



中南大學
CENTRAL SOUTH UNIVERSITY



面向对象程序设计 (Object Oriented Programing)

第七章 统一建模语言 (UML) —— 类图


张宝一

地理信息系 / Department of Geo-Informatics

zhangbaoyi@csu.edu.cn



UML类图

- UML类图 
- 类的表示方式
- 类的关系表示方式
- 课外作业

1、UML类图

- **UML**：统一建模语言（**Unified Modeling Language, UML**）是用来**设计软件的可视化建模语言**。它的特点是简单、统一、图形化、能表达软件设计中的动态与静态信息。
- **UML 包含**：
 - 用例图
 - **类图 (Class diagram)**
 - 对象图、状态图、活动图、时序图
 - 协作图、构件图、部署图 等 9 种图。

1、UML类图

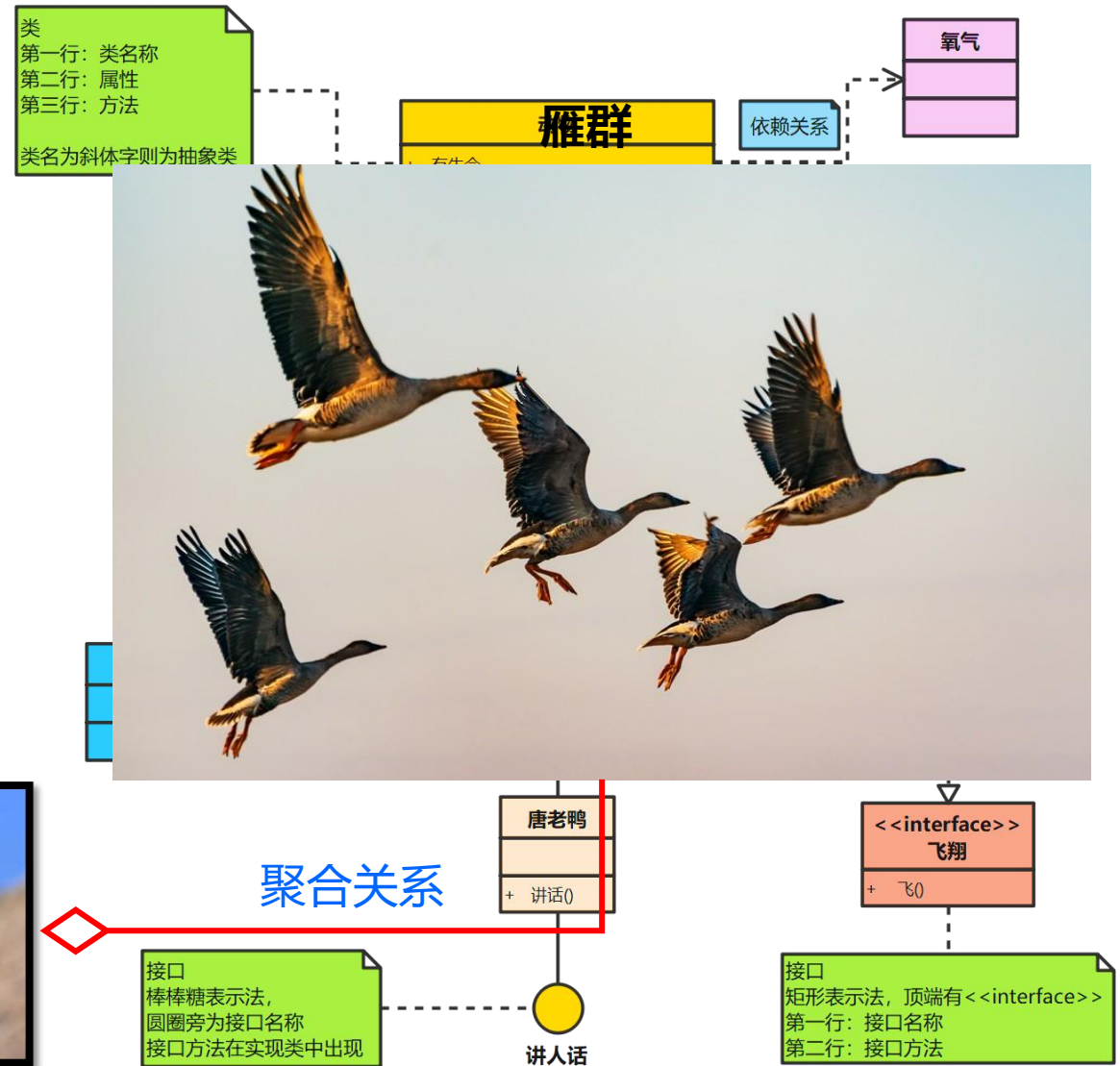
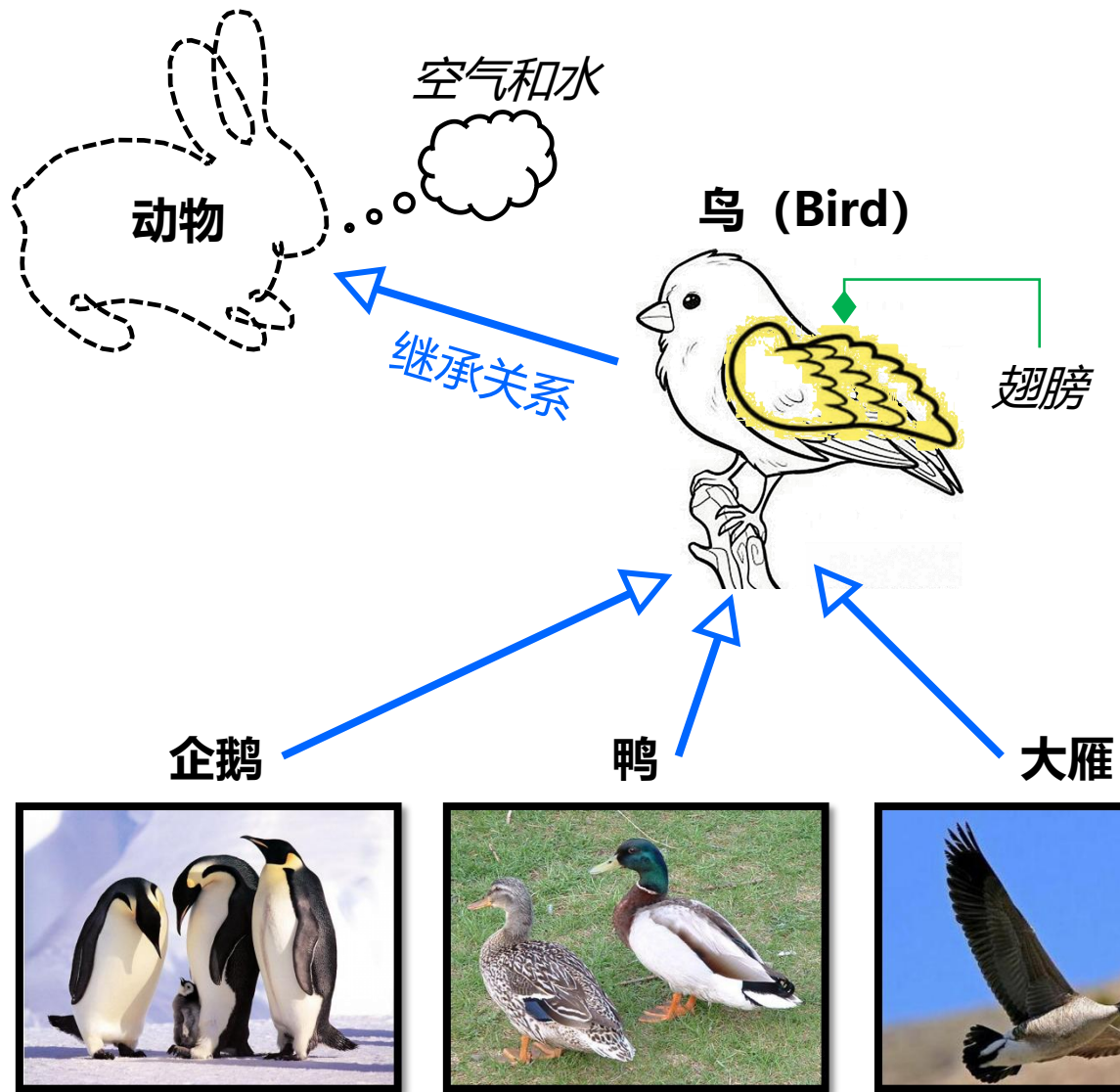
□ 类图的概述

