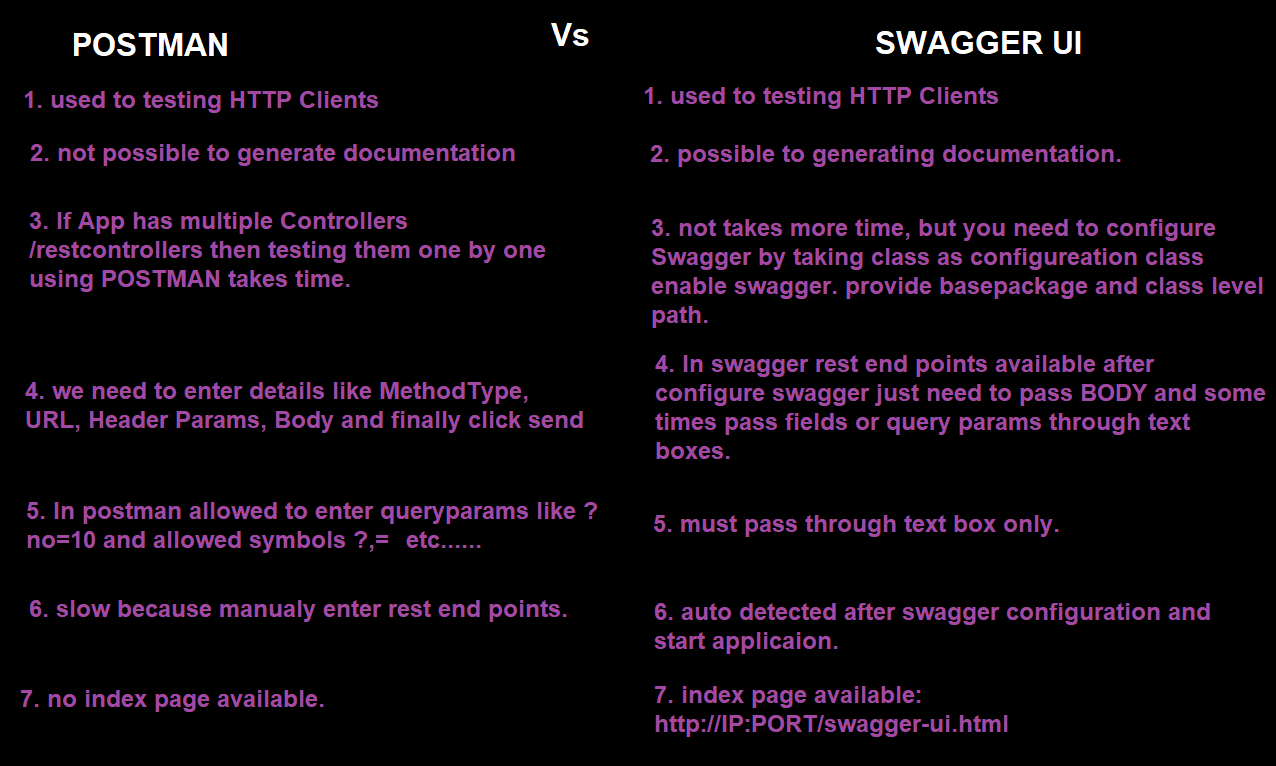
SWAGGER UI

Postman:

* Postman is tool used to Test Web/webservice applications.
* Here we need to enter all details manually, like URL, httpMethod, Headers, Body…ect..
* If App has multiple Controllers then testing them

one by one using POSTMAN takes time.

* We can use postman for .net,python..etc..
* POSTMAN: Tool used to make HTTP Request(to any producer application)
* It is used for Testing purpose only.
* Manually Tool, enter all details manually URL, HttpMethod, …etc..



