**Ssm项目整合**

# 新建maven项目

...

# 配置本地Tomcat。保证项目能正常运行

# 在web的pom.xml中加入依赖

包含了大部分常用jar

*<!-- 集中定义依赖版本号 -->*<**properties**>  
 <**junit.version**>4.12</**junit.version**>  
 <**spring.version**>4.1.3.RELEASE</**spring.version**>  
 <**mybatis.version**>3.2.8</**mybatis.version**>  
 <**mybatis.spring.version**>1.2.2</**mybatis.spring.version**>  
 <**mybatis.paginator.version**>1.2.15</**mybatis.paginator.version**>  
 <**mysql.version**>5.1.32</**mysql.version**>  
 <**slf4j.version**>1.6.4</**slf4j.version**>  
 <**jackson.version**>2.4.2</**jackson.version**>  
 <**druid.version**>1.0.9</**druid.version**>  
 <**httpclient.version**>4.3.5</**httpclient.version**>  
 <**jstl.version**>1.2</**jstl.version**>  
 <**servlet-api.version**>2.5</**servlet-api.version**>  
 <**jsp-api.version**>2.0</**jsp-api.version**>  
 <**joda-time.version**>2.5</**joda-time.version**>  
 <**commons-lang3.version**>3.3.2</**commons-lang3.version**>  
 <**commons-io.version**>1.3.2</**commons-io.version**>  
 <**commons-net.version**>3.3</**commons-net.version**>  
 <**pagehelper.version**>5.1.2</**pagehelper.version**>  
 <**jsqlparser.version**>0.9.1</**jsqlparser.version**>  
 <**commons-fileupload.version**>1.3.1</**commons-fileupload.version**>  
 <**jedis.version**>2.7.2</**jedis.version**>  
 <**solrj.version**>4.10.3</**solrj.version**>  
</**properties**>  
 <**dependencies**>  
 *<!-- 时间操作组件 -->* <**dependency**>  
 <**groupId**>joda-time</**groupId**>  
 <**artifactId**>joda-time</**artifactId**>  
 <**version**>${joda-time.version}</**version**>  
 </**dependency**>  
 *<!-- Apache工具组件 -->* <**dependency**>  
 <**groupId**>org.apache.commons</**groupId**>  
 <**artifactId**>commons-lang3</**artifactId**>  
 <**version**>${commons-lang3.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.apache.commons</**groupId**>  
 <**artifactId**>commons-io</**artifactId**>  
 <**version**>${commons-io.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>commons-net</**groupId**>  
 <**artifactId**>commons-net</**artifactId**>  
 <**version**>${commons-net.version}</**version**>  
 </**dependency**>  
 *<!-- Jackson Json处理工具包 -->* <**dependency**>  
 <**groupId**>com.fasterxml.jackson.core</**groupId**>  
 <**artifactId**>jackson-databind</**artifactId**>  
 <**version**>${jackson.version}</**version**>  
 </**dependency**>  
 *<!-- httpclient -->* <**dependency**>  
 <**groupId**>org.apache.httpcomponents</**groupId**>  
 <**artifactId**>httpclient</**artifactId**>  
 <**version**>${httpclient.version}</**version**>  
 </**dependency**>  
 *<!-- 单元测试 -->* <**dependency**>  
 <**groupId**>junit</**groupId**>  
 <**artifactId**>junit</**artifactId**>  
 <**version**>${junit.version}</**version**>  
 <**scope**>test</**scope**>  
 </**dependency**>  
 *<!-- 日志处理 -->* <**dependency**>  
