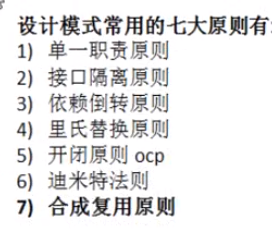
# 一、设计模式简介

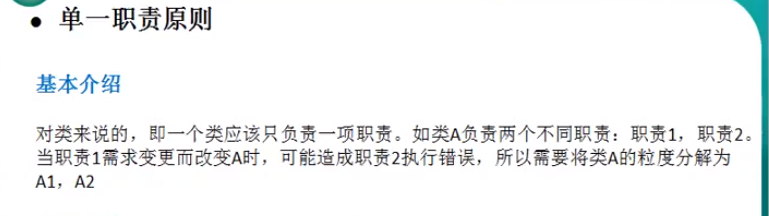
## 设计模式的目的



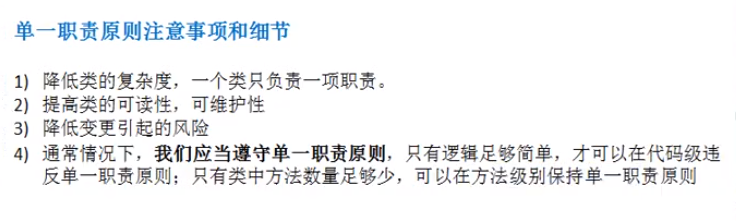
## 设计模式7大原则



### 单一职责原则:

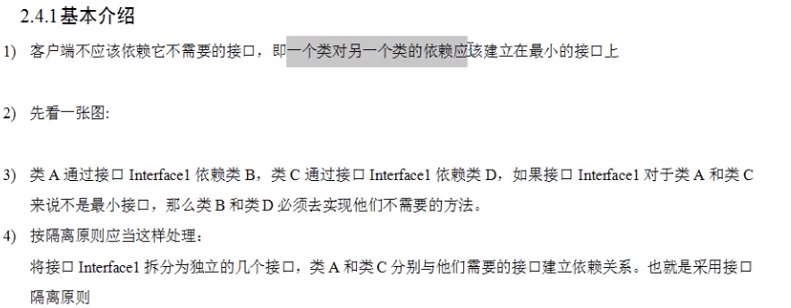


#### 注意事项

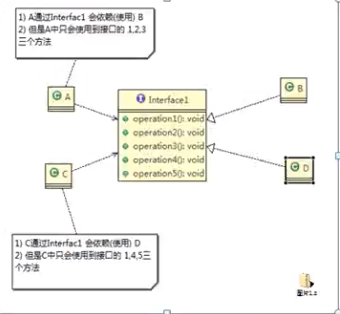


### 接口隔离原则:

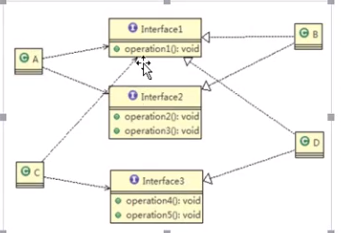
#### 基本介绍



#### 应用场景:

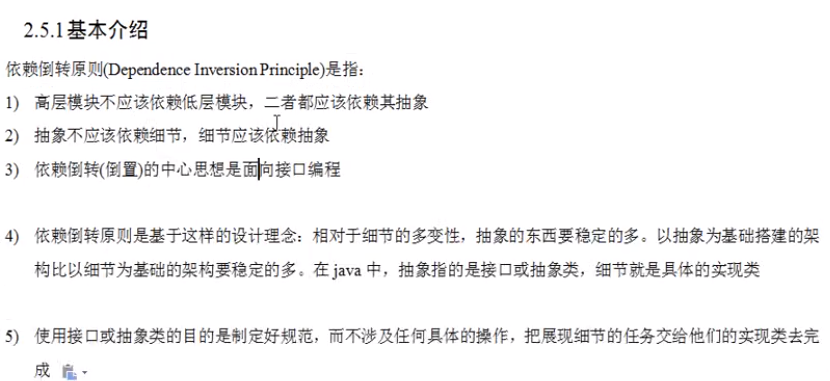


