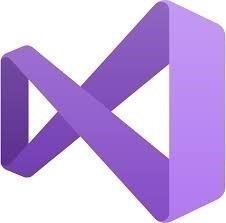
**CS 131C# - Beginner**

**HOP09 – Inheritance**

3/1/2020 Developed by Kim Nguyen

Center for Information Assurance (CIAE) @City University of Seattle (CityU)



**Before You Start**

* Version numbers may not match with the most current version at the time of writing. If given the option to choose between stable release (long-term support) or most recent, please choose the stable release rather than beta-testing version.
* This tutorial targets Windows users and MacOS users.
* There might be subtle discrepancies along the steps. Please use your best judgement while going through this cookbook style tutorial to complete each step.
* For your working directory, use your course number. This tutorial may use a different course number as an example.
* The directory path shown in screenshots may be different from yours.
* If you are not sure what to do or confused with any steps:
  1. Consult the resources listed below.
  2. If you cannot solve the problem after a few tries, ask a TA for help.

**Learning Outcomes**

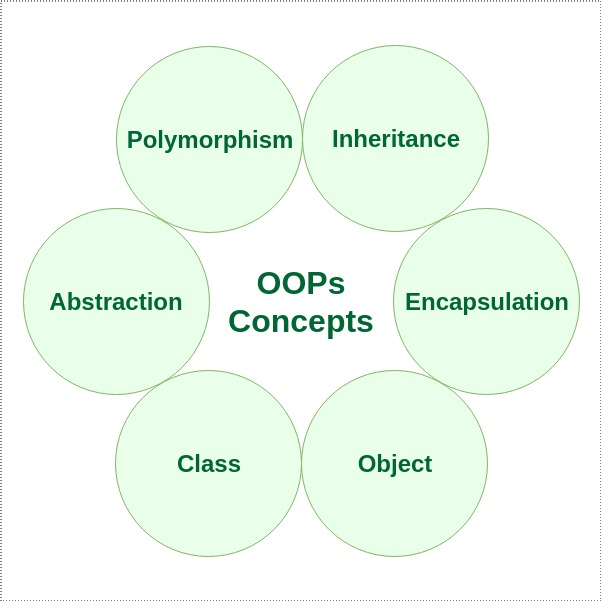
Students will be able to:

* Program using Inheritance.
* Extend and strengthen OOP knowledge of Class, Object and Encapsulation.

**Resources**

* C# Tutorial | Freecodecamp.org- <https://youtu.be/GhQdlIFylQ8>
* C# Tutorials | W3Schools.com- <https://www.w3schools.com/cs/default.asp>
* C# Tutorials | tutorials.com- [https://www.tutorialspoint.com/csharp/](https://www.tutorialspoint.com/csharp/csharp_strings.htm)

Last week, we have learned about Object-Oriented Programming (OOP), and practiced on 3 OOP concepts: Class, Object and Encapsulation. This week, let's dive into Inheritance.



One of the most important concepts in object-oriented programming is inheritance. Inheritance allows us to define a class in terms of another class, which makes it easier to create and maintain an application. This also provides an opportunity to reuse the code functionality and speeds up implementation time.

When creating a class, instead of writing completely new data members and member functions, the programmer can designate that the new class should inherit the members of an existing class. This existing class is called the base class, and the new class is referred to as the derived class.

The idea of inheritance implements the IS-A relationship. For example, mammal IS A animal, dog IS-A mammal hence dog IS-A animal as well, and so on.

In C#, inheritance are applied using the same concept:

Derived Class (child) - the class that inherits from another class

Base Class (parent) - the class being inherited from

To inherit from a class, use the : symbol.

Let's practice!

**Create a project**

1. Open Visual Studio.
2. File > New > Project
3. Select Console App (.NET Core), click Next
4. Type “Inheritance” in the Project name and save it in the following locations:

