**Introduction to Computer Programming (Java A)**

**Lab 7**

**[Objective]**

* Learn how to define a Java class and create its object
* Learn how to define and use instance variables
* Learn how to define and use instance methods
* Learn how to use get and set methods
* Learn how to use ArrayList and make the object as its element.

**[Before Exercises]**

**Step1: How to define a circle on a 2D plane?**

A circle has three attributes including the **radius**, the **x** coordinate and the **y** coordinate.

We can define a class named Circle, in which there are three private attributes.

**public** **class** Circle {

**private** **double** radius;

**private** **double** x;

**private** **double** y;

}

**Step2: Define the methods of a circle.**

Define three public methods for computing the area, perimeter and print position of the circle.

**public** **class** Circle {

**private** **double** radius;

**private** **double** x;

**private** **double** y;

**public** **double** area() {

**return** radius\*radius\*Math.***PI***;

}

**public** **double** perimeter () {

**return** 2\*Math.***PI***\*radius;

}

**public** **void** position() {

System.***out***.printf("Position of the cricle is (%.1f,%.1f)\n",x,y);

}

}

**Step3: How to use the class Circle?**

Create another class named CircleTest in the same package, in which there is a main method to be used.

In the main method, we can create an object of Circle by using the statement as follows:

Circle c1=**new** Circle( );

After that, we want to know the perimeter, area and position about the c1, so we need to invoke the method of c1.

**public** **class** **CircleTest** {

**public** **static** **void** main(String[] args) {

Circle c1=**new** Circle();

System.***out***.printf("The area of c1 is %.2f\n", c1.area());

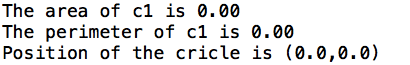
System.***out***.printf("The perimeter of c1 is %.2f\n", c1.perimeter());

c1.position();

}

}

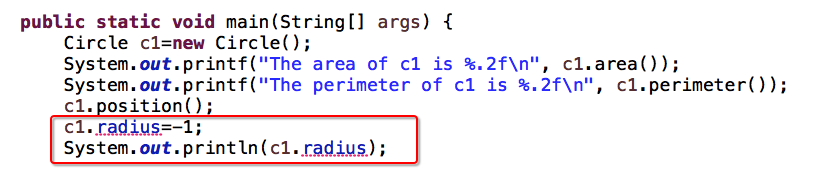
When we run the program, the result would as follows:



**Step4: Set and get the values of the attributes**

If we set or get the radius of a circle object in main method directly, it would lead to an error because of its private privilege.

In addition, the radius of a circle should not contain a negative number, how can we set the restriction?



We can define several public methods in class Circle for getting or setting the class variables, and we can check the validity of input value in the set method.

**public** **class** Circle {

**private** **double** radius;

**private** **double** x;

**private** **double** y;

**public** **double** area() {

**return** radius\*radius\*Math.***PI***;

}

**public** **double** perimeter () {

**return** 2\*Math.***PI***\*radius;

}

**public** **void** position() {

System.***out***.printf("Position of the cricle is (%.1f,%.1f)\n",x,y);

}

**public** **double** getRadius() {

**return** radius;

}

**public** **void** setRadius(**double** radius) {

**if** (radius > 0) {

**this**.radius = radius;

}

}

**public** **double** getX() {

**return** x;

}

**public** **void** setX(**double** x) {

**this**.x = x;

}

**public** **double** getY() {

**return** y;

}

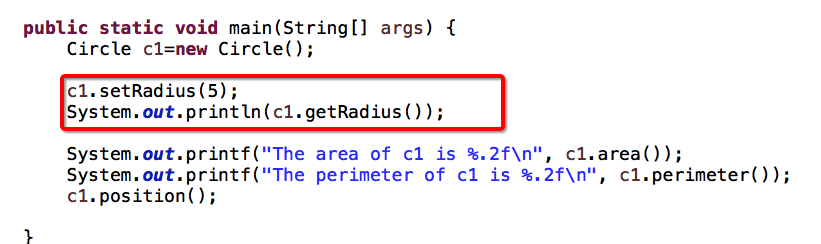
**public** **void** setY(**double** y) {

**this**.y = y;

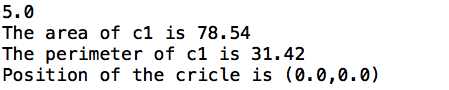
}

}

After that, we can access the attributes by the get and set methods.



Sample output:



**Step5: How to manage multiple circle objects ?**

We can use an array or an ArrayList to manage them.

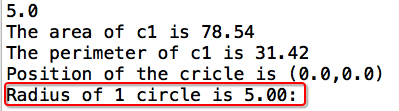
In the main method, create an arrayList with a Circle type, to store many objects of Circle. Add the following code at the end of main method.