#### 解决方案:

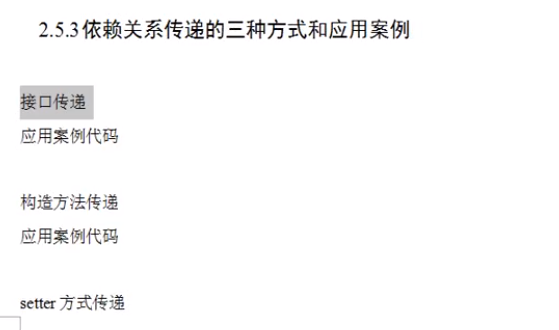


### 依赖倒转原则

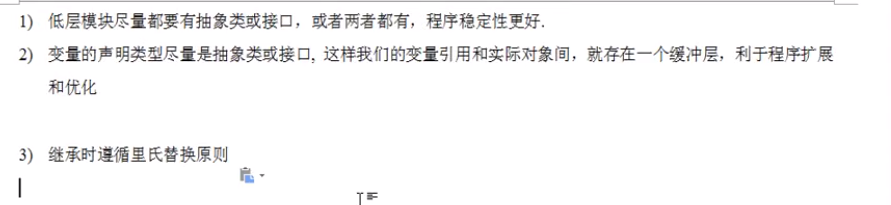
#### 基本介绍



#### 传递方式

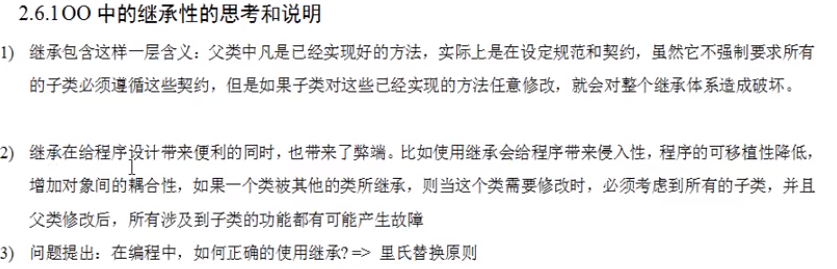


#### 注意事项:

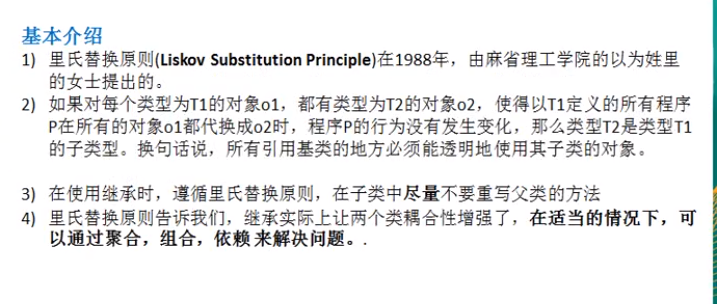


### 里氏替换原则:

#### 为什么使用里氏替换原则



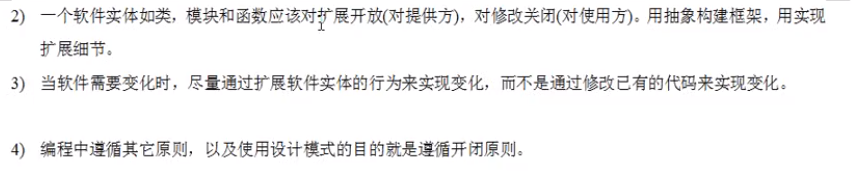
#### 基本介绍



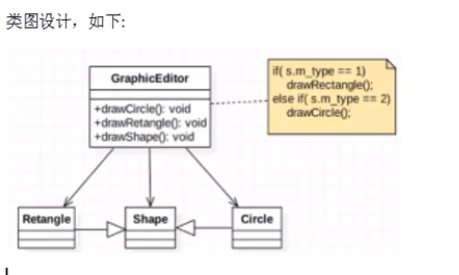
### 开闭原则:

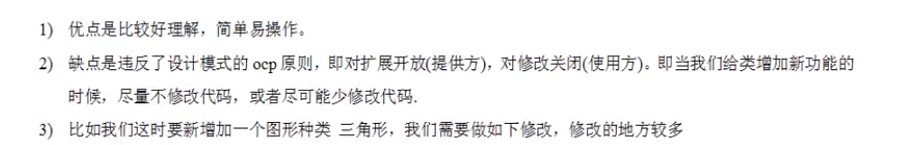
#### 基本概念



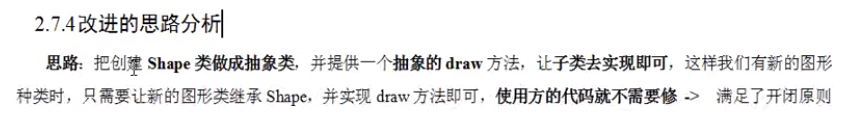


#### 原始类图



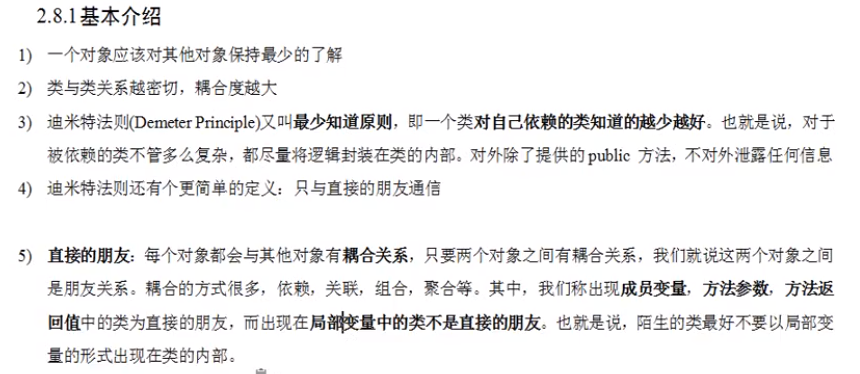


#### 修改

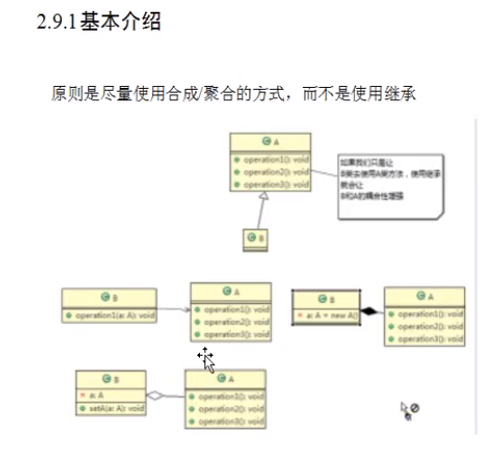


### 迪米特法则:

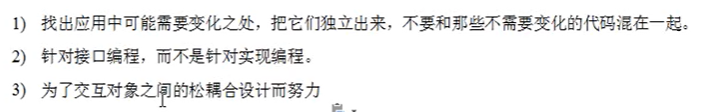
#### 基本介绍



### 合成复用原则:

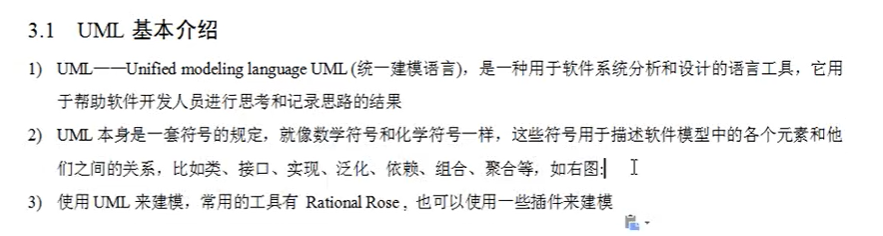


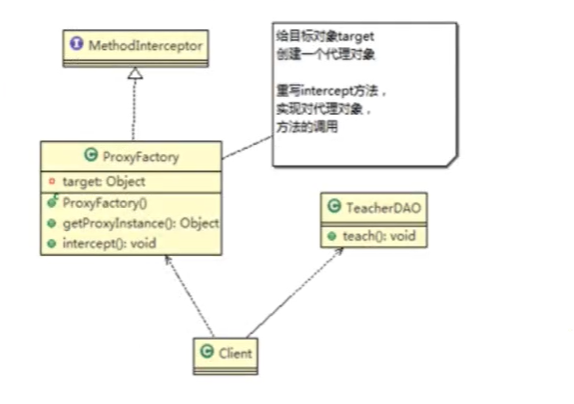
