ES6 Assignments

1. **Constants:** Declare a constant & confirm its value cannot be changed.

function print(str){

    "use strict";

    const sentence = str + " " + "how are you?";

    sentence = str + " " + " there";

      for (let i = 1; i < str.length; i += 2) {

          console.log(sentence);

          }

      }

      print("Hi");

Output: error TS2588: Cannot assign to 'sentence' because it is a constant.

1. **Scoping:** Declare a variable inside if condition & make sure that it is not accessible outside if condition.

var a=1;

var b=2;

if(a === 1){

var a=10;

let b=20;

console.log(a,b);

}

console.log(a,b);

Output:

10 20

10 2

1. **Enhanced object properties:** Create an ‘Order’ object having data members ‘id’, ‘title’, ‘price’. Add the methods printOrder() & getPrice(). Now, copy the order object using Object.assign().

function id(id: any, title: any, price: any) {

    throw new Error("Function not implemented.");

}

function title(id: any, title: any, price: any) {

    throw new Error("Function not implemented.");

}

function price(id: any, title: any, price: any) {

    throw new Error("Function not implemented.");

}

const Order = {

    id: 0,

    title: '',

    price: 0,

    printOrder() { console.log(id, title, price) },

    getPrice() { console.log(price) },

    assign(id, title, price) {

        this.id = id;

        this.title = title;

        this.price = price;

    },

};

Order.assign(1, 'WOW', 200);

console.log(Order);

Output:

{

id: 1,

title: 'WOW',

price: 200,

printOrder: [Function: printOrder],

getPrice: [Function: getPrice],

assign: [Function: assign]

}

1. **Arrow functions:** Take an array of strings & convert it into another array of object which has two properties {string, string\_length}. For example:

let names = [‘Tom’, ‘Ivan’, ‘Jerry’]

Output: [ {name: ’Tom’, length: 3}, {name: ’Ivan’, length: 4 }, {name: ’Jerry’, length: 5} ]

let ar=['tom','Ivan','Jerry'];

    for (let i in ar){

    console.log("name:"+ar[i],

    "length:" +ar[i].length);

};

Output:

name:Tom length:3

name:Ivan length:4

name:Jerry length:5

1. **Extended parameter handling:**
   1. Write a add() with default values.

let getValue = function(a=10,b=5){

    console.log(a+b);

};

getValue();

Output:

15

* 1. Write a function userFriends() that takes 2 arguments username & array of user friends. The function should print username & his list of friends. (Use rest parameters)

let displayFriends = function(username, ...arrays) {

    console.log(username);

    for (let i in arrays) {

      console.log(arrays[i]);

    }

    }

    let username = "userName"

    displayFriends(username, ['Harsha', 'Ammu', 'Siddiq']);

Output:

userName

[ 'Harsha', 'Ammu', 'Siddiq' ]

* 1. Write a function printCapitalNames() that takes five names as argument & prints them in capital letters. Use spread operator in order to call printCapitalNames() function.

let displayColors=function(...colours){

    for(let i in colours){

        console.log(colours[i].toUpperCase());

    }

}

let colourArray=['Harsha', 'Ammu', 'Siddiq', 'kanth'];

displayColors(...colourArray);

Output:

HARSHA

AMMU

SIDDIQ

KANTH

1. **Template literals:** Draft a ticket to Sysnet that describes problem with your laptop. Use ‘template literals’ to add value of laptop model, your desk no, your name etc.

const username= 'Harsha';

const laptopModel = 'DELL Inspiron 7570';

const deskNo = '5';

console.log(`Name:${username}, Desk No:${deskNo}, Laptop Model: ${laptopModel}`);

output:

Name:Harsha, Desk No:5, Laptop Model: DELL Inspiron 7570

1. **De-structuring assignment:**
   1. Suppose there is a javascript array with 4 elements. Print the value of 3rd element using array matching.

let emp = ["sun", "run", "win", "won"];

let [, , element3, ] = emp;

console.log(element3);

Output:

Win

* 1. Create an organization object having attributes name, address. Write a program to retrieve pin code of an address using object deep matching.

let org = {

    name1 : "Harsha",

    address : {

    city: "Vijayawada",

    state: "Andhra Pradesh",

    zipCode: "521241"

  }

  }

  let {name1, address} = org;

  console.log(name1);

   console.log(address.zipCode);

Output:

Harsha

521241

1. **Classes & Modules:** Write a class Account with attributes id, name, balance. Add two sub classes SavingAccount & CurrentAccount having specific attribute interest & cash\_credit respectively. Create multiple saving & current account objects. Write a functionality to find out total balance in the bank.

class Account{

    balance: any;

    constructor(id,name,balance){

        id= this.id;

        name = this.name;

        balance  = this.balance;

    }

    }

    class SavingAccount extends Account {

        interest: any;

        cash\_credit: any;

        constructor(id,name,balance,interest,cash\_credit){

              super(id,name,balance);

              interest = this.interest;

              cash\_credit = this.cash\_credit;

        }

          showName(){

              let n = this.name;

            console.log(n);

        }

        showBalance = ()=>{let b = this.interest\*this.cash\_credit

        console.log('the balance in your account is: '+b);}

    }

    class CurrentAccount extends Account {

        interest: any;

        cash\_credit: any;

        constructor(id,name,balance,interest,cash\_credit){

              super(id,name,balance);

              interest = this.interest;

              cash\_credit = this.cash\_credit;

        }

        showBalance = ()=>{let b = this.interest\*this.cash\_credit

        console.log('the balance in your account is: '+b);}

    }

    var p = new SavingAccount(1,'Tom',10000,0.5,2000);

    p.showBalance();

    p.showName();