OOPS ASSIGNMENT 2

1. Create a class named 'Student' with String variable 'name' and integer variable 'roll_no'. Assign the value of roll_no as '2' and that of name as "John" by creating an object of the class Student.

2. Assign and print the roll number, phone number and address of two students having names "Ram" and "Sanjay" respectively by creating two objects of class 'Student'.

3. Write a program to print the area and perimeter of a triangle having sides of 3, 4 and 5 units by creating a class named 'Triangle' without any parameter in its constructor..

4. Write a program to print the area and perimeter of a triangle having sides of 3, 4 and 5 units by creating a class named 'Triangle' with constructor having the three sides as its parameters.

5. Write a program to print the area of two rectangles having sides (4,5) and (5,8) respectively by creating a class named 'Rectangle' with a method named 'Area' which returns the area and length and breadth passed as parameters to its constructor.

6. Write a program to print the area of a rectangle by creating a class named 'Area' having two methods. First method named as 'setDim' takes length and breadth of rectangle as parameters and the second method named as 'getArea' returns the area of the rectangle. Length and breadth of rectangle are entered through keyboard.

7. Write a program to print the area of a rectangle by creating a class named 'Area' taking the values of its length and breadth as parameters of its constructor and having a method named 'returnArea' which returns the area of the rectangle. Length and breadth of rectangle are entered through keyboard.

8. Print the average of three numbers entered by user by creating a class named 'Average' having a method to calculate and print the average.

9. Print the sum, difference and product of two complex numbers by creating a class named 'Complex' with separate methods for each operation whose real and imaginary parts are entered by user.

10. Write a program that would print the information (name, year of joining, salary, address) of three employees by creating a class named 'Employee'. The output should be as follows:

Name	Year of joining	Address
Ram	1994	64C- Bhopal
Praveen	2000	68D- Bhopal
Yash	1999	26B- Bhopal

11. Add two distances in inch-feet by creating a class named 'AddDistance'.

12. Write a program by creating an 'Employee' class having the following methods and print the final salary.
1 - 'getInfo()' which takes the salary, number of hours of work per day of employee as parameter
2 - 'AddSal()' which adds Rs 10 to salary of the employee if it is less than Rs 500.
3 - 'AddWork()' which adds Rs 5 to salary of employee if the number of hours of work per day is more than 6 hours.

13. Create a class called 'Matrix' containing constructor that initializes the number of rows and number of columns of a new Matrix object. The Matrix class has the following information:
1 - number of rows of matrix
2 - number of columns of matrix
3 - elements of matrix in the form of 2D array

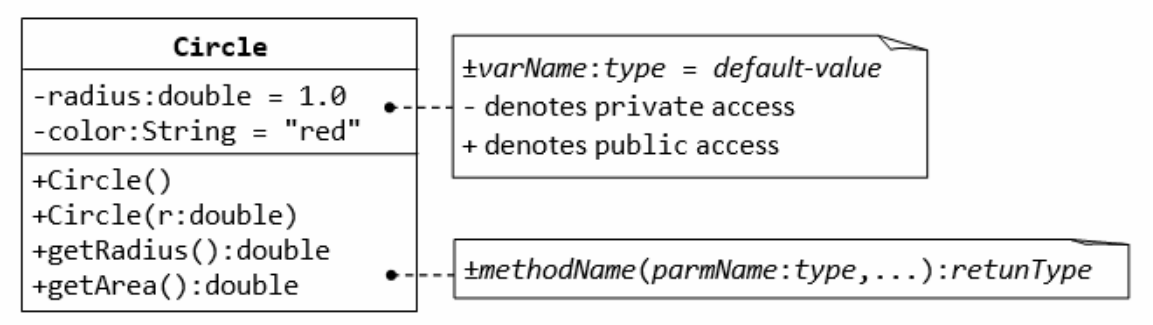
14. The Matrix class has methods for each of the following:
1 - get the number of rows
2 - get the number of columns
3 - set the elements of the matrix at given position (i,j)
4 - adding two matrices. If the matrices are not addable, "Matrices cannot be added" will be displayed.
5 - multiplying the two matrices

15. Create a class called Student that stores its name, class, section, and marks in four subjects. Write functions to accept, display and get/set of the data. Also write methods that return its total and percentage. Test the above code by creating an implementation program.

16. Create a class that counts the number of its objects created. If the object counter is equal to 5, it should display a message “Too Less…” on the screen at the time of object creation. Like when its is equal to 5 then “Will Do…” and if its more than 5 then it should display “Exceeding the Limits…” Write suitable constructors and methods. Also write methods that returns the number of objects created.

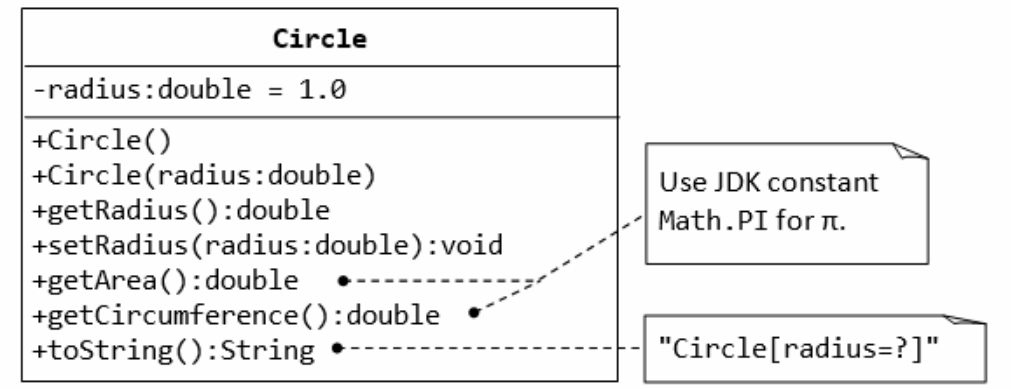
OOPS MODULE-2

1.The Circle Class (An Introduction to Classes and Instances)

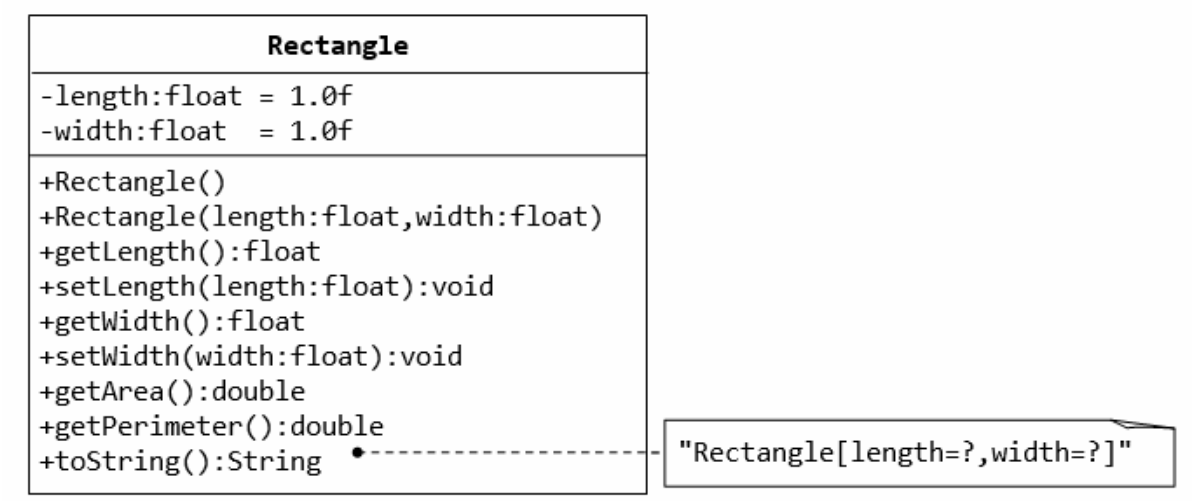


A class called **circle** is designed as shown in the following class diagram. It contains:

