

16

Swagger API 與文件產生





第十六單元大綱

- 16-1 Spring Swagger doc api應用
- 16-2 描述REST Service文件規範
- 16-3 單元測驗



16-1 Spring Swagger doc api應用

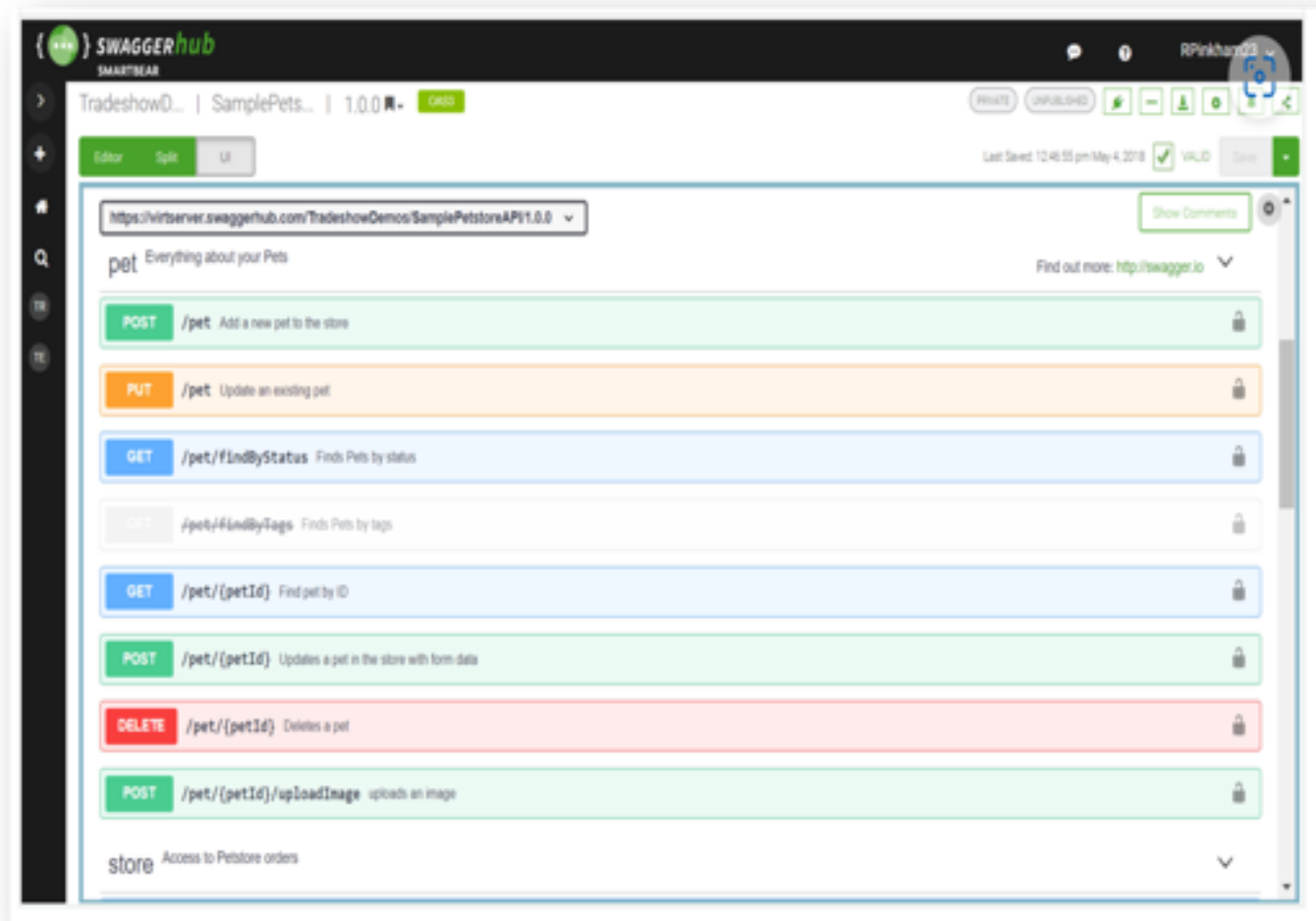


Swagger 是一個工具，可以讓您的API產生文件

透過Swagger UI工具可以產生HTML && CSS
&& Javascript產生網頁版的API文件

您亦可以借助Swagger文件進行API測試

<https://swagger.io/>



Springdoc套件是基於OpenAPI 3.0(Swagger 3)