 <**groupId**>org.slf4j</**groupId**>  
 <**artifactId**>slf4j-log4j12</**artifactId**>  
 <**version**>${slf4j.version}</**version**>  
 </**dependency**>  
 *<!-- Mybatis -->* <**dependency**>  
 <**groupId**>org.mybatis</**groupId**>  
 <**artifactId**>mybatis</**artifactId**>  
 <**version**>${mybatis.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.mybatis</**groupId**>  
 <**artifactId**>mybatis-spring</**artifactId**>  
 <**version**>${mybatis.spring.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>com.github.miemiedev</**groupId**>  
 <**artifactId**>mybatis-paginator</**artifactId**>  
 <**version**>${mybatis.paginator.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>com.github.pagehelper</**groupId**>  
 <**artifactId**>pagehelper</**artifactId**>  
 <**version**>${pagehelper.version}</**version**>  
 </**dependency**>  
 *<!-- MySql -->* <**dependency**>  
 <**groupId**>mysql</**groupId**>  
 <**artifactId**>mysql-connector-java</**artifactId**>  
 <**version**>${mysql.version}</**version**>  
 </**dependency**>  
 *<!-- 连接池 -->* <**dependency**>  
 <**groupId**>com.alibaba</**groupId**>  
 <**artifactId**>druid</**artifactId**>  
 <**version**>${druid.version}</**version**>  
 </**dependency**>  
 *<!-- Spring -->* <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-context</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-beans</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-webmvc</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-jdbc</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-aspects</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 *<!-- JSP相关 -->* <**dependency**>  
 <**groupId**>jstl</**groupId**>  
 <**artifactId**>jstl</**artifactId**>  
 <**version**>${jstl.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>javax.servlet</**groupId**>  
 <**artifactId**>servlet-api</**artifactId**>  
 <**version**>${servlet-api.version}</**version**>  
 <**scope**>provided</**scope**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>javax.servlet</**groupId**>  
 <**artifactId**>jsp-api</**artifactId**>  
 <**version**>${jsp-api.version}</**version**>  
 <**scope**>provided</**scope**>  
 </**dependency**>  
 *<!-- 文件上传组件 -->* <**dependency**>  
 <**groupId**>commons-fileupload</**groupId**>  
 <**artifactId**>commons-fileupload</**artifactId**>  
 <**version**>${commons-fileupload.version}</**version**>  
 </**dependency**>  
 *<!-- Redis客户端 -->* <**dependency**>  
 <**groupId**>redis.clients</**groupId**>  
 <**artifactId**>jedis</**artifactId**>  
 <**version**>${jedis.version}</**version**>  
 </**dependency**>  
 *<!-- solr客户端 -->* <**dependency**>  
 <**groupId**>org.apache.solr</**groupId**>  
 <**artifactId**>solr-solrj</**artifactId**>  
 <**version**>${solrj.version}</**version**>  
 </**dependency**>  
 </**dependencies**>

