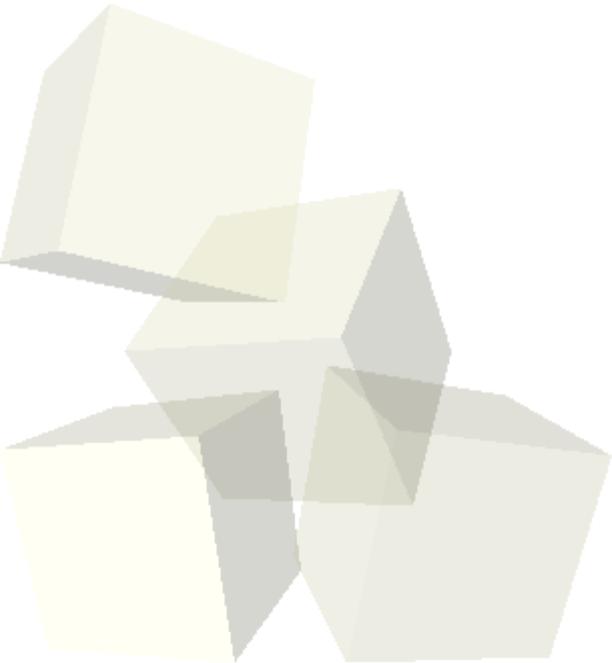




Types and Primitive Types

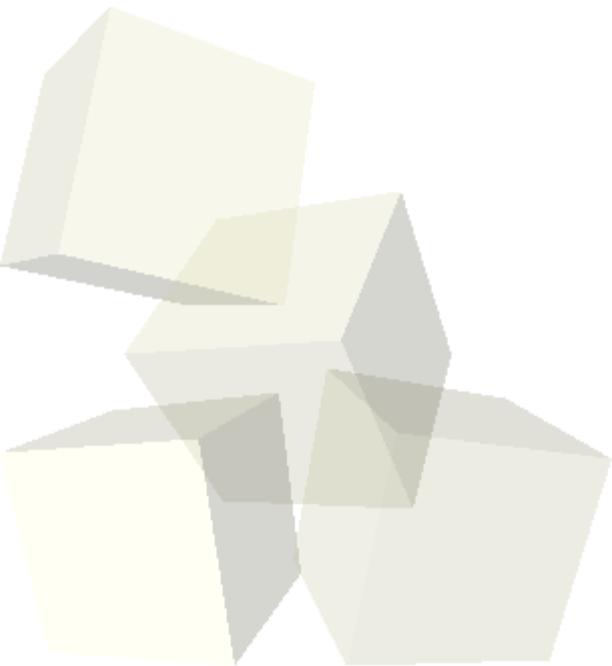
10/22/2007





Opening Discussion

- Let's look at some solutions to the interclass problem.





Primitive Types in Java

- We can break the types in Java into two distinct categories: primitive types and reference types.
- Primitive types are more basic and correspond to elements in the computer hardware.
 - ◆ Integer types – byte, short, int, long
 - ◆ Floating point types – float, double
 - ◆ Logic values – boolean
 - ◆ Character values – char
- Your book gives ranges. Unless otherwise required I will use int for all integers and double for all numbers including a fractional part.



- The most general numeric type is double. We will use these to represent real numbers.
- Technically they are double precision floating point numbers and aren't real numbers. This can have some odd effects, but we will try to ignore these in this class.
- Use final to say name refers to a constant.
- double literals
 - ◆ If you type in a number that contains a decimal point or the letter e for scientific notation, Java will treat it as a double literal.

