420-200-RE	Assignment 5	Due date: 30 November 2016
Creating Objects of Our Own Classes		

Objectives

- ⋆ To practice basic concepts in object-oriented programming (OOP).
- ⋆ To gain experience with implementing a simple class in C++.

Class Rectangle

In this assignment we are interested in modeling rectangular shapes. The rectangular shape objects we are interested in each have a *width*, a *height*, a *name*, and a *pen*. The attribute *pen* of a rectangular shape refers to a drawing character used to visually draw that shape.

So your task is to write a class named **Rectangle** to model rectangular shapes in terms of four of their common attributes: *width*, a *height*, *name*, and a *pen*.

Your **Rectangle** class should have the following public methods. Feel free to add other **private** methods to facilitate your task.

1. A constructor that takes three parameters that supply initial values for the instance variables width, height, and pen. This constructor will allow the client code to create a **Rectangle** object as follows:

   Source code

   ```
   1  Rectangle t = new Rectangle(7, 4, 'O', "Room");
   ```

2. A **toString()** method that returns a string representation of *this* rectangle. For example:

   Source code

   ```
   2  System.out.println(t);
   ```

   output

   ```
   1  O O O O O O O
   2  O O O O O O O
   3  O O O O O O O
   4  O O O O O O O
   ```

3. Four pairs of getter (accessor) and setter (mutator) methods, one pair for each instance variables.
4. A method named **area()** that returns the area of *this* rectangle.
5. A method named **perimeter()** that returns the perimeter of *this* rectangle.
6. A method **displayInfo()** that returns a **String** that includes the values of all instance variables, the area, and perimeter of *this* rectangle. For example:

Source code

```
3    t.diaplayInfo();
```

output

```
5  --------------------
6   A Rectangle object
7  --------------------
8  Name      : Room
9  Width     : 7
10 Height    : 4
11 Pen       : 0
12 Area      : 28
13 Perimeter: 22
14 --------------------
```

7. A method **rotate90()** method that rotates *this* rectangular shape by 90 degrees. For example:

Source code

```
4    t.rotate90();
5    System.out.println(t);
6    t.diaplayInfo();
```

output

```
O O O O
O O O O
O O O O
O O O O
O O O O
O O O O
O O O O

--------------------
 A Rectangle object
--------------------
Name      : Room
Width     : 4
Height    : 7
Pen       : O
Area      : 28
Perimeter: 22
--------------------
```

8. A method **sameSize(Rectangle t)** that determines whether *this* rectangle has the same width and height as the rectangle referenced by **t**. For example:

Source code

```
Rectangle t2 = new Rectangle(20, 4, '*', "Hallway");
System.out.println(t2);
t2.diaplayInfo();
boolean result = t.sameSize(t2);
System.out.printf("Recatngles named %s and %s are %s the same size\n",
    t.getName(), t2.getName(), (result?"":"not"));
```

output

```
* * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * * * * * * * * * * * * *

--------------------
 A Rectangle object
--------------------
Name      : Hallway
Width     : 20
Height    : 4
Pen       : *
Area      : 80
Perimeter: 48
--------------------
Recatngles named Room and Hallway are not the same size
```