**Assignmemt-1 & 2**

**Name: -** Makwana Dainik Kalabhai

**Div:-** B  **Roll no.: -** 3133

**Sub: -** JAVA Practical

**Question-1**

package Que1;

class Vehicle {

    String name = "Bike";

    String manufacturer = "Hero";

    void startEngine() {

        System.out.println("\nVehicle's engine is started.");

    }

}

class Car extends Vehicle {

    int numDoors = 4;

    void honkHorn() {

        System.out.println("Car honking its horn");

    }

}

class SportsCar extends Car {

    int topSpeed = 360;

    void activateTurbo() {

        System.out.println("Activating the turbo mode of the sports car.");

    }

}

class MainProgram {

    public static void main(String args[]) {

        Vehicle V1 = new Vehicle();

        Car C1 = new Car();

        SportsCar S1 = new SportsCar();

        V1.startEngine();

        System.out.printf("Name and Manufacturer of the Vahicle is %s and %s\n\n",V1.name,V1.manufacturer);

        C1.honkHorn();

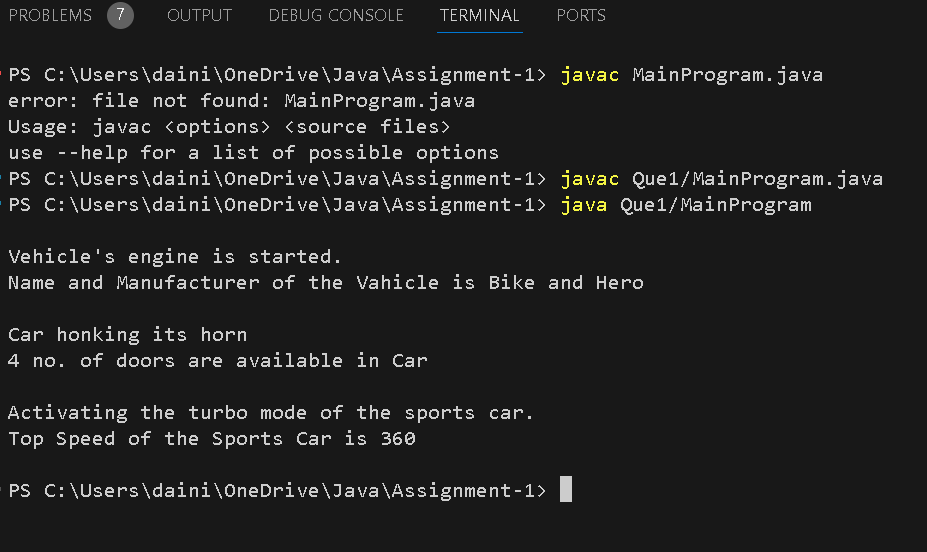
        System.out.printf("%d no. of doors are available in Car\n\n",C1.numDoors);

        S1.activateTurbo();

        System.out.printf("Top Speed of the Sports Car is %d\n\n",S1.topSpeed);

    }

}



**Question-2**

package Que2;

import java.util.\*;

abstract class Thali {

    private double price;

    Thali() {

        price = 0.0;

    }

    abstract void addSabji(double price);

    abstract void addDal(double price);

    abstract void addRice(double price);

    abstract void addRoti(double price);

    void makeThali() {

        System.out.println("\nVeg Thali will be ready in 30 minutes.\n");

    }

    public double getPrice() {

        return price;

    }

    protected void setPrice(double price) {

        this.price = price;

    }

}

class GujaratiThali extends Thali {

    void addSabji(double price) {

        price += getPrice();

        setPrice(price);

    }

    void addDal(double price) {

        price += getPrice();

        setPrice(price);

    }

    void addRice(double price) {

        price += getPrice();

        setPrice(price);

    }

    void addRoti(double price) {

        price += getPrice();

        setPrice(price);

    }

}

class PunjabiThali extends Thali {

    void addSabji(double price) {

        price += getPrice();

        setPrice(price);

    }

    void addDal(double price) {

        price += getPrice();

        setPrice(price);

    }

    void addRice(double price) {

        price += getPrice();

        setPrice(price);

    }

    void addRoti(double price) {

        price += getPrice();

        setPrice(price);

    }

}

class FactoryDesignPattern {

    static void choices(double... price) {

        System.out.println("\nChoices...");

        System.out.printf("1) Add Sabji (Rs.%d)\n", (int) price[0]);

        System.out.printf("2) Add Dal (Rs.%d)\n", (int) price[1]);

        System.out.printf("3) Add Rice (Rs.%d)\n", (int) price[2]);

        System.out.printf("4) Add Roti (Rs.%d)\n", (int) price[3]);

        System.out.printf("5) Complete Order\n");

        System.out.printf("6) Get Total Bill\n");

    }

    static void switchCase(Thali T, double... price) {

        choices(price);

        Scanner S = new Scanner(System.in);

        int choice;

        System.out.print("\nEnter your Choice: ");

        choice = S.nextInt();

        while (true) {

            switch (choice) {

                case 1:

                    T.addSabji(price[0]);

                    choices(price);

                    System.out.print("\nEnter your Choice: ");

                    choice = S.nextInt();

                    break;

                case 2:

                    T.addDal(price[1]);

                    choices(price);

                    System.out.print("\nEnter your Choice: ");

                    choice = S.nextInt();

                    break;

                case 3:

                    T.addRice(price[2]);

                    choices(price);

                    System.out.print("\nEnter your Choice: ");

                    choice = S.nextInt();

                    break;

                case 4:

                    T.addRoti(price[3]);

                    choices(price);

                    System.out.print("\nEnter your Choice: ");

                    choice = S.nextInt();

                    break;

                case 5:

                    T.makeThali();

                    choices(price);

                    System.out.print("\nEnter your Choice: ");

                    choice = S.nextInt();

                    break;

                case 6:

                    System.out.println("\nTotal Payable Amount = Rs." + T.getPrice() + "\n\n");

                    return;

                default:

                    break;

            }

        }

    }

    public static void main(String args[]) {

        Scanner S = new Scanner(System.in);

        System.out.print("Enter 1 for Gujarati Thali & 2 for Punjabi Thali: ");

        int choice = S.nextInt();

        switch (choice) {

            case 1:

                Thali G = new GujaratiThali();

                switchCase(G, 100, 60, 50, 20);

                break;

            case 2:

                Thali P = new PunjabiThali();

                switchCase(P, 120, 80, 70, 40);

                break;

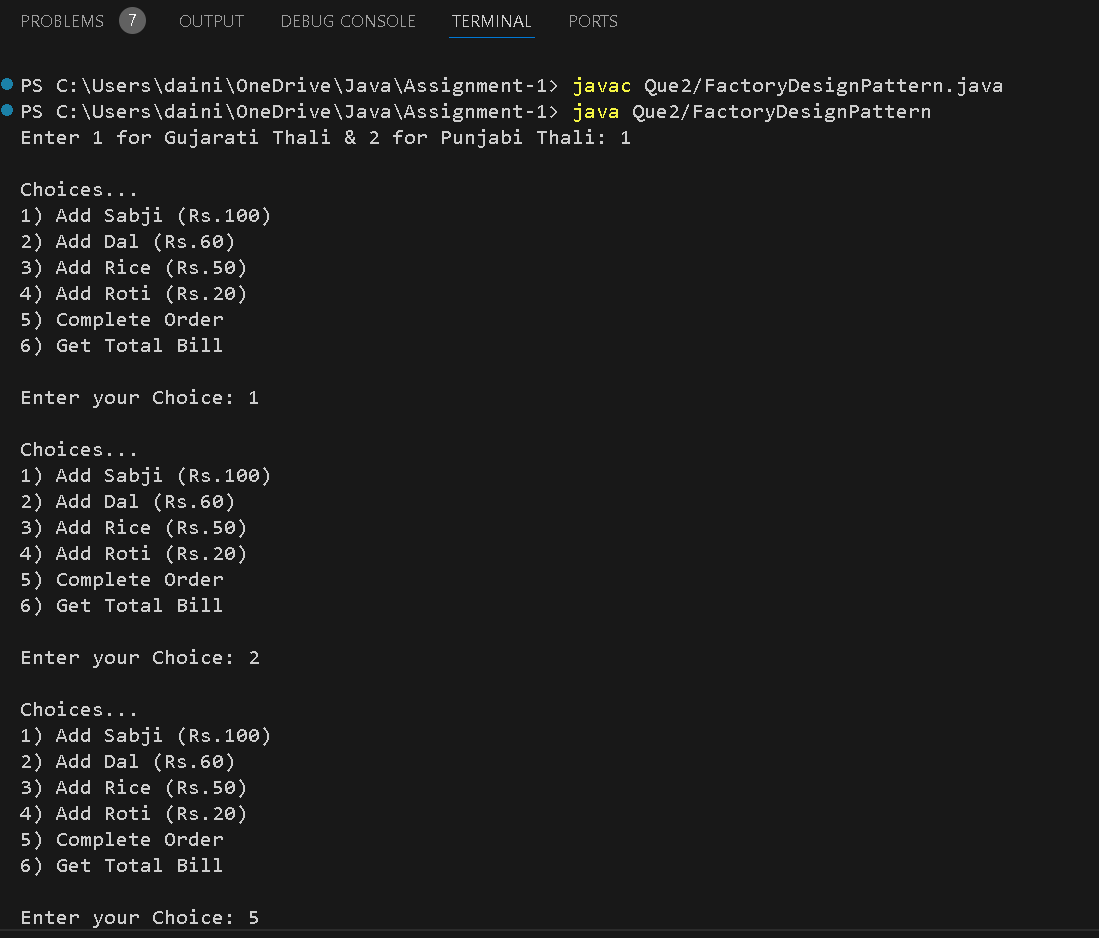
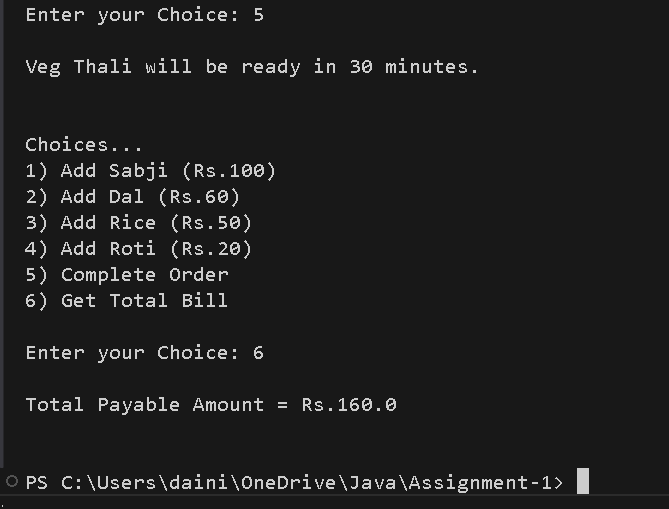
            default:

                break;

        }

    }

}



**Question-3**

package Que3;

import java.util.\*;

interface PizzaOrderSystem {

    Object[][] pizzas = {

            { "Margherita Pizza", 99.0000 },

            { "Cheese n Corn Pizza", 169.0000 },

            { "Cheese n Tomato Pizza", 169.0000 },

            { "Double Cheese Margherita Pizza", 189.0000 },

            { "Fresh Veggie Pizza", 189.0000 },

            { "Farmhouse Pizza", 229.0000 },

            { "Peppy Paneer Pizza", 229.0000 },

            { "Veggie Paradise Pizza", 229.0000 },

            { "Veg Extravaganza Pizza", 249.0000 }

    };

    void placeOrder(String pizzaType, int quantity);

