# 整合jsp

整合各大框架

搭微服务比较方便

springboot!=ssm 这个是不等的，只能说搭建这些框架比较方便

springboot!=微服务 也是不等的。只能说用它来搭建微服务比较方便

默认情况是返回页面视图，但是默认没有配置。

还不知道用什么视图

视图解决器没有，现在页面找不到

整合了很都资源，servlet都切入embed-core中了

tomcat-embed-jasper

已经把servlet jsp el 全部加入了

jstl标签库

支持jstl 加入jar

springboot官方也推荐使用jsp了

# 整合thymeleaf

下载jar 3个

配置模板方案，还是模板引擎，视图解决方案

springboot-start-thymeleaf

官方推荐 th

静态资源的控制

包括css js img 等等

放在模板中的页面是动态的，即使是html也是动态处理的。

都放在一个文件夹static

还有一个文件夹也可以

resources

xml解析方式两种

1.dom方式解析

把一个文档前部载入到dom内存节点中，然后对这个节点进行遍历查询，速度比较快的，但是耗内存

1G的文档，一次需要开销1G内存。只要读取其中的一小部分，配置文件。小文档的解析

2.sax方式解析

基于节点驱动。你要找什么什么节点，。它就只加载这个当前节点，读取节点中的内容

大文件的的解析，性能比较低。直接操作硬盘的，如果读取重复的数据，那么也要重新读取硬盘中的数据

没有框架的时候，我们写一些封装类

数据库的配置写在properties中的。

Propertie类去读取的。

load

如果这个配置文件是xml 的话，那么就要用xml解析了。

domj4 jdom 都可以，他们是第三方的解析xml的jar包

前面整合了两个视图功能

jsp thymeleaf

# 整合mybatis

ssm 中整合的话

1.书写mybatis.cfg.xml 总的配置文件

2.spring配置文件，数据源，spring-mybatis jar

接口扫描，整合的数据源

3.mapper映射文件

导入 mybatis.jar mybatis-spring.jar

spring-boot-starter-框架的名字

数据库的基本配置信息，包括，驱动，连接数据库，用户名，密码等

项目的架构方式。

事务操作

# 整合redis

导入jar包

主要的对对象操作比较麻烦

序列化和反序列化

json格式