

Springboot 学习笔记

北京金网安泰信息技术有限公司

密级：绝密□机密□秘密□公开■

作者：刘振兴

版本号：1.0

创建时间：2019年10月25日

**修改记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 修改日期 | 修改标记 | 修改人 | 审核 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

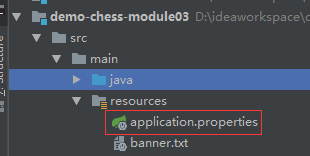
说明

demo-chess-module03 是在 demo-chess-module02 基础上做的二次开发

修改WEB容器配置

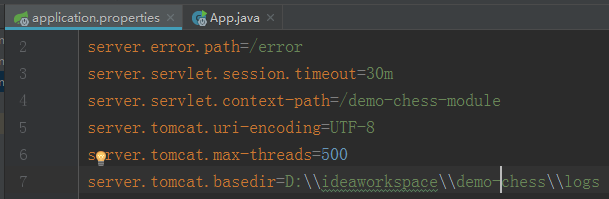
* 1. 设置容器基本配置

在 resource 目录下创建 application.properties 文件



在其中增加 WEB 容器的配置

* server.port：配置 WEB 容器端口
* server.error.path：配置项目出错时跳转的指定页面
* server.servlet.session.timeout：session 失效时间，默认单位为秒， 30m 表示 30 分
* server.servlet.context-path：项目名称，默认为 /
* server.tomcat.uri-encoding：tomcat 请求编码格式
* server.tomcat.max-threads：设置tomcat支持的最大线程数
* server.tomcat.basedir：存放tomcat 运行日志和临时文件的目录，不配置则使用系统临时目录。
* 其他配置待研究



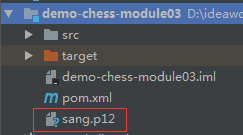
* 1. 设置https
     1. 通过 jdk 生成证书

|  |
| --- |
| keytool -genkey -alias tomcathttps -keyalg RSA -keysize 2048 -keystore sang.p12 -validity 360 |

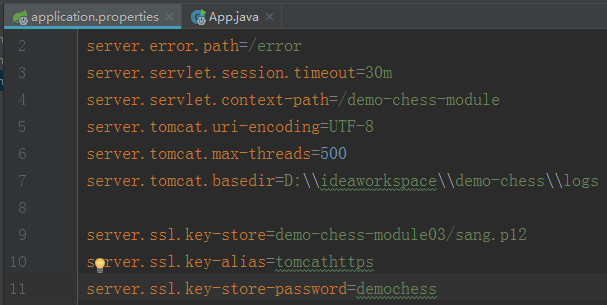
* -genkey 创建证书
* -alias 设置证书别名，上边代码设置证书别名为 tomcathttps
* -keyalg 设置证书的加密算法，上边代码设置为 RSA 算法
* -keysize 设置密钥长度
* -keystore 设置密钥存放位置，上边代码生成了 sang.p12 存放在了当前目录
* -validity 证书有效期，上边代码设置了有效期为 365 天。



* + 1. 将证书设置到工程中
       1. 将证书放到项目的根目录下



* + - 1. 在application.properties 中设置证书信息
* server.ssl.key-store 设置证书位置
* server.ssl.key-alias 设置证书别名
* server.ssl.key-store-password 设置证书密码



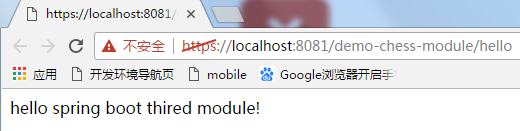
* + - 1. 启动并访问测试

访问：<https://localhost:8081/demo-chess-module/hello>

因为自己生成的证书不被浏览器承认，所以会提示不安全

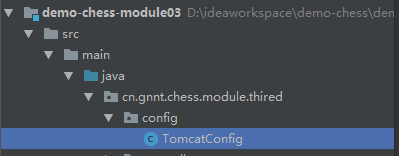


点击继续前往即可



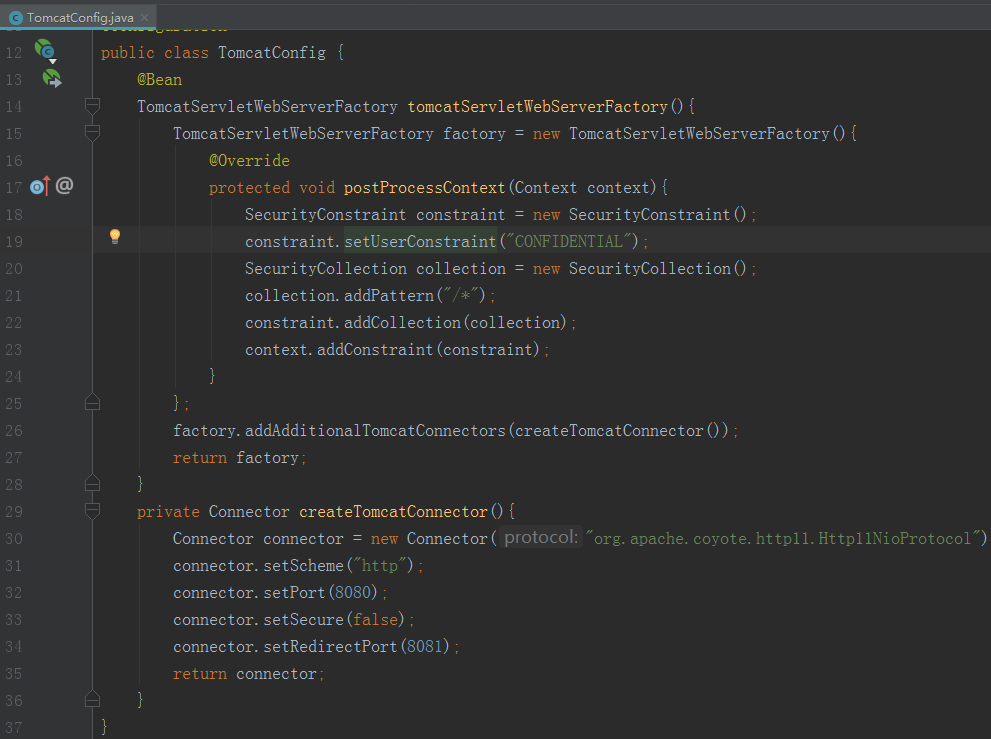
* + 1. http重定向https

在 config 下创建 TomcatConfig.java 文件



代码如下

|  |
| --- |
| @Configuration public class TomcatConfig {  @Bean  TomcatServletWebServerFactory tomcatServletWebServerFactory(){  TomcatServletWebServerFactory factory = new TomcatServletWebServerFactory(){  @Override  protected void postProcessContext(Context context){  SecurityConstraint constraint = new SecurityConstraint();  constraint.setUserConstraint("CONFIDENTIAL");  SecurityCollection collection = new SecurityCollection();  collection.addPattern("/\*");  constraint.addCollection(collection);  context.addConstraint(constraint);  }  };  factory.addAdditionalTomcatConnectors(createTomcatConnector());  return factory;  }  private Connector createTomcatConnector(){  Connector connector = new Connector("org.apache.coyote.http11.Http11NioProtocol");  connector.setScheme("http");  connector.setPort(8080);  connector.setSecure(false);  connector.setRedirectPort(8081);  return connector;  } } |



重启服务，浏览器输入 <http://localhost:8080/demo-chess-module/hello>

自动重新链接到 <https://localhost:8081/demo-chess-module/hello>