

Week 6

Question 1 : Design an abstract class having two methods. Create Rectangle and Triangle classes by inheriting the shape class and override the above methods to suitably implement for Rectangle and Triangle class.

Source Code :

```

import java.util.Scanner;
abstract class Shape {
    abstract double calculateArea();
    abstract double calculatePerimeter();}
class Rectangle extends Shape {
    private double length;
    private double width;
    public Rectangle(double length, double width) {
        this.length = length;
        this.width = width;}
    double calculateArea() {
        return length * width;}
    double calculatePerimeter() {
        return 2 * (length + width);}}
class Triangle extends Shape {
    private double side1;
    private double side2;
    private double side3;
    public Triangle(double side1, double side2, double side3) {
        this.side1 = side1;
        this.side2 = side2;
        this.side3 = side3;}
    double calculateArea() {
        // Using Heron's formula to calculate the area of a triangle
        double s = (side1 + side2 + side3) / 2;
        return Math.sqrt(s * (s - side1) * (s - side2) * (s - side3));}
    double calculatePerimeter() {
        return side1 + side2 + side3;}}
public class shape_triangle_rectangle {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the dimensions of the rectangle (length and width):");
        double length = sc.nextDouble();
        double width = sc.nextDouble();
        Rectangle r = new Rectangle(length, width);
        System.out.println("Area of the rectangle: " + r.calculateArea());
        System.out.println("Perimeter of the rectangle: " + r.calculatePerimeter());
        System.out.println("Enter the dimensions of the triangle (side1, side2, side3):");
        double side1 = sc.nextDouble();
        double side2 = sc.nextDouble();
        double side3 = sc.nextDouble();
        Triangle t = new Triangle(side1, side2, side3);
        System.out.println("Area of the triangle: " + t.calculateArea());
        System.out.println("Perimeter of the triangle: " + t.calculatePerimeter());
        sc.close();}}
```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\javac shape_triangle_rectangle.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\java shape_triangle_rectangle
Enter the dimensions of the rectangle (length and width):
8
9
Area of the rectangle: 72.0
Perimeter of the rectangle: 34.0
Enter the dimensions of the triangle (side1, side2, side3):
6
7
8
Area of the triangle: 20.33316256758894
Perimeter of the triangle: 21.0

```

Question 2 : Write a program in Java to illustrate the use of interface in Java

Source Code :

```

interface Animal {
    void makeSound();}
class Dog implements Animal {
```

```

public void makeSound() {
    System.out.println("Dog barks: Woof! Woof!");
}
class Cat implements Animal {
    public void makeSound() {
        System.out.println("Cat meows: Meow! Meow!");
    }
}
public class interface_eg {
    public static void main(String[] args) {
        Dog dog = new Dog();
        Cat cat = new Cat();
        dog.makeSound();
        cat.makeSound();
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac interface_eg.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java interface_eg
Dog barks: Woof! Woof!
Cat meows: Meow! Meow!

```

Question 3 : Create a general class ThreeDObject and derive the classes Box, Cube, Cylinder and Cone from it. The class ThreeDObject has methods wholeSurfaceArea() and volume(). Override these two methods in each of the derived classes to calculate the volume and whole surface area of each type of three-dimensional objects. The dimensions of the objects are to be taken from the users and passed through the respective constructors of each derived class. Write a main method to test these classes.

Source Code :

```

import java.util.Scanner;
class ThreeDObject {
    public double wholeSurfaceArea() {
        return 0;
    }
    public double volume() {
        return 0;
    }
}
class Box extends ThreeDObject {
    private double length;
    private double width;
    private double height;
    public Box(double length, double width, double height) {
        this.length = length;
        this.width = width;
        this.height = height;
    }
    public double wholeSurfaceArea() {
        return 2 * ((length * width) + (length * height) + (width * height));
    }
    public double volume() {
        return length * width * height;
    }
}
class Cube extends ThreeDObject {
    private double side;
    public Cube(double side) {
        this.side = side;
    }
    public double wholeSurfaceArea() {
        return 6 * side * side;
    }
    public double volume() {
        return side * side * side;
    }
}
class Cylinder extends ThreeDObject {
    private double radius;
    private double height;
    public Cylinder(double radius, double height) {
        this.radius = radius;
        this.height = height;
    }
    public double wholeSurfaceArea() {
        return 2 * Math.PI * radius * (radius + height);
    }
    public double volume() {
        return Math.PI * radius * radius * height;
    }
}
class Cone extends ThreeDObject {
    private double radius;
    private double height;
    public Cone(double radius, double height) {
        this.radius = radius;
        this.height = height;
    }
    public double wholeSurfaceArea() {
        double slantHeight = Math.sqrt(radius * radius + height * height);
        return Math.PI * radius * (radius + slantHeight);
    }
    public double volume() {
        return (1.0 / 3.0) * Math.PI * radius * radius * height;
    }
}

```

```

public class three_object {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the dimensions of the box (length, width, height):");
        double length = sc.nextDouble();
        double width = sc.nextDouble();
        double height = sc.nextDouble();
        Box box = new Box(length, width, height);
        System.out.println("Volume of the box: " + box.volume());
        System.out.println("Whole surface area of the box: " + box.wholeSurfaceArea());
        System.out.println("\nEnter the side of the cube:");
        double side = sc.nextDouble();
        Cube cube = new Cube(side);
        System.out.println("Volume of the cube: " + cube.volume());
        System.out.println("Whole surface area of the cube: " + cube.wholeSurfaceArea());
        System.out.println("\nEnter the dimensions of the cylinder (radius, height):");
        double radius = sc.nextDouble();
        height = sc.nextDouble();
        Cylinder cylinder = new Cylinder(radius, height);
        System.out.println("Volume of the cylinder: " + cylinder.volume());
        System.out.println("Whole surface area of the cylinder: " + cylinder.wholeSurfaceArea());
        System.out.println("\nEnter the dimensions of the cone (radius, height):");
        radius = sc.nextDouble();
        height = sc.nextDouble();
        Cone cone = new Cone(radius, height);
        System.out.println("Volume of the cone: " + cone.volume());
        System.out.println("Whole surface area of the cone: " + cone.wholeSurfaceArea());
        sc.close(); }}
```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>javac three_object.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>java three_object
Enter the dimensions of the box (length, width, height):
4
3
6
Volume of the box: 72.0
Whole surface area of the box: 108.0

Enter the side of the cube:
4
Volume of the cube: 64.0
Whole surface area of the cube: 96.0

Enter the dimensions of the cylinder (radius, height):
5
3
Volume of the cylinder: 235.61944901923448
Whole surface area of the cylinder: 251.32741228718345

Enter the dimensions of the cone (radius, height):
2
6
Volume of the cone: 25.132741228718345
Whole surface area of the cone: 52.304723677543585

```

Question 4 : Write a program to create a class named Vehicle having protected instance variables regnNumber, speed, color, ownerName and a method showData () to show “This is a vehicle class”. Inherit the Vehicle class into subclasses named Bus and Car having individual private instance variables routeNumber in Bus and manufacturerName in Car and both of them having showData () method showing all details of Bus and Car respectively with content of the super class’s showData () method.

Source Code :

```

import java.util.Scanner;
class Vehicle {
    protected String regnNumber;
    protected int speed;
    protected String color;
    protected String ownerName;
    public Vehicle(String regnNumber, int speed, String color, String ownerName) {
        this.regnNumber = regnNumber;
        this.speed = speed;
        this.color = color;
        this.ownerName = ownerName;}
    protected void showData() {
        System.out.println("This is a vehicle class");
        System.out.println("Registration Number: " + regnNumber);
        System.out.println("Speed: " + speed);
```

```

        System.out.println("Color: " + color);
        System.out.println("Owner Name: " + ownerName);}}
```

```

class Bus extends Vehicle {
    private String routeNumber;
    public Bus(String regnNumber, int speed, String color, String ownerName, String routeNumber) {
        super(regnNumber, speed, color, ownerName);
        this.routeNumber = routeNumber;}
    protected void showData() {
        super.showData();
        System.out.println("Route Number: " + routeNumber);}}
```

```

class Car extends Vehicle {
    private String manufacturerName;
    public Car(String regnNumber, int speed, String color, String ownerName, String manufacturerName) {
        super(regnNumber, speed, color, ownerName);
        this.manufacturerName = manufacturerName;}
    protected void showData() {
        super.showData();
        System.out.println("Manufacturer Name: " + manufacturerName);}}
```

```

public class vehicle_bus_car {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter vehicle details:");
        System.out.print("Registration Number: ");
        String regnNumber = sc.nextLine();
        System.out.print("Speed: ");
        int speed = sc.nextInt();
        sc.nextLine(); // Consume newline
        System.out.print("Color: ");
        String color = sc.nextLine();
        System.out.print("Owner Name: ");
        String ownerName = sc.nextLine();
        System.out.println("Enter Bus details:");
        System.out.print("Route Number: ");
        String routeNumber = sc.nextLine();
        Bus bus = new Bus(regnNumber, speed, color, ownerName, routeNumber);
        System.out.println("\nEnter Car details:");
        System.out.print("Manufacturer Name: ");
        String manufacturerName = sc.nextLine();
        Car car = new Car(regnNumber, speed, color, ownerName, manufacturerName);
        System.out.println("\nBus Details:");
        bus.showData();
        System.out.println("\nCar Details:");
        car.showData();
        sc.close(); }}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac vehicle_bus_car.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java vehicle_bus_car
Enter vehicle details:
Registration Number: 8976
Speed: 50
Color: red
Owner Name: sahin
Enter Bus details:
Route Number: 30c

Enter Car details:
Manufacturer Name: Tata

Bus Details:
This is a vehicle class
Registration Number: 8976
Speed: 50
Color: red
Owner Name: sahin
Route Number: 30c

Car Details:
This is a vehicle class
Registration Number: 8976
Speed: 50
Color: red
Owner Name: sahin
Manufacturer Name: Tata
```

Question 5 : Create three interfaces, each with two methods. Inherit a new interface from the three, adding a new method. Create a class by implementing the new interface and also inheriting from a concrete class. Now write four methods, each of which takes one of the four interfaces as an argument. In main (), create an object of your class and pass it to each of the methods.

Source Code :

```

interface Interface1 {
    void method1a();
    void method1b();
}
interface Interface2 {
    void method2a();
    void method2b();
}
interface Interface3 {
    void method3a();
    void method3b();
}
interface CombinedInterface extends Interface1, Interface2, Interface3 {
    void newMethod();
}
class MyClass extends SomeConcreteClass implements CombinedInterface {
    public void method1a() {
        System.out.println("Implementation of method1a");
    }
    public void method1b() {
        System.out.println("Implementation of method1b");
    }
    public void method2a() {
        System.out.println("Implementation of method2a");
    }
    public void method2b() {
        System.out.println("Implementation of method2b");
    }
    public void method3a() {
        System.out.println("Implementation of method3a");
    }
    public void method3b() {
        System.out.println("Implementation of method3b");
    }
    public void newMethod() {
        System.out.println("Implementation of newMethod");
    }
}
class SomeConcreteClass {}
public class interface4s {
    public static void methodWithInterface1(Interface1 obj) {
        System.out.println("Method with Interface1 argument");
        obj.method1a();
        obj.method1b();
    }
    public static void methodWithInterface2(Interface2 obj) {
        System.out.println("Method with Interface2 argument");
        obj.method2a();
        obj.method2b();
    }
    public static void methodWithInterface3(Interface3 obj) {
        System.out.println("Method with Interface3 argument");
        obj.method3a();
        obj.method3b();
    }
    public static void methodWithCombinedInterface(CombinedInterface obj) {
        System.out.println("Method with CombinedInterface argument");
        obj.method1a();
        obj.method1b();
        obj.method2a();
        obj.method2b();
        obj.method3a();
        obj.method3b();
        obj.newMethod();
    }
    public static void main(String[] args) {
        MyClass myObj = new MyClass();
        methodWithInterface1(myObj);
        methodWithInterface2(myObj);
        methodWithInterface3(myObj);
        methodWithCombinedInterface(myObj);
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac interface4s.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java interface4s
Method with Interface1 argument
Implementation of method1a
Implementation of method1b
Method with Interface2 argument
Implementation of method2a
Implementation of method2b
Method with Interface3 argument
Implementation of method3a
Implementation of method3b
Method with CombinedInterface argument
Implementation of method1a
Implementation of method1b
Implementation of method2a
Implementation of method2b
Implementation of method3a
Implementation of method3b
Implementation of newMethod

```

Question 6 : Create an interface Department containing attributes deptName and deptHead. It also has abstract methods for printing the attributes. Create a class hostel containing hostelName, hostelLocation and numberofRooms. The class contains methods for getting and printing the attributes. Then write Student class extending the Hostel class and implementing the Department interface. This class contains attributes studentName, regdNo, electiveSubject and avgMarks. Write suitable getData and printData methods for this class. Also implement the abstract methods of the Department interface. Write a driver class to test the Student class. The program should be menu driven containing the options: i) Admit new student ii) Migrate a student iii) Display details of a student For the third option a search is to be made on the basis of the entered registration number.

