1. **Spring框架**
2. 什么是Spring框架？它跟Spring MVC有什么关系？

答：Spring框架是一个开源的企业应用开发框架，包含20多个不同的模块，是一个轻量级

的解决方案。Spring MVC是Spring的一个子框架，依赖于Spring core和Spring

Bean模块，这两个模块为依赖注入解决方案提供了支持。

1. 什么叫依赖注入？有哪几种形式？

答：简单地说就是一个类A中的方法需要用到类B，那么则称A依赖于B。

一般情况下，在A中要使用类B的方法（非静态）时，就需要创建一个B的对象，如

果B只是一个类则可以直接new对象即可。关键是，很多时候为了增强代码的复用性，

A依赖的B往往是一个接口，并且对B接口又有很多不同的实现类，如果在A中用B

时直接采用创建B的一个实现类，就会提高代码的耦合度，假如某个时候需求变更或者

其他情况需要使用B的另一种实现时，就需要改代码，这降低了A的可重用性。

依赖注入就是用来实现上面描述情况的解耦，接管对象的创建工作，并将该对象的引用

注入需要该对象的组件。

依赖注入的形式，一种是通过将用到的组件设置为成员，并且新增对应的setter，这样

框架就会去调用它，以注入一个B的实例；另一种是采用构造器的方式注入。

3、Spring框架实现依赖注入的具体方式？

答：对于框架来说，注入依赖的方式主要有XML或注解两种配置方式。

XML方式就是在XML文件中的bean标签中预设组件对象创建成员值，然后依赖这个组

件的程序创建一个控制反转容器去解析XML，然后使用getBean方法来获得组件对象，

假如某个组件有多种实现，那么在XML中引用实现的类即可，这样子对程序来说，就

实现了接口具体实现类的动态引用，有变动时也只需要修改XML文件即可。

注解的方式首先需要引入依赖包，才能使用注解，然后定义一个接口包，内置接口类，

接着创建一个实现包，内置接口实现类，最后在程序中将接口作为一个类成员，无需

显式创建对象，程序运行时框架自动注入一个接口依赖对象。

从目前来看，XML方式更加流行，因为它不像使用注解一样需要在代码中引入注解包，

从而产生了依赖，这使得程序无法直接迁移到另一个依赖注入容器。

4、Spring容器与bean什么关系？

答：在Spring中，容器就像一个大管家，管理程序运行时需要的各种资源和依赖，这些资

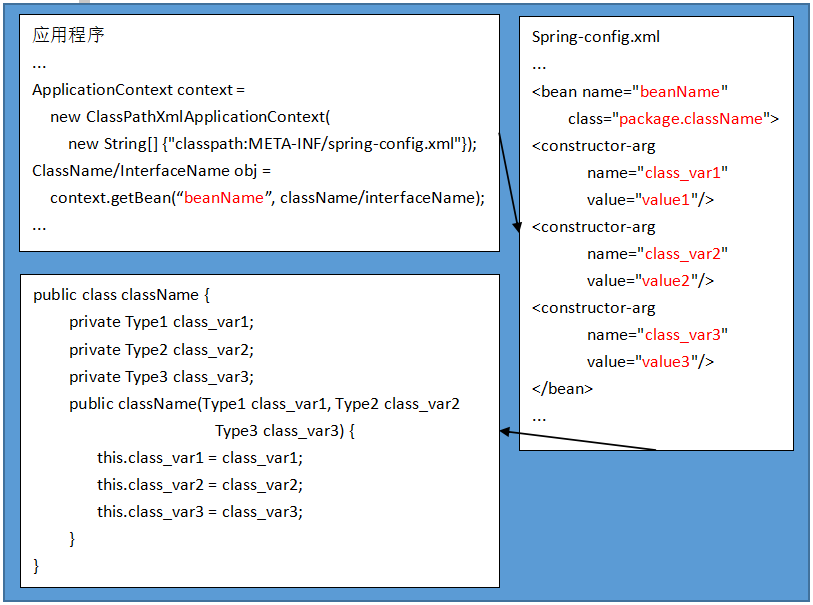
源和依赖就是一个个beans对象，在xml文件中表现为一个个bean标签。当程序运行

