

Software analysis and design

Module 2: Concepts of Object Orientation

Objectives

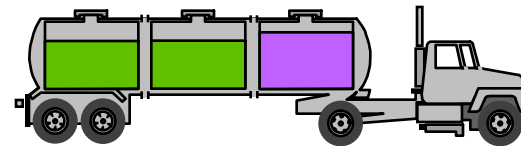
- Describe abstraction, encapsulation, modularity, and hierarchy.
- Describe the physical structure of a class.
- Describe the relationship between a class and an object.
- Define polymorphism and generalization.

Where Are We?

- What is an object?
- Four principles of OO
- What is a class?
- Polymorphism and generalization

What Is an Object?

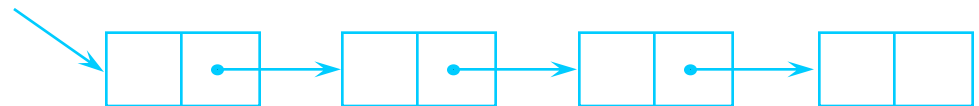
- Informally, an object represents an entity, either physical, conceptual, or software.
- Physical entity
- Conceptual entity
- Software entity



Truck



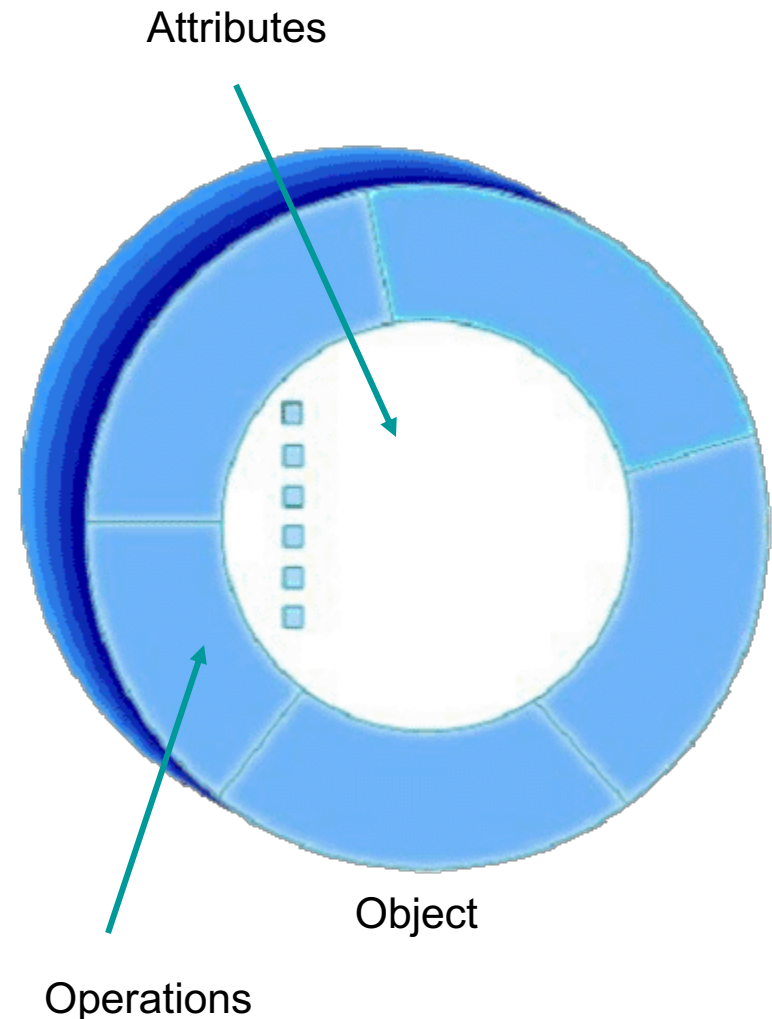
Chemical Process



Linked List

A More Formal Definition

- An object is an entity with a well-defined boundary and *identity* that encapsulates *state* and *behavior*.
 - State is represented by attributes and relationships.
 - Behavior is represented by operations, methods, and state machines.

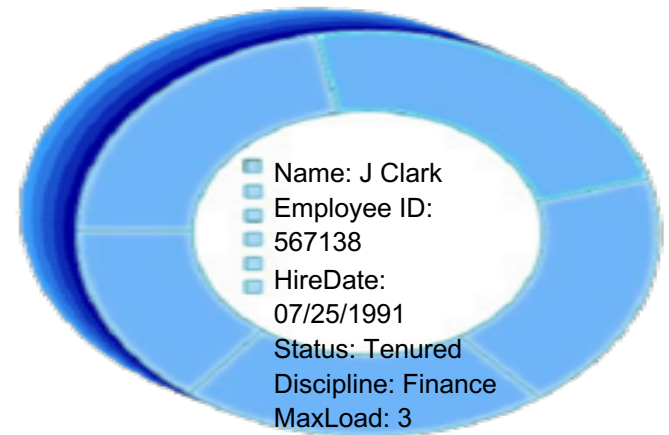


An Object Has State

- State is a condition or situation during the life of an object, which satisfies some condition, performs some activity, or waits for some event.
- The state of an object normally changes over time.



