# 1.构造方法注入

构造方法注入，就是在构造方法的参数列表中声明所有的被依赖对象，让IOC知道它依赖哪些对象。

配置方式：

# 2.setter方法注入

对于JavaBean来说，经常使用setter方法和getter方法来设置和获取属性，所以对于当前对象来说，如果想要获取其依赖的对象，只需要为其依赖对象所对应的属性添加setter方法，就可以通过setter方法将对应的依赖对象设置 被注入对象中。

# 3.接口注入

被注入对象如果想要IOC Service Provider为其注入依赖对象，就必须实现某个接口。这个接口提供一个方法，用来为其注入依赖对象。Ioc Service Provider最终通过这些接口来了解应该为被注入对象注入什么依赖对象。

# 4.三种注入方式的比较

**接口注入:** 接口注入强制要求被注入对象来实现不必要的接口，代码具有侵入性，逐渐被淘汰。

**构造函数注入：** 构造函数注入的优点是不具有侵入性，而且对象构造完成后，立马就进入可以使用的就绪状态。

缺点有四点：一是参数过多时会导构造函数参数列表过长。二是反射时对于相同类型的参数不好处理。三是对于非必须的依赖处理，可能会导致多个构造方法；四是无法通过属性名称注入，无法明确的确定注入属性的位置，只能通过序号来表示注入的位置，一旦属性的相对位置改变，序号就会改变。

**Setter方法注入:** setter方法注入的优点有三点：一是由于方法的可命名，所以在描述性上比构造函数注入更好；二是方法可继承，可以设置默认值；三是良好的IDE支持。

缺点是对象在构造完成后无法马上进入就绪状态。