Spring Boot專案pom.xml架構springdoc

```
<dependency>  
  <groupId>org.springdoc</groupId>  
  <artifactId>springdoc-openapi-ui</artifactId>  
  <version>1.6.6</version>  
</dependency>
```

<http://localhost:8080/v3/api-docs>

```
controller"], "operationId": "retrieve", "parameters": [{"name": "id", "in": "path", "required": true, "schema": {"type": "integer", "format": "int32"}}, {"name": "id", "in": "path", "required": true, "schema": {"type": "integer", "format": "int32"}}], "responses": {"200": {"description": "OK", "content": {"*/": {"schema": {"$ref": "#/components/schemas/Account"}}}}}}, "/remove/{id}": {"delete": {"tags": ["account"], "operationId": "remove", "parameters": [{"name": "id", "in": "path", "required": true, "schema": {"type": "integer", "format": "int32"}}, {"name": "id", "in": "path", "required": true, "schema": {"type": "integer", "format": "int32"}}], "responses": {"200": {"description": "OK", "content": {"*/": {"schema": {"type": "string"}}}}}}}}, "components": {"schemas": {"Customers": {"type": "object", "properties": {"id": {"type": "string"}, "name": {"type": "string"}, "address": {"type": "string"}, "phone": {"type": "string"}, "country": {"type": "string"}}, "Emoji": {"type": "object", "properties": {"index": {"type": "integer", "format": "int32"}, "length": {"type": "integer", "format": "int32"}, "productId": {"type": "string"}, "emojiId": {"type": "string"}}, "Event": {"type": "object", "properties": {"replyTo": {"type": "string"}, "type": {"type": "string"}, "mode": {"type": "string"}, "timestamp": {"type": "integer", "format": "int64"}, "source": {"$ref": "#/components/schemas/Source"}, "message": {"$ref": "#/components/schemas/MyMessage"}}, "Mention": {"type": "object", "properties": {"mentioned": {"type": "array", "items": {"$ref": "#/components/schemas/Mentionee"}}, "Mentionee": {"type": "object", "properties": {"index": {"type": "integer", "format": "int32"}, "length": {"type": "integer", "format": "int32"}, "userId": {"type": "string"}}, "MyMessage": {"type": "object", "properties": {"id": {"type": "string"}, "type": {"type": "string"}, "text": {"type": "string"}, "emojis": {"type": "array", "items": {"$ref": "#/components/schemas/Emoji"}}, "mention": {"$ref": "#/components/schemas/Mention"}}, "Source": {"type": "object", "properties": {"type":
```


