CS 1400-03 Introduction to Programming and Problem Solving
Coding Practice #5
(Due: 11:59 PM, Friday, 3/5/2021)

Except Coding Practice #1, I will not grade your coding practice submissions. Instead, they will be treated as participation points. On blackboard, you will receive full points as long as you work on the exercises, which don't necessary mean they are all correct. Please check your own programs carefully and make sure they do generate the desired output.

Objectives:

- Be able to create complete Java programs with
  - Methods (write methods and call methods)
  - Loops
  - Decision Structures
- Be able to test and debug a program

Change your working directory to `cs1400/codingPractice` for this assignment.

For tasks #1 to #3, you will edit Geometry.java program.

Task #1 void Methods

1. Copy the file as shown at the end of this document from /www/user/fcsang/download/Geometry.java. This program will compile, but when you run it, it doesn't appear to do anything except wait. That is because it is waiting for user input, but the user doesn't have the menu to choose from yet. We will need to create this.

2. Below the main method, but in the Geometry class, create a static method called `printMenu` that has no parameter list and does not return a value. It will simply print out instructions for the user with a menu of options for the user to choose from. The menu should appear to the user as:

```
This is a geometry calculator
Choose what you would like to calculate
1. Find the area of a circle
2. Find the area of a rectangle
3. Find the area of a triangle
4. Find the circumference of a circle
5. Find the perimeter of a rectangle
6. Find the perimeter of a triangle
Enter the number of your choice:
```

3. Add a line in the main method that calls the `printMenu` method as indicated by the comments.

4. Compile, debug, and run. You should be able to choose any option, but you will always get 0 for the answer. We will fix this in the next task.

Task #2 Value-Returning Methods

1. Write a static method called `circleArea` that takes in the radius of the circle and returns the area using the formula $A = \pi r^2$.

2. Write a static method called `rectangleArea` that takes in the length and width of the rectangle and

returns the area using the formula $A = lw$.

3. Write a static method called `triangleArea` that takes in the base and height of the triangle and returns the area using the formula $A = ½\ bh$.

4. Write a static method called `circleCircumference` that takes in the radius of the circle and returns the circumference using the formula $C = 2\pi r$.

5. Write a static method called `rectanglePerimeter` that takes in the length and the width of the rectangle and returns the perimeter of the rectangle using the formula $P = 2l + 2w$.

6. Write a static method called `trianglePerimeter` that takes in the lengths of the three sides of the triangle and returns the perimeter of the triangle which is calculated by adding up the three sides.

Task #3 Calling Methods

Add lines in the main method in the `Geometry` class which will call these methods. The comments indicate where to place the method calls.

Task #4 Prime Checker

A prime number is a whole number greater than 1, that is only evenly divisible by itself and 1. For example, the number 5 is a prime because it can only be evenly divided by 1 and 5. The number 6, however, is not a prime because it can be divided evenly by 1, 2, 3, and 6.

Write a `public static boolean` method name `isPrime`, which takes an integer as an argument and returns true if the argument is a prime number, or false otherwise. Write a main method that asks the user to enter a number, calls the method `isPrime,` and displays the result. The following are sample interactions, where the user's input is shown in bold.

```
fcsang@fluffy ~/cs1400/codingPractice $ java PrimeChecker
Enter a number: 1
That is not a prime number.

fcsang@fluffy ~/cs1400/codingPractice $ java PrimeChecker
Enter a number: 2
That is a prime number.

fcsang@fluffy ~/cs1400/codingPractice $ java PrimeChecker
Enter a number: 3
That is a prime number.

fcsang@fluffy ~/cs1400/codingPractice $ java PrimeChecker
Enter a number: 4
That is not a prime number.

fcsang@fluffy ~/cs1400/codingPractice $ java PrimeChecker
Enter a number: 5
That is a prime number.

fcsang@fluffy ~/cs1400/codingPractice $ java PrimeChecker
Enter a number: 6
That is not a prime number.
```

Task #5 String Comparison with Wild Card

Write a `public static boolean` method `match` which takes two `String`s and returns `true` if and only if the two `String`s are of equal length and are equal, character for character, except that a `'?'` in either `String` counts as a wild card which matches any character in the corresponding position of the other `String`. The `String`s may be any length, including 0. For example:

string1	*string2*	*result*
"abc"	"ab"	false
""	""	true
"abc"	"abc"	true
"abc"	"aeb"	false
"ab?"	"abd"	true
"a?c"	"adc"	true
"ab?"	"a?c"	true