**If you are an online student:**

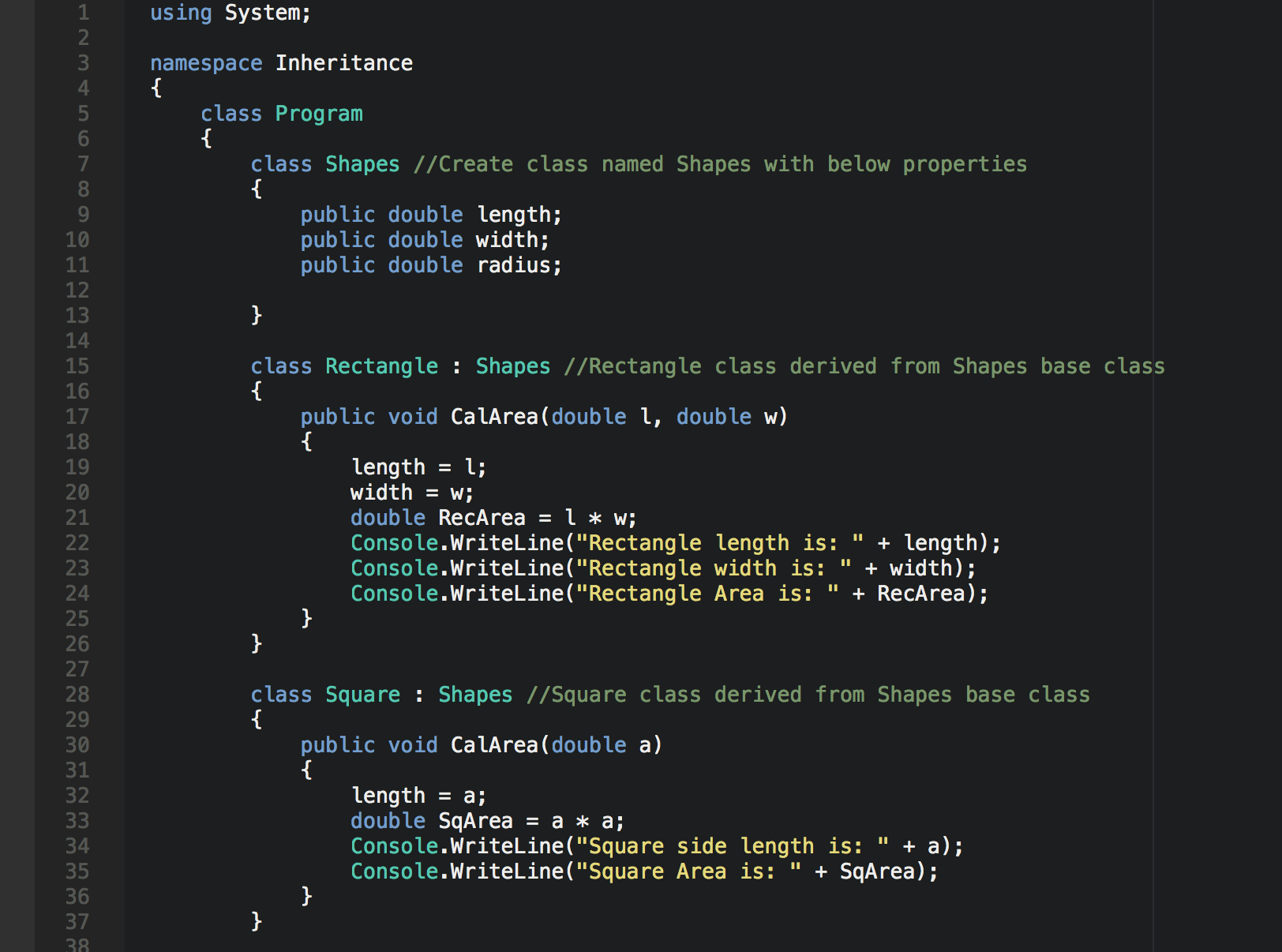
Save it here > CS131-Winter-2020\**ON**\FirstnameLastname/Module9/OOP-Inheritance

