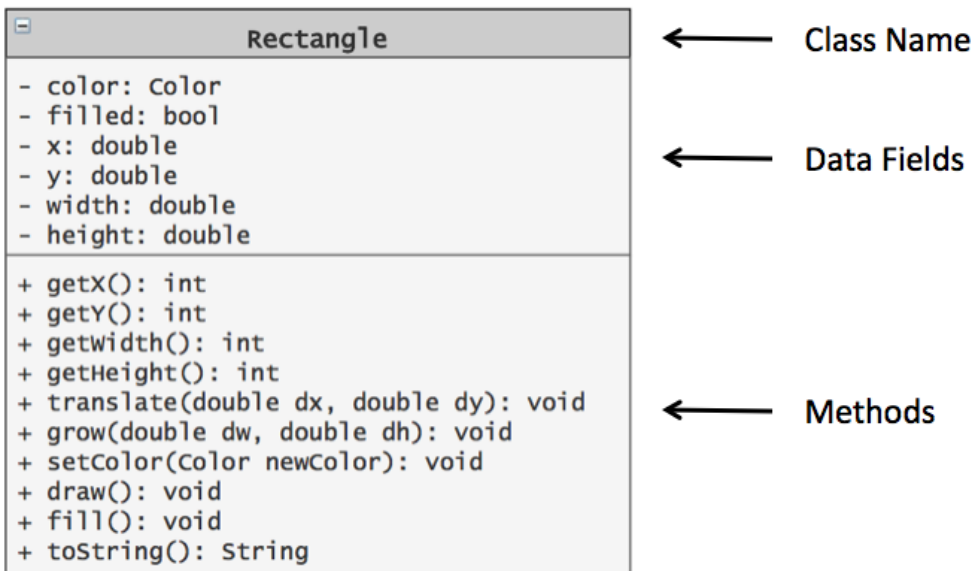


Activity 1 - Classes and Objects

Classes and Objects

Classes and objects are the two main aspects of **object oriented programming**. A **class** defines a new data type, whereas an **object** is an *instance* of a particular class. Objects store *data* belonging to the object in **fields** defined by its class. Objects also have *functionality* that can be activated by calling the **methods** that belong to its class.

A **UML diagram** is a graphical summary of the data and methods in a class. Below is the UML diagram for the Rectangle class from the Simple Java Graphics[1] library written by Cay S. Horstmann.



Critical Thinking Questions

1. Identify one example of a data field given in the above UML diagram for the Rectangle class.
 - a. What is the variable name of the data field?
 - b. What is the data type of the data field?
2. Identify one example of a Rectangle class method that takes no arguments.
 - a. What is the method name?
 - b. How do you know that the method does not take any arguments?
 - c. Does the method return a value? If so, what type of value will it return?
 - d. Do you think this method will modify one or more of the object's data fields? Why or why not?

3. Identify one example of a Rectangle class method that takes one or more parameters.
 - a. What is the method name?
 - b. What is the name and data type of each parameter for the method?
 - c. Does the method return a value? How do you know?
 - d. Do you think this method will modify one or more of the the object's data fields? Why or why not?
4. Imagine that you have a code segment that creates a rectangle (rect1) with the following values for each of its fields:
 - color: Magenta
 - filled: false
 - x: 50.0
 - y: 100.0
 - width: 200.0
 - height: 250.0

Give the code to do each of the following to rect1

- a. Change the color of rect1 to color Violet.
- b. Fill rect1.
- c. Grow the width of rect1 by 100 units and the height by 200 units.
- d. Get the width and height of rect1.

What values do you think will be returned by part d?

Terminology Note

- Methods that return data about the object are called **accessors**.
- Methods that change an object's data are called **mutators**.

A method should do *one thing*, either provide access to object data values or mutate data values. Never both!