Object-Oriented Technology - Classes

Xin Feng

Outline

- Object and class
- Class notation
- Class features

Object-Oriented Development

- A popular software development method
- Develop reusable systems
- The concept started in 1968
- Based on the use of single object

Objects

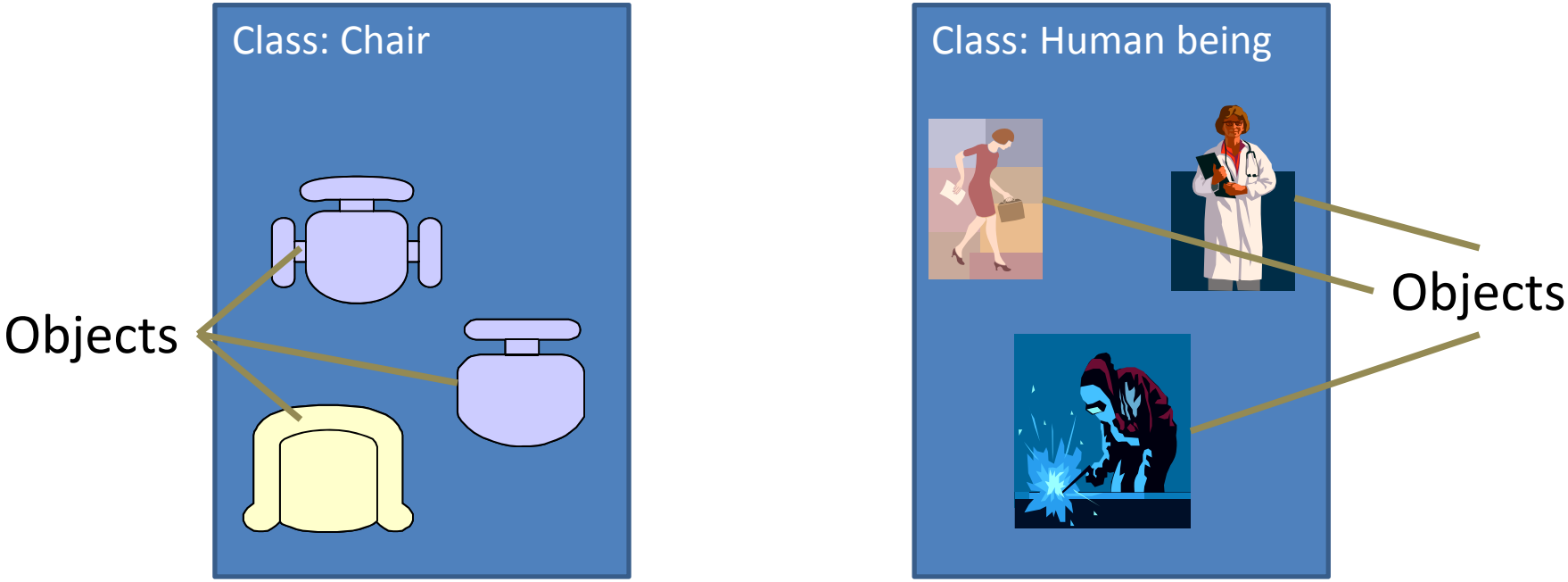
- Object
  - An entity (实体)
  - Physical: a chair, a desk, a person
  - Logical: a list, a stack, a rectangle
- Features
  - Can be described, e.g. shape, length
  - Can be created and destroyed

Classes

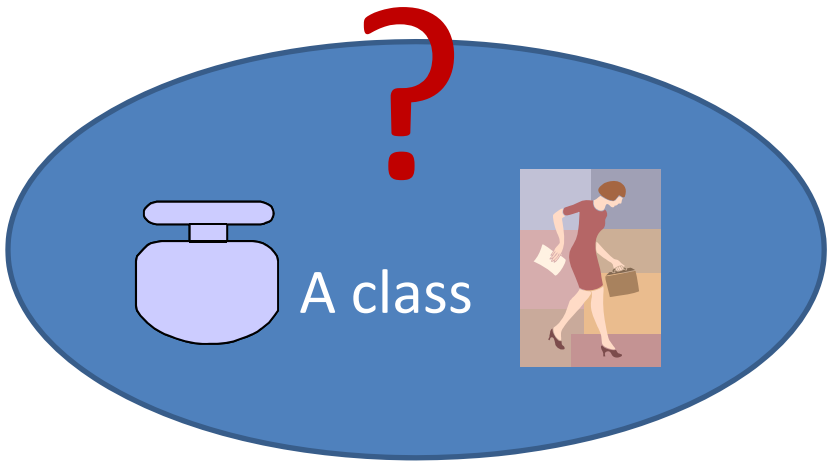
- Normal meaning
  - A set of entities with similar attributes
- OO
  - A description that encapsulates （封装）the data and procedural abstractions (抽象）that are required to describe the content and behavior of some real world entities
  - A generalized （概括的）description that describes a collection of similar objects.
  - Example: chair, human being