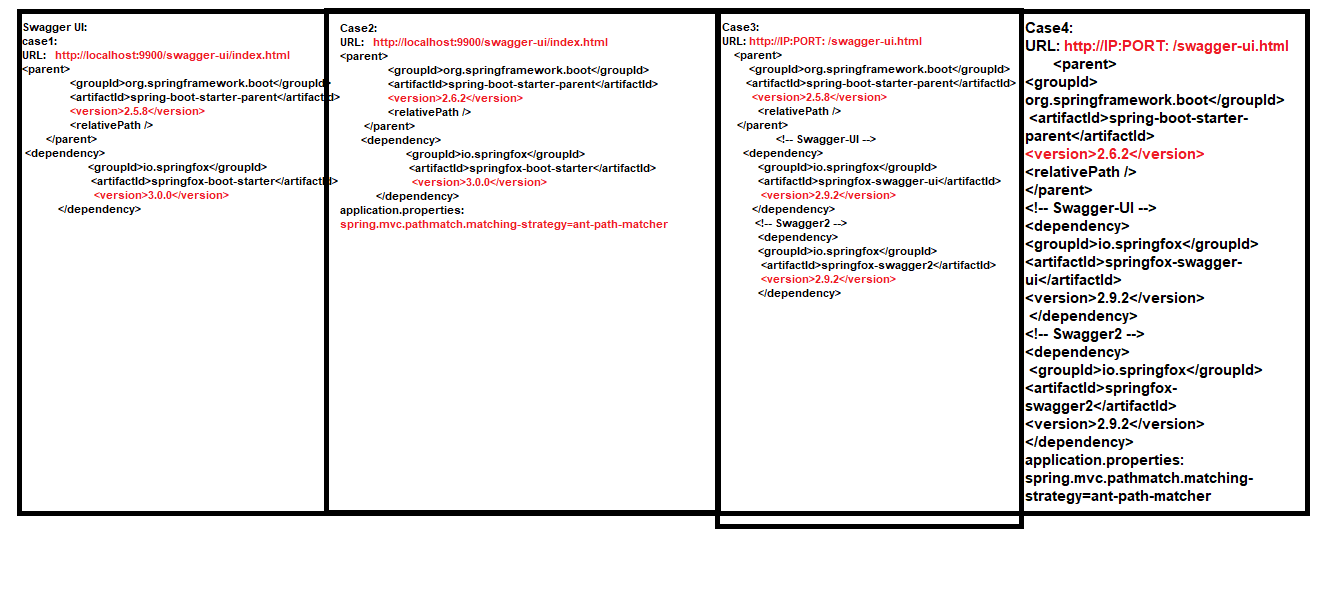
Swagger UI:( API/Tool)

* **Swagger UI will scan all possible methods and their URLs from Application and creates one dynamic UI (Docket).**
* **No need to enter URLs, Method types manually.**
* **In case of Project, it contains multiple RestControllers (ex: 10, Employee, Student, Course..etc) then Swagger is better.**
* **Swagger is not from Spring Boot, this is 3rd party ie SpringFox company.**
* **Swagger is used for java APIs testing.**
* **Swagger is mostly used which automates URL, HttpMethod, Detect all methods at one common UI generated at runtime.**
* **It is onetime configuration.**
* **It is 3rd party API(not from Spring Boot/Pivotal) it is given by SpringFox.**
* **/\*\* indicates any multi-level path. Ex: api/test/one/high**
* **Dynamic UI named as Docket.**
* **Used to testing controller/rest controller http methods.**

**Methods like GET, POST, PUT, DELETE, etc..**

* **Swagger used to generate API Documentation.**
* **Faster than postman because of auto detected all http methods swagger UI.**
* **No need to enter method names, contextpath, class path, method path and no need to enter query parameters like ?no=90.**
* **Need to configure Swagger UI in one configuration class.**
* **If your application using security provide .permitAll() access for swagger UI.**
* **Inside Swagger configuration class enable swagger UI through annotation** @EnableSwagger2 .
* **Create one bean method for Docket image.**
* **Pass basePackage and class level path to detect methods.**
* **If you want to detect all classes and methods then use any() method.**
* **Swagger UI This concept is used for Testing our RestController by making it as Easy UI.**
* **OpenAPI Specification (formerly Swagger Specification) is an API description format for REST APIs. An OpenAPI file allows you to describe your entire API.**
* **Swagger is a set of open-source tools built around the OpenAPI Specification that can help you design, build, document and consume REST APIs.**
* **Swagger also only HTTP Client used for Testing App.**
* **Swagger UI, we need to enter only Inputs (body/Params)**