类图(Class diagram)是面向对象系统建模中最常用和最重要的图，是定义其它图的基础。类图主要是用来**显示系统中的类、接口以及它们之间的静态结构和关系**的一种静态模型。类图中**最基本的元素是类、接口**。软件设计师设计出类图后，程序员就可以用代码实现类图中包含的内容。

□ 类图的作用

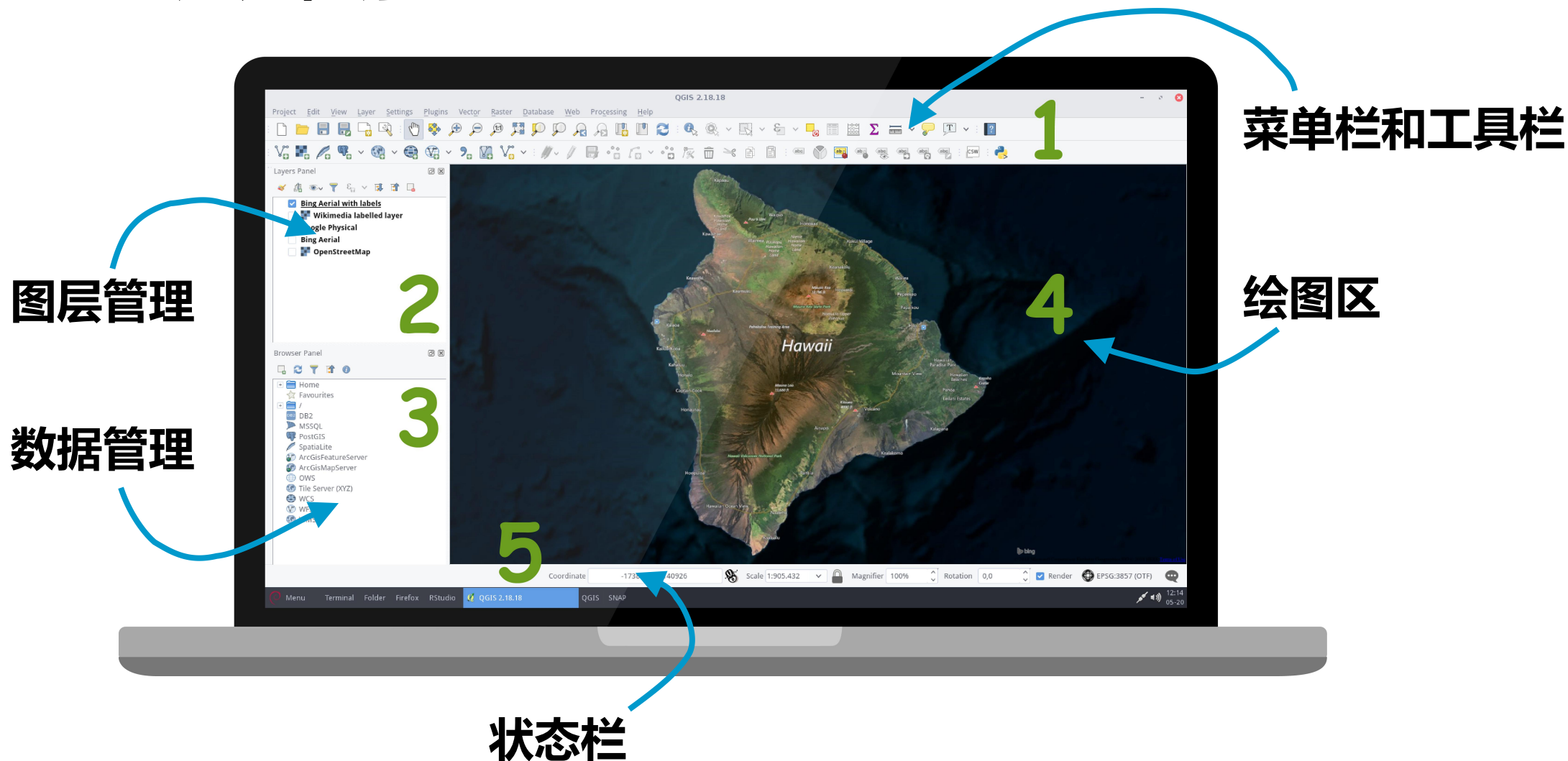
- 在软件工程中，类图描述了系统的结构、类的属性和类之间的关系，有助于人们对系统的理解。
- 类图是系统分析和设计阶段的产物，是系统开发和测试的指南。