    String checkOrderStatus(int orderId);

    boolean cancelOrder(int orderId);

    double calculateOrderCost(int orderId);

    Object[][] listAvailablePizzas();

}

class PizzaOrderProcessor implements PizzaOrderSystem {

    int orderId;

    ArrayList<ArrayList<Object>> orders = new ArrayList<ArrayList<Object>>();

    PizzaOrderProcessor() {

        // Default constructor

    }

    PizzaOrderProcessor(int orderId) {

        this.orderId = orderId;

    }

    public void placeOrder(String pizzaType, int quantity) {

        for (int i = 0; i < pizzas.length; i++) {

            if (pizzaType == pizzas[i][0]) {

                orders.add(new ArrayList<>(Arrays.asList(this.orderId, pizzaType, pizzas[i][1], quantity)));

                return;

            }

        }

        System.out.println("This item is not available");

    }

    public String checkOrderStatus(int orderId) {

        for (int i = 0; i < orders.size(); i++) {

            if (orders.get(i).contains(orderId)) {

                return "Your order is Activated";

            }

        }

        return "Please! Place your order first";

    }

    public boolean cancelOrder(int orderId) {

        orders.clear();

        return true;

    }

    public double calculateOrderCost(int orderId) {

        double sum = 0;

        double mulQuantity = 1;

        for (int i = 0; i < orders.size(); i++) {

            if (orders.get(i).contains(orderId)) {

                mulQuantity = ((double) orders.get(i).get(2)) \* ((int) orders.get(i).get(3));

                sum += mulQuantity;

            }

        }

        return sum;

    }

    public Object[][] listAvailablePizzas() {

        return pizzas;

    }

}

class PizzaOrderSystemExample {

    static void choices() {

        System.out.println("\nChoices...");

        System.out.println("1) Place Order");

        System.out.println("2) Check Order Status");

        System.out.println("3) Cancel the order");

        System.out.println("4) Get Total Cost: ");

    }

    static void printMenus() {

        PizzaOrderProcessor obj = new PizzaOrderProcessor();

        Object[][] pizzas = obj.listAvailablePizzas();

        System.out.println("\nMenu List....\n");

        for (int i = 0; i < pizzas.length; i++) {

            System.out.printf("%d): %s\n    Price: Rs.%f\n\n", i + 1, pizzas[i][0], pizzas[i][1]);

        }

    }

    public static void main(String args[]) {

        int choice, orderId = 1111, itemNo, quantity;

        PizzaOrderSystem P = new PizzaOrderProcessor(orderId);

        Scanner S = new Scanner(System.in);

        System.out.print("\nEnter 1 to place order: ");

        choice = S.nextInt();

        if (choice != 1) {

            return;

        }

        while (true) {

            switch (choice) {

                case 1:

                    printMenus();

                    System.out.println("\nPlace your order........");

                    System.out.print("Item Number: ");

                    itemNo = S.nextInt();

                    System.out.print("No. of Quantities: ");

                    quantity = S.nextInt();

                    for (int i = 0; i < P.pizzas.length; i++) {

                        if (i == (itemNo - 1)) {

                            P.placeOrder((String) P.pizzas[i][0], quantity);

                        }

                    }

                    choices();

                    System.out.print("\nEnter your choice: ");

                    choice = S.nextInt();

                    break;

                case 2:

                    System.out.println(P.checkOrderStatus(orderId));

                    choices();

                    System.out.print("\nEnter your choice: ");

                    choice = S.nextInt();

                    break;

                case 3:

                    if (P.cancelOrder(orderId)) {

                        System.out.println("\nYour order has been cancelled succefully");

                    }

                    choices();

                    System.out.print("\nEnter your choice: ");

                    choice = S.nextInt();

                    break;

                    case 4:

                    if (P.calculateOrderCost(orderId) != 0) {

                        System.out.printf("\nYour total payable amount = Rs.%f", P.calculateOrderCost(orderId));

                        System.out.println("\nHave a nice day!");

                        return;

                    }

                    else {

                        System.out.println("\nPlease! Place the order first.");

                        choices();

                        System.out.print("\nEnter your choice: ");

                        choice = S.nextInt();

                    }

                    break;

                default:

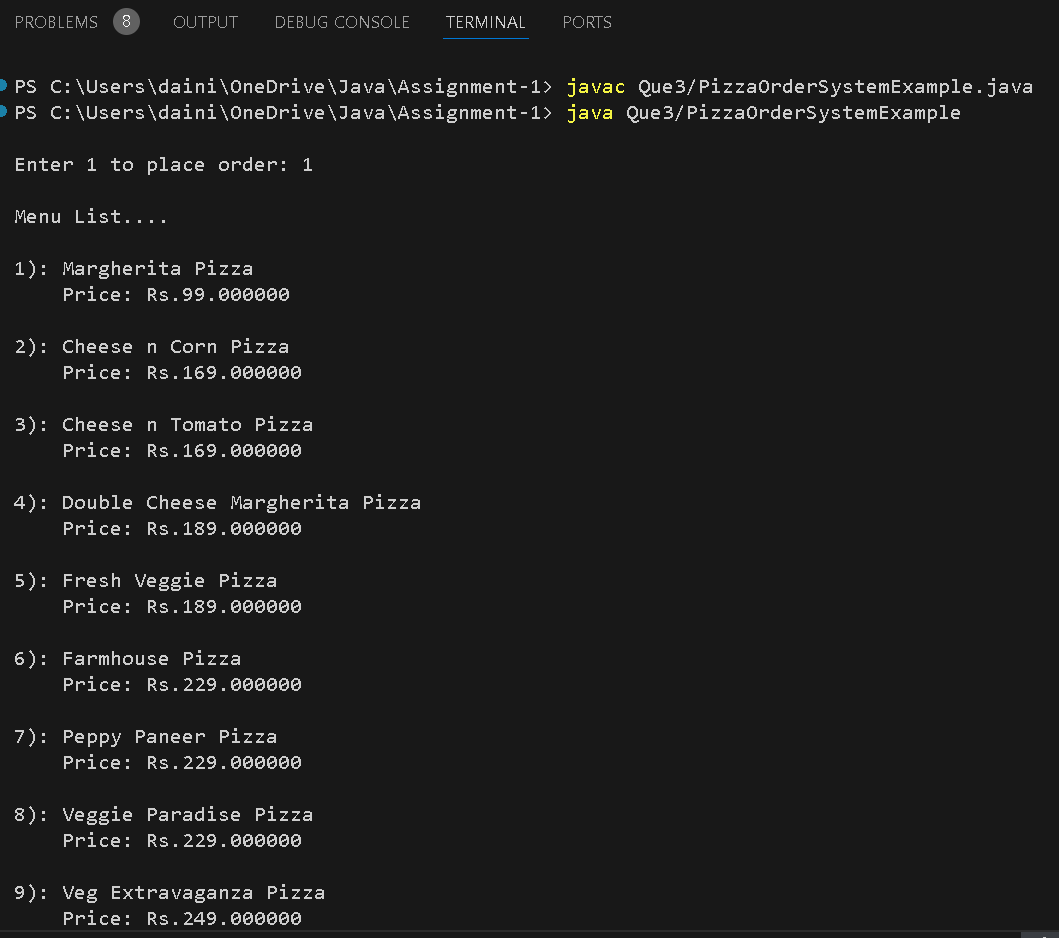
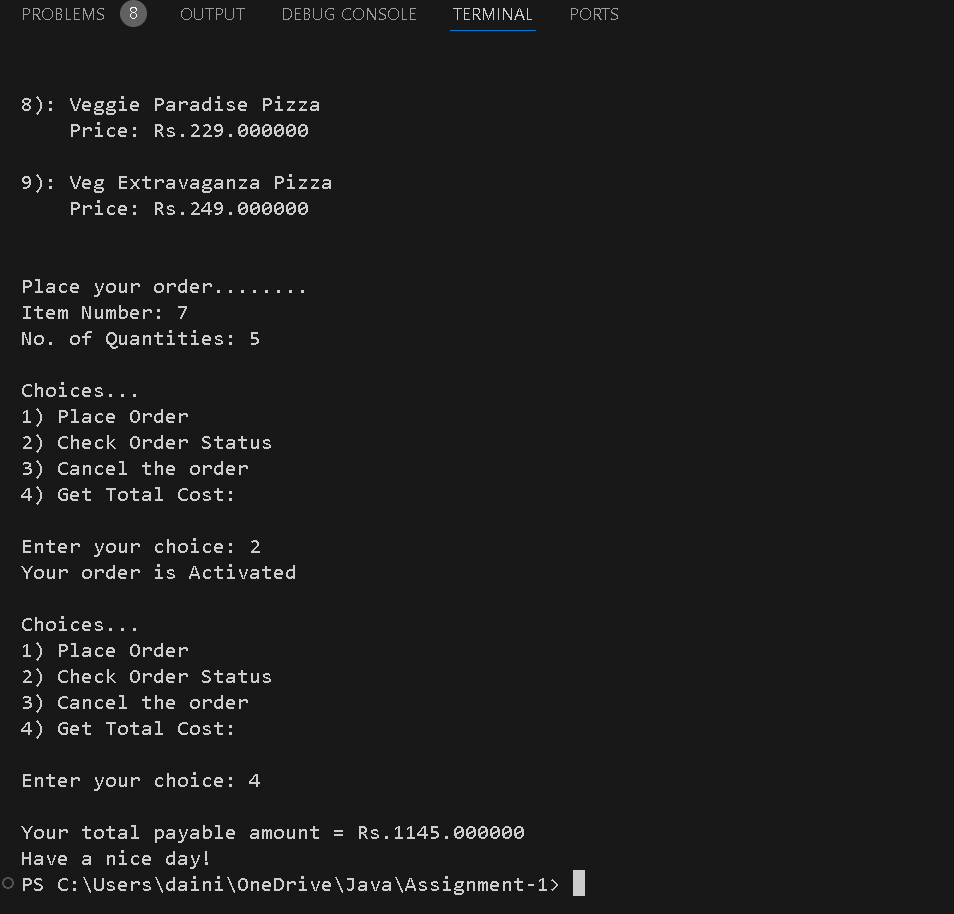
                    return;

