

依赖与关联关系

➤ 依赖关系

依赖关系(Dependency) 是一种使用关系，特定事物的改变有可能会影响到使用该事物的其他事物，在需要表示一个事物使用另一个事物时使用依赖关系。如人要过河时要使用到船，人与船之间就是依赖关系；我用锤子在墙上锤了一下，我和锤子之间就是依赖关系；装修工人安装灯具时需要使用工具，装修工人和工具之间就是依赖关系。大多数情况下，依赖关系体现在某个类的方法使用另一个类的对象作为参数。

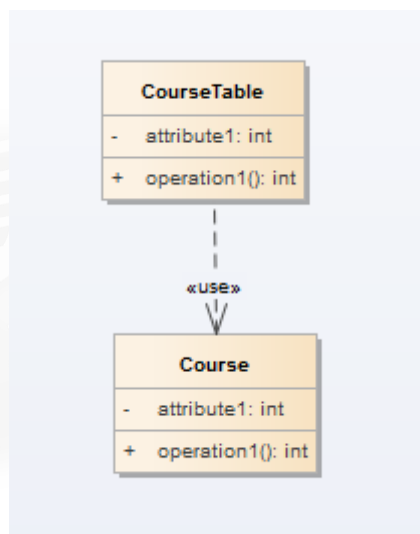
依赖关系有下列三种情况：

- A 类是 B 类中（某种方法的）局部变量
- A 类是 B 类方法其中的一个参数
- A 类向 B 类发送消息，从而影响 B 类发生变化

我们这里就依赖关系最常出现的情况——在某个类的方法使用另一个类的对象作为参数进行举例说明：

课程表与课程之间就是依赖关系，

UML 图表示为：



对应的 java 代码片段：

```
//课程表类
public class CourseTable {
    public void addCourse(Course course){
        course.show();
        System.out.println("增加课程");
    }
    public void reCourse(Course course){
        course.show();
        System.out.println("删除课程");
    }
}
//课程类
public class Course{
    public void show(){
        System.out.println("课程为...");
    }
}
```

课程表类中用课程的对象作为参数，进行增加课程和删除课程的功能。课程的变化会引起课程表的实时变更，这就是一种“use a”的关系，也就是课程表依赖于课程。

➤ 关联关系

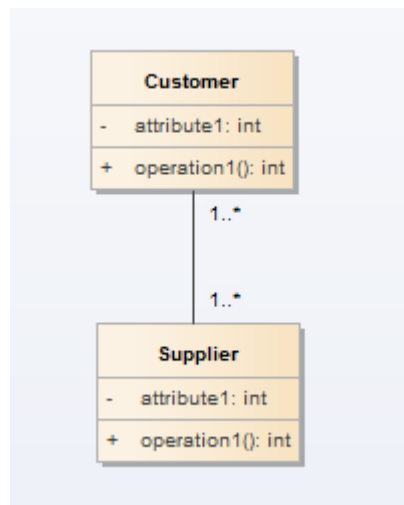
关联关系是类与类之间最常用的一种关系，它是一种结构化关系，代表类的对象之间的一组连接。如客户与订单、老师与学生、公司与职员，人体与大脑，这些关系不是使用时才有的，而是一种长期且稳定的关系。

关联关系可以分为双向关联、单向关联、自关联。

a) 双向关联：

指双方都知道对方的存在，都可以调用对方的公共属性和方法。默认情况下，关联是双向的。如供货商和顾客之间的关系，一家供货商可以向多个顾客供货，一个顾客也可以向多家供货商进货

UML 图表示为：



对应的java 代码片段：

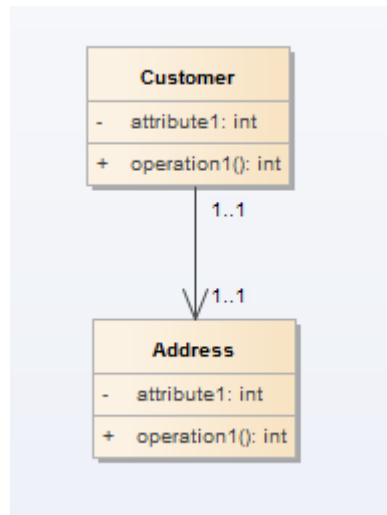
```
//顾客类
public class Customer{
    //供货商对象组作为Customer的属性成员
    private Supplier[] supplier;
    .....
}
//供货商类
public class Supplier{
    //顾客对象组作为Supplier的属性成员
    public Customer[] customer;
    .....
}
```

b) 单向关联：

类的关联关系也可以是单向的，单向关联用带箭头的实线表示。例如：顾客

(Customer)拥有地址(Address)，则 Customer 类与 Address 类具有单向关联关系

UML 图表示为：



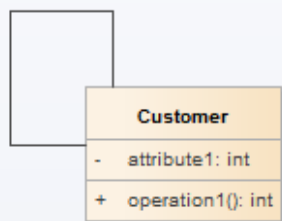
对应的 java 代码片段：

```
//顾客类
public class Customer{
    private Address address;
    .....
}
//地址类
public class Address {
    .....
}
```

c) 自身关联（反身关联）：

在系统中可能会存在一些类的属性对象类型为该类本身，这种特殊的关联关系称为自关联。顾客自身需要用到自己对象时，自己引用自己，带着一个自己的引用。

UML 图表示为：



对应的 java 代码片段：

```
//顾客类
public class Customer{
    private Customer customer;
    .....
}
```

➤ 补充扩展：

重数性关联：表示一个类的对象与另一个类的对象连接的个数。

在 UML 中多重性关系可以直接在关联直线上增加一个数字表示与之对应的另一个类的对象的个数。在关联的两端可写上一个被称为重数的范围，表示该类有多少个对象可与对方一个对象连接，重数的默认值为 1，重数的符号通常有：

表示方式	多重性说明
1..1	表示“1”，表示另一个类的一个对象只与一个该类对象有关系
0..*	表示“零或多”，表示另一个类的一个对象与零个或多个该类对象有关系
1..*	表示“1 或多”，表示另一个类的一个对象与一个或多个该类对象有关系
0..1	表示“零或 1”，表示另一个类的一个对象没有或只与一个该类对象有关系
m..n	表示另一个类的一个对象与最少 m、最多 n 个该类对象有关系 ($m \leq n$)