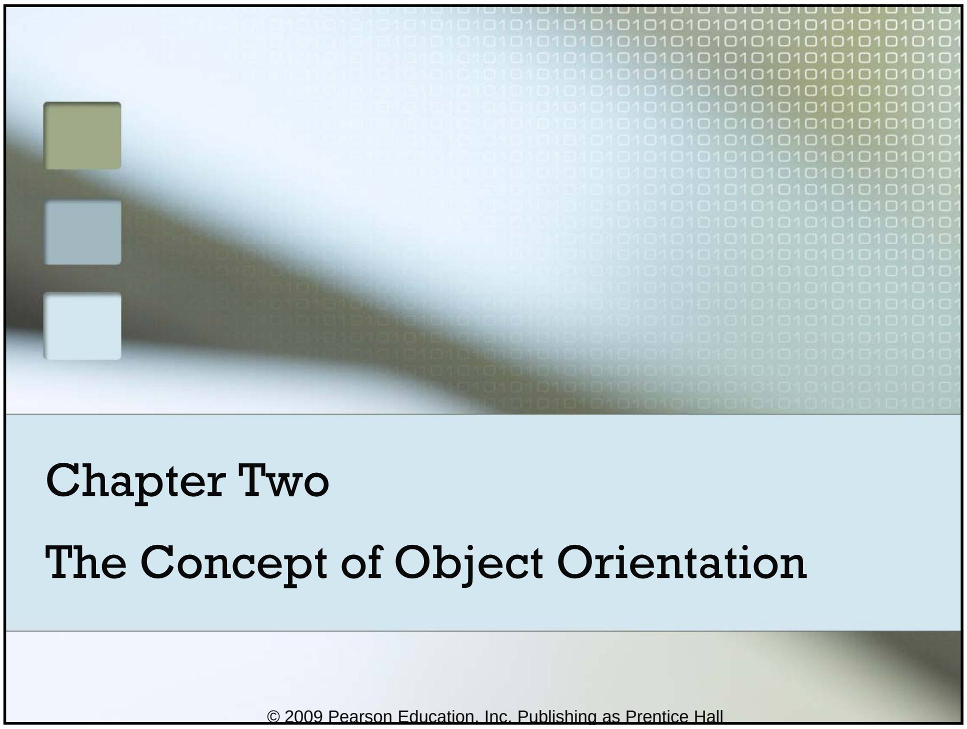
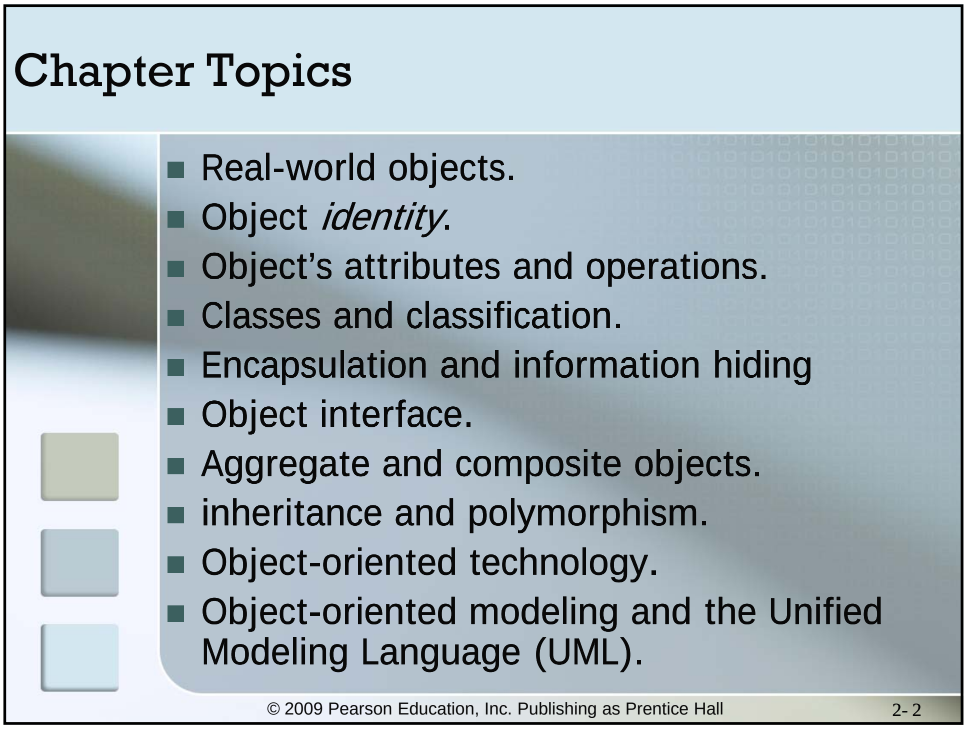


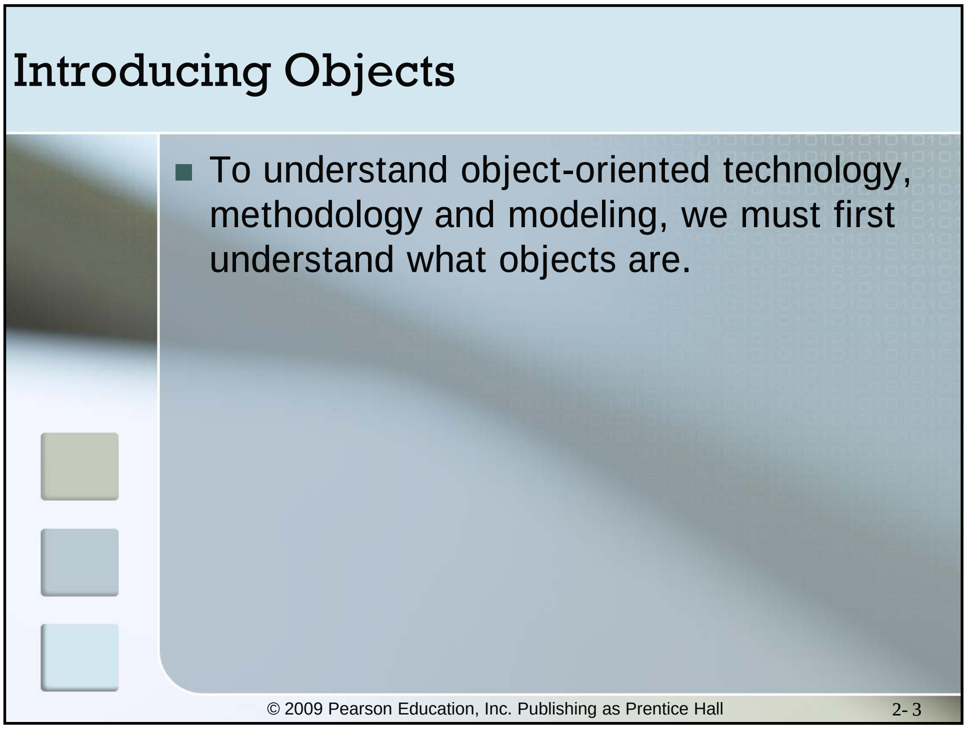
Chapter Two

The Concept of Object Orientation



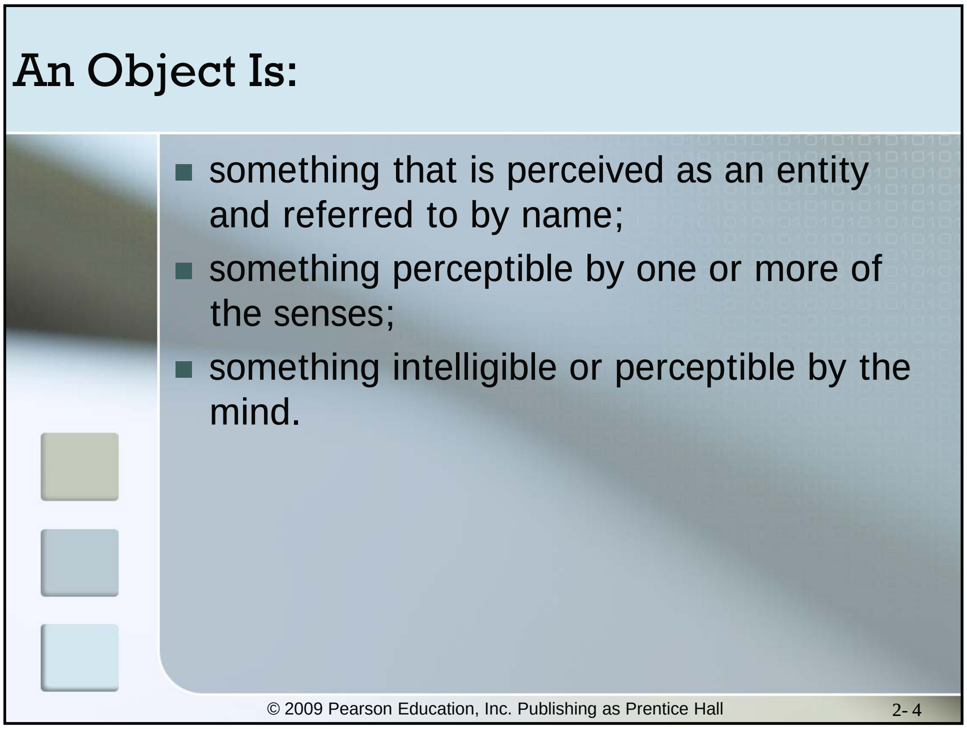
Chapter Topics

- Real-world objects.
- Object *identity*.
- Object’s attributes and operations.
- Classes and classification.
- Encapsulation and information hiding
- Object interface.
- Aggregate and composite objects.
- inheritance and polymorphism.
- Object-oriented technology.
- Object-oriented modeling and the Unified Modeling Language (UML).



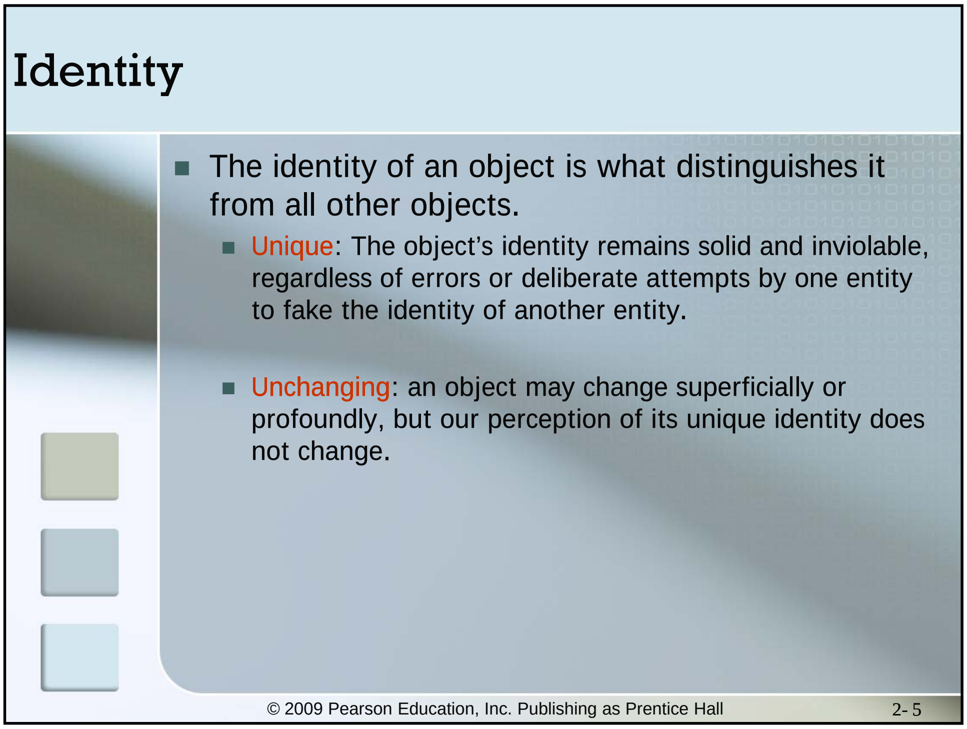
Introducing Objects

- To understand object-oriented technology, methodology and modeling, we must first understand what objects are.