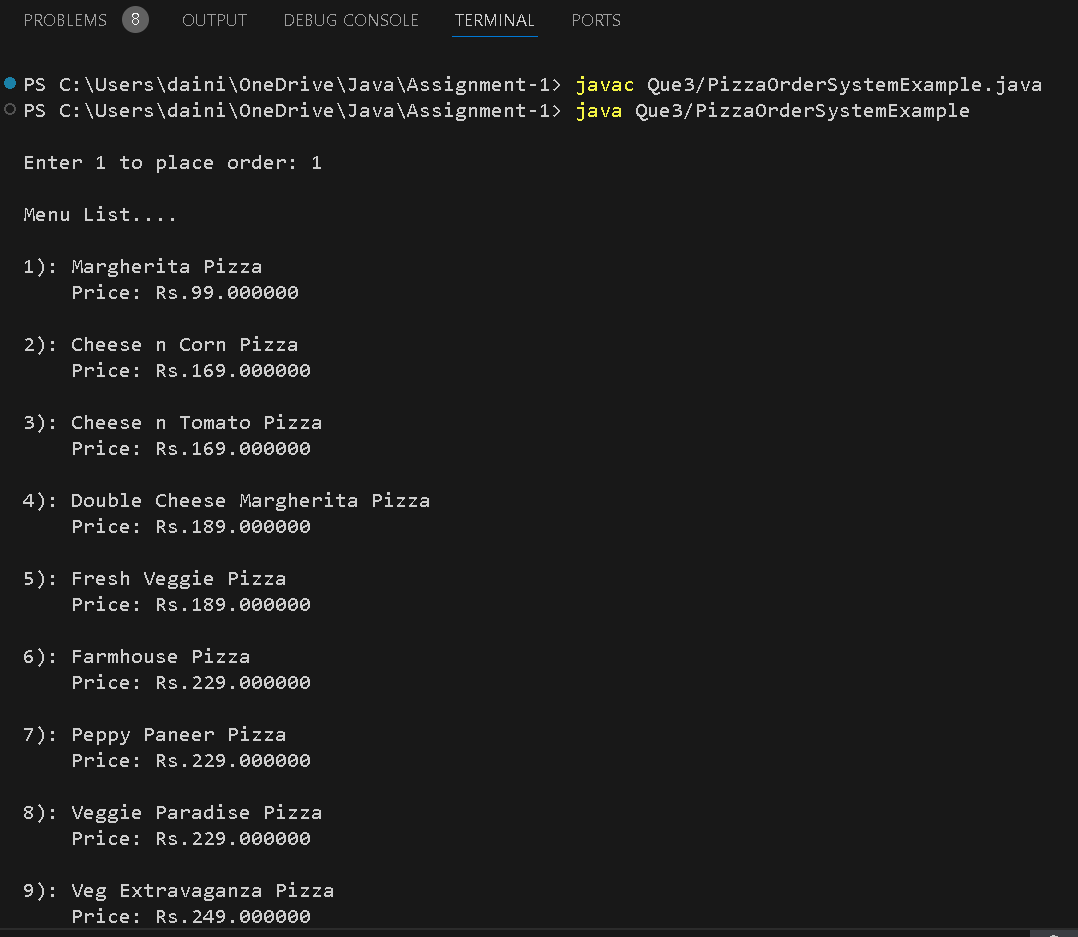
            }

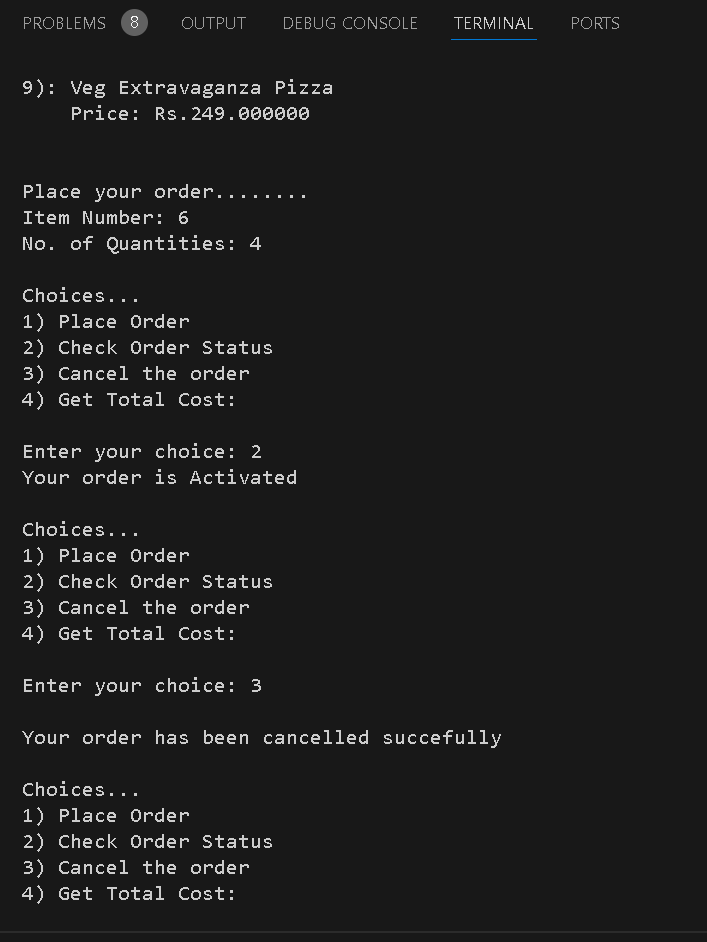
        }

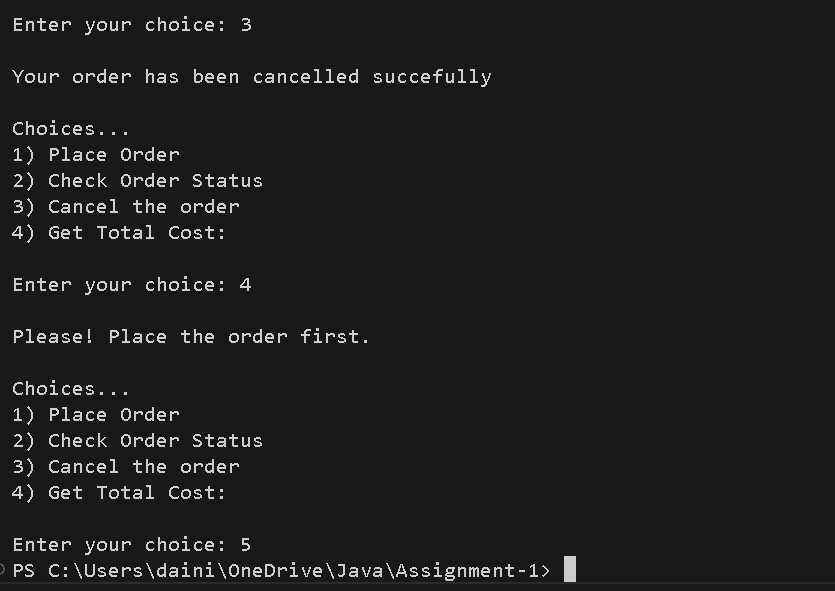
    }

}









**Question-4**

package Que4;

class Person {

    private String name;

    private int age;

    void setName(String name) {

        this.name = name;