Write a main method that asks the user to enter two strings, calls the method `match`, and displays the result. The following are sample interactions. The user's input is shown in bold.

```
fcsang@garrison ~/cs1400/codingPractice $ java StringComparisonWithWildCard
enter string 1: abc
enter string 2: abcd
they are different

fcsang@garrison ~/cs1400/codingPractice $ java StringComparisonWithWildCard
enter string 1: ab?de
enter string 2: abzde
they are equal
```

Submission:

Generate a `script` file `practice5.txt` with appropriate time stamps and the following steps visible:

1) a `pwd` to show the current working directory
2) a `ls -l` to show in long format the files in your `cs1400/codingPractice` directory
3) a `cat` to display `Geometry.java`
4) compile `Geometry.java`
5) run `Geometry` multiple times with different menu options
6) a `cat` to display `PrimeChecker.java`
7) compile `PrimeChecker.java`
8) run `PrimeChecker` multiple times with different user input
9) a `cat` to display `StringComparisonWithWildCard.java`
10) compile `StringComparisonWithWildCard.java`
11) run `StringComparisonWithWildCard` multiple times with different strings

Submit the script file `practice5.txt` on Bb, under the Coding Practice Folder, Practice #5 link.

Geometry.java

```
//CS1400 - Coding Practice #5
import java.util.Scanner;
public class Geometry
{
   public static void main (String [] args)
   {
      int choice;          //the user's choice
      double value = 0;    //the value returned from the method
      char letter;         //the Y or N from the user's decision to exit
      double radius;       //the radius of the circle
      double length;       //the length of the rectangle
      double width;        //the width of the rectangle
      double height;       //the height of the triangle
      double base;         //the base of the triangle
      double side1;        //the first side of the triangle
      double side2;        //the second side of the triangle
      double side3;        //the third side of the triangle

      //create a scanner object to read from the keyboard
      Scanner keyboard = new Scanner (System.in);
      //do loop was chose to allow the menu to be displayed first
      do
      {
         //call the printMenu method
         choice = keyboard.nextInt();
         switch (choice)
         {
            case 1:
               System.out.print("Enter the radius of the circle:  ");
               radius = keyboard.nextDouble();
               //call the circleArea method and
               //store the result in the value variable
               System.out.println("The area of the circle is " + value);
               break;
            case 2:
               System.out.print("Enter the length of the rectangle:  ");
               length = keyboard.nextDouble();
               System.out.print("Enter the width of the rectangle:  ");
               width = keyboard.nextDouble();
               //call the rectangleArea method
               //and store the result in the value variable
               System.out.println("The area of the rectangle is " + value);
               break;
            case 3:
               System.out.print("Enter the height of the triangle:  ");
               height = keyboard.nextDouble();
               System.out.print("Enter the base of the triangle:  ");
               base = keyboard.nextDouble();
               //call the triangleArea method
               //and store the result in the value variable
               System.out.println("The area of the triangle is " + value);
               break;
            case 4:
               System.out.print("Enter the radius of the circle:  ");
               radius = keyboard.nextDouble();
```

```
                //call the circumference method
                //and store the result in the value variable
                System.out.println("The circumference of the circle is "
                                   + value);
                break;
             case 5:
                System.out.print("Enter the length of the rectangle:  ");
                length = keyboard.nextDouble();
                System.out.print("Enter the width of the rectangle:  ");
                width = keyboard.nextDouble();
                //call the perimeter method
                //and store the result in the value variable
                System.out.println("The perimeter of the rectangle is "
                                   + value);
                break;
             case 6:
                System.out.print("Enter the length of side 1:  ");
                side1 = keyboard.nextDouble();
                System.out.print("Enter the length of side 2:  ");
                side2 = keyboard.nextDouble();
                System.out.print("Enter the length of side 3:  ");
                side3 = keyboard.nextDouble();
                //call the perimeter method
                //and store the result in the value variable
                System.out.println("The perimeter of the triangle is "
                                   + value);
                break;
             default:
                System.out.println("You did not enter a valid choice.");
          }

          //consumes the new line character after the number
          keyboard.nextLine();

          System.out.println("Do you want to exit the program (Y/N)?:  ");
          String answer = keyboard.nextLine();
          letter = answer.charAt(0);

       } while (letter != 'Y' && letter != 'y');
    }
}
```