GOVERNMENT POLYTECHNIC, NANDED

MICRO PROJECT

Academic year: 2019-20

TITLE OF THE PROJECT

Use Interface

Program: Information Tech. **Program code: IT 4 I**

Course: JAVA Programming **Course code: 22412**

Name of Guide:-Mr.S.G.Mundhe

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

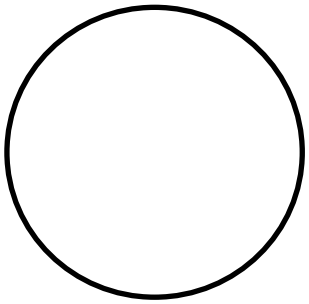
Certificate

This is to certify that Roll no **945, 948, 949 and 953** of **4th** Semester of Diploma in **INFORMATION TECTNOLOGY** of Institute, GOVERNMENT POLYTECHNIC has completed the **Micro Project satisfactorily** in Subject -**java (22412)** for the academic year **2019-2020** as prescribed in the curriculum.

Place: **Nanded**

Date:

Subject Teacher	Head of the Department	Principal
Mr.S.G.Mundhe	**Mr.S.N.Dhole**	**Dr.G.V.Garje**



WEEKLY PROGRESS REPORT

TITLE OF THE MICRO PROJECT:-Use Interface

WEEK	ACTIVITY PERFORMED	SIGN OF GUIDE	DATE
1ST	Discussion and finalization of Topic		
2ND	Discussion and finalization of Topic		
3RD	Preparation and submission of Abstract		
4TH	Literature Review		
5TH	Collection of Data		
6TH	Collection of Data		
7TH	Collection of Data		
8TH	Collection of Data		
9TH	Discussion and Outline of Content		
10TH	Formulation of Content		
11TH	Editing and 1st Proof Reading of Content		
12TH	Editing and 2nd Proof Reading of Content		
13TH	Compilation of Report and Presentation		
14TH	Seminar		
15TH	Viva-voce		
16TH	Final submission of Micro project		

Sign of the student

Sign of the faculty

Mr.S.G.Mundhe

ANEEXURE II

Evaluation Sheet for the Micro Project

Academic Year: 2019-20 **Name of the Faculty: Mr.S.G.Mundhe**

Course: JAVA Programming **Course code: 22412** **Semester: IV**

Title of the project: Use Interface

Cos addressed by Micro Project:

A: Develop programs using object oriented methodology in java.
B: Apply concept of inheritance for code reusability.
C: develop programs using multithreading.
D:Implement Exception handling

Major learning outcomes achieved by students by doing the project

(a) Practical outcome:
1) Deliver presentation (seminar) effectively.

(b) Unit outcomes in Cognitive domain:
1) Prepare the points for computer presentation.
2) Make seminar presentation.

(c) Outcomes in Affective domain:
1) Function as team member.
2) Follow Ethics.
3) Make proper use of computer and Internet

Comments/suggestions about team work /leadership/inter-personal communication (if any)

Roll No	Student Name	Marks out of 4 for performance in group activity (D5 Col.8)	Marks out of 2for performance in oral/ presentation (D5 Col.9)	Total out of 06
945	Harsh santosh zanwar			
948	Amaan Khan Pathan			
949	MD Hifaz ali Khan			
953	Aditya Pramod Joshi			

(Signature of Faculty)

INDEX

GROUP DETAILS

Roll No.	Name	Enrollment No.
945	Harsh Santosh Zanwar	1800200119
948	Amaan Khan Pathan	1815660
949	MD Hifaz Ali Khan	1815660141
953	Aditya Pramod Joshi	1900200225

Introduction

An **interface** in the Java programming language is an abstract type that is used to specify a behavior that classes must implement. They are similar to protocols. Interfaces are declared using the `interface` keyword, and may only contain method signature and constant declarations (variable declarations that are declared to be both `static` and `final`). All methods of an Interface do not contain implementation (method bodies) as of all versions below Java 8. Starting with Java 8, `default` and `static` methods may have implementation in the `interface` definition.[1] Then, in Java 9, `private` and `private static` methods were added. At present, a Java interface can have up to six different types.