Name: J Clark
Employee ID: 567138
Date Hired: July 25, 1991
Status: Tenured
Discipline: Finance
Maximum Course Load: 3 classes



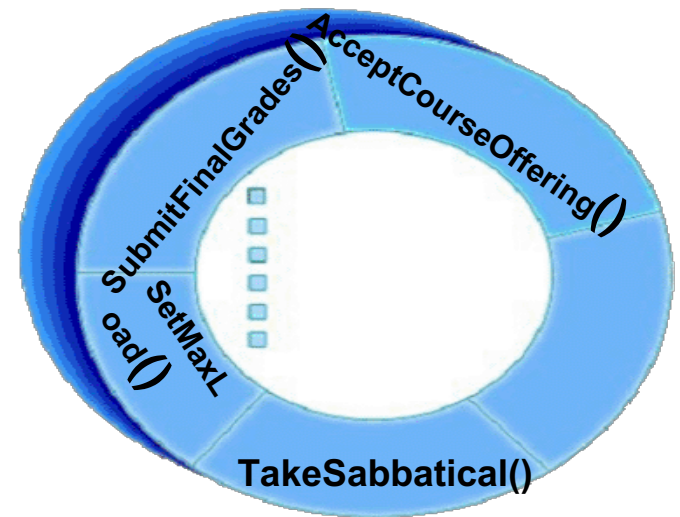
Professor Clark

An Object Has Behavior

- Behavior determines how an object acts and reacts.
- The visible behavior of an object is modeled by a set of messages it can respond to (operations that the object can perform).



Professor Clark's behavior
Submit Final Grades
Accept Course Offering
Take Sabbatical
Set Max Load



Professor Clark

An Object Has Identity

- Each object has a unique identity, even if the state is identical to that of another object.



Professor “J Clark”
teaches Biology

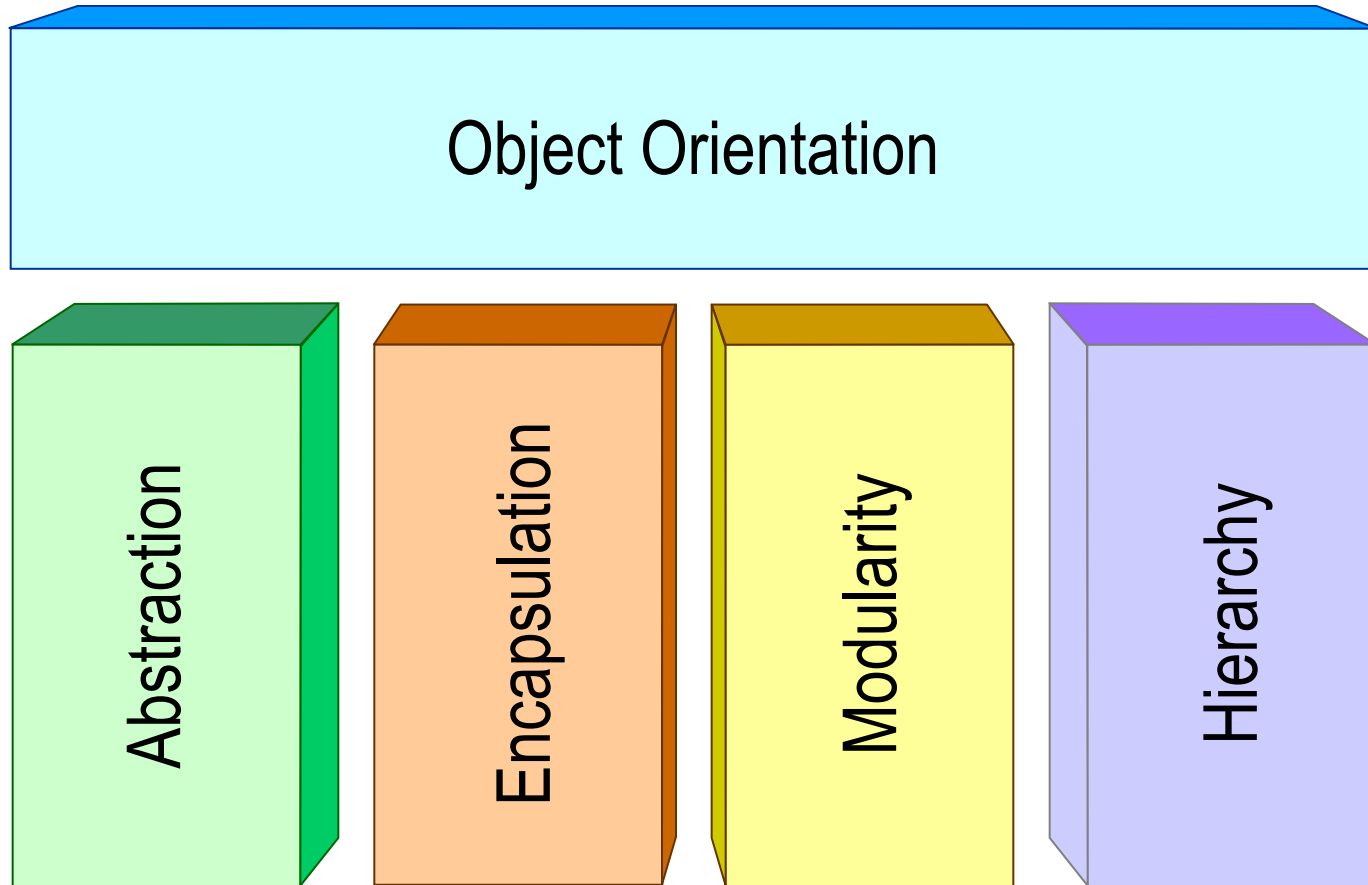


Professor “J Clark”
teaches Biology

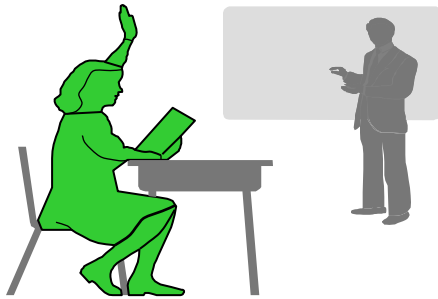
Where Are We?

