Web.xml

*<?***xml version="1.0" encoding="UTF-8"***?>*<**web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app\_3\_1.xsd"  
 version="3.1"**>  
  
 <**listener**>  
 <**description**>Spring容器加载监听器</**description**>  
 <**listener-class**>org.springframework.web.context.ContextLoaderListener</**listener-class**>  
 </**listener**>  
 <**context-param**>  
 <**param-name**>contextConfigLocation</**param-name**>  
 <**param-value**>classpath:applicationContext.xml</**param-value**>  
 </**context-param**>  
  
 *<!--Spring MVC 前置Servlet，中心控制器 -->* <**servlet**>  
 <**servlet-name**>springmvc</**servlet-name**>  
 <**servlet-class**>org.springframework.web.servlet.DispatcherServlet</**servlet-class**>  
 <**init-param**>  
 <**param-name**>contextConfigLocation</**param-name**>  
 *<!--Spring MVC配置文件路径 -->* <**param-value**>classpath:springmvc-servlet.xml</**param-value**>  
 </**init-param**>  
 *<!-- 启动动优先级，越小越早加载 -->* <**load-on-startup**>1</**load-on-startup**>  
 *<!--Servlet3.0以上文件上传配置 -->* <**multipart-config**>  
 *<!--上传文件的最大限制5MB -->* <**max-file-size**>5242880</**max-file-size**>  
 *<!--请求的最大限制20MB -->* <**max-request-size**>20971520</**max-request-size**>  
 *<!--当文件的大小超过临界值时将写入磁盘 -->* <**file-size-threshold**>0</**file-size-threshold**>  
 </**multipart-config**>  
 </**servlet**>  
 *<!-- Servlet访问的路径映射，所有的访问都必须经过调度用的前置控制品 -->* <**servlet-mapping**>  
 <**servlet-name**>springmvc</**servlet-name**>  
 <**url-pattern**>/</**url-pattern**>  
 </**servlet-mapping**>  
  
 *<!--编码过滤器 -->* <**filter**>  
 <**filter-name**>characterEncodingFilter</**filter-name**>  
 <**filter-class**>org.springframework.web.filter.CharacterEncodingFilter</**filter-class**>  
 <**init-param**>  
 <**param-name**>encoding</**param-name**>  
 <**param-value**>UTF-8</**param-value**>  
 </**init-param**>  
 <**init-param**>  
 <**param-name**>forceEncoding</**param-name**>  
 <**param-value**>true</**param-value**>  
 </**init-param**>  
 </**filter**>  
 *<!-- 路径映射 -->* <**filter-mapping**>  
 <**filter-name**>characterEncodingFilter</**filter-name**>  
 <**url-pattern**>/\*</**url-pattern**>  
 </**filter-mapping**>

注意：编码过滤器最好放在头上，因为如果再有一个过滤器的话，可能会导致这个玩意失效，记住，filter的执行顺序是按照filter-mapping的顺序来执行的

Pom.xml

*<?***xml version="1.0" encoding="UTF-8"***?>*<**project xmlns="http://maven.apache.org/POM/4.0.0"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"**>  
 <**modelVersion**>4.0.0</**modelVersion**>  
  
 <**groupId**>com.hua</**groupId**>  
 <**artifactId**>hello</**artifactId**>  
 <**version**>1.0-SNAPSHOT</**version**>  
  
  
 <**properties**>  
 <**project.build.sourceEncoding**>UTF-8</**project.build.sourceEncoding**>  
 <**spring.version**>4.3.0.RELEASE</**spring.version**>  
 </**properties**>  
  
 <**dependencies**>  
  
 *<!--mysql驱动包-->* <**dependency**>  
 <**groupId**>mysql</**groupId**>  
 <**artifactId**>mysql-connector-java</**artifactId**>  
 <**version**>5.1.35</**version**>  
 </**dependency**>  
  