    }

    void setAge(int age) {

        if (age > 0) {

            this.age = age;

        } else {

            System.out.println("Please! Enter the valid age");

        }

    }

    String getName() {

        return name;

    }

    int getAge() {

        return age;

    }

    void introduce() {

        System.out.println("My Name is " + getName() + " and I am " + getAge()+" years old.");

    }

}

class Student extends Person {

    private int studentId;

    void setStudentId(int studentId) {

        this.studentId = studentId;

    }

    int getStudentId() {

        return studentId;

    }

    @Override

    void introduce() {

        System.out.println("\nMy Name is " + getName() + " and I am " + getAge()+" years old. I am a Student and my ID is " + studentId);

    }

    void study() {

        System.out.println(getName()+" is studying now.");

    }

}

class Teacher extends Person {

    private String subject;

    void setSubject(String subject) {

        this.subject = subject;

    }

    String getSubject() {

        return subject;

    }

    @Override

    void introduce() {

        System.out.println("\nMy Name is " + getName() + " and I am " + getAge()+" years old."+getName()+" is teaching " + subject);

    }

    void teach() {

        System.out.println(getName()+" is teaching now.\n");

    }

}

class SchoolSystem {

    public static void main(String args[]) {

        Student S = new Student();

        S.setStudentId(3133);

        S.setName("Dainik Makwana");

        S.setAge(19);

        S.introduce();

        S.study();

        Teacher T = new Teacher();

        T.setName("Pritesh Vyas");

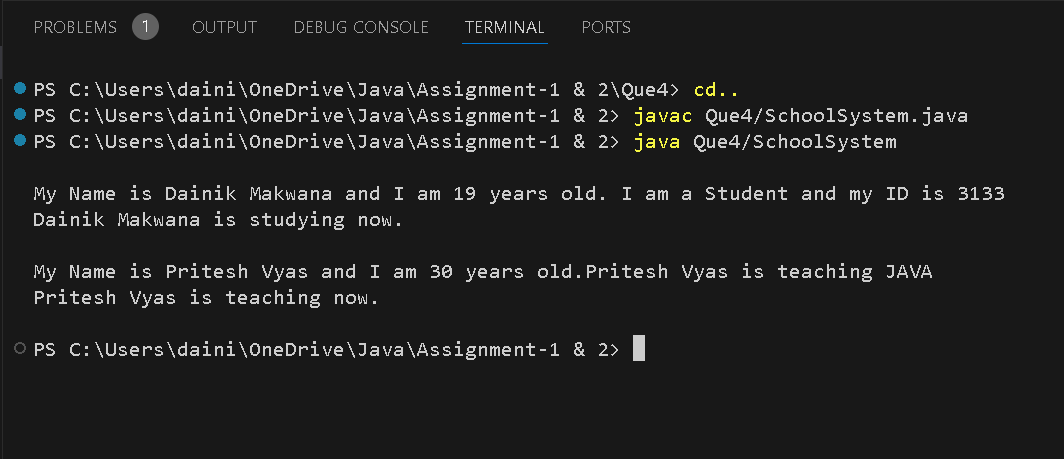
        T.setAge(30);

        T.setSubject("JAVA");

        T.introduce();

        T.teach();

    }

}

**Question-5**

package Que5;

class User {

    private String userName , emailId;

    User() {

    }

    User(String userName, String emailId) {

        this.userName = userName;

        this.emailId = emailId;

    }

    String getUserName() {

        return userName;

    }

    String getEmailId() {

        return emailId;

    }

}

class Professor extends User {

    private String department;

    Professor(String userName, String emailId, String department) {

        super(userName, emailId);

        this.department = department;

    }

    String getDepartment() {

        return department;

    }

}

class Course {

    private int code, creditHours;

    private String name;

    Course(int code, String name, int creditHours) {

        this.code = code;

        this.name = name;

        this.creditHours = creditHours;

    }

    int getCode() {

        return code;

    }

    String getName() {

        return name;

    }

    int getCreditHours() {

        return creditHours;

    }

}

class Department {

    String name, professor1,course1;

    Department(String name) {

        this.name = name;

    }

    void setProfessor1(String professor1) {

        this.professor1 = professor1;

    }

    void setCourse1(String course1) {

        this.course1 = course1;

    }

}

class UniversityDepartmentSystem {

    public static void main(String args[]) {

        Professor P = new Professor("Hitesh Sir", "hitesh123@gmail.com", "Computer Science");

        Course C = new Course(31, "Int. M.Sc.(CA & IT)", 8);

        Department D = new Department("Programming");

        D.setProfessor1("Pritesh Vyas");

        D.setCourse1("JAVA Development");

        System.out.println("\n\nDetails about Professor");

        System.out.printf("\tProfessor Name: %s",P.getUserName());

        System.out.printf("\n\tProfessor Email ID: %s",P.getEmailId());

        System.out.printf("\n\tDepartment: %s",P.getDepartment());

