Inheritance in Java

1-24-2006

Opening Discussion

- Do you have any questions about the reading?
- Do you have any questions about your assignment?
- Variables and scope in Java.
- When to make new classes.
- Understanding static. To do this we need to see code that uses more than one object.
- Starting small (a quadratic program).

Immutable Objects

- We start with a little detour.
- In your reading you have inevitably come across the term immutable. What does this mean?
- What are the advantages and pitfalls of immutability?
- How can you write code that takes advantage of immutability?
- The entire paradigm of functional languages is based on the idea that data is immutable.

Types and Polymorphism

- What are types in programming languages? What are some examples of types you are used to?
- What is polymorphism?
- Why is polymorphism important to our programming?
- Could you write polymorphic code in C?

Inheritance

- What is inheritance?
- What is the primary question you should ask yourself when you are considering using inheritance?
- How is inheritance denoted in UML diagrams?
- I want you to get together in groups of 2 or 3 and come up with a example of an inheritance hierarchy. Be inventive and let things get as complex as you can. If you come up with something interesting enough, we can use it for our programming example later on.

Two Sides of Inheritance

- Inheritance provides two functions.
- The original motivation for inheritance, and the root of the term, is that a subclass implicitly gets a copy of everything in the class that it is inheriting from. This means it has all data and functions. It can't directly access the things that are private.
- Inheritance also provides subtyping. If class B inherits from A, then any code that uses A will work with an object of type B. This is how we get our polymorphism in Java. We write code that works with supertypes and it automatically works with subtypes. This type of polymorphism is called inclusion polymorphism.

More Questions

- What is a virtual function?
- How are they related to inheritance?
- What methods in Java are virtual?
- How do you specify that a method is completely virtual and has no implementation at all in a given class?

Restrictions in Java

- Java places some restrictions on inheritance to simplify the language. The main restriction is that you can only extend one class. Doing otherwise, multiple inheritance, tends to make things very complex.
- There are times when you want to have a class be a subtype of two different types though. To allow this Java has a construct called an interface. Interfaces have no data (they can have static data) and all methods in them are abstract. They only define what you can do with them, not how to do it. You can implement as many interfaces as you want.

Code

- Let's actually write some code that demonstrates inheritance, method overriding, etc.

Minute Essay

- Both code reuse and polymorphism help you write less code. How does each accomplish this?
- Remember to post your design today and send me an e-mail letting me know it is there.
- The assignment itself is due on Thursday by midnight.
- I will be in this room this afternoon to help people. I will have to leave at 5:30 or earlier (Spurs game). Normally, if people are here I'll stay as late as 6pm. Otherwise I might leave earlier as Tuesday is my skating night.