Source Code :

```
import java.util.*;
interface department {
    String deptName = "";
    String deptHead = "";
    abstract void Department();
}
class hostel {
    String hostelName, hostelLocation;
    int numberofRooms;
    public void setHostelName(String hostelName) {
        this.hostelName = hostelName;
    }
    public void setHostelLocation(String hostelLocation) {
        this.hostelLocation = hostelLocation;
    }
    public void setNumberofRooms(int numberofRooms) {
        this.numberofRooms = numberofRooms;
    }
    public String getHostelName() {
        return hostelName;
    }
    public String getHostelLocation() {
        return hostelLocation;
    }
    public int getNumberofRooms() {
        return numberofRooms;
    }
    void Hostel() {
        System.out.println("Hostel Name: " + hostelName);
        System.out.println("Hostel Location: " + hostelLocation);
        System.out.println("Number of Rooms: " + numberofRooms);
    }
}
class student extends hostel implements department {
    String studentName, electiveSubject, deptName, deptHead, hostelName, hostelLocation;
    int regdNo, numberofRooms;
    double avgMarks;
    public void setStudentName(String studentName) {
        this.studentName = studentName;
    }
    public void setElectiveSubject(String electiveSubject) {
        this.electiveSubject = electiveSubject;
    }
    public void setRegdNo(int regdNo) {
        this.regdNo = regdNo;
    }
    public void setAvgMarks(double avgMarks) {
        this.avgMarks = avgMarks;
    }
    public void setDeptName(String deptName) {
        this.deptName = deptName;
    }
    public void setDeptHead(String deptHead) {
        this.deptHead = deptHead;
    }
    public String getStudentName() {
        return studentName;
    }
    public String getElectiveSubject() {
        return electiveSubject;
    }
    public int getregdNo() {
        return regdNo;
    }
    public double getAvgMarks() {
        return avgMarks;
    }
    public String getDeptName() {
        return deptName;
    }
    public String getDeptHead() {
        return deptHead;
    }
    public void setData(String studentName, String electiveSubject, int regdNo, double avgMarks, String deptHead,
        String deptName, String hostelName, String hostelLocation, int numberofRooms) {
        setStudentName(studentName);
        setElectiveSubject(electiveSubject);
        setRegdNo(regdNo);
        setAvgMarks(avgMarks);
        setDeptName(deptName);
        setDeptHead(deptHead);
        super.setHostelName(hostelName);
    }
}
```



```
System.out.println("Enter New Elective Subject: ");
migrant.setElectiveSubject(sc.next());
break;
case 'b':
System.out.println("Enter Registration Number: ");
migrant.setRegdNo(sc.nextInt());
break;
case 'c':
System.out.println("Enter New Average Marks: ");
migrant.setAvgMarks(sc.nextDouble());
break;
case 'd':
System.out.println("Enter New Department Name: ");
migrant.setDeptName(sc.next());
System.out.println("Enter New Department Head: ");
migrant.setDeptHead(sc.next());
break;}}
else {
break;}}
int newIndex = -1;
for (int i = 0; i < count; i++) {
if (ar[i] == null) {
newIndex = i;
break;}}
if (newIndex != -1) {
ar[newIndex] = migrant;
System.out.println("Student Migrated Successfully");
} else {
System.out.println("No Free Slots are Available Right Now");
}} else {
System.out.println("Student Not Found");
break;
}
case 3:
System.out.print("Enter Registration Number: ");
int r = sc.nextInt();
for (int i = 0; i < count; i++) {
if (ar[i].regdNo == r) {
ar[i].getData();
ar[i].Department();
break;}}
break;
}
case 4:
System.out.println("Exiting!");
System.exit(0);
break;}}}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac q6.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java q6
Enter Number of Students:
3
1. Admit New Student
2. Migrate a Student
3. Details of a Student
4. Exit
1
Enter Student Name:
Sahin
Enter Elective Subject:
Math
Enter Registration Number:
1233
Enter Average Marks:
75
Enter Department Name:
MCA
Enter Department Head:
KBH
Enter Hostel Name:
UEM Hostel
Enter Hostel Location:
Newtown
Enter Number of Rooms:
30
Student Added
1. Admit New Student
2. Migrate a Student
3. Details of a Student
4. Exit
2
Enter Registration Number of the Student:
1233
Do you want to Change any other Details of this Student?
1. Yes
2. No
1
a. Elective Subject
b. Registration Number
c. Average Marks
d. Department
d
Enter New Department Name:
Bengali
Enter New Department Head:
KJG
Do you want to Change any other Details of this Student?
1. Yes
2. No
2
Student Migrated Successfully
1. Admit New Student
2. Migrate a Student
3. Details of a Student
4. Exit
3
Enter Registration Number: 1233
Student Name: Sahin
Elective Subject: Math
Registration Number: 1233
Average Marks: 75.0
Hostel Name: UEM Hostel
Hostel Location: Newtown
```

```
Number of Rooms: 30
Department Name: Bengali
Department Head: KJG
1. Admit New Student
2. Migrate a Student
3. Details of a Student
4. Exit
4
Exiting!
```

Question 7 : Create an interface called Player. The interface has an abstract method called play() that displays a message describing the meaning of “play” to the class. Create classes called Child, Musician, and Actor that all implement Player. Create an application that demonstrates the use of the classes(UsePlayer.java)

Source Code :

```
import java.util.Scanner;
interface Player {
    void play();
}
class Child implements Player {
    public void play() {
        System.out.println("For a child, playing means having fun and enjoying activities.");
    }
}
class Musician implements Player {
    public void play() {
        System.out.println("For a musician, playing means performing music with instruments or vocals.");
    }
}
class Actor implements Player {
    public void play() {
        System.out.println("For an actor, playing means performing roles in theater, film, or television.");
    }
}
public class player_child_musician_actor {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Choose a player to learn about their definition of 'play':");
        System.out.println("1. Child");
        System.out.println("2. Musician");
        System.out.println("3. Actor");
        System.out.print("Enter your choice: ");
        int choice = sc.nextInt();
        sc.nextLine();
        Player player;
        switch (choice) {
            case 1:
```

```

player = new Child();
break;
case 2:
    player = new Musician();
    break;
case 3:
    player = new Actor();
    break;
default:
    System.out.println("Invalid choice.");
    return;
}
player.play(); }}
```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac player_child_musician_actor.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java player_child_musician_actor
Choose a player to learn about their definition of 'play':
1. Child
2. Musician
3. Actor
Enter your choice: 1
For a child, playing means having fun and enjoying activities.

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java player_child_musician_actor
Choose a player to learn about their definition of 'play':
1. Child
2. Musician
3. Actor
Enter your choice: 2
For a musician, playing means performing music with instruments or vocals.

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java player_child_musician_actor
Choose a player to learn about their definition of 'play':
1. Child
2. Musician
3. Actor
Enter your choice: 3
For an actor, playing means performing roles in theater, film, or television.
```

Question 8 : Create an abstract class Accounts with the following details: Data Members: (a) Balance (b) accountNumber (c) accountHoldersName (d) address Methods: (a) withdrawl()- abstract (b) deposit()- abstract (c) display() to show the balance of the account number Create a subclass of this class SavingsAccount and add the following details: Data Members: (a) rateOfInterest Methods: (a) calculateAount()

Source Code :

```

import java.util.Scanner;
abstract class Accounts {
    protected double balance;
    protected int accountNumber;
    protected String accountHolderName;
    protected String address;
    public Accounts(int accountNumber, String accountHolderName, String address) {
        this.accountNumber = accountNumber;
        this.accountHolderName = accountHolderName;
        this.address = address;
        this.balance = 0; }
    public abstract void withdrawal(double amount);
    public abstract void deposit(double amount);
    public void display() {
        System.out.println("Account Number: " + accountNumber);
        System.out.println("Account Holder's Name: " + accountHolderName);
        System.out.println("Address: " + address);
        System.out.println("Current Balance: " + balance); } }
class SavingsAccount extends Accounts {
    private double rateOfInterest;
    public SavingsAccount(int accountNumber, String accountHolderName, String address, double rateOfInterest) {
        super(accountNumber, accountHolderName, address);
        this.rateOfInterest = rateOfInterest; }
    public void calculateAmount() {
        double interest = balance * (rateOfInterest / 100);
        balance += interest; }
    public void withdrawal(double amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println("Withdrawal successful. Remaining balance: " + balance); }
        else {
            System.out.println("Insufficient funds."); } }
    public void deposit(double amount) {
        balance += amount; }}
```

```

        System.out.println("Deposit successful. Current balance: " + balance);}}
```

```

public class account_savingsAccount {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter account number:");
        int accountNumber = scanner.nextInt();
        scanner.nextLine(); // Consume newline
        System.out.println("Enter account holder's name:");
        String accountHolderName = scanner.nextLine();
        System.out.println("Enter address:");
        String address = scanner.nextLine();
        System.out.println("Enter rate of interest:");
        double rateOfInterest = scanner.nextDouble();
        SavingsAccount savingsAccount = new SavingsAccount(accountNumber, accountHolderName, address, rateOfInterest);
        System.out.println("Choose an option:");
        System.out.println("1. Deposit");
        System.out.println("2. Withdraw");
        System.out.println("3. Display account details");
        System.out.println("4. Exit");
        while (true) {
            System.out.print("Enter your choice: ");
            int choice = scanner.nextInt();
            double amount;
            switch (choice) {
                case 1:
                    System.out.println("Enter amount to deposit:");
                    amount = scanner.nextDouble();
                    savingsAccount.deposit(amount);
                    break;
                case 2:
                    System.out.println("Enter amount to withdraw:");
                    amount = scanner.nextDouble();
                    savingsAccount.withdrawal(amount);
                    break;
                case 3:
                    savingsAccount.display();
                    break;
                case 4:
                    System.out.println("Exiting...");
                    System.exit(0);
                default:
                    System.out.println("Invalid choice.");
            }
        }
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>javac account_savingsAccount.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>java account_savingsAccount
Enter account number:
898
Enter account holder's name:
Sahin
Enter address:
Baguiati
Enter rate of interest:
5
Choose an option:
1. Deposit
2. Withdraw
3. Display account details
4. Exit
Enter your choice: 1
Enter amount to deposit:
10000
Deposit successful. Current balance: 10000.0
Enter your choice: 3
Account Number: 898
Account Holder's Name: Sahin
Address: Baguiati
Current Balance: 10000.0
Enter your choice: 2
Enter amount to withdraw:
4000
Withdrawal successful. Remaining balance: 6000.0

```

Question 9 : Create an abstract class MotorVehicle with the following details: Data Members: (a) modelName (b)modelNumber (c) modelPrice Methods: (a) display() to show all the details Create a subclass of this class Carthat inherits the class MotorVehicle and add the following details: Data Members: (b) discountRate Methods: (a) display() method to display the Car name, model number, price and the discount rate. (b) discount() method to compute the discount

Source Code :

```
import java.util.Scanner;
```

```

abstract class MotorVehicle {
    protected String modelName;
    protected int modelNumber;
    protected double modelPrice;
    public MotorVehicle(String modelName, int modelNumber, double modelPrice) {
        this.modelName = modelName;
        this.modelNumber = modelNumber;
        this.modelPrice = modelPrice;
    }
    abstract void display();
}
class Car extends MotorVehicle {
    private double discountRate;
    public Car(String modelName, int modelNumber, double modelPrice, double discountRate) {
        super(modelName, modelNumber, modelPrice);
        this.discountRate = discountRate;
    }
    void display() {
        System.out.println("Car Name: " + modelName);
        System.out.println("Model Number: " + modelNumber);
        System.out.println("Price: $" + modelPrice);
        System.out.println("Discount Rate: " + discountRate + "%");
    }
    public double discount() {
        return modelPrice * (discountRate / 100);
    }
}
public class motorvehicle_carthat {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Car details:");
        System.out.print("Model Name: ");
        String modelName = sc.nextLine();
        System.out.print("Model Number: ");
        int modelNumber = sc.nextInt();
        System.out.print("Model Price: $");
        double modelPrice = sc.nextDouble();
        System.out.print("Discount Rate (%): ");
        double discountRate = sc.nextDouble();
        Car car = new Car(modelName, modelNumber, modelPrice, discountRate);
        System.out.println("\nCar Details:");
        car.display();
        System.out.println("Discount: $" + car.discount());
        sc.close();
    }
}

```

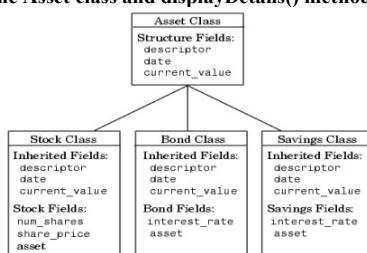
Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>javac motorvehicle_carthat.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>java motorvehicle_carthat
Enter Car details:
Model Name: Tata
Model Number: 8765
Model Price: $10000
Discount Rate (%): 5
Car Details:
Car Name: Tata
Model Number: 8765
Price: $10000.0
Discount Rate: 5.0%
Discount: $500.0

```

Question 10 : Implement the below Diagram. Here, Asset class is an abstract class containing an abstract method displayDetails() method. Stock, bond and Savings class inherit the Asset class and displayDetails() method is defined in every class.

**Source Code :**

```

import java.util.Scanner;
abstract class Asset {
    protected String descriptor;
    protected String date;
    protected double currentValue;
    public abstract void displayDetails();
}

```

```
class Stock extends Asset {  
    private int numShares;  
    private double sharePrice;  
    public Stock(String descriptor, String date, double currentValue, int numShares, double sharePrice) {  
        this.descriptor = descriptor;  
        this.date = date;  
        this.currentValue = currentValue;  
        this.numShares = numShares;  
        this.sharePrice = sharePrice;  
    }  
    public void displayDetails() {  
        System.out.println("Stock Details:");  
        System.out.println("Descriptor: " + descriptor);  
        System.out.println("Date: " + date);  
        System.out.println("Current Value: " + currentValue);  
        System.out.println("Number of Shares: " + numShares);  
        System.out.println("Share Price: " + sharePrice);  
    }  
}  
class Bond extends Asset {  
    private double interestRate;  
    public Bond(String descriptor, String date, double currentValue, double interestRate) {  
        this.descriptor = descriptor;  
        this.date = date;  
        this.currentValue = currentValue;  
        this.interestRate = interestRate;  
    }  
    public void displayDetails() {  
        System.out.println("Bond Details:");  
        System.out.println("Descriptor: " + descriptor);  
        System.out.println("Date: " + date);  
        System.out.println("Current Value: " + currentValue);  
        System.out.println("Interest Rate: " + interestRate);  
    }  
}  
class Savings extends Asset {  
    private double interestRate;  
    public Savings(String descriptor, String date, double currentValue, double interestRate) {  
        this.descriptor = descriptor;  
        this.date = date;  
        this.currentValue = currentValue;  
        this.interestRate = interestRate;  
    }  
    public void displayDetails() {  
        System.out.println("Savings Details:");  
        System.out.println("Descriptor: " + descriptor);  
        System.out.println("Date: " + date);  
        System.out.println("Current Value: " + currentValue);  
        System.out.println("Interest Rate: " + interestRate);  
    }  
}  
public class asset_stock_bond_saving {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.println("Enter stock details:");  
        System.out.print("Descriptor: ");  
        String stockDescriptor = sc.nextLine();  
        System.out.print("Date: ");  
        String stockDate = sc.nextLine();  
        System.out.print("Current Value: ");  
        double stockValue = sc.nextDouble();  
        System.out.print("Number of Shares: ");  
        int numShares = sc.nextInt();  
        System.out.print("Share Price: ");  
        double sharePrice = sc.nextDouble();  
        Stock stock = new Stock(stockDescriptor, stockDate, stockValue, numShares, sharePrice);  
        System.out.println("\nEnter bond details:");  
        System.out.print("Descriptor: ");  
        sc.nextLine(); // Consume newline character  
        String bondDescriptor = sc.nextLine();  
        System.out.print("Date: ");  
        String bondDate = sc.nextLine();  
        System.out.print("Current Value: ");  
        double bondValue = sc.nextDouble();  
        System.out.print("Interest Rate: ");  
        double interestRate = sc.nextDouble();  
        Bond bond = new Bond(bondDescriptor, bondDate, bondValue, interestRate);  
        System.out.println("\nEnter savings details:");  
        System.out.print("Descriptor: ");  
    }  
}
```

```

sc.nextLine(); // Consume newline character
String savingsDescriptor = sc.nextLine();
System.out.print("Date: ");
String savingsDate = sc.nextLine();
System.out.print("Current Value: ");
double savingsValue = sc.nextDouble();
System.out.print("Interest Rate: ");
double savingsInterestRate = sc.nextDouble();
Savings savings = new Savings(savingsDescriptor, savingsDate, savingsValue, savingsInterestRate);
System.out.println("\nDisplaying Details:");
stock.displayDetails();
bond.displayDetails();
savings.displayDetails();
sc.close();}}
```