An Object Is:

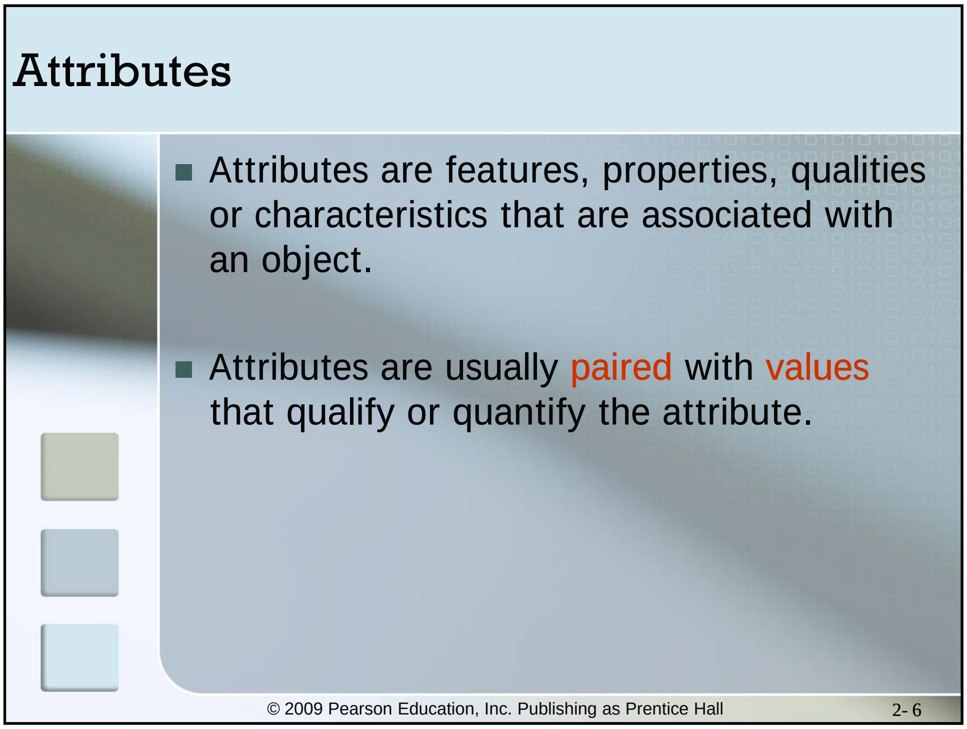
- something that is perceived as an entity and referred to by name;
- something perceptible by one or more of the senses;
- something intelligible or perceptible by the mind.



Identity

- The identity of an object is what distinguishes it from all other objects.
  - Unique: The object's identity remains solid and inviolable, regardless of errors or deliberate attempts by one entity to fake the identity of another entity.
  - Unchanging: an object may change superficially or profoundly, but our perception of its unique identity does not change.

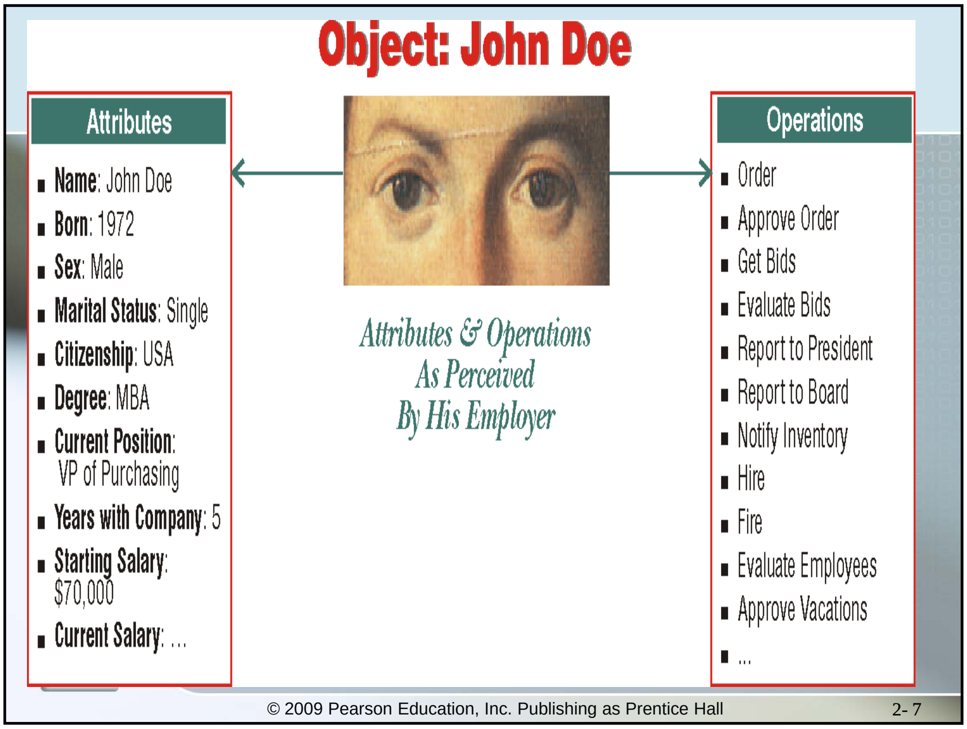
© 2009 Pearson Education, Inc. Publishing as Prentice Hall



Attributes

- Attributes are features, properties, qualities or characteristics that are associated with an object.
- Attributes are usually paired with values that qualify or quantify the attribute.

© 2009 Pearson Education, Inc. Publishing as Prentice Hall



Object: John Doe

Attributes

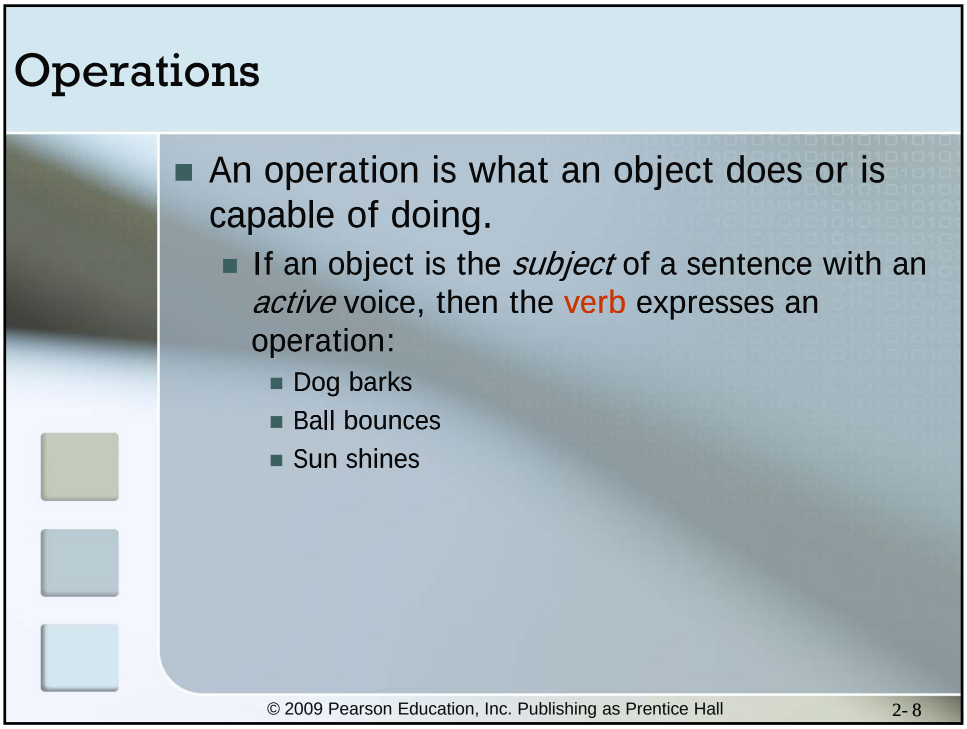
- **Name**: John Doe
- **Born**: 1972
- **Sex**: Male
- **Marital Status**: Single
- **Citizenship**: USA
- **Degree**: MBA
- **Current Position**: VP of Purchasing
- **Years with Company**: 5
- **Starting Salary**: $70,000
- **Current Salary**: …

Attributes & Operations
As Perceived
By His Employer

Operations

- Order
- Approve Order
- Get Bids
- Evaluate Bids
- Report to President
- Report to Board
- Notify Inventory
- Hire
- Fire
- Evaluate Employees
- Approve Vacations
- …