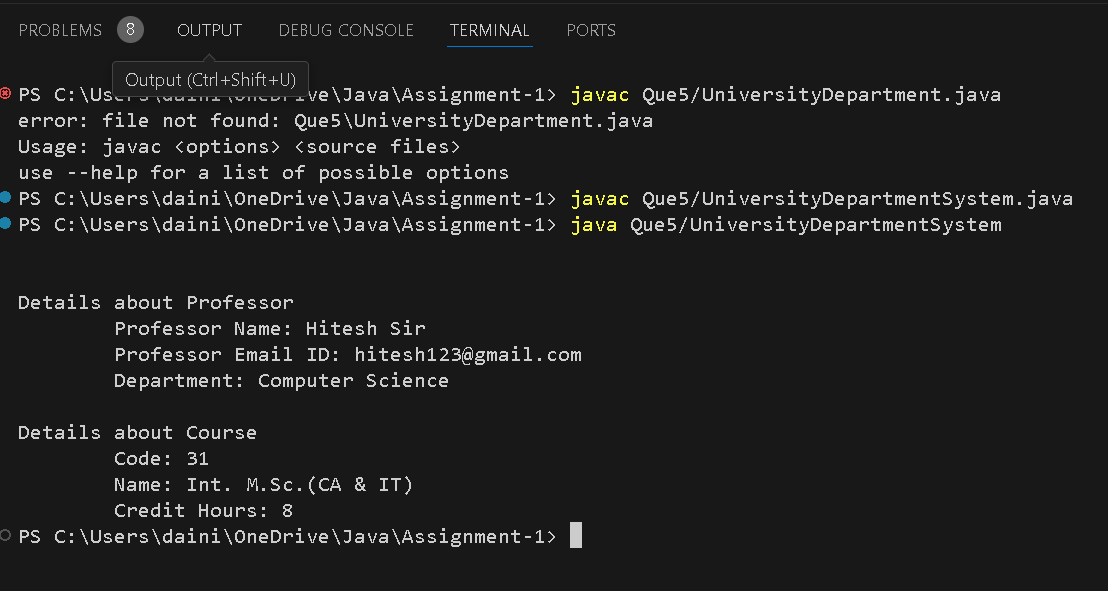
        System.out.println("\n\nDetails about Course");

        System.out.printf("\tCode: %d\n",C.getCode());

        System.out.printf("\tName: %s\n",C.getName());

        System.out.printf("\tCredit Hours: %d\n",C.getCreditHours());

    }

}

**Question-6**

package Que6;

class Pattern {

    public static void main(String args[]) {

        int k = 8;

        for (int i = 1; i < 7; i++) {

            for (int j = 1; j < 7; j++) {

                if (j > i) {

                    System.out.print(" " + (k - i));

                } else {

                    System.out.print(" " + (k - j));

                }

            }

            for (int j = 7; j >0; j--) {

                if (j > i) {

                    System.out.print(" " + (k - i));

                } else {

                    System.out.print(" " + (k - j));

                }

            }

            System.out.println("");

        }

        k = 8;

        for (int i = 7; i >0; i--) {

            for (int j = 1; j < 7; j++) {

                if (j > i) {

                    System.out.print(" " + (k - i));

                } else {

                    System.out.print(" " + (k - j));

                }

            }

            for (int j = 7; j >0; j--) {

                if (j > i) {

                    System.out.print(" " + (k - i));

                } else {

                    System.out.print(" " + (k - j));

                }

            }

            System.out.println("");

        }

    }

}

**Question-7**

package Que7;

import java.util.\*;

class GCD {

    public static void main(String args[]) {

        int num1, num2;

        Scanner S = new Scanner(System.in);

        System.out.print("Enter the Number1: ");

        num1 = S.nextInt();

        System.out.print("Enter the Number2: ");

        num2 = S.nextInt();

        int n1 = num1, n2 = num2;

        int gcd = 1;

        for (int i = 2; i < 10; i++) {

            while (n1 % i == 0 && n2 % i == 0) {

                n1 /= i;

                n2 /= i;

                gcd \*= i;

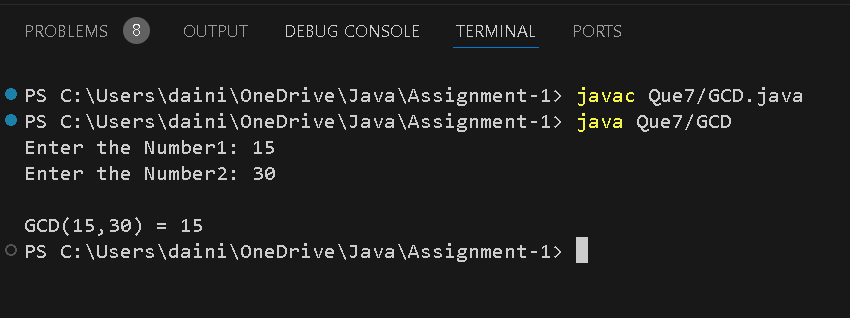
            }

        }

        System.out.printf("\nGCD(%d,%d) = %d", num1, num2, gcd);

    }

}



**Question-8**

package Que8;

class AddMatrix {

    public static void main(String args[]) {

        int[][] matrix1 = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

        int[][] matrix2 = {{9, 8, 7}, {6, 5, 4}, {3, 2, 1}};

        int[][] matrixSum = new int[3][3];

        for (int i = 0; i < 3; i++) {

            for (int j = 0; j < 3; j++) {

                matrixSum[i][j] = matrix1[i][j] + matrix2[i][j];

            }

        }

        System.out.println("\nMatrix1\t\tMatrix2");

        for (int i = 0; i < 3; i++) {

            for (int j = 0; j < 3; j++) {

                System.out.print(" "+matrix1[i][j]);

            }

            System.out.print("\t\t");

            for (int j = 0; j < 3; j++) {

                System.out.print(" "+matrix2[i][j]);

            }

            System.out.println("");

