Day 6 –Assignment

Bollepalli Sai Sreekanth

1.Write a JavaScript program to use let and const.

A.

Code:

let msg="hi,how are you";

console.log("let example: let variable can be changed and cannot re-declare")

console.log(msg);

{

    let msg="hello world";

    console.log(`msg inside a scope ${msg}`);

}

console.log(msg)

msg="i love to code";

console.log(`After rewriting msg ${msg}`)

console.log("const Example:")

const a="const once declared cannot change and cannot re-declare";

{const a="hi";

console.log(a);

}

console.log(a);

a="my name is sreekanth";

const a="kk";

console.log(a);

output:

let example: let variable can be changed and cannot re-declare

hi,how are you

msg inside a scope hello world

hi,how are you

After rewriting msg i love to code

const Example:

hi

const once declared cannot change and cannot re-declare

a="my name is sreekanth";

^

TypeError: Assignment to constant

SyntaxError: Identifier 'a' has already been declared

2.Write a Javascript program to use spread operator

A.

Code:

var a=[1,2,3];

var b=[5,6,7];

var c=[...a,...b];

console.log(c);

var d=['a','b','c','d'];

c=[...c,...d]

console.log(c);

output:

[ 1, 2, 3, 5, 6, 7 ]

[

1, 2, 3, 5, 6,

7, 'a', 'b', 'c', 'd'

]

3. Write a Javascript program to use rest operator.

A.

Code:

function add(...add){   //rest operator

    var sum=0;

    for(let i of add){

        sum=sum+i;

    }

    console.log(sum);

}

add(1,2,3,4,5);

output:

15

4. Write a Javascript program to use spread & rest operator in a same

Program.

A.

Code:

//rest operator

function add(...points){

    var sum=0;

    for(let i of points){

        sum=sum+i;

    }

    return sum;

}

var a=[10,20,30,40];

var b=[50,60,70,80];

console.log(add(...a,...b));//spread operator

output:

360

5. Export a Module from JS and import all the functions in a particular

Program.

A.

Code:

expo.js

var a={

    add: function(a,b){

        return a+b;

    },

    sub: function(a,b){

        return a-b;

    },

    multiply: function(a,b){

        return a\*b;

    }

}

module.exports=a;

q5.js

var mymod=require("./expo");

console.log(mymod.add(9,8));

console.log(mymod.sub(9,8));

console.log(mymod.multiply(9,8));

output:

17

1

72

6. Write a Javascript program using Promises using multiple Handlers.

A.

Code:

let p=new Promise((resolve,reject)=>{

    setTimeout(()=>{

        resolve(20);

    },3000)

});

p.then((result)=>{

    console.log(result\*result)

    return result\*2;

})

p.then(result=>{

    console.log(result+result);

    return result\*2

})

p.then((result)=>{

    console.log(result+result+result);

})

Output:

400

40

60

7. Write a Javascript program for Fibonacci of 5 and Factorial of 6 using

multiple Promises.

A.

code:

function fact(n){

    var sum=1;

    for(let i=n;i>=1;i--){

        sum=sum\*i;

    }

    return sum;

}

function fib(n){

    series=[]

    var a=-1;

    var b=1;

    for(i=1;i<=n;i++){

        var sum=a+b;

        a=b;

        b=sum;

        series.push(sum);

    }

    return series;

}

function fun(n,condition){

    return new Promise((resolve,reject)=>{

        if(condition=="factorial"){

            resolve(fact(n));

        }

        else if(condition==="fibonacci"){

            resolve(fib(n));

        }

        else{

            reject(`${condition} is not defined`);

        }

    });

}

try{

fun(6,"factorial").then(user=>console.log(user))

.catch(err=>console.log(err));}catch(error){

    console.log(`caught by try/catch ${error}`);

}

try{

    fun(5," fibonacci").then(user=>console.log(user))

    .catch(err=>console.log(err));}catch(error){

        console.log(`caught by try/catch ${error}`);

    }

    try{

        fun(5,"RandomCondition").then(user=>console.log(user))

        .catch(err=>console.log(err));}catch(error){

            console.log(`caught by try/catch ${error}`);

        }

Output:

720

[ 0, 1, 1, 2, 3 ]

RandomCondition is not defined

8. Write an Example of Arrow functions and sort an array reversely

A.

Code:

var Sort =(points)=>{

    return points.sort((a,b)=>b-a);

}

console.log(Sort([4,2,6,2,3,8,1,0]));

output:

[

8, 6, 4, 3,

2, 2, 1, 0

]

9. Give an Example of Async, Await and Promise.

A.

Code:

const a=10;

const b=10;

const c=10;

var sum=0;

var promise=new Promise(function(resolve,reject){

    setTimeout(function(){

        sum=a+b;

        resolve(sum);

    },4000);

});

async function add(){

    let wait=await promise;

    console.log(wait);

    sum=sum+c;

    console.log(sum);

}

add();

output:

20

30