

[CS304] Lab10 Software Documentation

Part 1 JavaDoc

JavaDoc tool is a document generator tool in Java programming language for generating standard documentation in HTML format. It generates API documentation. It parses the declarations and documentation in a set of source file describing classes, methods, constructors, and fields.

Before using JavaDoc tool, you must include JavaDoc comments `/**.....*/` providing information about classes, methods, and constructors, etc. For creating a good and understandable document API for any java file you must write better comments for every class, method, constructor.

Example :

Syntax	Parameter	Description
@author	author_name	Describes an author
@param	description	provide information about method parameter or the input it takes
@see	reference	generate a link to other element of the document
@version	version-name	provide version of the class, interface or enum.
@return	description	provide the return value

More Tag information links(Official links):

<https://docs.oracle.com/javase/8/docs/technotes/tools/windows/javadoc.html>

Part 1-1 Start a simple Java programe

Creat a simple Java program (int)a+b,please refer attachment(example_01.java),run command-line below the java path:

```
javadoc -author -version -encoding UTF-8 -charset UTF-8 example_01.java
```

文件 | D:/CS304_2023S/class%20doc/week10/test/src/example_01.html

程序包 类 树 索引 帮助

概要: 嵌套 | 字段 | 构造器 | 方法 详细资料: 字段 | 构造器 | 方法

SEARCH: Search

```
public class example_01
extends Object
```

版本:
jdk1.8.0

作者:
CS304

构造器概要

构造器

构造器	说明
example_01()	

方法概要

所有方法 静态方法 具体方法

修饰符和类型	方法	说明
static int	add(int n, int m)	This is a program for adding two numbers in java.
static void	main(String[] args)	This is the main method which is very important for execution for a java program.

从类继承的方法 java.lang.Object

方法详细资料

main

```
public static void main(String[] args)
```

This is the main method which is very important for execution for a java program.

参数:
args - system configuration

add

```
public static int add(int n,
int m)
```

This is a program for adding two numbers in java.

参数:
n - first element
m - second element

返回:
m+n

and if you run the command below,you will get a English version:

```
javadoc -author -version -J-Duser.language=en -J-Duser.country=US -encoding UTF-8
-charset UTF-8 example_01.java
```



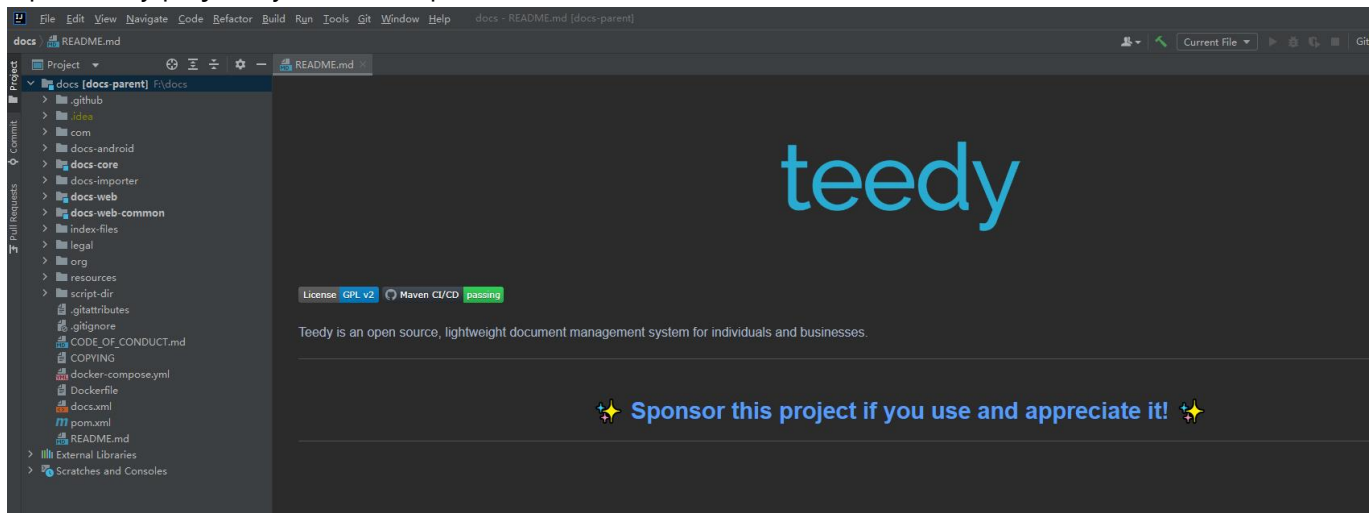
About command you can refer oracle document below:

<https://docs.oracle.com/javase/8/docs/technotes/tools/windows/javadoc.html>

`javadoc {packages|source-files} [options] [@argfiles]`

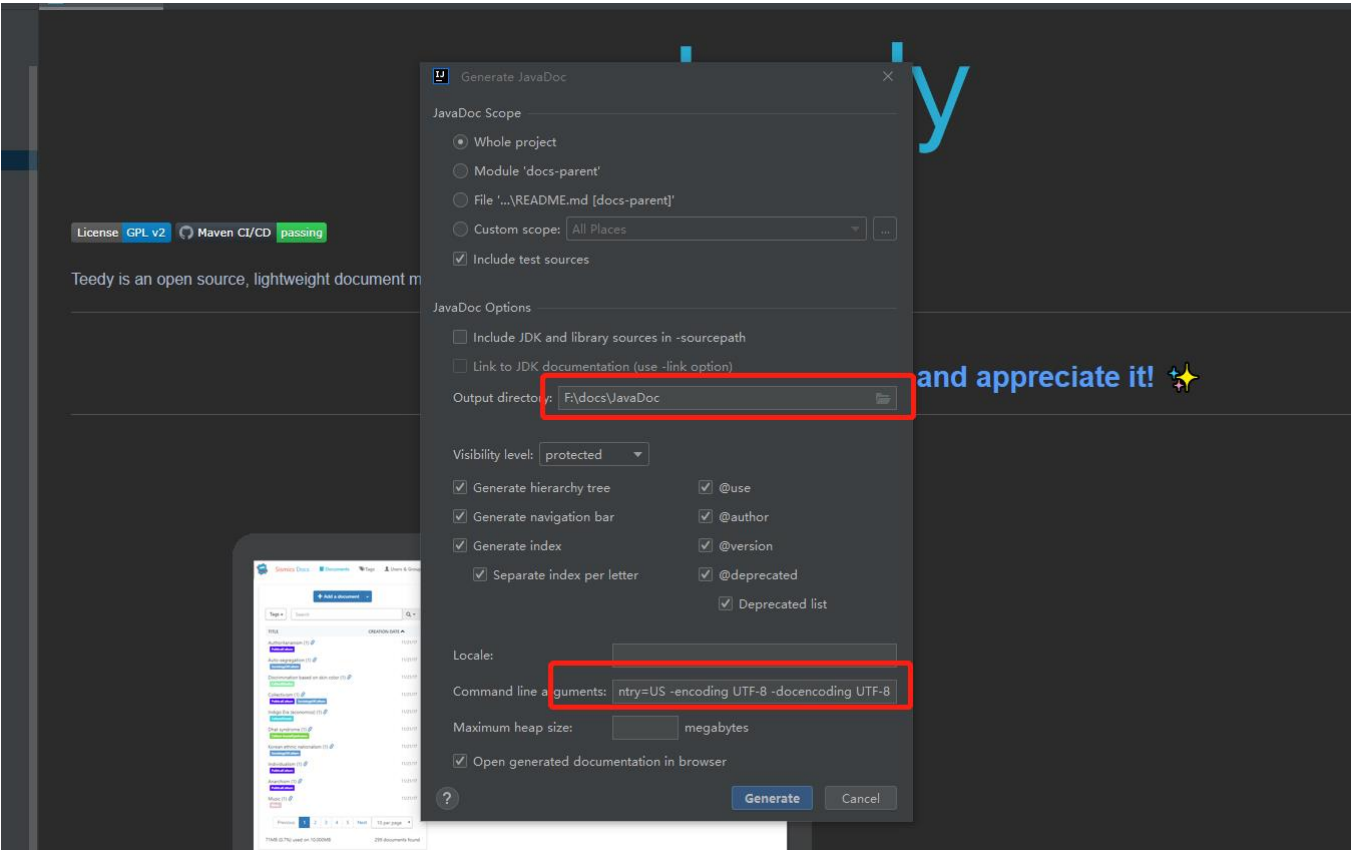
Part 1-2 Create document for Teedy Project

