**background：**

Developers need to write a development document after each module function is written, but there may be many ways to do it, so they have to compare their own methods.

It takes time and energy to develop corresponding documents.

**function**：

According to their own needs to automatically generate the development document word, you can customize the word document template, different classes are divided into different word documents, the method list out a table, the method has all the information corresponding, just need us to fill in the function can be

**Custom word template (non essential)**

(If you don't want to customize the way the document is generated from the built-in template, here's how the development document for your own project provides:)

1. Open a template for the development document word of your own project
2. save to desktop as XML format.
3. Find the corresponding location with Ctrl + f, and replace the variables. If it is a table, you need to add a < # list entity list as Li > on the label of the table. It is easier to configure the template for checking FTL documents online.。
4. variable comparison table

Class name：${className}

<#list entityList as li>

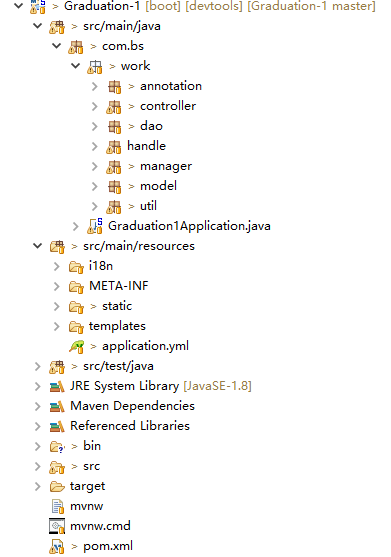
|  |  |
| --- | --- |
| Modifier | ${li.modifier} |
| Mothd Name | ${li.name} |
| Number of parameters | ${li.returnCount} |
| Parameter type | ${li.type} |
| return type | ${li.returnValue} |
| Functional description | ${li.fc} |

**Mode of use**：

Create a new method in any project that can be executed, such as a test class, or browser call in crontroller, as long as you can execute the following code:

For example, my project directory is as follows (this item is for SpringBoot project, all other items are acceptable).

Here's a way to generate functional descriptions of annotations, either manually in Word if you don't need them：



**Method 1：（Generate all the development documents under a package）**

ClassGet cn=**new** ClassGet();

String packages = "com.bs.work.model";//To generate the package name of the document

String prefix="src.main.java";

String path="D:/wordTest/";//To generate local paths for location

**boolean** classInfo = cn.classInfo(packages, prefix,path);

**Method 2：（Generate the specified class development document）**

ClassGet cn=**new** ClassGet();

String path="D:/wordTest/";

**Boolean** classInfo=cn.classInfo("com.bs.work.controller.UserController", path);//The first parameter needs to fill in the full path of the formulated class.

**Method 3：（Generate a development document for all classes under the package based on the word template.）**

//If my word template path is "src/createdoc/test.ftl"

String ftlName="test";//word Template name

String ftlPath="/createdoc";//word Template path

String packages = "Test";

String prefix="src";

String path="D:/test/";

**boolean** classInfo = cn.classInfo(packages, prefix,path,ftlName,ftlPath);

**Method 4：（Generate the specified class development document based on word template**

**）**

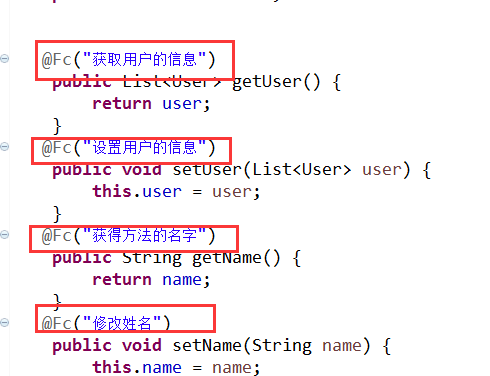
String ftlName="test";//Word template name

String ftlPath="/createdoc";//word Template path

String path="D:/wordTest/";//Locally saved paths

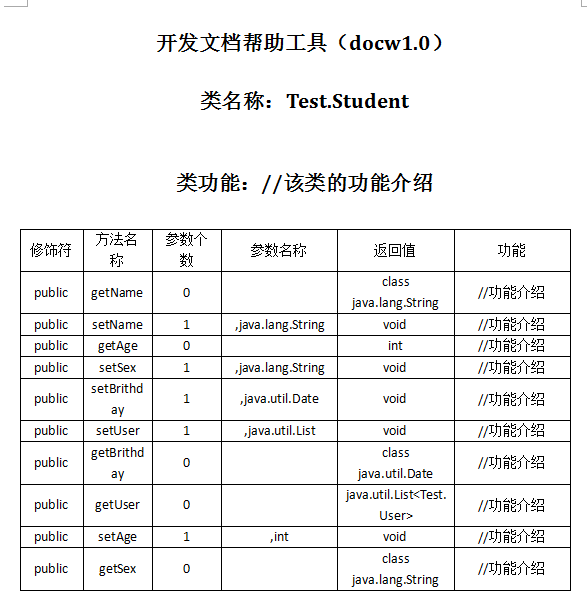
**boolean** classInfo = cn.classInfo("com.bs.work.controller.ExcelController", path,ftlName,ftlPath);

Use time screenshots：









Matters needing attention：

The final result is shown in Figure 1. It's very convenient to copy and paste the entire content onto your own development document by filling in the information described below.

But the path of the project is not the same, where you need to call it according to your own project.

Friendship hints that the prefix of the springboot project must be configured as "src.main.java".

Ordinary Java projects do not need to be considered. Of course, they should be considered according to the actual situation