// Q.1 JAVASCRIPT QUESTION

class BankAccount {

    constructor(name, balance) {

        this.name = name;

        this.balance = balance;

    }

    desposite(amount) {

        if (amount > 0) {

            this.balance += amount;

            console.log(`Deposited Rs. ${amount} New Balance Rs. ${this.balance}`);

        }

        else {

            console.log(`Invalid deposite amount. Please Enter a Postive amount`);

        }

    }

    withdraw(amount) {

        if (amount > 0) {

            if (this.balance >= amount) {

                this.balance -= amount;

                console.log(`Withdraw Rs. ${amount} New Balance Rs. ${this.balance}`);

            }

            else {

                console.log(`Insufficient Balance`);

            }

        }

        else {

            console.log(`Invalid withdraw amount. Please Enter a Postive amount`);

        }

    }

    checkBalance() {

        console.log(`Account Balance for ${this.name} Rs. ${this.balance}`);

    }

}

const account = new BankAccount('Abhi', 1000);

account.checkBalance();

account.desposite(500);

account.desposite(-50);

account.withdraw(200);

account.withdraw(1500);

account.withdraw(-300);

account.checkBalance();

// Q.2 JAVASCRIPT QUESTION

class Student {

  constructor(name, id) {

    this.name = name;

    this.id = id;

    this.enrolledCourses = [];

  }

  enroll(course) {

    this.enrolledCourses.push(course);

  }

  displayEnrolledCourses() {

    console.log(`${this.name}'s enrolled courses:`);

    this.enrolledCourses.forEach(course => {

      console.log(`-${course}`);

    });

  }

}

class Admission {

  constructor() {

    this.enrolledStudents = [];

    this.availableCourses = ['Math', 'Science', 'History'];

  }

  enrollStudent(name, id) {

    const student = new Student(name, id);

    this.enrolledStudents.push(student);

    return student;

  }

  admitStudentToCourse(student, course) {

    if (this.availableCourses.includes(course)) {

      student.enroll(course);

      console.log(`${student.name} has been enrolled in ${course}.`);

    } else {

      console.log(`Error: ${course} is not available.`);

    }

  }

  displayEnrolledStudents() {

    console.log('Enrolled Students:');

    this.enrolledStudents.forEach(student => {

      console.log(`${student.name} (ID: ${student.id})`);

    });

  }

}

// Example Usage:

const admission = new Admission();

const student1 = admission.enrollStudent('John Doe', 1);

const student2 = admission.enrollStudent('Jane Doe', 2);

admission.displayEnrolledStudents();

admission.admitStudentToCourse(student1, 'Math');

admission.admitStudentToCourse(student1, 'Science');

admission.admitStudentToCourse(student2, 'History');

student1.displayEnrolledCourses();

student2.displayEnrolledCourses();

// Q.3 JAVASCRIPT QUESTION

class Temperature{

    constructor(){

        this.\_celsius = 0;

        this.\_fahrenheit = 32;

    }

    get celsius(){

        return this.\_celsius;

    }

    set celsius(value){

        if(typeof value !== 'number'){

            throw new TypeError('Temperature must be a number');

        }

        this.\_celsius = value;

        this.\_fahrenheit = (value\*9/5)+32;

    }

    get fahrenheit(){

        return this.\_fahrenheit;

    }

    set fahrenheit(value){

        if(typeof value !== 'number'){

            throw new TypeError('Temperature must be a number');

        }

        this.\_fahrenheit = value;

        this.\_celsius = (value-32)\*5/9;

    }

    toCelsius(Temperature){

        return (Temperature-32)\*5/9;

    }

    tofahrenheit(Temperature){

        return (Temperature\*9/5)+32;

    }

}

const temp = new Temperature();

console.log(`Intial Celsius : ${temp.celsius} C`);

console.log(`Intial Fahrenheit : ${temp.fahrenheit} C`);

temp.celsius = 25;

console.log(`Celsius : ${temp.celsius} C`);

console.log(`Fahrenheit : ${temp.fahrenheit} C`);

temp.fahrenheit = 68;

console.log(`Celsius : ${temp.celsius} C`);

console.log(`Fahrenheit : ${temp.fahrenheit} C`);

// Q.4 JAVASCRIPT QUESTION

class Shape {

    constructor() { }

    area() {

        return 0;

    }

    perimeter() {

        return 0;

    }

}

class Circle extends Shape{

    constructor(radius){

        super();

        this.radius = radius;

    }

    area(){

        return Math.PI\*this.radius\*this.radius;

    }

    perimeter(){

        return 2\*Math.PI\*this.radius;

    }

}

class Rectangle extends Shape{

    constructor(height, width){

        super();

        this.height = height;

        this.width = width;

    }

    area(){

        return this.width\*this.height;

    }

    perimeter(){

        return 2\*(this.width+this.height);

    }

}

class Triangle extends Shape{

    constructor(base, height){

        super();

        this.base = base;

        this.height = height;

    }

    area(){

        return (this.base\*this.height)/2;

    }

    perimeter(){

        return this.base + this.height + Math.sqrt(this.base \* this.base + this.height \* this.height);

    }

}

const circle = new Circle(5);

console.log(`Circle area ${circle.area()} and Perimeter ${circle.perimeter()}`);

const rectangle = new Rectangle(4,6);

console.log(`Rectangle area ${rectangle.area()} and Perimeter ${rectangle.perimeter()}`);

const triangle = new Triangle(3,4,5,4,3);

console.log(`Triangle area ${triangle.area()} and Perimeter ${triangle.perimeter()}`);

// Q.5 JAVASCRIPT QUESTION

function Product(name, category, price, stock) {

    this.name = name;

    this.category = category;

    this.price = price;

    this.stock = stock;

}

function Inventory() {

    this.products = [];

}

Inventory.prototype.addProduct = function (product) {

    if (product instanceof Product) {

        this.products.push(product);

        console.log(`Product ${product.name} added to inventory`);

    }

    else {

        console.log(`Invalid Product.`);

    }

}

Inventory.prototype.delteProduct = function (name) {

    const index = this.products.findIndex(product => product.name === name);

    if (index !== -1) {

        this.products.splice(index, 1);

        console.log(`Product "${name}" deleted from inventory.`);

    } else {

        console.log(`Product "${name}" not found in inventory.`);

    }

}

const inventory = new Inventory();

const product1 = new Product('Laptop', 'Electronics', 5500, 8);

inventory.addProduct(product1);

inventory.delteProduct('Laptop');

inventory.delteProduct('Laptop');