 *<!--Spring框架核心库 -->* <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-context</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 *<!-- Spring MVC -->* <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-webmvc</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-context-support</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 *<!-- aspectJ AOP 织入器 -->* <**dependency**>  
 <**groupId**>org.aspectj</**groupId**>  
 <**artifactId**>aspectjweaver</**artifactId**>  
 <**version**>1.8.9</**version**>  
 </**dependency**>  
 *<!--mybatis-spring适配器 -->* <**dependency**>  
 <**groupId**>org.mybatis</**groupId**>  
 <**artifactId**>mybatis-spring</**artifactId**>  
 <**version**>1.3.0</**version**>  
 </**dependency**>  
 *<!--Spring java数据库访问包，在本例中主要用于提供数据源 -->* <**dependency**>  
 <**groupId**>org.springframework</**groupId**>  
 <**artifactId**>spring-jdbc</**artifactId**>  
 <**version**>${spring.version}</**version**>  
 </**dependency**>  
 *<!--log4j日志包 -->* <**dependency**>  
 <**groupId**>org.apache.logging.log4j</**groupId**>  
 <**artifactId**>log4j-core</**artifactId**>  
 <**version**>2.6.1</**version**>  
 </**dependency**>  
 *<!-- mybatis ORM框架 -->* <**dependency**>  
 <**groupId**>org.mybatis</**groupId**>  
 <**artifactId**>mybatis</**artifactId**>  
 <**version**>3.4.1</**version**>  
 </**dependency**>  
 *<!-- JUnit单元测试工具 -->* <**dependency**>  
 <**groupId**>junit</**groupId**>  
 <**artifactId**>junit</**artifactId**>  
 <**version**>4.10</**version**>  
 </**dependency**>  
 *<!--c3p0 连接池 -->* <**dependency**>  
 <**groupId**>c3p0</**groupId**>  
 <**artifactId**>c3p0</**artifactId**>  
 <**version**>0.9.1.2</**version**>  
 </**dependency**>  
 *<!-- JSTL -->* <**dependency**>  
 <**groupId**>javax.servlet</**groupId**>  
 <**artifactId**>jstl</**artifactId**>  
 <**version**>1.2</**version**>  
 </**dependency**>  
 *<!-- Servlet核心包 -->* <**dependency**>  
 <**groupId**>javax.servlet</**groupId**>  
 <**artifactId**>javax.servlet-api</**artifactId**>  
 <**version**>3.0.1</**version**>  
 <**scope**>provided</**scope**>  
 </**dependency**>  
 *<!--JSP -->* <**dependency**>  
 <**groupId**>javax.servlet.jsp</**groupId**>  
 <**artifactId**>jsp-api</**artifactId**>  
 <**version**>2.1</**version**>  
 <**scope**>provided</**scope**>  
 </**dependency**>  
  
 </**dependencies**>  
  
</**project**>

applicationContext.xml

*<?***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:p="http://www.springframework.org/schema/p"  
 xmlns:aop="http://www.springframework.org/schema/aop"  
 xmlns:context="http://www.springframework.org/schema/context"  
 xmlns:tx="http://www.springframework.org/schema/tx"  
 xsi:schemaLocation="http://www.springframework.org/schema/beans  
 http://www.springframework.org/schema/beans/spring-beans-4.3.xsd  
 http://www.springframework.org/schema/context  
 http://www.springframework.org/schema/context/spring-context-4.3.xsd  
 http://www.springframework.org/schema/aop  
 http://www.springframework.org/schema/aop/spring-aop-4.3.xsd  
 http://www.springframework.org/schema/tx  
 http://www.springframework.org/schema/tx/spring-tx-4.3.xsd"**>  
  
 *<!--1 引入属性文件，在配置中占位使用 -->* <**context:property-placeholder location="classpath\*:db.properties"** />  
  
 *<!--2 配置C3P0数据源 -->* <**bean id="datasource" class="com.mchange.v2.c3p0.ComboPooledDataSource" destroy-method="close"**>  
 *<!--驱动类名 -->* <**property name="driverClass" value="${jdbc.driverClassName}"** />  
 *<!-- url -->* <**property name="jdbcUrl" value="${jdbc.url}"** />  
 *<!-- 用户名 -->* <**property name="user" value="${jdbc.username}"** />  
 *<!-- 密码 -->* <**property name="password" value="${jdbc.password}"** />  
  
  
 </**bean**>  
  
 *<!--3 会话工厂bean sqlSessionFactoryBean -->* <**bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean"**>  
 *<!-- 配置文件路径 -->* <**property name="configLocation" value="classpath:MybatisCfg.xml"**/>  
 *<!-- 数据源 -->* <**property name="dataSource" ref="datasource"**/>  
 *<!-- sql映射文件路径 -->* <**property name="mapperLocations" value="classpath\*:mapper/\*Mapper.xml"**/>  
 </**bean**>  
  