Output :

```

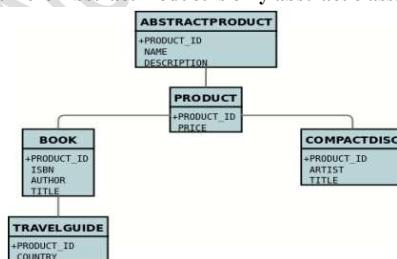
C:\Users\nayak\OneDrive\Desktop\mca assigments\sem2\java\code\week6\week6>javac asset_stock_bond_saving.java
C:\Users\nayak\OneDrive\Desktop\mca assigments\sem2\java\code\week6\week6>java asset_stock_bond_saving
Enter stock details:
Descriptor: Sahin
Date: 12 march
Current Value: 10000
Number of Shares: 5
Share Price: 100

Enter bond details:
Descriptor: sahin
Date: 12 feb
Current Value: 200
Interest Rate: 5

Enter savings details:
Descriptor: Sahin
Date: 3 july
Current Value: 100
Interest Rate: 5

Displaying Details:
Stock Details:
Descriptor: Sahin
Date: 12 march
Current Value: 10000.0
Number of Shares: 5
Share Price: 100.0
Bond Details:
Descriptor: sahin
Date: 12 feb
Current Value: 200.0
Interest Rate: 5.0
Savings Details:
Descriptor: Sahin
Date: 3 july
Current Value: 100.0
Interest Rate: 5.0

```

Question 11 : . Implement the below Diagram. Here AbstractProduct is only abstract class.**Source Code :**

```

import java.util.Scanner;
abstract class AbstractProduct {
    protected String productId;
    protected String name;
    protected String description;
    public AbstractProduct(String productId, String name, String description) {
        this.productId = productId;
        this.name = name;
        this.description = description;
    }
    public abstract void display();
}
class Product extends AbstractProduct {
    public Product(String productId, String name, String description) {
        super(productId, name, description);
    }
    public void display() {
        System.out.println("Product ID: " + productId);
        System.out.println("Name: " + name);
    }
}
```

```

        System.out.println("Description: " + description);}}
class Book extends Product {
    private String isbn;
    private String author;
    public Book(String productId, String name, String description, String isbn, String author) {
        super(productId, name, description);
        this.isbn = isbn;
        this.author = author;
    }
    public void display() {
        super.display();
        System.out.println("ISBN: " + isbn);
        System.out.println("Author: " + author);}}
class CompactDisc extends Product {
    private String artist;
    private String title;
    public CompactDisc(String productId, String name, String description, String artist, String title) {
        super(productId, name, description);
        this.artist = artist;
        this.title = title;
    }
    public void display() {
        super.display();
        System.out.println("Artist: " + artist);
        System.out.println("Title: " + title);}}
class TravelGuide extends Book {
    private String country;
    public TravelGuide(String productId, String name, String description, String isbn, String author, String country) {
        super(productId, name, description, isbn, author);
        this.country = country;
    }
    public void display() {
        super.display();
        System.out.println("Country: " + country);}}
public class abstractproduct_product_book_compactdisc_travelguide {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Travel Guide details:");
        System.out.print("Product ID: ");
        String productId = sc.nextLine();
        System.out.print("Name: ");
        String name = sc.nextLine();
        System.out.print("Description: ");
        String description = sc.nextLine();
        System.out.print("ISBN: ");
        String isbn = sc.nextLine();
        System.out.print("Author: ");
        String author = sc.nextLine();
        System.out.print("Country: ");
        String country = sc.nextLine();
        TravelGuide travelGuide = new TravelGuide(productId, name, description, isbn, author, country);
        System.out.println("\nTravel Guide details:");
        travelGuide.display();
        sc.close();}}

```

Output :

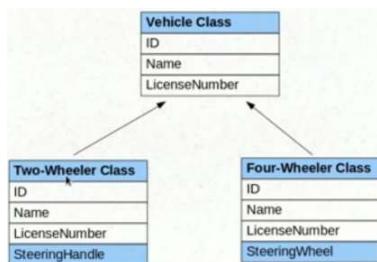
```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac abstractproduct_product_book_compactdisc_travelguide.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java abstractproduct_product_book_compactdisc_travelguide
Enter Travel Guide details:
Product ID: 8768
Name: Marie
Description: Biscuit
ISBN: 7266
Author: Chetan Bhagat
Country: India

Travel Guide details:
Product ID: 8768
Name: Marie
Description: Biscuit
ISBN: 7266
Author: Chetan Bhagat
Country: India

```

Question 12 :

**Source Code :**

```

import java.util.Scanner;
class Vehicle {
    private int ID;
    private String name;
    private String licenseNumber;
    public Vehicle(int ID, String name, String licenseNumber) {
        this.ID = ID;
        this.name = name;
        this.licenseNumber = licenseNumber;
    }
    public int getID() {
        return ID;
    }
    public String getName() {
        return name;
    }
    public String getLicenseNumber() {
        return licenseNumber;
    }
}
class TwoWheeler extends Vehicle {
    private String steeringHandle;
    public TwoWheeler(int ID, String name, String licenseNumber, String steeringHandle) {
        super(ID, name, licenseNumber);
        this.steeringHandle = steeringHandle;
    }
    public String getSteeringHandle() {
        return steeringHandle;
    }
}
class FourWheeler extends Vehicle {
    private String steeringWheel;
    public FourWheeler(int ID, String name, String licenseNumber, String steeringWheel) {
        super(ID, name, licenseNumber);
        this.steeringWheel = steeringWheel;
    }
    public String getSteeringWheel() {
        return steeringWheel;
    }
}
public class vehicle_2_4 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Two-Wheeler details:");
        System.out.print("ID: ");
        int twoWheelerID = sc.nextInt();
        sc.nextLine();
        System.out.print("Name: ");
        String twoWheelerName = sc.nextLine();
        System.out.print("License Number: ");
        String twoWheelerLicenseNumber = sc.nextLine();
        System.out.print("Steering Handle: ");
        String twoWheelerSteeringHandle = sc.nextLine();
        TwoWheeler twoWheeler = new TwoWheeler(twoWheelerID, twoWheelerName, twoWheelerLicenseNumber,
        twoWheelerSteeringHandle);
        System.out.println("\nEnter Four-Wheeler details:");
        System.out.print("ID: ");
        int fourWheelerID = sc.nextInt();
        sc.nextLine();
        System.out.print("Name: ");
        String fourWheelerName = sc.nextLine();
        System.out.print("License Number: ");
        String fourWheelerLicenseNumber = sc.nextLine();
        System.out.print("Steering Wheel: ");
        String fourWheelerSteeringWheel = sc.nextLine();
        FourWheeler fourWheeler = new FourWheeler(fourWheelerID, fourWheelerName, fourWheelerLicenseNumber,
        fourWheelerSteeringWheel);
        System.out.println("\nTwo-Wheeler Details:");
    }
}
  
```

```

System.out.println("ID: " + twoWheeler.getID());
System.out.println("Name: " + twoWheeler.getName());
System.out.println("License Number: " + twoWheeler.getLicenseNumber());
System.out.println("Steering Handle: " + twoWheeler.getSteeringHandle());
System.out.println("\nFour-Wheeler Details:");
System.out.println("ID: " + fourWheeler.getID());
System.out.println("Name: " + fourWheeler.getName());
System.out.println("License Number: " + fourWheeler.getLicenseNumber());
System.out.println("Steering Wheel: " + fourWheeler.getSteeringWheel());
sc.close();}}
```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>javac vehicle_2_4.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>java vehicle_2_4
Enter Two-Wheeler details:
ID: 5353
Name: sahin
License Number: 5252
Steering Handle: auto

Enter Four-Wheeler details:
ID: 8766
Name: Rohit
License Number: 7368
Steering Wheel: auto

Two-Wheeler Details:
ID: 5353
Name: sahin
License Number: 5252
Steering Handle: auto

Four-Wheeler Details:
ID: 8766
Name: Rohit
License Number: 7368
Steering Wheel: auto

```

Question 13 : Write a program to implement the Multiple Inheritance (Bank Interface, Customer & Account classes).**Source Code :**

```

import java.util.Scanner;
interface Bank {
    void deposit(double amount);
    void withdraw(double amount);}
class Customer {
    private String name;
    private String address;
    private String phoneNumber;
    public Customer(String name, String address, String phoneNumber) {
        this.name = name;
        this.address = address;
        this.phoneNumber = phoneNumber;}
    public String getName() {
        return name;}
    public String getAddress() {
        return address;}
    public String getPhoneNumber() {
        return phoneNumber;}}
class Account implements Bank {
    private String accountNumber;
    private double balance;
    public Account(String accountNumber) {
        this.accountNumber = accountNumber;
        this.balance = 0.0;}
    public String getAccountNumber() {
        return accountNumber;}
    public double getBalance() {
        return balance;}
    public void deposit(double amount) {
        balance += amount;
        System.out.println(amount + " deposited successfully.");}
    public void withdraw(double amount) {
        if (balance >= amount) {
            balance -= amount;
            System.out.println(amount + " withdrawn successfully.");
        } else {
            System.out.println("Insufficient funds.");}}}
```

```

public class multiple_inheritance_bank_customer_account {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter Customer Details:");
        System.out.print("Name: ");
        String name = scanner.nextLine();
        System.out.print("Address: ");
        String address = scanner.nextLine();
        System.out.print("Phone Number: ");
        String phoneNumber = scanner.nextLine();
        Customer customer = new Customer(name, address, phoneNumber);
        System.out.println("\nEnter Account Details:");
        System.out.print("Account Number: ");
        String accountNumber = scanner.nextLine();
        Account account = new Account(accountNumber);
        System.out.println("\nPerform Transactions:");
        while (true) {
            System.out.println("\n1. Deposit");
            System.out.println("2. Withdraw");
            System.out.println("3. Exit");
            System.out.print("Enter your choice: ");
            int choice = scanner.nextInt();
            double amount;
            switch (choice) {
                case 1:
                    System.out.print("Enter amount to deposit: ");
                    amount = scanner.nextDouble();
                    account.deposit(amount);
                    break;
                case 2:
                    System.out.print("Enter amount to withdraw: ");
                    amount = scanner.nextDouble();
                    account.withdraw(amount);
                    break;
                case 3:
                    System.out.println("Thank you!");
                    scanner.close();
                    return;
                default:
                    System.out.println("Invalid choice. Please try again.");
            }
        }
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>javac multiple_inheritance_bank_customer_account.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6\week6>java multiple_inheritance_bank_customer_account
Enter Customer Details:
Name: Sahin
Address: baguiati
Phone Number: 9087654321

Enter Account Details:
Account Number: 908635429

Perform Transactions:
1. Deposit
2. Withdraw
3. Exit
Enter your choice: 1
Enter amount to deposit: 10000
10000.0 deposited successfully.

1. Deposit
2. Withdraw
3. Exit
Enter your choice: 2
Enter amount to withdraw: 6000
6000.0 withdrawn successfully.

1. Deposit
2. Withdraw
3. Exit
Enter your choice: 3
Thank you!

```

Question 14 : Write a program to implement the Multiple Inheritance (Gross Interface, Employee & Salary classes).**Source Code :**

```

import java.util.Scanner;
interface Gross {
    double calculateGross();
}
interface Employee {

```

```

void getEmployeeDetails();
class Salary implements Gross, Employee {
    private String name;
    private double basicSalary;
    private double allowances;
    public void getEmployeeDetails() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter employee name: ");
        name = sc.nextLine();
        System.out.print("Enter basic salary: ");
        basicSalary = sc.nextDouble();
        System.out.print("Enter allowances: ");
        allowances = sc.nextDouble();
    }
    public double calculateGross() {
        return basicSalary + allowances;
    }
    public void displayDetails() {
        System.out.println("\nEmployee Name: " + name);
        System.out.println("Basic Salary: " + basicSalary);
        System.out.println("Allowances: " + allowances);
        System.out.println("Gross Salary: " + calculateGross());
    }
}
public class multiple_inheritance_gross {
    public static void main(String[] args) {
        Salary employee = new Salary();
        employee.getEmployeeDetails();
        employee.displayDetails();
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac multiple_inheritance_gross.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java multiple_inheritance_gross
Enter employee name: Sahin
Enter basic salary: 25000
Enter allowances: 5000

Employee Name: Sahin
Basic Salary: 25000.0
Allowances: 5000.0
Gross Salary: 30000.0

```

Question 15 : Program to create a interface 'Mango' and implement it in classes 'Winter' and 'Summer'.**Source Code :**

```

interface Mango {
    void displayType();
}
class Winter implements Mango {
    public void displayType() {
        System.out.println("This is a Winter Mango.");
    }
}
class Summer implements Mango {
    public void displayType() {
        System.out.println("This is a Summer Mango.");
    }
}
public class mango_summer_winter {
    public static void main(String[] args) {
        Winter winterMango = new Winter();
        Summer summerMango = new Summer();
        winterMango.displayType();
        summerMango.displayType();
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac mango_summer_winter.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java mango_summer_winter
This is a Winter Mango.
This is a Summer Mango.

```

Question 16 : Program to implement the Multiple Inheritance (Exam Interface, Student & Result classes).**Source Code :**

```

import java.util.Scanner;
interface Exam {
    void conductExam();
}
class Student {
    private String name;
    private int rollNumber;
    public Student(String name, int rollNumber) {
        this.name = name;
    }
}

```

```

        this.rollNumber = rollNumber;
    public void displayDetails() {
        System.out.println("Student Name: " + name);
        System.out.println("Roll Number: " + rollNumber);}}
```

class Result implements Exam {

```

    private int marks;
    public void conductExam() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter marks obtained: ");
        marks = sc.nextInt();}
    public void displayResult() {
        System.out.println("Marks Obtained: " + marks);}}
```

public class student_exam_result {

```

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter student name: ");
        String name = sc.nextLine();
        System.out.print("Enter roll number: ");
        int rollNumber = sc.nextInt();
        Student student = new Student(name, rollNumber);
        System.out.println("\nConducting exam for student...");
```

Result result = new Result();

```

        result.conductExam();
        System.out.println("\nStudent Details:");
        student.displayDetails();
        System.out.println("\nExam Result:");
        result.displayResult();
        sc.close();}}
```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac student_exam_result.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java student_exam_result
Enter student name: Sahin
Enter roll number: 53

Conducting exam for student...
Enter marks obtained: 89

Student Details:
Student Name: Sahin
Roll Number: 53

Exam Result:
Marks Obtained: 89
```

Question 17 : Program to demonstrate use of hierarchical inheritance using interface.**Source Code :**

```

interface Animal {
    void eat();}
interface Mammal extends Animal {
    void walk();}
interface Reptile extends Animal {
    void crawl();}
class Dog implements Mammal {
    public void eat() {
        System.out.println("Dog is eating.");}
    public void walk() {
        System.out.println("Dog is walking.");}}
class Snake implements Reptile {
    public void eat() {
        System.out.println("Snake is eating.");}
    public void crawl() {
        System.out.println("Snake is crawling.");}}
```

public class hierachial_inheritance {

```

    public static void main(String[] args) {
        Dog dog = new Dog();
        Snake snake = new Snake();
        dog.eat();
        snake.eat();
        dog.walk();
        snake.crawl();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac hierachial_inheritance.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java hierachial_inheritance
Dog is eating.
Snake is eating.
Dog is walking.
Snake is crawling.
```