### 设计原则核心思想



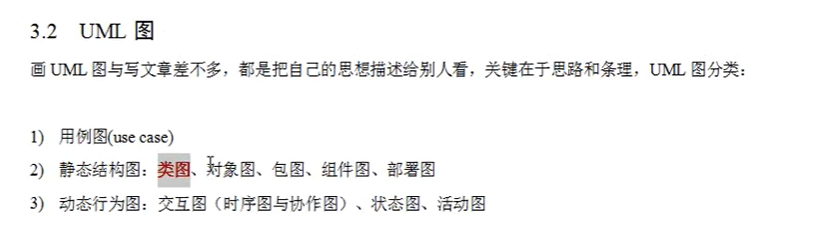
# 二、UML类图

## 类、接口、实现、泛化、依赖、组合、聚合

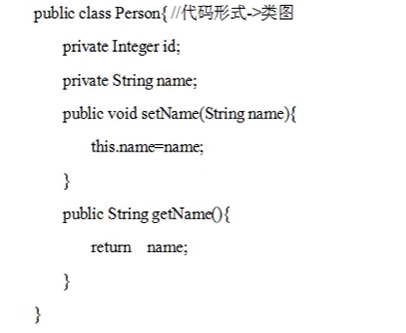




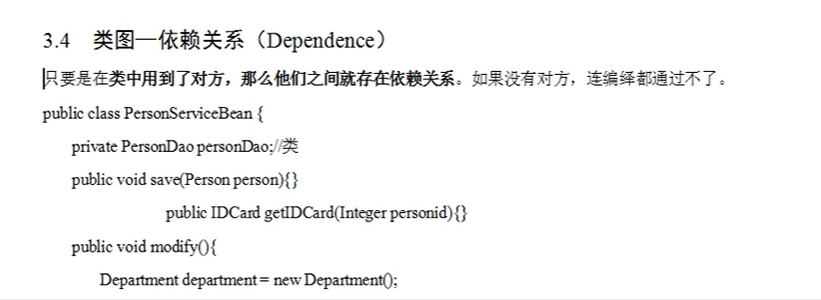






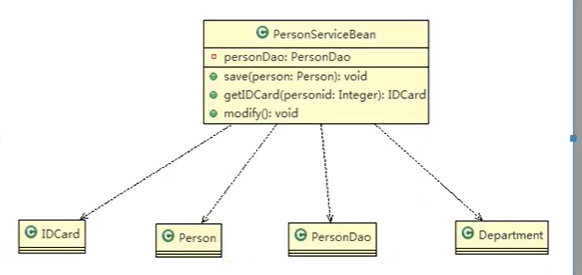


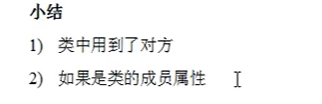


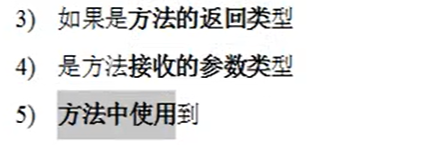


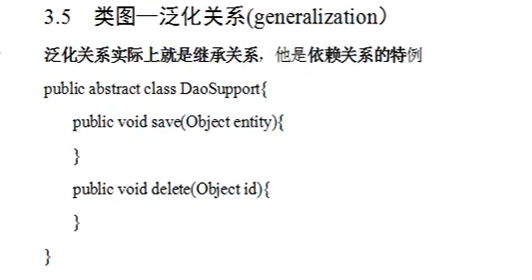


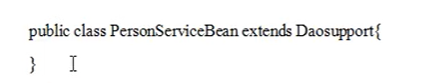
