

CS61B Summer 2006

Instructor: Erin Korber

Lecture 3, 28 June

Reading for tomorrow: Ch. 4 (to end), Ch. 12 (pg 375-381)

1 More on variables

- Java is pass-by-value
 - That means by *copy*
 - Method doesn't change the contents of the calling argument variable.
- Initialization
 - Instance variables - get a default value
 - * `null` for object references
 - * `0` or `0.0` for numeric primitives
 - * `false` for booleans, null character for `char`
 - Local variables - no default value. Must be initialized!

2 Advanced methods

- Methods that depend on state (instance variables)
- Methods that take arguments and return values
 - Implicit promoting to larger types
 - Explicit casting to smaller types
 - Doing this for primitives only right now - will talk about objects later when we discuss inheritance and polymorphism.

3 Constructors

- To make objects, classes need to have *constructors*.
- Have been using the default constructor already.
- Additional things to do with constructors:

- Arguments to the constructor
- Do things in it (besides creating the object)
- Default constructor vanishes when you write others - must rein-
sert if you want it.

4 The Java API

One of the reasons that people like to program in Java is that so much is provided for you, and it is so well documented in the API. Good use of the API will make you a happier and more efficient Java programmer.

- Organized into *packages*
 - `java.lang` is “pre-imported” - you’ve already been using it
 - Others you will use often: `java.util`, `java.io`
 - Can import packages to save typing the full name of a class
- `ArrayList` - a lot like an array, but:
 - size not fixed at creation time
 - easy `add`, `remove`, and `contains` methods
 - part of `java.util` package
- I/O
 - should import `java.io`
 - Will use prewritten code for now
 - the `parseInt` method