- What is an object?
- **Four principles of OO**
- What is a class?
- Polymorphism and generalization

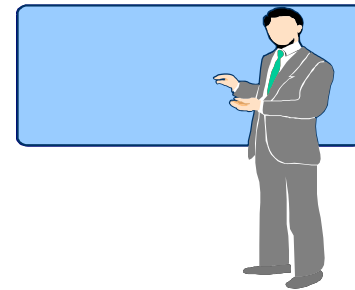
Basic Principles of Object Orientation



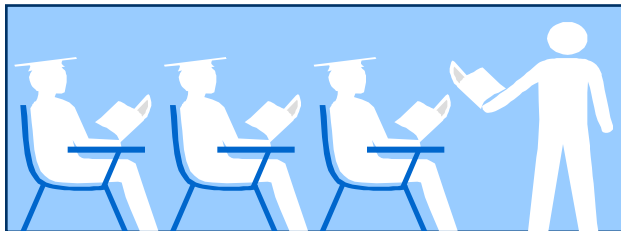
Example: Abstraction



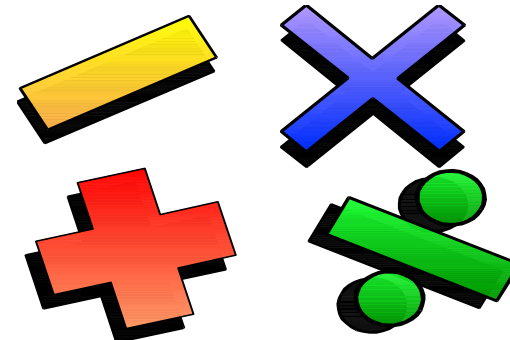
Student



Professor



Course Offering (9:00 a.m.,
Monday-Wednesday-Friday)



Course (e.g. Algebra)

What Is Abstraction?

- The essential characteristics of an entity that distinguishes it from all other kinds of entities.
- Defines a boundary relative to the perspective of the viewer.
- Is not a concrete manifestation, denotes the ideal essence of something.

What Is Encapsulation?

- Hides implementation from clients.
 - Clients depend on interface.



Improves Resiliency

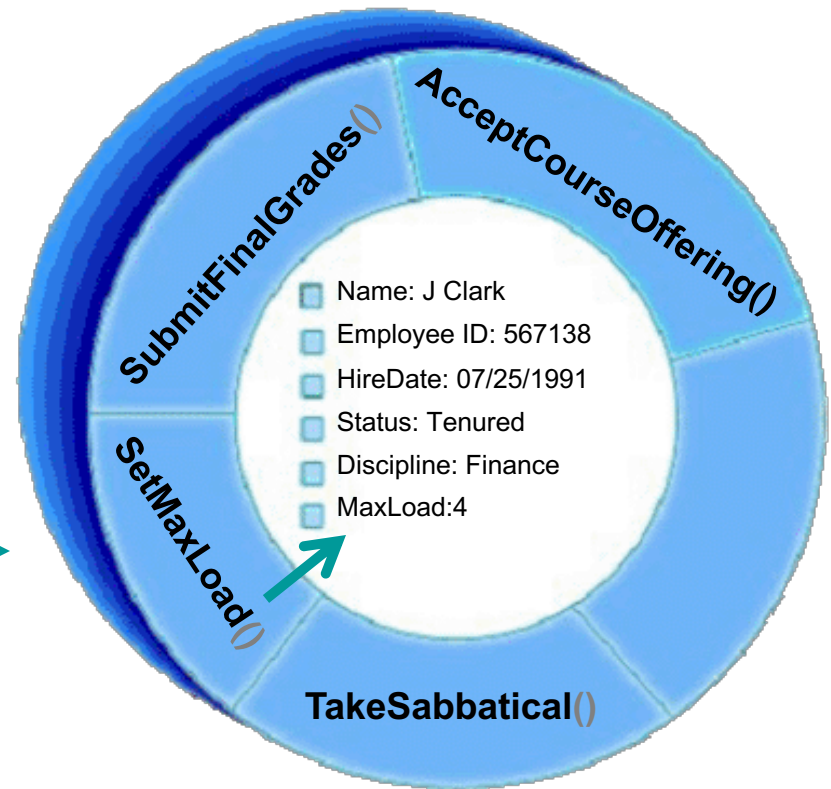
Encapsulation Illustrated

- Professor Clark needs to be able to teach four classes in the next semester.

SetMaxLoad(4)

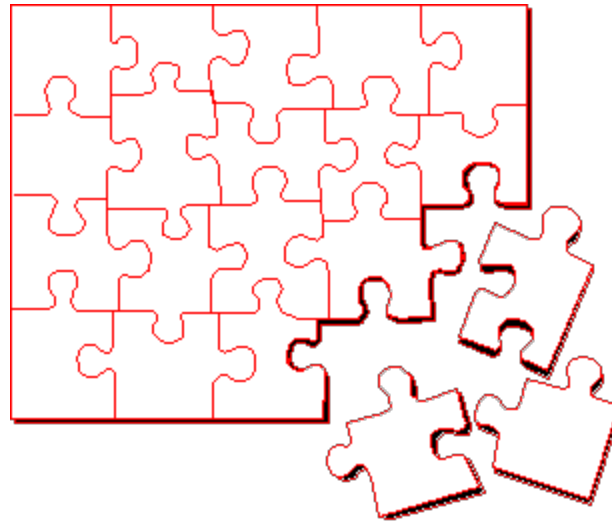


Professor Clark



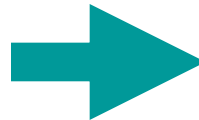
What Is Modularity?

- Breaks up something complex into manageable pieces.
- Helps people understand complex systems.

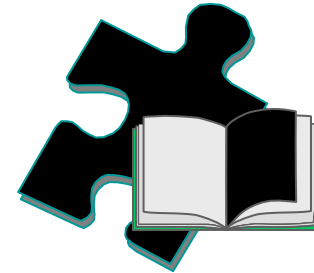


Example: Modularity

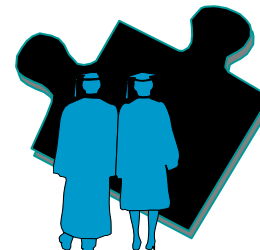
- For example, break complex systems into smaller modules.



