iGCSE Computer Science – Topic 5: Develop Code

English Name: ____________________

Worksheet: `Geometry` Class

1. Take your `UsingMathClass`, and separate out the geometry-related methods into a separate class named `Geometry`. So that we can keep our `UsingMathClass` code undisturbed, also make a UsingGeometryClass that will call the `Geometry` class methods. Here are the steps to follow:
   - Create a class named `Geometry` – do not include a `main` method in this class!
   - Copy all geometry-related methods from `UsingMathClass` into the `Geometry` class. (The methods `calculateCircumference` and `hypotenuseLength`).
   - Create a class name `UsingGeometryClass`. Include a `main` method in this class.
   - Copy the code from the `main` method of `UsingMathClass` into the `main` method of the `UsingGeometryClass`.
   - At this point, the `UsingGeometryClass` will not know where to find the methods `calculateCircumference` nor `hypotenuseLength`. To let the compiler know which class to find these methods, put “`Geometry.`” in front of each of these methods. Note that this is similar to how we call `Math` class methods – using the class name and a period in front of the method name, such as: `Math.sqrt(2)`.
   - Run your `UsingGeometryClass` and confirm the output is the same as it was for the `UsingMathClass`. Here is the expected output of `UsingMathClass`:

```
Circumference: 18.84955592153876
Hypotenuse Length: 5.0
```

After coding and testing your solution, copy your working code from each class into the appropriate box below:

```
// Geometry class

   public class Geometry {

      public static double hypotenuseLength(double x, double y) {
         return Math.sqrt(x*x + y*y);
      }

      public static double calculateCircumference(double radius) {
         return 2 * Math.PI * radius;
      }

   }
```

```
// UsingGeometryClass class

   public class UsingGeometryClass {

      public static void main(String[] args) {
         System.out.println("Using Geometry");

         double c = Geometry.calculateCircumference(3.0);
         System.out.println("Circumference: " + c);

         double len = Geometry.hypotenuseLength(3.0, 4.0);
         System.out.println("Hypotenuse Length: " + len);
      }
   }
```