**Other things are taken care by Swagger UI only.**

****

**Case1:**

**Dependencies:**

<parent>

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

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

<version>2.5.8</version>

<relativePath />

</parent>

**<!-- Swagger-UI -->**

**<dependency>**

**<groupId>io.springfox</groupId>**

**<artifactId>springfox-swagger-ui</artifactId>**

**<version>2.9.2</version>**

**</dependency>**

**<!-- Swagger2 -->**

**<dependency>**

**<groupId>io.springfox</groupId>**

**<artifactId>springfox-swagger2</artifactId>**

**<version>2.9.2</version>**

**</dependency>**

* **Swagger UI index page:**

[**http://IP:PORT**](http://IP:PORT)**: /swagger-ui.html**

**Example Swagger config class:**

package com.config;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import springfox.documentation.builders.PathSelectors;

import springfox.documentation.builders.RequestHandlerSelectors;

import springfox.documentation.spi.DocumentationType;

import springfox.documentation.spring.web.plugins.Docket;

import springfox.documentation.swagger2.annotations.EnableSwagger2;

**@Configuration**

**@EnableSwagger2**

**public class SwaggerConfig {**

**@Bean**

**//Create one Docket (Final Screen View)**

**public Docket createDocket() {**

**return new Docket(DocumentationType.SWAGGER\_2)**

**.select() //select all rest controllers common package //.apis(RequestHandlerSelectors.basePackage("com.controller"))**

**//inside package**

**.apis(RequestHandlerSelectors.any())**

**//.paths(PathSelectors.regex("/app.\*")) //restcontroller class level path**

**.paths(PathSelectors.any())**

**//Call build method that create docket with RestController objects**

**.build().useDefaultResponseMessages(false);**

**}**

}

**Case-2:**

Step1:

Dependencies:

<parent>

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

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

<version>2.6.2</version>

<relativePath></relativePath>

</parent>

<dependency>

<groupId>io.springfox</groupId>

<artifactId>springfox-boot-starter</artifactId>

<version>3.0.0</version>

</dependency>

Step2:

application.properties:

# Server Properties

server.port=9900

#Swagger UI

**spring.mvc.pathmatch.matching-strategy=ant-path-matcher**

step3:

SwaggerConfig:

package com.config;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import springfox.documentation.builders.PathSelectors;

import springfox.documentation.builders.RequestHandlerSelectors;

import springfox.documentation.spi.DocumentationType;

import springfox.documentation.spring.web.plugins.Docket;

import springfox.documentation.swagger2.annotations.EnableSwagger2;

@Configuration

@EnableSwagger2

public class SwaggerConfig {

@Bean

public Docket createDocket() {

return new Docket(DocumentationType.SWAGGER\_2)

.select()

.apis(RequestHandlerSelectors.any())//restcontrollers package

.paths(PathSelectors.any()) //restcontroller class level path

.build()

;

}

}

Step4:

package com.controller;

import org.springframework.http.HttpStatus;

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

@RequestMapping("/test")

public class SwaggerUIRestController {

/\*\*

\* URL: http://localhost:9900/test/message

\* @return

\*/

@GetMapping("/message")

public ResponseEntity<String> showMessage(){

return new ResponseEntity

<String>("Welcome to Spring Boot Rest Programming..",

HttpStatus.OK);

}

}

Step6:

URL:

**http://localhost:9900/swagger-ui/index.html**

case3:

URL: **http://localhost:9900/swagger-ui/index.html**

<parent>

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

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

**<version>2.5.8</version>**

<relativePath />

</parent>

<dependency>

<groupId>io.springfox</groupId>

<artifactId>springfox-boot-starter</artifactId>

**<version>3.0.0</version>**

</dependency>

Case4:

URL**: http://IP:PORT: /swagger-ui.html**

<parent>

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

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

**<version>2.6.2</version>**

<relativePath />

</parent>

<!-- Swagger-UI -->

<dependency>

<groupId>io.springfox</groupId>

<artifactId>springfox-swagger-ui</artifactId>

**<version>2.9.2</version>**

</dependency>

<!-- Swagger2 -->

<dependency>

<groupId>io.springfox</groupId>

<artifactId>springfox-swagger2</artifactId>

**<version>2.9.2</version>**

</dependency>

application.properties:

**spring.mvc.pathmatch.matching-strategy=ant-path-matcher**

**Disabling default Response messages:**

**package com.config;**

**import org.springframework.context.annotation.Bean;**

**import org.springframework.context.annotation.Configuration;**

**import springfox.documentation.builders.PathSelectors;**

**import springfox.documentation.builders.RequestHandlerSelectors;**

**import springfox.documentation.spi.DocumentationType;**

**import springfox.documentation.spring.web.plugins.Docket;**

**import springfox.documentation.swagger2.annotations.EnableSwagger2;**

**@Configuration**

**@EnableSwagger2**

**public class SwaggerConfig {**

**@Bean**

**public Docket createDocket() {**

**return new Docket(DocumentationType.SWAGGER\_2)**

**.select()**

**.apis(RequestHandlerSelectors.any())//restcontrollers path**

**.paths(PathSelectors.any()) //starter class path**

**/\*\***

**\* Disabling defualt Response messages**

**\* 401 : Unauthorized**

**\* 403 : Forbidden**

**\* 404 : Not Found**

**\*/**

**.build().useDefaultResponseMessages(false)**

**;**

**}**

**}**

**Appliction1:**

**pom.xml:**

<parent>

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

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

<version>2.5.8</version>

<relativePath /> <!-- lookup parent from repository -->

</parent>

<properties>

<java.version>11</java.version>

</properties>

<dependencies>

<!-- Swagger -->

<dependency>

<groupId>io.springfox</groupId>

<artifactId>springfox-boot-starter</artifactId>

<version>3.0.0</version>

</dependency>

**Application.properties:**

# Server Properties

server.port=9900

**SwaggerConfig:**

package com.config;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import springfox.documentation.builders.PathSelectors;

import springfox.documentation.builders.RequestHandlerSelectors;

import springfox.documentation.spi.DocumentationType;

import springfox.documentation.spring.web.plugins.Docket;

import springfox.documentation.swagger2.annotations.EnableSwagger2;

@Configuration

@EnableSwagger2

public class SwaggerConfig {

@Bean

public Docket createDocket() {

return new Docket(DocumentationType.SWAGGER\_2)

.select()

.apis(RequestHandlerSelectors.basePackage("com"))//starter path like com

/\*\*

\* RestController class level path

\* . means single character and \* means more characters..

\*/

.paths(PathSelectors.regex("/.\*"))

/\*\*

\* Disabling defualt Response messages

\* 401 : Unauthorized

\* 403 : Forbidden

\* 404 : Not Found

\*/

.build().useDefaultResponseMessages(false)

;

}

}

RestController:

package com.controller;

import org.springframework.http.HttpStatus;

import org.springframework.http.ResponseEntity;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RestController;

@RestController

@RequestMapping("/test")

public class SwaggerUIRestController {

/\*\*

\* URL: http://localhost:9900/test/message

\* @return

\*/

@GetMapping("/message")

public ResponseEntity<String> showMessage(){

return new ResponseEntity

<String>("Welcome to Spring Boot Rest Programming..",

HttpStatus.OK);

}

}

**//Added Contact info, APIInfo, VendorExpression:**

**package** com.config;

**import** java.util.ArrayList;

**import** java.util.List;

**import** org.springframework.context.annotation.Bean;

**import** org.springframework.context.annotation.Configuration;

**import** springfox.documentation.builders.PathSelectors;

**import** springfox.documentation.builders.RequestHandlerSelectors;

**import** springfox.documentation.service.ApiInfo;

**import** springfox.documentation.service.Contact;

**import** springfox.documentation.service.VendorExtension;

**import** springfox.documentation.spi.DocumentationType;

**import** springfox.documentation.spring.web.plugins.Docket;

**import** springfox.documentation.swagger2.annotations.EnableSwagger2;

@Configuration

@EnableSwagger2

**public** **class** SwaggerConfig {

@Bean

**public** Docket createDocket() {

**return** **new** Docket(DocumentationType.***SWAGGER\_2***)

.select()

.apis(RequestHandlerSelectors.*basePackage*("com"))//starter path like com

/\*\*

\* RestController class level path

\* . means single character and \* means more characters..

