

COURSE : CIS 104

Lab 10

INTERFACES

```
public interface Grading {  
    public void computeGrade();  
  
}  
  
public abstract class Student implements Grading{  
    private String name;  
    private String Id;  
    private int totalMarks;  
  
    public Student() {  
        this.name = " ";  
        this.Id = " ";  
        this.totalMarks = 0;  
    }  
    public Student(String name, String id, int totalMarks) {  
        this.name = name;  
        this.Id = id;  
        this.totalMarks = totalMarks;  
    }  
    public String getName() {  
        return name;  
    }  
    public void setName(String name) {  
        this.name = name;  
    }  
    public String getId() {  
        return Id;  
    }  
    public void setId(String id) {  
        Id = id;  
    }  
    public int getTotalMarks() {  
        return totalMarks;  
    }  
}
```

```
public void setTotalMarks(int totalMarks) {
    this.totalMarks = totalMarks;
}
@Override
public String toString() {
    return "Student [name=" + name + ", Id=" + Id + ",
totalMarks=" + totalMarks + "]";
}

public void display()
{
    System.out.println("The name is: " + name);
    System.out.println("The Id is: " + Id);
    System.out.println("The total marks is: " + totalMarks );
}

public abstract void computeGrade();
}

public class underGradStud extends Student{
    private boolean coop;;

    public underGradStud(String name, String Id, int totalMarks,
boolean coop)
    {
        super(name, Id, totalMarks);
        this.coop = coop;
    }

    public void computeGrade()
    {
        if( getTotalMarks()>=60)
            System.out.println("Pass");
        else
            System.out.println("Fail");
    }
}
```

```
public class postgradStud extends Student{
    private boolean teach;

    public postgradStud(String name, String Id, int totalMarks, boolean
teach)
    {
        super(name, Id,totalMarks );
        this.teach = teach;
    }

    public void computeGrade()
    {
        if(getTotalMarks() >=70)
            System.out.println("Pass");
        else
            System.out.println("Fail");
    }

}
```

```
public class Kids implements Grading{

    private String name;
    private int nbAnswers;

    public Kids(String name, int nbAnswers)
    {
        this.name = name;
        this.nbAnswers= nbAnswers;
    }

    public void computeGrade()
    {
        if( nbAnswers>= 100)
            System.out.println("You are the Winner.");
        else
            System.out.println("You lost.");
    }

}
```

```
public class Test_Student {
    public static void main(String[] args)
    {

        Grading Arr[] = new Grading[6];

        underGradStud un1 = new underGradStud("Ali", "1111", 16, true);
        underGradStud un2 = new underGradStud("Naef", "2222", 78, false);

        postgradStud grad1= new postgradStud("Bader", "3333", 78, true);
        postgradStud grad2= new postgradStud("Ahmad", "4444", 65, false );

        Kids kk1 = new Kids("Samy", 120);
        Kids kk2 = new Kids("Bader", 23);

        Arr[0] = un1;
        Arr[1] = un2;
        Arr[2] = grad1;
        Arr[3] = grad2;
        Arr[4] = kk1;
        Arr[5] = kk2;

        System.out.println("The results are: ");

        for(int i =0; i< Arr.length; i++)
        {
            Arr[i].computeGrade();
        }

        int count = 0;

        for(int i =0; i< Arr.length; i++)
        {
            if(Arr[i] instanceof postgradStud)
                count++;
        }

        System.out.println("The number of PostGraduate Students is: " +
        count);

    }
}
```