Billing System

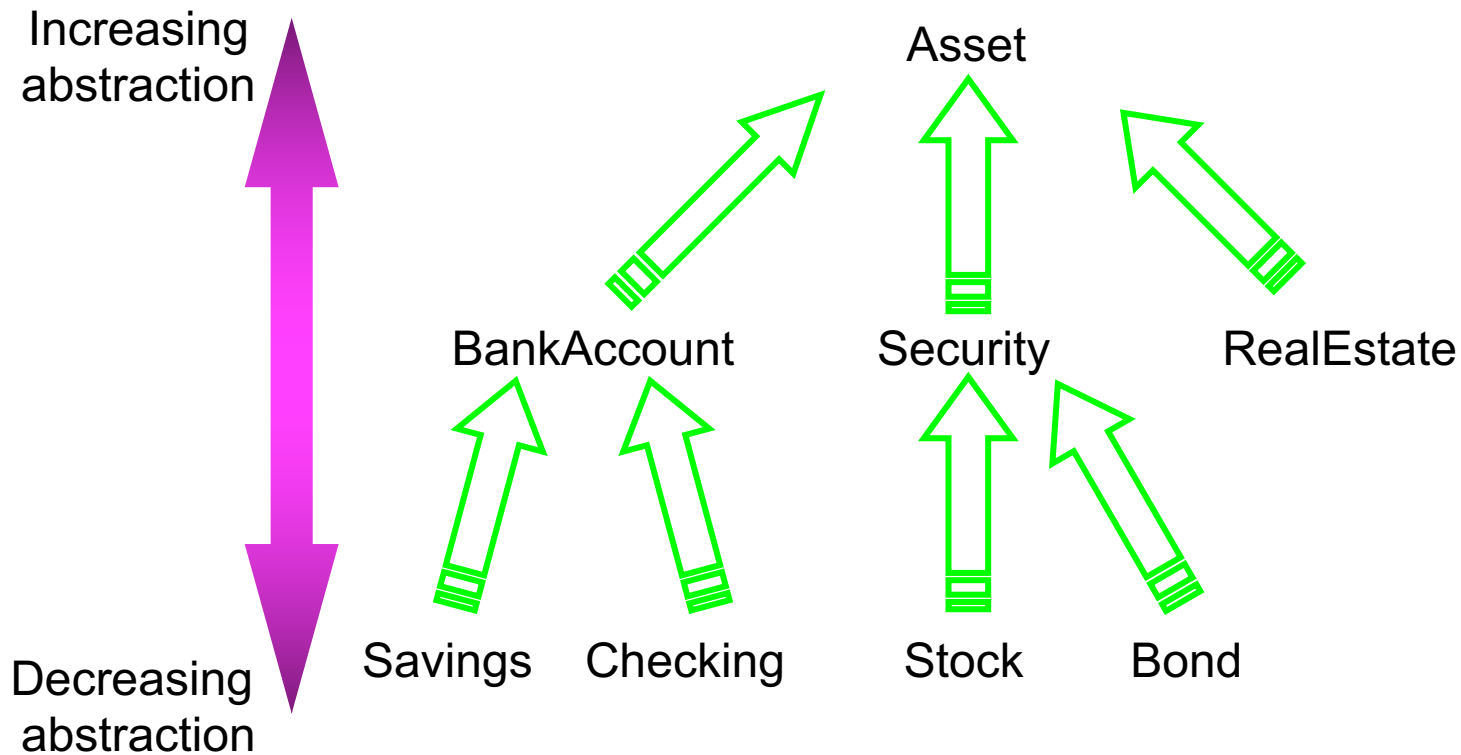


Course Catalog System



Student Management System

What Is Hierarchy?



Elements at the same level of the hierarchy should be at the same level of abstraction.

Where Are We?

- What is an object?
- Four principles of OO
- **What is a class?**
- Polymorphism and generalization

What Is a Class?

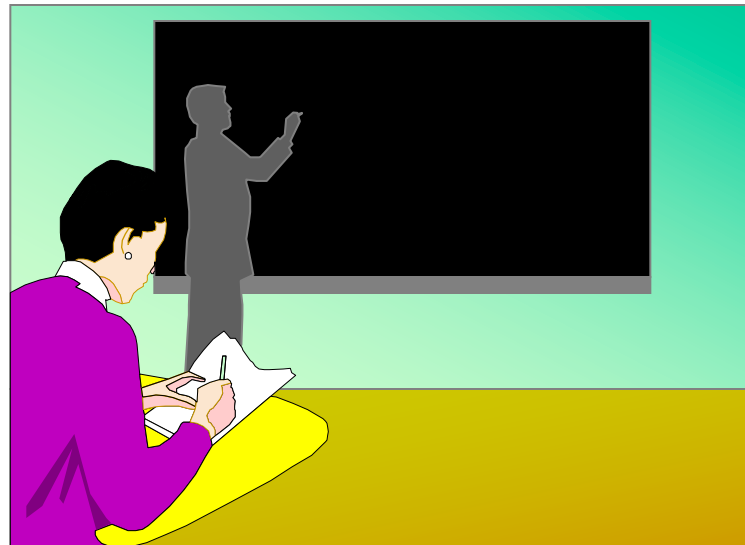
- A class is a description of a set of objects that share the same *attributes, operations, relationships*, and *semantics*.
 - An object is an instance of a class.
- A class is an abstraction in that it
 - Emphasizes relevant characteristics.
 - Suppresses other characteristics.

What Is a Class?

