)
    }
    console.log("-------------");
    for (let key of mymap.keys()) {
      console.log(key)
    }

    console.log("-------------");
    for (let value of mymap.values()) {
      console.log(value)
    }
    console.log("-------------");
    for (let [key, value] of mymap.entries()) {
      console.log(key + ' = ' + value)
    }
```

```
    console.log("-------------");
    mymap.forEach((value, key) => {
      console.log(key + ' = ' + value)
    })
    console.log("Done!!!");
  </script>
</body>
</html>
```

demo37.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Map object</h3>

  <script>
    let myset = new Set()
    console.log(myset);
    console.log(myset.size);

    myset.add(10);
    myset.add(20);
    myset.add(30);
    myset.add(40);
    myset.add(50);
    myset.add(30);
    myset.add(40);

    console.log(myset);
    console.log(myset.size);
    console.log("-------------");

    console.log(myset.has(10));
    console.log(myset.has(50));
    console.log(myset.has(60));
```

```
    console.log("----------");

    console.log(myset);
     console.log(myset.size);

     console.log(myset.delete(10));
     console.log(myset.delete(60));

     console.log(myset);
     console.log(myset.size);

     console.log("Done!!!");
  </script>
</body>
</html>
```

demo38.html

```
<!DOCTYPE html>
<html lang="en">
<body>
  <h3> Map object</h3>

  <script>
     let myset = new Set()
     console.log(myset);
     console.log(myset.size);

     myset.add(10);
     myset.add(20);
     myset.add(30);
     myset.add(40);
     myset.add(50);

     console.log(myset);
     console.log(myset.size);
```

```
    console.log("-----1--------");
    for (let item of myset) {
      console.log(item)
    }

    console.log("------2-------");
    for (let item of myset.keys()) {
      console.log(item)
    }

    console.log("-----3--------");
    for (let item of myset.values()) {
      console.log(item)
    }

    console.log("------4-------");
    for (let item of myset.entries()) {
      console.log(item)
    }

    console.log("------5-------");
    myset.forEach((value) => {
      console.log(value)
    })

    console.log("Done!!!");
  </script>
</body>

</html>
```