Lab Session #11

Part 1: Theory

1. Consider the following code. What are the compiler errors and how can you fix them?   
     
   public class Methods2 {

public static void main(String[] args) {

double x = 0;

System.out.println(x);

confusing(x);

System.out.println(x);

}

public static int confusing(int y) {

System.out.println(y);

}

}

1. Consider the following code. What are the compiler errors and how can you fix them?   
     
   public class Methods2 {

public static void main(String[] args) {

int x = 0;

System.out.println(x);

confusing(x);

System.out.println(x);

}

public static void confusing(int y) {

System.out.println(y);

return 0;

}

}

1. Consider the following code. What are the compiler errors and how can you fix them?   
     
   public class Methods2 {

public static void main(String[] args) {

int x = 0;

System.out.println(x);

confusing(x);

System.out.println(x);

}

public static int confusing(int y) {

if (y > 0) {

return 0;

}

}

1. Consider the following code. What are the compiler errors and how can you fix them?   
     
   public class Methods2 {

public static void main(String[] args) {

java.util.Scanner reader = new java.util.Scanner(System.in);

int x = reader.next();

if (x > 0) {

int y = 10;

}

System.out.println(y);

}

}

1. Consider the following code. What are the compiler errors and how can you fix them?   
     
   public class Methods2 {

public static void main(String[] args) {

int x = 0;

System.out.println(x);

confusing(x);

System.out.println(x);

}

public static int confusing(int y) {

return System.out.println(y);

}

}

**Part 2: Application**

**If you have not completed the perimeterRightTriangle program from the last lab, make sure to do so first as this question builds on that one.**

1. You will now largely repeat the program you did in the last lab, but with fewer hints. It is a good idea to look at your previous code and try to see what parts you can adapt from it.

Write a program *RectangleStatsCalculator.* In this program, you should ask the user to enter 2 numbers for the base and height of the rectangle. You should then calculate the area of the rectangle with those dimensions, the perimeter, and the length of the diagonal bit of the rectangle.

* You should have a method *perimeterRectangle* which takes 2 values as input (for the length and height) and returns the perimeter of a rectangle with those dimensions.  
    
  Hint: Your method header should look:  
    
  public static double perimeterRectangle (double side1, double side2)
* You should also have a method *areaRectangle* which also takes 2 values as input.
* You should have a 3rd method *findDiagonal* which takes 2 values as input and calculates and returns the length of the diagonal. (Hint: This is also Pythagorean theorem!)
* Your main method should ask the user to enter the 2 values. As in the previous lab, it should check, one at a time, if the values are positive. If they are not, it should print “invalid length” or “invalid width” (depending on which one was just typed. Once these two numbers are given, your main method should call the other methods and then print the values returned by them. *All printing and scanning should be done in the main method, not in any of the other methods.*

2. You are writing an application for a bank. It wants to assess a service charge for cashing a check, where the service charge depends on the amount of the check.

* if the check value < 10, charge 1% of the check value
* if greater than or equal to 10 but less than 100, charge 10% of the check value
* if greater than or equal to 100 but less than 1000, charge $5 plus 5% of the check value
* if over 1000, charge 40 plus 1%

Ask the user for the check value and display the check value as well as the service charge. NOTE: Use else-if statements efficiently!!!

**Sample run:**

Check value: 98.0

Service charge 9.8