1、UML类图



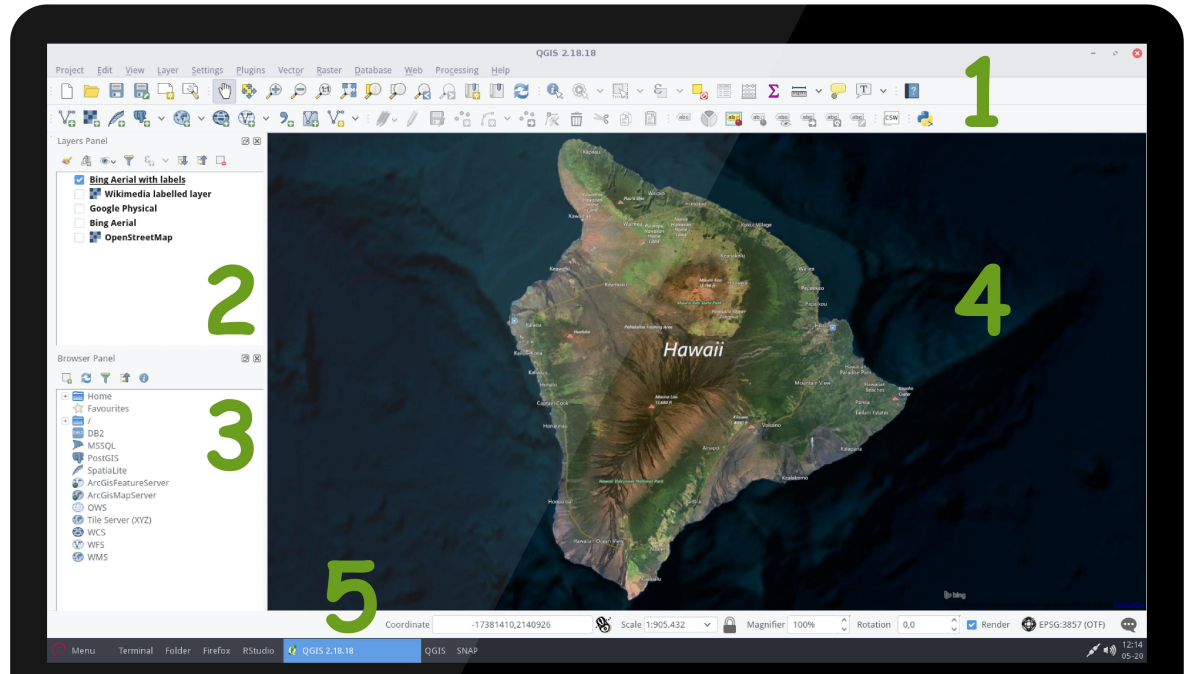
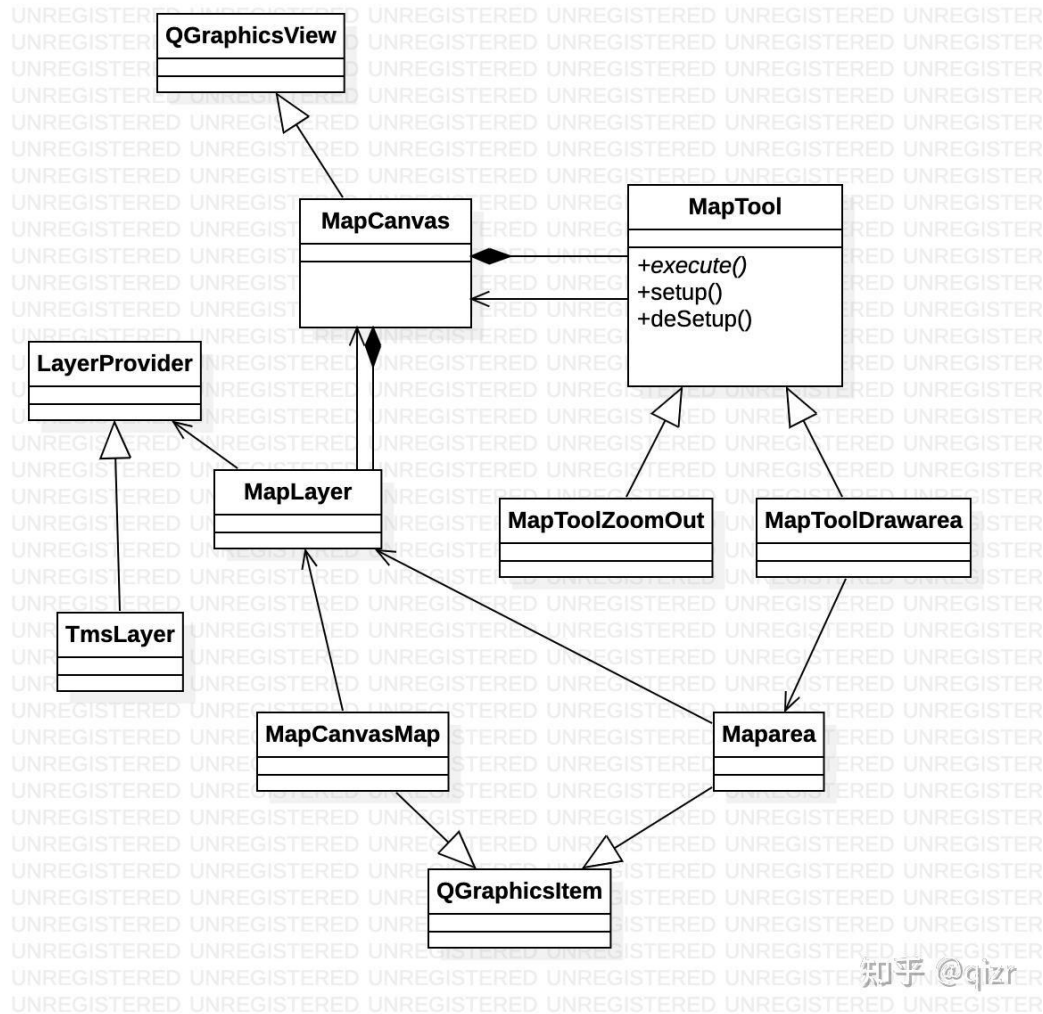
1、UML类图