        }

        System.out.println("\nSum of the Matrix1 & Matrix2...");

        for (int i = 0; i < 3; i++) {

            for (int j = 0; j < 3; j++) {

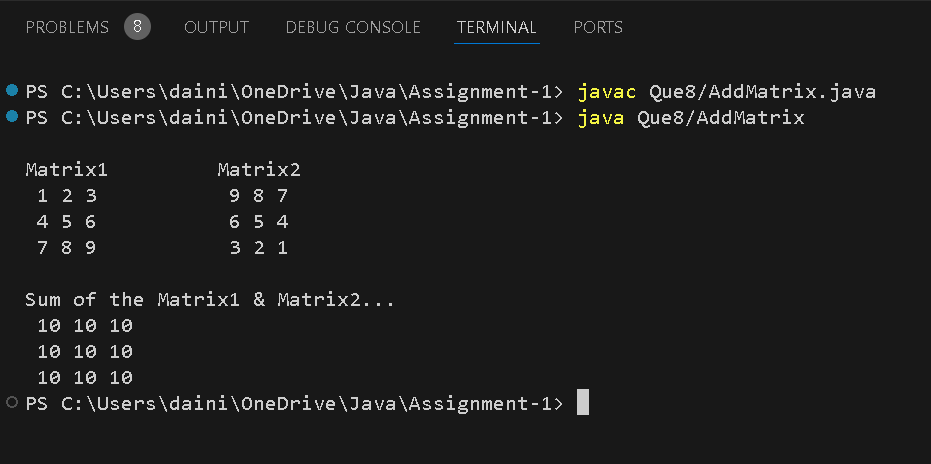
                System.out.print(" "+matrixSum[i][j]);

            }

            System.out.println("");

        }

    }

}

**Question-9**

package Que9;

import java.util.\*;

class Prime {

    boolean checkPrime(int num, int count) {

        if(count==1)

        {

            return true;

        }

        return (num%count==0)?false:checkPrime(num,count-1);

    }

    public static void main(String rags[]) {

        Prime P = new Prime();

        Scanner S = new Scanner(System.in);

        System.out.print("Enter the Number: ");

        int num = S.nextInt();

        int count = (int) Math.sqrt(num);

        if (P.checkPrime(num, count)) {

            System.out.printf("\n%d is a Prime number.", num);

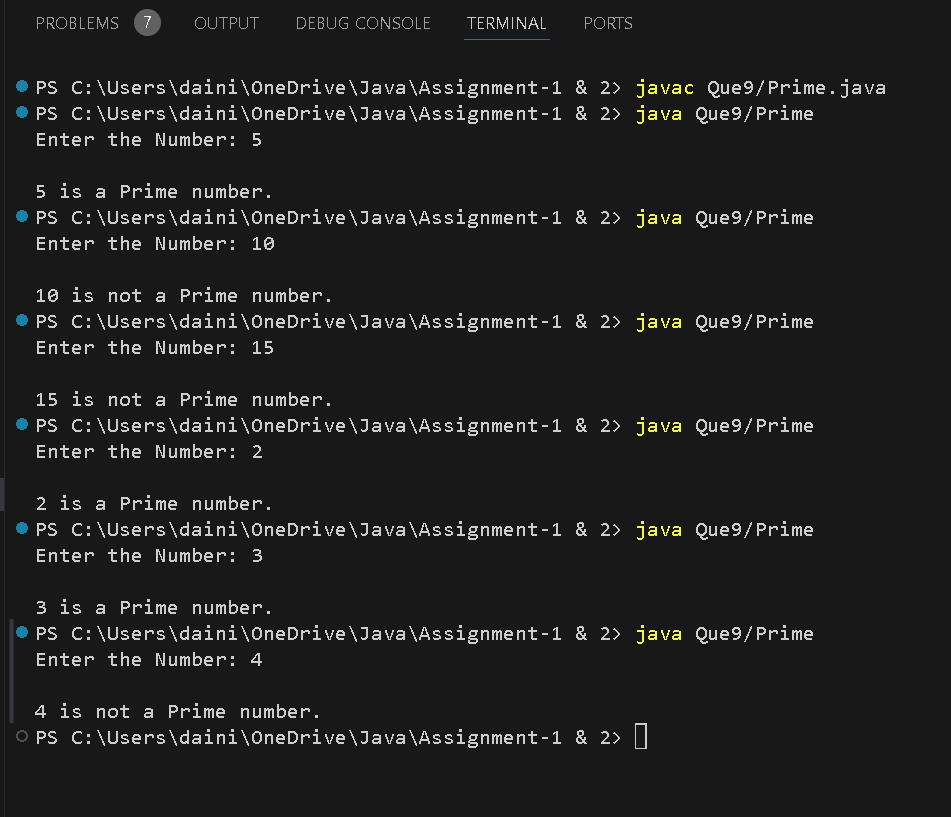
        } else {

            System.out.printf("\n%d is not a Prime number.", num);

        }

    }

}



**Question-10**

package Que10;

class StringBufferExample {

    public static void main(String args[])

    {

        StringBuffer stringBuffer = new StringBuffer("Hello, World!");

        System.out.println("\nOriginal String: "+stringBuffer);

        //! append Method

        System.out.println("\n1) stringBuffer.append(\" Welcome to Java!\"):  "+stringBuffer.append(" Welcome to Java!"));

        //! insert Method

        System.out.println("\n2) stringBuffer.insert(12,\"from \"):  "+stringBuffer.insert(12, "from "));

        //! replace Method

        System.out.println("\n3) stringBuffer.replace(7,12,\"Universe\"):  "+stringBuffer.replace(7, 12, "Universe"));

        //! setCharAt Method

        stringBuffer.setCharAt(0, 'h'); //\* It not returns any value \*/

        System.out.println("\n4) stringBuffer.setCharAt(0, 'h'):  "+stringBuffer);

        //! delete Method

        System.out.println("\n5) stringBuffer.delete(2,5):  "+stringBuffer.delete(2, 5));

        //! deleteCharAt()

        System.out.println("\n6) stringBuffer.deleteCharAt(10):  "+stringBuffer.deleteCharAt(10));

        //! reverse Method

        System.out.println("\n7) stringBuffer.reverse():  "+stringBuffer.reverse());

        //! toString Method

        System.out.println("\n8) stringBuffer.toString():  "+stringBuffer.toString());

        //! length Method

        System.out.println("\n9) stringBuffer.length():  "+stringBuffer.length());

        //! length Method

        System.out.println("\n10) stringBuffer.capacity():  "+stringBuffer.capacity()+"\n\n");

    }

}

