

UML类图&类的六大关系

寂然

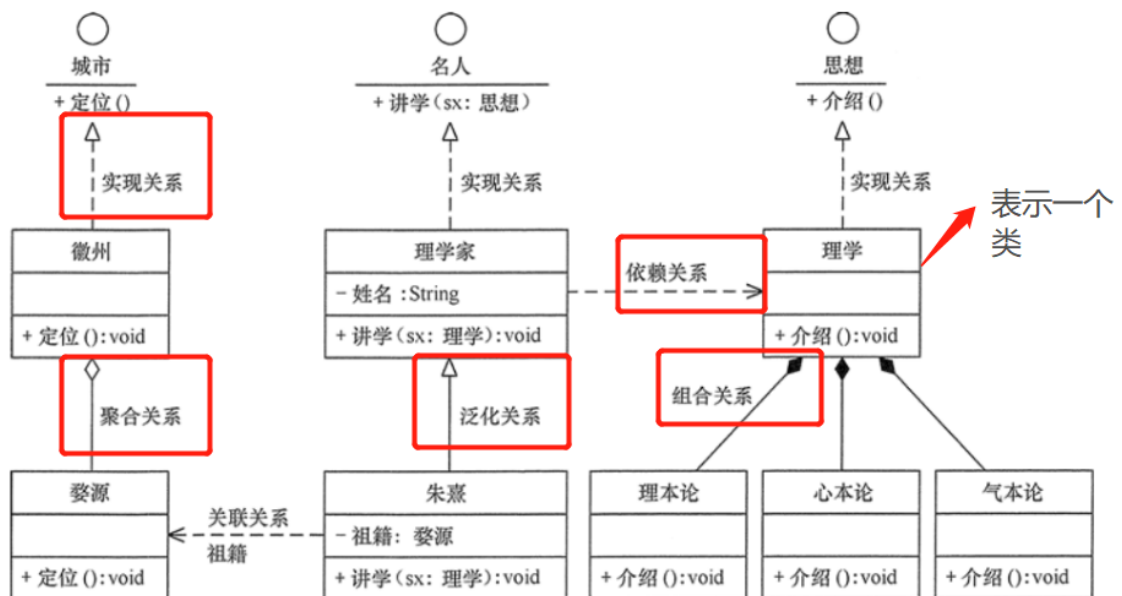
官方定义

UML - Unified modeling language UML (统一建模语言)，是一种用于软件系统分析和设计的语言工具，

用于帮助软件开发人员进行思考和记录思路的结果

UML 图形化的语言

基本介绍



UML图：通过不同的图形和符号，来描述软件模型以及各个元素之间的关系

UML 图分类

- 用例图(use case)
- 静态结构图：**类图**、对象图、包图、组件图、部署图
- 动态行为图：交互图（时序图与协作图）、状态图、活动图

UML类图：描述类之间的关系

建模工具

- word，利用word工具可以绘制简单的UML图，但是这是一种非专业的手段，而且绘制起来比较麻烦
- **利用VISIO工具进行UML图的绘制**
- 利用PowerDesigner工具绘制UML图形，PowerDesigner支持标准数据建模UML
- 利用Eclipse中的AmaterasUML插件，也可以进行UML的绘制
- IDEA中可以使用PlantUML插件来画类图，也可以**下载安装UML Support插件 附录图**

依赖关系

是一种使用关系，一个类的实现，需要另一个类的协助，逻辑上 use a 表示

只要在类中使用到了对方，是依赖关系，是一种最基本的关系

其他的五种关系，都是依赖关系的特例

代码体现

- 类中使用到了对方
- 类的成员属性
- 方法的接受的参数类型
- 方法的返回值类型
- 方法内部使用到了

```
//案例演示 - 依赖关系的形式
public class PersonServiceBean {

    private PersonDao personDao; //类的成员属性
```

```

    public void save(Person person){ //方法的接收参数类型

    }

    public IDCard getIDCard(){ //方法的返回值类型

        return new IDCard();
    }

    public void modify(){

        //违反了迪米特法则 可以这样写，
        Department department = new Department(); //方法内部使用到

    }

}

class PersonDao{ }

class IDCard{ }

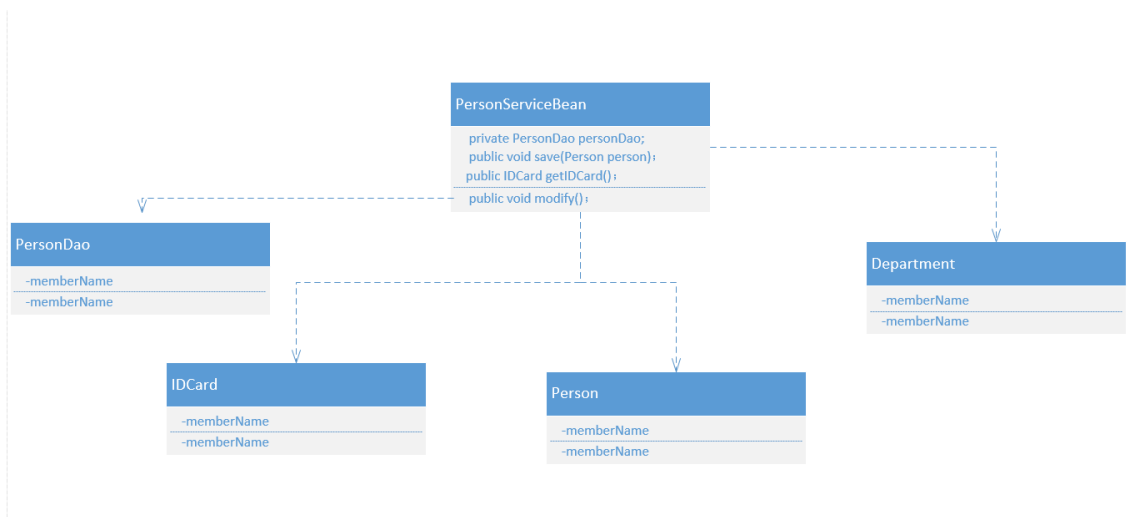
class Person{ }

class Department{ }