□ QGIS应用程序




1、UML类图

□ QGIS应用程序



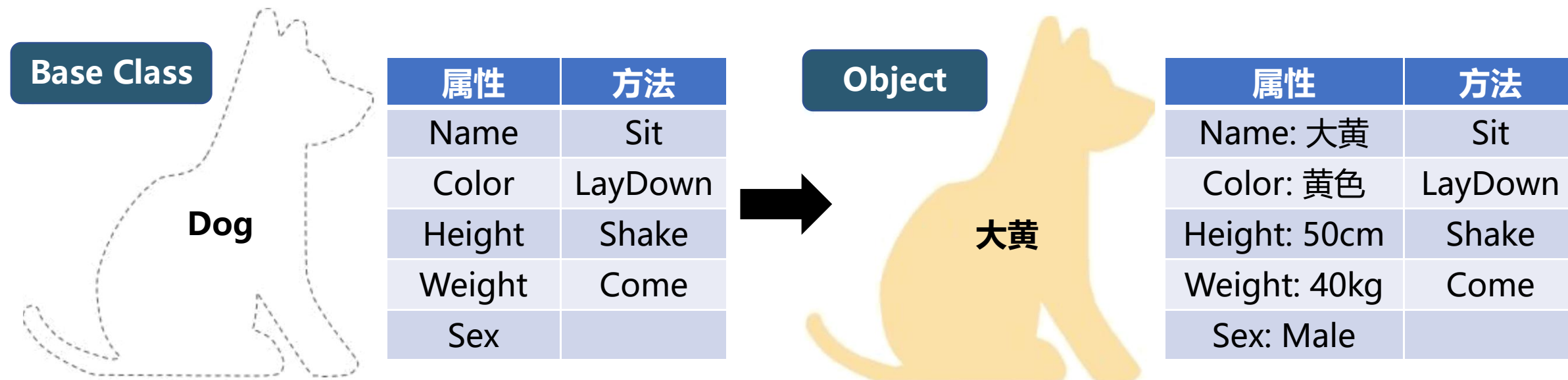
UML类图

- UML类图
- 类的表示方式 
- 类的关系表示方式
- 课外作业

2、UML中类的表示方式

□ 什么是类？

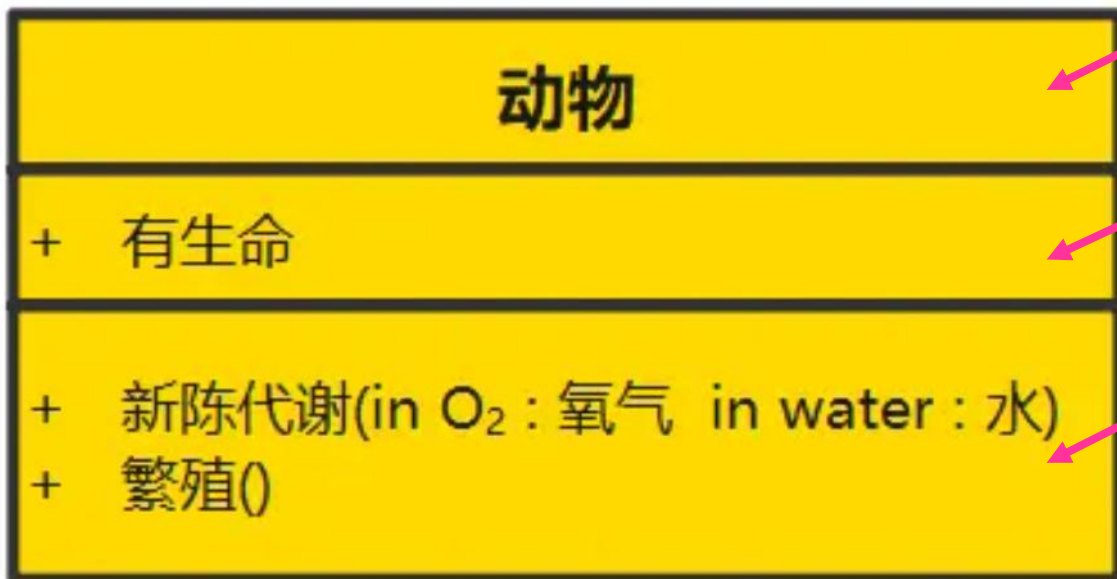
- 类是具有相似结构、行为 and 关系的一组对象的描述，是面向对象系统中最重要构造块。类描述了对对象的类型，而对象是类的可用实例。每个对象都是从同一组蓝图构建的，因此包含相同的成员（属性和方法）。
- 对象是类的实例——对象具有状态和行为。