Class
Course

Properties

Name
Location
Days offered
Credit hours
Start time
End time

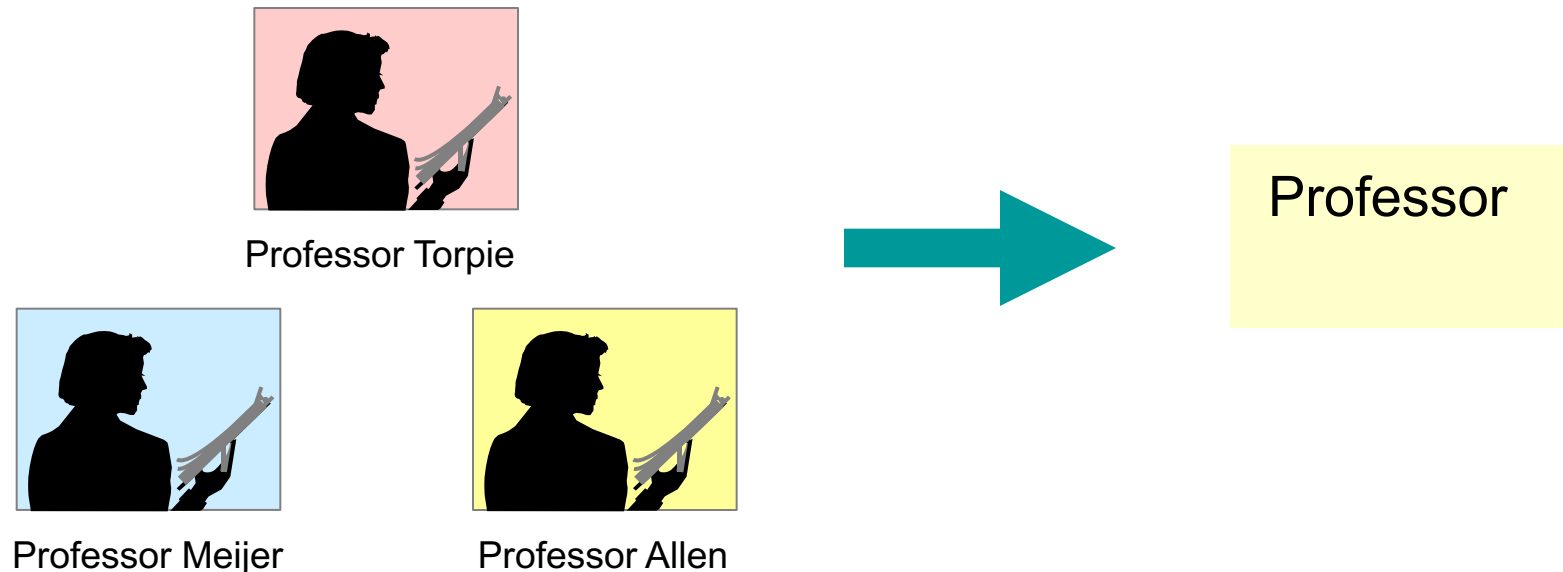


Behavior

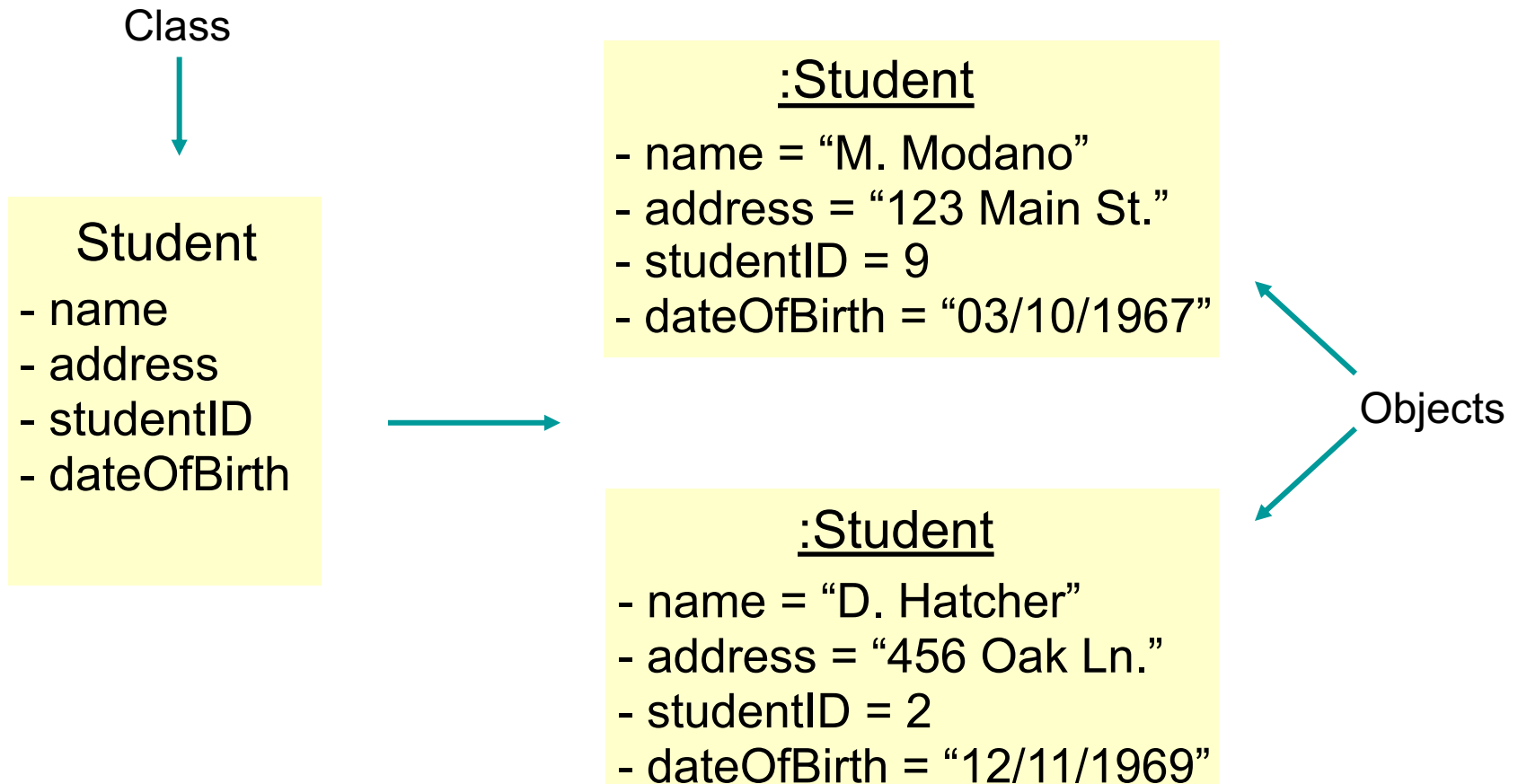
Add a student
Delete a student
Get course roster
Determine if it is full