对应的类图





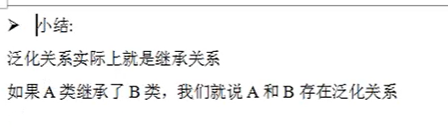




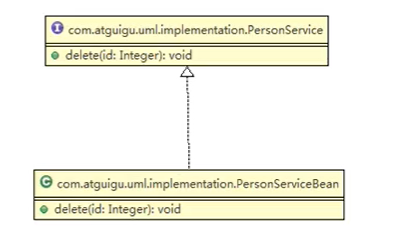


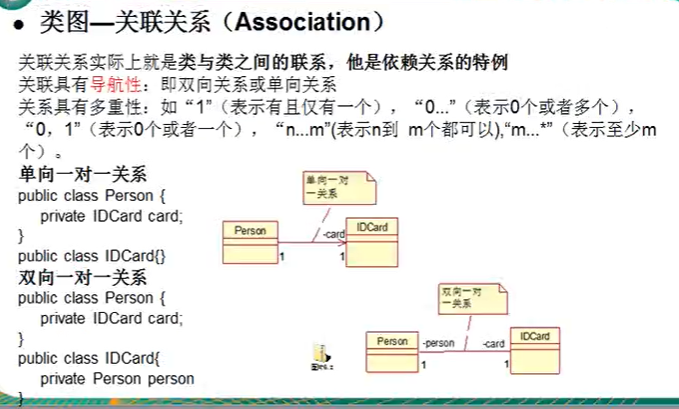
对应的类图

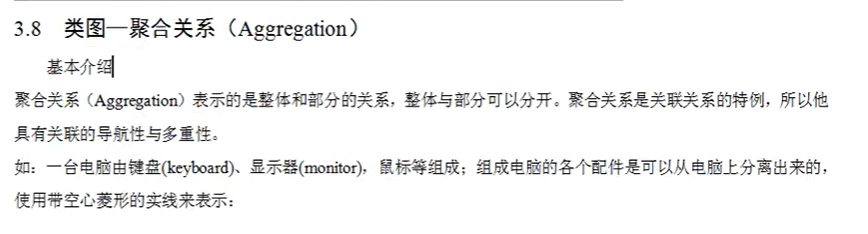


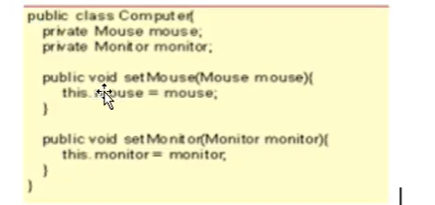
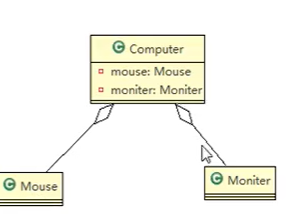




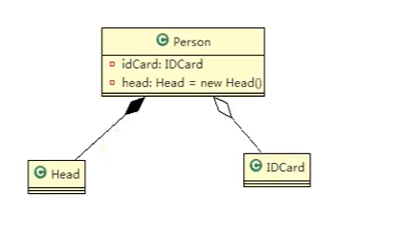
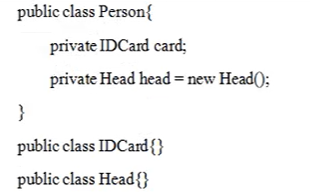