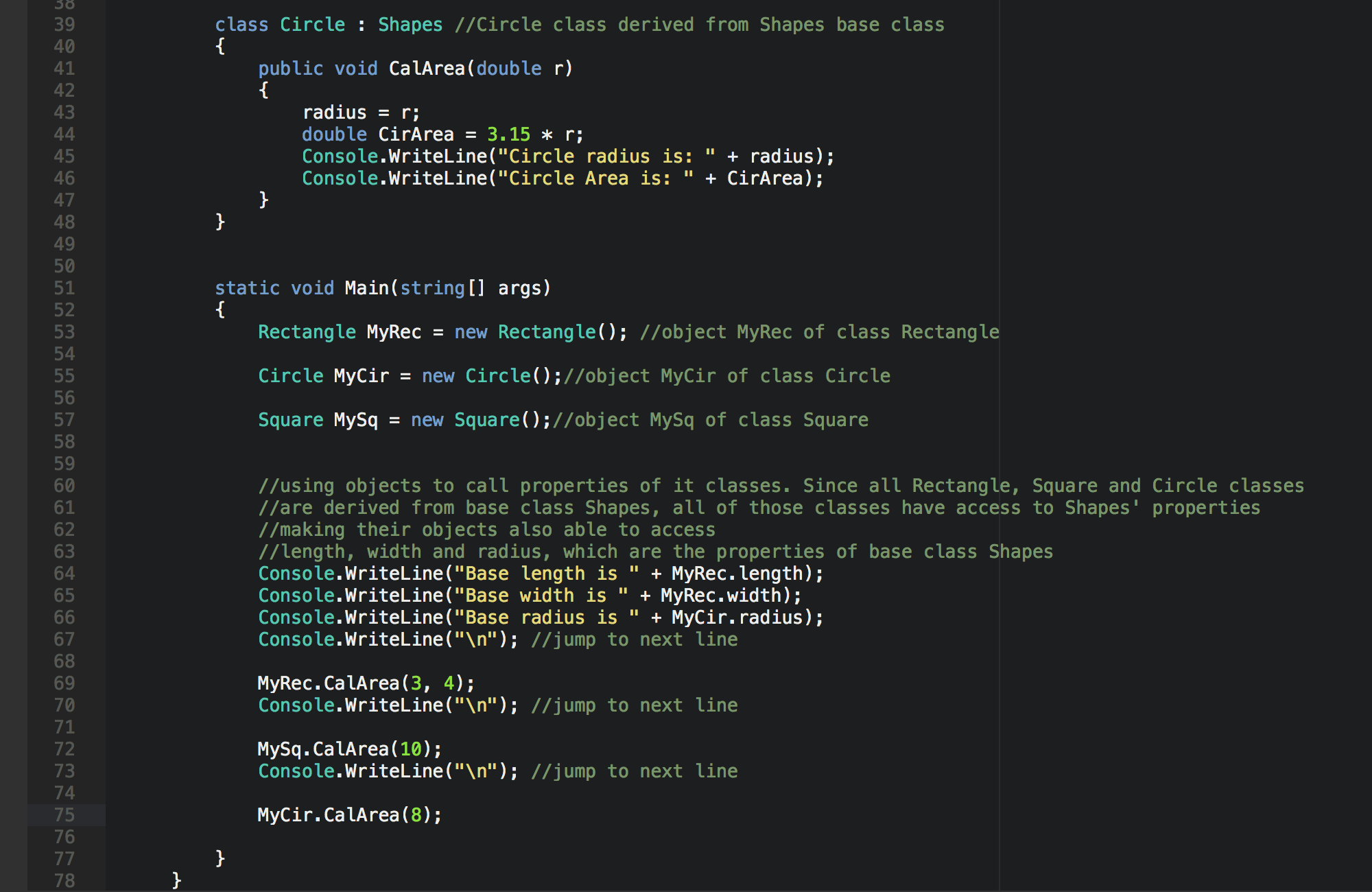
**If you are an onsite student:**

Save it here > CS131-Winter-2020\**IN**\FirstnameLastname/Module9/OOP- Inheritance

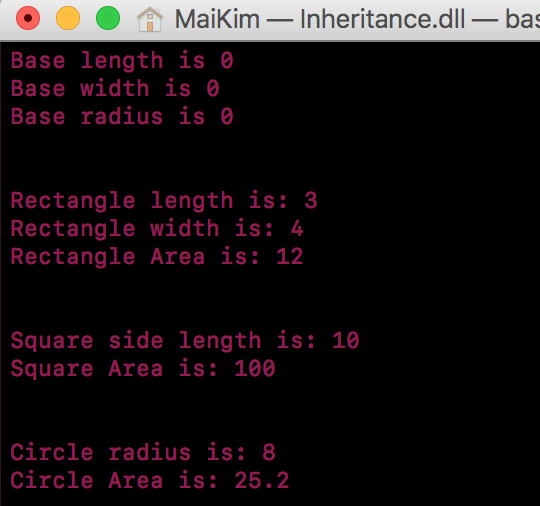
Let's continue using the example we have from last week HOP: Creating a base class named Shape. This time, instead of having specific functions for specified shapes, let's keep our base class Shape as general as possible, and create several derived classes that are more specific.

