**Modules**

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script type="module" src="main.js"></script>

</body>

</html>

lib.js

export function halfOf(x){

    return x / 2;

}

main.js

import {halfOf} from './lib.js';

console.log(halfOf(84));

run this index.html with live server.

Example 2 – continuation from example 1

lib.js

export function halfOf(x){

    return x / 2;

}

export function multiply(x,y){

    return x \* y;

}

main.js

import {halfOf,multiply} from './lib.js';

console.log(halfOf(84));

console.log(multiply(21, 2));

run the index.html with live server.

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script type="module" src="main.js"></script>

</body>

</html>

doSomething.js

export default function (){

    console.log('I did something')

};

main.js

import doSomething from './doSomething.js';

doSomething();

run the index.html file with live server.

**Modules Export Bindings and not values**

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script type="module" src="main.js"></script>

</body>

</html>

validator.js

export let flag = false;

export function touch() {

   flag = true;

}

main.js

import { flag, touch } from './validator.js';

console.log(flag);

touch();

console.log(flag);

run the index.html file with live server.

**CLASSES**

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script src="class1.js"></script>

</body>

</html>

class1.js

class Vehicle {

    constructor(wheels) {

       this.wheels = wheels;

    }

    toString() {

       return '(' + this.wheels + ')';

    }

 }

 class Car extends Vehicle {

    constructor(color) {

       super(4);

       this.color = color;

    }

    toString() {

       return super.toString() + ' colored: ' + this.color;

    }

 }

 let car = new Car('blue');

 car.toString();

 console.log(car instanceof Car);

 console.log(car instanceof Vehicle);

run the index.html file with live server.

**Arrow Functions**

**Normal Function**

function sum1(a, b) {

    return a + b;

}

let result = sum1(1, 2);

console.log(result);

**Arrow Function**

const sum1 = (a, b) => {

    return a + b;

}

let result = sum1(1, 2);

console.log(result);

**OR**

const sum1 = (a, b) => a + b;

let result = sum1(1, 2);

console.log(result);

**Let and Const**

**Const**

**Index.html**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <script src="class1.js"></script>

</body>

</html>

Class1.js

function reinitConst(){

    const me = 1;

    console.log(me);

    me = 2;

    console.log(me);

 }

 reinitConst();

run the index.html file with live server.

**Let**

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <script src="class1.js"></script>

</body>

</html>

class1.js

function reinitConst(){

    let me = 1;

    console.log(me);

    me = 2;

    console.log(me);

 }

 reinitConst();

run the index.html file with live server.

**Var**

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <script src="class1.js"></script>

</body>

</html>

class1.js

function iterateVar() {

   for(var i = 0; i < 10; i++) {

      console.log(i);

   }

   console.log(i);

}

iterateVar();

run the index.html file with live server.

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <script src="class1.js"></script>

</body>

</html>

**Let**

class1.js

function iterateLet() {

   for(let i = 0; i < 10; i++) {

      console.log(i);

   }

   console.log(i);

}

iterateLet();

run the index.html file with live server.

**DEFAULT, REST, SPREAD**

DEFAULT

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script src="def.js"></script>

</body>

</html>

def.js

function add(x, y = 0) {

    console.log(x + y);

 }

 add(1);

 add(1,2);

run the index.html with live server.

**REST**

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script src="rest.js"></script>

</body>

</html>

Rest.js

function userFriends(user, ...friends) {

    console.log(user + ' has ' + friends.length + ' friends');

 }

 userFriends('User', 'Bob', 'Alice');

run the index.html with live server.

**SPREAD**

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script src="spread.js"></script>

</body>

</html>

spread.js

function userTopFriends(firstFriend, secondFriend, thirdFriends) {

    console.log(firstFriend);

    console.log(secondFriend);

    console.log(thirdFriends);

 }

 userTopFriends(...['Alice', 'Bob', 'Michelle']);

run the index.html with live server.

**JAVASCRIPT CLOSURES**

Index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script src="clos.js"></script>

</body>

</html>

Clos.js

function parent() {

    const message = 'Hello World';

    function child() {

       alert (message);

    }

    child();

 }

 parent();

run the index.html with live server.

index.html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>JS MOSULE</title>

</head>

<body>

    <script src="cos2.js"></script>

</body>

</html>

cos2.js

function parent() {

    const message = 'Hello World';

    function child() {

       alert (message);

    }

    return child;

 }

 const childFN = parent();

 childFN();

run the index.html with live server.