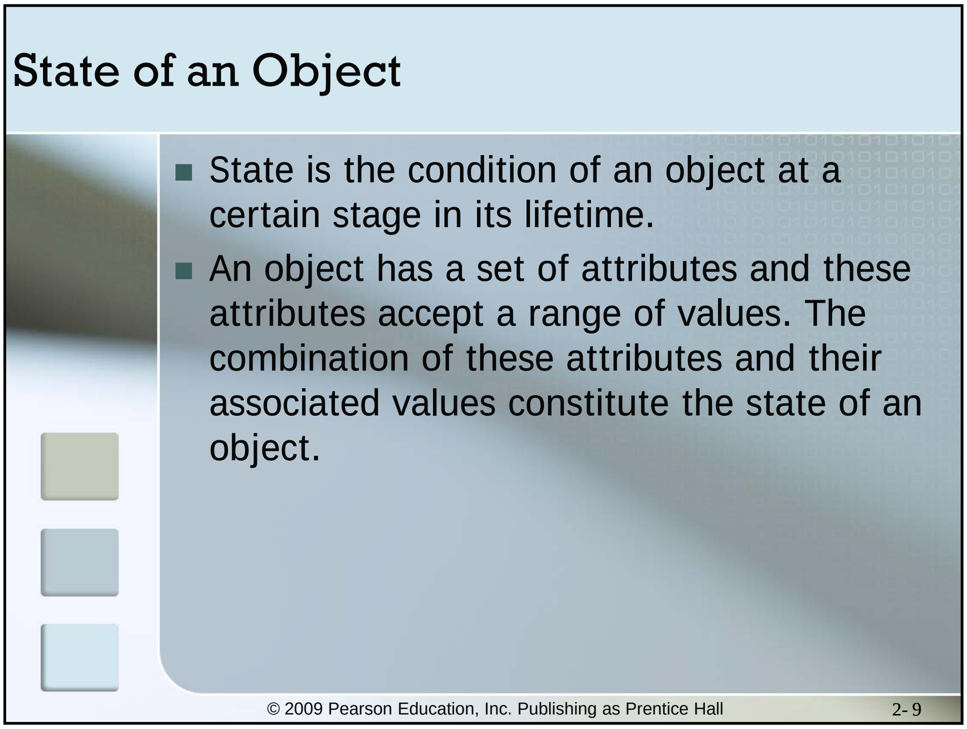
2- 7



Operations

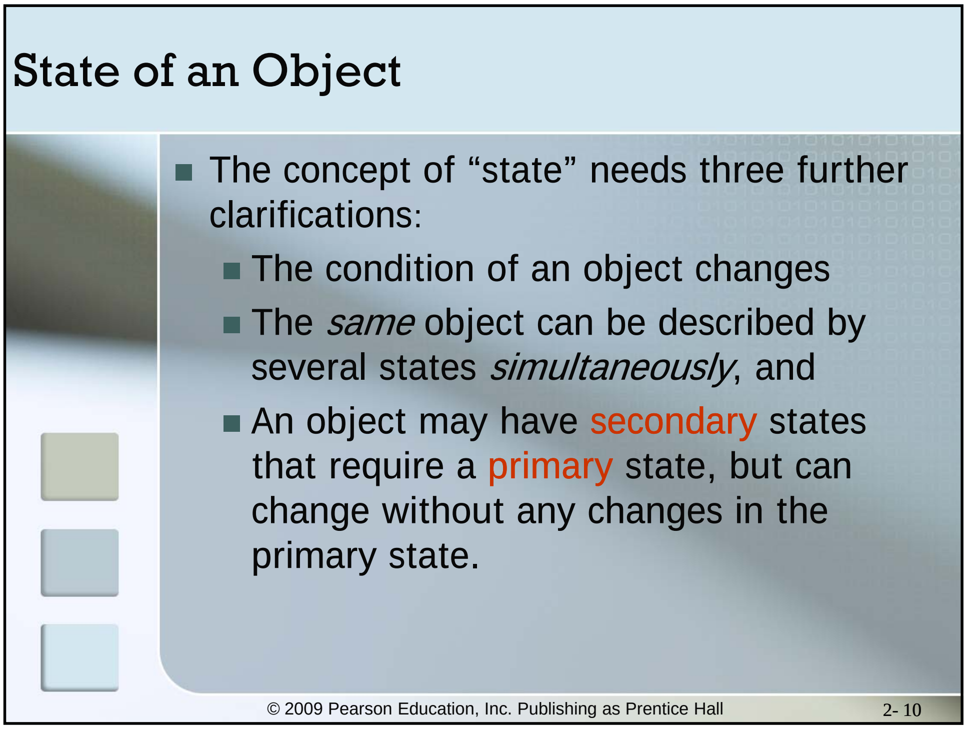
- An operation is what an object does or is capable of doing.
  - If an object is the *subject* of a sentence with an *active* voice, then the verb expresses an operation:
    - Dog barks
    - Ball bounces
    - Sun shines

2- 8



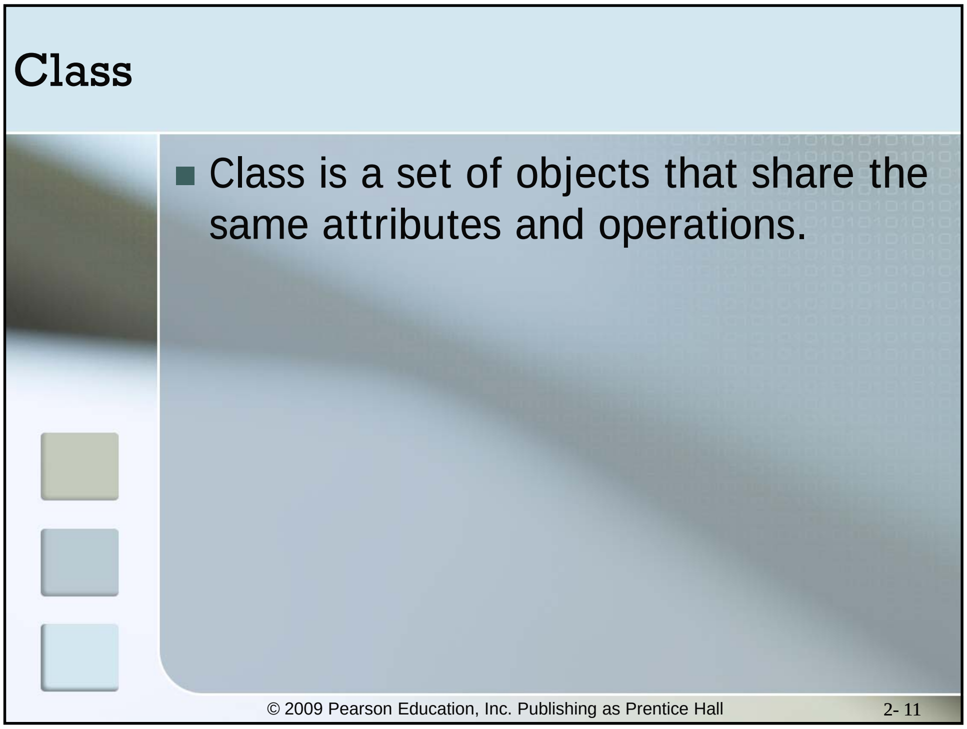
State of an Object

- State is the condition of an object at a certain stage in its lifetime.
- An object has a set of attributes and these attributes accept a range of values. The combination of these attributes and their associated values constitute the state of an object.



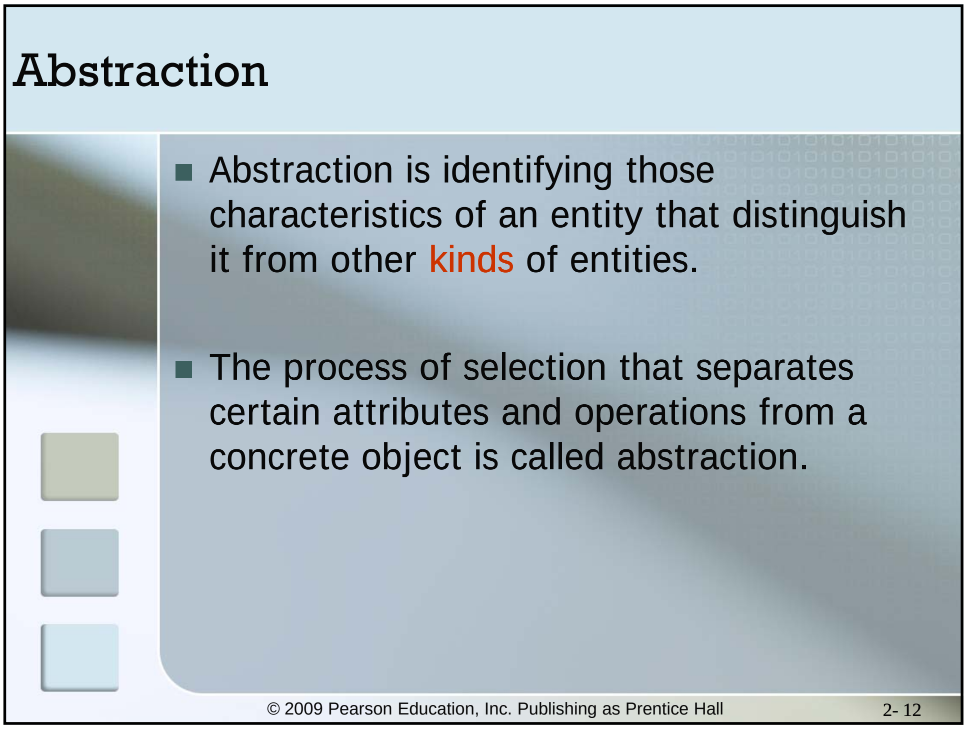
State of an Object

- The concept of “state” needs three further clarifications:
  - The condition of an object changes
  - The *same* object can be described by several states *simultaneously*, and
  - An object may have secondary states that require a primary state, but can change without any changes in the primary state.



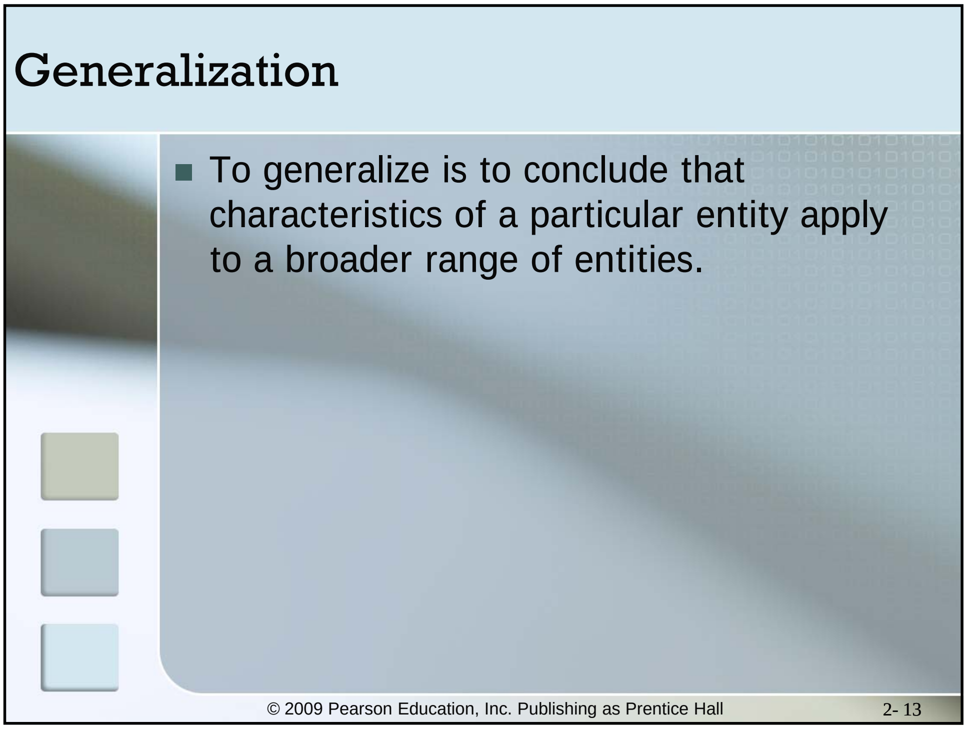
Class

- Class is a set of objects that share the same attributes and operations.



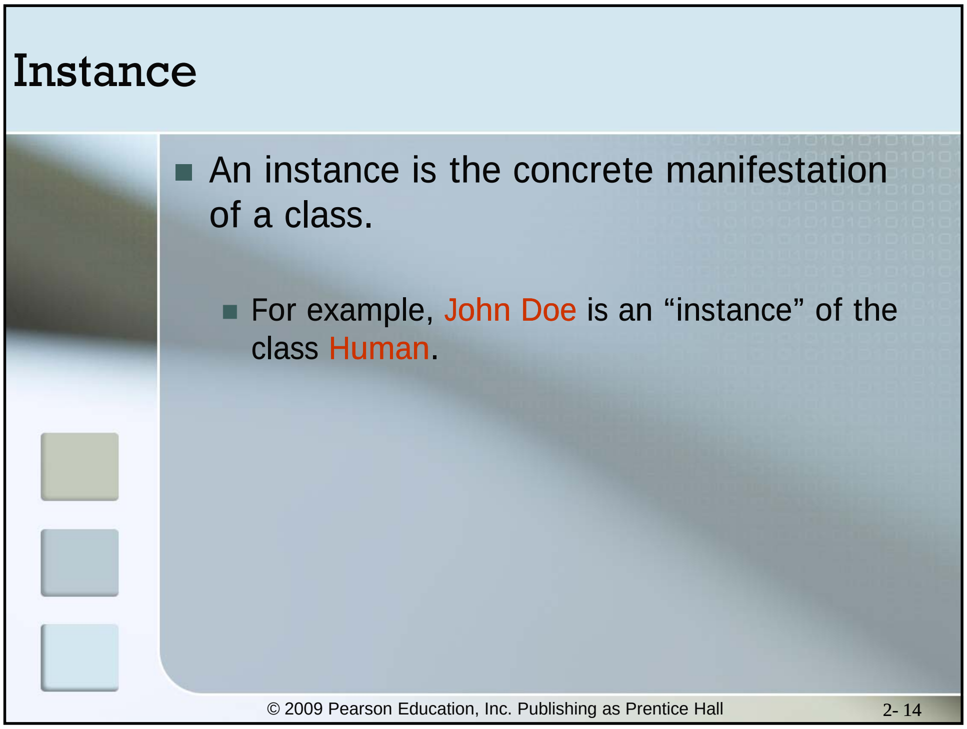
Abstraction

- Abstraction is identifying those characteristics of an entity that distinguish it from other kinds of entities.
- The process of selection that separates certain attributes and operations from a concrete object is called abstraction.