Question 18 : Java program to Perform Payroll Using Interface (Multiple Inheritance).**Source Code :**

```
import java.util.Scanner;
interface Payable {
    double calculateBasicSalary();
}
interface Taxable {
    double calculateTax();
}
class Employee implements Payable, Taxable {
    private double basicSalary;
    private double taxRate;
    public Employee(double basicSalary, double taxRate) {
        this.basicSalary = basicSalary;
        this.taxRate = taxRate;
    }
    public double calculateBasicSalary() {
        return basicSalary;
    }
    public double calculateTax() {
        return basicSalary * (taxRate / 100);
    }
}
public class multiple_inheritance_payroll_emp {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the employee's basic salary: ");
        double basicSalary = scanner.nextDouble();
        System.out.print("Enter the tax rate (in percentage): ");
        double taxRate = scanner.nextDouble();
        Employee employee = new Employee(basicSalary, taxRate);
        System.out.println("Basic Salary: $" + employee.calculateBasicSalary());
        System.out.println("Tax Amount: $" + employee.calculateTax());
        scanner.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>javac multiple_inheritance_payroll_emp.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week6\week6>java multiple_inheritance_payroll_emp
Enter the employee's basic salary: 10000
Enter the tax rate (in percentage): 3
Basic Salary: $10000.0
Tax Amount: $300.0
```

Week 7

Question 1 : Write a Java program to show the use of all keywords for exception handling.**Source Code :**

```
public class Q_1 {
    public static void main(String[] args) {
        try {
            int result = divide(10, 0);
            System.out.println("Result: " + result);
        } catch (ArithmaticException e) {
            System.out.println("Exception occurred: Division by zero");
        } finally {
            System.out.println("This block always executes.");
        }
        throwExample();
    }
    public static int divide(int a, int b) throws ArithmaticException {
        if (b == 0) {
            throw new ArithmaticException("Division by zero");
        }
        return a / b;
    }
    public static void throwExample() {
        throw new NullPointerException("This exception is thrown intentionally.");
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_1.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_1
Exception occurred: Division by zero
This block always executes.
Exception in thread "main" java.lang.NullPointerException: This exception is thrown intentionally.
  at Q_1.throwExample(Q_1.java:16)
  at Q_1.main(Q_1.java:10)
```

Question 2 : Write a Java program using try and catch to generate NegativeArrayIndex Exception and Arithmetic Exception.**Source Code :**

```
public class Q_2 {
    public static void main(String[] args) {
        try {
            int[] arr = {1, 2, 3};
            int index = -1;
            int element = arr[index];
            System.out.println("Element at index " + index + ": " + element);
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("ArrayIndexOutOfBoundsException occurred: " + e.getMessage());
        }
        try {
            int result = 5 / 0;
            System.out.println("Result: " + result);
        } catch (ArithmaticException e) {
            System.out.println("ArithmaticException occurred: " + e.getMessage());
        }
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_2.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_2
ArrayIndexOutOfBoundsException occurred: Index -1 out of bounds for length 3
ArithmaticException occurred: / by zero
```

Question 3 : Define an exception called “NoMatchFoundException” that is thrown when a string is not equal to “University”. Write a program that uses this exception.**Source Code :**

```
import java.util.Scanner;
public class Q_3 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter a string: ");
        String inputStr = scanner.nextLine();
        scanner.close();
        try {
            checkString(inputStr);
            System.out.println("Match found!");
        } catch (NoMatchFoundException e) {
            System.out.println(e.getMessage());
        }
    }
    static void checkString(String inputStr) throws NoMatchFoundException {
        if (!inputStr.equals("University")) {
            throw new NoMatchFoundException("Input string does not match 'University'");
        }
    }
}
class NoMatchFoundException extends Exception {
    public NoMatchFoundException(String message) {
        super(message);
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_3.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_3
Enter a string:
Sahin
Input string does not match 'University'

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_3
Enter a string:
University
Match found!
```

Question 4 : Write a class that keeps a running total of all characters passed to it (one at a time) and throws anexception if it is passed a non-alphabetic character.**Source Code :**

```
public class Q_4 {
    private int totalAlphabeticCharacters;
    public Q_4() {
        this.totalAlphabeticCharacters = 0;
    }
}
```

```

public void addCharacter(char c) throws IllegalArgumentException {
    if (!Character.isLetter(c)) {
        throw new IllegalArgumentException("Non-alphabetic character passed: " + c);
    }
    totalAlphabeticCharacters++;
}
public int getTotalAlphabeticCharacters() {
    return totalAlphabeticCharacters;
}
public static void main(String[] args) {
    Q_4 counter = new Q_4();
    try {
        counter.addCharacter('a');
        counter.addCharacter('b');
        counter.addCharacter('1');
    } catch (IllegalArgumentException e) {
        System.out.println(e.getMessage());
    }
    System.out.println("Total alphabetic characters: " + counter.getTotalAlphabeticCharacters());
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_4.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_4
Non-alphabetic character passed: 1
Total alphabetic characters: 2

```

Question 5 : Write a program called Factorial.java that computes factorials and catches the result in an array of type long for reuse. The long type of variable has its own range. For example 20! Is as high as the range of long type. So check the argument passes and “throw an exception”, if it is too big or too small. • If x is less than 0 throw an IllegalArgumentException with a message “Value of x must be positive”. • If x is above the length of the array throw an IllegalArgumentException with a message “Result will overflow”. Here x is the value for which we want to find the factorial.

Source Code :

```

public class Q_5 {
    private static final int MAX_SIZE = 21;
    private static long[] factorialArray = new long[MAX_SIZE];
    static {
        factorialArray[0] = 1; // 0! = 1
        for (int i = 1; i < MAX_SIZE; i++) {
            factorialArray[i] = factorialArray[i - 1] * i;
        }
    }
    public static long computeFactorial(int x) {
        if (x < 0) {
            throw new IllegalArgumentException("Value of x must be positive");
        }
        if (x >= MAX_SIZE) {
            throw new IllegalArgumentException("Result will overflow");
        }
        return factorialArray[x];
    }
    public static void main(String[] args) {
        try {
            int x = 21;
            long result = computeFactorial(x);
            System.out.println("Factorial of " + x + " is: " + result);
        } catch (IllegalArgumentException e) {
            System.out.println("Exception occurred: " + e.getMessage());
        }
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_5.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_5
Exception occurred: Result will overflow

```

Question 7 : Write a program that outputs the name of the capital of the country entered at the command line. The program should throw a “NoMatchFoundException” when it fails to print the capital of the country entered at the command line.

Source Code :

```

import java.util.HashMap;
import java.util.Map;
public class Q_7 {
    public static void main(String[] args) {
        if (args.length != 1) {
            System.out.println("Please provide a country name as a command line argument.");
            return;
        }
        String countryName = args[0];
        Map<String, String> capitals = createCapitalMap();
        try {
            String capital = capitals.get(countryName);
            if (capital == null) {

```

```

        throw new NoMatchFoundException("No capital found for country: " + countryName);
        System.out.println("The capital of " + countryName + " is " + capital);
    } catch (NoMatchFoundException e) {
        System.out.println(e.getMessage());
    }
private static Map<String, String> createCapitalMap() {
    Map<String, String> capitals = new HashMap<>();
    capitals.put("France", "Paris");
    capitals.put("Germany", "Berlin");
    capitals.put("Italy", "Rome");
    capitals.put("Spain", "Madrid");
    return capitals;
}
public static class NoMatchFoundException extends Exception {
    public NoMatchFoundException(String message) {
        super(message);
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_7.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_7 Spain
The capital of Spain is Madrid

```

Question 8 : Write a program that takes a value at the command line for which factorial is to be computed. The program must convert the string to its integer equivalent. There are three possible user input errors that can prevent the program from executing normally.

- The first error is when the user provides no argument while executing the program and an `ArrayIndexOutOfBoundsException` is raised.
- You must write a catch block for this.
- The second error is `NumberFormatException` that is raised in case the user provides a non-integer (float double) value at the command line.
- The third error is `IllegalArgumentException`. This needs to be thrown manually if the value at the command line is 0.

Source Code :

```

public class Q_8 {
    public static void main(String[] args) {
        try {
            if (args.length == 0) {
                throw new ArrayIndexOutOfBoundsException("No argument provided. Please enter an integer.");
            }
            int number = Integer.parseInt(args[0]);
            if (number == 0) {
                throw new IllegalArgumentException("Factorial of 0 is not allowed.");
            }
            int factorial = 1;
            for (int i = 1; i <= number; i++) {
                factorial *= i;
            }
            System.out.println("Factorial of " + number + " is: " + factorial);
        } catch (ArrayIndexOutOfBoundsException e) {
            System.out.println("Error: " + e.getMessage());
        } catch (NumberFormatException e) {
            System.out.println("Error: Please enter a valid integer.");
        } catch (IllegalArgumentException e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_8.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_8 10
Factorial of 10 is: 3628800

```

Question 9 : Create a user-defined exception named `CheckArgument` to check the number of arguments passed through the command line. If the number of argument is less than 5, throw the `CheckArgumentexception`, else print the addition of all the five numbers.

Source Code :

```

class CheckArgument extends Exception {
    public CheckArgument(String message) {
        super(message);
    }
}
public class Q_9 {
    public static void main(String[] args) {
        try {
            if (args.length < 5) {
                throw new CheckArgument("Insufficient arguments. Please provide at least 5 numbers.");
            }
            int sum = 0;
            for (int i = 0; i < 5; i++) {
                sum += Integer.parseInt(args[i]);
            }
            System.out.println("The sum of the first five numbers is: " + sum);
        } catch (CheckArgument e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}

```

```
} catch (NumberFormatException e) {
    System.out.println("Error: Please ensure all arguments are integers.");}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_9.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_9 10 20 30 40 50
The sum of the first five numbers is: 150
```

Question 10 : Consider a Student examination database system that prints the mark sheet of students. Input the following from the command line. (a) Student's Name (b) Marks in six subjects These marks should be between 0 to 50. If the marks are not in the specified range, raise a RangeException, else find the total marks and prints the percentage of the students.

Source Code :

```
public class Q_10 {
    public static void main(String[] args) {
        try {
            if (args.length != 7) {
                throw new IllegalArgumentException("Please provide the student's name and marks for six subjects.");
            }
            String studentName = args[0];
            int totalMarks = 0;
            int marks;
            for (int i = 1; i <= 6; i++) {
                marks = Integer.parseInt(args[i]);
                if (marks < 0 || marks > 50) {
                    throw new RangeException("Marks for subject " + i + " are out of range.");
                }
                totalMarks += marks;
            }
            double percentage = (totalMarks / 300.0) * 100;
            System.out.println("Mark Sheet for " + studentName);
            System.out.println("Total Marks: " + totalMarks);
            System.out.println("Percentage: " + percentage + "%");
        } catch (NumberFormatException e) {
            System.out.println("Please enter valid integer marks.");
        } catch (RangeException e) {
            System.out.println(e.getMessage());
        } catch (Exception e) {
            System.out.println("An unexpected error occurred: " + e.getMessage());
        }
    }
    static class RangeException extends Exception {
        public RangeException(String message) {
            super(message);
        }
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_10 Sahin 30 20 45 40 25 49
Mark Sheet for Sahin
Total Marks: 209
Percentage: 69.66666666666667%
```

Question 11 : Write a java program to create a custom Exception that would handle at least 2 kind of Arithmetic Exceptions while calculating a given equation (e.g. $X+Y*(P/Q)Z-I$).

Source Code :

```
public class Q_11 {
    public static void main(String[] args) {
        try {
            int X = 10;
            int Y = 20;
            int P = 30;
            int Q = 0;
            int Z = 40;
            int I = 50;
            int result = calculateEquation(X, Y, P, Q, Z, I);
            System.out.println("The result of the equation is: " + result);
        } catch (ArithmaticExceptionHandler e) {
            System.out.println("Arithmatic Exception Occurred: " + e.getMessage());
        }
    }
    public static int calculateEquation(int X, int Y, int P, int Q, int Z, int I) throws ArithmaticExceptionHandler {
        if (Q == 0) {
            throw new ArithmaticExceptionHandler("Cannot divide by zero");
        }
        if (P % Q != 0) {
            throw new ArithmaticExceptionHandler("Invalid operation: Non-integer division");
        }
        return X + Y * (P / Q) * Z - I;
    }
    static class ArithmaticExceptionHandler extends Exception {
        public ArithmaticExceptionHandler(String message) {

```

```
super(message);}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_11.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_11
Arithmatic Exception Occurred: Cannot divide by zero
```

Question 12 : Create two user-defined exceptions named “TooHot” and “TooCold” to check the temperature (in Celsius) given by the user passed through the command line is too hot or too cold. • If temperature > 35, throw exception “TooHot”. • If temperature

Source Code :

```
public class Q_12 {
    public static void main(String[] args) {
        if (args.length != 1) {
            System.out.println("Please provide a temperature value in Celsius as a command line argument.");
            return;
        }
        double celsius;
        try {
            celsius = Double.parseDouble(args[0]);
        } catch (NumberFormatException e) {
            System.out.println("Invalid temperature format. Please enter a number.");
            return;
        }
        try {
            checkTemperature(celsius);
        } catch (TooHotException e) {
            System.out.println("Exception: " + e.getMessage());
        } catch (TooColdException e) {
            System.out.println("Exception: " + e.getMessage());
        }
        public static void checkTemperature(double celsius) throws TooHotException, TooColdException {
            if (celsius > 35) {
                throw new TooHotException("Temperature is too hot!");
            } else if (celsius < 5) {
                throw new TooColdException("Temperature is too cold!");
            } else {
                double fahrenheit = (celsius * 9 / 5) + 32;
                System.out.println("Normal temperature");
                System.out.printf("Temperature in Fahrenheit: %.2f\n", fahrenheit);
            }
        }
        public static class TooHotException extends Exception {
            public TooHotException(String message) {
                super(message);
            }
        }
        public static class TooColdException extends Exception {
            public TooColdException(String message) {
                super(message);
            }
        }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_12.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_12 35
Normal temperature
Temperature in Fahrenheit: 95.00
```

Question 13 : Consider an Employee recruitment system that prints the candidate name based on the age criteria. The name and age of the candidate are taken as Input.Create two user-defined exceptions named “TooOlder” and “TooYounger” • If age>45, throw exception “TooOlder”. • If age

Source Code :

```
import java.util.Scanner;
public class Q_13 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter candidate name: ");
        String name = scanner.nextLine();
        System.out.print("Enter candidate age: ");
        int age = scanner.nextInt();
        try {
            validateAge(age);
            System.out.println("Eligible candidate: " + name);
        } catch (TooOlderException e) {
            System.out.println("Exception: " + e.getMessage());
        } catch (TooYoungerException e) {
            System.out.println("Exception: " + e.getMessage());
        }
    }
    private void validateAge(int age) {
        if (age < 18) {
            throw new TooYoungerException("Candidate is too young");
        } else if (age > 45) {
            throw new TooOlderException("Candidate is too old");
        }
    }
}
```

```

public static void validateAge(int age) throws TooOlderException, TooYoungerException {
    if (age > 45) {
        throw new TooOlderException("Candidate is too old for the position.");
    } else if (age < 20) {
        throw new TooYoungerException("Candidate is too young for the position.");}
public static class TooOlderException extends Exception {
    public TooOlderException(String message) {
        super(message);}}
public static class TooYoungerException extends Exception {
    public TooYoungerException(String message) {
        super(message);}}}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_13.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_13
Enter candidate name: Sahin
Enter candidate age: 23
Eligible candidate: Sahin

```

Question 14 : Consider a “Binary to Decimal” Number conversion system which only accepts binary number as Input. If user provides a decimal number a custom Exception “WrongNumberFormatException” exception will be thrown. Otherwise, it will convert into decimal and print into the screen.

