# Spring MVC笔记

三层架构：

1、表现层 ：常说的web 层。负责接收客户端请求，向客户端响应结果

表现层包括展示层和控制层。控制层负责接收请求，展示层负责结果的展示。

表现层依赖业务层，接收到客户端请求⼀般会调⽤业务层进⾏业务处理，并将处理结果响应给客户端。MVC 是表现层的设计模型，和其他层没有关系

2、业务层：常说的service 层。负责业务逻辑处理

web 层依赖业务层，但是业务层不依赖web 层。

3、持久层 ：常说的dao 层。负责数据持久化，包括数据层即数据库和数据访问层。

持久层就是和数据库交互，对数据库表进⾏增删改查的。

MVC设计模式：

Model（模型）：包含业务模型和数据模型，数据模型⽤于封装数据，业务模型⽤于处理业务。

View（视图）： 通常指的就是我们的jsp 或者html。作⽤⼀般就是展示数据的。通常视图是依据

模型数据创建的。

Controller（控制器）： 是应⽤程序中处理⽤户交互的部分。作⽤⼀般就是处理程序逻辑的。

MVC提倡：每⼀层只编写⾃⼰的东⻄，不编写任何其他的代码；分层是为了解耦，解耦是为了维

护⽅便和分⼯协作。

请求参数绑定：

参数绑定：取出参数值绑定到handler⽅法的形参上

可支持简单数据类型：⼋种基本数据类型及其包装类型

Restful基础知识：

基本定义：Restful 是⼀种web 软件架构⻛格，它不是标准也不是协议，它倡导的是⼀个资源定位及资源操作的⻛格。

它本身并没有什么实⽤性，其核⼼价值在于如何设计出符合REST ⻛格的⽹络接⼝。

优点：结构清晰、符合标准、易于理解、扩展⽅便，所以正得到越来越多⽹站的采⽤。

GET ⽤来获取资源，POST ⽤来新建资源，PUT ⽤来更新资源，DELETE ⽤来删除资源。

Ajax Json交互：

1）前端到后台：前端ajax发送json格式字符串，后台直接接收为pojo参数，使⽤注解@RequstBody

2）后台到前端：后台直接返回pojo对象，前端直接接收为json对象或者字符串，使⽤注解

@ResponseBody

@responseBody注解的作⽤是将controller的⽅法返回的对象通过适当的转换器转换为指定的格式之

后，写⼊到response对象的body区，通常⽤来返回JSON数据或者是XML数据。

拦截器基本概念：

Servlet：处理Request请求和Response响应

过滤器（Filter）：对Request请求起到过滤的作⽤，作⽤在Servlet之前

监听器（Listener）：随Web应⽤的启动⽽启动，只初始化⼀次，然后会⼀直运⾏监视，随Web应⽤的停⽌⽽销毁。1、初始化⼯作 2、监听web中的特定事件

拦截器（Interceptor）：只会拦截访问的控制器⽅法（Handler）

serlvet、filter、listener是配置在web.xml中的，⽽interceptor是配置在表现层框架⾃⼰的配置⽂件中

1、在Handler业务逻辑执⾏之前拦截⼀次

2、在Handler逻辑执⾏完毕但未跳转⻚⾯之前拦截⼀次

3、在跳转⻚⾯之后拦截⼀次

Mybatis整合Spring:

1、数据库连接池以及事务管理都交给Spring容器来完成

2、SqlSessionFactory对象应该放到Spring容器中作为单例对象管理

3、Mapper动态代理对象交给Spring管理，我们从Spring容器中直接获得Mapper的代理对象

Spring Data JPA：

概念：是Spring 基于JPA 规范的基础上封装的⼀套JPA 应⽤框架

作用：Spring Data JPA 极⼤简化了数据访问层代码

构建步骤：

1、创建⼯程导⼊坐标（Java框架于我们⽽⾔就是⼀堆jar）

2、配置Spring 的配置⽂件（配置指定框架执⾏的细节）

3、编写实体类Resume，使⽤JPA 注解配置映射关系

4、编写⼀个符合Spring Data JPA 的Dao 层接⼝（ResumeDao接⼝）