Objects and Classes

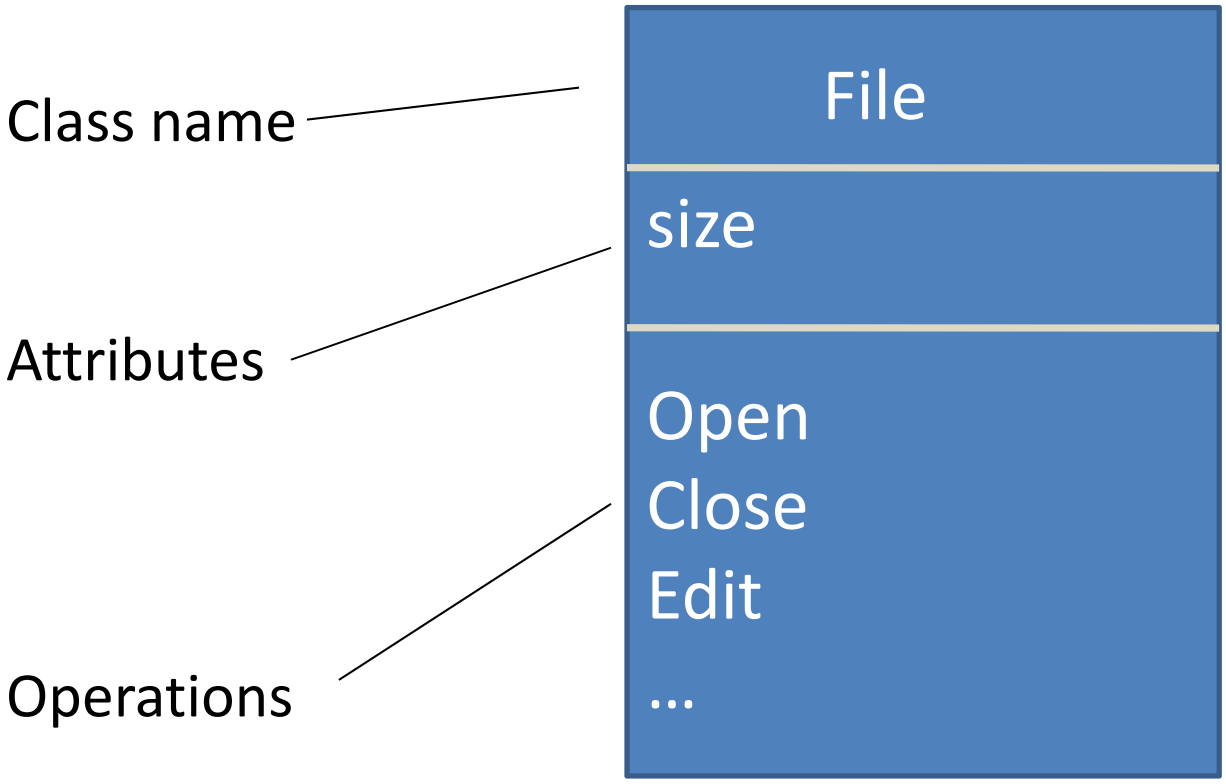
- An object is an instance（实例） of a class



Objects and Classes



Class Representation



Features of A Class

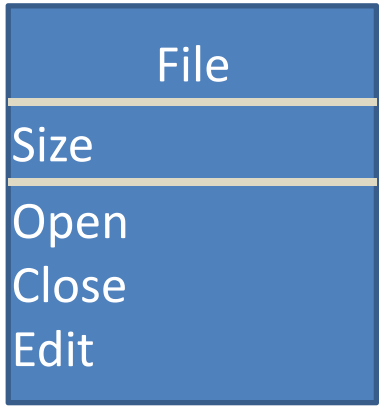
- Four features
  - Abstraction (抽象)
  - Encapsulation （封装）
  - Inheritance （继承）
  - Polymorphism （多态性）

Abstraction

- Focusing on the essential aspects (attributes under consideration) of an entity and ignoring its accidental (非主要的) properties.

Abstraction

- Make some decision as late as possible
  - E,g, file type
- Avoid to make decisions on the design and implementation decisions before the problems are understood
  - E.g., how to open, close, edit is not the consideration at this moment.



Encapsulation (Information Hiding)

- The attributes and operations of a class are hidden from outside world
- Separate the external aspects of an object, which are accessible to other objects.
  - For the modules outside this class, they only know the aspect shown to them. E.g., how to open a file is not beyond their consideration.

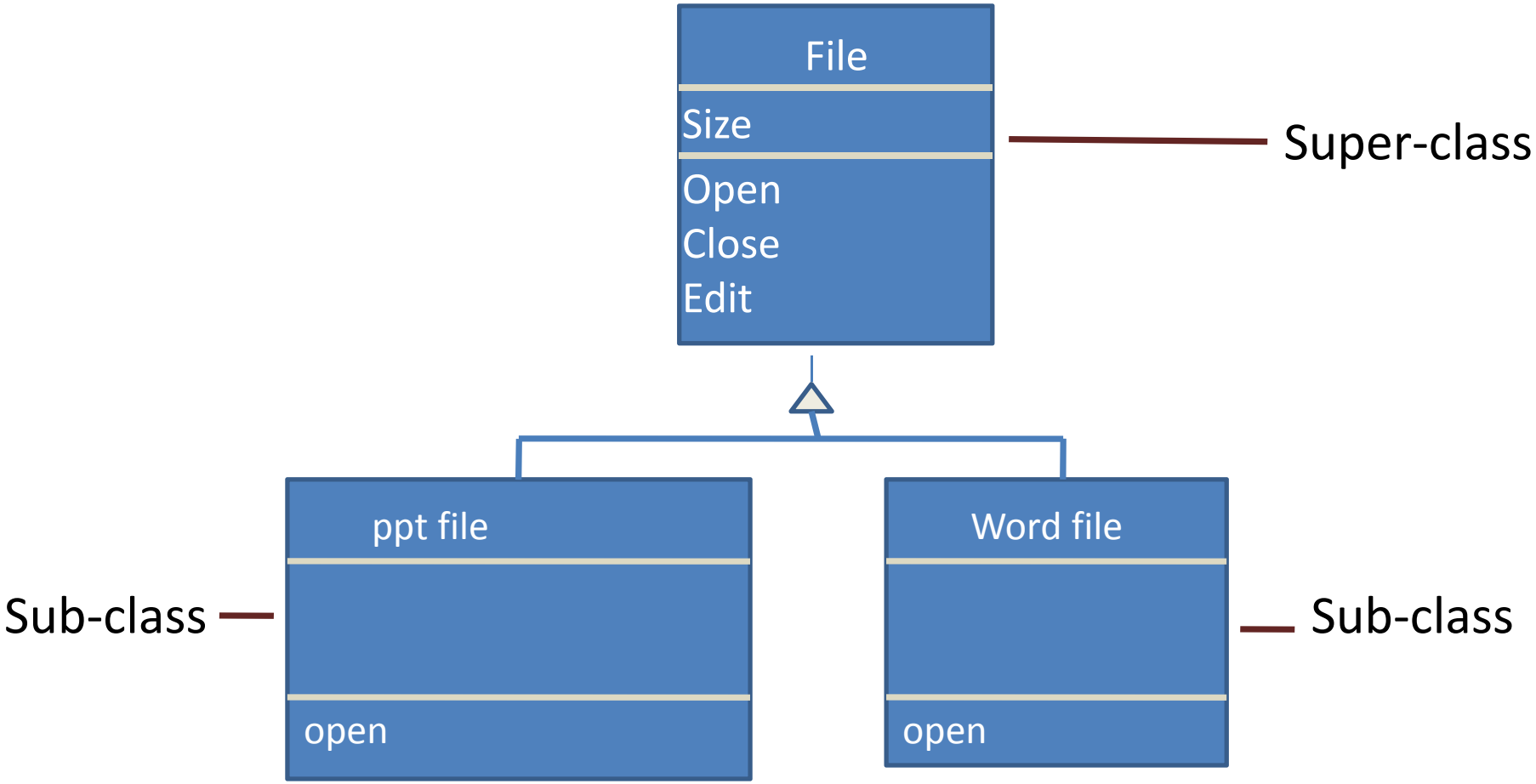
Encapsulation

- Reduce the affection（影响）of the changes
- Facilitate （使容易）component reuse
- Simplify （简化） the interfaces

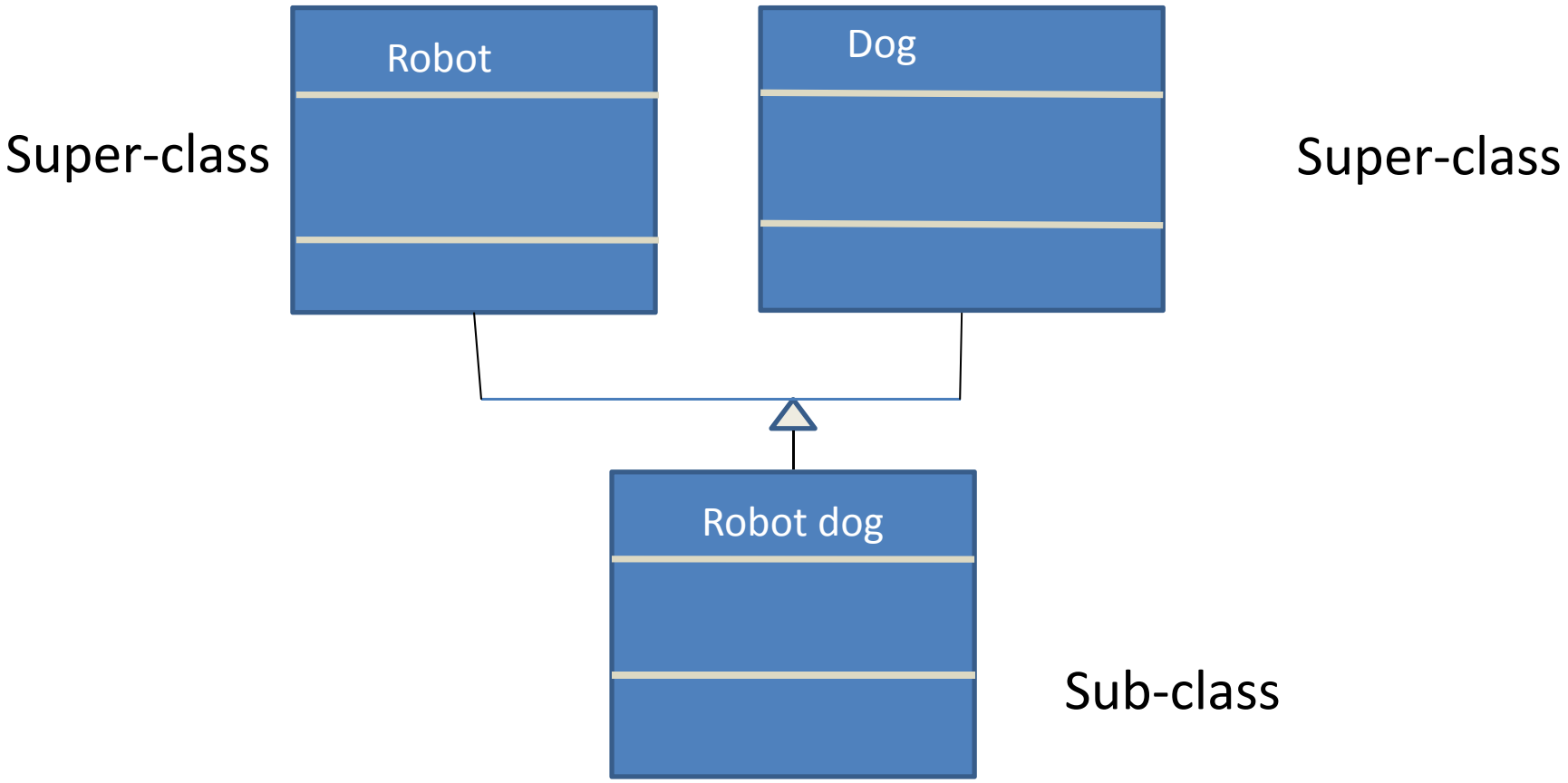
Inheritance

- Super-class and sub-class
  - Sub-class include the attributes in the super-class with some added or specialized （特有）features
- It is the key （关键） feature of OO
- The implementation in the super-class can be reused
- The common changes to all the sub-classes can be implemented in the super-class
- A sub-class can have several super-classes.

Inheritance



Multiple Inheritance



Polymorphism (多态性)

- Reduce the cost that is needed to extend the system
- Allow a set of different implementations (semantic meanings）-similar operations to have the same name.
  - E.g.1
    - `insert(IMAGE oImage);`
    - `insert(char *text);`
  - E.g.2
    - `add(int i1, int i2);`
    - `add(char *str1, char *str2);`

What Is “Object-Oriented”

Object-oriented

Object + Classification + Inheritance + Communication

About OO

OO in one sentence:

keep it DRY, keep it shy and tell the other guy

DRY: Do not Repeat Yourself

Shy: Should not reveal the information about itself unless really necessary

Tell the other guy: Send a message rather than a function call.

- By Andy Hunt and Dave Thomas.

Class Exercise

- Figure out the classes and relations from the following description
  - UIC is developing a system for the campus people. Each has a mobile no. Each student has info like student ID, dorm no, major, home address. Each teacher has a office number, staff ID, office phone number. A teacher can be a mentor of students. For a mentor, we need to know the email address. too; For a security guard, the staff ID is needed. A teacher can teach several courses in a semester. Each course has course ID, name.

Summary

- A class has class name, attributes and behaviors
- Class has four features: abstraction, encapsulation, inheritance, and polymorphism