

### Administrivia

- "Lecture topics and assignments" Web page.
- Class e-mail list.
- More things to note in syllabus — quizzes, "open lab" office hours.
- Details about first homework on Web by Thursday.

Slide 1

### What's An Object?

- Object — set of data (attributes) and associated functions (methods, behaviors, operations) that can act on data.
- Objects interact by calling each other's methods, or by sending each other messages.
- Often makes sense to have many similar objects — hence "classes".

Slide 3

### "Object Orientation"?

- A "programming paradigm" — contrast with procedural programming, functional programming, etc.
- No accepted-by-all definition, but most definitions mention encapsulation:
  - Data and functionality grouped together into "objects".
  - Some data/functionality is hidden.
- Origins in simulation/modeling, where the goal is to model complex systems consisting of many (real-world) objects.

Slide 2

### What's a Class?

- Can be thought of as a blueprint for objects of a given type; individual objects are "instances" of the class.
- Defines attributes and methods each object will have (instance variables/methods), attributes and methods shared by all objects of a class (class variables/methods).
- Public interface — attributes and methods visible from outside the class.

Slide 4

### Java and Object Orientation

- Java is not purely object-oriented — also includes “primitive types” for efficiency — but it's much more strongly object-oriented than a hybrid language such as C++.
- Java programs consist of definitions of classes.
- Java variables (except primitives) are references to objects, classes define types.

Slide 5

### Polymorphism (Short Version)

- “Many shapes” — something that works with many types.
- E.g., a function that works on Mammals should work on Dogs, Cats, ...

Slide 7

### Inheritance (Short Version)

- Given a class, it can be useful to define specialized versions — “subclasses”.
- A subclass inherits attributes and operations from its superclass (which can in turn have a superclass ...).
- Subclasses also form “subtypes” — e.g., if Dog is a subclass of Mammal, can use a Dog anywhere we need a Mammal.

Slide 6

### UML Class Diagrams

- “Unified Modeling Language” — formal graphic representation of software analysis and design.
- We will mainly use class diagrams:
  - Box representing a class has name, attributes, operations.
  - Subclass points to its superclass (represents the path to follow to figure out inheritance).

Slide 8

### Java and Object Orientation, Continued

- Classes, attributes, methods have varying "visibilities" (from public to private).
- All functions are methods of some class.
- Every Java class can have a main program. (How is that possible? Because with Java there's no link step — compare C++ compile / link with Java compile / execute runtime.)
- Let's write some code ...

Slide 9

### Minute Essay

- Was there anything today that was particularly unclear?

Slide 10