需要用到这些资源时，就会先创建一个容器，去读取xml配置文件，再获取一个个依赖

和资源对象，将程序运行和资源依赖引用分离开有好处，就是实现了控制反转的效果，

本来是容器操控这些资源和依赖对象，但通过xml配置文件，间接变成你在控制，这就

控制反转的含义。



1. XML配置文件详解
2. 基本结构

Spring基于XML配置，虽然平时经常在用，但我相信很多人都不知道XML文件中很多标签和元素代表什么意思，下面是一个最基本的XML空文件。



第一行是一个XML元素，表示这是一个XML文件，版本是1.0，编码是UTF-8。

接下来是一个beans元素，它是xml文件的根节点。

**xmlns**属性表示xml命名空间，因为XML文件的标签名称都是自定义的，自己写的和其

他人定义的标签很有可能会重复命名，而功能却不一样，所以需要加上一个namespace

来区分这个xml文件和其他的xml文件，类似于java中的package。

**xmlns:xsi**是指xml文件遵守xml规范，xsi全名：xml schema instance，是指具体用到的

schema资源文件里定义的元素所准守的规范。即/spring-beans.xsd这个文件里定义的元

素遵守什么标准。

**xsi:schemaLocation**是指，本文档里的xml元素所遵守的规范，schemaLocation 属性用

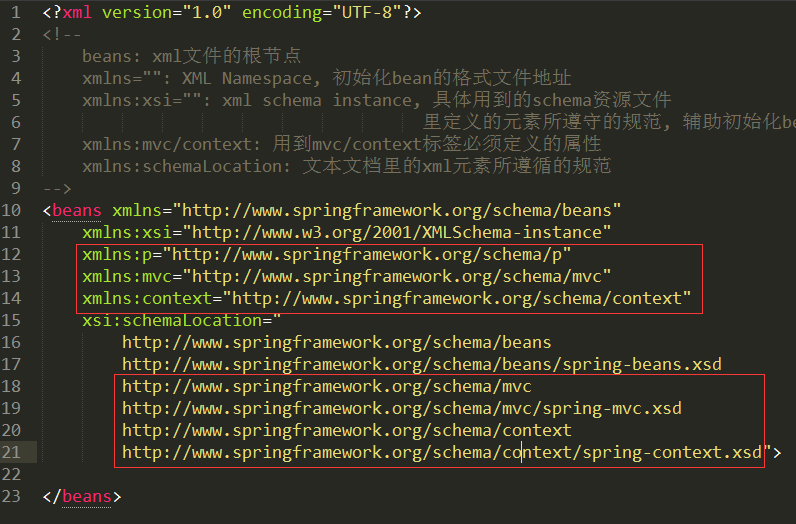
来引用(schema)模式文档，解析器可以在需要的情况下使用这个文档对 XML 实例文档