\*/

.paths(PathSelectors.*regex*("/.\*"))

/\*\*

\* Disabling defualt Response messages

\* 401 : Unauthorized

\* 403 : Forbidden

\* 404 : Not Found

\*/

.build().useDefaultResponseMessages(**false**)

//meta data

.apiInfo(apiInfo())

;

}

/\*\*

\* Meta data,

\* only display content

\* **@return**

\*/

**private** ApiInfo apiInfo() {

**return** **new** ApiInfo(

"TestApp", //title

"Testing for URLs", //Description

"swagger 3.0", //version

"https://github.com/karrasankar158 ", // Service URL

contactInfo(), //contact

"own", // license

"https://github.com/karrasankar158", // License URL

expressionInfo()); // Vendor Expressions

}

//Contact info

**private** Contact contactInfo() {

**return** **new** Contact("sankar","https://github.com/karrasankar158","sankar@gmail");

}

// expression

@SuppressWarnings("rawtypes")

**private** List<VendorExtension> expressionInfo() {

ArrayList<VendorExtension> list=**new** ArrayList<VendorExtension>();

list.toString();

**return** list;

}

}

**@ApiIgnore**

This annotation applied in class, method levels

This annotation is used to ignore

or hide single or multiple methods

Or all methods in RestController.

Q) How can we avoid showing a method at Swagger UI?

A)

top of method applies:

@ApiIgnore

Display our specified method name for restController methods:

Instead of showing method name at Swagger UI

@ApiOperation("hello")

Display our specified restController name:

Instead of showing controller name

@Api(~~description~~ = "test")

Q) How can we restrict using Swagger only at Dev and Test Environments not in production?

A) Profiles Concept:

@Profile({"dev","qa"})

//--spring.profiles.active=dev

package com.config;

import java.util.ArrayList;

import java.util.List;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import org.springframework.context.annotation.Profile;

import springfox.documentation.builders.PathSelectors;

import springfox.documentation.builders.RequestHandlerSelectors;

import springfox.documentation.service.ApiInfo;

import springfox.documentation.service.Contact;

import springfox.documentation.service.VendorExtension;

import springfox.documentation.spi.DocumentationType;

import springfox.documentation.spring.web.plugins.Docket;

import springfox.documentation.swagger2.annotations.EnableSwagger2;

@Configuration

@EnableSwagger2

@Profile({"dev","qa"})

//--spring.profiles.active=dev

public class SwaggerConfig {

@Bean

public Docket createDocket() {

return new Docket(DocumentationType.SWAGGER\_2)

.select()

.apis(RequestHandlerSelectors.basePackage("com"))//starter path like com

/\*\*

\* RestController class level path

\* . means single character and \* means more characters..

\*/

.paths(PathSelectors.regex("/.\*"))

/\*\*

\* Disabling defualt Response messages

\* 401 : Unauthorized

\* 403 : Forbidden

\* 404 : Not Found

\*/

.build().useDefaultResponseMessages(false)

//meta data

.apiInfo(apiInfo())

;

}

/\*\*

\* Meta data,

\* only display content

\* @return

\*/

private ApiInfo apiInfo() {

return new ApiInfo(

"TestApp", //title

"Testing for URLs", //Description

"swagger 3.0", //version

"https://github.com/karrasankar158 ", // Service URL

contactInfo(), //contact

"own", // license

"https://github.com/karrasankar158", // License URL

expressionInfo()); // Vendor Expressions

}

//Contact info

private Contact contactInfo() {

return new Contact("sankar","https://github.com/karrasankar158","sankar@gmail");

}

// expression

@SuppressWarnings("rawtypes")

private List<VendorExtension> expressionInfo() {

ArrayList<VendorExtension> list=new ArrayList<VendorExtension>();

list.toString();

return list;

}

}

application-dev.properties:

# Server properties

Server.port=9089

application-qa.properties:

# Server properties

Server.port=9089

Add dependency then automatically enabled security:

<dependency>

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

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

</dependency>

**----------------------------The End---------------------------------------**