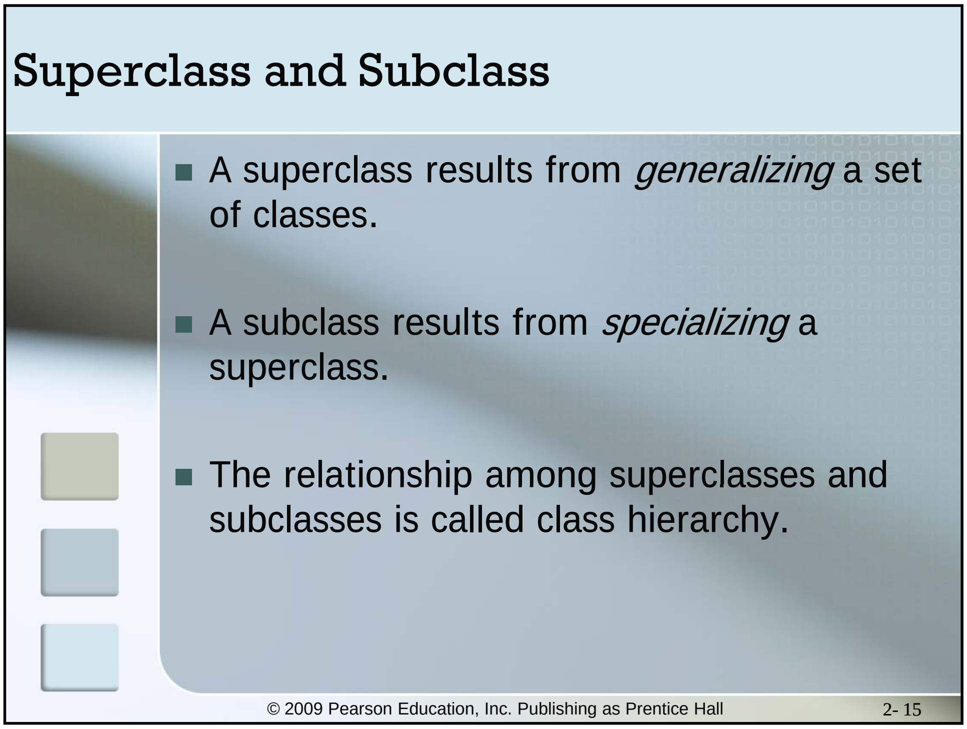
Generalization

- To generalize is to conclude that characteristics of a particular entity apply to a broader range of entities.



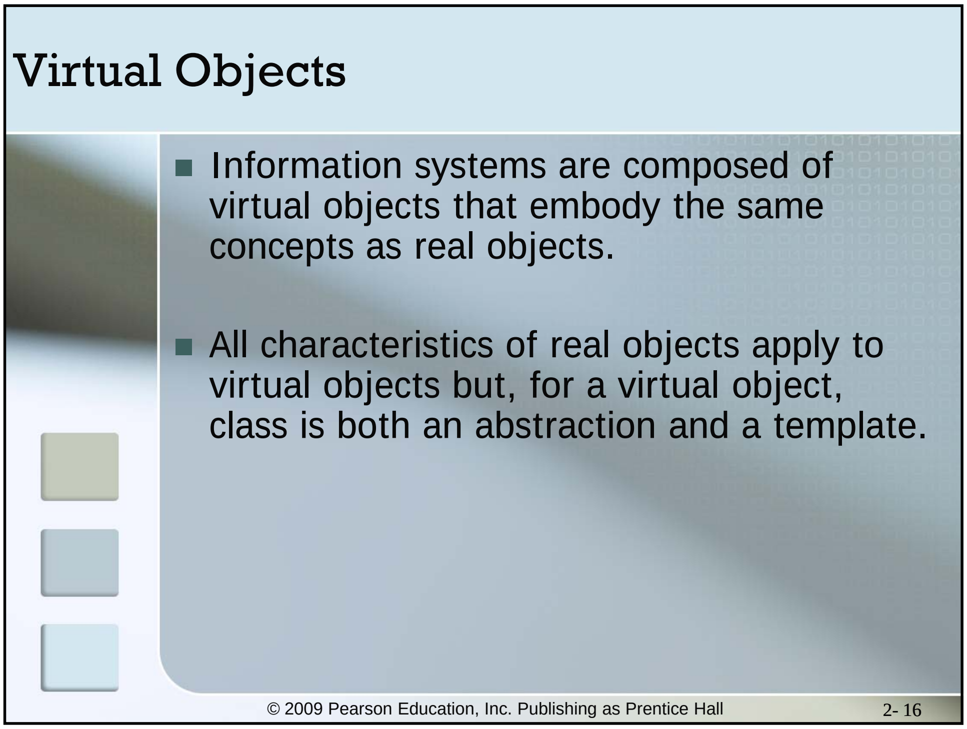
Instance

- An instance is the concrete manifestation of a class.
  - For example, John Doe is an “instance” of the class Human.



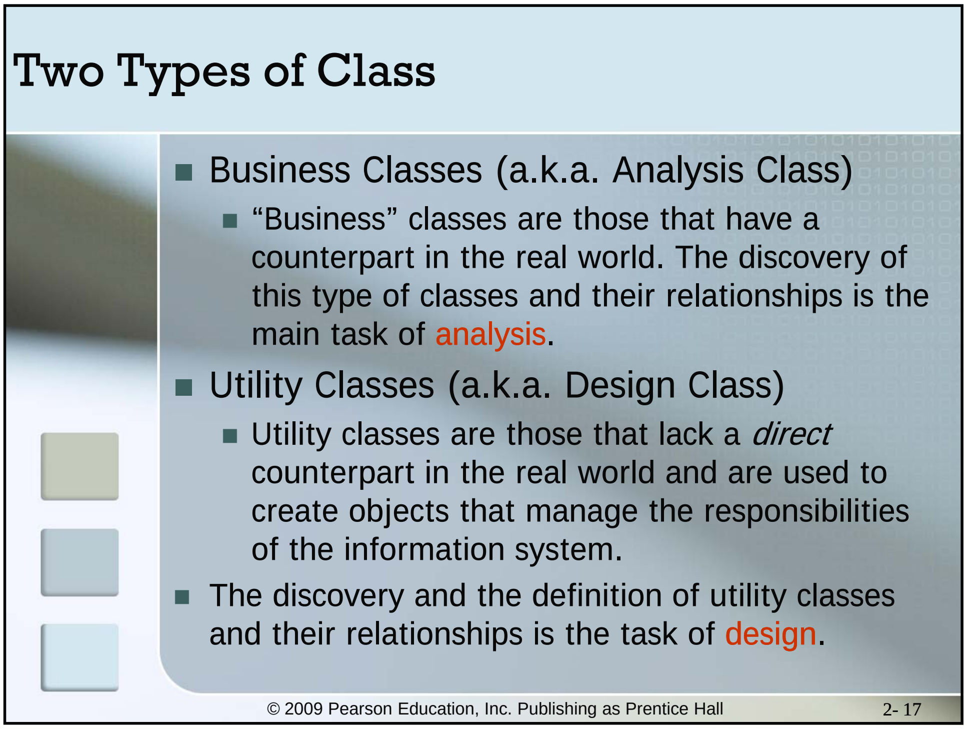
Superclass and Subclass

- A superclass results from *generalizing* a set of classes.
- A subclass results from *specializing* a superclass.
- The relationship among superclasses and subclasses is called class hierarchy.



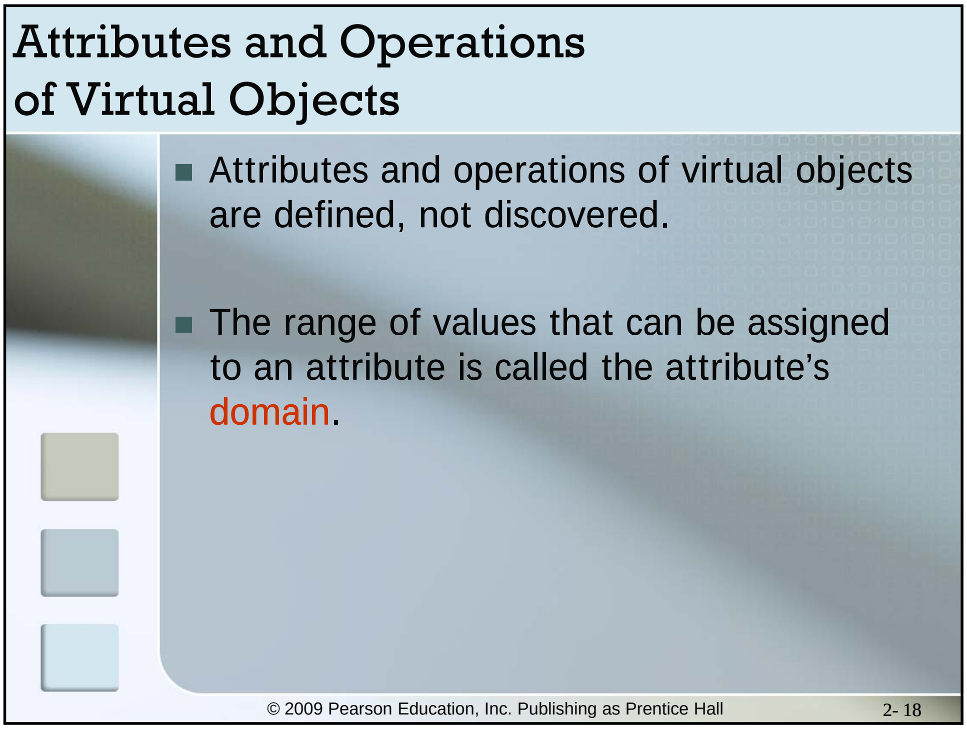
Virtual Objects

- Information systems are composed of virtual objects that embody the same concepts as real objects.
- All characteristics of real objects apply to virtual objects but, for a virtual object, class is both an abstraction and a template.



