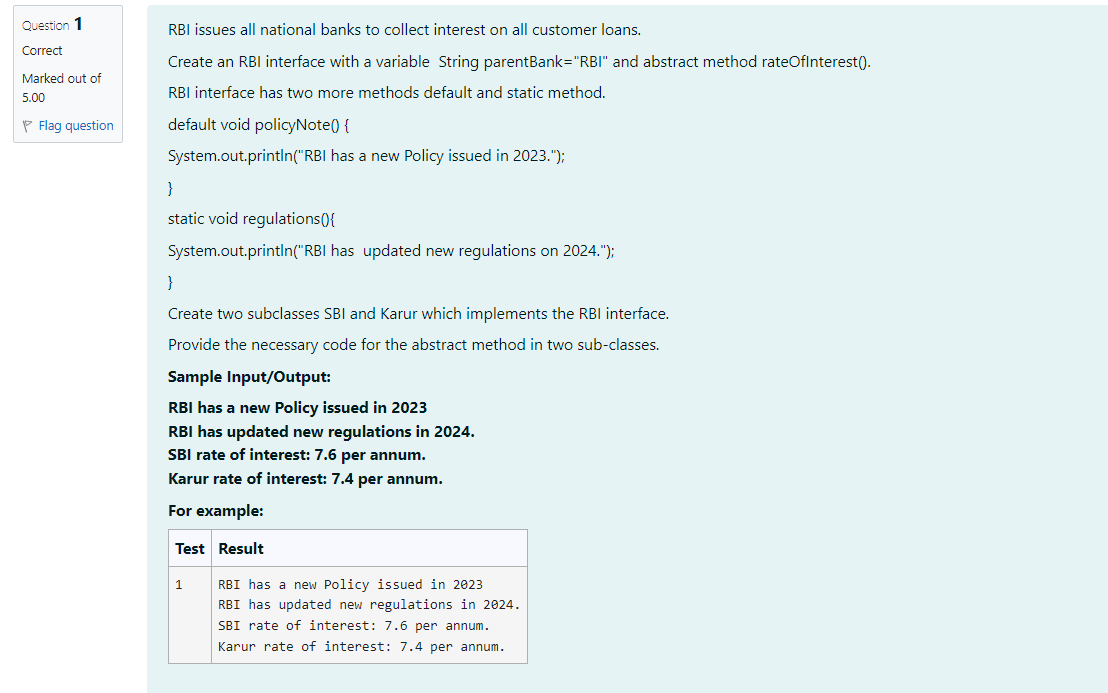
**Week\_7 INTERFACES**

**NAME: ENIYA.B.A**

**ROLL NO:230701085**

****

CODE:

interface RBI{

    String parentBank ="RBI";

    double rateofInterest();

    default void policyNote() {

       System.out.println("RBI has updated new regulations in 2024.");}

    static void regulations() {

        System.out.println("RBI has a new Policy issued in 2023");}

}

class SBI implements RBI{

  public double rateofInterest() {

         return 7.6;

}

}

class Karur implements RBI {

    public double rateofInterest() {

        return 7.4;

}

}

public class Main {

 public static void main(String[] args) {

    RBI.regulations();

SBI sbiBank =new SBI();

Karur karurBank =new Karur();

sbiBank.policyNote();

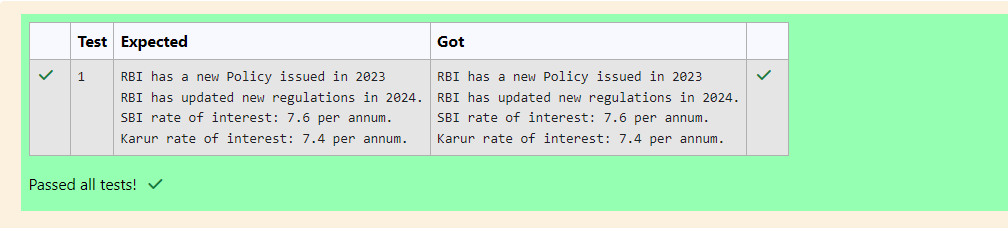
System.out.println("SBI rate of interest: "+ sbiBank.rateofInterest()+"per annum.");

