Practical 07: Inheritance & Abstract Classes

Exercise 01:

Try following code. What is the outcome? Why?

Class 01:

```
final class Student {
        final int marks = 100;
        final void display();
}
```

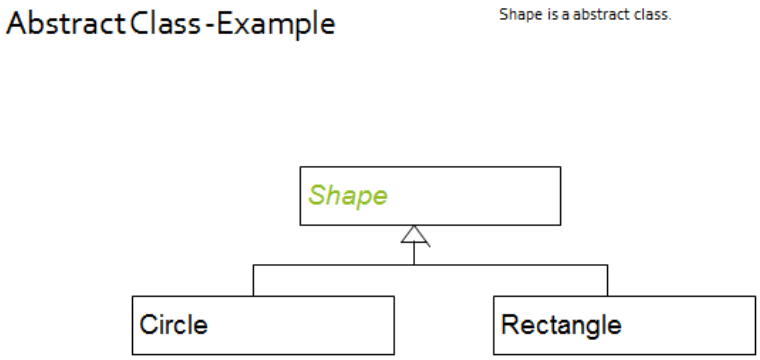
Class 02:

```
class Undergraduate extends Student{}
```

- This code gives a compilation error. Because class 01 is declared as final and class 02 is extended class 01. Since the final classes cannot be sub-classed this code will result in a compilation error.

Exercise 02:

Develop a code base for the following scenario. Shape class contains an abstract method called "calculateArea" and non-abstract method called "display". Try to pass required values at the instantiation. Recall what we have done at the lecture…




```
public abstract class Shape
{
   public abstract double calculateArea();

   public void display()
   {
   }
```

```
}

public class Circle extends Shape
{
   private double r;
   protected static final double pi=3.14159;

   public void circle(double r)
   {
      r=this.r;
   }
   public double calculateArea()
   {
      return pi*r*r;
   }
}

public class Rectangle extends Shape
{
   private double h,w;

   public void rect(double h,double w)
   {
      h=this.h;
      w=this.w;
   }
   public double calculateArea()
```

```
    {
        return w*h;
    }
}
```