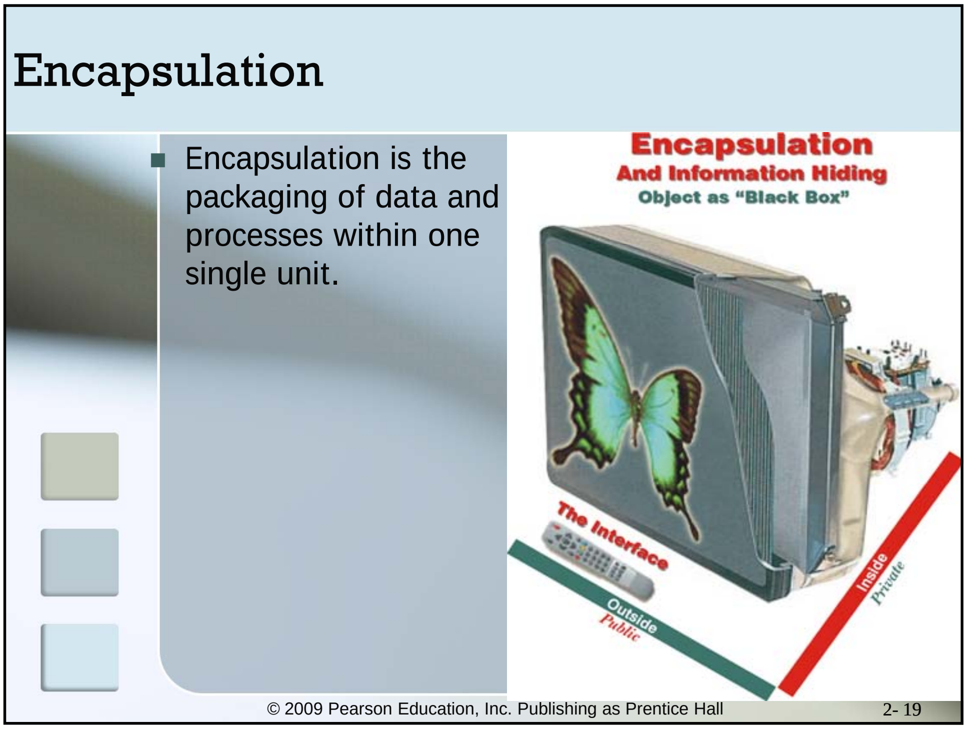
Two Types of Class

- Business Classes (a.k.a. Analysis Class)
  - “Business” classes are those that have a counterpart in the real world. The discovery of this type of classes and their relationships is the main task of analysis.
- Utility Classes (a.k.a. Design Class)
  - Utility classes are those that lack a *direct* counterpart in the real world and are used to create objects that manage the responsibilities of the information system.
- The discovery and the definition of utility classes and their relationships is the task of design.



Attributes and Operations of Virtual Objects

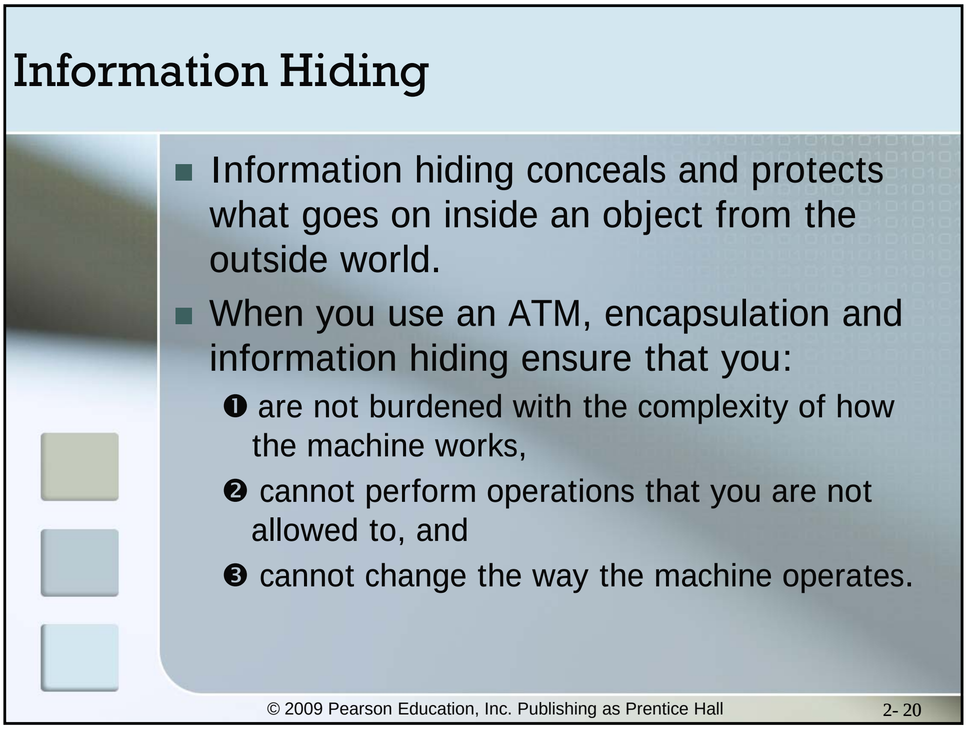
- Attributes and operations of virtual objects are defined, not discovered.
- The range of values that can be assigned to an attribute is called the attribute’s domain.



Encapsulation

- Encapsulation is the packaging of data and processes within one single unit.

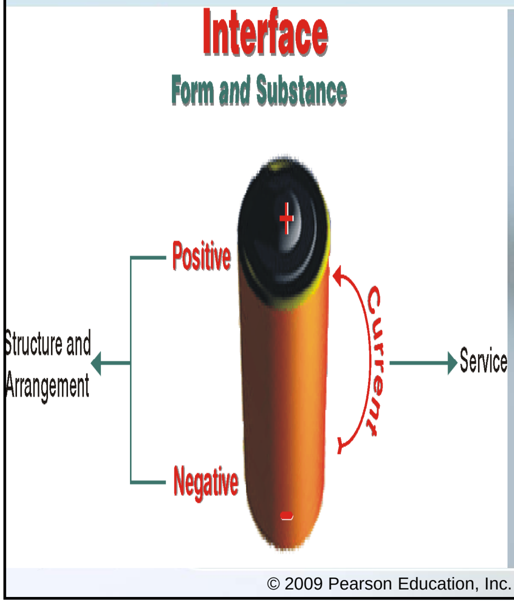
Encapsulation
And Information Hiding
Object as "Black Box"
The Interface
Outside
Public
Inside
Private



Information Hiding

- Information hiding conceals and protects what goes on inside an object from the outside world.
- When you use an ATM, encapsulation and information hiding ensure that you:
  - ❶ are not burdened with the complexity of how the machine works,
  - ❷ cannot perform operations that you are not allowed to, and
  - ❸ cannot change the way the machine operates.

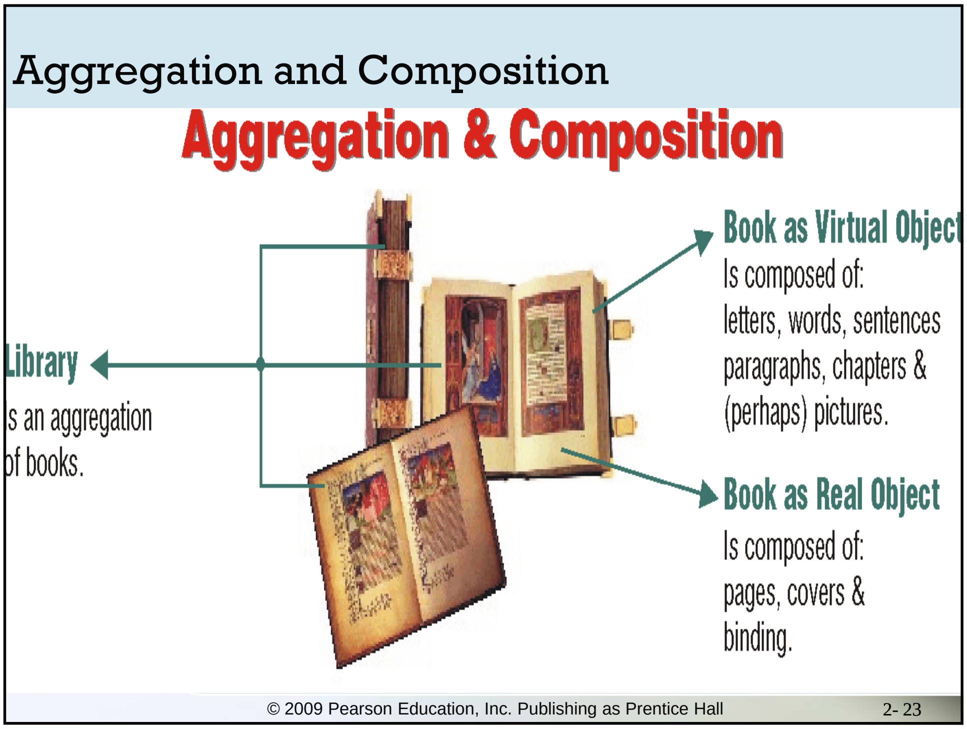
Interface



- An object’s interface consists of operations that are available to the public.

Aggregation and Composition

- In object-oriented terminology, the relationship of one object to its component objects is called aggregation.
- A strong form of aggregation in which the life of components relies on the life of the whole is called composition.



Aggregation and Composition

Aggregation & Composition

Library

s an aggregation
of books.

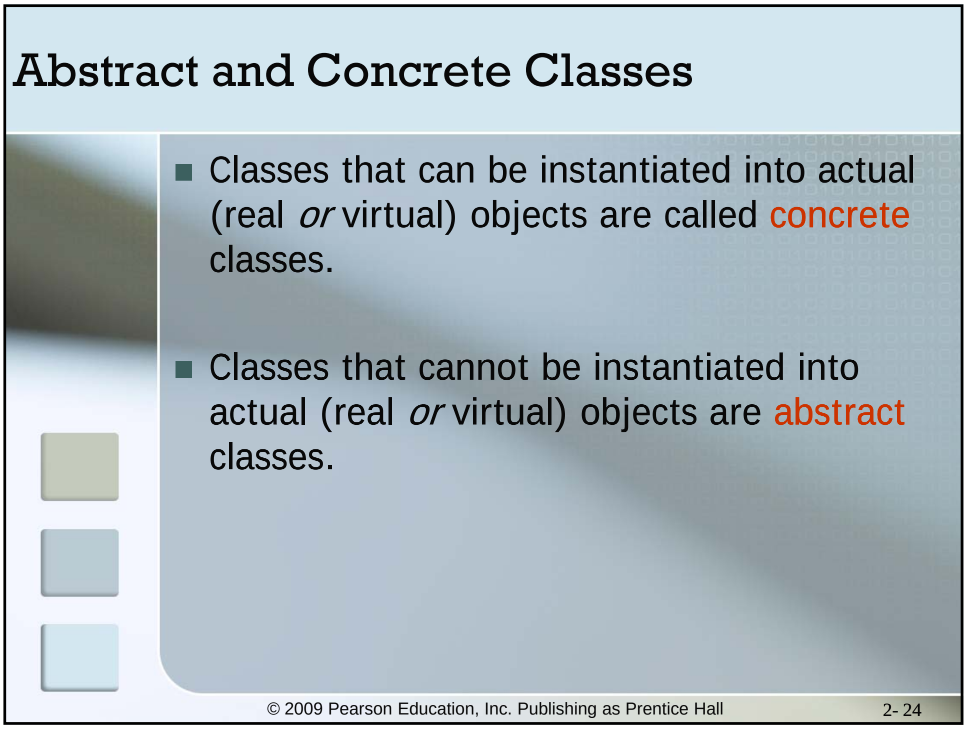
Book as Virtual Object

Is composed of:
letters, words, sentences
paragraphs, chapters &
(perhaps) pictures.