Source Code :

```

import java.util.Scanner;
public class Q_14 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a binary number: ");
        String binaryString = scanner.nextLine();
        try {
            int decimal = convertBinaryToDecimal(binaryString);
            System.out.println("Decimal equivalent: " + decimal);
        } catch (WrongNumberFormatException e) {
            System.out.println("Exception: " + e.getMessage());
        }
        scanner.close();}}
public static int convertBinaryToDecimal(String binaryString) throws WrongNumberFormatException {
    int decimal = 0;
    int base = 1;
    for (int i = binaryString.length() - 1; i >= 0; i--) {
        char digit = binaryString.charAt(i);
        if (digit != '0' && digit != '1') {
            throw new WrongNumberFormatException("Invalid binary digit. Only '0' and '1' allowed.");
        }
        int digitValue = Character.getNumericValue(digit);
        decimal += digitValue * base;
        base *= 2;}
    return decimal;}
public static class WrongNumberFormatException extends Exception {
    public WrongNumberFormatException(String message) {
        super(message);}}}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_14.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_14
Enter a binary number: 111
Decimal equivalent: 7

```

Question 15 : Write a Java Program that Implement the Nested Try Statements.

Source Code :

```

public class Q_15 {
    public static void main(String[] args) {
        try {
            int[] numbers = {1, 2, 3};
            System.out.println("Before exception is generated.");
            try {
                System.out.println(numbers[5]);
            } catch (ArrayIndexOutOfBoundsException e) {
                System.out.println("Array index is out of bounds: " + e);}
            System.out.println(5 / 0);
        } catch (ArithmaticException e) {
            System.out.println("Division by zero: " + e);}}}

```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_15.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_15
Before exception is generated.
Array index is out of bounds: java.lang.ArrayIndexOutOfBoundsException: Index 5 out of bounds for length 3
Division by zero: java.lang.ArithmaticException: / by zero
```

Question 16 : Write a Java Program to Create Account with 500 Rs Minimum Balance, Deposit Amount, Withdraw Amount and Also Throws LessBalanceException. • Java Program Which has a Class Called LessBalanceException Which returns the Statement that Says WithDraw Amount(_Rs) is Not Valid • Java Program that has a Class Which Creates 2 Accounts, Both Account Deposit Money and One Account Tries to WithDraw more Money Which Generates a LessBalanceException Take Appropriate Action for the Same.

Source Code :

```
class LessBalanceException extends Exception {
    public LessBalanceException(String message) {
        super(message);}}
class Account {
    private static final int MIN_BALANCE = 500;
    private int balance;
    public Account(int initialDeposit) throws LessBalanceException {
        if (initialDeposit < MIN_BALANCE) {
            throw new LessBalanceException("Initial deposit must be at least Rs " + MIN_BALANCE);}
        this.balance = initialDeposit;}
    public void deposit(int amount) {
        balance += amount;
        System.out.println("Deposited Rs " + amount + ". Current Balance: Rs " + balance);}
    public void withdraw(int amount) throws LessBalanceException {
        if (amount > balance) {
            throw new LessBalanceException("Withdrawal amount Rs " + amount + " is not valid. Current Balance: Rs " + balance);}
        balance -= amount;
        System.out.println("Withdrawn Rs " + amount + ". Current Balance: Rs " + balance);}
    public int getBalance() {
        return balance;}}
public class Q_16 {
    public static void main(String[] args) {
        try {
            Account account1 = new Account(1000);
            Account account2 = new Account(1500);
            account1.deposit(500);
            account1.withdraw(200);
            account2.deposit(400);
            account2.withdraw(2000);
        } catch (LessBalanceException e) {
            System.out.println(e.getMessage());}}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_16.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_16
Deposited Rs 500. Current Balance: Rs 1500
Withdrawn Rs 200. Current Balance: Rs 1300
Deposited Rs 400. Current Balance: Rs 1900
withdrawal amount Rs 2000 is not valid. Current Balance: Rs 1900
```

Question 17 : Consider a Library Management System, where a user wants to find a book. If the book is present in Library (Hint: Use predefined array), then it will print the book. Otherwise it will throw an exception “BookNotFoundException”.

Source Code :

```
class BookNotFoundException extends Exception {
    public BookNotFoundException(String message) {
        super(message);}}
class Library {
    private String[] availableBooks;
    public Library(String[] books) {
        this.availableBooks = books;}}
    public void findBook(String bookName) throws BookNotFoundException {
        for (String book : availableBooks) {
            if (book.equalsIgnoreCase(bookName)) {
                System.out.println("Book found: " + book);
                return;}}
        throw new BookNotFoundException("BookNotFoundException: " + bookName + " does not exist in the library.");}
    public class Q_17 {
```

```
public static void main(String[] args) {
    String[] books = {"The Alchemist", "The Da Vinci Code", "Harry Potter", "The Lord of the Rings"};
    Library library = new Library(books);
    try {
        library.findBook("The Da Vinci Code");
        library.findBook("The Great Gatsby");
    } catch (BookNotFoundException e) {
        System.out.println(e.getMessage());
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_17.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_17
Book found: The Da Vinci Code
BookNotFoundException: 'The Great Gatsby' does not exist in the library.
```

Question 18 : Consider a Quiz Management System, where a user needs to answer 5 questions. If any of the answer is wrong, throw an exception “NotCorrectException”. If the answer is correct give a message “good! The answer is correct”.

Source Code :

```
import java.util.Scanner;
class NotCorrectException extends Exception {
    public NotCorrectException(String message) {
        super(message);
    }
}
class QuizManagementSystem {
    private String[] questions = {"What is the capital of France?", "Who wrote 'Romeo and Juliet'", "What is the largest ocean on Earth?", "What is the result of 7 * 6?", "What year did World War II end?"};
    private String[] answers = {"Paris", "William Shakespeare", "Pacific", "42", "1945"};
    public void checkAnswer(int questionNumber, String answer) throws NotCorrectException {
        if (answers[questionNumber].equalsIgnoreCase(answer)) {
            System.out.println("Good! The answer is correct.");
        } else {
            throw new NotCorrectException("NotCorrectException: The answer is incorrect.");
        }
    }
    public void startQuiz() {
        for (int i = 0; i < questions.length; i++) {
            System.out.println((i + 1) + ". " + questions[i]);
            String userAnswer = getUserAnswer();
            try {
                checkAnswer(i, userAnswer);
            } catch (NotCorrectException e) {
                System.out.println(e.getMessage());
                break;
            }
        }
    }
    private String getUserAnswer() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Your answer: ");
        String answer = scanner.nextLine();
        return answer;
    }
}
public class Q_18 {
    public static void main(String[] args) {
        QuizManagementSystem quiz = new QuizManagementSystem();
        quiz.startQuiz();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_18.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_18
1. What is the capital of France?
Your answer: Paris
Good! The answer is correct.
2. Who wrote 'Romeo and Juliet'?
Your answer: Shakespear
NotCorrectException: The answer is incorrect.
```

Question 19 : Write a program to raise a user defined exception if username is less than 6 characters and password does not match.

Source Code :

```
import java.util.Scanner;
public class Q_19 {
    private static final String VALID_USERNAME = "Sahin_Nayak";
    private static final String VALID_PASSWORD = "Sahin@2024";
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter username: ");
        String username = scanner.nextLine();
        System.out.print("Enter password: ");
```

```

String password = scanner.nextLine();
try {
    authenticate(username, password);
    System.out.println("Authentication successful!");
} catch (InvalidCredentialsException e) {
    System.out.println(e.getMessage());
    scanner.close();
}
private static void authenticate(String username, String password) throws InvalidCredentialsException {
if (username.length() < 6) {
    throw new InvalidCredentialsException("Username must be at least 6 characters long.");
}
if (!username.equals(VALID_USERNAME) || !password.equals(VALID_PASSWORD)) {
    throw new InvalidCredentialsException("Invalid username or password.");
}
class InvalidCredentialsException extends Exception {
    public InvalidCredentialsException(String message) {
        super(message);
    }
}
Output :

```

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_19.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_19
Enter username: Sahin Nayak
Enter password: Sahin@2024
Authentication successful!

```

Question 20 : Write a program to accept a password from the user and throw 'Authentication Failure' exception if the password is incorrect.

Source Code :

```

import java.util.Scanner;
class AuthenticationFailureException extends Exception {
    public AuthenticationFailureException(String message) {
        super(message);
    }
}
public class Q_20 {
    private static final String CORRECT_PASSWORD = "Sahin@1234";
    public static void authenticate(String password) throws AuthenticationFailureException {
        if (!password.equals(CORRECT_PASSWORD)) {
            throw new AuthenticationFailureException("Authentication Failure: Incorrect password.");
        }
        System.out.println("Authentication Successful!");
    }
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter password: ");
        String password = scanner.nextLine();
        scanner.close();
        try {
            authenticate(password);
        } catch (AuthenticationFailureException e) {
            System.out.println(e.getMessage());
        }
    }
}
Output :

```

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_20.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_20
Enter password: Sahin@1234
Authentication Successful!

```

Question 21 : Write a program to input name and age of a person and throw a user-defined exception, if the entered age is negative.

Source Code :

```

import java.util.Scanner;
class NegativeAgeException extends Exception {
    public NegativeAgeException(String message) {
        super(message);
    }
}
public class Q_21 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter name: ");
        String name = scanner.nextLine();
        System.out.print("Enter age: ");
        int age = scanner.nextInt();
        try {
            validateAge(age);
            System.out.println("Name: " + name);
            System.out.println("Age: " + age);
        }
    }
    void validateAge(int age) {
        if (age < 0) {
            throw new NegativeAgeException("Age cannot be negative.");
        }
    }
}

```

```

} catch (NegativeAgeException e) {
    System.out.println(e.getMessage());
    scanner.close();
}
private static void validateAge(int age) throws NegativeAgeException {
    if (age < 0) {
        throw new NegativeAgeException("Age cannot be negative.");
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_21.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_21
Enter name: Sahin
Enter age: 23
Name: Sahin
Age: 23

```

Question 22 : Write a program to throw user defined exception if the given number is not positive.

Source Code :

```

class NotPositiveException extends Exception {
    public NotPositiveException(String message) {
        super(message);
    }
}
class Q_22 {
    public static void checkNumber(int number) throws NotPositiveException {
        if (number <= 0) {
            throw new NotPositiveException("Number is not positive.");
        } else {
            System.out.println(number + " is a positive number.");
        }
    }
    public static void main(String[] args) {
        try {
            checkNumber(-5);
        } catch (NotPositiveException e) {
            System.out.println("Caught Exception: " + e.getMessage());
        }
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>javac Q_22.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week7\Java_7>java Q_22
Caught Exception: Number is not positive.

```

Week 8

Question 1 : Write a Java program for calculating Factorial. Number should be taken through user input (Using Scanner, BufferedReader both).

Source Code :**Using Scanner Class :**

```

import java.util.Scanner;
public class q1 {
    public static int calculateFactorial(int n) {
        if (n == 0 || n == 1)
            return 1;
        return n * calculateFactorial(n - 1);
    }
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter the number : ");
        int number = scanner.nextInt();
        int fact = calculateFactorial(number);
        System.out.println(fact);
        scanner.close();
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q1.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q1
Enter the number :
5
120

```

Using Buffer Class :

```

import java.io.BufferedReader;
import java.io.InputStreamReader;
public class q1_2nd {
    public static int calculateFactorial(int n) {

```

```

if (n == 0 || n == 1)
    return 1;
return n * calculateFactorial(n - 1);
}
public static void main(String[] args) throws Exception {
    BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));
    System.out.println("Enter a number: ");
    String input = reader.readLine();
    int number = Integer.parseInt(input);
    int factorial = calculateFactorial(number);
    System.out.println(factorial);}}
```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q1_2nd.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q1_2nd
Enter a number:
4
24

```

Question 2 : Design a palindrome class that will input a string from console and check whether the string is palindrome or not.**Source Code :**

```

import java.util.Scanner;
class Palindrome {
    String str = "";
    Palindrome(String s) {
        this.str = s;}
    public boolean isPalindrome(String str) {
        int left = 0;
        int right = str.length() - 1;
        while (left < right) {
            if (str.charAt(left) != str.charAt(right)) {
                return false;}
            left++;
            right-;}
        return true;}}
public class q2 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String name = sc.nextLine();
        Palindrome pd = new Palindrome(name);
        if (pd.isPalindrome(name)) {
            System.out.println("Palindrome");}
        else {
            System.out.println("Not Palindrome");}
        sc.close();}}
```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q2.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q2
Enter a string :
Sahin
Not Palindrome
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q2
Enter a string :
amma
Palindrome

```

Question 3 : Write a Java program to merge two strings.**Source Code :**

```

import java.util.Scanner;
public class q3 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter two strings : ");
        String str1 = sc.nextLine();
        String str2 = sc.nextLine();
        String merged = str1 + str2;
        System.out.println(merged);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q3.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q3
Enter two strings :
Sahin
Nayak
SahinNayak
```

Question 4 : Write a Java program for reverse a string. (String will be taken as user input through console).**Source Code :**

```
import java.util.Scanner;
public class q4 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        String revStr = "";
        for (int i = str.length() - 1; i >= 0; i--) {
            revStr += str.charAt(i);
        }
        System.out.println(revStr);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q4.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q4
Enter a string :
Sahin
nihaiS
```

Question 5 : Write a Java Program to Concatenate Two Strings.**Source Code :**

```
import java.util.Scanner;
public class q5 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter two strings : ");
        String str1 = sc.nextLine();
        String str2 = sc.nextLine();
        StringBuilder sb = new StringBuilder();
        sb.append(str1);
        sb.append(str2);
        System.out.println(sb.toString());
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q5.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q5
Enter two strings :
Sahin
Nayak
SahinNayak
```

Question 6 : Write a Java Program to check if a Given String is getChar from Specific Index.**Source Code :**