The Relationship between Classes and Objects

- A class is an abstract definition of an object.
 - It defines the structure and behavior of each object in the class.
 - It serves as a template for creating objects.
- Classes are not collections of objects.

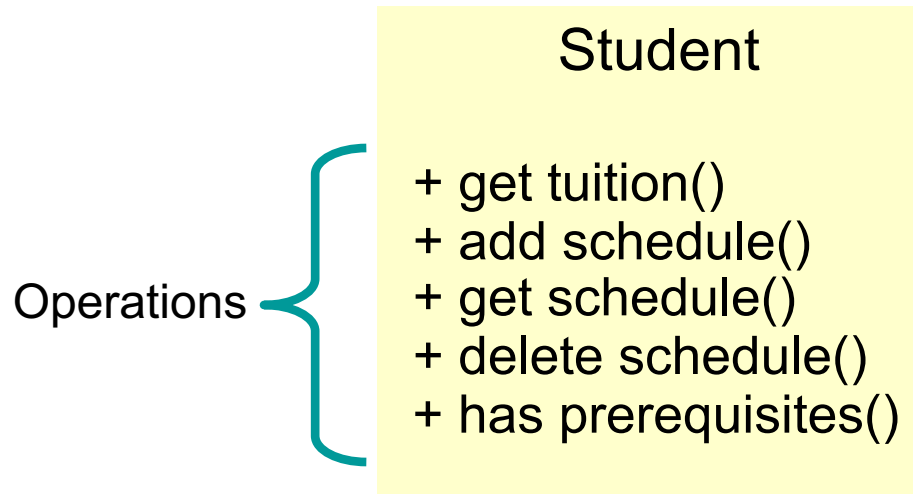


Attributes in Classes and Objects



What Is an Operation?

- A service that can be requested from an object to effect behavior. An operation has a signature, which may restrict the actual parameters that are possible.
- A class may have any number of operations or none at all.

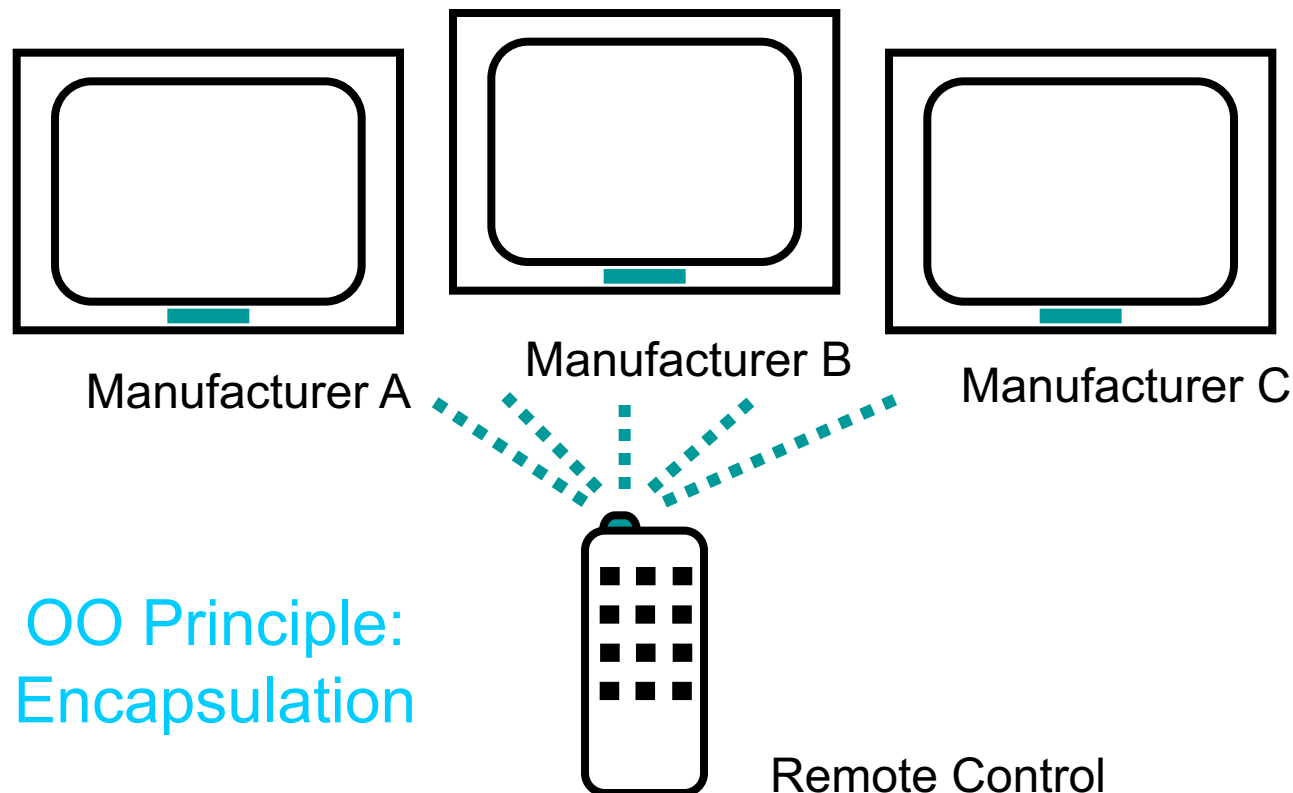


Where Are We?