2、UML中类的表示方式

□ **类的表示**：类表示封装状态（属性）和行为（操作）的概念。每个属性都有一个类型。每个操作都有一个签名。**类名是唯一的强制性信息。**

Class: Animal



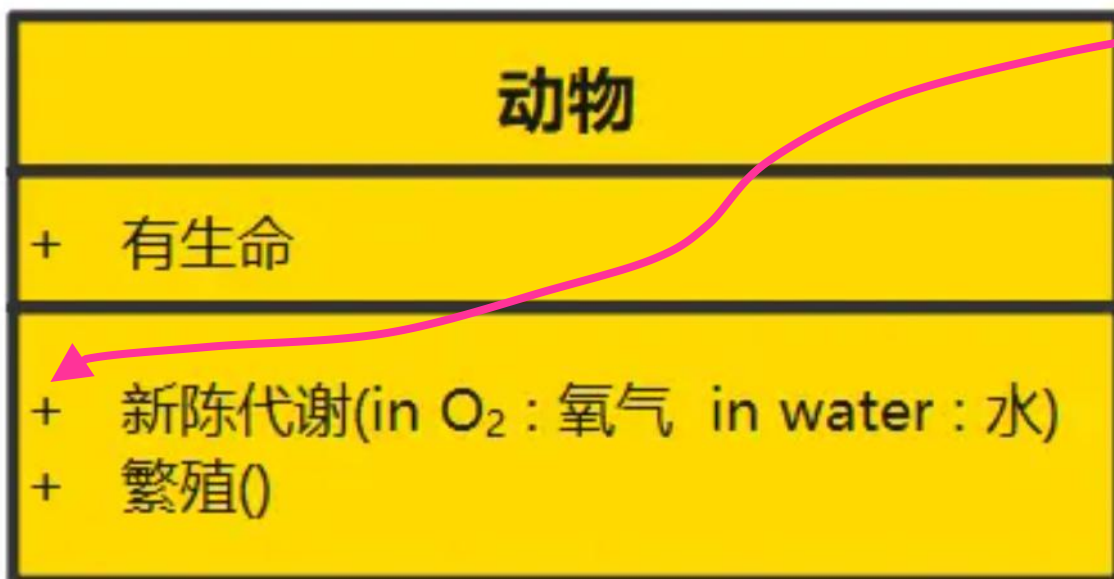
三个格子从上至下分别表示：

- **类名称**（如果是接口/抽象类，就使用**斜体**表示）
- **类的特性**（一般是类的字段和属性，可以没有）
- **类的操作**（一般是类的方法或行为）

2、UML中类的表示方式

□ **类的表示**：类表示封装状态（属性）和行为（操作）的概念。每个属性都有一个类型。每个操作都有一个签名。**类名是唯一的强制性信息。**

Class: Animal



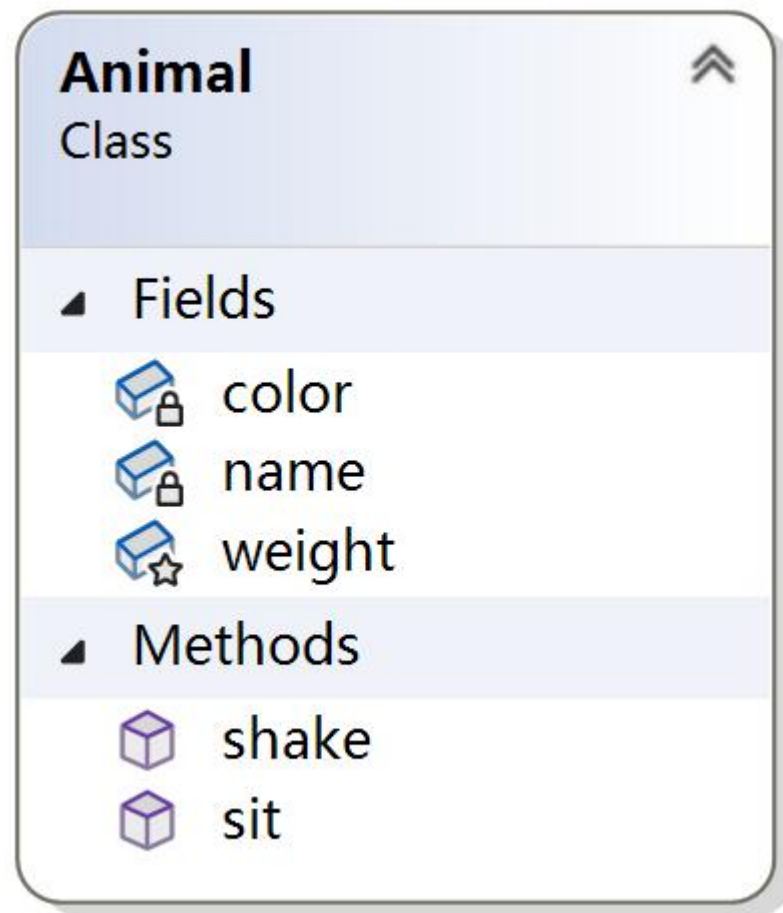
属性或方法前面的符合含义：

- "+" : 表示public成员
- "-" : 表示private成员
- "#" : 表示protected成员


2、UML中类的表示方式

□ **类的表示：** 通过Visual Studio中的类图功能查看自定义类。

```
class Animal {  
    public:  
        void sit();  
        void shake();  
    protected:  
        double weight;  
    private:  
        int color;  
        string name;  
};
```