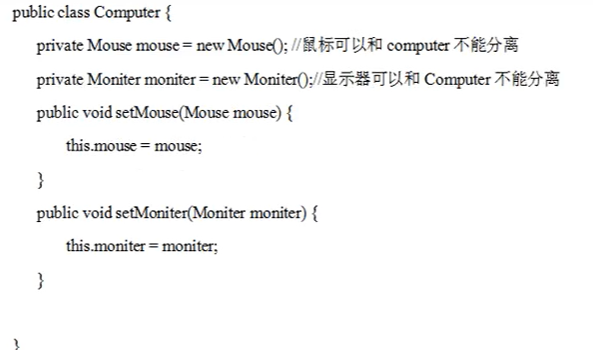


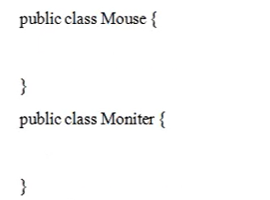




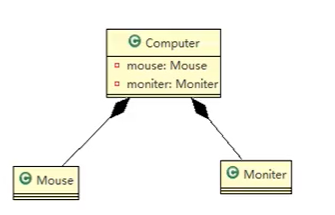
head是组合（整体与部分不可分开） idcard是聚合（整体与部分可分开）

案例二

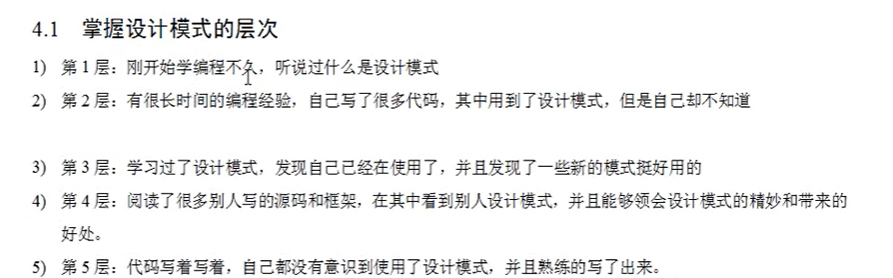


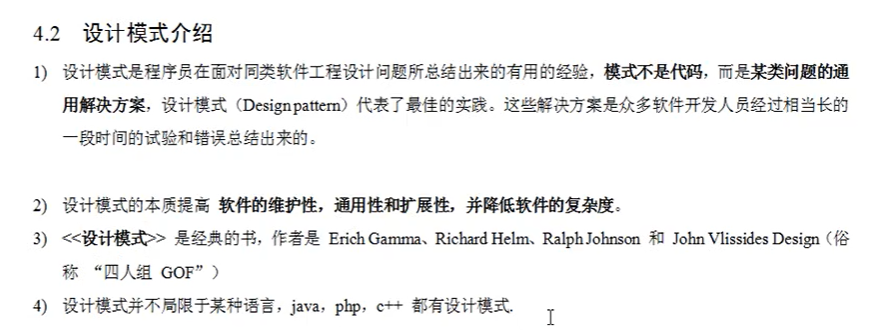


对应的类图



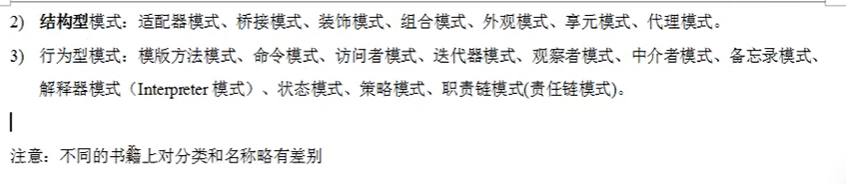
# 三、设计模式概述





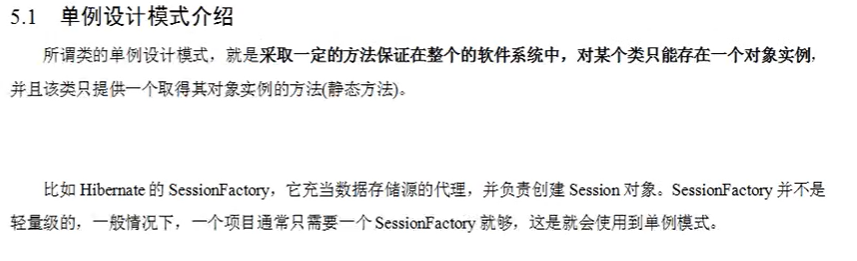


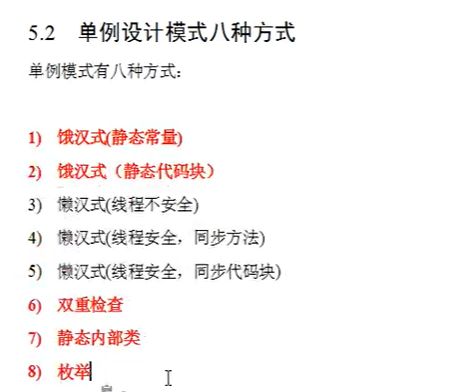




# 23种设计模式

## 1.单例设计模式





ps:红色是可以使用的

静态内部类:当加载外部类的时候静态内部类并不会被加载,只有用到才会加载

静态内部类代码:



双重检查:

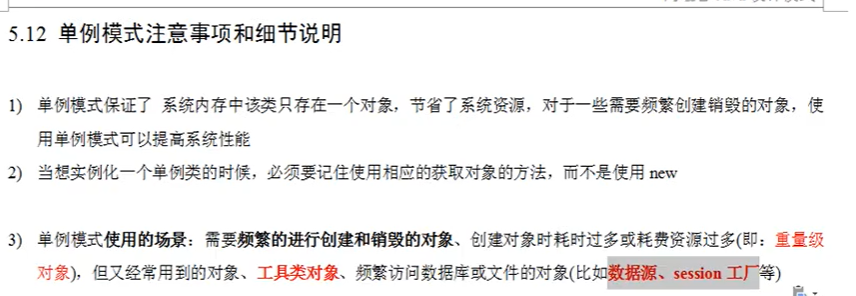


枚举方式:



jdk中用到的单例:

Runtime类



## 2.简单工厂模式

## 3.工厂方法模式

## 4.抽象工厂模式

## 5.原型模式

## 6.建造者模式

## 7.适配器模式

## 8.桥接模式

## 9.装饰者模式