**Resoure目录下添加配置文件 Application.xml**

**如下图**



**一，Application.xml的作用**



**加载配置文件，形成Spring容器**

**二，<bean>标签以及bean**

**1，<bean>标签**



**<bean>标签**

**以下是bean比较常用的属性**

**①id——必须手动配置，唯一标识名，用于getBean()**

**②class——必须手动配置，全限定名，反射时使用**

**③scope——对象的多用范围**

**——singleton（默认）单例**

**——prototype 多例**

**——request Spring创建Bean对象的时候，会把Bean对象放入request域**

**——seesion Spring创建Bean对象的时候，会把Bean对象放入session域**

**——global session**

**④init-method——可选属性，用于对象初始化——对象创建后自动调用该属性指定方法**

**⑤destory-method——可选属性，用于对象销毁——对象销毁前自动调用**

**2，Bean对象生命周期**

**①bean被设置为单例模式**

**对象创建：创建Spring容器时，创建bean对象，只创建一次**

**对象运行：Spring容器运行，那么对象就一直存在**

**对象销毁：Spring容器销毁or应用卸载**

**②bean被设置为多例模式**

**对象创建：创建Spring容器时，不会创建；每调用一次getBean()就创建一个对象**

**对象运行：只要对象在使用中，对象就一直活着**

**对象销毁：当对象不被使用时，被GC销毁**

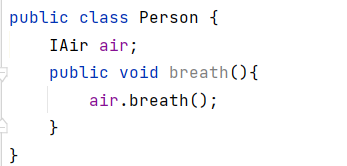
**！！如果bean中指定了init-method，那么每创建一个对象后就会执行一次初始化方法**

**！！如果bean中指定了destory-method，那么每个对象销毁前都会执行一次销毁方法**

**三，依赖注入(Dependency Injection)——IOC的具体体现**

**依赖：类A的某个功能的实现，需要类B的实例对象——A依赖B**

**例如：人要完成breath()，需要Air对象—Person依赖Air**



**常规的开发中，我们有2种方法实现这个依赖**

**①set方法——setAir()传入IAir对象**

**②构造方法传入IAir对象**

**但这样做，Person和IAir对象的耦合度太高**

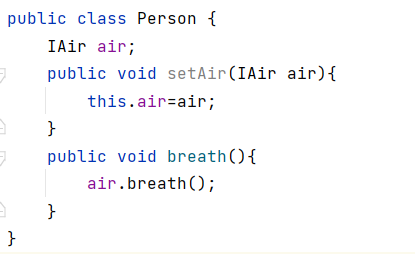
**所以我们可以通过 Application.xml来实现依赖注入,降低Person和IAir的耦合度**

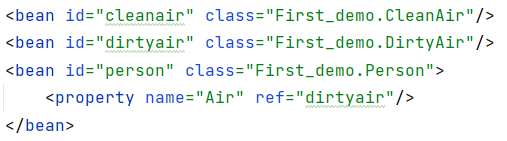
**——我们在xml文件中设置后，使用Person对象时，Spring会自动将IAir对象传递给Person对象，不需要我们去获取——也就是让Spring帮我们维护依赖关系**

**依赖注入的两种实现方法**

**!!!如果注入的是Bean对象，必须保证Bean对象在Spring容器中**

**①set方法——<property>标签为Person的set方法注入对象**

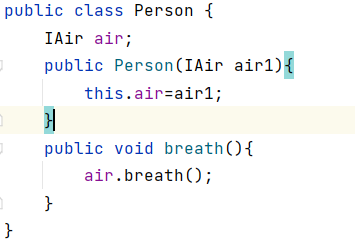


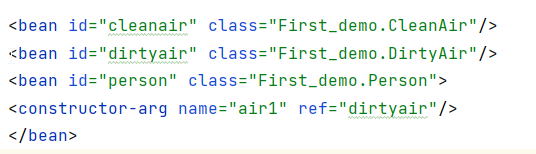


**<property>标签——name：要注入的目标方法名-set**

**——ref：向目标方法注入的对象的id**

**②构造方法——<construcrot-arg>标签为Person的构造方法注入对象**





**<constructor-arg>标签——name：构造方法中，传入参数的名字**

**——ref：要向构造方法注入的对象的id**

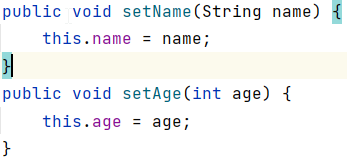
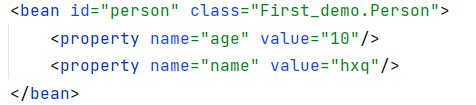
**四，依赖注入的数据类型**

**依赖注入的数据类型—Bean对象，普通数据类型，集合，数组都可以**

**———也是通过上述两种方法注入，只不过有点差别罢了**

**———！！！注入Bean对象时，必须保证Bean对象存在于Spring容器中**

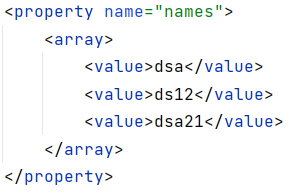
**①注入普通数据类型**

**②注入集合**



**③注入数组**



**五，配置文件中引入其他application.xml**

**通过<import>标签**

**<import resoure=”applicationContext-XXX.xm”/>**