

SpringBoot非官方教程 | 第十篇： 用spring Restdocs创建API文档

转载请标明出处：

<http://blog.csdn.net/forezp/article/details/71023510>

本文出自[方志朋的博客](#)

这篇文章将带你了解如何用spring官方推荐的restdoc去生成api文档。本文创建一个简单的springboot工程，将http接口通过Api文档暴露出来。只需要通过 JUnit单元测试和Spring的MockMVC就可以生成文档。

准备工作

- 你需要15min
- Jdk 1.8
- maven 3.0+
- idea

创建工程

引入依赖，其pom文件：

```
1 <dependencies>
2     <dependency>
3         <groupId>org.springframework.boot</groupId>
4         <artifactId>spring-boot-starter-web</artifactId>
5     </dependency>
6
7     <dependency>
8         <groupId>org.springframework.boot</groupId>
9         <artifactId>spring-boot-starter-test</artifactId>
10        <scope>test</scope>
11    </dependency>
12
13    <dependency>
```

```
14         <groupId>org.springframework.restdocs</groupId>
15         <artifactId>spring-restdocs-mockmvc</artifactId>
16         <scope>test</scope>
17     </dependency>
18 </dependencies>
19 1
20 2
21 3
22 4
23 5
24 6
25 7
26 8
27 9
28 10
29 11
30 12
31 13
32 14
33 15
34 16
35 17
36 18
```

通过@SpringBootApplication,开启springboot

```
1 @SpringBootApplication
2 public class Application {
3
4     public static void main(String[] args) {
5         SpringApplication.run(Application.class, args);
6     }
7 }
8 1
9 2
10 3
11 4
12 5
13 6
14 7
15 8
```

在springboot通常创建一个controller:

```
1 @RestController
2 public class HomeController {
3
4     @GetMapping("/")
5     public Map<String, Object> greeting() {
6         return Collections.singletonMap("message", "Hello World");
7     }
8
9 }
10 1
11 2
12 3
13 4
14 5
15 6
16 7
17 8
18 9
```

启动工程，访问localhost:8080，浏览器显示：

```
{ "message" : " Hello World" }
```

证明接口已经写好了，但是如何通过restdoc生存api文档呢

Restdoc,通过单元测试生成api文档

restdocs是通过单元测试生存snippets文件，然后snippets根据插件生成htm文档的。

建一个单元测试类：

```
1 @RunWith(SpringRunner.class)
2 @WebMvcTest(HomeController.class)
```

```

3  @AutoConfigureRestDocs(outputDir = "target/snippets")
4  public class WebLayerTest {
5
6      @Autowired
7      private MockMvc mockMvc;
8
9      @Test
10     public void shouldReturnDefaultMessage() throws Exception {
11
12         this.mockMvc.perform(get("/")).andDo(print()).andExpect(status().isOk())
13             .andExpect(content().string(containsString("Hello World")))
14             .andDo(document("home"));
15     }
16 }
17 1
18 2
19 3
20 4
21 5
22 6
23 7
24 8
25 9
26 10
27 11
28 12
29 13
30 14
31 15
32 16

```

其中，@ AutoConfigureRestDocs注解开启了生成snippets文件，并指定了存放位置。

启动单元测试，测试通过，你会发现在target文件下生成了一个snippets文件夹，其目录结构如下：

```

1  └─ target
2      └─ snippets
3          └─ home
4              └─ httpie-request.adoc
5              └─ curl-request.adoc

```

```
6      └─ http-request.adoc
7      └─ http-response.adoc
8  1
9  2
10 3
11 4
12 5
13 6
14 7
15 8
```

默认情况下，snippets是Asciidoctor格式的文件，包括request和response，另外其他两种httpie和curl两种流行的命令行的http请求模式。

到目前为止，只生成了Snippets文件，需要用Snippets文件生成文档。

怎么用Snippets

创建一个新文件src/main/asciidoc/index.adoc：

```
1  = 用 Spring REST Docs 构建文档
2
3  This is an example output for a service running at http://localhost:8080:
4
5  .request
6  include::{snippets}/home/http-request.adoc[]
7
8  .response
9  include::{snippets}/home/http-response.adoc[]
10
11  这个例子非常简单，通过单元测试和一些简单的配置就能够得到api文档了。
12  1
13  2
14  3
15  4
16  5
17  6
18  7
19  8
20  9
21  10
```

```
22 11
23 12
```

adoc的书写格式，参考：<http://docs.spring.io/spring-restdocs/docs/current/reference/html5/>，这里不多讲解。

需要使用asciidoctor-maven-plugin插件，在其pom文件加上：

```
1 <plugin>
2   <groupId>org.asciidoctor</groupId>
3   <artifactId>asciidoctor-maven-plugin</artifactId>
4   <executions>
5     <execution>
6       <id>generate-docs</id>
7       <phase>prepare-package</phase>
8       <goals>
9         <goal>process-asciidoc</goal>
10      </goals>
11      <configuration>
12        <sourceDocumentName>index.adoc</sourceDocumentName>
13        <backend>html</backend>
14        <attributes>
15          <snippets>${project.build.directory}/snippets</snippets>
16        </attributes>
17      </configuration>
18    </execution>
19  </executions>
20 </plugin>
21 1
22 2
23 3
24 4
25 5
26 6
27 7
28 8
29 9
30 10
31 11
32 12
33 13
```

```
34 14
35 15
36 16
37 17
38 18
39 19
40 20
41 21
```

这时只需要通过`mvnw package`命令就可以生成文档了。

在`/target/generated-docs`下有个`index.html`，打开这个html,显示如下，界面还算简洁：

用 Spring REST Docs 构建文档

This is an example output for a service running at <http://localhost:8080>:

request

```
GET / HTTP/1.1
Host: localhost:8080
```

response

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Content-Length: 25

{"message":"Hello World"}
```

这个例子非常简单，通过单元测试和一些简单的配置就能够得到api文档了。

结语

通过单元测试，生存adoc文件，再用adoc文件生存html，只需要简单的几步就可以生成一个api文档的html文件，这个html文件你可以通网站发布出去。整个过程很简单，对代码无任何影响