```
import java.util.Scanner;
public class q6 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String str = sc.nextLine();
        int index = sc.nextInt();
        boolean charExists = checkChar(str, index);
        if (charExists) {
            System.out.println("The character at index " + index + " is " + str.charAt(index) + "'");
        } else {
            System.out.println("Character does not exist at index " + index);
        }
        sc.close();}
    public static boolean checkChar(String str, int index) {
        if (index < 0 || index >= str.length()) {
            return false;
        }
        return true;}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q6
Enter a string :
Sahin
Enter the index :
3
The character at index 3 is 'i'
```

Question 7 : Write a Java Program to Find the Length of the String.**Source Code :**

```
import java.util.Scanner;
public class q7 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        System.out.println("Length is " + str.length());
        sc.close(); }}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q7.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q7
Enter a string :
Sahin
Length is 5
```

Question 8 : Write a Java Program to Find All Possible Subsets of given Length in String.**Source Code :**

```
import java.util.Scanner;
public class q8 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter string and length");
        String str = sc.nextLine();
        int length = sc.nextInt();
        findSubsets(str, length);
        sc.close(); }
    public static void findSubsets(String str, int length) {
        for (int i = 0; i < str.length(); i++) {
            findSubsetsHelper(str, i, length, new StringBuilder());}
    private static void findSubsetsHelper(String str, int startIndex, int length, StringBuilder currentSubset) {
        if (currentSubset.length() == length) {
            System.out.println(currentSubset);
            return; }
        for (int i = startIndex; i < str.length(); i++) {
            currentSubset.append(str.charAt(i));
            findSubsetsHelper(str, i + 1, length, currentSubset);
            currentSubset.deleteCharAt(currentSubset.length() - 1); }}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q8.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q8
Enter string
Sahin
Enter string
3
Sah
Sai
San
Shi
Shn
Sin
ahi
ahn
ah
hin
ahi
ahn
ain
hin
hin
```

Question 9 : Write a Java Program to Remove the White Spaces from a String.**Source Code :**

```
import java.util.Scanner;
public class q9 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        str = str.trim();}
```

```
str = str.replaceAll("\s", "");
System.out.println(str);
sc.close();}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q9
Enter a string :
i am a boy
iamaboy
```

Question 10 : Write a Java Program to Compare two Strings.**Source Code :**

```
import java.util.Scanner;
public class q10 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter two strings : ");
        String str1 = sc.nextLine();
        String str2 = sc.nextLine();
        boolean areEqual = str1.equalsIgnoreCase(str2);
        System.out.println(areEqual);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q10
Enter two strings :
Sahin
Nayak
false
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q10
Enter two strings :
sahin
sahin
true
```

Question 12 : Write a Java Program to Use Equals Method In a String Class.**Question 13 : Write a Java Program to Use EqualsIgnoreCase Method In a String Class.****Source Code :**

```
import java.util.Scanner;
public class q12andq13 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter 1st String :");
        String str1 = sc.nextLine();
        System.out.println("Enter 2nd String :");
        String str2 = sc.nextLine();
        // Method 1: Using equals() method (case-sensitive)
        boolean areEqual = str1.equals(str2);
        System.out.println("Using equals(): " + str1 + " == " + str2 + " is " + areEqual);
        // Method 2: Using equalsIgnoreCase() method (case-insensitive)
        boolean areEqualIgnoreCase = str1.equalsIgnoreCase(str2);
        System.out.println("Using equalsIgnoreCase(): " + str1 + " == " + str2 + " is " + areEqualIgnoreCase);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q12andq13.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q12andq13
Enter 1st String :
sahin
Enter 2nd String :
Sahin
Using equals(): sahin == Sahin is false
Using equalsIgnoreCase(): sahin == Sahin is true
```

Question 14 : Write a Java Program to Use compareTo Method In a String Class.**Source Code :**

```
import java.util.Scanner;
public class q14 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter 1st String :");
        String str1 = sc.nextLine();
        System.out.println("Enter 2nd String :");
        String str2 = sc.nextLine();
        int result1 = str1.compareTo(str2);
```

```

int result2 = str2.compareTo(str1);
System.out.println("str1.compareTo(str2): " + result1);
System.out.println("str2.compareTo(str1): " + result2);
sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q14.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q14
Enter 1st String :
sahin
Enter 2nd String :
sahin
str1.compareTo(str2): 0
str2.compareTo(str1): 0
```

Question 15 : With a Java Program to Use compareToIgnoreCase Method In a String Class.**Source Code :**

```

import java.util.Scanner;
public class q15 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter 1st String");
        String str1 = sc.nextLine();
        System.out.println("Enter 2nd String");
        String str2 = sc.nextLine();
        int result1 = str1.compareToIgnoreCase(str2);
        int result2 = str2.compareToIgnoreCase(str1);
        System.out.println("str1.compareToIgnoreCase(str2): " + result1);
        System.out.println("str2.compareToIgnoreCase(str1): " + result2);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q15.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q15
Enter 1st String
sahin
Enter 2nd String
SAHIN
str1.compareToIgnoreCase(str2): 0
str2.compareToIgnoreCase(str1): 0
```

Question 16 : Write a Java Program to Replace Character or String.**Source Code :**

```

import java.util.Scanner;
public class q16 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        System.out.println("Enter an old and new char to change : ");
        String old = sc.next();
        String neww = sc.next();
        String replacedString = str.replace(old, neww);
        System.out.println("Original String: " + str);
        System.out.println("Replaced String: " + replacedString);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q16.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q16
Enter a string :
Sahon
Enter an old and new char to change :
o
i
Original String: Sahon
Replaced String: Sahin
```

Question 17 : Write a Java Program to Search Last Occurrence of a Substring Inside a Substring.**Source Code :**

```

import java.util.Scanner;
public class q17 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter string : ");
        String str = sc.nextLine();
        System.out.println("Enter sub string : ");
        String subStr = sc.nextLine();
```

```

int lastIndex = -1;
for (int i = str.length() - subStr.length(); i >= 0; i--) {
    boolean match = true;
    for (int j = 0; j < subStr.length(); j++) {
        if (str.charAt(i + j) != subStr.charAt(j)) {
            match = false;
            break;
        }
    }
    if (match) {
        lastIndex = i;
        break;
    }
}
if (lastIndex != -1) {
    System.out.println("Found at " + lastIndex);
} else {
    System.out.println("Not found");
}
sc.close();
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q17.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q17
Enter string :
nihis
Enter sub string :
i
Found at 3

```

Question 18 : Write a Java Program to Remove a Particular Character from a String**Source Code :**

```

import java.util.Scanner;
public class q18 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        System.out.println("Enter the character : ");
        String ch = sc.next();
        String replacedString = str.replace(ch, ""); // Replace with a space
        System.out.println("Original String: " + str);
        System.out.println("String after removing " + ch + ": " + replacedString);
        sc.close();
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q18.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q18
Enter a string :
nihis
Enter the character :
i
Original String: nihis
String after removing i: nhs

```

Question 19 : Write a Java Program to Replace a Substring Inside a String by Another One.**Source Code :**

```

import java.util.Scanner;
public class q19 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        System.out.println("Enter the string to be replaced and to replace : ");
        String subStrToBeReplaced = sc.next();
        String subStrToReplace = sc.next();
        String newStr = str.replace(subStrToBeReplaced, subStrToReplace);
        System.out.println(str);
        System.out.println(newStr);
        sc.close();
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q19.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q19
Enter a string :
goutamSadhukha
Enter the string to be replaced and to replace :
tamSadhu
rbose
goutamSadhukha
gouri_bosekha

```

Question 20 : Write a Java Program to Reverse a String.

Source Code :

```
import java.util.Scanner;
public class q20 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        String newStr = "";
        for (int i = str.length() - 1; i >= 0; i--) {
            newStr += str.charAt(i);}
        System.out.println(newStr);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q20.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q20
Enter a string :
Sahin
nihas
```

Question 21 : Write a Java Program to Search a Word Inside a String.**Source Code :**

```
import java.util.Scanner;
public class q21 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        System.out.println("Enter the word : ");
        String word = sc.next();
        int index = str.indexOf(word);
        if (index != -1)
            System.out.println("Found at " + index);
        else
            System.out.println("Not found");
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q21.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q21
Enter a string :
zedreya
Enter the word :
dre
Found at 2
```

Question 22 : Write a Java Program to Split a String into a Number of Substrings.**Source Code :**

```
import java.util.Scanner;
public class q22 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        String[] substrings = str.split("\s");
        System.out.println("Substrings:");
        for (String substring : substrings) {
            System.out.println(substring);}
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q22.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q22
Enter a string :
Sizuka Nobi
Substrings:
Sizuka
Nobi
```

Question 23 : Write a Java Program to Search a Particular Word in a String.**Source Code :**

```
import java.util.Scanner;
public class q23 {
```

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter a string : ");
    String str = sc.nextLine();
    String word = sc.next();
    int index = str.indexOf(word);
    if (index != -1) {
        System.out.println("Found at: " + index);
    } else {
        System.out.println("Not found");
    }
    sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q23.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q23
Enter a string :
Sizuka_Nobi
No
Found at: 7
```

Question 24 : Write a Java Program to Replace All Occurings of a String.**Source Code :**

```
import java.util.Scanner;
public class q24 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the word : ");
        String str = sc.nextLine();
        System.out.println("Enter the word that you want to replace : ");
        String word = sc.next();
        System.out.println("Enter the word by what you want to replace : ");
        String newWord = sc.next();
        String newStr = str.replace(word, newWord);
        System.out.println(str);
        System.out.println(newStr);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q24.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q24
Enter the word :
hello i am a student of school
Enter the word that you want to replace :
school
Enter the word by what you want to replace :
MCA
hello i am a student of school
hello i am a student of MCA
```

Question 25 : Write a Java Program to Make First Character of Each Word in Uppercase.**Source Code :**

```
import java.util.Scanner;
public class q25 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter String: ");
        String str = sc.nextLine();
        char[] charArray = str.toCharArray();
        boolean isWordStart = true;
        for (int i = 0; i < charArray.length; i++) {
            if (Character.isLetter(charArray[i])) {
                if (isWordStart) {
                    charArray[i] = Character.toUpperCase(charArray[i]);
                    isWordStart = false;
                } else {
                    isWordStart = true;
                }
            }
        }
        String modifiedString = new String(charArray);
        System.out.println("Original String: " + str);
        System.out.println("Modified String: " + modifiedString);
        sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q25.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q25
Enter String:
hello i am sahin nayak
Original String: hello i am sahin nayak
Modified String: Hello I Am Sahin Nayak
```

Question 26 : Write a Java Program to Delete All Repeated Words in String.**Source Code :**

```
import java.util.HashSet;
import java.util.Scanner;
public class q26 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter String: ");
        String str = sc.nextLine();
        String[] words = str.split("\\s+");
        HashSet<String> uniqueWords = new HashSet<>();
        StringBuilder modifiedString = new StringBuilder();
        for (String word : words) {
            if (uniqueWords.add(word)) {
                modifiedString.append(word).append(" ");
            }
        }
        System.out.println("Original String: " + str);
        System.out.println("String after removing duplicates: " + modifiedString.toString().trim());
        sc.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q26.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q26
Enter String:
hello i am a boy. i am a student. i am student of Mca.
Original String: hello i am a boy. i am a student. i am student of Mca.
String after removing duplicates: hello i am a boy. student. student of Mca.
```

Question 27 : Write a Java Program to Reverse the String Using Both Recursion and Iteration.**Source Code :**

```
import java.util.Scanner;
public class q27 {
    public static String reverseIteration(String s) {
        String newS = "";
        for (int i = s.length() - 1; i >= 0; i--) {
            newS += s.charAt(i);
        }
        return newS;
    }
    public static String reverseRecursion(String s) {
        if (s.isEmpty()) {
            return s;
        } else {
            return reverseRecursion(s.substring(1)) + s.charAt(0);
        }
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
        String str = sc.nextLine();
        String strReverseIteration = reverseIteration(str);
        String strReverseRecursion = reverseRecursion(str);
        System.out.println("Iteration : " + strReverseIteration);
        System.out.println("Recursion : " + strReverseRecursion);
        sc.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q27.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q27
Enter a string :
Joreamon
Iteration : nomaerod
Recursion : nomaerod
```

Question 28 : Write a Java Program to Convert a String Totally into Upper Case.**Source Code :**

```
import java.util.Scanner;
public class q28 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string : ");
```

```
String str = sc.nextLine();
str = str.toUpperCase();
System.out.println(str);
sc.close();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week 8\week 8>javac q28.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week 8\week 8>java q28
Enter a string :
sahin
SAHIN
```

Question 29 : Write a Java Program to Remove all Characters in Second String which are Present in First String.**Source Code :**

```
import java.util.HashSet;
import java.util.Scanner;
public class q29 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter First String : ");
        String str1 = sc.nextLine();
        System.out.println("Enter Second String : ");
        String str2 = sc.nextLine();
        String result = removeChars(str1, str2);
        System.out.println("Original String (First): " + str1);
        System.out.println("Original String (Second): " + str2);
        System.out.println("String after removing characters: " + result);
        sc.close();}
    public static String removeChars(String str1, String str2) {
        StringBuilder sb = new StringBuilder();
        HashSet<Character> charSet = new HashSet<>();
        for (char c : str1.toCharArray()) {
            charSet.add(c);}
        for (char c : str2.toCharArray()) {
            if (!charSet.contains(c)) {
                sb.append(c);}}
        return sb.toString();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week 8\week 8>java q29
Enter First String :
doreamon
Enter Second String :
doreamlyan
Original String (First): doreamon
Original String (Second): doreamlyan
String after removing characters: iy
```

Question 30 : Write a Java Program to Find the Consecutive Occurrence of any Vowel in a String.**Source Code :**

```
import java.util.Scanner;
public class q30 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a string");
        String str = sc.nextLine();
        System.out.println("Consecutive vowel substrings: " + findConsecutiveVowels(str));
        sc.close();}
    public static String findConsecutiveVowels(String str) {
        StringBuilder result = new StringBuilder();
        String vowels = "aeiouAEIOU";
        int startIndex = 0;
        for (int i = 0; i < str.length(); i++) {
            char ch = str.charAt(i);
            if (vowels.indexOf(ch) != -1) {
                if (i == 0 || !vowels.contains(String.valueOf(str.charAt(i - 1)))) {
                    startIndex = i;}
                } else if (startIndex != 0) {
                    result.append(str.substring(startIndex, i));
                    startIndex = 0;}}
            if (startIndex != 0) {
                result.append(str.substring(startIndex));}
            return result.toString().isEmpty() ? "No consecutive vowel substrings found." : result.toString();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q30.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q30
Enter a string
Sahin
Consecutive vowel substrings: ai
```

