CS1400 Lab #12: Class Diagrams

Introduction

In this assignment you will learn about **Class Diagrams**. Class Diagrams are part of a graphical language used to convey object oriented design information, called the Unified Modeling Language, or **UML**. Class Diagrams are used to specify the attributes and behaviors of a class, and to show the static relationships between classes.

Example

Consider a class named Box, that models a cardboard box that you might use to ship something via FedEx. If you were to buy a box to ship something in, what attributes of the box would be important to you? Probably the dimensions of the box, and perhaps its volume. The attributes of the class become its data members. In a well designed class, the class provides methods (behaviors) to give us access to the variables (attributes) of the class. In this case, we will define methods to get and set the dimensions, and one to calculate and return the volume of the box.

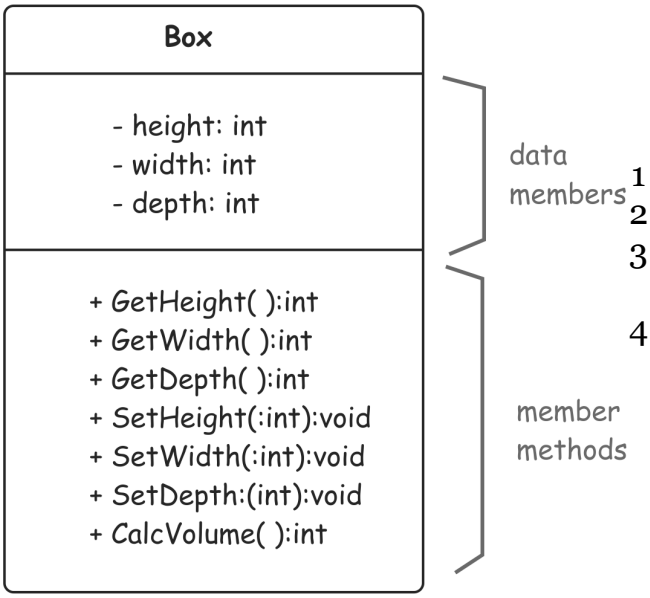
We can represent a class by drawing a class diagram. A class diagram is drawn as a rectangle, and has three basic parts:

1. The top section of the diagram contains the class name. It is usually written in bold text.
2. The center section of the diagram contains the **attributes** or data members of the class.
3. The bottom section of the diagram contains the **behaviors** or member methods of the class.

The figure below shows the class diagram for the Box class.

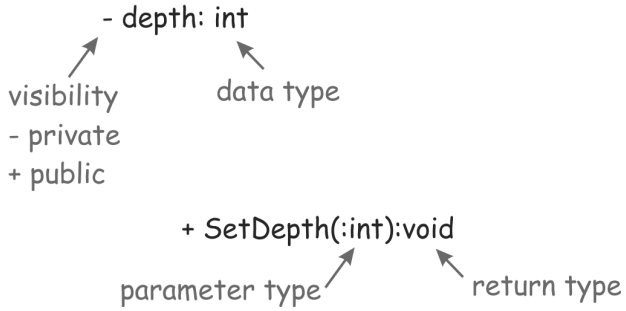
Notice that the declaration of an attribute or data member has three parts:

1. The attribute's visibility,
2. the attribute's name, and
3. the attribute's data type.



The declaration of a behavior or member method is similar. It has:

1. The method's visibility,
2. the method's name,
3. the parameters of the method,
4. the method's return type.



Note that the rules for drawing a class diagram are quite strict. Software developers use class diagrams to convey important design information. If the diagram is not correct, then the wrong information may be conveyed.

Reference

Martin Fowler, *UML Distilled, Third Edition*, Addison-Wesley 2004, pages 35-52.

Programming Problem

It will be helpful if you read the chapter on *Designing Your Own Classes* and the slides for this week.

For this assignment think about the attributes and behaviors for a token dispensing machine that you might find in a carwash or in an arcade. To simplify the problem, this token dispensing machine only takes quarters, no other coin denominations or bills are allowed. When you put a quarter in the machine, you always get one

token in return. Now, design a class that represents this token dispensing machine. Create a class diagram using standard UML notation as explained in this lab. You may hand draw your diagram and take a picture of it, or you can use any drawing software. I often use PowerPoint to make a simple class diagram. You must be able to convert your class diagram into a PDF file or a standard image file (jpg, png, etc) for submission. A PDF file is preferred.

File(s) to Submit:

Place your UML class diagram in a zip file and name the zip file lab_12_your-initials_V1.0.zip. For example, I would name my file lab_12_RKD_V1.0.zip. Submit this assignment as Lab #12 on Canvas. Do not submit any other files.

Grading Guidelines

Description	**Points possible**
Assignment meets grading guidelines: o Your Class Diagram contains a declaration that this is your own work. o Assignment has been properly submitted to Canvas	2
You have a correctly drawn Class Diagram that models a Token Dispensing device.	3
Total	5