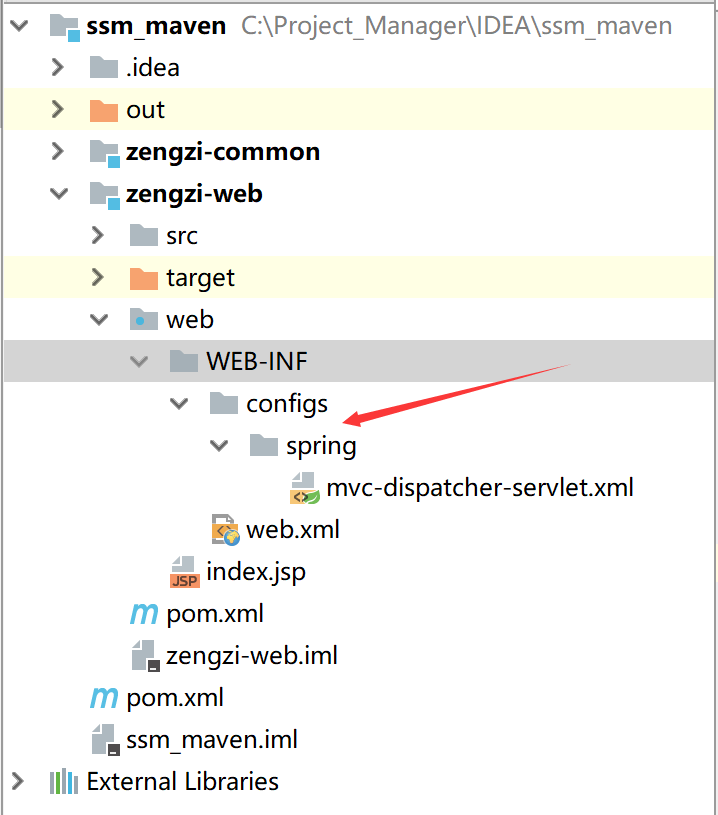
# 在web.xml中配置servlet

## 4.1 servlet的配置

*<!--配置springMVC的DispatcherServlet-->*<**servlet**>  
 <**servlet-name**>mvc-dispatcher</**servlet-name**>  
 <**servlet-class**>org.springframework.web.servlet.DispatcherServlet</**servlet-class**>  
 *<!--DispatcherServlet对应的上下文配置，默认为/WEB-INF/$servlet-name$-servlet.xml-->* <**init-param**>  
 <**param-name**>contextConfigLocation</**param-name**>  
 <**param-value**>/WEB-INF/configs/spring/mvc-dispatcher-servlet.xml</**param-value**>  
 </**init-param**>  
 *<!--应用程序启动时装载servlet并调用他的init方法-->* <**load-on-startup**>1</**load-on-startup**>  
</**servlet**>  
<**servlet-mapping**>  
 <**servlet-name**>mvc-dispatcher</**servlet-name**>  
 *<!--拦截所有请求-->* <**url-pattern**>/</**url-pattern**>  
</**servlet-mapping**>  
*<!--配置springMVC的DispatcherServlet end-->*

## 4.2 上下文配置xml的创建及目录结构

*<!--DispatcherServlet对应的上下文配置，默认为/WEB-INF/$servlet-name$-servlet.xml-->*<**init-param**>  
 <**param-name**>contextConfigLocation</**param-name**>  
 <**param-value**>/WEB-INF/configs/spring/mvc-dispatcher-servlet.xml</**param-value**>  
</**init-param**>



# mvc-dispatcher-servlet.xml编写

*<?***xml version="1.0" encoding="UTF-8"***?>*<**beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xmlns:mvc="http://www.springframework.org/schema/mvc"  
 xsi:schemaLocation="http://www.springframework.org/schema/beans  
 http://www.springframework.org/schema/beans/spring-beans.xsd http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context.xsd http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc/spring-mvc.xsd"**>  
  
 *<!-- 启用Spring基于annotation的DI, 使用户可以在Spring MVC中使用Spring的强大功能。 激活 @Required  
 @Autowired,JSR 250's @PostConstruct, @PreDestroy and @Resource 等标注 -->* <**context:annotation-config**/>  
  
 *<!-- DispatcherServlet上下文， 只管理@Controller类型的bean， 忽略其他型的bean, 如@Service -->* <**context:component-scan base-package="com.zeng.admin.controller"**>  
 <**context:include-filter type="annotation"  
 expression="org.springframework.stereotype.Controller"**/>  
 </**context:component-scan**>  
  
 *<!-- HandlerMapping, 无需配置， Spring MVC可以默认启动。 DefaultAnnotationHandlerMapping  
 annotation-driven HandlerMapping -->  
  
 <!-- 扩充了注解驱动，可以将请求参数绑定到控制器参数 -->* <**mvc:annotation-driven**/>  
  
 *<!-- 配置ViewResolver视图解析器。 可以用多个ViewResolver。 使用order属性排序。 InternalResourceViewResolver放在最后。 -->* <**bean  
 class="org.springframework.web.servlet.view.InternalResourceViewResolver"**>  
 <**property name="viewClass"  
 value="org.springframework.web.servlet.view.JstlView"**/>  
 *<!--存放jsp的文件夹-->* <**property name="prefix" value="/WEB-INF/jsps/"**/>  
 <**property name="suffix" value=".jsp"**/>  
 </**bean**>  
</**beans**>

# 理解springMVC上下文与spring上下文

SpringMVC管理controlller

spring不用管理controlller



# Mybatis与sptring集成