Open Teedy project by IDEA, and open tools->Generate Javadoc

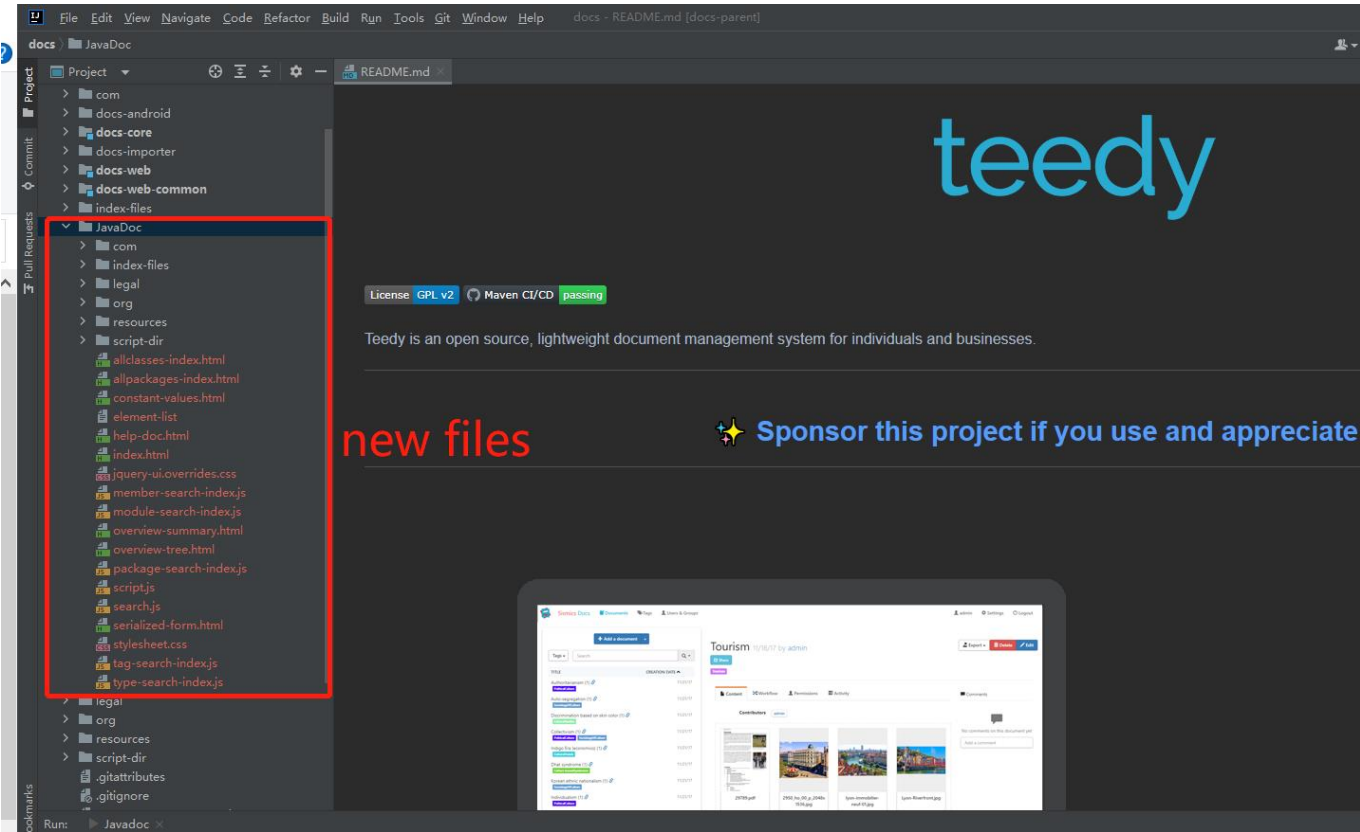


config command line argument: tips:Javadoc created by yourself

```
-J-Duser.language=en -J-Duser.country=US -encoding UTF-8 -docencoding UTF-8
```



then you will see :



Open index.html:

←

↺

📄 文件 | F:/docs/JavaDoc/index.html

A

🔖

📄

🔍

👤

⋮

OVERVIEW

PACKAGE

CLASS

USE

TREE

INDEX

HELP

SEARCH: 🔍 Search

Packages

Package	Description
com.sismics.docs	
com.sismics.docs.core.constant	
com.sismics.docs.core.dao	
com.sismics.docs.core.dao.criteria	
com.sismics.docs.core.dao.dto	
com.sismics.docs.core.dao.jpa	
com.sismics.docs.core.event	
com.sismics.docs.core.listener.async	
com.sismics.docs.core.model.context	
com.sismics.docs.core.model.jpa	
com.sismics.docs.core.service	
com.sismics.docs.core.util	
com.sismics.docs.core.util.action	
com.sismics.docs.core.util.authentication	
com.sismics.docs.core.util.format	

Package **com.sismics.docs.core.dao**

package com.sismics.docs.core.dao

Related Packages

Package	Description
com.sismics.docs.core.dao.criteria	
com.sismics.docs.core.dao.dto	
com.sismics.docs.core.dao.jpa	

Classes

Class	Description
AclDao	ACL DAO.
AuditLogDao	Audit log DAO.
AuthenticationTokenDao	Authentication token DAO.
CommentDao	Comment DAO.
ConfigDao	Configuration parameter DAO.

Class **AclDao**

Method Details

create

public String create(Acl acl, String userId)

Creates a new ACL.

Parameters:

- acl - ACL
- userId - User ID

Returns:

- New ID

getByTargetId

public List<Acl> getByTargetId(String targetId)

Search ACLs by target ID.

Parameters:

- targetId - Target ID

```
21 * ACL DAO.
22 *
23 * @author bgamard
24 */
25
26 public class AclDao {
27     /**
28      * Creates a new ACL.
29      *
30      * @param acl ACL
31      * @param userId User ID
32      * @return New ID
33      */
34     public String create(Acl acl, String userId) {
35         // Create the UUID
36         acl.setId(UUID.randomUUID().toString());
```