```

UML图示

UML符号：一条虚线加箭头的形式



泛化关系

泛化关系指的就是继承关系（语法）

代码体现

```
//案例演示 - 泛化关系
class PersonBean{

    public void save(){
        //TODO...
    }

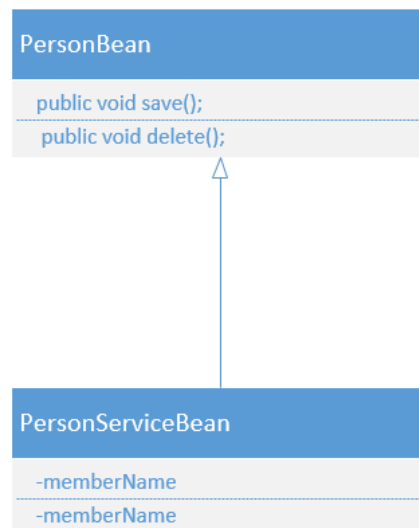
    public void delete(){
        //TODO...
    }
}

public class PersonServiceBean extends PersonBean{

}
```

UML图示

UML符号：一条实线加三角箭头，**箭头要指向父类**



实现关系

实现关系实际上 A类实现了B接口，A,B存在实现关系（语法层面）

代码体现

//案例演示 - 实现关系

```
interface PersonService{

    void delete();
}

public class PersonServiceBean implements PersonService{

    @Override
    public void delete() {
        System.out.println("delete");
    }
}
```

UML图示

UML符号：一条虚线加三角箭头 箭头方向指向接口



关联关系

关联关系体现的是业务逻辑上的关系

关联关系具有导航性：单向关联，双向关联（代码层面上）

关联关系还具有多重性：一对一，一对多，多对多（业务逻辑上）

代码体现

```
//单向一对一
class Person{

    public IDCard idCard;
}

class IDCard{

    //并没有使用到Person类

}

-----

//双向一对一
class Person{

    public IDCard idCard;
}

class IDCard{

    public Person person;
}

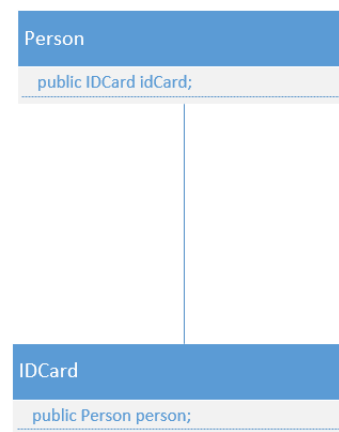
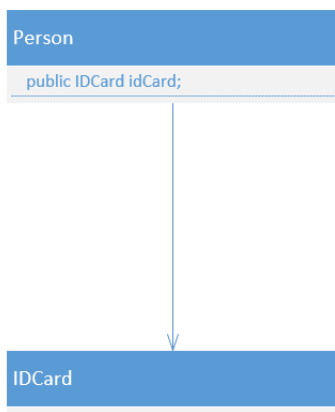
}
```

UML图示

UML符号

单向关联 一条实线加一个箭头

双向关联 一条实线 （一条实线加两个箭头）



聚合关系

聚合关系，是关联关系的一个特例

具有导航性 具有多重性

聚合关系描述的是**业务逻辑上**整体和部分的关系 整体和部分如果可以分开 就构成聚合关系

代码体现

```
//案例演示 - 聚合关系
public class Computer {

    public Monitor monitor;

    public Mouse mouse;

}

class Monitor{

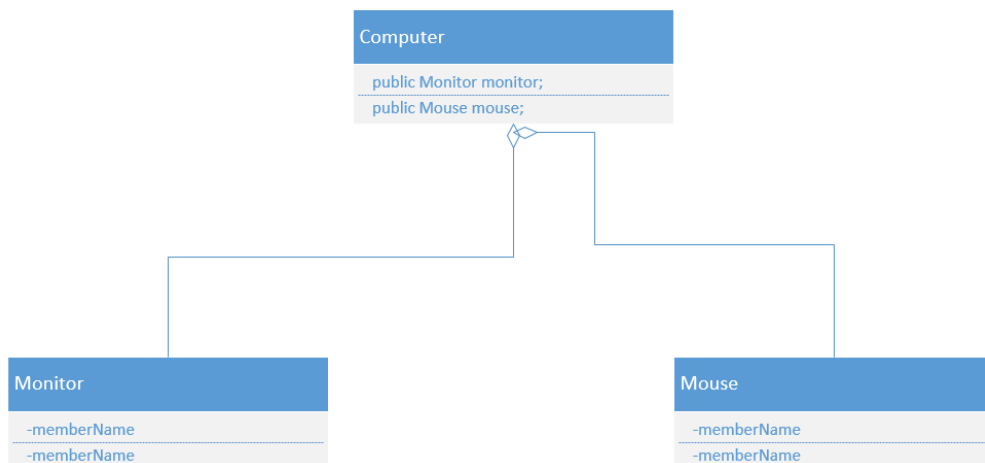
}

class Mouse{

}
```

UML图示

UML符号：一条实线加上空心菱形来表示聚合关系



组合关系

下节预告

设计模式 - 单例模式 八种写法 ,