 *<!--4 自动扫描对象关系映射 -->* <**bean class="org.mybatis.spring.mapper.MapperScannerConfigurer"**>  
 *<!--指定会话工厂，如果当前上下文中只定义了一个则该属性可省去 -->* <**property name="sqlSessionFactoryBeanName" value="sqlSessionFactory"**/>  
 *<!-- 指定要自动扫描接口的基础包，实现接口 -->* <**property name="basePackage" value="com.hua.dao"**/>  
 </**bean**>  
  
 *<!--5 声明式事务管理 -->  
 <!--定义事物管理器，由spring管理事务 -->* <**bean id="transactionManager"  
 class="org.springframework.jdbc.datasource.DataSourceTransactionManager"**>  
 <**property name="dataSource" ref="datasource"**/>  
 </**bean**>  
 *<!--支持注解驱动的事务管理，指定事务管理器 -->* <**tx:annotation-driven transaction-manager="transactionManager"** />  
  
 *<!--6 容器自动扫描IOC组件 -->* <**context:component-scan base-package="com.hua"**/>  
  
 *<!--7 aspectj支持自动代理实现AOP功能 -->* <**aop:aspectj-autoproxy proxy-target-class="true"**/>  
  
</**beans**>

Springmvc-servlet.xml

*<?***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-4.3.xsd  
 http://www.springframework.org/schema/mvc  
 http://www.springframework.org/schema/mvc/spring-mvc-4.3.xsd"**>  
 *<!-- 自动扫描包，实现支持注解的IOC -->* <**context:component-scan base-package="com.hua"** />  
  
 *<!-- Spring MVC不处理静态资源 -->* <**mvc:default-servlet-handler** />  
  
 *<!-- 支持mvc注解驱动 -->* <**mvc:annotation-driven enable-matrix-variables="true"** />  
  
 *<!-- 配置映射媒体类型的策略 -->* <**bean  
 class="org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerMapping"**>  
 <**property name="removeSemicolonContent" value="false"** />  
 </**bean**>  
  
  
 *<!-- 内部视图解析器，JSP与JSTL模板 -->* <**bean class="org.springframework.web.servlet.view.InternalResourceViewResolver"  
 id="internalResourceViewResolver"**>  
 *<!--指定视图渲染类 -->* <**property name="viewClass"  
 value="org.springframework.web.servlet.view.JstlView"** />  
 *<!--自动添加到路径中的前缀 -->* <**property name="prefix" value="/WEB-INF/views/"** />  
 *<!--自动添加到路径中的后缀 -->* <**property name="suffix" value=".jsp"** />  
 *<!--设置所有视图的内容类型，如果视图本身设置内容类型视图类可以忽略 -->* <**property name="contentType" value="text/html;charset=UTF-8"** />  
 *<!-- 优先级，越小越前 -->* <**property name="order" value="1"** />  
 </**bean**>  
  
 *<!--文件上传解析器 -->  
 <!--Spring MVC默认不能识别multipart格式的文件内容 -->* <**bean id="multipartResolver"  
 class="org.springframework.web.multipart.support.StandardServletMultipartResolver"**>  
 </**bean**>  
  
</**beans**>

MybatisCfg.xml

*<?***xml version="1.0" encoding="UTF-8"***?>***<!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN" "http://mybatis.org/dtd/mybatis-3-config.dtd"*>***<**configuration**>  
 <**properties resource="db.properties"**/>  
 <**settings**>  
 *<!--指定mybatis使用日志组件 -->* <**setting name="logImpl" value="LOG4J2"** />  
 *<!--开启全局的二级缓存 -->* <**setting name="cacheEnabled" value="false"** />  
 *<!--开启延时加载，如果有关联关系，则默认不会获取数据  
 延迟加载的全局开关。当开启时，所有关联对象都会延迟加载。  
 特定关联关系中可通过设置fetchType属性来覆盖该项的开关状态。  
 在association中指定fetchType="eager(立即)" 或者 lazy(延迟)  
 默认：false  
 -->* <**setting name="lazyLoadingEnabled" value="true"** />  
 *<!--true时，对任意延迟属性的调用会使带有延迟加载属性的对象完整加载；  
 false，每种属性将会按需加载。  
 默认为：true-->* <**setting name="aggressiveLazyLoading" value="false"** />  
 </**settings**>  
 <**typeAliases**>  
 <**package name="com.hua.entities"** />  
 </**typeAliases**>  
</**configuration**>

Db.properties

**mysql.user**=**root  
mysql.password**=**171911  
mysql.url**=**jdbc:mysql://localhost:3306/educ  
mysql.driverClass**=**com.mysql.jdbc.Driver**