Book as Real Object

Is composed of:
pages, covers &
binding.

© 2009 Pearson Education, Inc. Publishing as Prentice Hall 2- 23

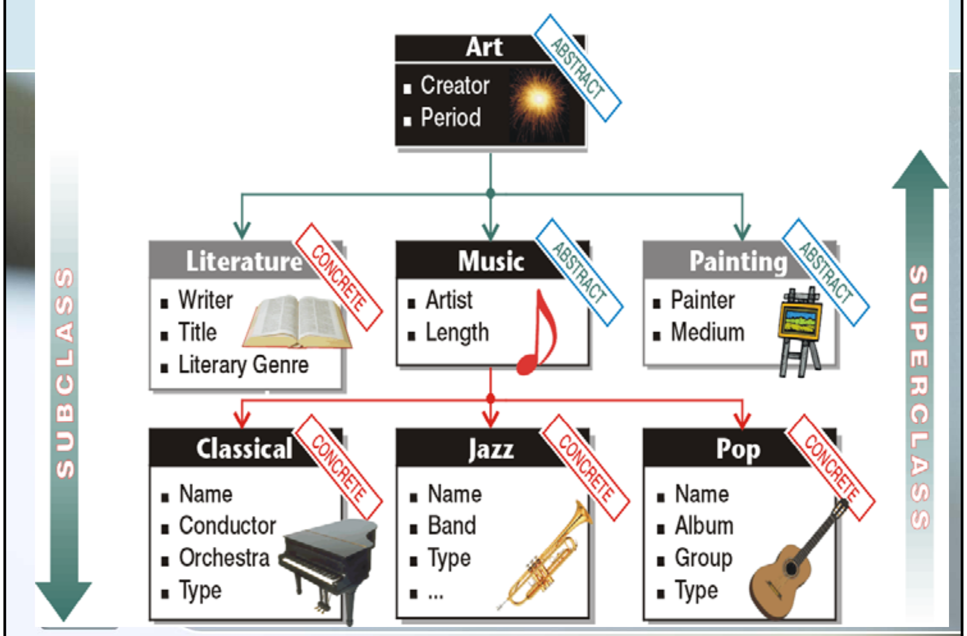


Abstract and Concrete Classes

- Classes that can be instantiated into actual (real *or* virtual) objects are called concrete classes.
- Classes that cannot be instantiated into actual (real *or* virtual) objects are abstract classes.

© 2009 Pearson Education, Inc. Publishing as Prentice Hall 2- 24

Abstract & Concrete Classes



Art ABSTRACT
- Creator
- Period

Literature CONCRETE
- Writer
- Title
- Literary Genre

Music ABSTRACT
- Artist
- Length

Painting ABSTRACT
- Painter
- Medium

Classical CONCRETE
- Name
- Conductor
- Orchestra
- Type

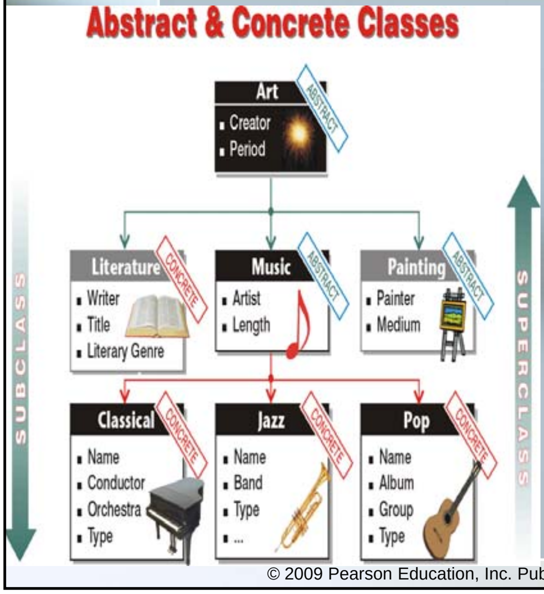
Jazz CONCRETE
- Name
- Band
- Type
- ...

Pop CONCRETE
- Name
- Album
- Group
- Type

SUBCLASS

SUPERCLASS

Inheritance



Abstract & Concrete Classes

Art ABSTRACT
- Creator
- Period

Literature CONCRETE
- Writer
- Title
- Literary Genre

Music ABSTRACT
- Artist
- Length

Painting ABSTRACT
- Painter
- Medium

Classical CONCRETE
- Name
- Conductor
- Orchestra
- Type

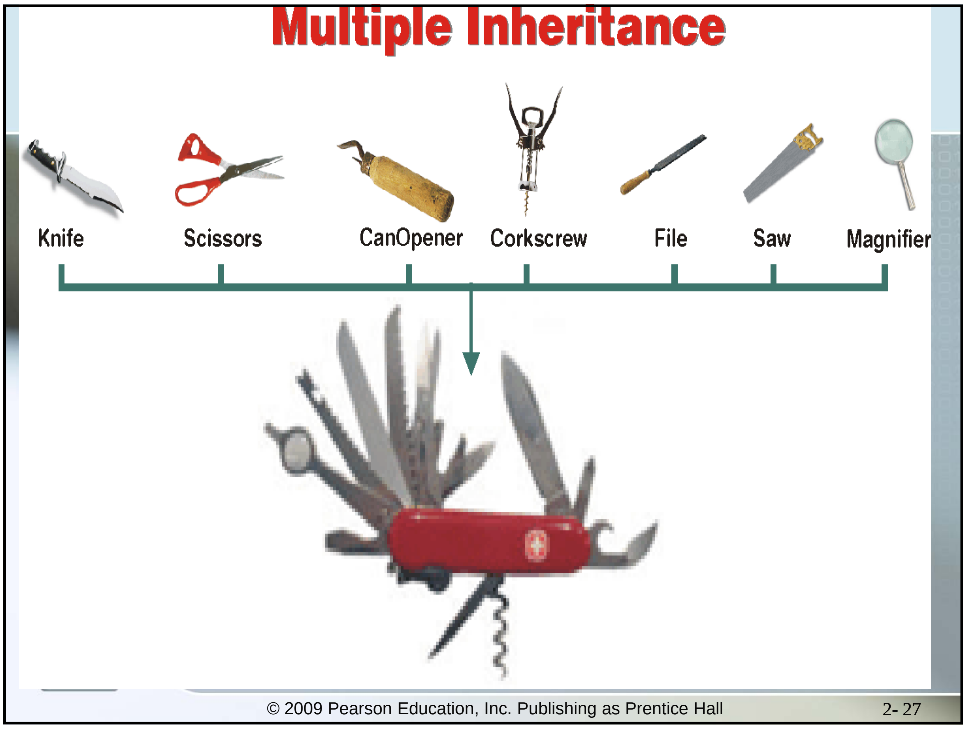
Jazz CONCRETE
- Name
- Band
- Type
- ...

Pop CONCRETE
- Name
- Album
- Group
- Type

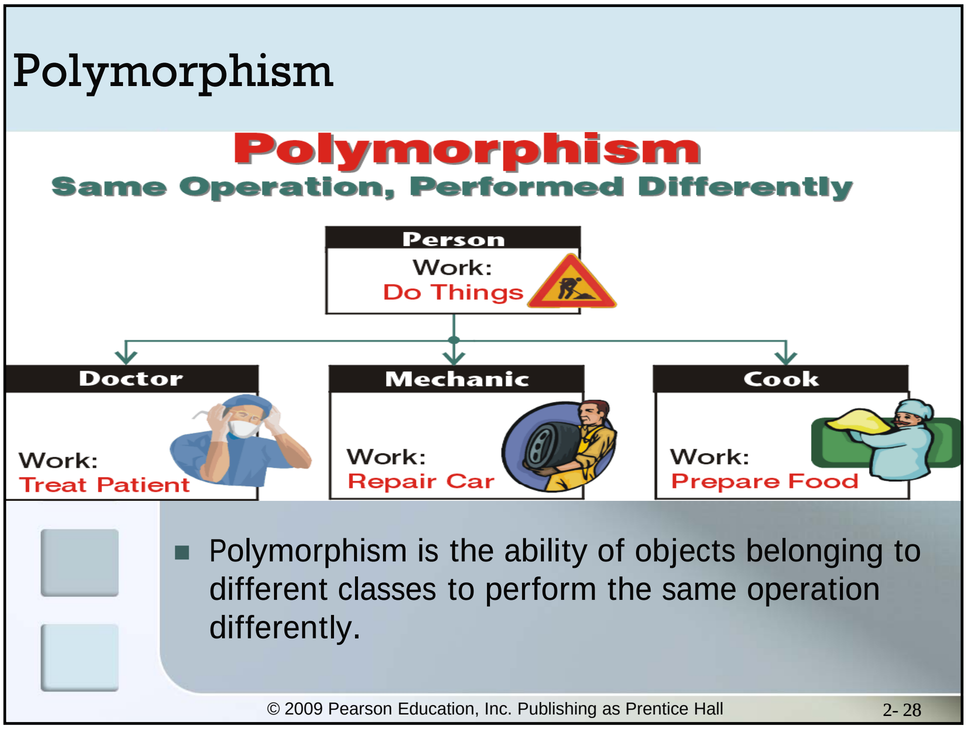
SUBCLASS

SUPERCLASS

- Inheritance is the mechanism by which a subclass incorporates the behavior of a superclass.



