

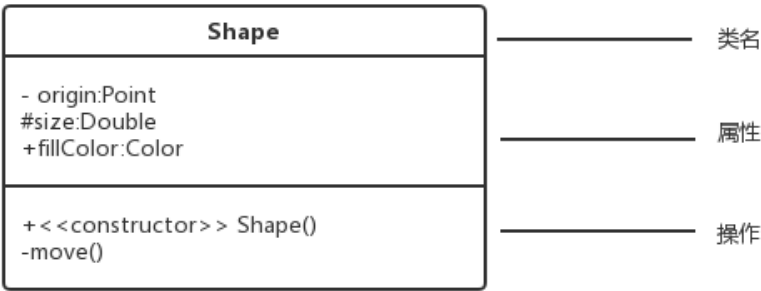
# 类图和对象图

原创xyzso1z最后发布于2019-06-30 04:52:26阅读数 377☆收藏

编辑展开

## 1.类的定义

类的定义：类是具有相似结构、行为和关系的一组对象的描述符。



<https://blog.csdn.net/xyzso1z>

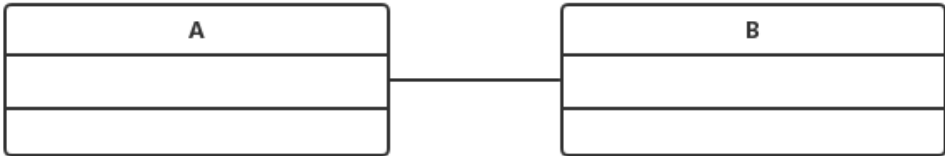
## 2.类之间的关系

### 2.1 关联

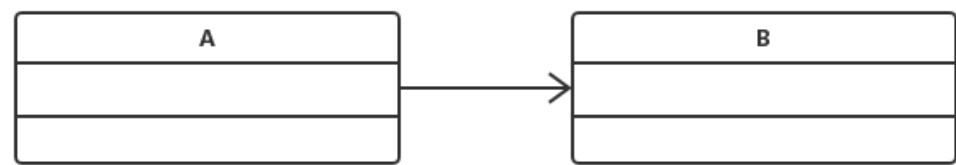
关联（association）是模型元素间的一种语义联系，它是对具有共同的结构特征、行为特性、关系和语义的链的描述。

在上面的定义中，需要注意的是链这个概念，链是一个实例，就像对象是类的实例一样，而链表示的是对象与对象之间的关系。

在类图中，关联用一条把类连接在一起的实线表示，如图：

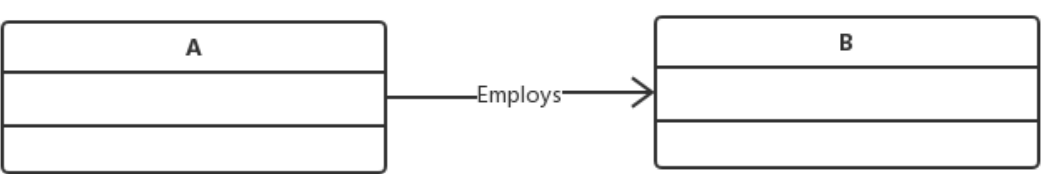


一个关联可以有两个或多个关联端，每个关联端连接到一个类。关联也可以有方向，可以是单向关联或双向关联。如下图表示的是双向关联，表示的是从类A到类B的单向关联。



2.1.1 关联名

可以给关联加上关联名，来描述关联的作用。如图使用关联名的一个例子，其中Company类和Person类之间的关联如果不使用关联名，则可以有多种解释，如Person类可以表示是公司的客户、雇员或所有者等。但如果在关联上加上Employs这个关联名，则表示Company类和Person类之间是雇佣（ Employs ）关系，显然这样语义上更加明确。**一般说来，关联名通常是动词或动词短语。**



2.1.2 关联的角色

关联两端的类可以某种角色参与关联。如下图，Company类以employer的角色、Person类以employee的角色参与关联，employer和employee称为角色类。如果在关联上没有标出角色名，则隐含地用类的名称作为角色名。



2.2

聚集和组合

聚集是一种特殊形式的关联。聚集表示类之间整体与部分的关系。在对系统进行分析和设计时，需求描述中的“包含”、“组成”、“分为.....部分”等词常常意味着存在聚集关系。