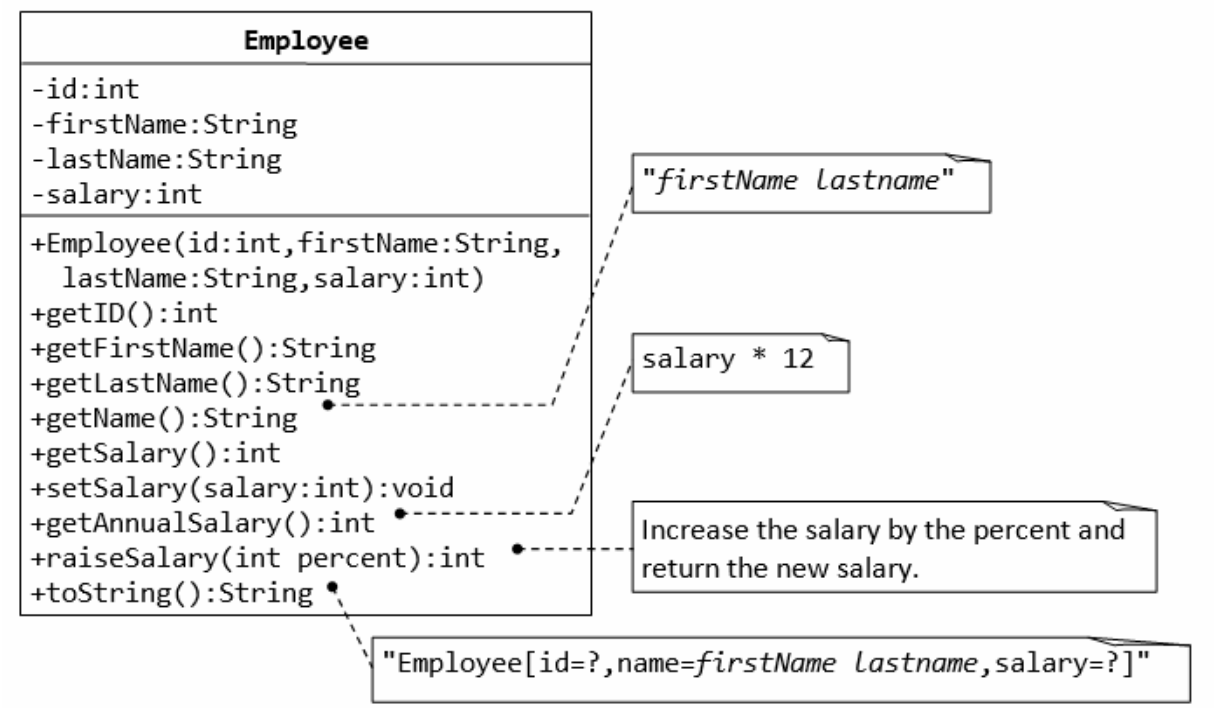
- Two private instance variables: radius (of the type double) and color (of the type String), with default value of 1.0 and "red", respectively.
- Two *overloaded* constructors - a *default* constructor with no argument, and a constructor which takes a double argument for radius.
- Two public methods: getRadius() and getArea(), which return the radius and area of this instance, respectively.



