

## 一、整合Freemarker模版引擎视图层

### 1、引入依赖

```
<dependency>

    <groupId>org.springframework.boot</groupId>

    <artifactId>spring-boot-starter-freemarker</artifactId>

</dependency>
```

2、在src/main/resources/目录下创建**templates**目录，在该目录下创建 **.ftl** 视图模板文件

### 3、创建Controller类:

- 1、类的注解只能是@Controller 表示方法返回为视图
- 2、方法的返回为String类型的视图文件名
- 3、启动项目访问@RequestMapping("/xxxx") localhost:8081/index

```
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;

@Controller
public class indexFTLconn {

    @RequestMapping("/index")
    public String get(){
        return "index";    //返回视图
    }

}
```

## 二、SpringBoot整合jsp视图层

注意：创建**SpringBoot**整合**jsp**，一定要为**War**类型，否则项目会找不到**jsp**文件

### 1、引入依赖有两个

```
<dependency>

    <groupId>org.springframework.boot</groupId>

    <artifactId>spring-boot-starter-tomcat</artifactId>

</dependency>
```

```
<dependency>

    <groupId>org.apache.tomcat.embed</groupId>

    <artifactId>tomcat-embed-jasper</artifactId>

</dependency>
```

## 2、application.properties文件设置

#jsp 支持

**spring.mvc.view.suffix=.jsp**

**spring.mvc.view.prefix=/WEB-INF/jsp/**

#关闭默认模板引擎

spring.thymeleaf.cache=false

spring.thymeleaf.enabled=false

## 3、将jsp页面放入到src/main/webapp/WEB-INF/jsp/xxxx.jsp

## 4、创建Controller类：

- 1、类的注解只能是@Controller 表示方法返回为视图
- 2、方法的返回为String类型的视图文件名
- 3、启动项目访问@RequestMapping("/xxxx") localhost:8081/index

```
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;

@Controller
public class indexFTLconn {

    @RequestMapping("/index")
    public String get(){
        return "index"; //index.jsp 返回视图
    }

}
```