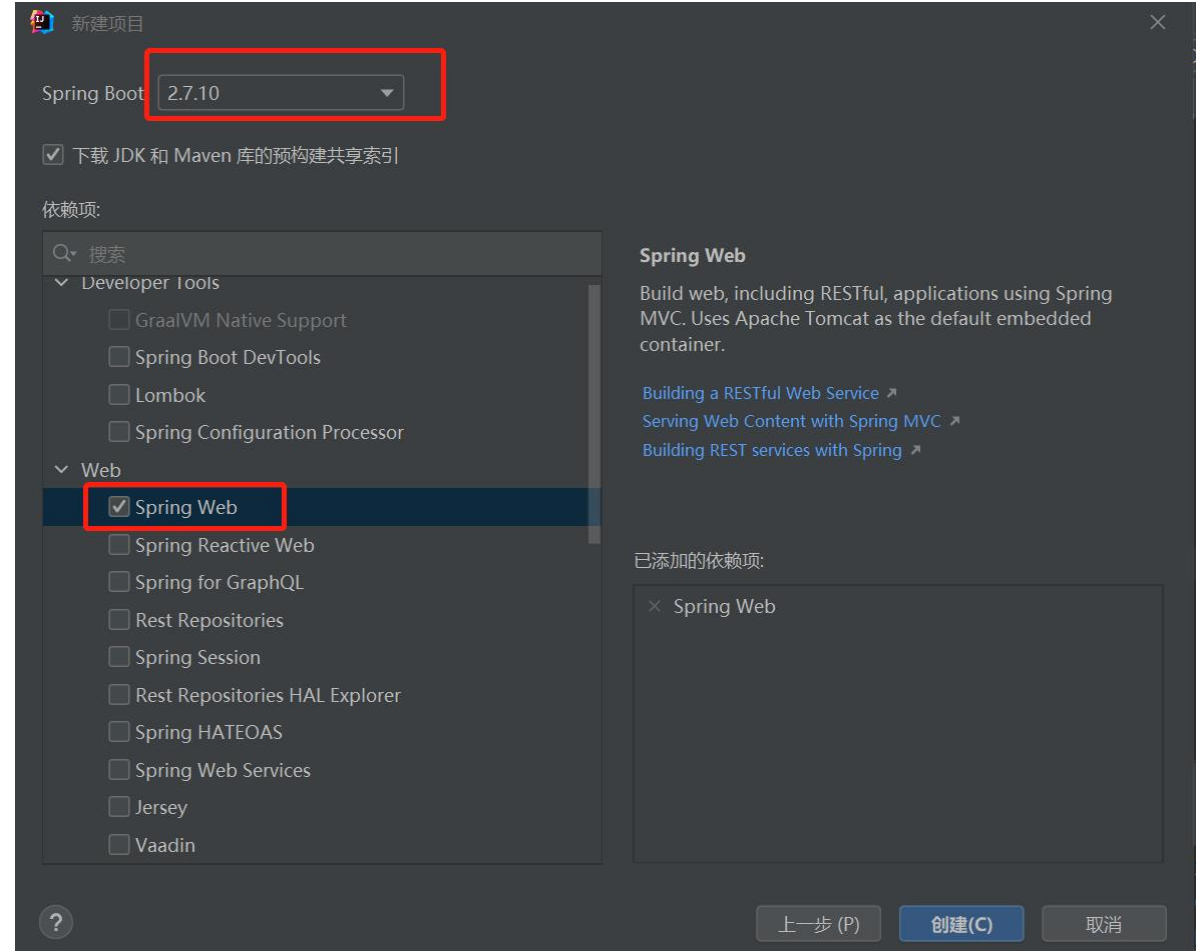
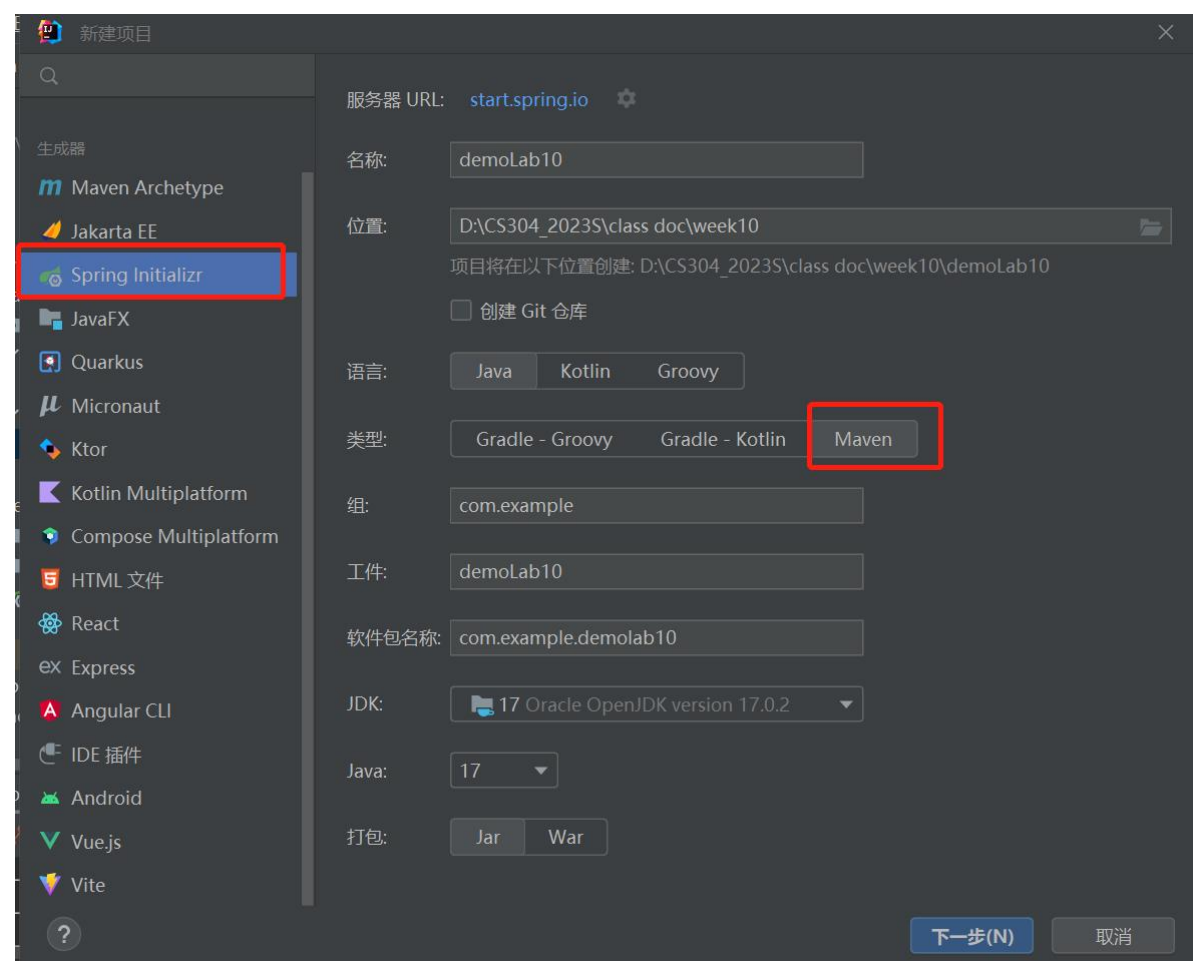
Part 2 Swagger

