# 创建SpringMvc项目

Maven下载安装：<https://www.cnblogs.com/eagle6688/p/7838224.html>

Idea下载安装：<https://jingyan.baidu.com/article/25648fc16db2909191fd0082.html>

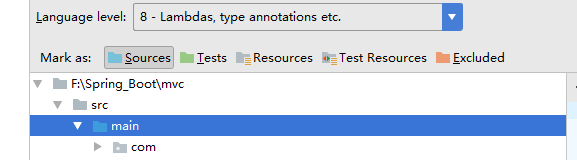
<http://www.jb51.net/softjc/598538.html>

git的简单教程：https://www.cnblogs.com/schaepher/p/5561193.html

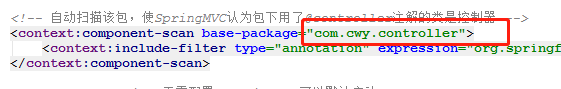
IDEA ,Maven ,Tomcat

避坑：

1. IDEA，目录需要点击Sources，不然目录下的文件找不到创建class的入口



但如果Sources定位到具体包，扫描包这里会报错



## ApplicationContext.xml及相关xml配置

1. 头部文件配置问题

缺少红色部分导致bean注入失败！！！（applicationContext.xml）

<**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"**>  
 *<!--<?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"  
 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-3.1.xsd  
 http://www.springframework.org/schema/tx  
 http://www.springframework.org/schema/tx/spring-tx.xsd">-->*

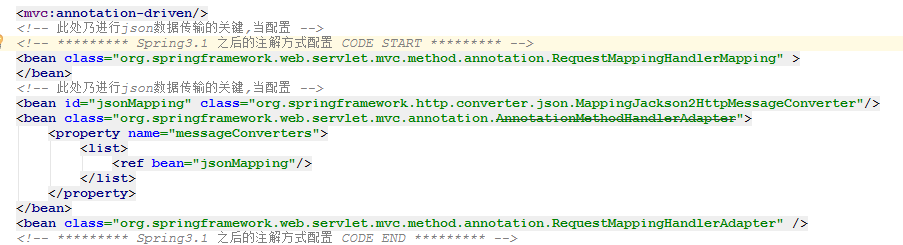
### 2.json中文乱码

添加下面json中文乱码解决时报错，Line 49 in XML document from class path resource [dispatcher-servlet.xml] is invalid; nested exception is org.xml.sax.SAXParseException; lineNumber: 49; columnNumber: 29; cvc-complex-type.2.4.c: 通配符的匹配很全面, 但无法找到元素 'mvc:annotation-driven' 的声明。（dispatcher-servlet.xml）



1[[1]](#footnote-1)

1.上面的代码依旧报错，修改代码



依赖添加

<**dependency**>  
 <**groupId**>com.fasterxml.jackson.core</**groupId**>  
 <**artifactId**>jackson-annotations</**artifactId**>  
 <**version**>2.7.5</**version**>  
</**dependency**>  
  
<**dependency**>  
 <**groupId**>com.fasterxml.jackson.core</**groupId**>  
 <**artifactId**>jackson-core</**artifactId**>  
 <**version**>2.7.5</**version**>  
</**dependency**>  
<**dependency**>  
 <**groupId**>com.fasterxml.jackson.core</**groupId**>  
 <**artifactId**>jackson-databind</**artifactId**>  
 <**version**>2.7.5</**version**>  
</**dependency**>

2.fastJson处理

*<!—pom.xml添加fastjson -->*<**dependency**>  
 <**groupId**>com.alibaba</**groupId**>  
 <**artifactId**>fastjson</**artifactId**>  
 <**version**>1.2.7</**version**>  
</**dependency**>