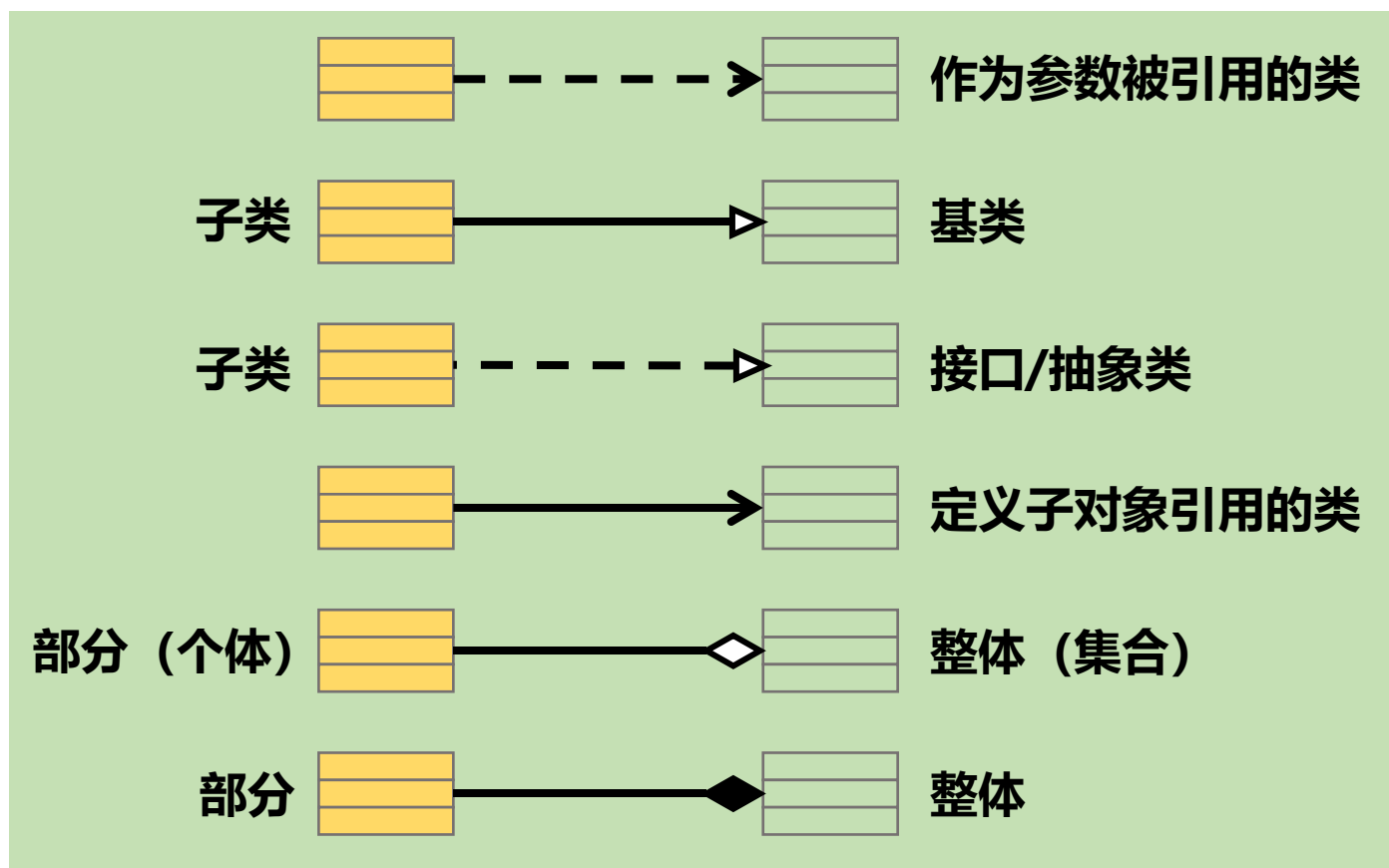
UML类图

- UML类图
- 类的表示方式
- 类的关系表示方式 
- 课外作业

3、UML中类的关系表示方式

□ **类的关系**：除了类，类图中还有一个重要元素，即类之间的关系。根据类的关系的不同，具体可分为6种：

- **依赖关系**
- **继承（泛化）关系**
- **接口实现关系**
- **关联关系**
- **聚合关系**
- **组合关系**

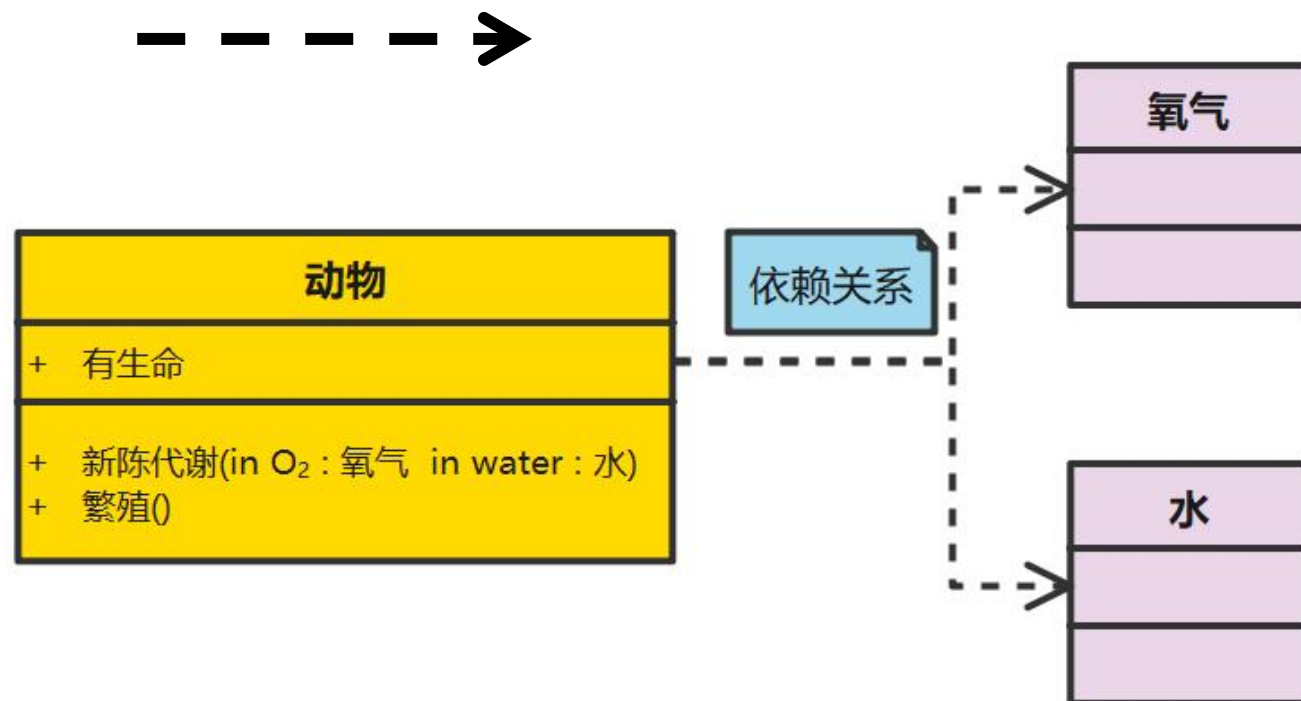


3、UML中类的关系表示方式

□ **类的关系——依赖关系**：所谓依赖关系，就是构造这个类的时候，需要依赖其他的类，比如：动物依赖水和氧气。

```
class Oxygen; // 氧气类
class Water;  // 水类
class Animal {
public:
    bool alive; // 有生命
public:
    void metaboly( // 新陈代谢
        Oxygen o, Water w);
    void breed(); // 繁殖
};
```