- What is an object?
- Four principles of OO
- What is a class?
- **Polymorphism and generalization**

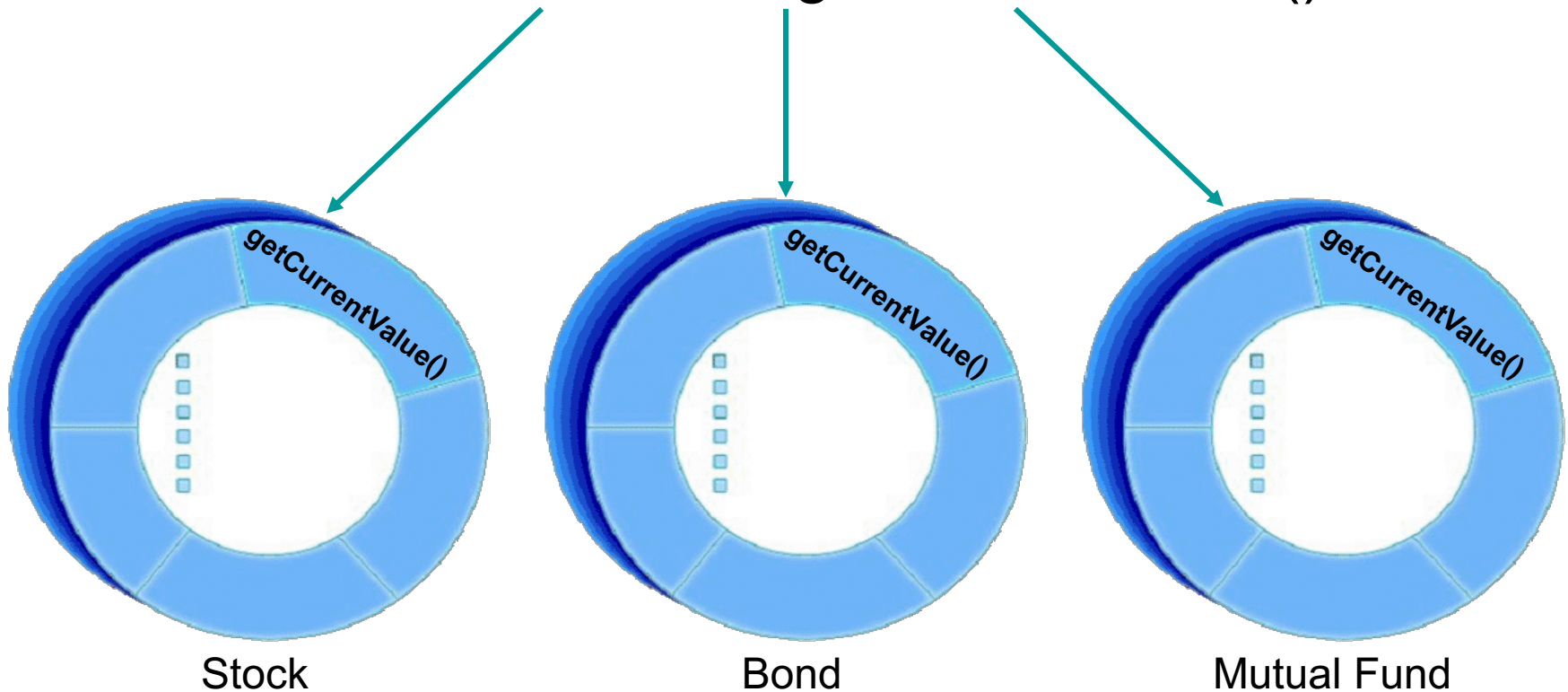
What Is Polymorphism?