Multiple Inheritance

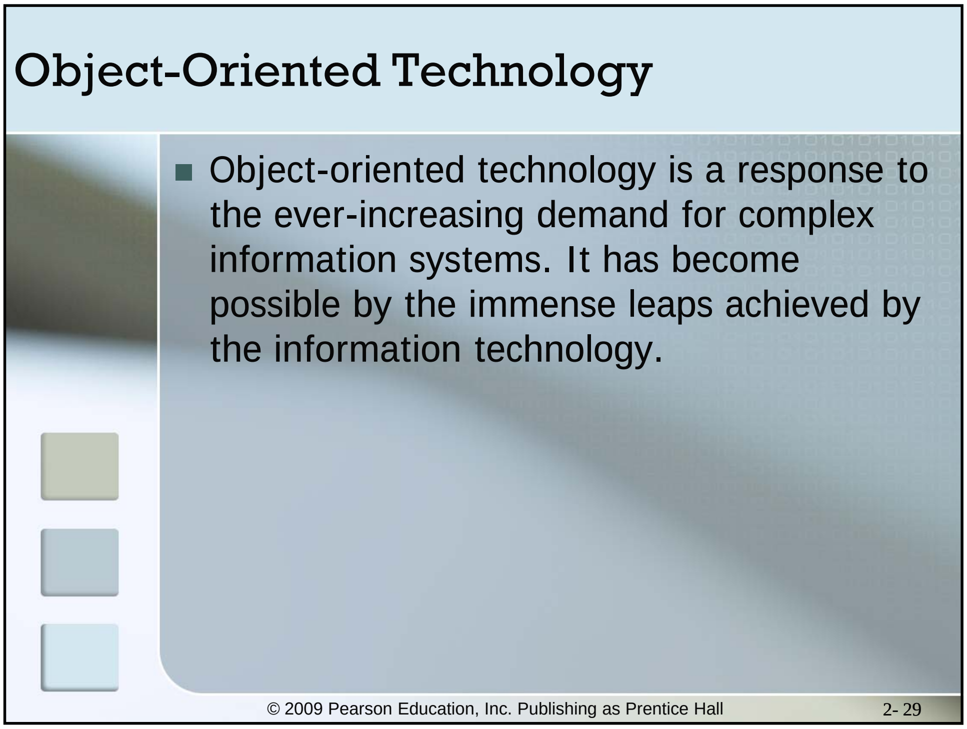


Polymorphism

Polymorphism

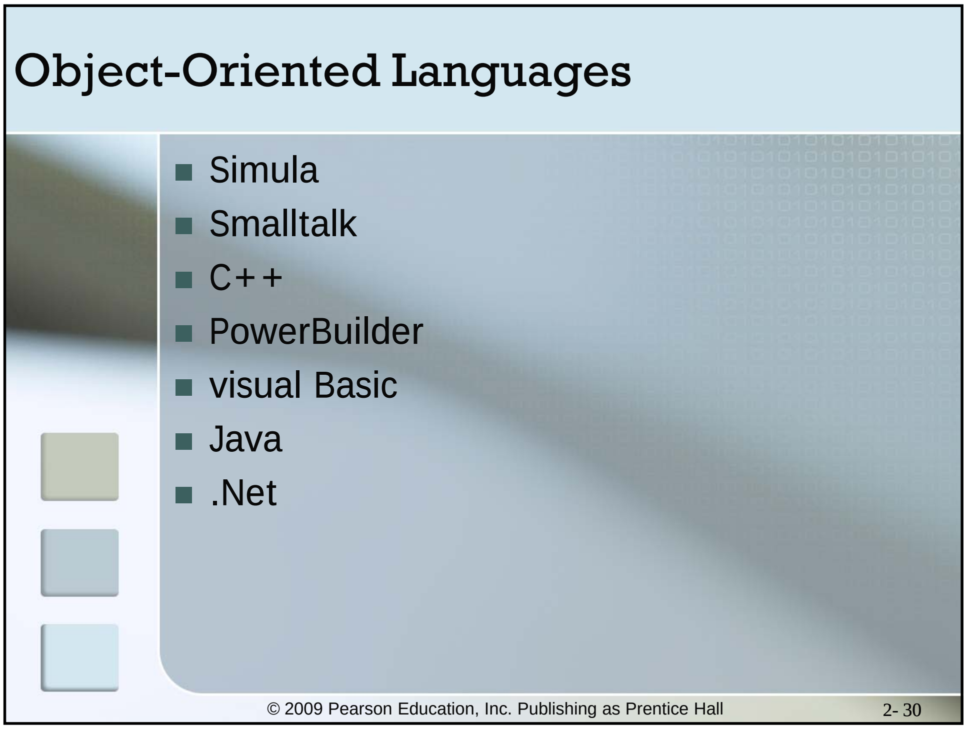
Same Operation, Performed Differently

- Polymorphism is the ability of objects belonging to different classes to perform the same operation differently.



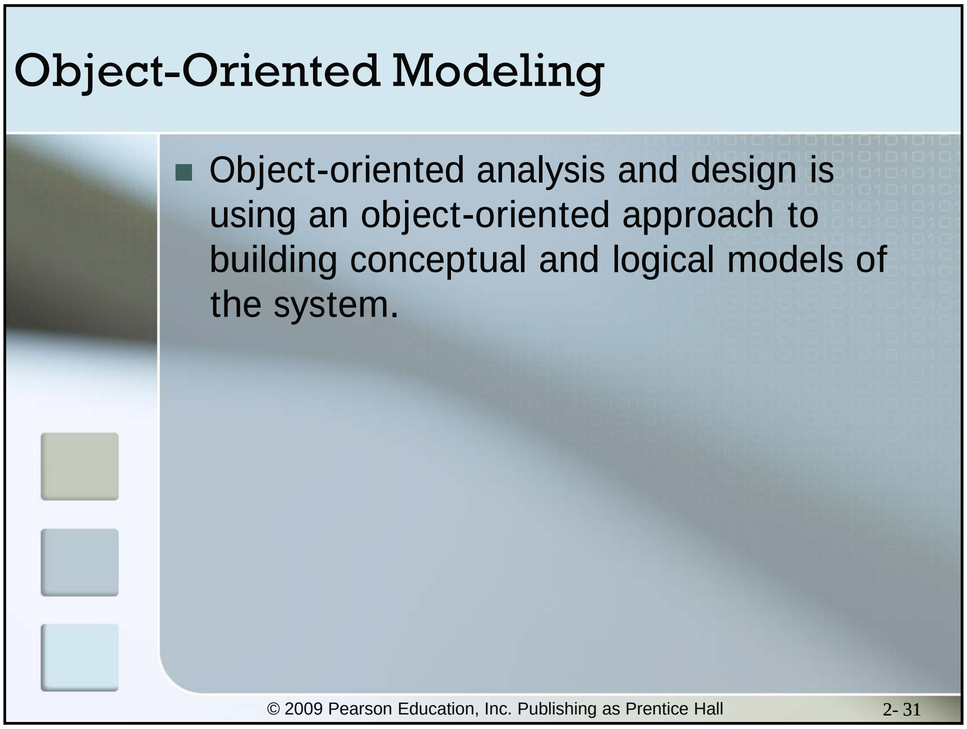
Object-Oriented Technology

- Object-oriented technology is a response to the ever-increasing demand for complex information systems. It has become possible by the immense leaps achieved by the information technology.



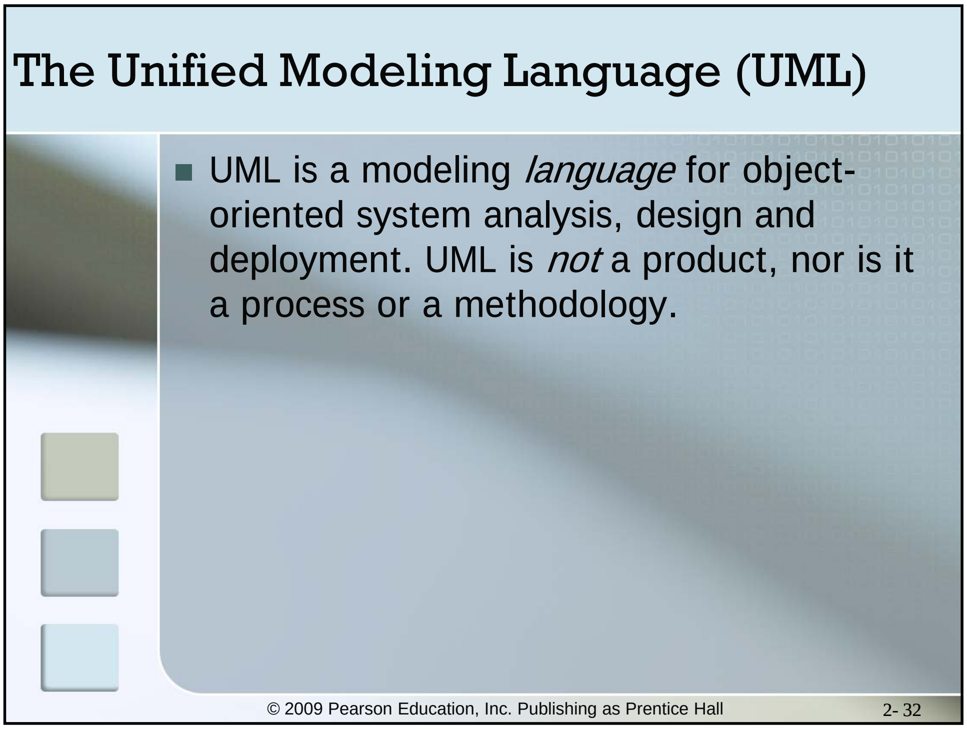
Object-Oriented Languages

- Simula
- Smalltalk
- C++
- PowerBuilder
- visual Basic
- Java
- .Net



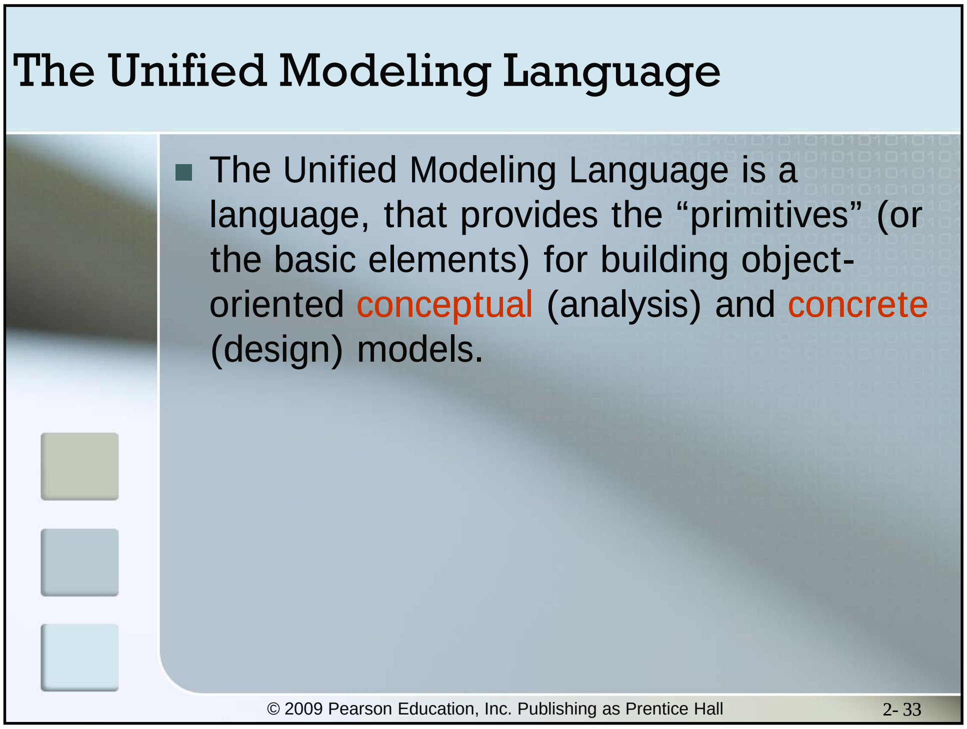
Object-Oriented Modeling

- Object-oriented analysis and design is using an object-oriented approach to building conceptual and logical models of the system.