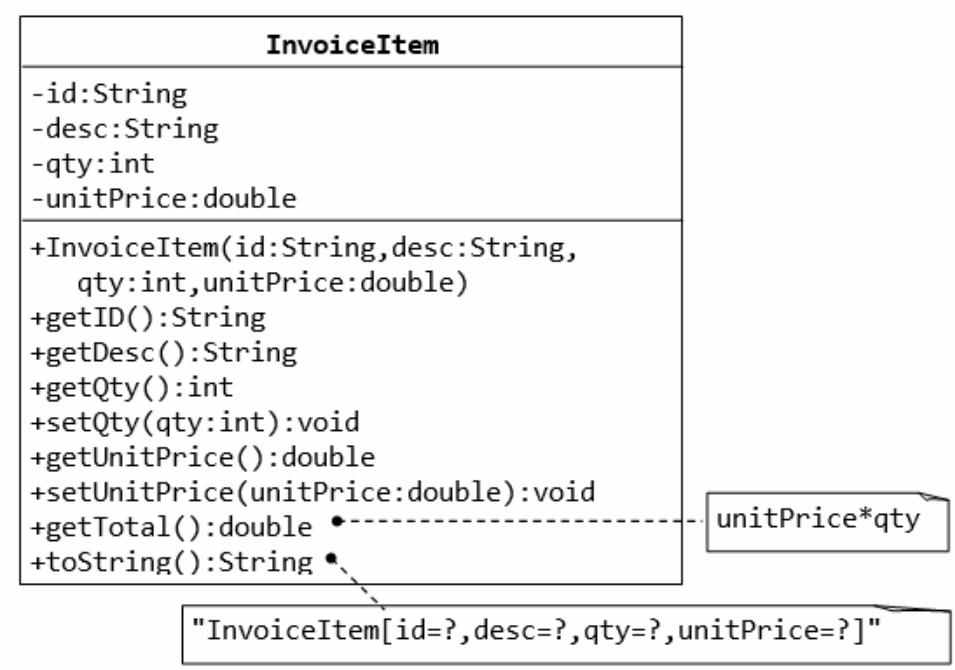
2 The Rectangle Class



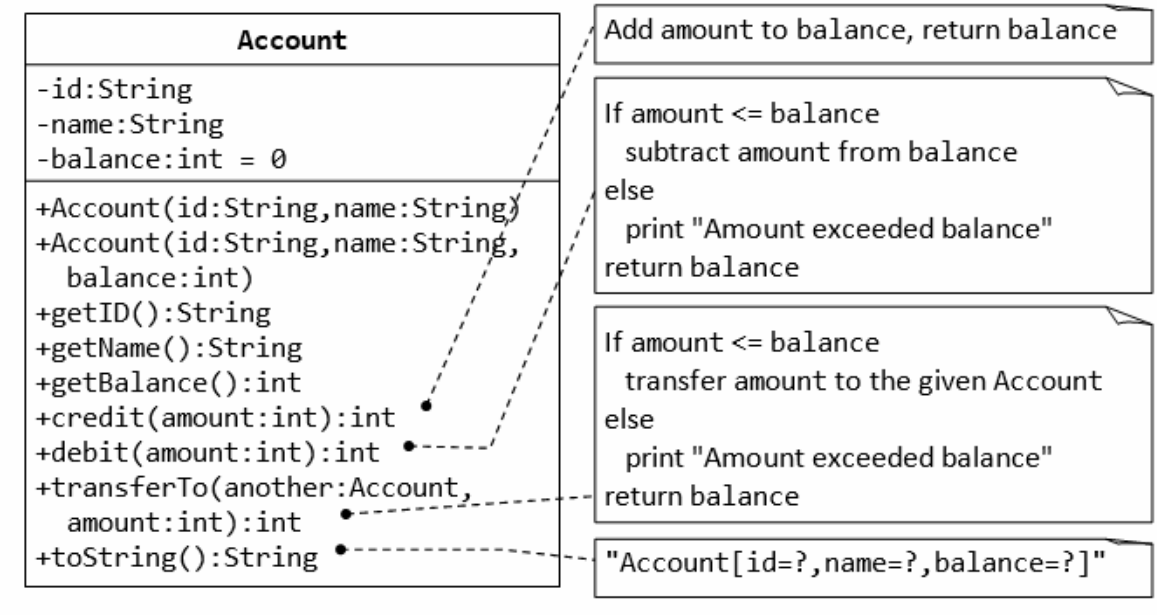
3.The Employee Class



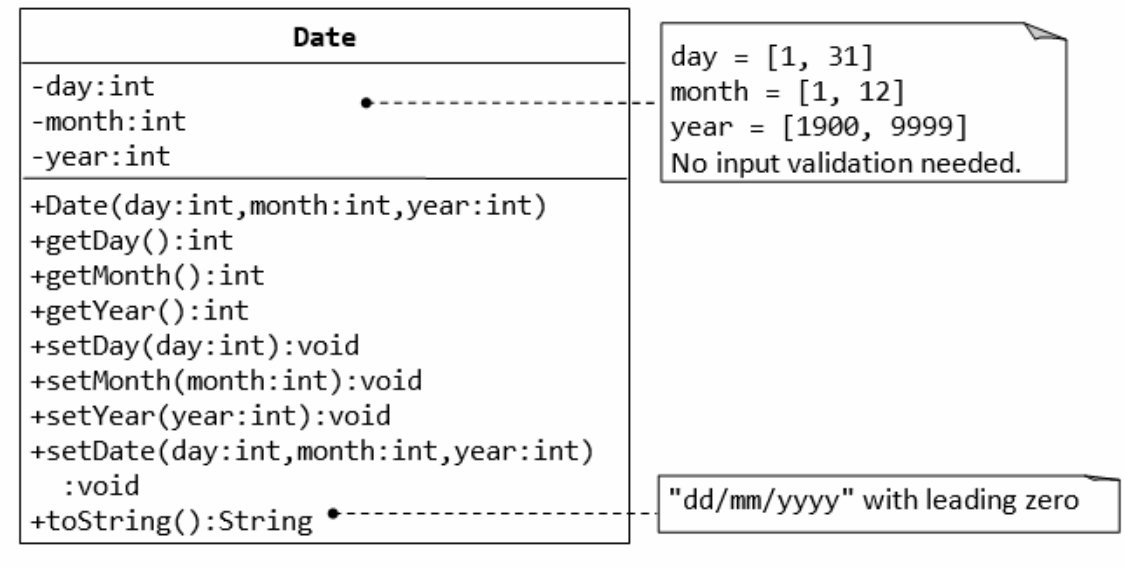
4.The InvoiceItem Class



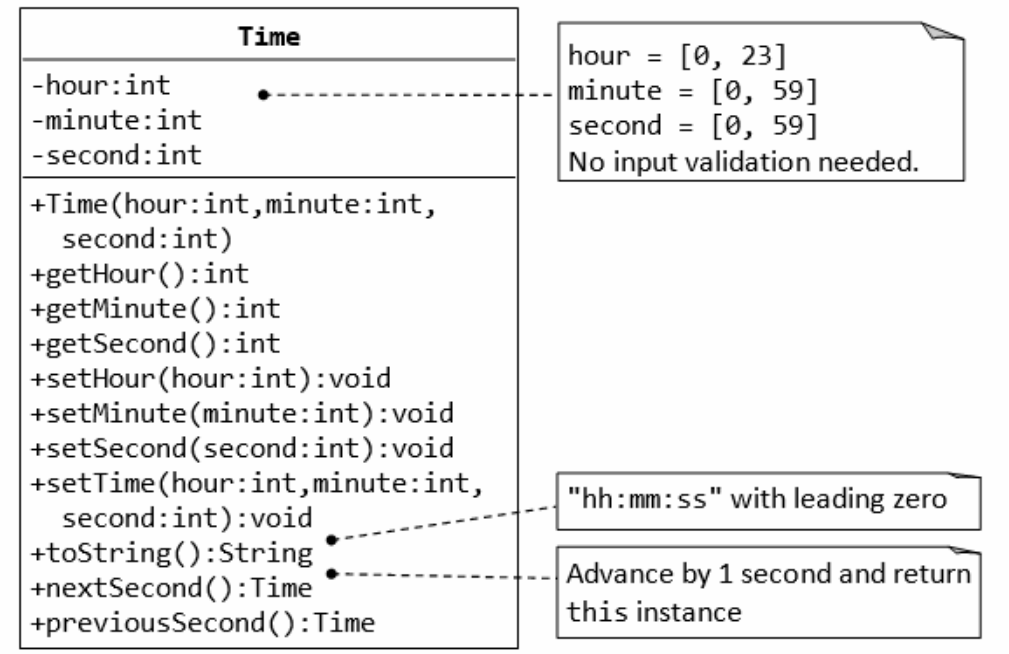
5. The Account Class



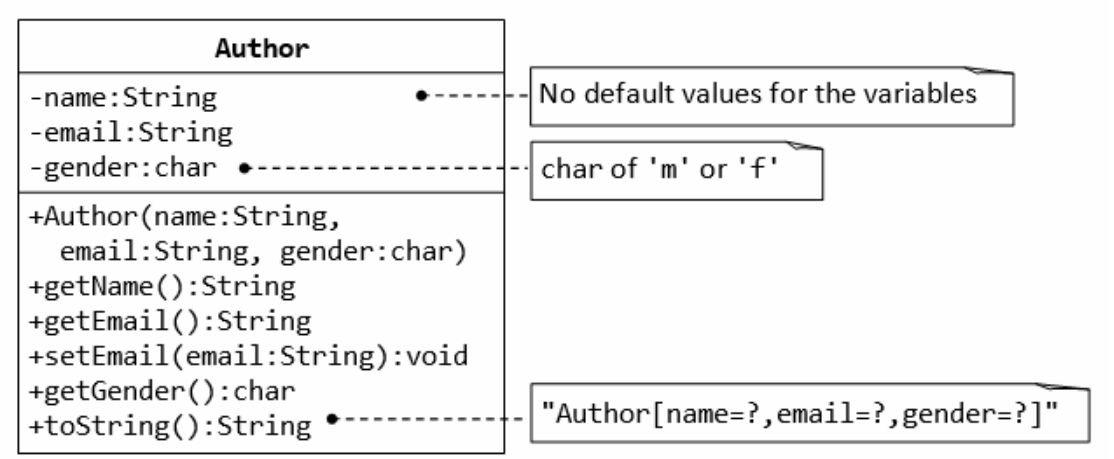
6. The Date Class

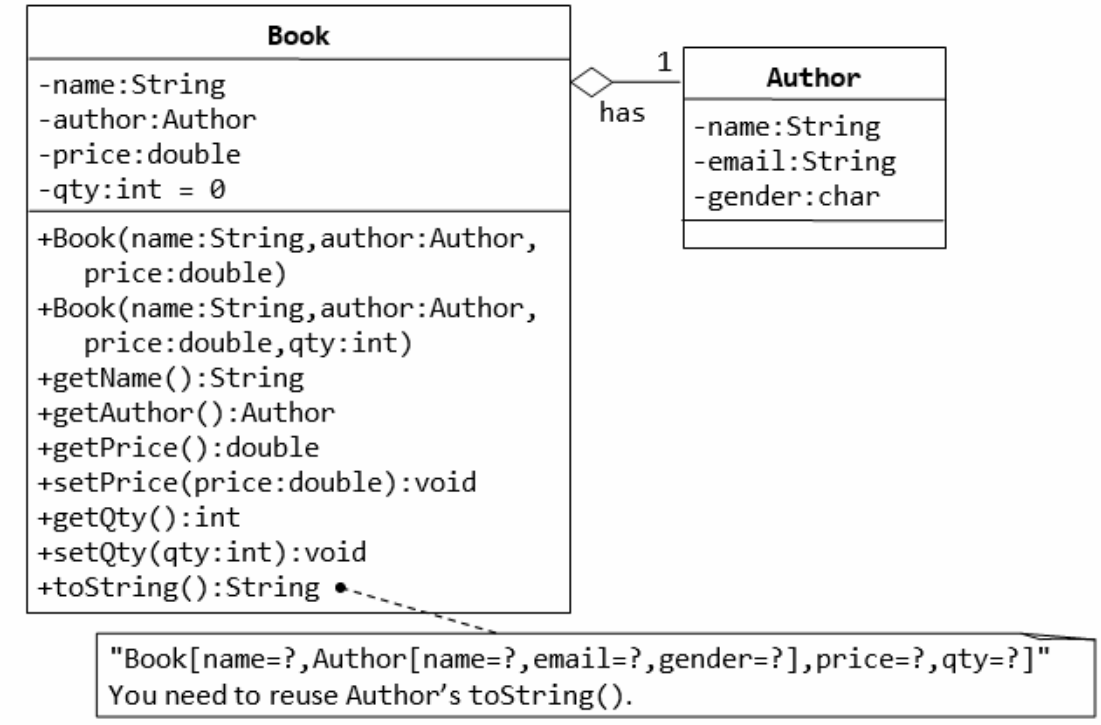


7. The Time Class

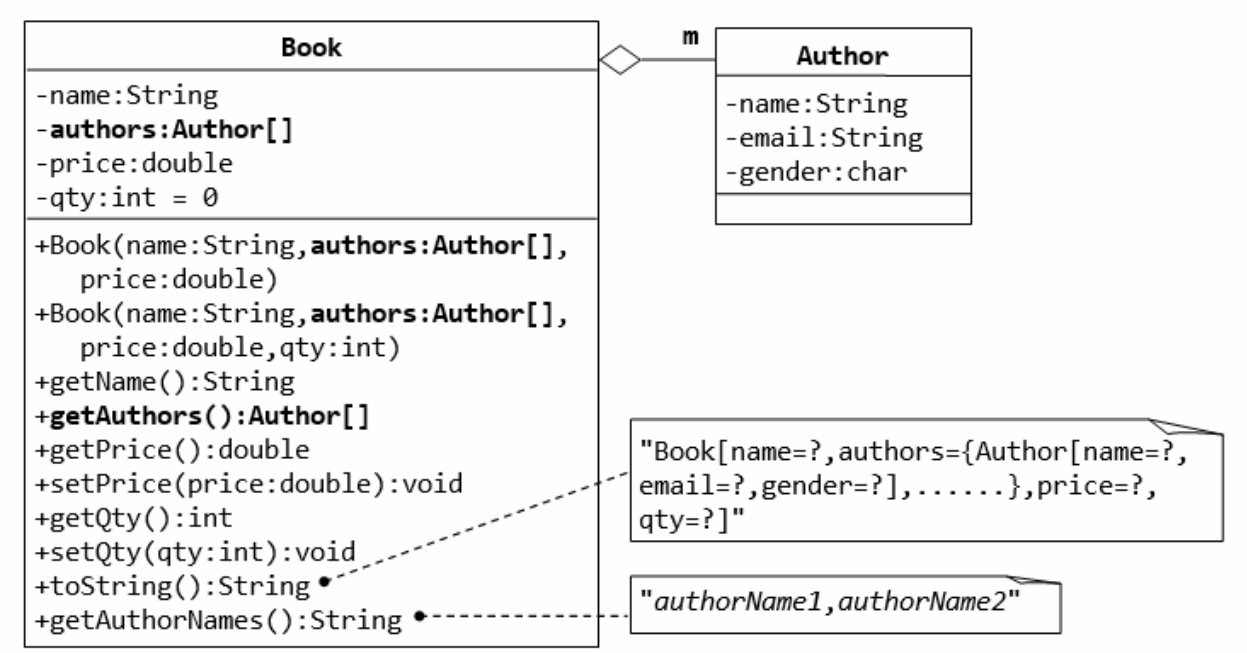


8. The Author and Book Classes

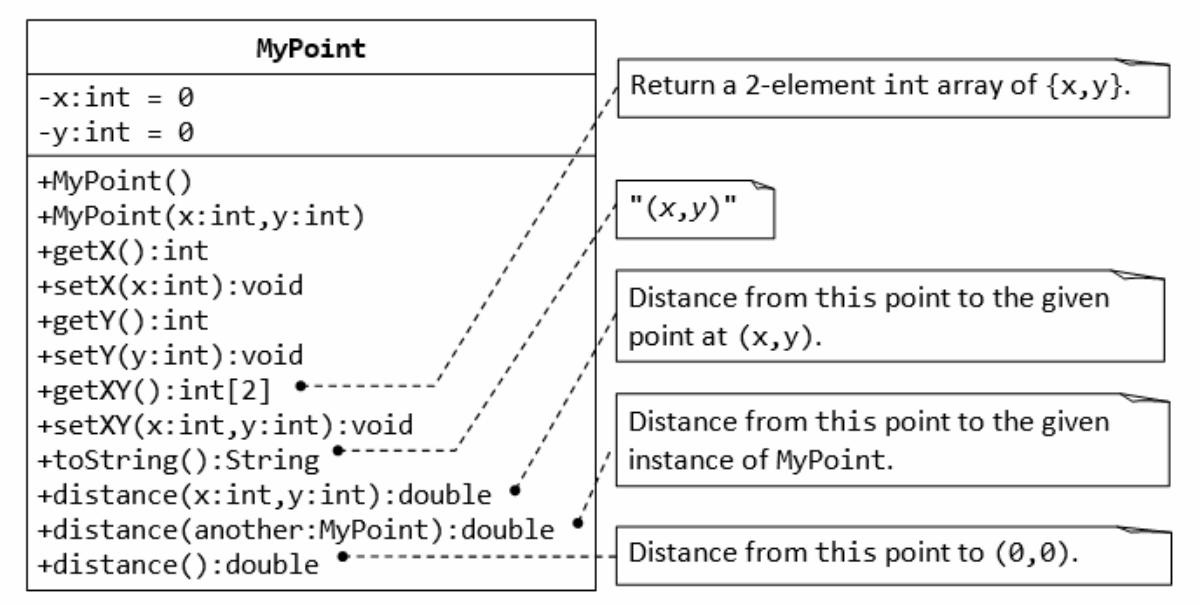




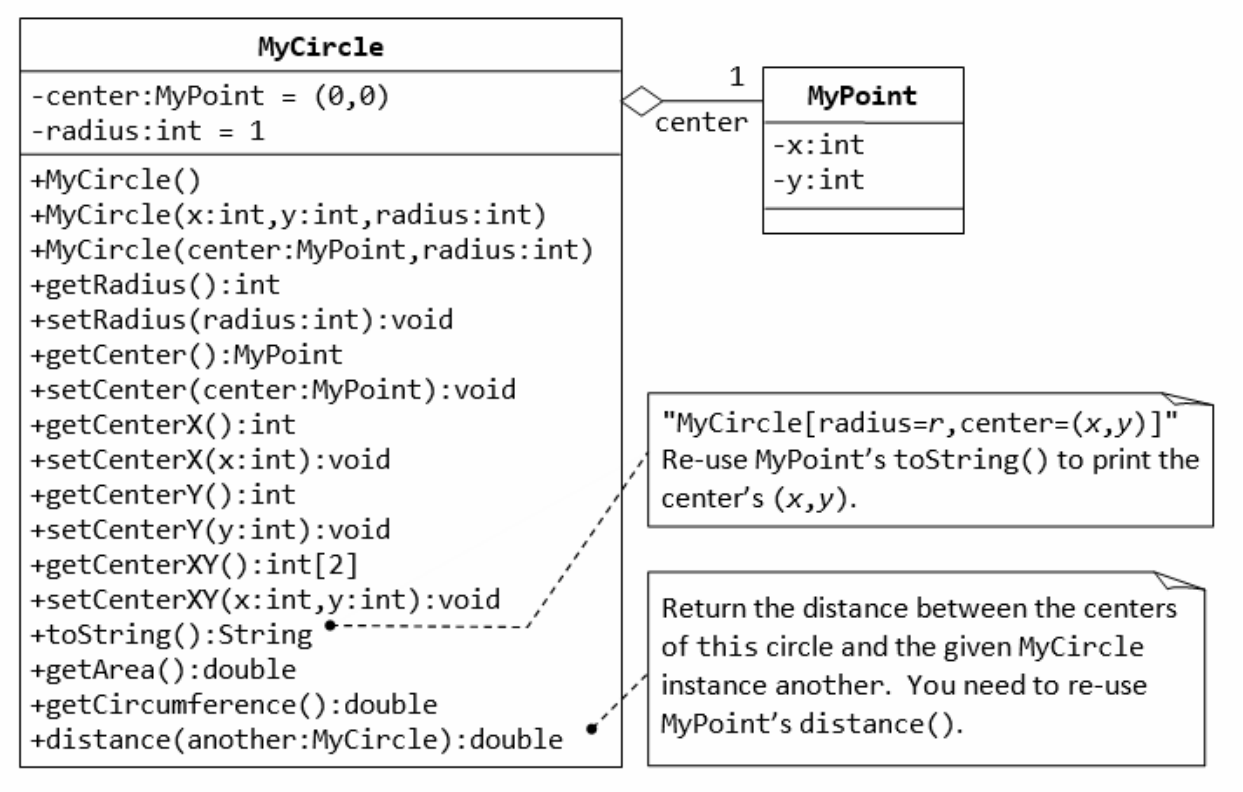
9. Book and Author Classes Again - An Array of Objects as an Instance Variable



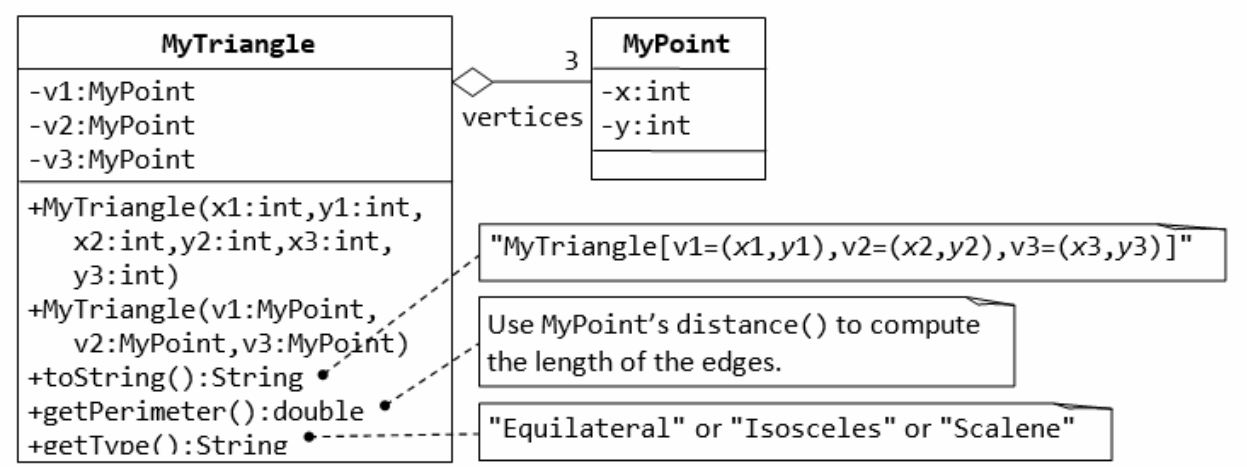
10 . The MyPoint Class



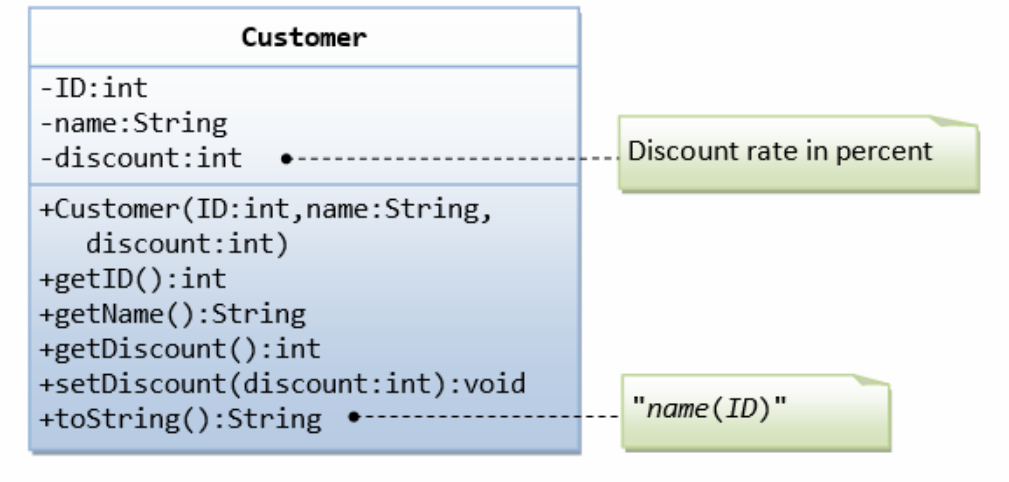
11. The MyCircle and MyPoint Classes

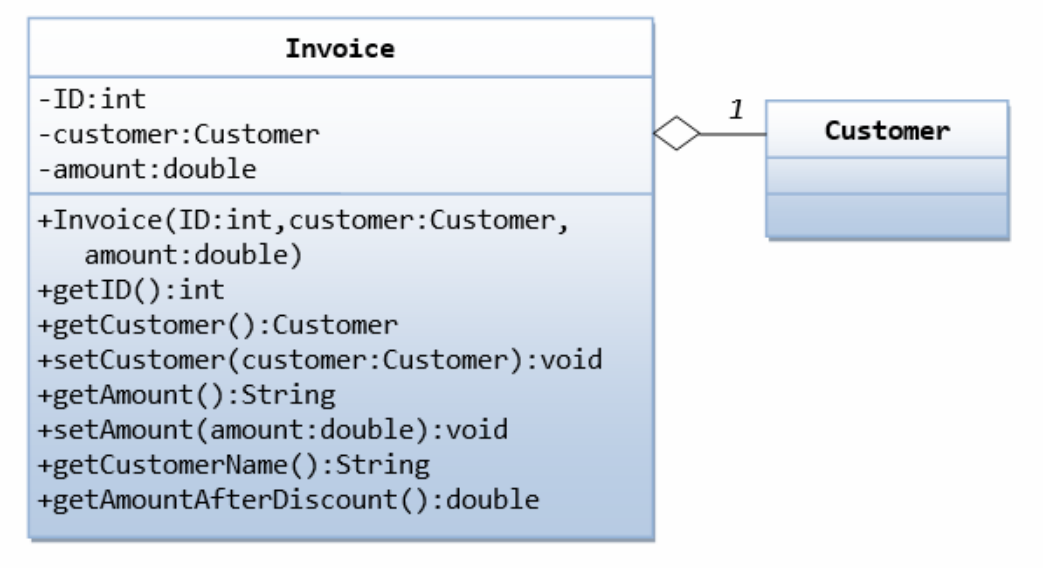


12. The MyTriangle and MyPoint Classes

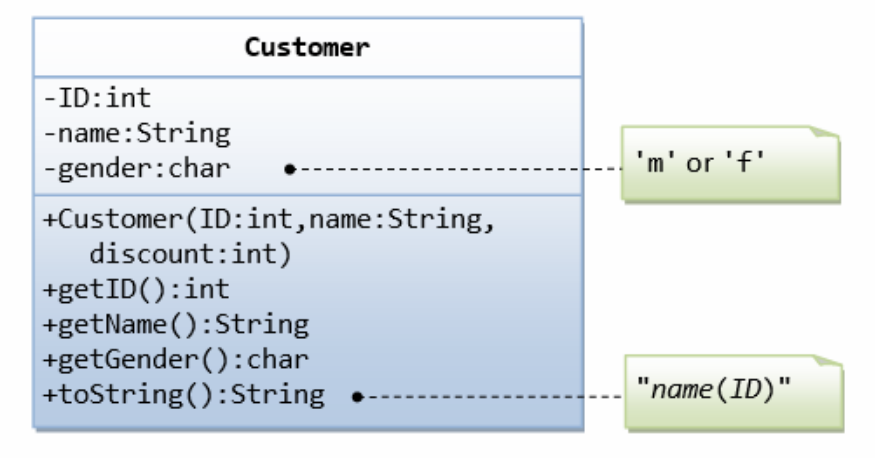


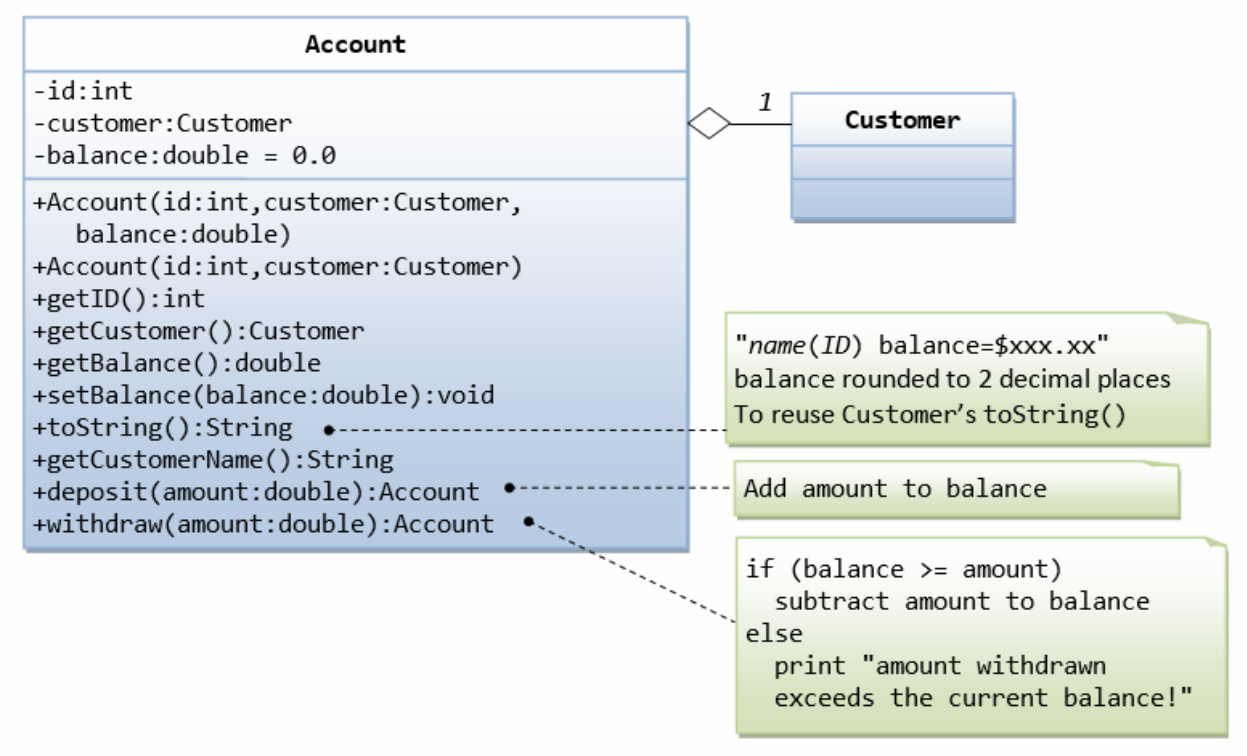
13. The Customer and Invoice classes



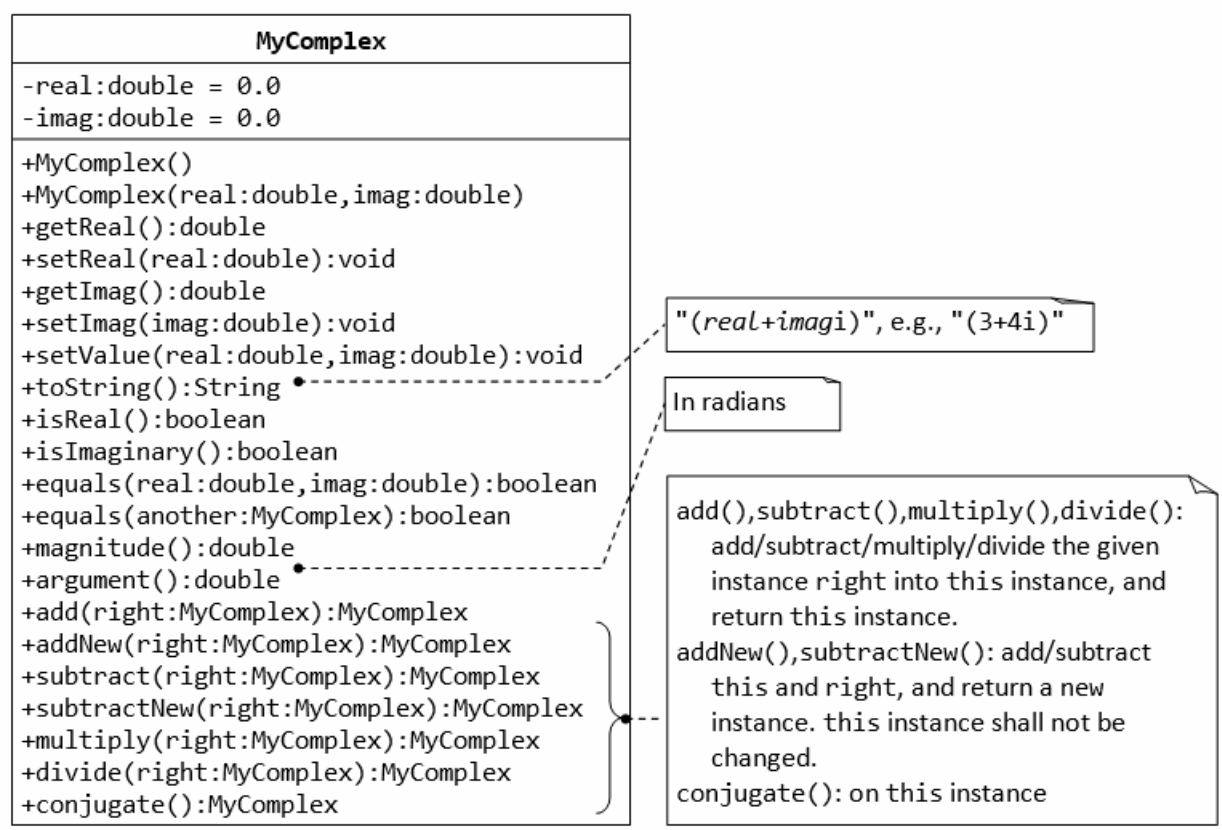


14 .The Customer and Account classes

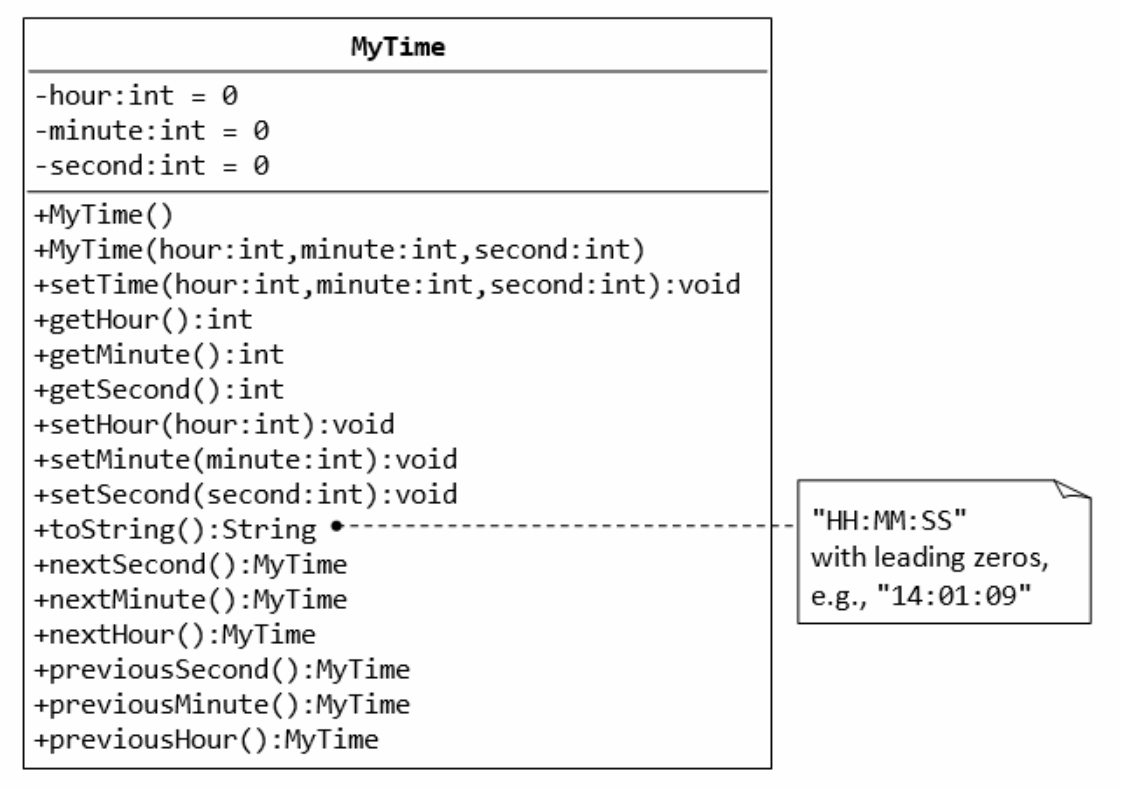




15. The MyComplex class

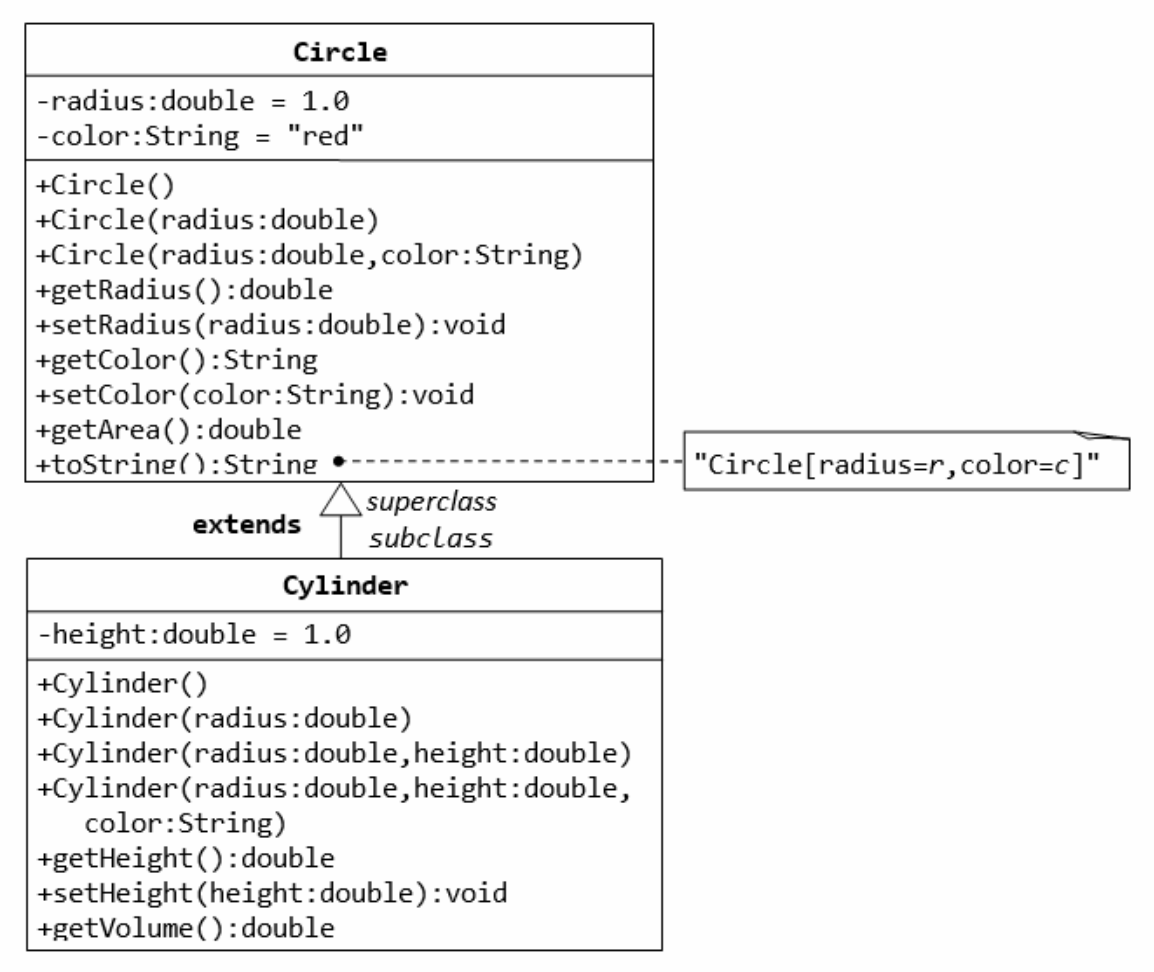


16. The MyTime Class

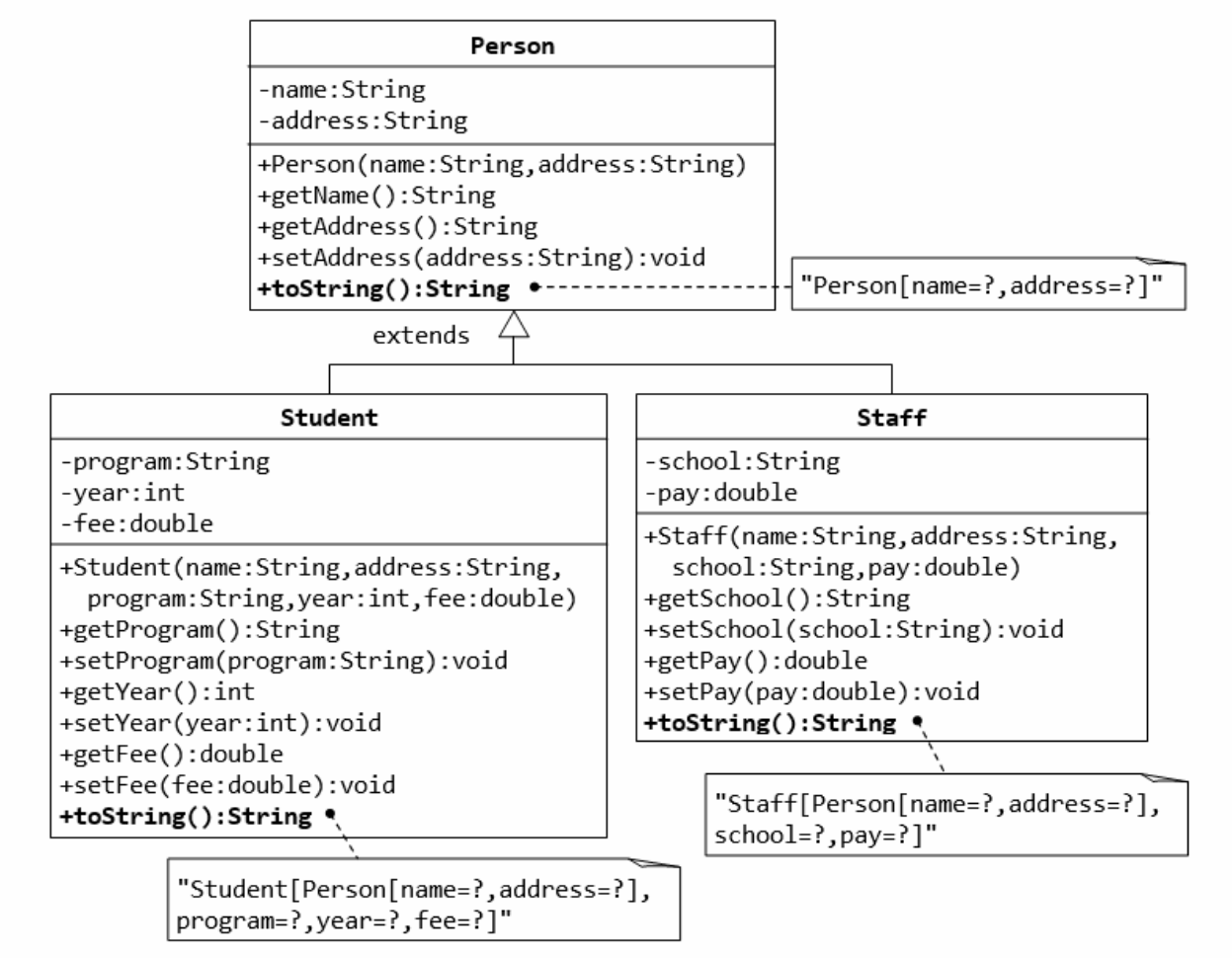


17 The MyDate Class

