**Tutorial 2: Advanced OOP**

**Req1:** Create a “Shape” abstract class, which contains the following attributes and methods



* Two protected instance variables color(String) and filled(boolean). The protected variables can be accessed by its subclasses and classes in the same package. They are denoted with a '#' sign in the class diagram.
* Getter and setter for all the instance variables, and toString().
* Two abstract methods getArea() and getPerimeter() (shown in italics in the class diagram).

Write main method to initiate 3 objects, 1 for circle, 1 for Rectangle and 1 for Square, initiate with corresponding constructor and use toString() method to print out perimeter and area of each shape.

**Req 2:**

Provide an interface Measurable with a method double getMeasure() that measures an object in some way. Make Employee implement Measurable. The method getMeasure() will return salary of employee object.

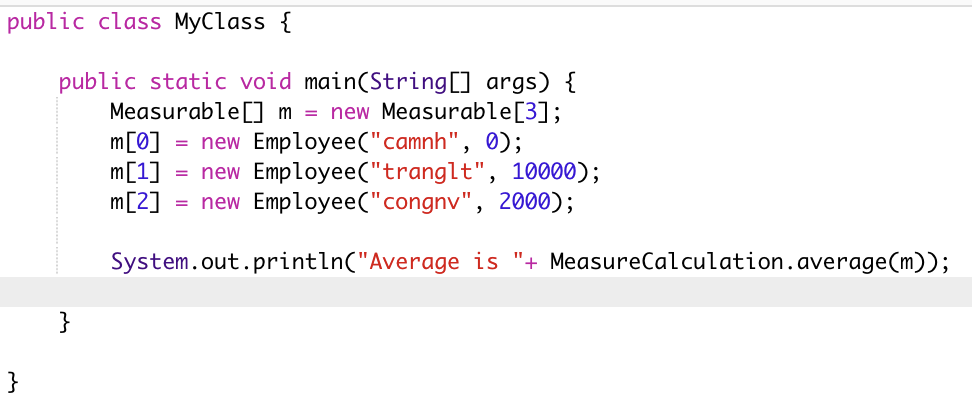
Attribute of Employee includes:

+ String name;

+ double salary;

Create a class MeasureCalculation which provides a method double average(Measurable[] objects) that computes the average measure. Use it to compute the average salary of an array of employees.

Example output

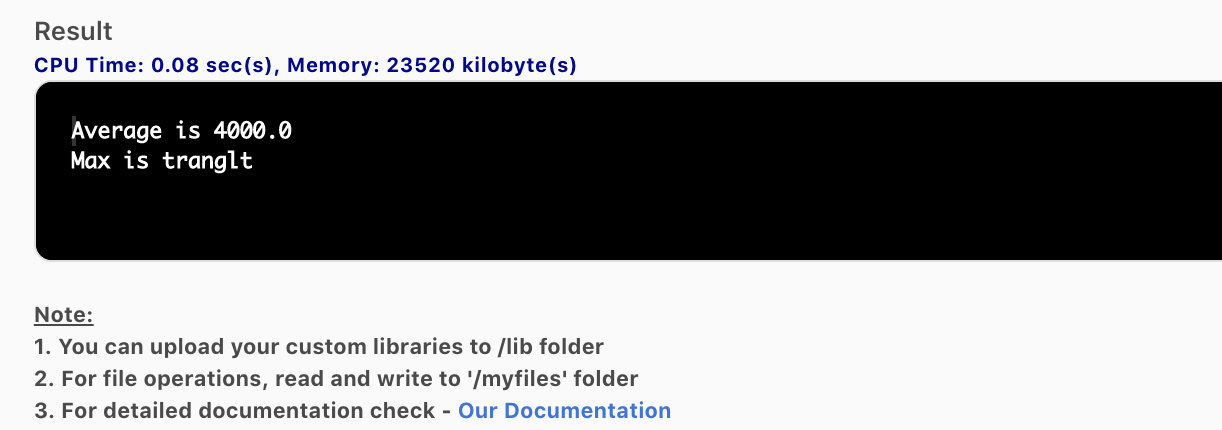




**Req 3**

Continue with the preceding exercise and provide a method Measurable largest(Measurable[] objects)to class MeasureCalculation to find highest value. Use it to find employee name with highest salary among employees. Why do you need a cast?