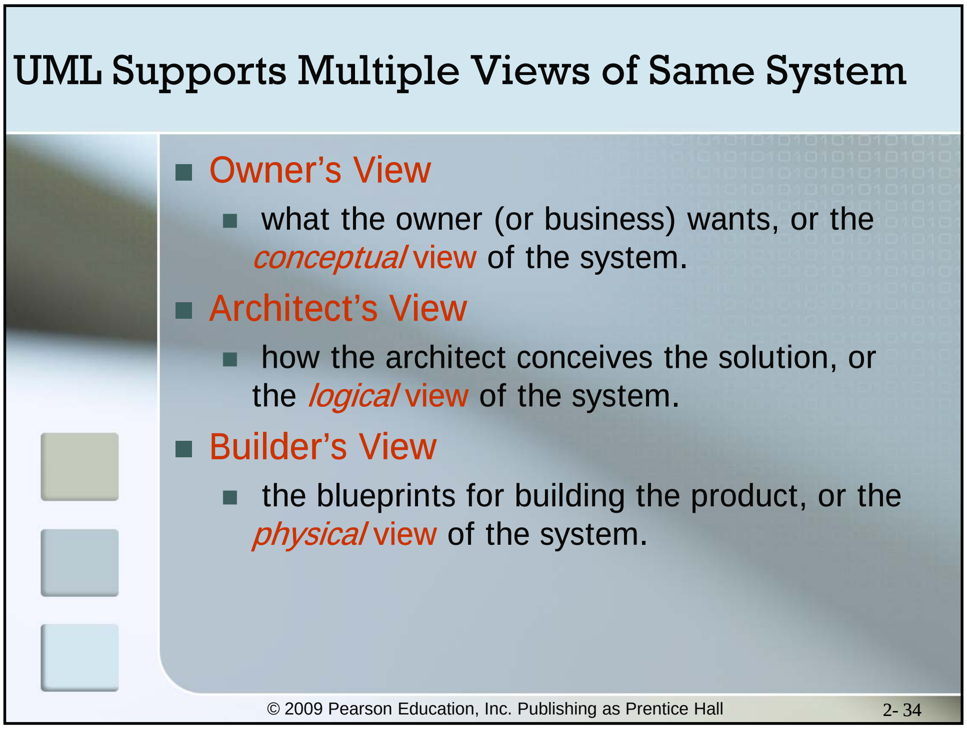
The Unified Modeling Language (UML)

- UML is a modeling *language* for object-oriented system analysis, design and deployment. UML is *not* a product, nor is it a process or a methodology.



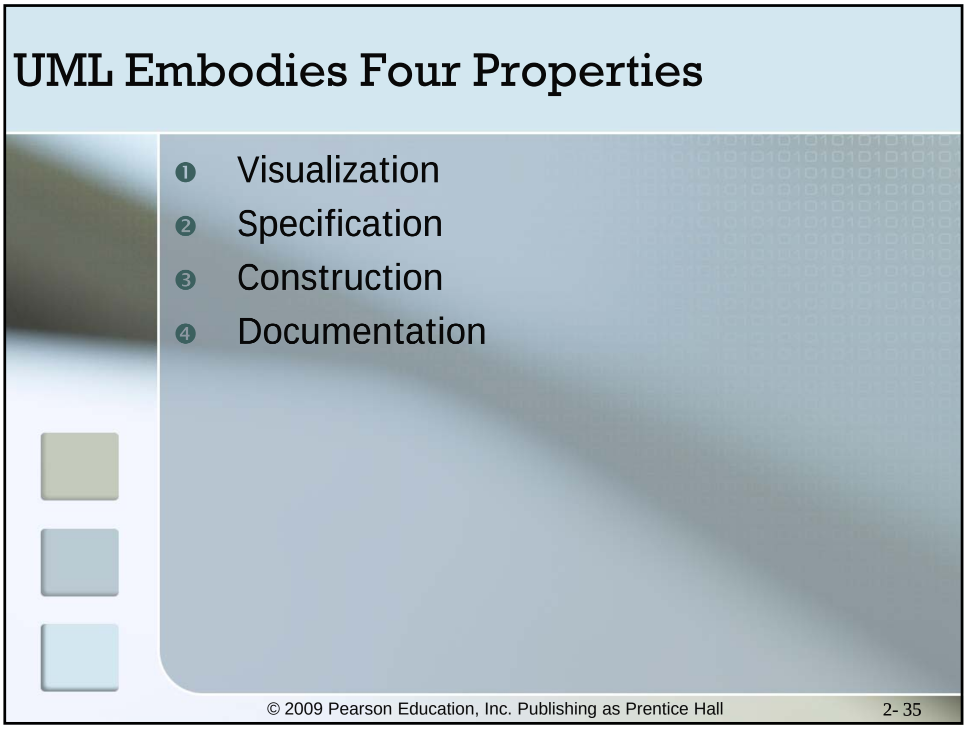
The Unified Modeling Language

- The Unified Modeling Language is a language, that provides the “primitives” (or the basic elements) for building object-oriented conceptual (analysis) and concrete (design) models.



UML Supports Multiple Views of Same System

- Owner’s View
  - what the owner (or business) wants, or the *conceptual* view of the system.
- Architect’s View
  - how the architect conceives the solution, or the *logical* view of the system.
- Builder’s View
  - the blueprints for building the product, or the *physical* view of the system.



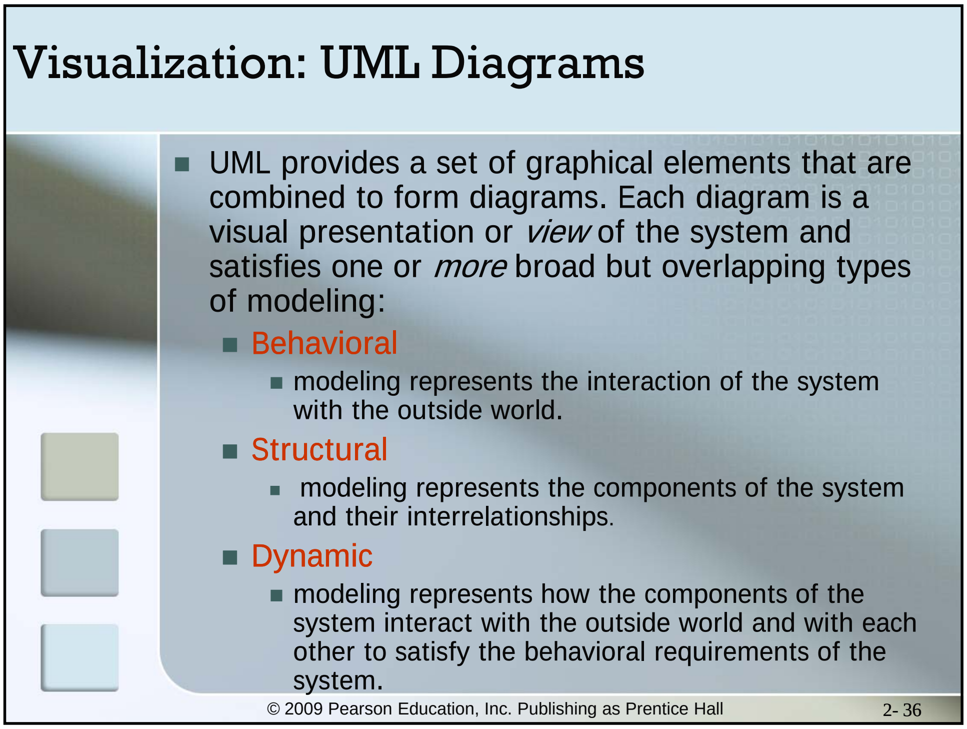
UML Embodies Four Properties

❶ Visualization

❷ Specification

❸ Construction

❹ Documentation

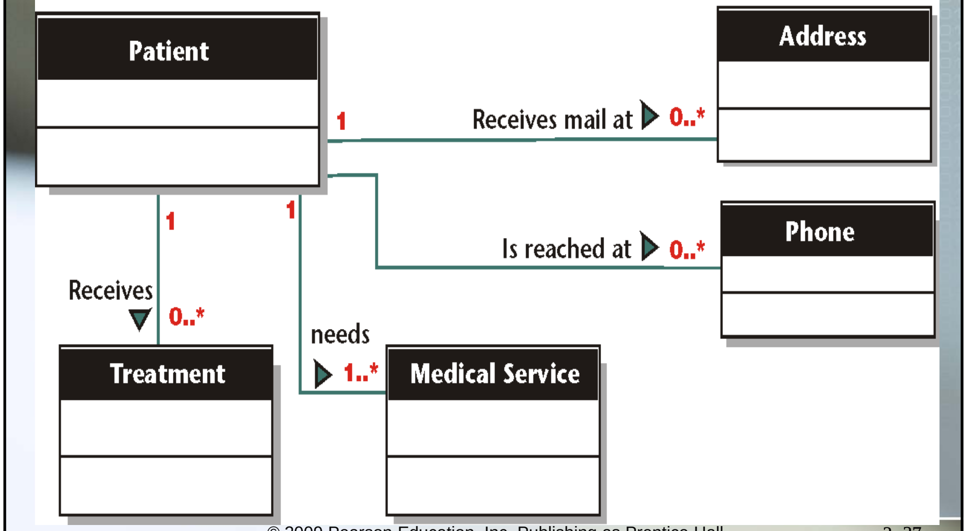


Visualization: UML Diagrams

- UML provides a set of graphical elements that are combined to form diagrams. Each diagram is a visual presentation or *view* of the system and satisfies one or *more* broad but overlapping types of modeling:
  - Behavioral
    - modeling represents the interaction of the system with the outside world.
  - Structural
    - modeling represents the components of the system and their interrelationships.
  - Dynamic
    - modeling represents how the components of the system interact with the outside world and with each other to satisfy the behavioral requirements of the system.

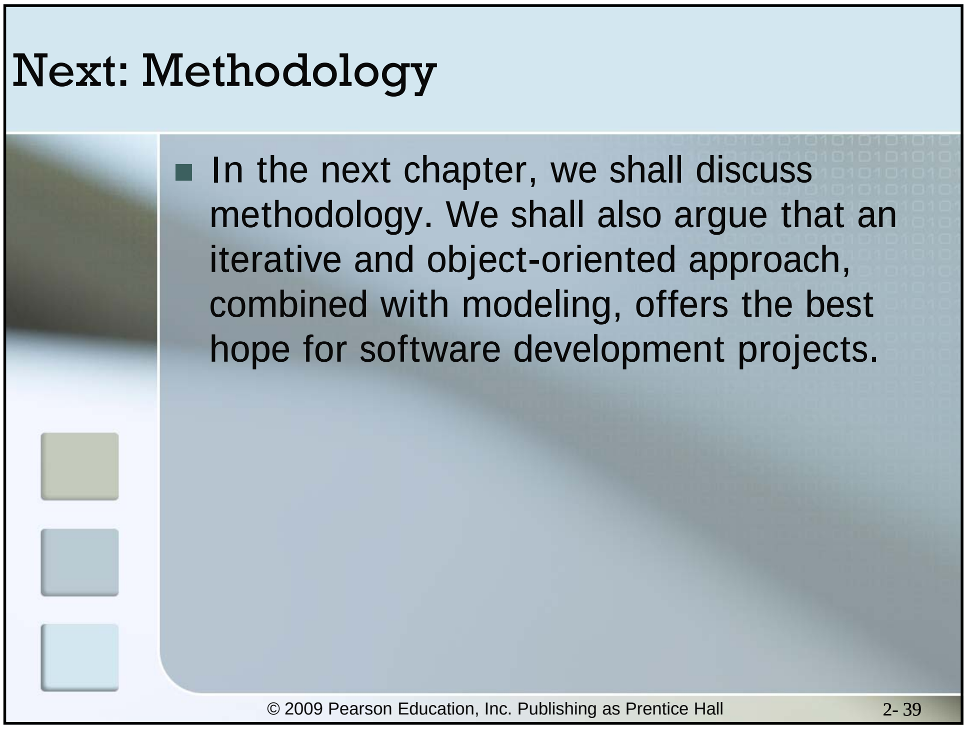
A UML Diagram

Modeling Object-Oriented Concepts



Specification, Construction, and Documentation

- Specification
  - UML provides precise and complete models for the three major activities of system development: analysis, design, and implementation.
- Construction
  - UML models are compatible with object-oriented languages.
- Documentation
  - UML modeling tracks major development activities throughout the system lifecycle.



Next: Methodology

- In the next chapter, we shall discuss methodology. We shall also argue that an iterative and object-oriented approach, combined with modeling, offers the best hope for software development projects.