Question 31 : Write a Java Program to Find the Largest & Smallest Word in a String.**Source Code :**

```
import java.util.Scanner;
public class q31 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter a stdng : ");
        String str = sc.nextLine();
        findLargestSmallestWords(str);
        sc.close();
    }
    public static void findLargestSmallestWords(String str) {
        if (str.isEmpty()) {
            System.out.println("String is empty.");
            return;
        }
        String[] words = str.trim().split("\\s+");
        if (words.length == 1) {
            System.out.println("String contains only one word: " + words[0]);
            return;
        }
        String largestWord = words[0];
        String smallestWord = words[0];
        for (String word : words) {
            if (word.length() > largestWord.length()) {
                largestWord = word;
            } else if (word.length() < smallestWord.length()) {
                smallestWord = word;
            }
        }
        System.out.println("Largest word: " + largestWord);
        System.out.println("Smallest word: " + smallestWord);
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q31.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q31
Enter a stdng :
hello i am a live in kolkata
Largest word: kolkata
Smallest word: i
```

Question 32 : Write a Java Program to Find First and Last Occurrence of Given Character in a String.**Source Code :**

```
import java.util.Scanner;
public class q32 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string : ");
        String str = sc.nextLine();
        System.out.println("Enter the character : ");
        String ch = sc.next();
        int firstIndex = str.indexOf(ch);
        int lastIndex = str.lastIndexOf(ch);
        System.out.println("First index : " + firstIndex + ", " + "Last index : " + lastIndex);
        sc.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q32.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q32
Enter the string :
this is doreami
Enter the character :
i
First index : 2, Last index : 14
```

Question 33 : Write a Java Program to Display the Characters in Prime Position a Given String.**Source Code :**

```
import java.util.Scanner;
public class q33 {
```

```

public static boolean isPrime(int pos) {
    boolean flag = false;
    for (int i = 2; i <= pos / 2; i++) {
        if (pos % i == 0) {
            flag = true;
            break;
        }
    }
    if (!flag)
        return true;
    return false;
}
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter the string");
    String str = sc.nextLine();
    StringBuilder sb = new StringBuilder();
    System.out.println("Characters at prime position : ");
    for (int i = 0; i < str.length() - 1; i++) {
        if (isPrime(i))
            sb.append(str.charAt(i)).append(" ");
    }
    System.out.println(sb.toString());
    sc.close();
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q33.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q33
Enter the string
hello this is sahin
Characters at prime position :
 h e l l   h i   i

```

Question 34 : Write a Java Program to Sort String Ignoring Whitespaces and Repeating Characters Only Once.**Source Code :**

```

import java.util.HashSet;
import java.util.Scanner;
import java.util.Arrays;
public class q34 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string");
        String str = sc.nextLine();
        String sortedStr = sortUnique(str);
        System.out.println(sortedStr);
        sc.close();
    }
    public static String sortUnique(String str) {
        StringBuilder sb = new StringBuilder();
        HashSet<Character> uniqueChars = new HashSet<>();
        for (char ch : str.toCharArray()) {
            if (ch != ' ' && !uniqueChars.contains(ch)) {
                sb.append(ch);
                uniqueChars.add(ch);
            }
        }
        char[] charArray = sb.toString().toCharArray();
        Arrays.sort(charArray);
        return new String(charArray);
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q34.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q34
Enter the string
hello doreamon love dora cakes
acdehklmnorsv

```

Question 35 : Write a Java Program to Count Replace First Occurrence of a String.**Source Code :**

```

public class q35 {
    public static void main(String[] args) {
        String text = "This is a string with two occurrences of 'is'.";
        String oldStr = "is";
        String newStr = "was";
        int count = countReplaceFirst(text, oldStr, newStr);
        System.out.println("Original text: " + text);
        System.out.println("Replaced text: " + text);
        System.out.println("Number of replacements: " + count);
    }
}

```

```
public static int countReplaceFirst(String text, String oldStr, String newStr) {
    int count = text.length() - text.replace(oldStr, "").length();
    if (count > 0) {
        text = text.replaceFirst(oldStr, newStr);
    }
    return count;
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q35.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q35
Original text: This is a string with two occurrences of 'is'.
Replaced text: This is a string with two occurrences of 'is'.
Number of replacements: 6
```

Question 36 : Write a Java Program to Know the Last Index of a Particular Word in a String.**Source Code :**

```
import java.util.Scanner;
public class q36 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string : ");
        String str = sc.nextLine();
        System.out.println("Enter the word : ");
        String word = sc.next();
        int index = str.lastIndexOf(word);
        System.out.println(index);
        sc.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q36.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q36
Enter the string :
hello i am sahin
Enter the word :
n
15
```

Question 37 : Write a Java Program to Access the Index of the Character or String.**Source Code :**

```
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
public class q37 {
    public static int[] findAllIndices(String text, String searchValue) {
        List<Integer> indices = new ArrayList<>();
        int startIndex = 0;
        while (true) {
            int index = text.indexOf(searchValue, startIndex);
            if (index == -1) {
                break;
            }
            indices.add(index);
            startIndex = index + 1;
        }
        return indices.stream().mapToInt(Integer::intValue).toArray();
    }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string : ");
        String str = sc.nextLine();
        System.out.println("Enter the character : ");
        String ch = sc.next();
        int[] indices = findAllIndices(str, ch);
        for (int index : indices) {
            System.out.print(index + " ");
        }
        sc.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q37.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q37
Enter the string :
hello this is me
Enter the character :
i
8 11
```

Question 38 : Write a Java Program to Access the Characters or the ASCII of the Character Available in the String.**Question 39 : Write a Java Program to Display the Character and the Corresponding Ascii Present in the String.**

Source Code :

```
import java.util.Scanner;
public class q38 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the string : ");
        String str = sc.nextLine();
        for (int i = 0; i < str.length(); i++) {
            char character = str.charAt(i);
            int asciiValue = character;
            System.out.println("Character: " + character + ", ASCII : " + asciiValue);
        }
        sc.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q38.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q38
Enter the string :
sahin
Character: s, ASCII : 115
Character: a, ASCII : 97
Character: h, ASCII : 104
Character: i, ASCII : 105
Character: n, ASCII : 110
```

Question 40 : Write a Java Program to Accept 2 String & Check Whether all Characters in First String is Present in Second String & Print.

Source Code :

```
import java.util.Scanner;
public class q40 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the first string: ");
        String str1 = scanner.nextLine();
        System.out.print("Enter the second string: ");
        String str2 = scanner.nextLine();
        StringBuilder sb = new StringBuilder();
        for (int i = 0; i < str1.length(); i++) {
            char currentChar = str1.charAt(i);
            if (str2.indexOf(currentChar) != -1) {
                sb.append(str1.charAt(i)).append(" ");
            }
        }
        System.out.println("Common characters are : " + sb.toString());
        scanner.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q40.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q40
Enter the first string: hello
Enter the second string: world
Common characters are : l l o
```

Question 41 : Write a Java Program to Check whether a Given Character is Present in a String, Find Frequency & Position of Occurrence.

Source Code :

```
import java.util.Scanner;
public class q41 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String str = sc.nextLine();
        System.out.print("Enter a character to search for: ");
        char ch = sc.next().charAt(0);
        int freq = 0;
        System.out.print("Position(s) of occurrence: ");
        for (int i = 0; i < str.length(); i++) {
            if (str.charAt(i) == ch) {
                freq++;
                System.out.print((i + 1) + " ");
            }
        }
        System.out.println("\nfreq of character " + ch + ": " + freq);
        sc.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q41.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q41
Enter a string: sahin
Enter a character to search for: i
Position(s) of occurrence: 4
freq of character 'i': 1
```

Question 42 : Write a Java Program to Count the Number of Occurrence of Each Character Ignoring the Case of Alphabets & Display them.

Source Code :

```
import java.util.Scanner;
import java.util.HashMap;
public class q42 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter a string: ");
        String inputString = sc.nextLine();
        inputString = inputString.toLowerCase();
        System.out.println("Occurrences of each character (ignoring case):");
        HashMap<Character, Integer> hm = new HashMap<>();
        for (char c : inputString.toCharArray()) {
            hm.put(c, hm.getOrDefault(c, 0) + 1);
        }
        System.out.println(hm);
        sc.close();
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>javac q42.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week8\week 8>java q42
Enter a string: hello This is sahin. HELLO I am a Man
Occurrences of each character (ignoring case):
{ =8, a=4, s=3, t=1, e=2, h=4, i=4, l=4, m=2, n=2, .=1, o=2}
```

Week 9

Question 1 : Write a Java program in which total 4 threads should run. Set different priorities to the thread.

Source Code :

```
public class q1 {
    public static void main(String[] args) {
        Thread t1 = new Thread(new MyRunnable(), "Thread 1");
        Thread t2 = new Thread(new MyRunnable(), "Thread 2");
        Thread t3 = new Thread(new MyRunnable(), "Thread 3");
        Thread t4 = new Thread(new MyRunnable(), "Thread 4");
        t1.setPriority(Thread.MIN_PRIORITY);
        t2.setPriority(Thread.NORM_PRIORITY);
        t3.setPriority(Thread.NORM_PRIORITY);
        t4.setPriority(Thread.MAX_PRIORITY);
        t1.start();
        t2.start();
        t3.start();
        t4.start();
    }
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println(Thread.currentThread().getName() + " is running with priority: "
                + Thread.currentThread().getPriority());
        }
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q1.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q1
Thread 4 is running with priority: 10
Thread 3 is running with priority: 5
Thread 2 is running with priority: 5
Thread 1 is running with priority: 1
```

Question 2 : Create 4 threads with priority 1,3,5,7 respectively. Update a counter in each of the threads for 10 ms. Print the final value of count for each thread

Source Code :

```
public class q2 {
    static int counter = 0;
    public static void main(String[] args) {
        Thread t1 = new Thread(new MyRunnable(1));
        Thread t2 = new Thread(new MyRunnable(3));
```

```

Thread t3 = new Thread(new MyRunnable(5));
Thread t4 = new Thread(new MyRunnable(7));
t1.start();
t2.start();
t3.start();
t4.start();
try {
    t1.join();
    t2.join();
    t3.join();
    t4.join();
} catch (InterruptedException e) {
    e.printStackTrace();
}
System.out.println("Final count for Thread 1: " + MyRunnable.count1);
System.out.println("Final count for Thread 2: " + MyRunnable.count2);
System.out.println("Final count for Thread 3: " + MyRunnable.count3);
System.out.println("Final count for Thread 4: " + MyRunnable.count4);
static class MyRunnable implements Runnable {
    static int count1 = 0, count2 = 0, count3 = 0, count4 = 0;
    int priority;
    MyRunnable(int priority) {
        this.priority = priority;
    }
    public void run() {
        Thread.currentThread().setPriority(priority);
        for (int i = 0; i < 10; i++) {
            synchronized (this) {
                counter++;
                switch (priority) {
                    case 1:
                        count1++;
                        break;
                    case 3:
                        count2++;
                        break;
                    case 5:
                        count3++;
                        break;
                    case 7:
                        count4++;
                        break;
                }
                try {
                    Thread.sleep(10);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
            }
        }
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q2.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q2
Final count for Thread 1: 10
Final count for Thread 2: 10
Final count for Thread 3: 10
Final count for Thread 4: 10

```

Question 3 : Write a Java Program to Use Method Level Synchronization.**Source Code :**

```

public class q3 {
    private static int counter = 0;
    public static void main(String[] args) {
        Thread t1 = new Thread(new MyRunnable(), "Thread 1");
        Thread t2 = new Thread(new MyRunnable(), "Thread 2");
        t1.start();
        t2.start();
        try {
            t1.join();
            t2.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        System.out.println("Final count: " + counter);
    }
    static class MyRunnable implements Runnable {
        public void run() {
            for (int i = 0; i < 1000; i++) {

```

```

        incrementCounter();}
private synchronized void incrementCounter() {
    counter++;}}}

```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q3.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q3
Final count: 1864
```

Question 4 : Write a Java Program to Use Block Level Synchronization.**Source Code :**

```

public class q4 {
    private static int counter = 0;
    private static final Object lock = new Object();
    public static void main(String[] args) {
        Thread t1 = new Thread(new MyRunnable(), "Thread 1");
        Thread t2 = new Thread(new MyRunnable(), "Thread 2");
        t1.start();
        t2.start();
        try {
            t1.join();
            t2.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        System.out.println("Final count: " + counter);
    }
    static class MyRunnable implements Runnable {
        public void run() {
            for (int i = 0; i < 1000; i++) {
                incrementCounter();
            }
        }
        private void incrementCounter() {
            synchronized (lock) {
                counter++;
            }
        }
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q4.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q4
Final count: 2000
```

Question 5 : Write a Java Program to Check Whether Define run() Method as Synchronized.**Source Code :**

```

public class q5 {
    private static int counter = 0;
    public static void main(String[] args) {
        Thread t1 = new Thread(new MyRunnable(), "Thread 1");
        Thread t2 = new Thread(new MyRunnable(), "Thread 2");
        t1.start();
        t2.start();
        try {
            t1.join();
            t2.join();
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        System.out.println("Final count: " + counter);
    }
    static class MyRunnable implements Runnable {
        public synchronized void run() {
            for (int i = 0; i < 1000; i++) {
                incrementCounter();
            }
        }
        private void incrementCounter() {
            counter++;
        }
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q5.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q5
Final count: 1960
```

Question 6 : Write a Java Program to Solve Producer Consumer Problem Using Synchronization.**Source Code :**

```

import java.util.LinkedList;
public class q6 {

```

```

public static void main(String[] args) {
    Buffer buffer = new Buffer();
    new Thread(() -> buffer.produce()).start();
    new Thread(() -> buffer.consume()).start();}}
class Buffer {
    private final LinkedList<Integer> queue = new LinkedList<>();
    private final int capacity = 5;
    public synchronized void produce() {
        try {
            for (int i = 0; i < 10; i++) {
                while (queue.size() == capacity) wait();
                queue.add(i);
                System.out.println("Produced: " + i);
                notify();
                Thread.sleep(1000);
            } catch (InterruptedException e) {
                e.printStackTrace();}}
    public synchronized void consume() {
        try {
            for (int i = 0; i < 10; i++) {
                while (queue.isEmpty()) wait();
                int consumed = queue.removeFirst();
                System.out.println("Consumed: " + consumed);
                notify();
                Thread.sleep(2000);
            } catch (InterruptedException e) {
                e.printStackTrace();}}}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q6.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q6
Produced: 0
Produced: 1
Produced: 2
Produced: 3
Produced: 4
Consumed: 0
Consumed: 1
Consumed: 2
Consumed: 3
Consumed: 4
Produced: 5
Produced: 6
Produced: 7
Produced: 8
Produced: 9
Consumed: 5
Consumed: 6
Consumed: 7
Consumed: 8
Consumed: 9

```

Question 7 : Write a Java Program to Show that Method Will be Verified Whether it is Synchronized or Not.**Source Code :**

