# 

# 

# 

# 

# 

# LAB 07

# INTERFACES

EXERCISE 1

create an interface Playable with a method play() that takes no arguments and returns void. Create three classes Football, Volleyball, and Basketball that implement the Playable interface and override the play() method to play the respective sports.

interface Playable {  
 void play();  
}

class Football implements Playable {  
 String name;  
 public Football(String name){  
 this.name=name;  
 }  
 public void play() {  
 System.out.println(name+" is Playing football");  
 }  
}

Similarly, create Volleyball and Basketball classes.

Sample output:

Sadhvin is Playing football

Sanjay is Playing volleyball

Sruthi is Playing basketball

**For example:**

| **Test** | **Input** | **Result** |
| --- | --- | --- |
| 1 | Sadhvin  Sanjay  Sruthi | Sadhvin is Playing football  Sanjay is Playing volleyball  Sruthi is Playing basketball |
| 2 | Vijay  Arun  Balaji | Vijay is Playing football  Arun is Playing volleyball  Balaji is Playing basketball |

Answer:

import java.util.\*;

interface playable{

void play();

}

class football implements playable{

String name;

public football(String name){

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=name;

}

public void play(){

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

}

}

class basketball implements playable{

String name;

public basketball(String name){

this.name=name;

}

public void play(){

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

}

}

public class Main{

public static void main(String[] args){

Scanner inp = new Scanner(System.in);

String a= inp.nextLine();

football f= new football(a);

f.play();

String b=inp.nextLine();

volleyball v=new volleyball(b);

v.play();

String c=inp.nextLine();

basketball b1= new basketball(c);

b1.play();

}

}

|  | **Test** | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | Sadhvin  Sanjay  Sruthi | Sadhvin is Playing football  Sanjay is Playing volleyball  Sruthi is Playing basketball | Sadhvin is Playing football  Sanjay is Playing volleyball  Sruthi is Playing basketball |  |
|  | 2 | Vijay  Arun  Balaji | Vijay is Playing football  Arun is Playing volleyball  Balaji is Playing basketball | Vijay is Playing football  Arun is Playing volleyball  Balaji is Playing basketball |  |

EXERCISE 2

Create interfaces shown below.

interface Sports {  
public void setHomeTeam(String name);  
public void setVisitingTeam(String name);   
}  
 interface Football extends Sports {  
public void homeTeamScored(int points);  
public void visitingTeamScored(int points);}  
create a class College that implements the Football interface and provides the necessary functionality to the abstract methods.

sample Input:

Rajalakshmi  
Saveetha  
22  
21

Output:

Rajalakshmi 22 scored  
Saveetha 21 scored  
Rajalakshmi is the Winner!

**For example:**

| **Test** | **Input** | **Result** |
| --- | --- | --- |
| 1 | Rajalakshmi  Saveetha  22  21 | Rajalakshmi 22 scored  Saveetha 21 scored  Rajalakshmi is the winner! |

Answer:

import java.util.Scanner;

interface Sports {

public void setHomeTeam(String name);

public void setVisitingTeam(String name);

}

interface Football extends Sports {

public void homeTeamScored(int points);

public void visitingTeamScored(int points);

}

class College implements Football {

String homeTeam;

String visitingTeam;

int points;

public void setHomeTeam(String name){

homeTeam=name;

}

public void setVisitingTeam(String name){

visitingTeam=name;

}

public void homeTeamScored(int points){

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

}

public void visitingTeamScored(int points){

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

}

public void winningTeam(int p1, int p2){

if(p1>p2){

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

else if(p1<p2){

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

else{

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

}

}}

public class Main{

public static void main(String[] args){

String hname;

Scanner sc= new Scanner(System.in);

hname=sc.nextLine();

String vteam=sc.nextLine();

int a = sc.nextInt();

int b = sc.nextInt();

int htpoints=a;

int vtpoints=b;

College s= new College();

s.setHomeTeam(hname);

s.setVisitingTeam(vteam);

s.homeTeamScored(htpoints);

s.visitingTeamScored(vtpoints);

s.winningTeam(htpoints, vtpoints);

}

}

|  | **Test** | **Input** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | Rajalakshmi  Saveetha  22  21 | Rajalakshmi 22 scored  Saveetha 21 scored  Rajalakshmi is the winner! | Rajalakshmi 22 scored  Saveetha 21 scored  Rajalakshmi is the winner! |  |
|  | 2 | Anna  Balaji  21  21 | Anna 21 scored  Balaji 21 scored  It's a tie match. | Anna 21 scored  Balaji 21 scored  It's a tie match. |  |
|  | 3 | SRM  VIT  20  21 | SRM 20 scored  VIT 21 scored  VIT is the winner! | SRM 20 scored  VIT 21 scored  VIT is the winner! |  |

EXERCISE 3

RBI issues all national banks to collect interest on all customer loans.

Create an RBI interface with a variable String parentBank="RBI" and abstract method rateOfInterest().

RBI interface has two more methods default and static method.

default void policyNote() {

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

}

static void regulations(){

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

}

Create two subclasses SBI and Karur which implements the RBI interface.

Provide the necessary code for the abstract method in two sub-classes.

Sample Input/Output:

RBI has a new Policy issued in 2023  
RBI has updated new regulations in 2024.  
SBI rate of interest: 7.6 per annum.  
Karur rate of interest: 7.4 per annum.

**For example:**

| **Test** | **Result** |
| --- | --- |
| 1 | RBI has a new Policy issued in 2023  RBI has updated new regulations in 2024.  SBI rate of interest: 7.6 per annum.  Karur rate of interest: 7.4 per annum. |

Answer:

interface RBI{

String parentBank="RBI";

void rateofinterest();

default void policyNote(){

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

}

static void regulation(){

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

}

}

class SBI implements RBI{

public void rateofinterest(){

System.out.println("SBI rate of interest: 7.6 per annum.");

}

}

class Karur implements RBI{

public void rateofinterest(){

System.out.println("Karur rate of interest: 7.4 per annum.");

}

}

public class Main{

public static void main(String[] args){

SBI s = new SBI();

Karur k = new Karur();

s.policyNote();

RBI.regulation();

s.rateofinterest();

k.rateofinterest();

}

}

|  | **Test** | **Expected** | **Got** |  |
| --- | --- | --- | --- | --- |
|  | 1 | RBI has a new Policy issued in 2023  RBI has updated new regulations in 2024.  SBI rate of interest: 7.6 per annum.  Karur rate of interest: 7.4 per annum. | RBI has a new Policy issued in 2023  RBI has updated new regulations in 2024.  SBI rate of interest: 7.6 per annum.  Karur rate of interest: 7.4 per annum. |  |