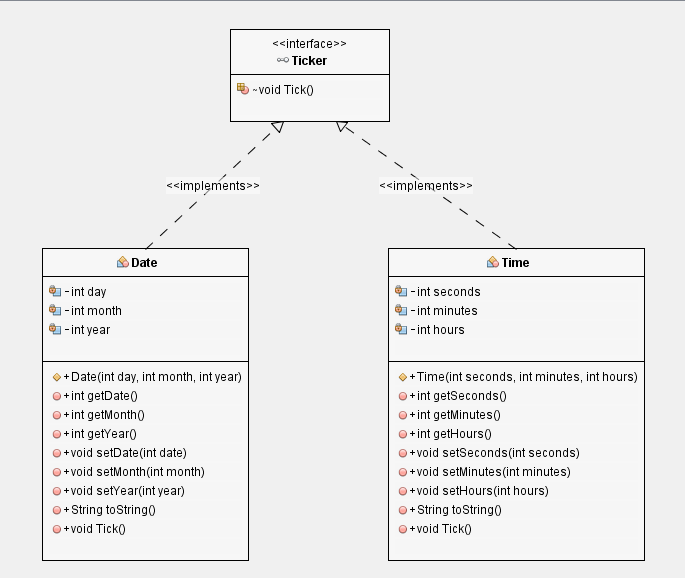
# **Course: Object Oriented Programming**

# **Lab 9b**

**Interfaces**

**Task 1:**

Convert following UML diagram to java code.



**Test Class:**

Tick method in Time class

Increment in seconds and if second is equal to 60 rest it 0 and incremnt in minutes. If minutes is equal to 60 rest minutes to 0 and increment in hours. If hours is equal to 24 rest it to 0.

Tick method in Date class

Increment in days,if day is equal to 30 rest it 1 and incremnt in month. If month is equal to 12 rest month to 1 and increment in year.

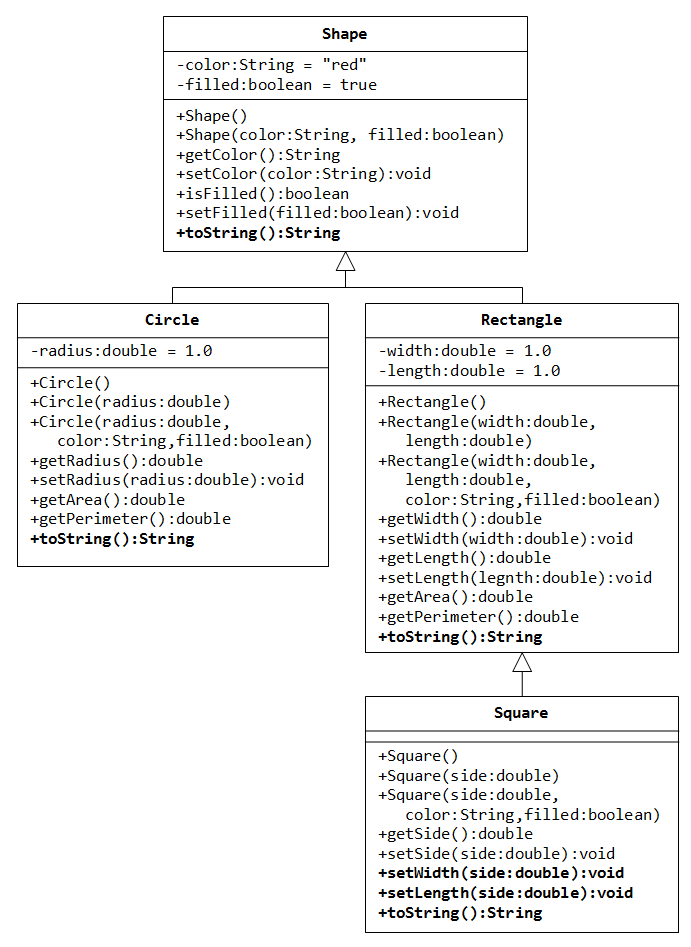
Check expected output and create object of date and time class accordingly and call *toString()* for both.

Now set data and time and and call *toString().*

Now create polymorphic reference of **Ticker** interface with name dateTime and call tick method for date and time and then call *toString().*

****

**Task 2:**



Use the above class hierarchy and modify the code as follows:

1. The Stats interface: Create an interface named **Stats** containing 2 methods called **computeArea() and reset()**.
   * double computeArea();
     + The computeArea method returns the area
   * void reset();
     + The reset method resets all those fields to 0, which are involved in computing the stats (i.e. area)
   * The interface also contains a **string field** called units, which represent the units of area (e.g. square kilometer – “sq.km” or square meter – “sq.m” etc.)
2. The Shapes Hierarchy: The **Shape** class should now implement the Stats interface. But, the concrete implmentation of computeArea() and reset() methods should be provided in the **Rectangle** class.
   * The computeArea method should print the area of rectangle along with units.
   * The reset method should reset width and height to 0.
3. The Country Class: Create a class called country, which has following fields:
   * name
   * provincesAreas (an array of integers), inline initialization
   * statesAreas (an arrayList of integers), initialization in constructor
   * This class implements the computeArea() method of Stats interface. The area of country is computed by adding together all the areas of provinces and states.
   * This class also implments the reset() method of Stats interface. Reset all the values of provincesAreas and statesAreas to 0.
4. The Test Class:Create a test class named TestStats,
   * Create a static method named
     + Void printStats(interface i);
     + Which calls the computeArea method to print the area of passed object. Before exiting the method, also call the reset method to reset the object statistics.
   * Inside the main method, create 3 objects of Rectangle, Square and Country classes, polymorphically
     + Call the printStats method for each of the created objects to print its area and reset its stats using an enhanced for.