

Programming Lab Exercise 8

Before you start:

Create a folder called **lab8** inside your personal **java** folder you created at the start. Save all your work for lab8 in this folder.

Purpose:

Tests your understanding of the concept of abstract classes and inheritance in Java.

Q1.

- a) Download the source files in the **Lab8** folder from Blackboard.
- b) Compile them and run the `Driver` program.
- c) Examine everything that is going on in the various files.
- d) It has been decided that every `shape` should also have a `colour` associated with it.
- e) Make the necessary changes to allow for this.
- f) Modify the code so that the colour of the objects also gets printed out.

New output should begin:

```
---Using circle reference---  
Shape Name = Circle One  
Shape colour = Red  
Radius = 10.0  
  
---Using rectangle reference---  
Shape Name = Rectangle One  
Shape colour = Yellow  
Length = 15.0  
Breadth = 20.0  
  
---Using cylinder reference---  
Shape Name = Cylinder One  
Shape colour = Green  
Radius = 6.0  
Height = 8.0
```

- g) Create a new subclass called `Triangle`
 - a. It should have 2 member variables: `base` (double) and `height` (double).
 - b. The constructor should initialise both these variables and the colour of the triangle using a parameter list.
 - c. Note $\text{area} = 0.5 * \text{base} * \text{height}$ (we are assuming a right angled triangle).
- h) Add the necessary statements to the driver program which creates and prints a `Triangle` object.

Q2.

Create an abstract class called `Person` with two concrete subclasses `Employee` and `Student`. `Person` has a `String` to store the name and a method returning a `String` called `getName`. It has a constructor that takes the name as a parameter. It also has an abstract method `getDescription`.

`Employee` has a constructor that takes the name and annual salary as parameters. It has a `getDescription` method that returns the `String` "An employee with a salary of " followed by the annual salary.

`Student` has a constructor that takes the name and the course they are studying as parameters. It has a `getDescription` method that returns the `String` "A student studying " followed by the course.

A `TestPerson` class creates a polymorphic array of type `Person` with one `Employee` and one `Student`. Using the `getName` and `getDescription` methods, print the name and description of the two elements in the array.