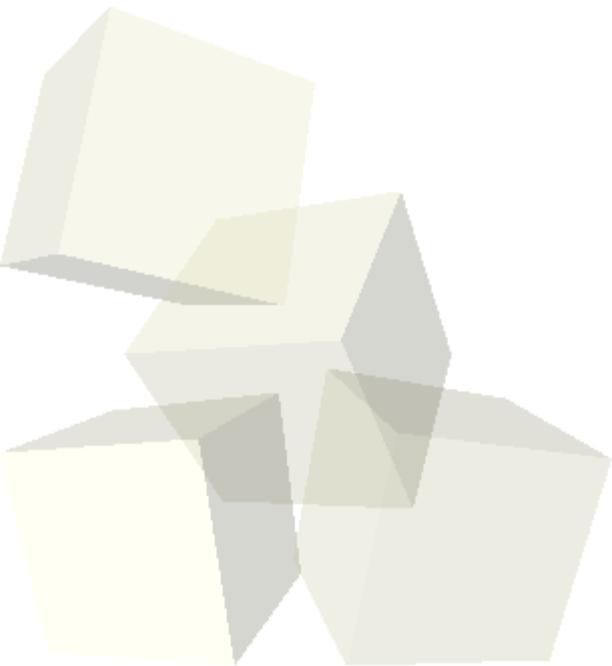


- Expressions are things we put into a program that have a type and a value.
- We can combine simple values with operators to build complex expressions.
- Some operators in Java you are used to: +, -, *, /.
- A single equals sign is used for assignment in Java. Assignment is an operator and the expression has the value of the value assigned.
 - ◆ In Java the use of = is not like it is in math. When you see = in Java, think that you are storing a value in a variable. It does not imply equality like in math.
- Programs are made with statements. These end with ;.



The Math Class

- There is a class in `java.lang` that helps us to do normal mathematical operations.
- The `Math` class defines some constants and many different mathematical functions that we can use in our programs if we wish.
- Let's look in the `API` to see what it has.

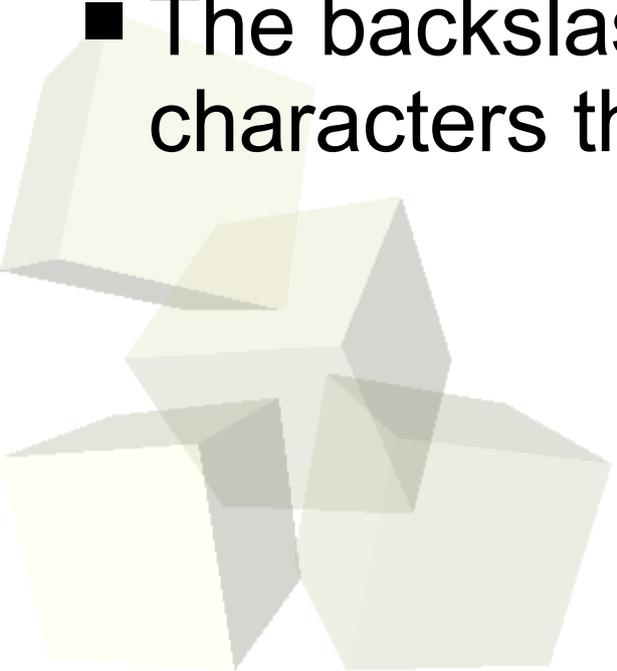




- For numbers that should not have a fractional part you should use an int.
- You will find that I use an int by default and only use double when I really need the fractional part.
- The / operators used on ints does integer division. It only gives you the whole number. Use % to get the remainder.
- It is common to use shortcut operators to do common tasks where a variable is changed in Java.
 - ◆ $n++$ is the same as $n=n+1$
 - ◆ $n+=2$ is the same as $n=n+2$
 - ◆ You can put = after any binary operator.



- Single characters are represented by the char type. This is actually a number that is interpreted through Unicode. They are non-negative values between 0 and 2^{16} , but we will almost always treat them as just characters.
- Character literals have single quotes around a single character.
- The backslash can be used to produce special characters that you can't type easily into Java.





- You “met” this type in Alice. It is either true or false and is used for logical operations.
- Relational operators can produce boolean valued expressions (`==`, `!=`, `<`, `>`, `<=`, `>=`).
- Boolean expressions can be combined with boolean operators (`&&`, `||`, `!`).
- The `&&` and `||` operators are short-circuit. This means that if the first operand determines the value, the second one isn't evaluated. This is not only efficient, it can be helpful for not doing dumb things like dividing by zero.



- Why do you think that Java provides us with so many different types?
- Interclass Problem – Write a program where you declare at least one integer type variable, one floating point variable, one char, and one boolean. Play with them a bit to see what you can do with them.