Interfaces cannot be instantiated, but rather are implemented. A class that implements an interface must implement all of the non-default methods described in the interface, or be an abstract class. Object references in Java may be specified to be of an interface type; in each case, they must either be null, or be bound to an object that implements the interface.

One benefit of using interfaces is that they simulate multiple inheritance. All classes in Java must have exactly one base class, the only exception being `java.lang.Object` (the root class of the Java type system); multiple inheritance of classes is not allowed. However, an interface may inherit multiple interfaces and a class may implement multiple interfaces.

➢ Interface

An interface is just like Java Class, but it only has static constants and abstract method. Java uses Interface to implement multiple inheritance. A Java class can implement multiple Java Interfaces. All methods in an interface are implicitly public and abstract.

Syntax for Declaring Interface

Interface <interface Name>

{

}

To use an interface in your class, append the keyword "implements" after your class name followed by the interface name.

➢ Interface required

To understand the concept of Java Interface better, let see an example. The class "Media Player" has two subclasses: CD player and DVD player. Each having its unique implementation method to play music.

Another class "Combo drive" is inheriting both CD and DVD (see image below). Which play method should it inherit? This may cause serious design issues. And hence, Java does not allow multiple inheritance.

Suppose you have a requirement where class "dog" inheriting class "animal" and "Pet" (see image below). But you cannot extend two classes in Java. So what would you do? The solution is Interface.

➢ When to use Interface and Abstract Class?

- Use an abstract class when a template needs to be defined for a group of subclasses
- Use an interface when a role needs to be defined for other classes, regardless of the inheritance tree of these classe.