进行校验。它的值(URI)是成对出现的，第一个值表示**命名空间**，第二个值则表示描述

**该命名空间的模式文档的具体位置**，两个值之间以空格分隔。

1. 引入标签和规范

上面几个属于xml文件中最基本的东西，一般情况下要某个框架的标签定义xml，就要声明标签元素规范、命名空间和具体位置，如下图所示：



1. 导入其他配置文件

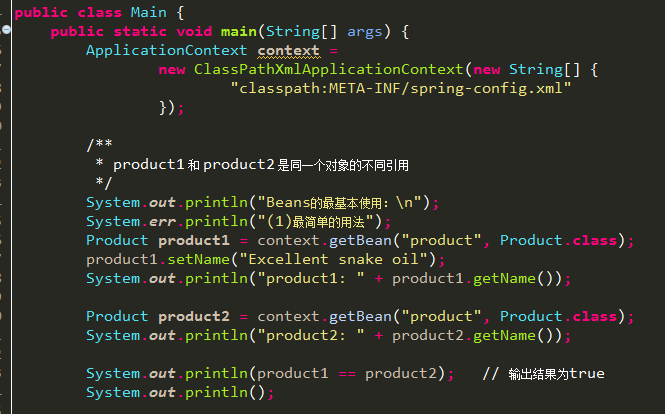


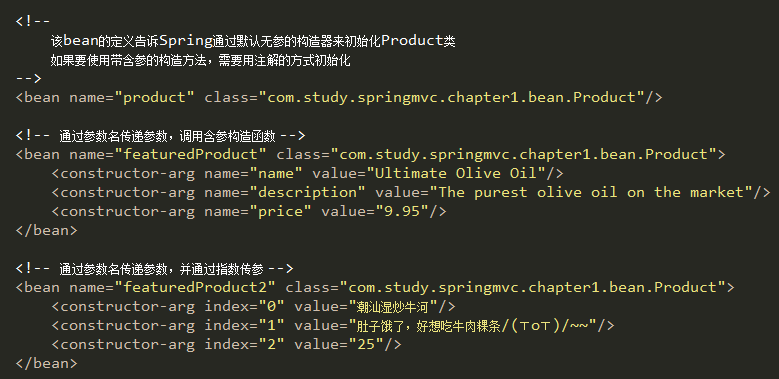
1. Spring控制反转容器的使用

在xml中的bean元素必备id/name属性，用来唯一标识bean，还有class属性（完整报名加类名），用来表示具体实例化的类。

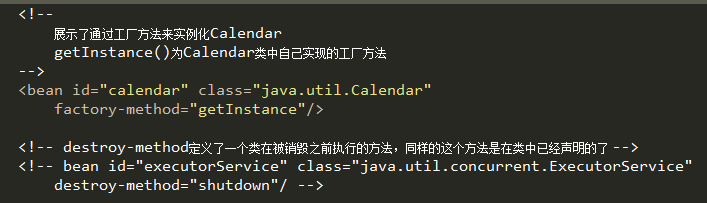
<bean id=”identify” class=”package.classname”/>

1. 通过构造器创建一个bean实例

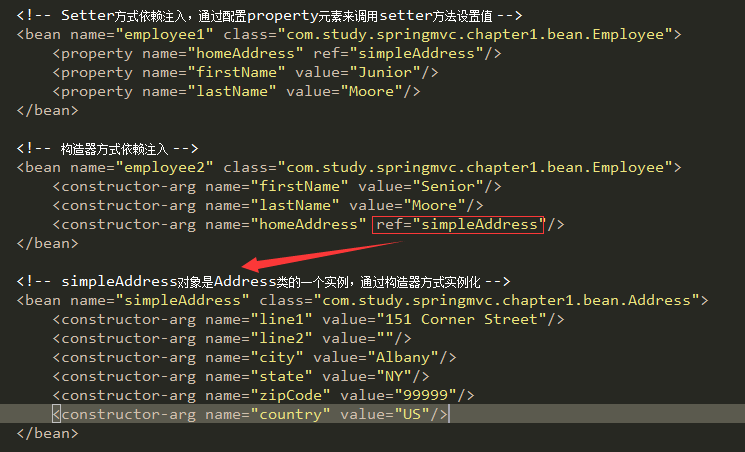




1. 通过工厂方法创建一个bean实例



1. 构造器方式和settre方式依赖注入对比



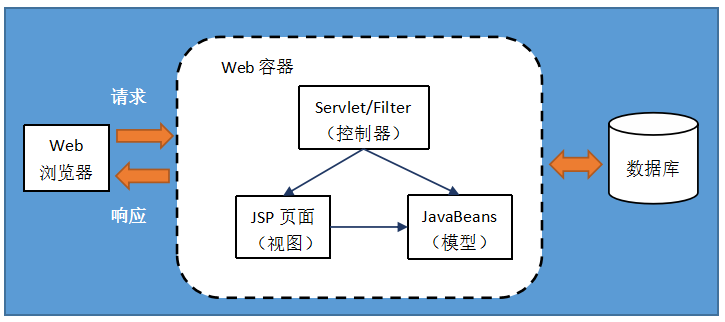
1. **Java Web两种设计模型**
2. 两种模型简介

第一种模型是以页面为中心，通过链接方式进行JSP页面间的跳转，这种方式比较直接，在大中型应用中会带来维护上的问题，因为单一个JSP页面的名字被修改时可能导致大量页面中的链接需要修改。

第二种模型是MVC模式，就是模型、视图和控制器啦，模型负责封装数据和业务逻辑，视图负责展示应用界面，控制器对用户输入进行接收、使用模型以及调整视图的展示。

1. Java Web MVC模式

架构图如下图所示：



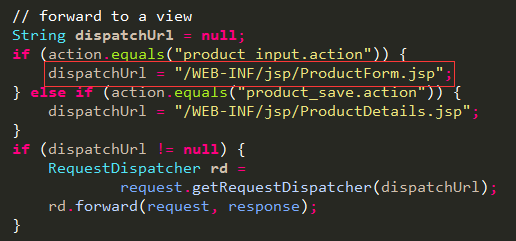
1. 使用Servlet控制器的请求响应逻辑
2. 部署描述符解析URI



如上图所示，根节点是web-app元素，属性跟spring-config.xml大同小异，servlet-mapping标签表示请求url跟处理请求的servlet程序名的映射关系，servlet元素表示处理请求的servlet具体程序和对它的命名。这段配置的意思就是，对于url后缀是.action的请求，交给servlet-name为ControlleServlet的控制器处理，这个控制器的位置是app02a.servlet.ControllerServlet。

