

CS23333-Object Oriented Programming Using Java-2023

[Dashboard](#) / [My courses](#) / [CS23333-OOPJ-2023](#) / [Lab-07-Interfaces](#) / [Lab-07-Logic Building](#)

Quiz navigation

- 1
- 2
- 3

[Show one page at a time](#)

[Finish review](#)

Status	Finished
Started	Tuesday, 8 October 2024, 3:30 PM
Completed	Tuesday, 8 October 2024, 3:33 PM
Duration	2 mins 57 secs

Question 1

Correct

Marked out of 5.00

Flag question

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: (penalty regime: 0 %)

```
1 // Define the RBI interface  
2  
3 interface RBI {  
4  
5     // Variable declaration  
6     String parentBank = "RBI";  
7  
8     // Abstract method  
9     double rateOfInterest();  
10  
11     // Default method  
12     default void policyNote() {  
13         System.out.println("RBI has a new Policy issued in 2023");  
14     }  
15  
16     // Static method  
17     static void regulations() {  
18         System.out.println("RBI has updated new regulations in 2024.");  
19     }  
20 }  
21  
22 // SBI class implementing RBI interface  
23 class SBI implements RBI {  
24     // Implementing the abstract method  
25     public double rateOfInterest() {  
26         return 7.6;  
27     }  
28 }  
29  
30 // Karur class implementing RBI interface  
31 class Karur implements RBI {  
32     // Implementing the abstract method  
33     public double rateOfInterest() {  
34         return 7.4;  
35     }  
36 }  
37  
38 // Main class to test the functionality  
39 public class Main {  
40     public static void main(String[] args) {  
41         // RBI policies and regulations  
42         RBI rbi = new SBI(); // Can be any class implementing RBI  
43         rbi.policyNote(); // Default method  
44         RBI.regulations(); // Static method  
45  
46         // SBI bank details  
47         SBI sbi = new SBI();  
48         System.out.println("SBI rate of interest: " + sbi.rateOfInterest() + " per annum.");  
49  
50         // Karur bank details  
51         Karur karur = new Karur();  
52         System.out.println("Karur rate of interest: " + karur.rateOfInterest() + " per annum.");
```

	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.

Passed all tests!

Question 2

Correct

Marked out of 5.00

Flag question

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	Sadhvin is Playing football
	Sanjay	Sanjay is Playing volleyball
	Sruthi	Sruthi is Playing basketball
2	Vijay	Vijay is Playing football
	Arun	Arun is Playing volleyball
	Balaji	Balaji is Playing basketball

Answer: (penalty regime: 0 %)

```
1 import java.util.Scanner;
2
3
4
5 // Define the Playable interface
6 interface Playable {
7     // Abstract method to play the respective sport
8     void play();
9 }
10
11 // Football class implementing Playable interface
12 class Football implements Playable {
13     String name;
14
15     // Constructor
16     public Football(String name) {
17         this.name = name;
18     }
19
20     // Override the play method
21     public void play() {
22         System.out.println(name + " is Playing football");
23     }
24 }
25
26 // Volleyball class implementing Playable interface
27 class Volleyball implements Playable {
28     String name;
29
30     // Constructor
31     public Volleyball(String name) {
32         this.name = name;
33     }
34
35     // Override the play method
36     public void play() {
37         System.out.println(name + " is Playing volleyball");
38     }
39 }
40
41 // Basketball class implementing Playable interface
42 class Basketball implements Playable {
43     String name;
44
45     // Constructor
46     public Basketball(String name) {
47         this.name = name;
48     }
49
50     // Override the play method
51     public void play() {
52         System.out.println(name + " is Playing basketball");
```

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

Passed all tests!

Question 3

Correct

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: (penalty regime: 0 %)

Reset answer

```
1 import java.util.Scanner;
2
3
4
5 interface Sports {
6     void setHomeTeam(String name);
7     void setVisitingTeam(String name);
8 }
9
10 interface Football extends Sports {
11     void homeTeamScored(int points);
12     void visitingTeamScored(int points);
13 }
14
15 class College implements Football {
16     private String homeTeam;
17     private String visitingTeam;
18     private int homeTeamPoints = 0;
19     private int visitingTeamPoints = 0;
20
21     public void setHomeTeam(String name) {
22         this.homeTeam = name;
23     }
24
25     public void setVisitingTeam(String name) {
26         this.visitingTeam = name;
27     }
28
29     public void homeTeamScored(int points) {
30         homeTeamPoints += points;
31         System.out.println(homeTeam + " " + points + " scored");
32     }
33
34     public void visitingTeamScored(int points) {
35         visitingTeamPoints += points;
36         System.out.println(visitingTeam + " " + points + " scored");
37     }
38
39     public void winningTeam() {
40         if (homeTeamPoints > visitingTeamPoints) {
41             System.out.println(homeTeam + " is the winner!");
42         } else if (homeTeamPoints < visitingTeamPoints) {
43             System.out.println(visitingTeam + " is the winner!");
44         } else {
45             System.out.println("It's a tie match.");
46         }
47     }
48 }
49
50 public class Main {
51     public static void main(String[] args) {
52         Scanner sc = new Scanner(System.in);
```

	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!

Passed all tests!

Finish review