- The ability to hide many different implementations behind a single interface



Example: Polymorphism

`financialInstrument.getCurrentValue()`

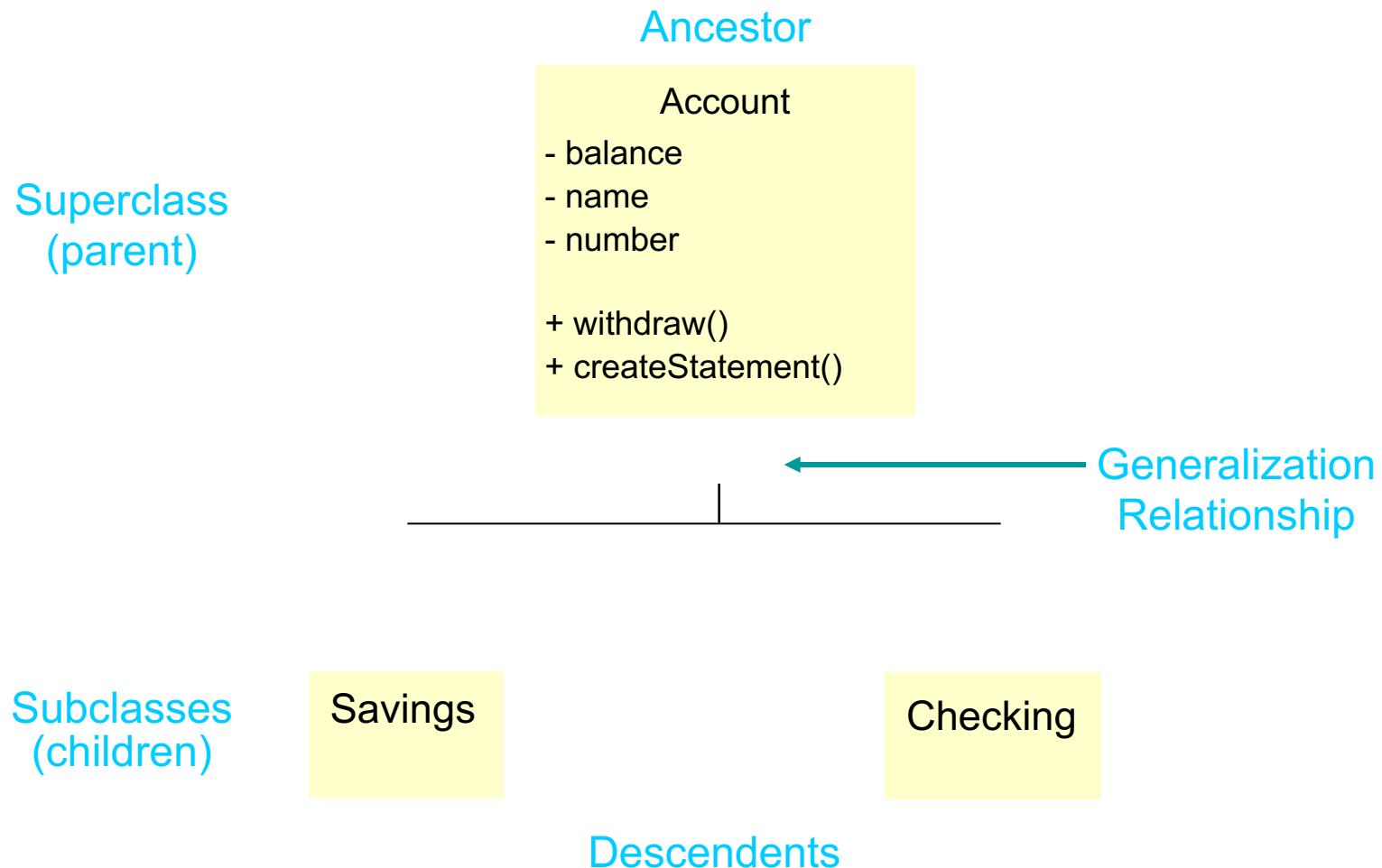


What Is Generalization?

- A relationship among classes where one class shares the structure and/or behavior of one or more classes.
- Defines a hierarchy of abstractions in which a subclass inherits from one or more superclasses.
 - Single inheritance.
 - Multiple inheritance.
- Is an “is a kind of” relationship.

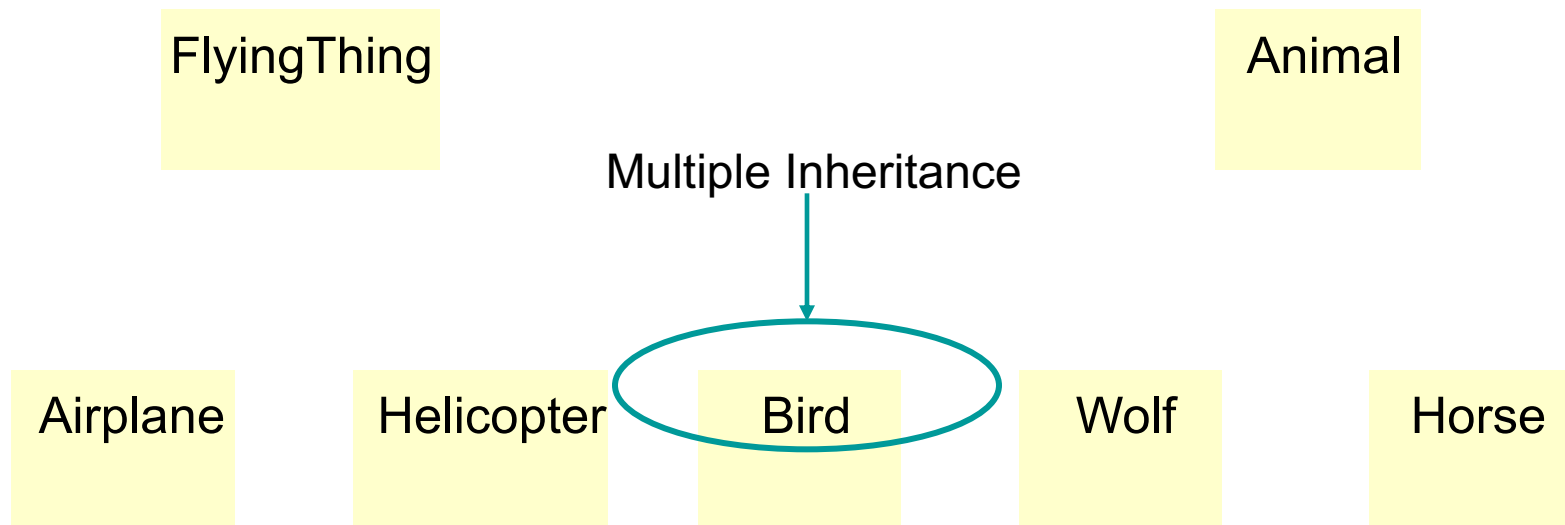
Example: Single Inheritance

- One class inherits from another.



Example: Multiple Inheritance

- A class can inherit from several other classes.



Use multiple inheritance only when needed and
always with caution!

What Is Inherited?

- A subclass inherits its parent's attributes, operations, and relationships.
- A subclass may:
 - Add additional attributes, operations, relationships.
 - Redefine inherited operations. (Use caution!)
- Common attributes, operations, and/or relationships are shown at the highest applicable level in the hierarchy.

Review

- What is an object?
- What are the four principles of object orientation? Describe each.
- What is a class? How are classes and objects related?
- What is an attribute? An operation?
- Define polymorphism. Provide an example of polymorphism.
- What is generalization?