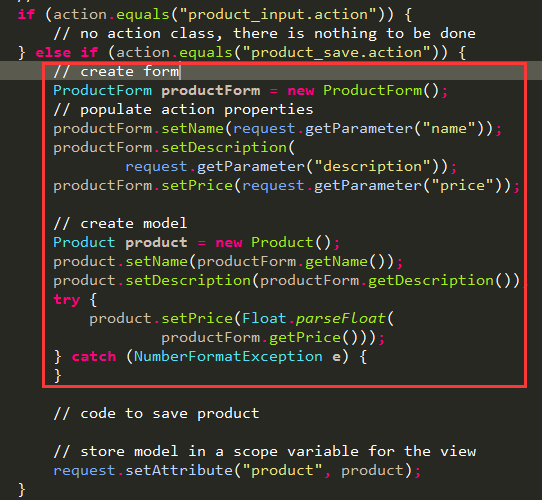
1. Servlet控制器处理

控制器中会对请求的具体类型进行解析，知道请求目的才能做接下来操作，比如有些请求只是页面请求，这种情况下servlet程序只需要跳转到jsp页面即可。

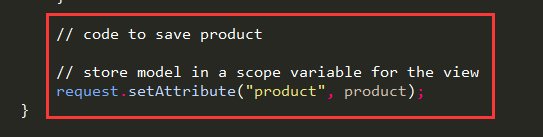


1. 数据操作与持久化

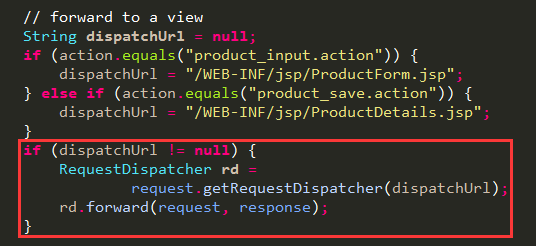
大多数请求都是带着数据来的，那么就要对这些数据进行处理，必要时持久化处理结果，并返回处理结果到显示对应的页面去。