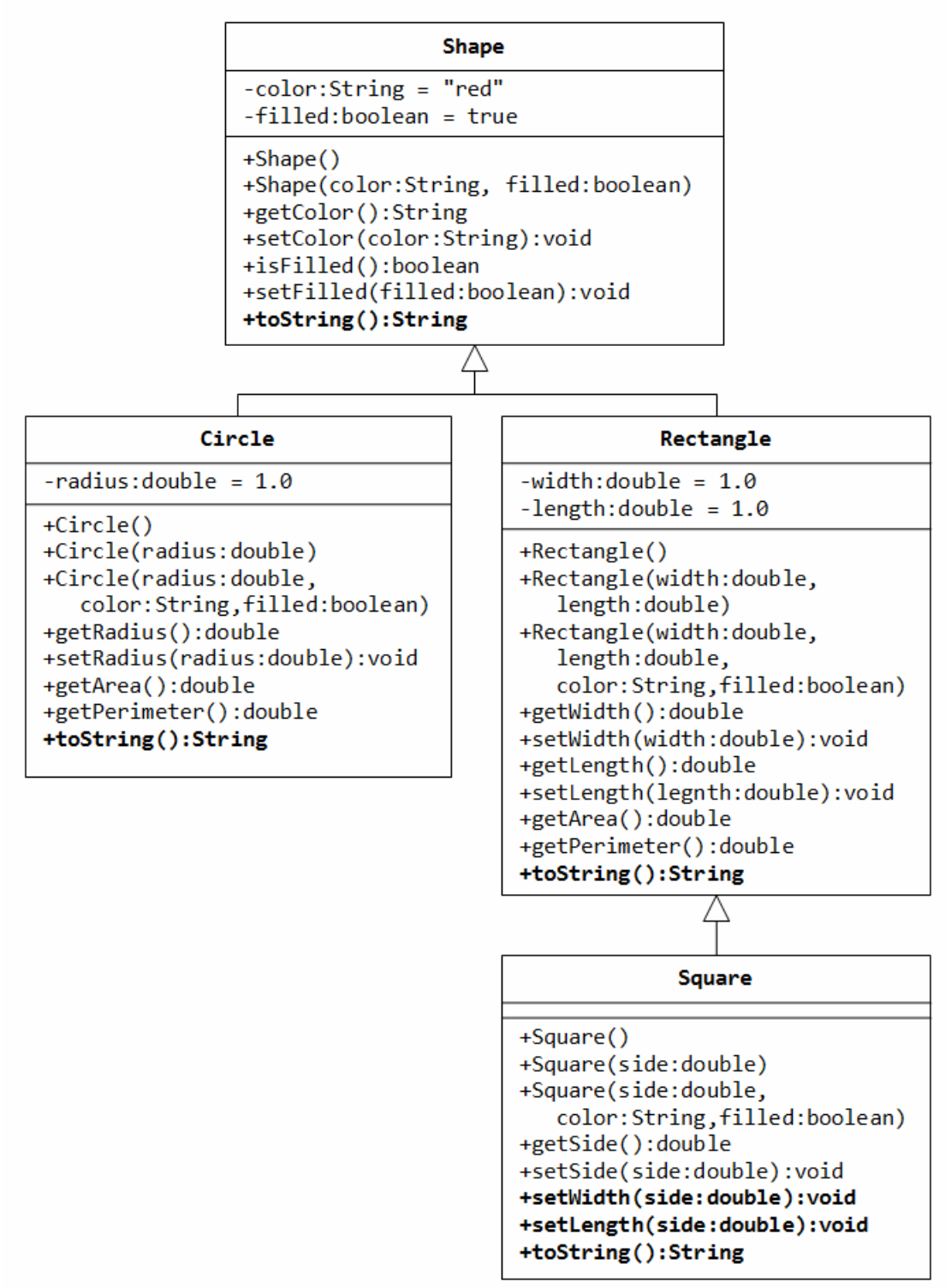
18. The Circle and Cylinder Classes



19. Superclass Person and its subclasses



20. Superclass Shape and its subclasses Circle, Rectangle and Square


Shape
-color:String = "red"
-filled:boolean = true
+Shape()
+Shape(color:String, filled:boolean)
+getColor():String
+setColor(color:String):void
+isFilled():boolean
+setFilled(filled:boolean):void
+toString():String
Circle
-radius:double = 1.0
+Circle()
+Circle(radius:double)
+Circle(radius:double,
color:String,filled:boolean)
+getRadius():double
+setRadius(radius:double):void
+getArea():double
+getPerimeter():double
+toString():String
Rectangle
-width:double = 1.0
-length:double = 1.0
+Rectangle()
+Rectangle(width:double,
length:double)
+Rectangle(width:double,
length:double,
color:String,filled:boolean)
+getWidth():double
+setWidth(width:double):void
+getLength():double
+setLength(legnth:double):void
+getArea():double
+getPerimeter():double
+toString():String
Square
+Square()
+Square(side:double)
+Square(side:double,
color:String,filled:boolean)
+getSide():double
+setSide(side:double):void
+setWidth(side:double):void
+setLength(side:double):void
+toString():String

21. Abstract Superclass Shape and Its Concrete Subclasses.

22. The Discount System

You are asked to write a discount system for a beauty saloon, which provides services and sells beauty products. It offers 3 types of memberships: Premium, Gold and Silver. Premium, gold and silver members receive a discount of 20%, 15%, and 10%, respectively, for all services provided. Customers without membership receive no discount. All members receives a flat 10% discount on products purchased (this might change in future). Your system shall consist of three classes: Customer, Discount and Visit, as shown in the class diagram. It shall compute the total bill if a customer purchases $x of products and $y of services, for a visit. Also write a test program to exercise all the classes.