✓ “依赖关系” 的表示：用虚箭线表示



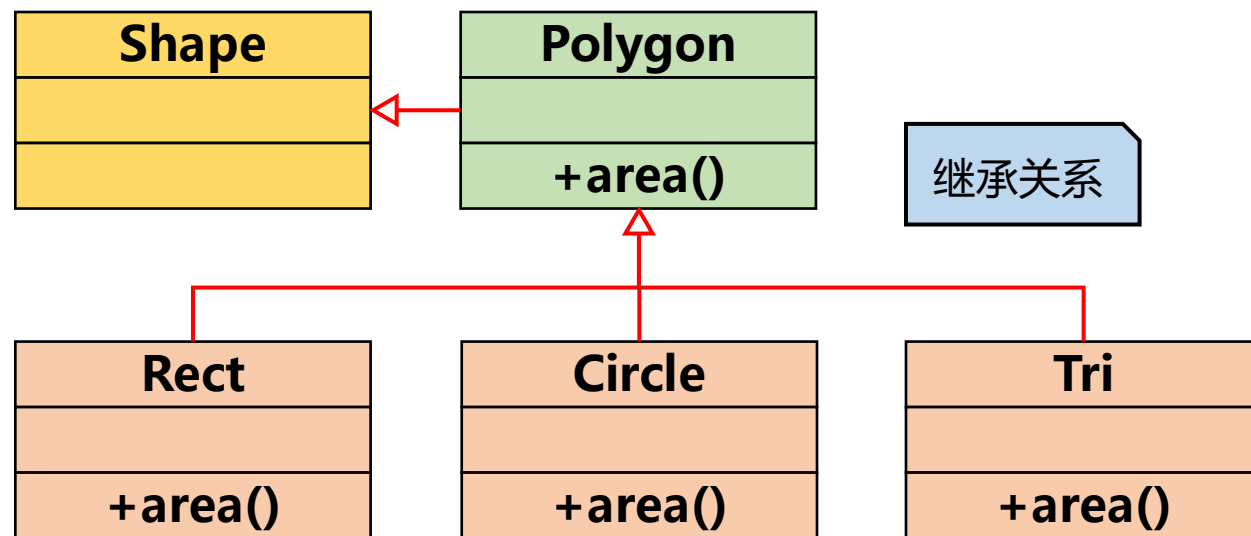
3、UML中类的关系表示方式

□ 类的关系——**继承关系**：继承（泛化）关系指定了子类如何特化父类的所有特征和行为。如：鸟是动物的一种，企鹅、鸭、大雁是鸟的一种。

```
class Shape; // 图形类
class Polygon : public Shape {
//...virtual float area();
};
class Rect : public Polygon {
//... float area();
};
class Circle : public Polygon
{///...};
class Tri : public Polygon {///...};
```

✓ “**继承关系**” 的表示：用带空心三角形的实线表示

子类  基类



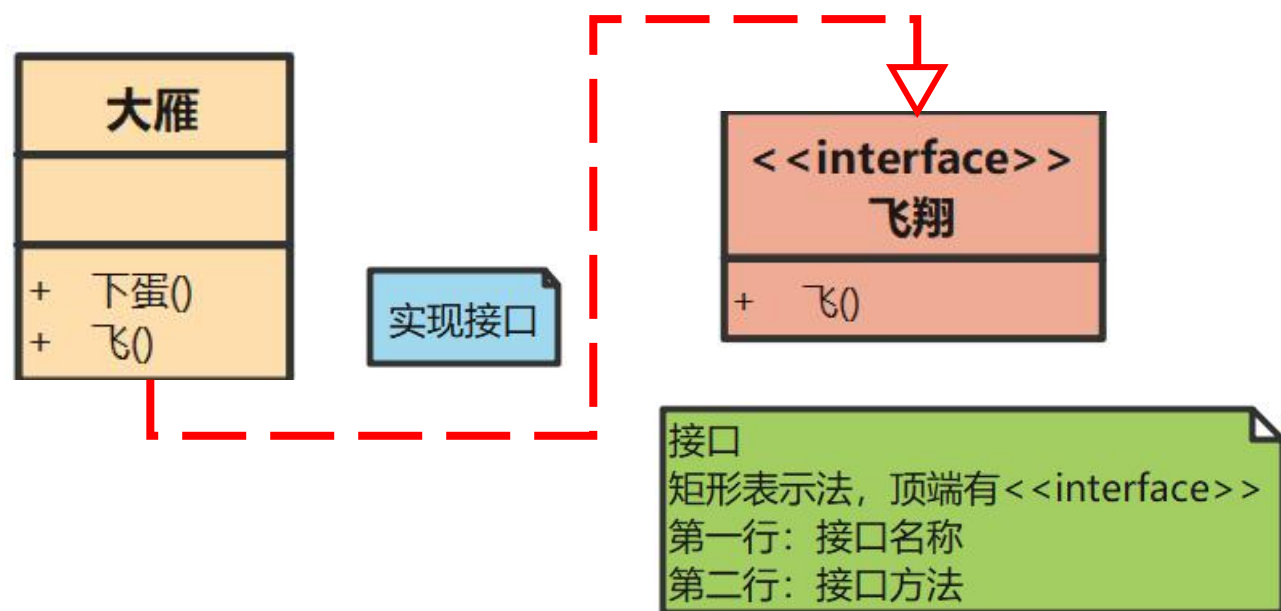
3、UML中类的关系表示方式

□ 类的关系——**接口实现关系**：接口实现关系是一种类与接口的关系，表示类是接口所有特征和行为的实现。

```
class Fly { // 飞翔接口类 (抽象类)
public:
    virtual void fly()=0;
};
```

```
class Goose : public Fly {
public:
    void fly(){ // goose fly ...}
};
```

✓ “**接口实现关系**” 的表示：用带空心三角形的虚线表示 子类 — — — — ▷ 接口类(抽象类)