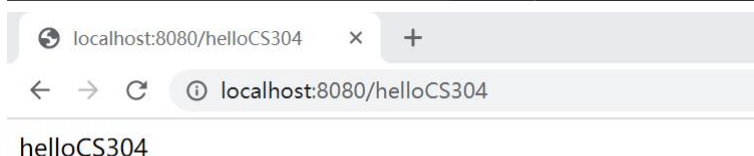
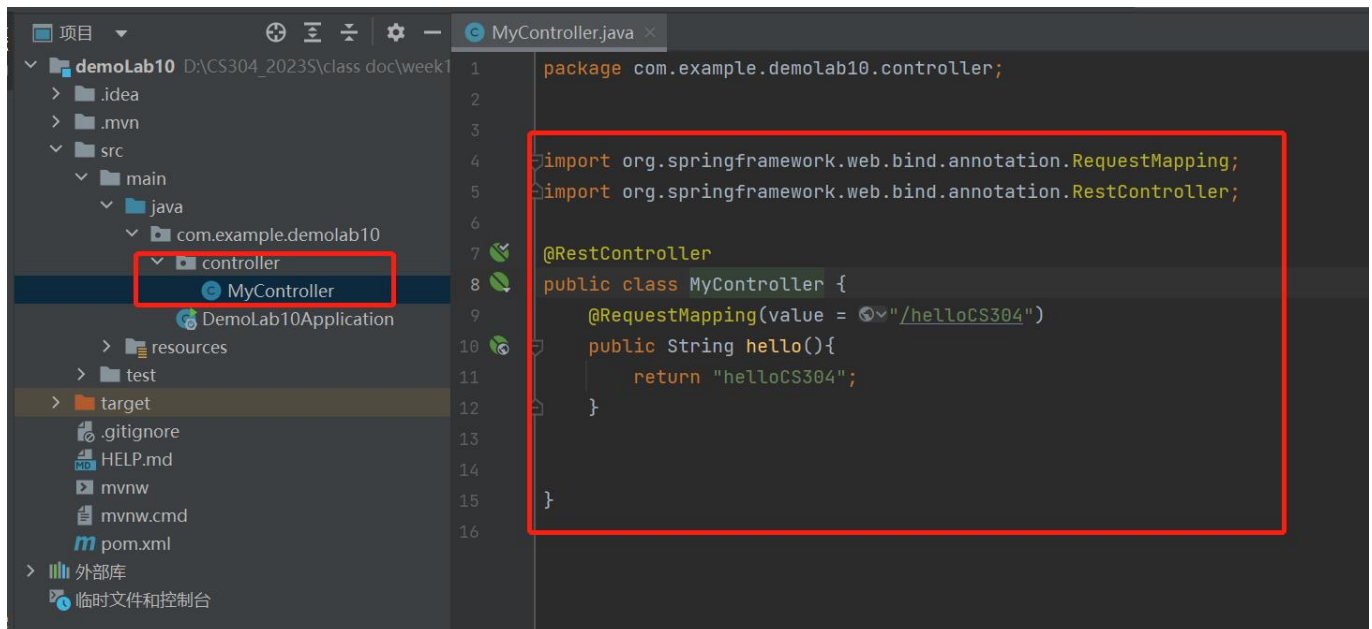
Official website: <https://swagger.io/>

Swagger, is an API description format for RESTful APIs. It provides a standardized way to describe the structure of an API, making it easy for developers to understand and interact with the API. The OpenAPI Specification (OAS) defines a set of rules for describing the API's endpoints, methods, input parameters, and response objects.

Part 2-1 Start a simple SpringBoot project with Swagger

Tips: If you want to create a SpringBoot template project by IDEA, you can apply a IDEA education account.





Config Swagger and start:

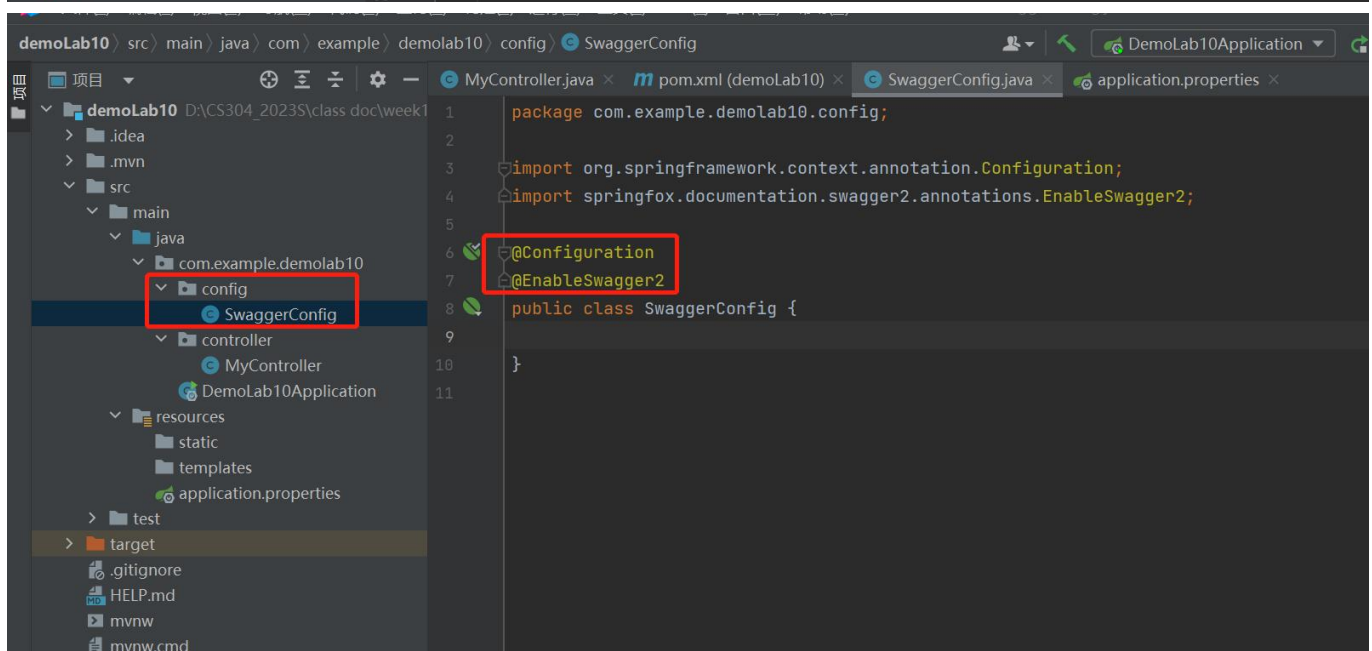
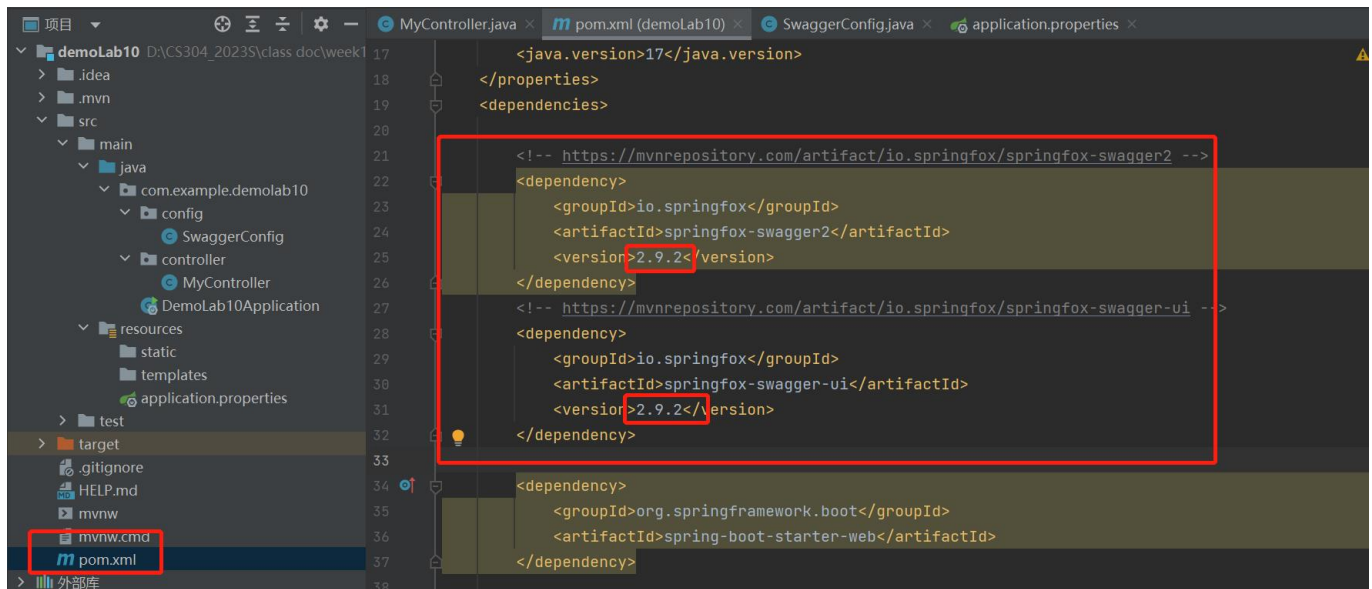
mvn repo:

<https://mvnrepository.com/artifact/io.springfox/springfox-swagger2/2.9.2>

<https://mvnrepository.com/artifact/io.springfox/springfox-swagger-ui/2.9.2>

```
<!-- https://mvnrepository.com/artifact/io.springfox/springfox-swagger2 -->
<dependency>
    <groupId>io.springfox</groupId>
    <artifactId>springfox-swagger2</artifactId>
    <version>2.9.2</version>
</dependency>

<!-- https://mvnrepository.com/artifact/io.springfox/springfox-swagger-ui -->
<dependency>
    <groupId>io.springfox</groupId>
    <artifactId>springfox-swagger-ui</artifactId>
    <version>2.9.2</version>
</dependency>
```

maybe there's version conflicts, you need config:

```
spring.mvc.pathmatch.matching-strategy=ant_path_matcher
```

The image shows a development environment with an IDE and a web browser. In the IDE, the file `application.properties` is open, showing the property `spring.mvc.pathmatch.matching-strategy=ant_path_matcher`. The project structure on the left includes `demoLab10` with subdirectories like `src`, `main`, `java`, `resources`, and `test`. The web browser shows the Swagger UI at `localhost:8080/swagger-ui.html/#/`. The Swagger UI header has a "Select a spec" dropdown set to "default". The main content area shows "Api Documentation 1.0" with a "basic information" label. Below this, there are sections for "basic-error-controller" (Basic Error Controller) and "my-controller" (My Controller), both labeled "APIs". At the bottom, there is a "Models" section labeled "class".

Config basic information:

```
@Bean
public Docket docket(){
    return new Docket(DocumentationType.SWAGGER_2)
        .apiInfo(apiInfo());
}
private ApiInfo apiInfo(){
    Contact contact =new
Contact("CS304","https://sustech.edu.cn","123@mail.sustech.edu.cn");
    return new ApiInfo(
        "CS304hello Documentation",
        "Api Documentation for lab10",
        "1.0",
        "https://sustech.edu.cn",
```

```

        contact,
        "Apache 2.0",
        "https://www.apache.org/licenses/LICENSE-2.0",
        new ArrayList());
    }
}

```

The screenshot shows an IDE with the `SwaggerConfig.java` file open. The code defines a `SwaggerConfig` class with a `Docket` bean and an `apiInfo()` method. The `apiInfo()` method returns a `ApiInfo` object with the following properties:

- `name`: "CS304"
- `url`: "https://sustech.edu.cn"
- `email`: "123@mail.sustech.edu.cn"
- `title`: "CS304hello Documentation"
- `description`: "Api Documentation for lab10"
- `version`: "1.0"
- `termsOfServiceUrl`: "https://sustech.edu.cn"
- `contact`: a `Contact` object with the same properties as the `apiInfo()` method.
- `license`: "Apache 2.0"
- `licenseUrl`: "https://www.apache.org/licenses/LICENSE-2.0"
- `new ArrayList()`: an empty `ArrayList`.