**Example output**

****

**Req 4**

Create new class call BankAccount with the following attribute

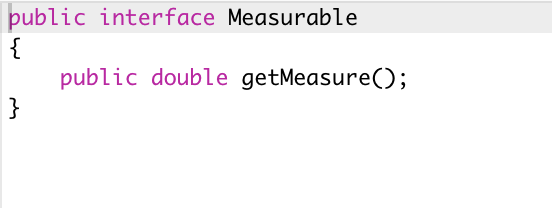
+ String accountName;

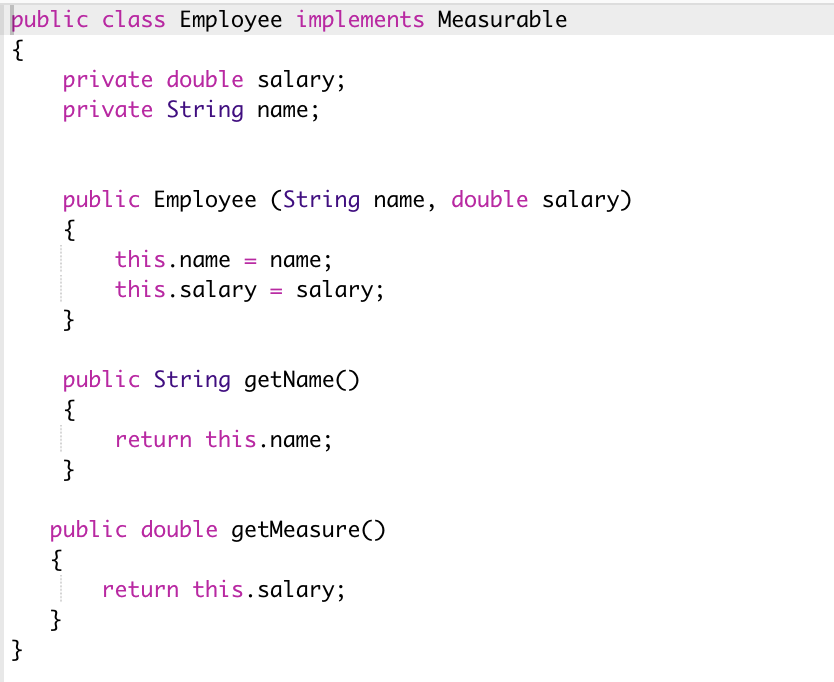
+ double balanceNumber;

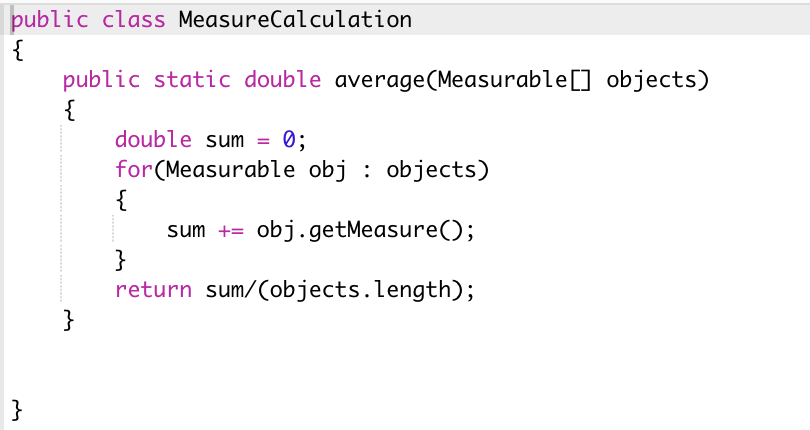
In order to integrate with existing codebase structure with minimal modification, what is the pre-condition for creating this class. If this class exist, make changes in needed parts to integrate with existing codebase with minimal modifications.

