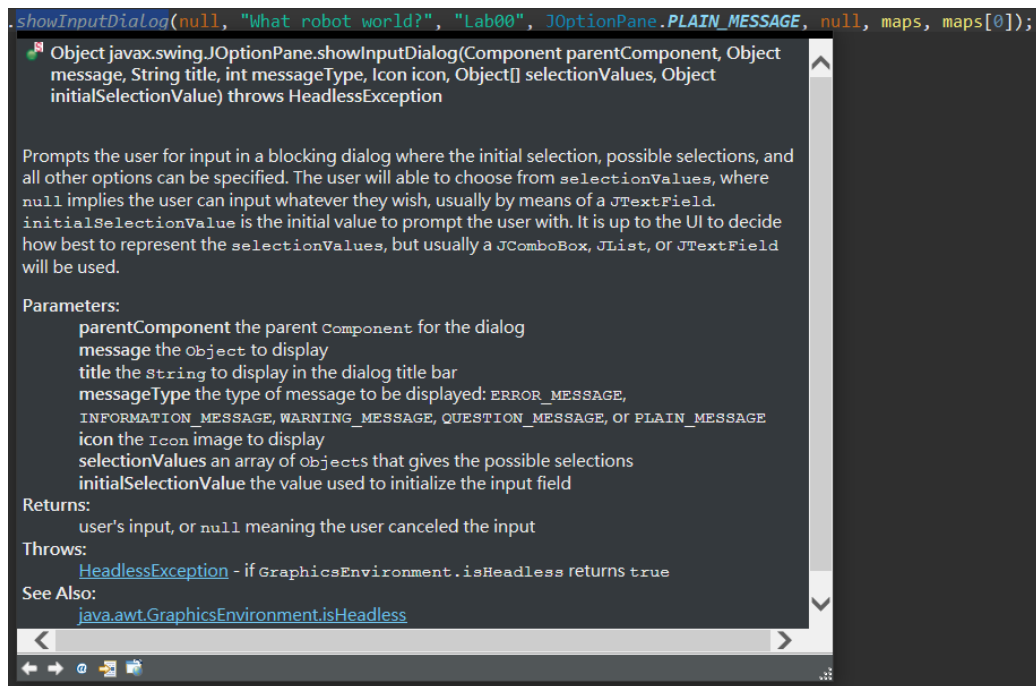


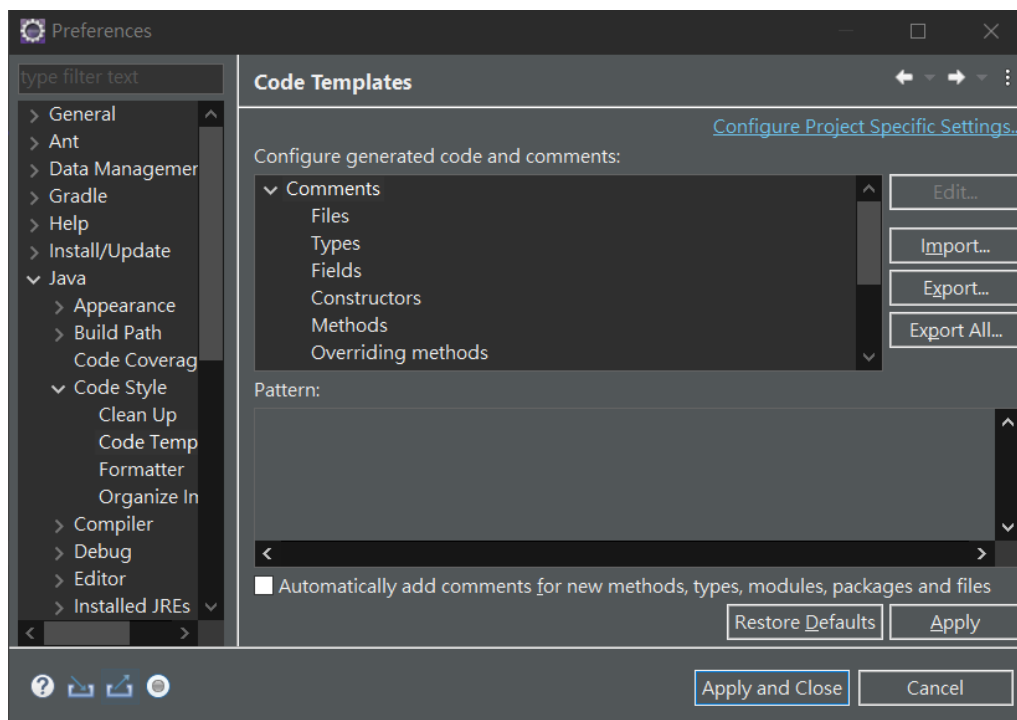
# How to generate javadoc in eclipse

在 Java 中，我們可以看到不少的 API 上面都包含著註解，可以幫助提醒使用者這個 API 所包含的功能，使用 Eclipse 中內建的 Javadoc Generation 能讓我們更容易的為 Java 的 project 生成 Javadoc 註解。



## 建立註解：

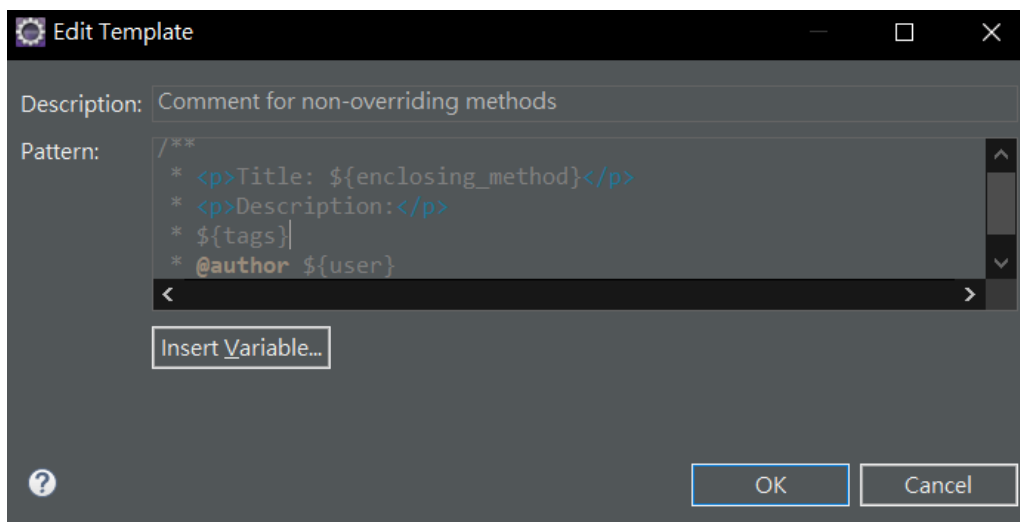
我們可以在 Window > Preference > Java > Code Style > Code Templates > Comments 找到設定自動生成註解的模板。



常用的模板設定選項如下：

- Files：對整個.java 檔的註解模板
- Constructors：對建構方式的註解模板
- Methods：對函式的註解模板
- Overriding Methods：對改寫的函式的註解模板
- Getter：對 get 的函式的註解模板

點擊 Edit 可分別對不同的模板進行設定



Insert Variable 可新增不同的參數進模板中。

e.g.：

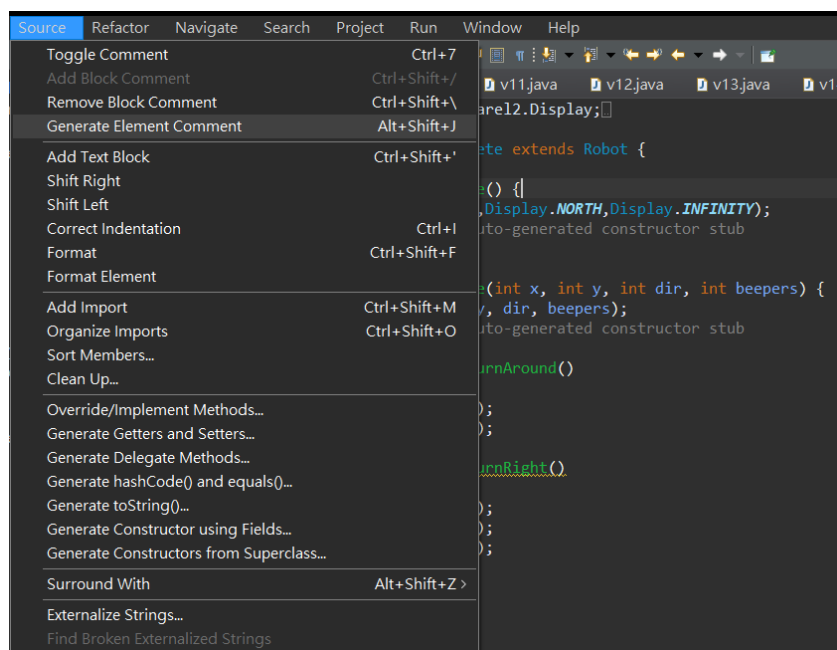
\${enclosing\_method}：method 名稱

\${tags}：Generated Javadoc tags (@param, @return...)