Below the IDE, a browser window shows the Swagger UI at `localhost:8080/swagger-ui.html/#/`. The UI displays the following information:

- CS304hello Documentation** (version 1.0)
- Base URL: `localhost:8080/`
- API Docs: <http://localhost:8080/v2/api-docs>
- Api Documentation for lab10
- [Terms of service](#)
- [CS304 - Website](#)
- [Send email to CS304](#)
- [Apache 2.0](#)

The UI also lists the controllers and models:

- basic-error-controller** Basic Error Controller
- my-controller** My Controller
- Models**

Config start choice:

If I want start different properties like development environment, production environment etc..

application.properties

```

spring.mvc.pathmatch.matching-strategy=ant_path_matcher
spring.profiles.active=pro

```

application-pro.properties

```
server.port=9090
```

application-dev.properties

```
server.port=8899
```

SwaggerConfig:

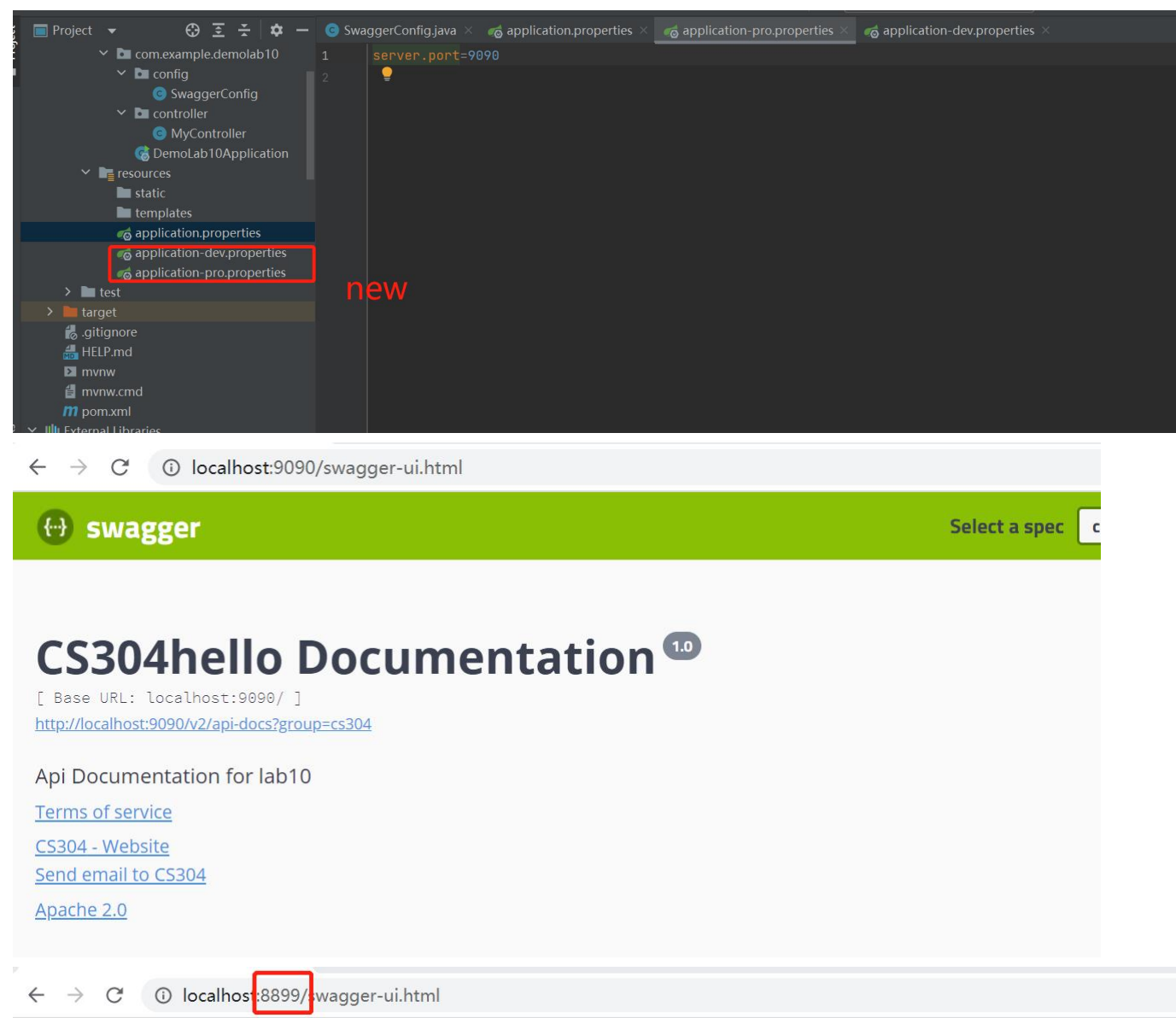
```
@Bean
public Docket docket(Environment environment){

    Profiles profiles=Profiles.of("dev","pro");
    boolean flag=environment.acceptsProfiles(profiles);

    return new Docket(DocumentationType.SWAGGER_2)
        .apiInfo(apiInfo())
        .enable(flag)
        .select()

    .apis(RequestHandlerSelectors.basePackage("com.example.demolab10.controller"))
        .paths(PathSelectors.any())
        .build();
}
```

you can get different result by using different config.