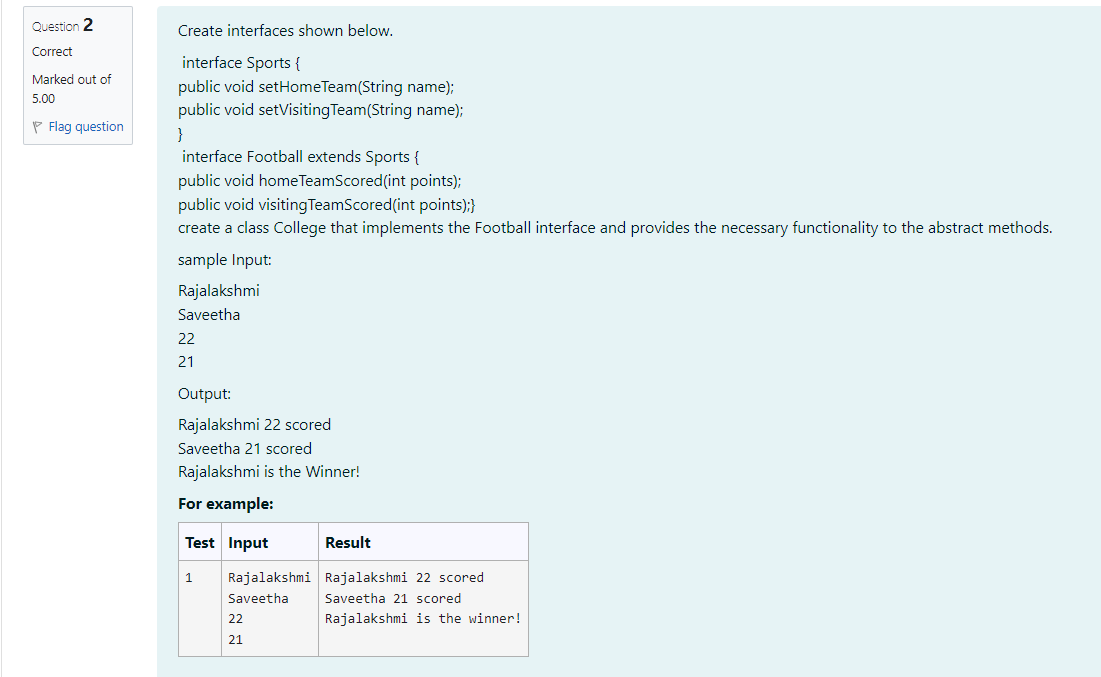
System.out.println("Karur rate of interest: "+karurBank.rateofInterest()+" per annum.");

}

}

OUTPUT:





CODE:

import java.util.Scanner;

// Sports Interface

interface Sports {

    public void setHomeTeam(String name);

    public void setVisitingTeam(String name);

}

// Football Interface extending Sports

interface Football extends Sports {

    public void homeTeamScored(int points);

    public void visitingTeamScored(int points);

}

// College class implementing Football interface

class College implements Football {

    private String homeTeam;

    private String visitingTeam;

    private int homeTeamPoints;

    private int visitingTeamPoints;

    // Implementing setHomeTeam method

    @Override

    public void setHomeTeam(String name) {

        this.homeTeam = name;

    }

    // Implementing setVisitingTeam method

    @Override

    public void setVisitingTeam(String name) {

        this.visitingTeam = name;

    }

    // Implementing homeTeamScored method

    @Override

    public void homeTeamScored(int points) {

        this.homeTeamPoints = points;

    }

    // Implementing visitingTeamScored method

    @Override

    public void visitingTeamScored(int points) {

        this.visitingTeamPoints = points;

    }

    // Method to display the result

    public void displayResult() {

        System.out.println(homeTeam + " " + homeTeamPoints + " scored");

        System.out.println(visitingTeam + " " + visitingTeamPoints + " scored");

        if (homeTeamPoints > visitingTeamPoints) {

            System.out.println(homeTeam + " is the Winner!");

        } else if (visitingTeamPoints > homeTeamPoints) {

            System.out.println(visitingTeam + " is the Winner!");

        } else {

            System.out.println("It's a tie match.");

        }

    }

}

// Main class to execute the program

public class Main {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        // Reading input dynamically

        String homeTeam = scanner.nextLine(); // First input: Home team name

        String visitingTeam = scanner.nextLine(); // Second input: Visiting team name

        int homeTeamScore = scanner.nextInt(); // Third input: Home team score

        int visitingTeamScore = scanner.nextInt(); // Fourth input: Visiting team score

        // Creating an instance of College class

        College collegeMatch = new College();

        // Setting teams and scores

        collegeMatch.setHomeTeam(homeTeam);

        collegeMatch.setVisitingTeam(visitingTeam);

        collegeMatch.homeTeamScored(homeTeamScore);

        collegeMatch.visitingTeamScored(visitingTeamScore);

