**Objects Assignment A3 –by Jyothi Gangarsha K**

**Q1.**

class Rectangle {

    constructor(width,height){

      this.width=width;

      this.height=height;

    }

    getwidth(){

       console.log('The width is : '+this.width);

   }

   getheight(){

      console.log('The height is : '+this.height);

   }

  }

  var r1 = new Rectangle(10,10)

  r1.getwidth();

  r1.getheight();

  var r1 = new Rectangle(11,11)

  r1.getwidth();

  r1.getheight();

**Q2.**

class Rectangle {

    constructor(width,height){

      this.width=width;

      this.height=height;

    }

    getwidth(){

       console.log('The width is : '+this.width);

   }

   getheight(){

      console.log('The height is : '+this.height);

   }

  getArea(){

    console.log(this.width\*this.height);

  }

  }

  var r1 = new Rectangle(10,10)

  r1.getwidth();

  r1.getheight();

  r1.getArea()

  var r1 = new Rectangle(11,11)

  r1.getwidth();

  r1.getheight();

  r1.getArea()

**Q3.**

class rectangle{

    constructor(width, height) {

        this.width = width;

        this.height = height;

    }

    getArea()

    {

      console.log(this.width\*this.height);

    }

    }

    const harsha = new rectangle(10,10);

    console.log(harsha.width)

    console.log(harsha.height)

    harsha.height=11;

    console.log(harsha.height)

    harsha.getArea()

   /\* class Rectangle {

        constructor(width,height){

          this.width=width;

          this.height=height;

        }

      getArea(){

        console.log(this.width\*this.height);

      }

      }

      var r1 = new Rectangle(10,10)

      r1.getArea()

      r1.height=50;

    r1.getArea();

    \*/

**Q4.**

function person(firstName, lastName) {

    this.firstName = firstName;

    this.lastName = lastName;

}

const harsha = new person('Ammu', 'Dasari');

console.log(harsha.firstName)

console.log(harsha.lastName)

harsha.lastName="Karampudi";

console.log(harsha.firstName)

console.log(harsha.lastName)

Q5.

function person(firstName, lastName) {

    this.firstName = firstName;

    this.lastName = lastName;

}

const harsha = new person('Jyothi', 'karampudi');

console.log(harsha.middleName);

// This gives undefined as the person class doesn't have middleName attribute in it and undefined.

harsha.middleName='Gangarsha';

console.log(harsha.firstName);

console.log(harsha.middleName);

console.log(harsha.lastName);

Q6.

firstName = 'Jyothi'

lastName = 'karampudi'

var fullName = `${firstName} ${lastName}`;

eval(`${firstName} ${lastName}`);

Q7.

firstName = 'Jyothi';

lastName = 'karampudi';

var fullName = `${firstName} ${lastName}`;

parseJSON(fullName);

Q8.

function person(fname, lname, age, skills, dateofbirth, address, married, profession) {

    this.fname = fname;

    this.lname = lname;

    this.age = age;

    this.skills = skills;

    this.dateofbirth = dateofbirth;

    this.address = address;

    this.married = married;

    this.profession = profession;

}

person1 = new person("harsha", "goud", 22, ["c"], "24/10/1996", { city: "hyderabad", pincode: "521185" }, "false", "sr analyst")

person2 = new person("harish", "chinna", 21, "HTML", "08/06/1997", { city: "Ameerpet", pincode: "500038" }, "false", "jr analyst")

console.log(person1);

console.log(person2);

Q9.

function person(fname, lname, age, skills, dateofbirth, address, married, profession) {

    this.fname = fname;

    this.lname = lname;

    this.age = age;

    this.skills = skills;

    this.dateofbirth = dateofbirth;

    this.address = address;

    this.married = married;

    this.profession = profession;

}

amithab = new person("amithab", "bachan", 22, ["c"], "24/10/1996", { city: "hyderabad", pincode: "521185" }, "false", "sr analyst")

abhisheik = new person("abhisheik", 21, "HTML", "08/06/1997", "false", "jr analyst")

var abhisheik = Object.create(amithab);

console.log(amithab);

console.log(abhisheik.lname);

console.log(abhisheik.address);

Q10.

function person(fname, lname, age, skills, dateofbirth, address, married, profession) {

    this.fname = fname;

    this.lname = lname;

    this.age = age;

    this.skills = skills;

    this.dateofbirth = dateofbirth;

    this.address = address;

    this.married = married;

    this.profession = profession;

}

amithab = new person("amithab", "bachan", 22, ["c"], "24/10/1996", { city: "hyderabad", pincode: "521185" }, "false", "sr analyst")

abhisheik = new person("abhisheik", 21, "HTML", "08/06/1997", "false", "jr analyst")

var abhisheik = Object.create(amithab);

Aaradhya = new person("aaradhya", amithab.lname, amithab.age, "HTML", "08/06/1997", abhisheik.address, abhisheik.married, amithab.profession);

console.log(amithab);

console.log(abhisheik.lname);

console.log(abhisheik.address);

console.log(Aaradhya);

Q11.

function BankAccount(accountNumber, accountHolderName, accountBalance, isSalary = false, odLimit = NaN) {

    this.accountNumber = accountNumber;

    this.accountHolderName = accountHolderName;

    this.accountBalance = accountBalance;

    this.isSalary = isSalary;

    this.odLimit = odLimit;

}

BankAccount.withDraw() = function(drawAmount) {

    if (this.isSalary) {

        if (this.accountBalance > drawAmount) {

            this.accountBalance -= drawAmount;

            console.log('Withdrawn Successfully');

        }

        console.log('Insufficient Funds');

    } else {

        if (drawAmount < accountBalance + odLimit) {

            this.accountBalance -= drawAmount;

            console.log('Withdrawn Successfully');

        }

        console.log('Insufficient Funds');

    }

}

var account1 = new BankAccount(1, 'harsha', 1000, true);

var account1 = new BankAccount(2, 'nishanth', 5000, false, 1000);