无法访问此网站

localhost 拒绝了我们的连接请求。

请试试以下办法：

- 检查网络连接
- 检查代理服务器和防火墙

ERR_CONNECTION_REFUSED

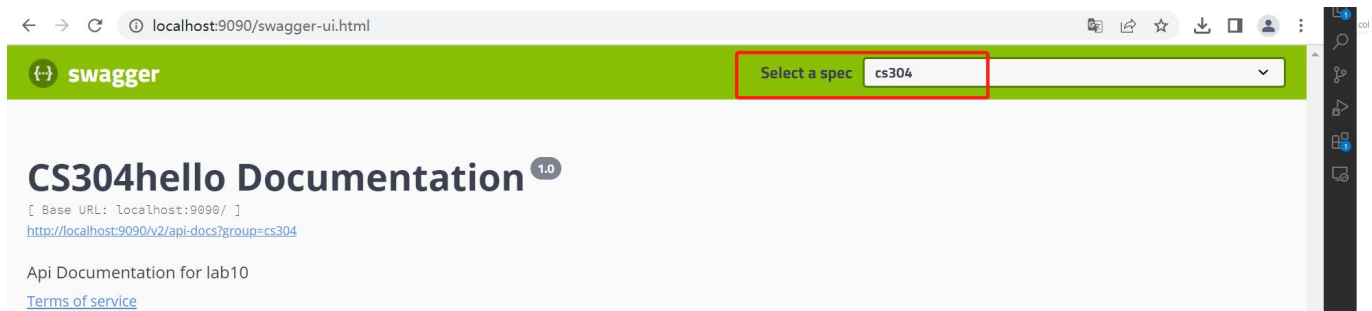
Add group:

```
@Bean
public Docket docket(Environment environment){
```

```
Profiles profiles=Profiles.of("dev","pro");
boolean flag=environment.acceptsProfiles(profiles);

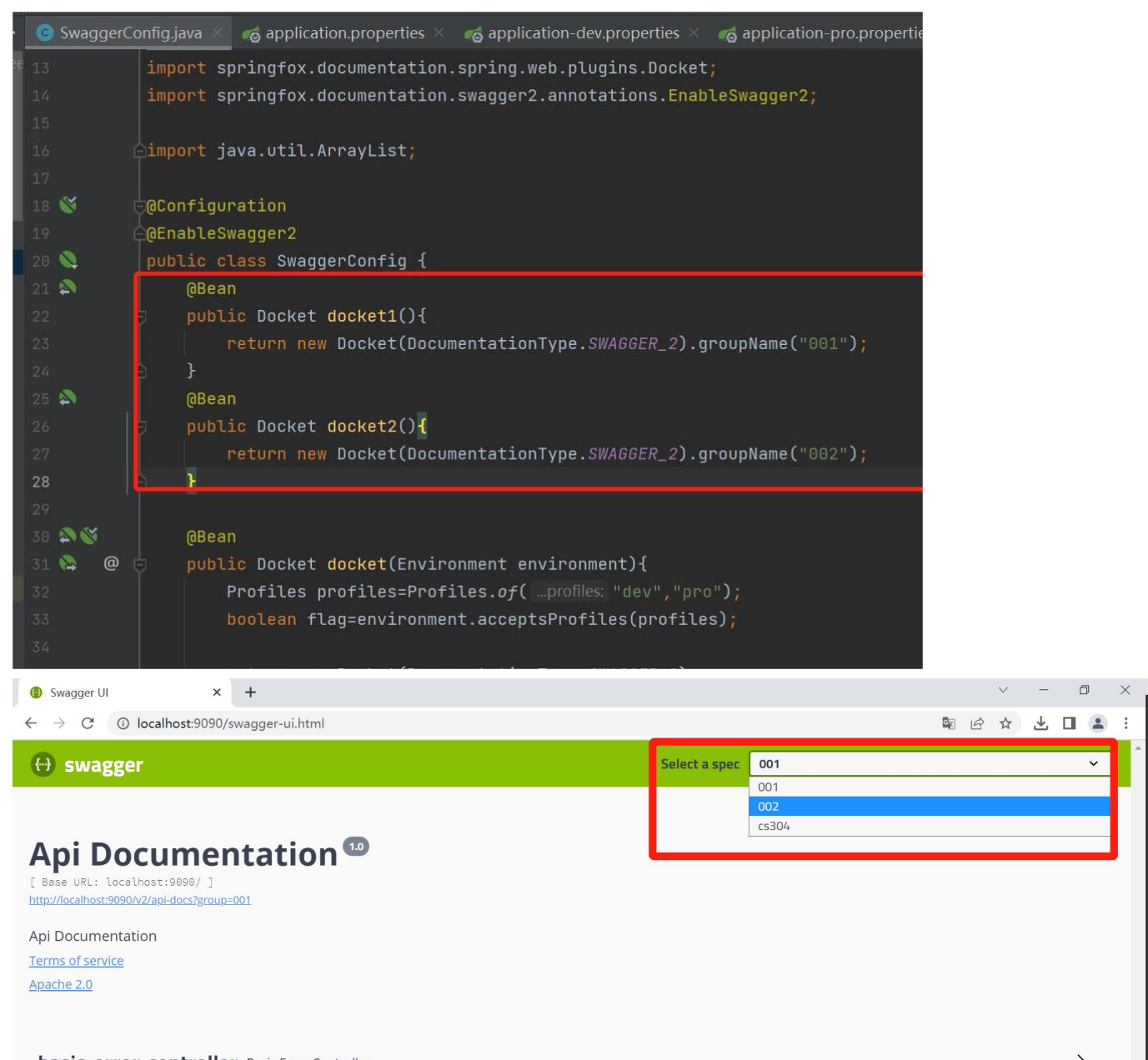
return new Docket(DocumentationType.SWAGGER_2)
    .apiInfo(apiInfo())
    .groupName("cs304")
    .enable(flag)
    .select()

.apis(RequestHandlerSelectors.basePackage("com.example.demolab10.controller"))
    .paths(PathSelectors.any())
    .build();
}
```



Add two more Docket

```
@Bean
public Docket docket1(){
    return new Docket(DocumentationType.SWAGGER_2).groupName("001");
}
@Bean
public Docket docket2(){
    return new Docket(DocumentationType.SWAGGER_2).groupName("002");
}
```



Annotation rules:

