Object, Classes, and UML

1-16-2005

Opening Discussion

- Do you have any questions about the reading?
- Have you thought any more about potential game ideas?

Objects and Classes

- From your reading you should have a general idea of what object orientation is.
- What is encapsulation?
- What is the distinction between and object and a class?
- What are the different visibility levels you can use in Java?
- How does visibility of members impact encapsulation?

Car Example

- I want you to break into groups and put together what you think a car would be like in Java. You can either write it in code format (ignore the fact that it will have errors) or you can write it in a comment in code.
- Be as detailed as you can be here and don't be afraid to go out on a limb. Remember that you want to have both data and methods.

Writing Java

- In Java all code goes inside of classes. Methods in a class have innate access to the data in that class.
- Java is not 100% object-oriented. Primitive types in Java (int, double, etc.) are not objects, they have no members. This decision was made for efficiency.
- Public members of a class define a “public interface”. These are the things that are known to outside code. They are hard to change without breaking other code. Data should never be public.

ACMEBank

- Now let's do another example together. This is a rather standard example of a bank account. We will enhance it a bit to help demonstrate some features of Java.

this and Using Members

- When you are writing a method of a class, it has direct access to the member data and methods of that class. You don’t have to use the ‘.’ notation.
- To be explicit, you can use the ‘this’ keyword which implies the object that the method was invoked on.

static

- The term static in the C-family languages implies something like “there is only one”. This is true in Java as well.
- A static member or method is associated with the class itself, not with an object/instance of that class.
- They can be reached or invoked without having an object of that class too.

main in Java

- Like C/C++, Java programs always begin in a special method named main. However, in Java main is a static method of a class (remember there are no stand alone functions). Every class can have its own main which can be very helpful for debugging.
- The signature of main is
  - public static void main(String[] args) { }

Documentation Comments

- In Java, comments that start with /** are documentation comments. These comments are used by javadoc to produce HTML documentation.
- These comments should go above all classes and methods. Inside the comment you start with a summary sentence then have a paragraph describing the class or method. After that can come certain “tags” that begin with @.

Minute Essay

- What are the reasons you should never make data public?