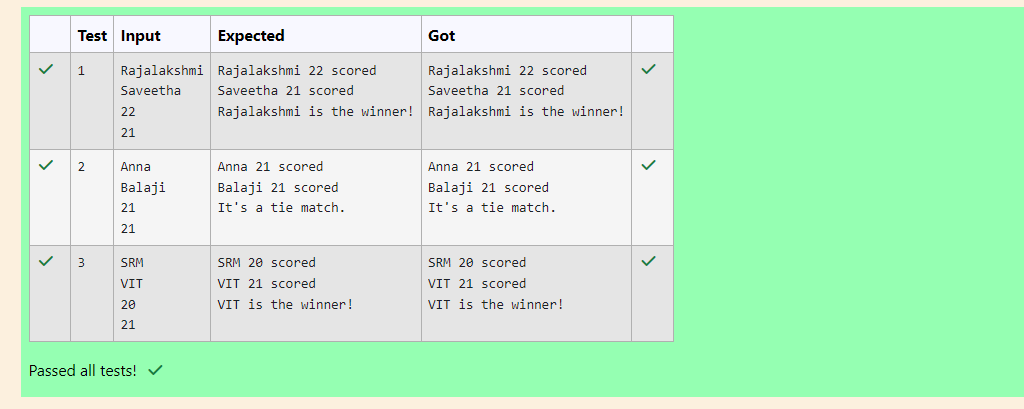
        // Displaying the result

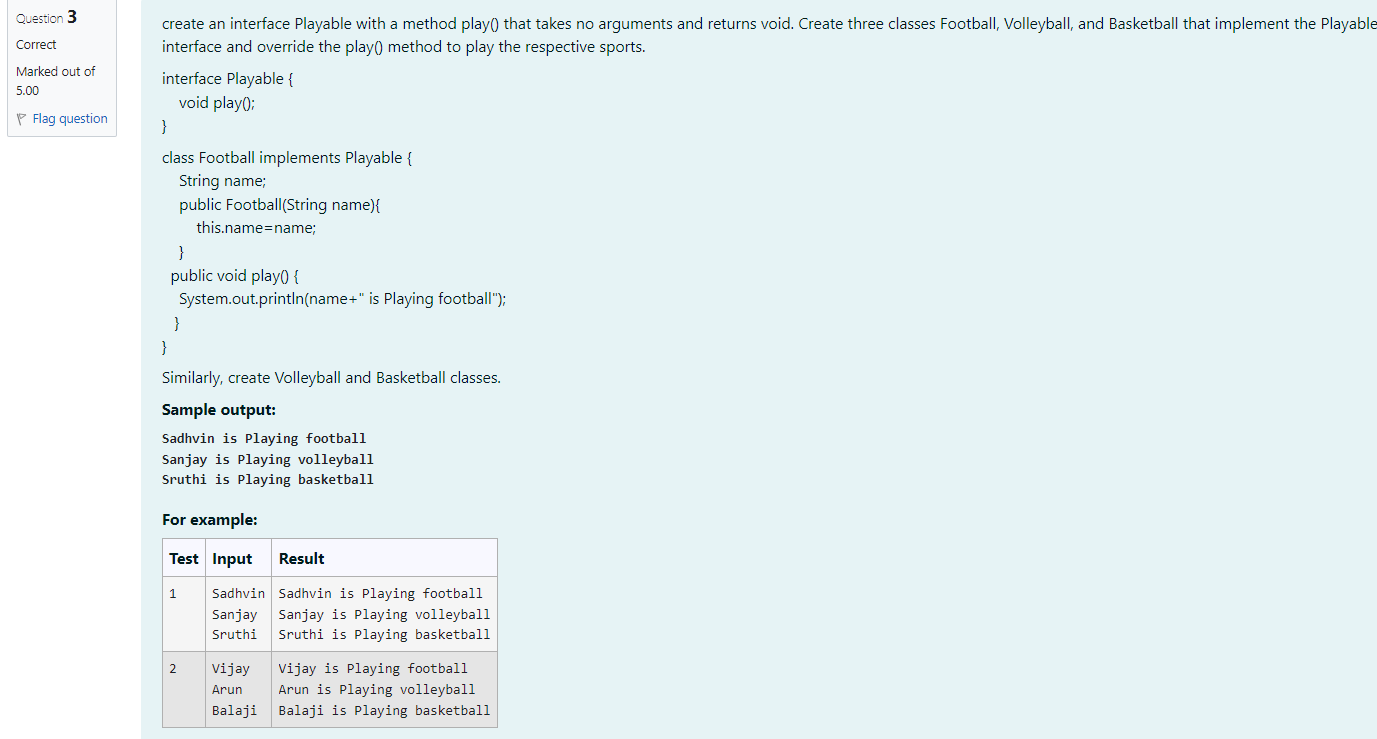
        collegeMatch.displayResult();

    }

}

OUTPUT:





CODE:

import java.util.Scanner;

interface Playable {

    void play();

}

class Football implements Playable {

    String name;

    public Football(String name) {

[this.name](http://this.name/) = name;

    }

    public void play() {

        System.out.println(name + " is Playing football");

    }

}

class Volleyball implements Playable {

    String name;

    public Volleyball(String name) {

[this.name](http://this.name/) = name;

    }

    public void play() {

        System.out.println(name + " is Playing volleyball");

    }

}

class Basketball implements Playable {

    String name;

    public Basketball(String name) {

[this.name](http://this.name/) = name;

    }

    public void play() {

        System.out.println(name + " is Playing basketball");

    }

}

public class Main {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        // Reading names from input dynamically

        String footballPlayerName = scanner.nextLine();

        String volleyballPlayerName = scanner.nextLine();

        String basketballPlayerName = scanner.nextLine();

        // Create players dynamically based on input

        Football footballPlayer = new Football(footballPlayerName);

        Volleyball volleyballPlayer = new Volleyball(volleyballPlayerName);

        Basketball basketballPlayer = new Basketball(basketballPlayerName);

        // Play the respective sports

        footballPlayer.play();

        volleyballPlayer.play();

        basketballPlayer.play();

        scanner.close();

    }

}

OUTPUT:

