**PSG COLLEGE OF TECHNOLOGY**

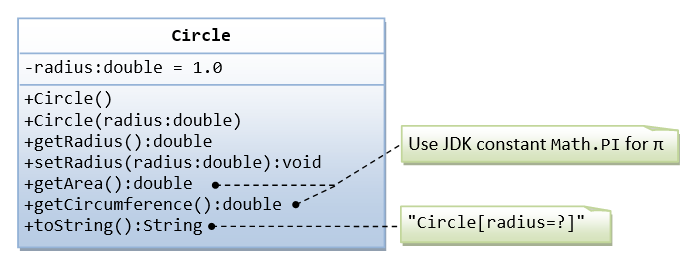
**DEPARTMENT OF COMPUTER APPLICATIONS**

**23MX26 - Java Programming Laboratory**

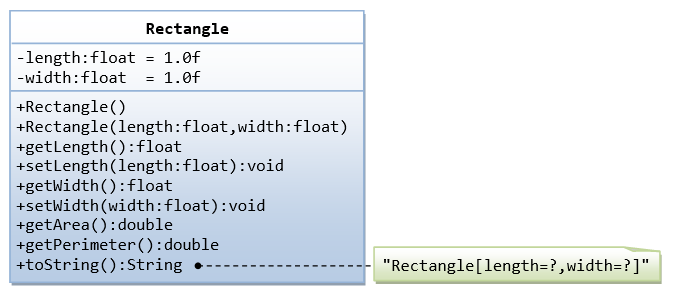
**Hands-on Worksheet 7**

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

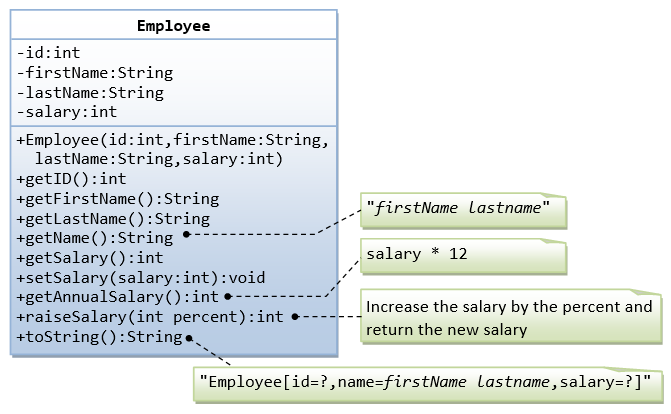
* Two private instance variables: radius (of the type double) and color (of the type String), with default value of 1.0 and "red", respectively.
* Two *overloaded* constructors - a *default* constructor with no argument, and a constructor which takes a double argument for radius.
* Two public methods: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



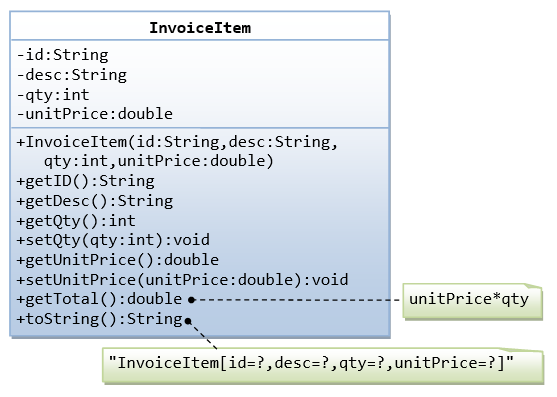
2. A class called Rectangle, which models a rectangle with a length and a width (in float), is designed as shown in the following class diagram. Write the Rectangle class.



3. A class called Employee, which models an employee with an ID, name and salary, is designed as shown in the following class diagram. The method raiseSalary(percent) increases the salary by the given percentage. Write the Employee class.



4. A class called InvoiceItem, which models an item of an invoice, with ID, description, quantity and unit price, is designed as shown in the following class diagram. Write the InvoiceItem class.