组合表示的也是类之间的整体与部分的关系，但组合关系中的整体与部分具有同样的生存期。也就是说，组合是一种特殊形式的聚集。

如图分别是聚集关系和组合关系的例子



聚集关系

<https://blog.csdn.net/xyzso1z>

组合关系

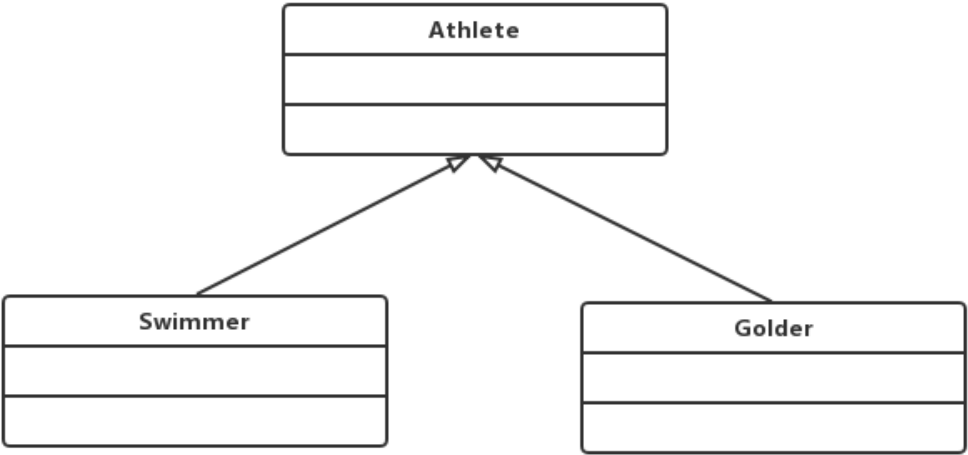
<https://blog.csdn.net/xyzso1z>

Circle类和Style类之间是聚集关系。一个圆可以有颜色、是否填充这些样式方面的属性，可以用一个style对象表示这些属性，但同一个style对象也可以表示别的对象如三角形的一些样式方面的属性，也就是说，style对象可以用于不同的地方。如果circle这个对象不存了，不一定意味着style这个对象也不存在了。

下面列出聚集和组合之间的一些区别：

- 聚集关系也称为“has -a”关系，组合关系也称为“contains-a”关系。
- 聚集关系表示事物的整体/部分关系的较弱的情况，组合关系表示事物的整体/部分关系的较强的情况。
- 在聚集关系中，代表部分事物的对象可以属于多个聚集对象，可以为多个聚集对象所共享，而且可以随时改变它所从属的聚集对象。代表部分事物的对象与代表聚集事物对象的生存期无关，一旦删除了它的一个聚集对象，不一定也就随即删除代表部分事物的对象。在组合关系中，代表整体事物的对象负责创建和删除代表部分事物的对象，代表部分事物的对象只属于一个组合对象。一旦删除了组合对象，也就随即删除了相应的代表部分事物的对象。

• 2.3 泛化关系



<https://blog.csdn.net/xyzso1z>

泛化(generalization)定义了一般元素和特元素之间的分类关系，如果从面向对象程序设计语言的角度来说，类与类之间的泛化关系就是平常所说的类与类之间的继承关系。  
泛化关系也称为" a -kind -of"关系。

• 2.4 依赖关系



<https://blog.csdn.net/xyzso1z>

假设有两个元素X、Y,如果修改元素X的定义可能会导致对另一个元素Y的定义的修改，则称元素Y依赖于元素X。  
对于类而言，依赖(dependency)关系可能由各种原因引起，如一个类想另一个类发送消息，或者一个类是另一个类的数据成员类型，或者一个类是另一个类的操作的参数型等。  
有时依赖关系和关联关系比较难区分。事实上，如果类A和类B之间有关联关系，那么类A和类B之间也就有依赖关系了。但如果两个类之间有关联关系，那么一般只要表示出关联关系即可，不用再表示这两个类之间还有依赖关系。而且，如果在一个类图中有过多的依赖关系，反而会使类图难以理解。  
与关联关系不一样的是，依赖关系本身不生成专门的实现代码。