*<!—- dispatcher-servlet.xml添加下面配置-->*

<**mvc:annotation-driven**>  
 <**mvc:message-converters register-defaults="true"**>  
 *<!-- fastjosn spring support -->* <**bean id="jsonConverter" class="com.cwy.utils.MappingFastJsonHttpMessageConverter"**>  
 <**property name="supportedMediaTypes"**>  
 <**list**>  
 <**value**>text/html;charset=UTF-8</**value**>  
 <**value**>application/json</**value**>  
 </**list**>  
 </**property**>  
 <**property name="serializerFeature"**>  
 <**list**>  
 <**value**>WriteMapNullValue</**value**>  
 <**value**>QuoteFieldNames</**value**>  
 </**list**>  
 </**property**></**bean**>  
 </**mvc:message-converters**>  
</**mvc:annotation-driven**>

*<!—- 添加json转化的类-->*

**package** com.cwy.utils;  
  
**import** com.alibaba.fastjson.JSON;  
**import** com.alibaba.fastjson.serializer.SerializerFeature;  
**import** org.springframework.http.HttpInputMessage;  
**import** org.springframework.http.HttpOutputMessage;  
**import** org.springframework.http.MediaType;  
**import** org.springframework.http.converter.AbstractHttpMessageConverter;  
**import** org.springframework.http.converter.HttpMessageNotReadableException;  
**import** org.springframework.http.converter.HttpMessageNotWritableException;  
  
**import** java.io.ByteArrayOutputStream;  
**import** java.io.IOException;  
**import** java.io.OutputStream;  
**import** java.nio.charset.Charset;  
  
*/\*\*  
 \** ***@Description*** *：  
 \** ***@Created*** *by 蔡文粤  
 \** ***@date*** *2018/4/13 14:48  
 \*/***public class** MappingFastJsonHttpMessageConverter **extends** AbstractHttpMessageConverter<Object> {  
  
 **public static final** Charset ***DEFAULT\_CHARSET*** = Charset.*forName*(**"UTF-8"**);  
  
 **private** SerializerFeature[] **serializerFeature**;  
  
 **public** SerializerFeature[] getSerializerFeature() {  
 **return serializerFeature**;  
 }  
  
 **public void** setSerializerFeature(SerializerFeature[] serializerFeature) {  
 **this**.**serializerFeature** = serializerFeature;  
 }  
  
 **public** MappingFastJsonHttpMessageConverter() {  
 **super**(**new** MediaType(**"application"**, **"json"**, ***DEFAULT\_CHARSET***));  
 }  
  
 @Override  
 **public boolean** canRead(Class<?> clazz, MediaType mediaType) {  
 **return true**;  
 }  
  
 @Override  
 **public boolean** canWrite(Class<?> clazz, MediaType mediaType) {  
 **return true**;  
 }  
  
 @Override  
 **protected boolean** supports(Class<?> aClass) {  
 **throw new** UnsupportedOperationException();  
 }  
  
 @Override  
 **protected** Object readInternal(Class<?> aClass, HttpInputMessage httpInputMessage) **throws** IOException, HttpMessageNotReadableException {  
 ByteArrayOutputStream baos = **new** ByteArrayOutputStream();  
 **int** i;  
 **while** ((i = httpInputMessage.getBody().read()) != -1) {  
 baos.write(i);  
 }  
 **return** JSON.*parseArray*(baos.toString(), aClass);  
 }  
  
 @Override  
 **protected void** writeInternal(Object o, HttpOutputMessage httpOutputMessage) **throws** IOException, HttpMessageNotWritableException {  
 String jsonString = JSON.*toJSONString*(o, **serializerFeature**);  
 OutputStream out = httpOutputMessage.getBody();  
 out.write(jsonString.getBytes(***DEFAULT\_CHARSET***));  
 out.flush();  
 }  
}

# 数据库

遇到字符集编码问题，中文乱码，每一次创建新数据库时选择utf-8

1.找到mysql配置文件my-default.ini添加一下字段

[client]

default\_character\_set=utf8

[mysql]

default\_character\_set = utf8

[mysqld]

character\_set\_server = utf8

collation-server=utf8\_general\_ci

1. 找到mysql配置文件my-default.ini复制一份，重命名为my.ini，底下要有上面新添的字段

1. [↑](#footnote-ref-1)