Below is a test driver to test the InvoiceItem class:

public class TestMain {

public static void main(String[] args) {

// Test constructor and toString()

InvoiceItem inv1 = new InvoiceItem("A101", "Pen Red", 888, 0.08);

System.out.println(inv1); // toString();

// Test Setters and Getters

inv1.setQty(999);

inv1.setUnitPrice(0.99);

System.out.println(inv1); // toString();

System.out.println("id is: " + inv1.getID());

System.out.println("desc is: " + inv1.getDesc());

System.out.println("qty is: " + inv1.getQty());

System.out.println("unitPrice is: " + inv1.getUnitPrice());

// Test getTotal()

System.out.println("The total is: " + inv1.getTotal());

}

}

The expected output is:

InvoiceItem[id=A101,desc=Pen Red,qty=888,unitPrice=0.08]

InvoiceItem[id=A101,desc=Pen Red,qty=999,unitPrice=0.99]

id is: A101

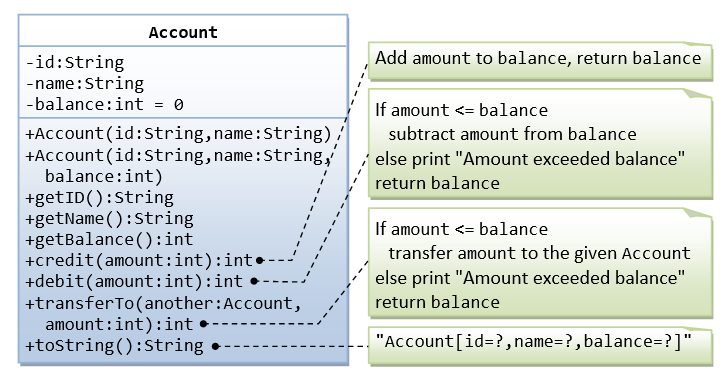
desc is: Pen Red

qty is: 999

unitPrice is: 0.99

The total is: 989.01

5. A class called Account, which models a bank account of a customer, is designed as shown in the following class diagram. The methods credit(amount) and debit(amount) add or subtract the given amount to the balance. The method transferTo(anotherAccount, amount) transfers the given amount from this Account to the given anotherAccount. Write the Account class.



Below is a test driver to test the Account class:

public class TestMain {

public static void main(String[] args) {

// Test constructor and toString()

Account a1 = new Account("A101", "Tan Ah Teck", 88);

System.out.println(a1); // toString();

Account a2 = new Account("A102", "Kumar"); // default balance

System.out.println(a2);

// Test Getters

System.out.println("ID: " + a1.getID());

System.out.println("Name: " + a1.getName());

System.out.println("Balance: " + a1.getBalance());

// Test credit() and debit()

a1.credit(100);

System.out.println(a1);

a1.debit(50);

System.out.println(a1);

a1.debit(500); // debit() error

System.out.println(a1);

// Test transfer()

a1.transferTo(a2, 100); // toString()

System.out.println(a1);

System.out.println(a2);

}

}

The expected output is:

Account[id=A101,name=Tan Ah Teck,balance=88]

Account[id=A102,name=Kumar,balance=0]

ID: A101

Name: Tan Ah Teck

Balance: 88

Account[id=A101,name=Tan Ah Teck,balance=188]

Account[id=A101,name=Tan Ah Teck,balance=138]

Amount exceeded balance

Account[id=A101,name=Tan Ah Teck,balance=138]

Account[id=A101,name=Tan Ah Teck,balance=38]

Account[id=A102,name=Kumar,balance=100]

6. You will compute a car’s gas mileage and the amount you spend per mile traveled. In particular, ask the user for the following.

• their first name (or their full name) (String),

• the price per gallon of gasoline paid at the time of the last fill-up (double),

• an initial odometer reading (int),

• a final odometer reading (int).

• the total amount paid for the fill-up (double), to be calculated as per

After inputting these values, compute the total miles drive, the number of gallons of gas used (total amount for fill-up / price per gallon), and the miles per gallon that the car achieved (total miles / number of gallons). Note that number of gallons and miles per gallon will both be doubles.