ArrayList<Circle> circleList=**new** ArrayList<Circle>();

circleList.add(c1);

System.***out***.printf("Radius of %d circle is %.2f: \n",1,circleList.get(0).getRadius());

Sample output:



**Step5: Add more circles in the ArrayList.**

Add the following code at the end of main method.

**for**(**int** i=1;i<5;i++) {

circleList.add(**new** Circle());

circleList.get(i).setRadius(i);

circleList.get(i).setX(Math.*random*()\*5);

circleList.get(i).setY(Math.*random*()\*5);

}

System.***out***.println("---Begin to print the circle list---");

**for**(**int** i=0;i<5;i++) {

System.***out***.printf("The area of %d circle is %.2f\n",

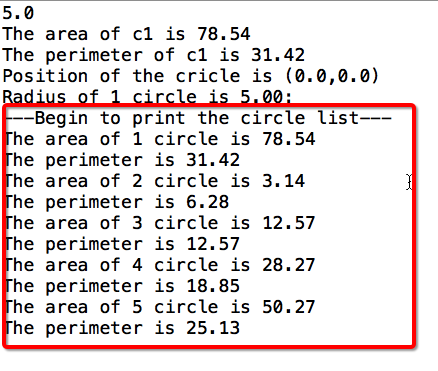
i+1, circleList.get(i).area());

System.***out***.printf("The perimeter is %.2f\n",

circleList.get(i).perimeter());

}

Sample output:



**[Exercises]**

1. Declare a class named **User**. The class contains:

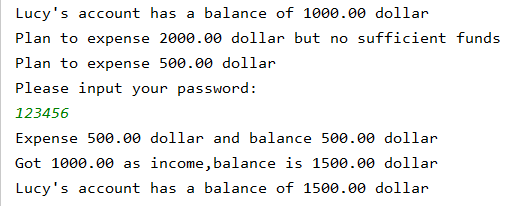
1. Private data fields:
   * 1. String **account;**
     2. **String password**;
     3. **double money**;
2. Implement a public method named **introduce()** to print the user name and his account balance.
3. Implement a public method **expense(double value,Scanner in)**. It withdraws the money from the user account if the password is correct.
4. Implement a public method **income(double value).** It deposits the money to the user account.
5. Implement the **getter** and **setter** methods for each private field of the class User.

In the same package, we create a class named **UserTest**, which has a main method.

**Statements in main method:**

User user =**new** User();  
 Scanner in = **new** Scanner(System.***in***);  
 user.setUser(**"Lucy"**);  
 user.setPassword(**"123456"**);  
 user.setMoney(1000);  
 user.introduce();  
 user.expense(2000,in);  
 user.expense(500,in);  
 user.income(1000);  
 user.introduce();  
 in.close();

**Sample output:**



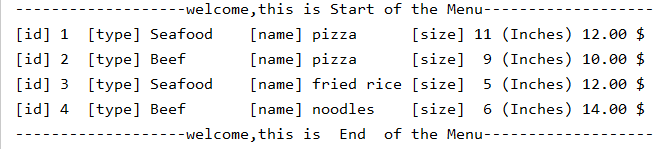
1. Design a class named **Food**. The class contains:
2. Private data fields:
   * 1. int **id；**
     2. String **name;**
     3. String **type**;
     4. int **size**;
     5. double **price;**
3. Implement a public method named **getMenu()** to print all the information of this food object*.*
4. Implement the **getter** and **setter** method for each private field of Food.

In **FoodTest** class, create four objects of Food as follows:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Object Name | id | name | type | size | price |
| **pizza1** | 1 | pizza | Seafood | 11 | 12 |
| **pizza2** | 2 | pizza | Beef | 9 | 10 |
| **Fried rice** | 3 | fried rice | Seafood | 5 | 12 |
| **Noodles** | 4 | noodles | Beef | 6 | 14 |

Create an ArrayList<Food> to add those four Food objects, and then show the information of them as follows by iterating the ArrayList<Food> we created.

**Sample output:**



1. Design a class named **softOpening**. The class contains no data fields but:
2. Implement a public static method named **generateMenu()** to generate 4 object of Food and add them to the ArrayList<Food>.
3. Implement a public static method named **getMenu(ArrayList<Food>)** to print the items in the ArrayList<Food> as designed.
4. Implement a public static method named to generateUser(Scanner in) to generated a user whose account and money is get by using the Scanner object ‘in’.
5. Implement a public static method named **UserConsume(ArrayList<Food>,User user,Scanner in)** to invoke the getMenu, ask user to select the foods in the Menu, count the cost and invoke the expense of the user.
6. Invoke the method **introduce()** of the User object to show his/hers balance.

**Statements in main method**:

Scanner in = **new** Scanner(System.**in**);  
ArrayList<Food> foodList = genarateMenu(); //generate a Menu  
User user = genarateUser(in); //generate a user  
user.introduce(); //show the account of the user  
userConsume(foodList,user,in); //user consume  
user.introduce(); //show the account of the user  
in.close();

**Sample output:**

