Inheritance and Abstract classes

Exercise-01

```
public interface MyFirstInterface {
   int x=67;
   //final public static x=6;
   //all variables declared inside an interface are by default final public static whether it is imlicitly stated or not
   void display();
  //abstract void display();
  //all the method declared inside an interface are abstract
   }

public class InterfaceImplemented implements MyFirstInterface{
   public void display()
   {
      x++;
      System.out.println("x="+x);
      //value of x cant be changed beacuse its a by default final variable
     }
   }
```

Exercise-02

Main class

```
public class Practical052 {

   public static void main(String[] args) {
      Speaker p=new Politician();
      p. speak("Vote me");
```

```
        Speaker p2=new Priest();
        p2. speak("Bless you");

        Speaker p3=new Lecturer();
        p3. speak("Good morning");
       }
}
```

Speaker interface

```
interfacer Speaker {
  int i=100;
  void speak(String line);
    }
```

Politician class

```
public class Politician implements Speaker {
   @Override
   public void speak(String phrase)
   {
      System.out.println(i+" Politician "+phrase);
   }
}
```

Priest class

```
public class Priest implements Speaker{
   @Override
   public void speak(String phrase)
   {
      System.out.println(i+" Priest "+phrase);
   }
}
```

Lecturer class

```
public class Lecturer implements Speaker{
   @Override
   public void speak(String phrase)
   {
      System.out.println(i+" Lecturer "+phrase);
   }
}
```

Exercise-03

Student class

```
final class Student
 {
  final int marks = 100;
  //This shows an error because following doesn't have a method body
  final void display();
}
```

Undergraduate class

```
/*This cannot inherit from Student class because
Subclasses can't inherit a final class or a final class cannot be inherited by any subclass.
So, we can restrict class inheritance by making use of a final class.*/
public class Undergraduate extends Student
 {
 }
```

Exercise-04

Main class

```
public class Exercise4 {
public static void main(String[] args) {
        Rectangle r=new Rectangle(23,45);
        r.calculatearea();
        r.display();

        Circle c=new Circle(5);
        c.calculatearea();
        c.display();
        }
}
```

Shape class

```
abstract class  Shape {
   abstract  void calculatearea();
   public void dispaly()
   {
    System.out.println("Shape ");
   }
}
```

Rectangle class

```
public class Rectangle extends Shape {
   private double width,length,area;
   public Rectangle(double length,double width)
   {
    this.length=length;
    this.width=width;
   }
   @Override
```

```
  public void calculatearea()
  {
    area=length*width;
  }

  public void display()
  {
     System.out.println("Area of rectangle is:"+area);
  }
}
```

Circle class

```
public class Circle extends Shape{
    double r,Area;
    public Circle (double r)
    {
       this.r=r;
    }
    @Override
   public void calculatearea()
   {
      Area=Math.PI*r*r;
   }
   public void display()
   {
      System.out.println("Area of circle is: "+Area);
   }
}
```