Abstract class Activity

1. Create **abstract** class **Shape**
   Datatype – **int** , field name: **value**

   Use getter and setter methods
   Create abstract method name **calculateArea(int value)**.

   Create the class **"Circle"** extends the **Shape**. Use **calculateArea(int value)**. Find area of circle

   Create the class **"Square"** extends the Shape. Use **calculateArea(int value)**. Find area of rectangle.

   Create a **ShapeMain** class to access the Square and Circle class and test in main method.

   Sample Input
   Circle
   Square
   Enter the shape
   Circle
   Enter the radius
   25

   Sample output
   Area of circle is: 1962.50

   Sample Input
   Circle
   Square
   Enter the shape
   Square

Enter the side
23

Sample output
Area of Square is: 529.00

2. Create abstract class as **“Match”**.
   Add the following private members.

Data type	Field name
Int	currentscore
Float	currentover
Int	target

   Use getter and setter methods.

   Create another class ODIMatch that extends the Match. (50 overs)

   Create another class TestMatch that extends the Match. Consider test match is in the last day. (90 Overs)

   Create another class T20Match that extends the Match. (20 Overs)

Create the following abstract methods in the Match Class. Implement it in other respective class.

1. float calculateRunRate()
2. int calculateBalls()
3. void display(double reqRunrate, int balls)

Create the MatchMain class and the inherited class using main method. Calculate the required runrate and number of balls for each derived class.

Sample Input and Output:

Enter the match format

1. ODI
2. T20
3. Test

1

Enter the Current score

256

Enter the current Over

30

Enter the Target Score

400

Requirements:

Need 144 runs in 120 balls

Required Runrate: 7.20

2nd Sample Input and Output:

Enter the match format

1. ODI
2. T20
3. Test

2

Enter the Current score

120

Enter the current Over

15

Enter the Target Score

170

Requirements:

Need 50 runs in 30 balls

Required Runrate: 10.00