```

import java.lang.reflect.Method;
import java.lang.reflect.Modifier;
public class q7 {
    public static void main(String[] args) {
        Method[] methods = SynchronizedClass.class.getDeclaredMethods();
        for (Method method : methods) {
            System.out.println("Method: " + method.getName());
            boolean isSynchronized = isMethodSynchronized(method);
            System.out.println("Synchronized: " + (isSynchronized ? "Yes" : "No"));}}
    private static boolean isMethodSynchronized(Method method) {
        return (method.getModifiers() & Modifier.SYNCHRONIZED) != 0;}}
class SynchronizedClass {
    public synchronized void synchronizedMethod() {
        System.out.println("Synchronized method is being executed.");}
    public void nonSynchronizedMethod() {
        System.out.println("Non-Synchronized method is being executed.");}}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q7.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q7
Method: synchronizedMethod
Synchronized: Yes
Method: nonsynchronizedMethod
Synchronized: No

```

Question 8 : Write a Java Program to Show How Can Class Object be Locked Using Method Level Synchronization.**Source Code :**

```
public class q8 {
    public static void main(String[] args) {
        Thread thread1 = new Thread(new MyRunnable(), "Thread 1");
        Thread thread2 = new Thread(new MyRunnable(), "Thread 2");
        thread1.start();
        thread2.start();}}
class MyClass {
    public synchronized void synchronizedMethod() {
        System.out.println(Thread.currentThread().getName() + " is executing synchronized method.");
        try {
            Thread.sleep(2000);
        } catch (InterruptedException e) {
            e.printStackTrace();}}}
class MyRunnable implements Runnable {
    private static MyClass myClass = new MyClass();
    public void run() {
        myClass.synchronizedMethod();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q8.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q8
Thread 1 is executing synchronized method.
Thread 2 is executing synchronized method.
```

Question 9 : Write a Java Program to Synchronize the Threads Acting on the Same Object. The Synchronized Block in the Program can be Executed by Only One Thread at a Time.**Source Code :**

```
public class q9 {
    public static void main(String[] args) {
        SharedObject sharedObject = new SharedObject();
        Thread thread1 = new Thread(new MyRunnable(sharedObject), "Thread 1");
        Thread thread2 = new Thread(new MyRunnable(sharedObject), "Thread 2");
        thread1.start();
        thread2.start();}}
class SharedObject {
    public void synchronizedMethod() {
        synchronized (this) {
            System.out.println(Thread.currentThread().getName() + " is executing synchronized method.");
            try {
                Thread.sleep(2000);
            } catch (InterruptedException e) {
                e.printStackTrace();}}}}
class MyRunnable implements Runnable {
    private SharedObject sharedObject;
    public MyRunnable(SharedObject sharedObject) {
        this.sharedObject = sharedObject;}
    public void run() {
        sharedObject.synchronizedMethod();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q9.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q9
Thread 1 is executing synchronized method.
Thread 2 is executing synchronized method.
```

Question 10 : Write a Java Program to Avoid Dead Locks.**Source Code :**

```
public class q10_2nd {
    public static void main(String[] args) {
        final q10 acc1 = new q10();
        final q10 acc2 = new q10();
        Thread thread1 = new Thread(() -> {
            for (int i = 0; i < 1000; i++) {
                acc1.transfer(acc2, 10);}});
        Thread thread2 = new Thread(() -> {
            for (int i = 0; i < 1000; i++) {
                acc2.transfer(acc1, 10);});}}
```

```

thread1.start();
thread2.start();
try {
    thread1.join();
    thread2.join();
} catch (InterruptedException e) {
    e.printStackTrace();
}
System.out.println("Balance in Account 1: " + acc1.getBalance());
System.out.println("Balance in Account 2: " + acc2.getBalance());

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q10_2nd.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q10_2nd
Balance in Account 1: 10000
Balance in Account 2: 10000

```

Question 11 : Write a Java Program to Solve Deadlock Using Thread.**Source Code :**

```

public class q11 {
    private static final Object lock1 = new Object();
    private static final Object lock2 = new Object();
    public static void main(String[] args) {
        Thread thread1 = new Thread(() -> {
            synchronized (lock1) {
                System.out.println("Thread 1: Holding lock 1...");
                try {
                    Thread.sleep(100);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
                System.out.println("Thread 1: Waiting for lock 2...");
                synchronized (lock2) {
                    System.out.println("Thread 1: Holding lock 1 and lock 2..."));
                }
            }
        });
        Thread thread2 = new Thread(() -> {
            synchronized (lock1) {
                System.out.println("Thread 2: Holding lock 1...");
                try {
                    Thread.sleep(100);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
                System.out.println("Thread 2: Waiting for lock 2...");
                synchronized (lock2) {
                    System.out.println("Thread 2: Holding lock 1 and lock 2..."));
                }
            }
        });
        thread1.start();
        thread2.start();
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q11.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q11
Thread 1: Holding lock 1...
Thread 1: Waiting for lock 2...
Thread 1: Holding lock 1 and lock 2...
Thread 2: Holding lock 1...
Thread 2: Waiting for lock 2...
Thread 2: Holding lock 1 and lock 2...

```

Question 12 : Write a Java Program to Create a Thread that Implement the Runnable Interface.**Source Code :**

```

public class q12 implements Runnable {
    public void run() {
        System.out.println("Thread is running...");
    }
    public static void main(String[] args) {
        q12 myRunnable = new q12();
        Thread thread = new Thread(myRunnable);
        thread.start();
    }
}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q12.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q12
Thread is running...

```

Question 13 : Write a Java Program to Show the Priority in Threads.**Source Code :**

```

public class q13 {
    public static void main(String[] args) {
        Thread thread1 = new Thread(new MyRunnable(), "Thread 1");
        Thread thread2 = new Thread(new MyRunnable(), "Thread 2");
        Thread thread3 = new Thread(new MyRunnable(), "Thread 3");
        thread1.setPriority(Thread.MIN_PRIORITY);
        thread2.setPriority(Thread.NORM_PRIORITY);
        thread3.setPriority(Thread.MAX_PRIORITY);
        thread1.start();
        thread2.start();
        thread3.start();
        System.out.println(thread1.getName() + " priority: " + thread1.getPriority());
        System.out.println(thread2.getName() + " priority: " + thread2.getPriority());
        System.out.println(thread3.getName() + " priority: " + thread3.getPriority());
    }
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println(Thread.currentThread().getName() + " is running...");}}}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q13.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q13
Thread 1 priority: 1
Thread 2 priority: 5
Thread 3 priority: 10
Thread 2 is running...
Thread 3 is running...
Thread 1 is running...

```

Question 14 : Write a Java Program to Check Priority Level of a Thread.**Source Code :**

```

public class q14 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable(), "MyThread");
        int priority = thread.getPriority();
        System.out.println("Thread priority: " + priority);
    }
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println(Thread.currentThread().getName() + " is running...");}}}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q14.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q14
Thread priority: 5

```

Question 15 : Write a Java Program to Set the Priority of a Thread**Source Code :**

```

public class q15 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable(), "MyThread");
        thread.setPriority(Thread.MAX_PRIORITY);
        thread.start();
    }
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println(Thread.currentThread().getName() + " is running with priority: " + Thread.currentThread().getPriority());}}}

```

Output :

```

C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q15.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q15
MyThread is running with priority: 10

```

Question 16 : Write a Java Program to Get the Priorities of Running Threads.**Source Code :**

```

public class q16 {
    public static void main(String[] args) {
        Thread thread1 = new Thread(new MyRunnable(), "Thread 1");
        Thread thread2 = new Thread(new MyRunnable(), "Thread 2");
        thread1.start();
        thread2.start();
        int priority1 = thread1.getPriority();
        int priority2 = thread2.getPriority();
        System.out.println(thread1.getName() + " priority: " + priority1);
        System.out.println(thread2.getName() + " priority: " + priority2);}}
```

```
static class MyRunnable implements Runnable {
    public void run() {
        System.out.println(Thread.currentThread().getName() + " is running with priority: " + Thread.currentThread().getPriority());}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q16.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q16
Thread 1 is running with priority: 5
Thread 1 priority: 5
Thread 2 is running with priority: 5
Thread 2 priority: 5
```

Question 17 : Write a Java Program to Access the Priority You Can Use Method With Thread Object.**Source Code :**

```
public class q17 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable(), "MyThread");
        int priority = thread.getPriority();
        System.out.println("Thread priority before setting: " + priority);
        thread.setPriority(Thread.MAX_PRIORITY);
        int updatedPriority = thread.getPriority();
        System.out.println("Thread priority after setting: " + updatedPriority);}
```

```
static class MyRunnable implements Runnable {
    public void run() {
        System.out.println(Thread.currentThread().getName() + " is running with priority: " + Thread.currentThread().getPriority());}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q17.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q17
Thread priority before setting: 5
Thread priority after setting: 10
```

Question 18 : Write a Java Program to Use Join Thread**Source Code :**

```
public class q18 {
    public static void main(String[] args) {
        Thread thread1 = new Thread(new MyRunnable(), "Thread 1");
        Thread thread2 = new Thread(new MyRunnable(), "Thread 2");
        thread1.start();
        try {
            thread1.join();
            System.out.println("Thread 1 has finished.");
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        thread2.start();
    }
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println(Thread.currentThread().getName() + " is running...");
            try {
                Thread.sleep(2000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
            System.out.println(Thread.currentThread().getName() + " finished.");}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q18.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q18
Thread 1 is running...
Thread 1 finished.
Thread 1 has finished.
Thread 2 is running...
Thread 2 finished.
```

Question 19 : Write a Java Program Defining Thread By Extending Thread**Source Code :**

```
public class q19 extends Thread {
    public void run() {
        System.out.println("Thread is running...");}
```

```
public static void main(String[] args) {
    q19 thread = new q19();
    thread.start();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q19.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q19
Thread is running...
```

Question 20 : Write a Java Program to Handle IllegalThreadStateException.**Source Code :**

```
public class q20 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable(), "MyThread");
        thread.start();
        try {
            thread.start();
        } catch (IllegalThreadStateException e) {
            System.out.println("IllegalThreadStateException caught: " + e.getMessage());
        }
        static class MyRunnable implements Runnable {
            public void run() {
                System.out.println(Thread.currentThread().getName() + " is running...");
            }
        }
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q20.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q20
MyThread is running...
IllegalThreadStateException caught: null
```

Question 21 : Write a Java Program to Check Whether Static Block will be Used.**Source Code :**

```
public class q21 {
    static {
        System.out.println("Static block is executed.");
    }
    public static void main(String[] args) {
        System.out.println("Main method is executed.");
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q21.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q21
Static block is executed.
Main method is executed.
```

Question 22 : Write a Java Program to Show Why Exit Method is Used in Static Method.**Source Code :**

```
public class q22 {
    public static void main(String[] args) {
        System.out.println("Main method is executing...");
        myStaticMethod();
    }
    public static void myStaticMethod() {
        System.out.println("Static method is executing...");
        System.exit(0);
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q22.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q22
Main method is executing...
Static method is executing...
```

Question 23 : Write a Java Program to Illustrate Thread Example for setName(string name)**Source Code :**

```
public class q23 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable());
        thread.setName("MyThread");
        thread.start();
    }
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println("Thread name: " + Thread.currentThread().getName());
        }
    }
}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q23.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q23
Thread name: MyThread
```

Question 26 : Write a Java Program to Illustrate Thread Example for currentThread().**Source Code :**

```
public class q26 {
    public static void main(String[] args) {
        Thread thread = Thread.currentThread();
        System.out.println("Current thread: " + thread.getName());}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q26.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q26
Current thread: main
```

Question 27 : Write a Java Program to Illustrate Thread Example for run().**Source Code :**

```
public class q27 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable());
        thread.start();
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println("Thread is running...");}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q27.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q27
Thread is running...
```

Question 28 : Write a Java Program to Illustrate Thread Example for getThreadGroup().**Source Code :**

```
public class q28 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable());
        System.out.println("Thread group: " + thread.getThreadGroup().getName());
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println("Thread is running...");}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q28.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q28
Thread group: main
```

Question 29 : Write a Java Program to Illustrate Thread Example for getPriority().**Source Code :**

```
public class q29 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable());
        int priority = thread.getPriority();
        System.out.println("Thread priority: " + priority);
    static class MyRunnable implements Runnable {
        public void run() {
            System.out.println("Thread is running...");}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q29.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q29
Thread priority: 5
```

Question 30 : Write a Java Program to Illustrate Thread Example for Alive().**Source Code :**

```
public class q30 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable());
        System.out.println("Thread is alive before starting: " + thread.isAlive());
        thread.start();
        System.out.println("Thread is alive after starting: " + thread.isAlive());
    static class MyRunnable implements Runnable {
        public void run() {
```

```
System.out.println("Thread is running...");}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q30.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q30
Thread is alive before starting: false
Thread is running...
Thread is alive after starting: true
```

Question 31 : Design an abstract class having two methods. Create Rectangle and Triangle classes by inheriting the shape class and override the above methods to suitably implement for Rectangle and Triangle class.

Source Code :

```
public class q31 {
    public static void main(String[] args) {
        Thread thread = new Thread(new MyRunnable(), "MyThread");
        System.out.println("Thread name: " + thread.getName());
        static class MyRunnable implements Runnable {
            public void run() {
                System.out.println("Thread is running...");}
    }}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q31.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q31
Thread name: MyThread
```

Question 32 : Design an abstract class having two methods. Create Rectangle and Triangle classes by inheriting the shape class and override the above methods to suitably implement for Rectangle and Triangle class.

Source Code :

```
interface MyInterface {
    void display();}
interface MyExtendedInterface extends MyInterface {
    void show();}
public class q32 implements MyExtendedInterface {
    public void display() {
        System.out.println("Display method implementation");}
    public void show() {
        System.out.println("Show method implementation");}
    public static void main(String[] args) {
        q32 example = new q32();
        example.display();
        example.show();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q32.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q32
Display method implementation
Show method implementation
```

Question 33 : Design an abstract class having two methods. Create Rectangle and Triangle classes by inheriting the shape class and override the above methods to suitably implement for Rectangle and Triangle class.

Source Code :

```
interface MyInterface {
    void display();}
interface MyExtendedInterface extends MyInterface {
    void show();}
public class q33 implements MyExtendedInterface {
    public void display() {
        System.out.println("Display method implementation");}
    public void show() {
        System.out.println("Show method implementation");}
    public static void main(String[] args) {
        q33 example = new q33();
        example.display();
        example.show();}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q33.java
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q33
Display method implementation
Show method implementation
```

Question 34 : Design an abstract class having two methods. Create Rectangle and Triangle classes by inheriting the shape class and override the above methods to suitably implement for Rectangle and Triangle class.

Source Code :

```
public class q34 {  
    public static void main(String[] args) {  
        Thread thread = new Thread(new MyRunnable(), "MyThread");  
        thread.start();  
        System.out.println("Running thread name: " + Thread.currentThread().getName());}  
    static class MyRunnable implements Runnable {  
        public void run() {  
            System.out.println("Thread is running...");}}}
```

Output :

```
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>javac q34.java  
C:\Users\nayak\OneDrive\Desktop\mca assingments\sem2\java\code\week9\Week 9 _ Multithreading>java q34  
Thread is running...  
Running thread name: main
```