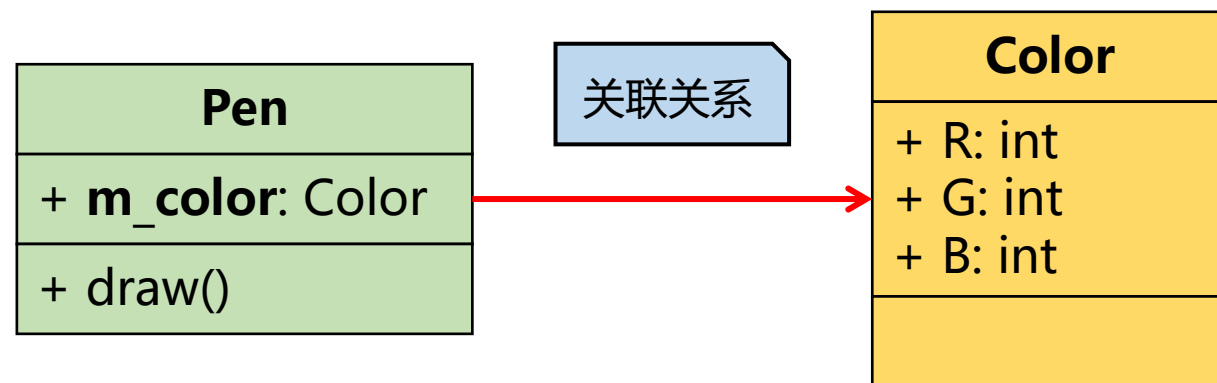


3、UML中类的关系表示方式

□ **类的关系——关联关系**：所谓关联关系，就是这个类有一个属性是其他类（包含**子对象**的类）。

```
class Color { // 颜色类
    public:
        int R, G, B;
};
class Pen { // 画笔类
    public:
        Color m_color; // 颜色属性
        void draw();
};
```

✓ “**关联关系**” 的表示：用**实箭线**表示



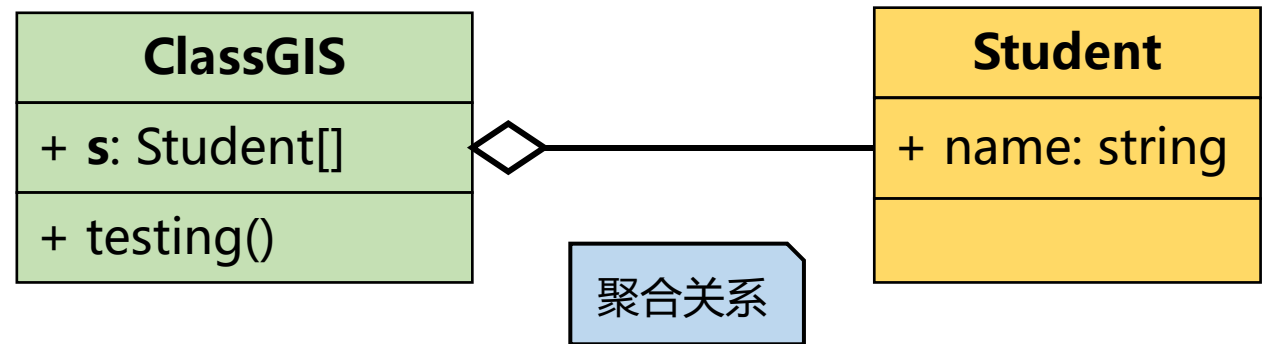
3、UML中类的关系表示方式

□ **类的关系——聚合关系**：聚合关系是表达部分与整体之间的一种关联关系，是强的关联关系。特点：部分对象的生命周期并不由整体对象来管理，整体对象不存在时，部分的对象仍可以存在。如：大雁与雁群。

```
class Student{ // 学生类
    public: string name;
};
class ClassGIS { // GIS班级类
    public:
        Student s[30];
        void testing();
};
```

✓ **“聚合关系” 的表示**：用带空心菱形的实线表示

部分 ———— ◇ 整体（集合）



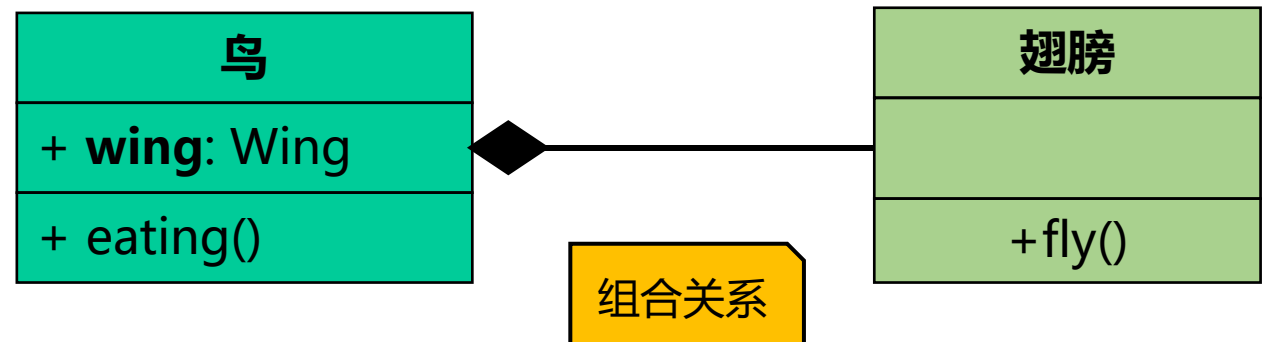
3、UML中类的关系表示方式

□ **类的关系——组合关系**：组合关系同样是表达部分与整体间的关联关系，是比聚合关系还要强的关系。特点：在组合中，部分与整体生命期一致，部分与组合同时创建并同时消亡。比如：鸟与翅膀的关系。

```
class Wing{ // 翅膀类
    public:
        void fly();
};
class Bird { // 鸟类
    public:
        Wing wing;
        void eating();
};
```

✓ “**组合关系**” 的表示：用带实心菱形的实线表示

部分 ———— ◆ 整体



UML类图

- UML类图
- 类的表示方式
- 类的关系表示方式
- 课外作业 

4、课后作业

- 根据本节所讲的UML类图知识，绘制一个你所熟悉的软件系统的基本类图，如：选课系统、购物系统、GIS程序等。
- 学会在Visual Studio中查看类图的基本方法，编写基本的类与及派生图，查看VS中类图的表示方式。
- 通过UML类图的学习，进一步理解面向对象程序设计的基本思想，对比《C++程序设计》课程中所学的结构化编程知识，总结面向对象编程的特点与具有的优势。