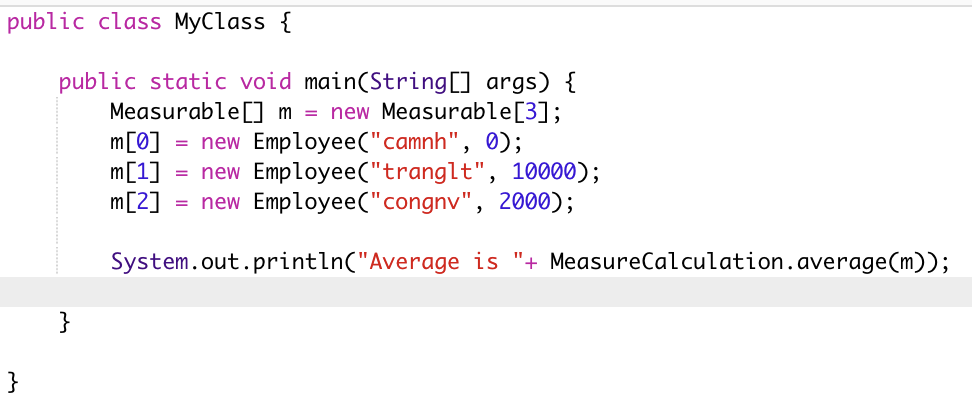
**Answer**

**Req 2**

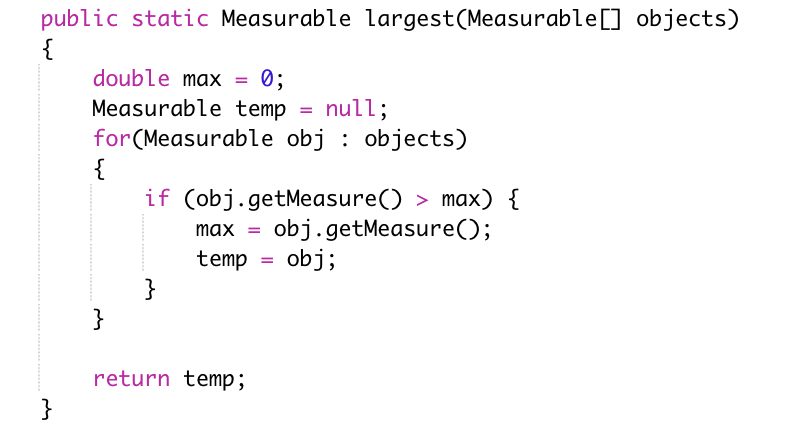
****

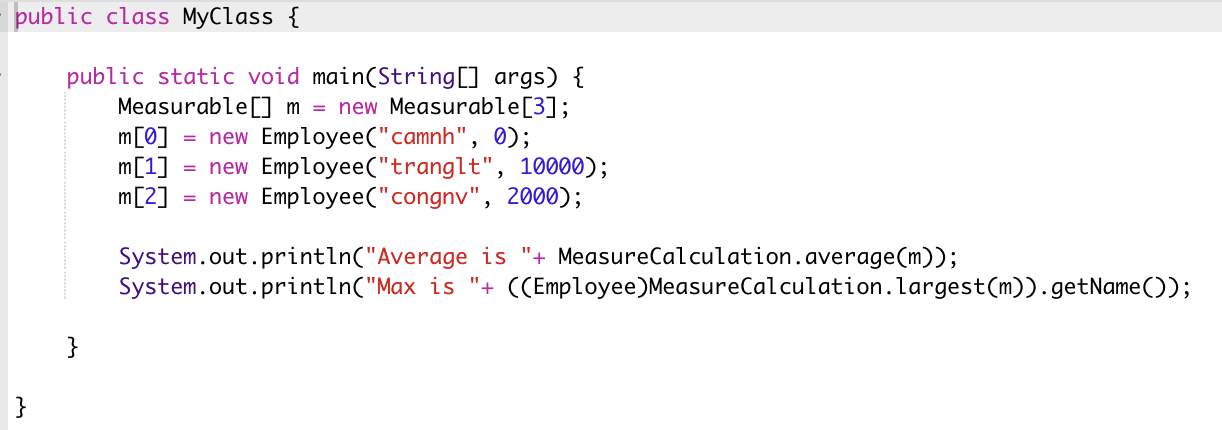
****

****



**Req 3**

****

****

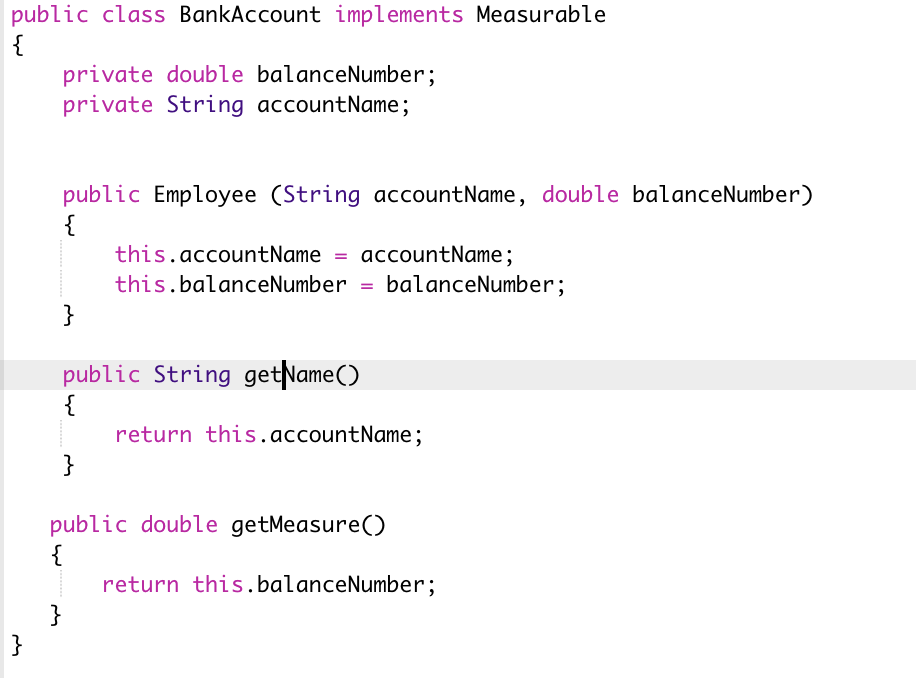
The reason why we need a cast is because return type of method largest()is an interface, which doesn’t cast to specific class, so we need a cast to convert from

Interface -> class

**Req4:**

The BankAccount class needs to implement Measurable interface

Create BankAccount



Only change class Main, no need to change others