配置Configuration Class

透過方法產生一個OpenAPI Bean

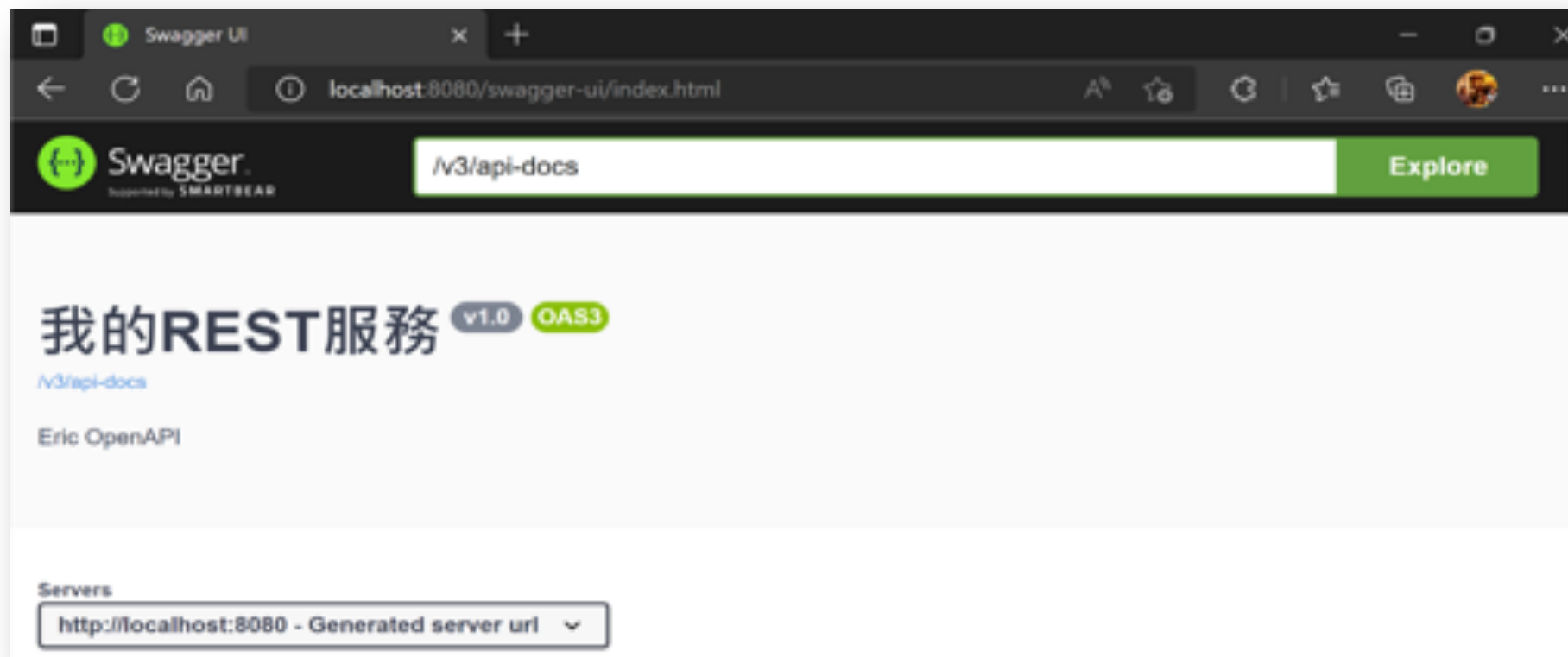
```
//組態配置
@Configuration
public class OpenApiConfig {
    //註冊一個OpenAPI物件至Spring Container中
    @Bean
    public OpenAPI initOpenAPI() {
        //建構OpenAPI物件
        //設定Info物件
        return new OpenAPI().info(|
            new Info()
                .title("我的REST服務")
                .description("Eric OpenAPI")
                .version("v1.0")
        );
    }
}
```

localhost:8080/v3/api-docs

```
{
  "openapi": "3.0.1",
  "info": {
    "title": "我的REST服務",
    "description": "Eric OpenAPI",
    "version": "v1.0"
  },
  "servers": [
    {
      "url": "http://localhost:8080",
      "description": "Generated server url"
    }
  ],
  "tags": [
    {
      "name": "串接CHT IoT物聯網專案",
      "description": "傳輸物聯網資料"
    }
  ],
  "paths": {
    "/customers/update": {
      "put": {
        "tags": [
          "customer-controller"
        ],
        "operationId": "customersUpdate",
        "requestBody": {
          "content": {
            "application/json": {
              "schema": {
                "$ref": "#/components/schemas/Customers"
              }
            }
          },
          "required": true
        },
        "responses": {
          "200": {
            "description": "OK",
            "content": {
              "*/*": {
                "schema": {

```

<http://localhost:8080/swagger-ui/index.html>





總結：16-1 Spring Swagger doc api應用

將Spring Boot配置springdoc api，並且配置
OpenAPI UI組態。接下來我們細部規劃Web API規範。