Ex,

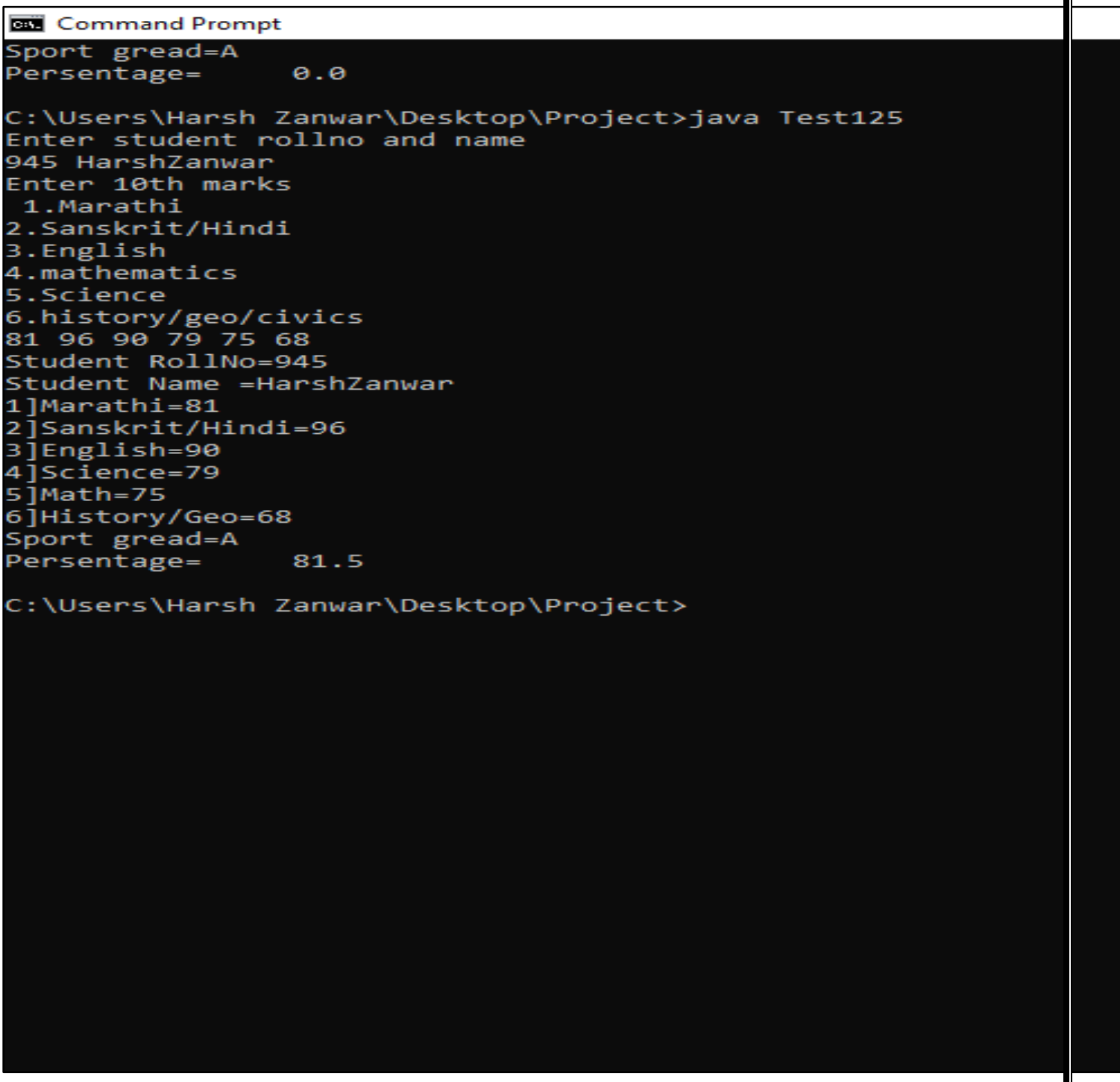
```
import java.util.*;
class StudentInfo
{
        int rollno;
        String name="  ";
        Scanner s1=new Scanner(System.in);
        public void getinfo()
        {
        try{
        System.out.println("Enter student rollno and name");
        rollno=s1.nextInt();
        name=s1.next();
        }catch(Exception e){}
        }
}
 class Marks extends StudentInfo
{
        int marathi,san_hindi,Eng,sci,math,his_geo;
        public void getmarks()
        {
        try{
        System.out.println("Enter 10th marks\n
1.Marathi\n2.Sanskrit/Hindi\n3.English\n4.mathematics\n5.Science\n6.history/geo/civics");
        marathi=s1.nextInt();
        san_hindi=s1.nextInt();
```

```
            Eng=s1.nextInt();
            sci=s1.nextInt();
            math=s1.nextInt();
            his_geo=s1.nextInt();
}catch(Exception e){}
}
}
interface Per
{
             String spgread="A";
             public void display();
}
class DisplayResult extends Marks implements Per
{
            float totalper;
            float persentage;
            public void display()
            {
             System.out.println("Student RollNo="+rollno);
            System.out.println("Student Name ="+name);
            System.out.println("1]Marathi="+marathi);
            System.out.println("2]Sanskrit/Hindi="+san_hindi);
            System.out.println("3]English="+Eng);
            System.out.println("4]Science="+sci);
            System.out.println("5]Math="+math);
            System.out.println("6]History/Geo="+his_geo);
            System.out.println("Sport gread="+spgread);
```

```
        }


        public void displayresult()
        {
        totalper=marathi+san_hindi+Eng+sci+math+his_geo;
        persentage=totalper/600*100;
        System.out.println("Persentage=\t"+persentage);
        }
}
class Test125  extends DisplayResult
{
        public static void main(String args[])
        {
        Test125 t1=new Test125();
        t1.getinfo();
        t1.getmarks();
        t1.display();
        t1.displayresult();
        }
}
```

Output:


Command Prompt
Sport gread=A
Persentage=	0.0
C:\Users\Harsh Zanwar\Desktop\Project>java Test125
Enter student rollno and name
945 HarshZanwar
Enter 10th marks
1.Marathi
2.Sanskrit/Hindi
3.English
4.mathematics
5.Science
6.history/geo/civics
81 96 90 79 75 68
Student RollNo=945
Student Name =HarshZanwar
1]Marathi=81
2]Sanskrit/Hindi=96
3]English=90
4]Science=79
5]Math=75
6]History/Geo=68
Sport gread=A
Persentage=	81.5
C:\Users\Harsh Zanwar\Desktop\Project>

- **Advantages of Interface**

1) through interfaces we can implement multiple inheritance in java.

2) Interfaces function to break up the complex designs and clear the dependencies between objects.

3) Interfaces makes your application loosely coupled

Nested Interface in Java

We can declare interfaces as member of a class or another interface. Such an interface is called as member interface or nested interface.

Syntax:-

```
public interface Outer
 {
    // Body of interface.
    public interface inner
    {
       // Body of interface.
    }
}
```

➢ Conclusion

Inheritance and interfaces are powerful mechanisms in Java. We have seen that UML generalization and realization relationships can be used to visualize these two Java concepts. We have noted a couple of weaknesses of UML in this area, especially when working informally. However, a couple of additional non-standard shorthand conventions can help UML class diagrams continue to communicate clearly the overall structure of classes, interfaces, and the different relationships between them. The next article in the series will introduce one of the main ways of extending UML, and look at one way we can use it to help us build better object models. I'll leave you with a quote from Peter Coad's book:

➢ REFERENCE:

1] Interface (Java) - Wikipedia

en.wikipedia.org › wiki › Interface_(Java).

2]Text Max

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that **Mr.Harsh Santosh Zanwar** Roll No **945** of **4th** Semester of Diploma in **INFORMATION TECTNOLOGY** of Institute, GOVERNMENT POLYTECHNIC has completed the **Micro Project satisfactorily** in Subject -**java (22412)** for the academic year 2020-2021 as prescribed in the curriculum.

Place: **Nanded**

Date: ……

Subject Teacher	Head of the Department	Principal
Mr.S.G.Mundhe	**Mr.S.N.Dhole**	**Dr.G.V.Garje**

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that **Mr.Aditya Pramod Joshi** Roll no **953** of **4th** Semester of Diploma in **INFORMATION TECTNOLOGY** of Institute, GOVERNMENT POLYTECHNIC has completed the **Micro Project satisfactorily** in Subject -**java (22412)** for the academic year 2020-2021 as prescribed in the curriculum.

Place: **Nanded**

Date: ……

Subject Teacher	Head of the Department	Principal
Mr.S.G.Mundhe	**Mr.S.N.Dhole**	**Dr.G.V.Garje**

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that **Mr.Md Hifaz Khan** Roll no **949** of **4th** Semester of Diploma in **INFORMATION TECTNOLOGY** of Institute, GOVERNMENT POLYTECHNIC has completed the **Micro Project satisfactorily** in Subject -**java (22412)** for the academic year 2020-2021 as prescribed in the curriculum.

Place: **Nanded**

Date: ……

Subject Teacher	Head of the Department	Principal
Mr.S.G.Mundhe	**Mr.S.N.Dhole**	**Dr.G.V.Garje**

MAHARASHTRA STATE

BOARD OF TECHNICAL EDUCATION

Certificate

This is to certify that **Mr.Amaan Khan Pathan** Roll no **948** of **4th** Semester of Diploma in **INFORMATION TECTNOLOGY** of Institute, GOVERNMENT POLYTECHNIC has completed the **Micro Project satisfactorily** in Subject -**java (22412)** for the academic year 2020-2021 as prescribed in the curriculum.

Place: **Nanded**

Date: ……

Subject Teacher	Head of the Department	Principal
Mr.S.G.Mundhe	**Mr.S.N.Dhole**	**Dr.G.V.Garje**