1. 返回数据模型



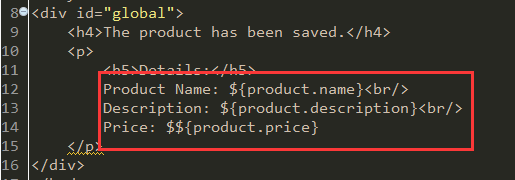
1. 跳转JSP页面



1. 视图
2. 表单提交

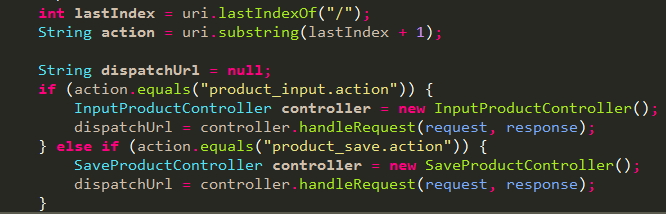


1. 提交结果展示



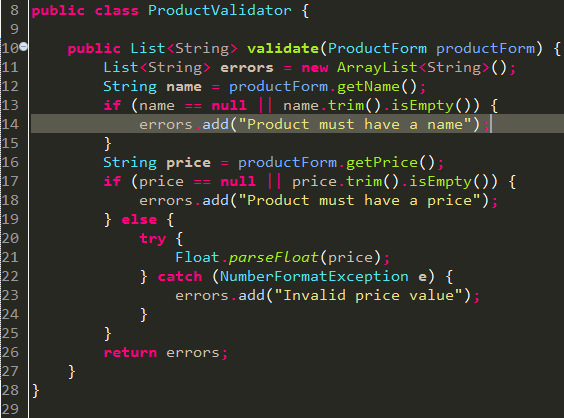
1. 解耦设计

让servlet专门做路由解析和服务调用的工作，具体的请求和数据处理独立出来。

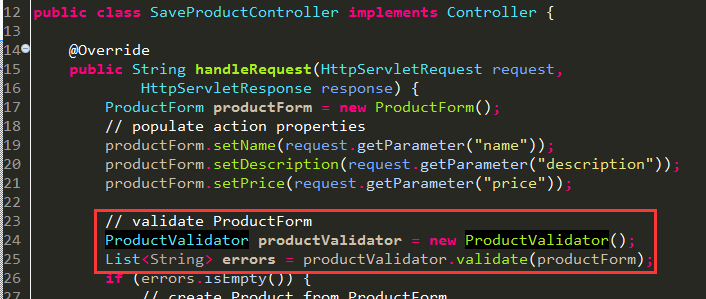


1. 校验器

（1）校验器制作



（2）在处理业务逻辑的控制器中的使用



1. **Spring MVC**
2. 设计一个MVC框架的Dispatcher servlet必须具备什么特性？

答：从名字上看，dispatcher servlet就是后台调度程序的意思，所有的客户端请求都要经过

调度程序才能执行请求响应程序，这样说来，调度程序就必须能识别各种各样的客户端

请求，并调用正确的响应程序，所以**必备特性之一**就是**根据URI调用相应的action**；根

据请求调用相应的action，就必须**实例化正确的控制器类**；请求往往是带着参数来的，

所以还要**根据请求参数来构造表单bean**；一个控制器中往往有多个方法，能响应同类

不同请求，所以还要**能够执行正确的方法**；对请求进行处理后，往往还要返回请求响应

操作结果到客户端，**跳转到一个新的JSP页面**。

1. Spring MVC提供了哪些加快开发的特性？

答：（1）内置Dispatcher Servlet，无需额外开发

（2）使用基于XML的配置文件，依赖配置与使用隔离，配置变动时无需重新编译代码

（3）实例化控制器，根据用户输入构造bean

所以不会看到new xxController()的代码，框架自动完成

（4）自动绑定用户输入，并转换为正确的类型，无需显式将数字字符串转换为相应类

型

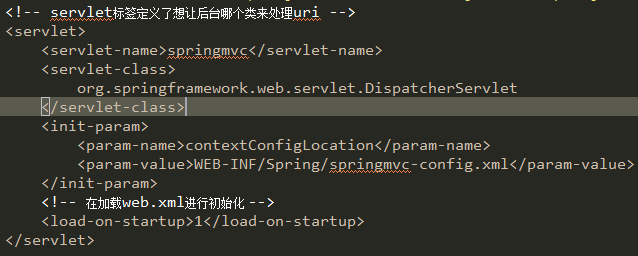
（5）可以校验用户输入，若输入不通过，则重定向回输入表单

框架内置了校验器，对一些常见的输入进行校验

（6）执行使用Sping的其他组件；国际化多语言支持；多种页面技术

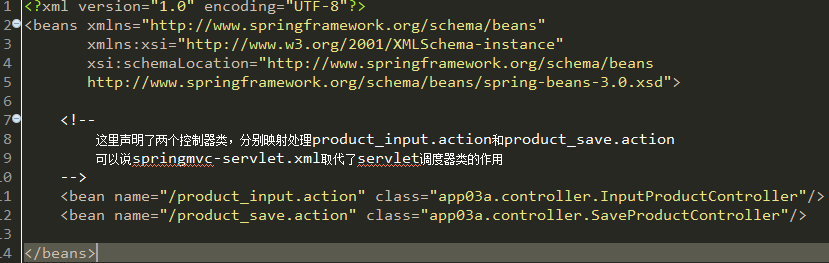
3、Spring MVC的Dispatcher Servlet

Spring MVC内置了Dispatcher Servlet，开箱即用，只要将web.xml（部署描述符）中的servlet-class元素的值设置为org.springframework.web.servlet.DispatcherServlet即可，此外还有一个配置beans的spring xml文件，文件名默认为 **[** servlet-name元素值 **]**-config.xml，文件位置为WEB-INF/，如果想更改默认文件名或位置，在servlet元素内添加init-param元素即可，如下图所示：



1. Spring的Controller接口

在Spring2.5之前，开发一个控制器的唯一方法是实现org.springframework.web.servlet.mvc.Controller接口，而这个接口只有一个方法handleRequest，在Springmvc-config.xml的bean元素中，name表示请求路由，class表示对请求进行处理的控制器类，默认调用的是handleRequest方法，也不能指定方法，所以每个方法只能处理一个请求，如下图所示：



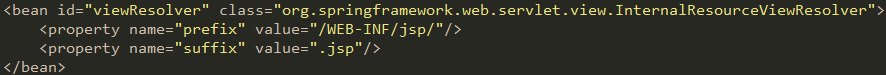
如果使用基于注解的控制器，则可以实现一个控制器相应多个action请求。

1. 视图解析器View Resolver

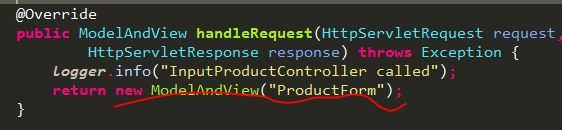
使用视图解析器，能够缩短控制器返回页面时指定路径的长度。

视图解析器用法：

Springmvc-config.xml中配置



Controller中用法



1. **基于注解的控制器**

1、基于注解控制器的优点