集成SpringMVC+Swagger

swagger是restful管理项目API工具

【1】pom.xml增加依赖包

```xml
<!-- swagger-mvc -->
<dependency>
    <groupId>io.springfox</groupId>
    <artifactId>springfox-swagger2</artifactId>
    <version>2.4.0</version>
</dependency>
<dependency>
    <groupId>io.springfox</groupId>
    <artifactId>springfox-swagger-ui</artifactId>
    <version>2.4.0</version>
</dependency>
<!-- json -->
<dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-core</artifactId>
    <version>2.6.5</version>
</dependency>
<dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-databind</artifactId>
    <version>2.6.5</version>
</dependency>
<dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-annotations</artifactId>
    <version>2.6.5</version>
</dependency>
```

【2】Swagger配置文件

```java
import org.springframework.context.annotation.Bean;
    import org.springframework.context.annotation.ComponentScan;
    import org.springframework.context.annotation.Configuration;
    import org.springframework.web.servlet.config.annotation.EnableWebMvc;
    import springfox.documentation.builders.ApiInfoBuilder;
    import springfox.documentation.builders.PathSelectors;
```

```
import springfox.documentation.builders.RequestHandlerSelectors;
import springfox.documentation.service.ApiInfo;
import springfox.documentation.service.Contact;
import springfox.documentation.spi.DocumentationType;
import springfox.documentation.spring.web.plugins.Docket;
import springfox.documentation.swagger2.annotations.EnableSwagger2;

/**
 * Created with wyh.
 * Date: 2017/7/3
 * Time: 下午2:23
 */

@EnableWebMvc
@EnableSwagger2//启用Swagger2
@Configuration//让Spring来加载该类配置
@ComponentScan(basePackages ="com.wyh2020.fstore")
public class SwaggerConfig {
    @Bean
    public Docket buildDocket() {
        return new Docket(DocumentationType.SWAGGER_2)
                .apiInfo(buildApiInf())
                .select()

.apis(RequestHandlerSelectors.basePackage("com.wyh2020.fstore.controller")
)//controller路径
                .paths(PathSelectors.any())
                .build();
    }

    private ApiInfo buildApiInf() {
        return new ApiInfoBuilder()
                .title("SpringMVC中使用Swagger2构建RESTful APIs")
                .termsOfServiceUrl("https://wyh2020/openapi/")
                .description("")
                .contact(new Contact("wyh2020", "https://wyh2020.com/",
"wangyonghua@qianmi.com"))
                .version("1.0")
                .build();

    }
}
```

【3】Controller中使用注解添加API文档

```
import com.wyh2020.fstore.pojo.Test;
```

```java
import com.wyh2020.fstore.service.TestService;
import io.swagger.annotations.Api;
import io.swagger.annotations.ApiOperation;
import org.apache.log4j.Logger;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RestController;

import java.util.List;

/**
 * Created with wyh.
 * Date: 2017/7/2
 * Time: 上午12:47
 */
@RestController
@RequestMapping("/test")
@Api(value = "TestController", description = "测试接口")
public class TestController {

    @Autowired
    private TestService testService;


    Logger logger = Logger.getLogger(TestController.class);


    @ApiOperation(value = "查询Test列表", notes = "查询Test列表",
httpMethod = "GET")
    @RequestMapping(value = "/queryList", method = {RequestMethod.GET,
RequestMethod.POST})
    public String getTestList(){
        logger.info("进入接口啦啦啦啦啦啦");
        List<Test> testList = testService.getTestList();

        return testList.toString();
    }

}
```

【4】在spring-mvc.xml中声明swagger配置bean

```xml
<bean
class="springfox.documentation.swagger2.configuration.Swagger2Documentatio
nConfiguration" id="swagger2Config"/>
```

【5】在spring-mvc.xml中配置资源文件

```xml
<mvc:resources location="classpath:/META-INF/resources/" mapping="swagger-ui.html"/>
<mvc:resources location="classpath:/META-INF/resources/webjars/" mapping="/webjars/**"/>
```

【6】通过以上配置启动项目访问即可看到

```
localhost:8080/swagger-ui.html
```