1) Write the following code into your Program.cs:

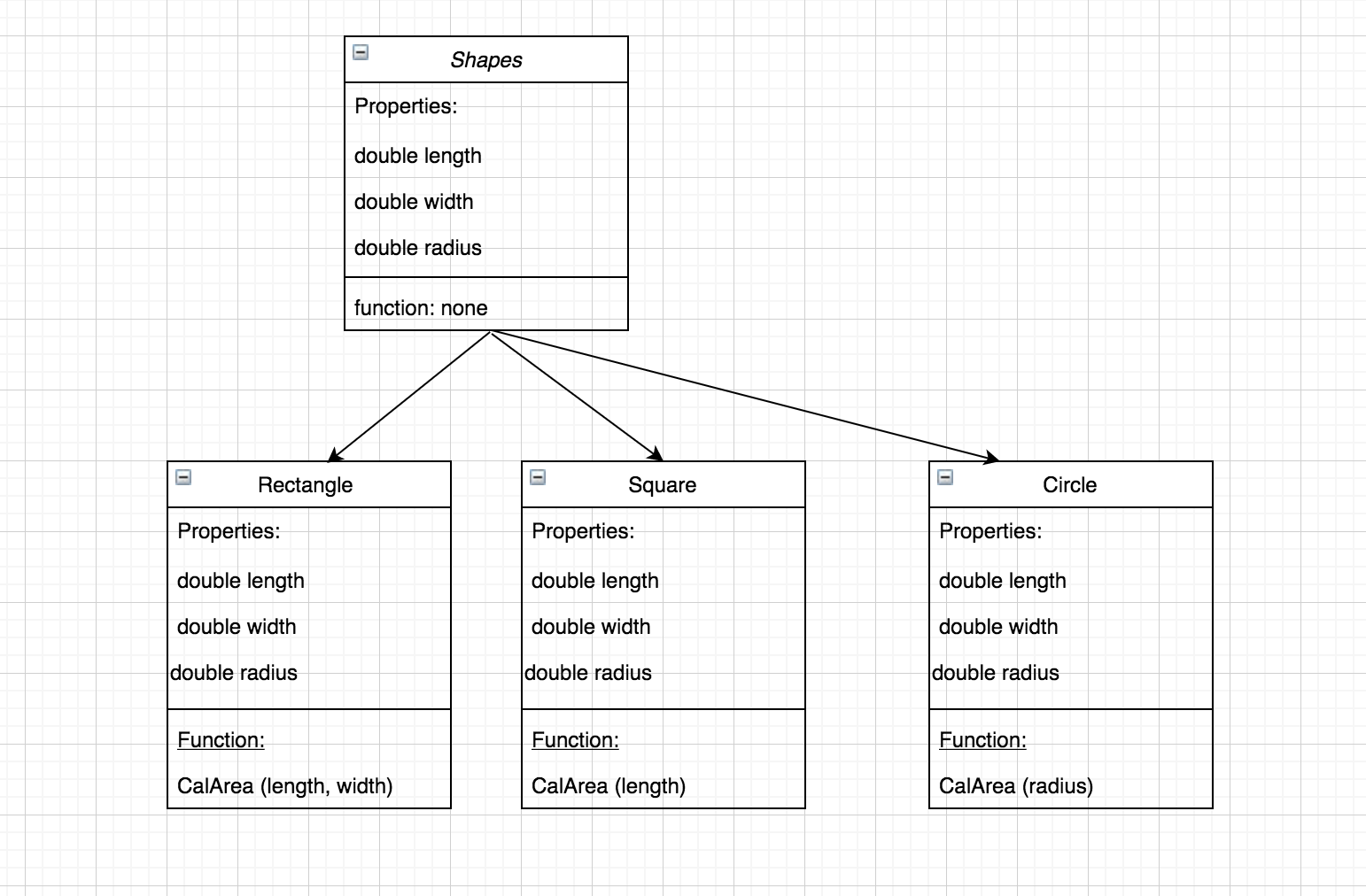




2) Run your program:



Here's a visualization for you to understand this concept and this program better:



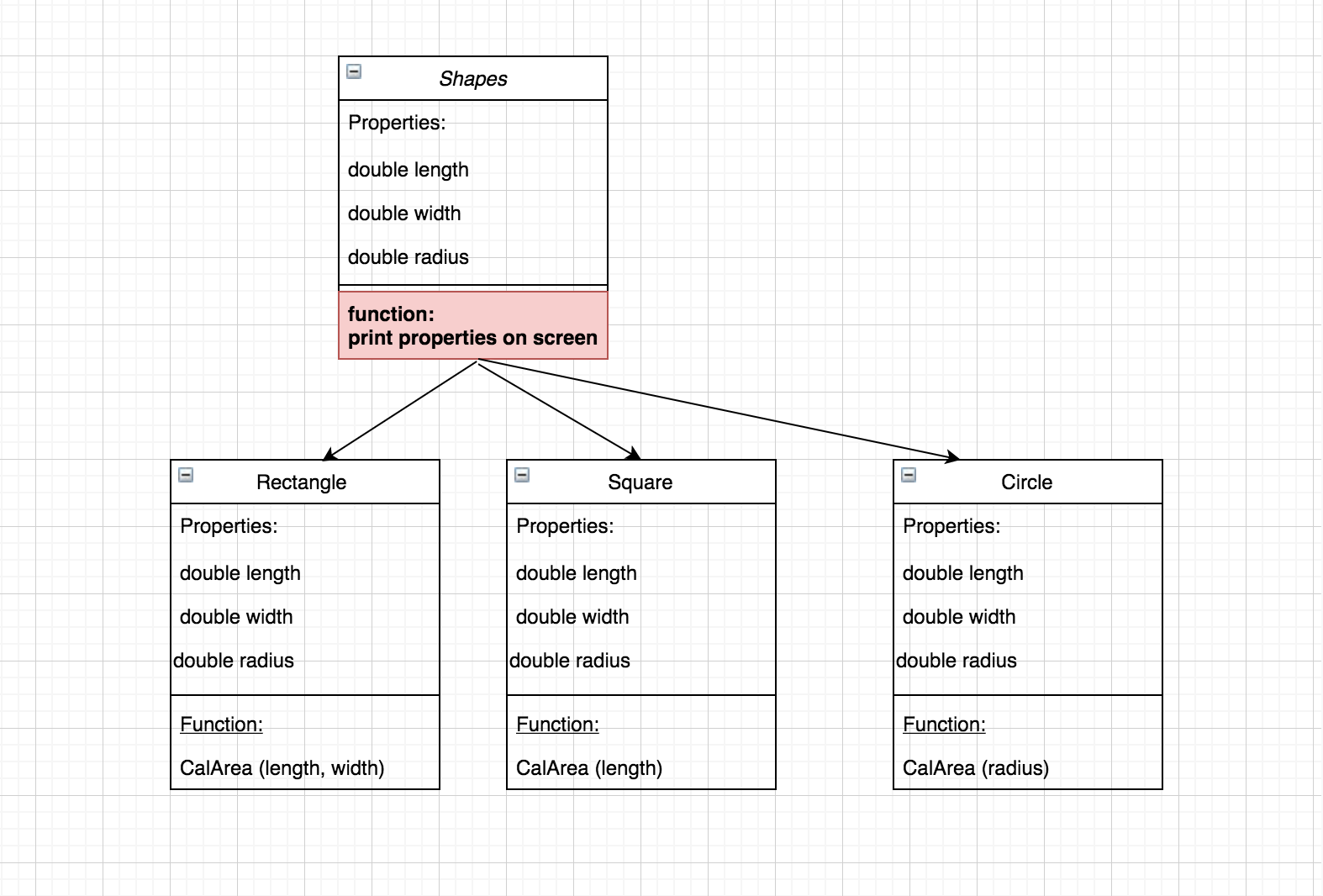
**CHALLENGE:**

Create a method/function in Shapes class that display the base properties, to replace line 64 to 67 of the above code. Then use any derived class to call the method in Main(). Take a screen shot of your code.

When you run the program, result should be the same as above.

Hint:

Changes should be made for highlighted area below:



**Push your work to GitHub**

**Commit changes**

1. Click on the **Home** button > **Changes**
2. Type commit message
3. Select **Commit All and Push**

**Create a pull request**

1. Go to your fork page on GitHub website
2. Near the top left side, change the active branch to your new branch
3. Click on the "New Pull Request" button next to the branch name.