\${todo}：Todo task tag

\${user}：User Name

點擊欲加入註解的位置 Source > Generate Element Comment



對 Constructors 加入註解後：

```
/**
 * <p>Title: Athlete.java </p>
 * <p>Description: </p>
 * @param x
 * @param y
 * @param dir
 * @param beepers
 * @author EE303_Henry
 */
public Athlete(int x, int y, int dir, int beepers) {
    super(x, y, dir, beepers);
    // TODO Auto-generated constructor stub
}
```

```
Athlete(3,4,Display.EAST,Display.INFINITY);
Athlete.Athlete(int x, int y, int dir, int beepers)
Title: Athlete.java
Description:
Parameters:
    x
    y
    dir
    beepers
Author:
    EE303_Henry
```

對 Method 加入註解後：

```
/**
 * <p>Title: turnAround</p>
 * <p>Description: make Athlete turn around</p>
 * @author EE303_Henry
 */
public void turnAround()
{
    turnLeft();
    turnLeft();
}
```

```
karel.turnAround();
void Athlete.turnAround()
Title: turnAround
Description: make Athlete turn around
Author:
    EE303_Henry
```

帶有參數的 Method：

```
/**
 * <p>Title: escapeMaze</p>
 * <p>Description: try to escape the maze </p>
 * @param K
 * @author EE303_Henry
 */
public static void escapeMaze(Athlete K) {
    /**
     * Your code goes here.
     */
}
```

```
escapeMaze(karel);
void Lab00.escapeMaze(Athlete K)
Title: escapeMaze
Description: try to escape the maze
Parameters:
    K
Author:
    EE303_Henry
```

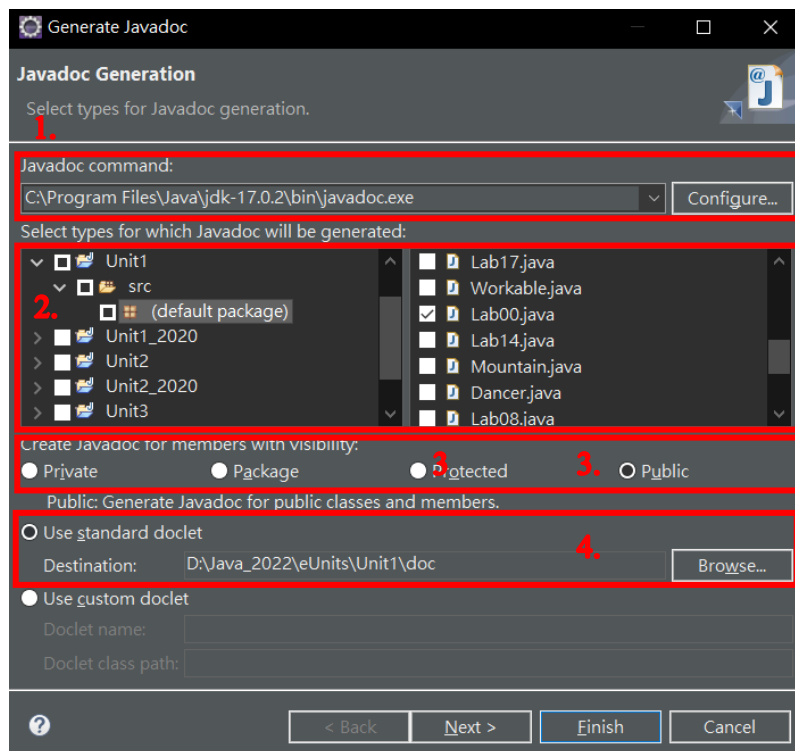
帶有 return 的函式：

```
/**
 * <p>Title: counter</p>
 * <p>Description: return c+1 </p>
 * @param c
 * @return
 * @author EE303_Henry
 */
public static int counter(int c) {
    int a = c + 1;
    return a;
}
```

```
counter(3);
int Lab00.counter(int c)
Title: counter
Description: return c+1
Parameters:
    c
Returns:
Author:
    EE303_Henry
```

或是用「/\*\*」作為開頭和「\*/」作為結尾的方式來標記我們 Javadoc 所用的註解。

產生 Javadoc：在 Eclipse 中，Project > Generate Javadoc 開啟：



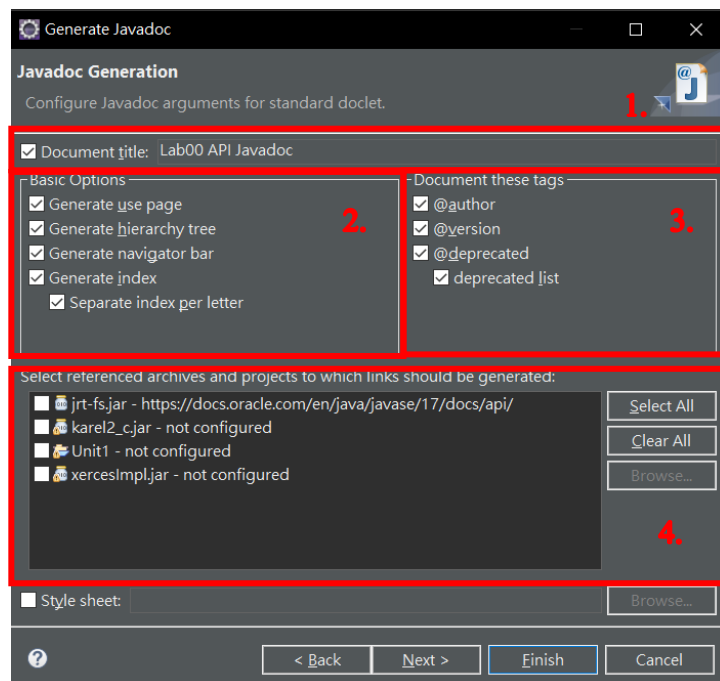
1.Javadoc 檔案在電腦中的位置(通常就是我們環境變數設定的位置)。

2.選擇哪些檔案要生成 Javadoc 檔案。

3.選擇哪些類型的 API 要生成 Javadoc。

4.指定要將 Javadoc 放入哪個目錄。

點擊 Next 可設定更多選項：



1.文檔標題

2.文件結構

3.文件紀錄的標籤

4.生成引用的 JAR 和 project 的連結

一般來說，除了文檔標題(1.)要設定以外，其他參數使用默認的選項即可。

點擊 Next 後可進入更進一步的設定畫面：

The screenshot shows the 'Generate Javadoc' dialog box with the following sections and annotations:

- Overview:** A text field with a 'Browse...' button. Annotated with a red '1'.
- VM options (prefixed with '-J', e.g. -J-Xmx180m for larger heap space):** A text field. Annotated with a red '2'.
- Extra Javadoc options (path names with white spaces must be enclosed in quotes):** A large text area. Annotated with a red '3'.
- JRE source compatibility:** A dropdown menu set to '1.7'.
- Save the settings of this Javadoc export as an Ant script:** A checked checkbox. Annotated with a red '4'.
- Ant Script:** A text field containing 'D:\Java\_2022\Units\Unit1\javadoc.xml' and a 'Browse...' button. Annotated with a red '4'.
- Open generated index file in browser:** A checked checkbox. Annotated with a red '5'.
- Buttons:** '< Back', 'Next >', 'Finish', and 'Cancel'.

- 1.如果有 overview 的文件可以在這邊指定
- 2.使用虛擬機時的選項
- 3.是否有額外的 Javadoc 檔案
- 4.生成 Ant script(強烈建議勾選，因為我們在開發的過程中往往會多次生成 Javadoc)
- 5.使用瀏覽器開啟生成的 Javadoc 文件(選擇此選項幫助我們不用每次 generate 時都重新設定)

執行結果：

PACKAGECLASSUSE TREE INDEX HELP

SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHODSEARCH:

Class Lab00

java.lang.Object<sup>Ⓔ</sup>  
Lab00

public class Lab00  
extends Object<sup>Ⓔ</sup>

Constructor Summary

Constructors

Constructor	Description
Lab00()	

Method Summary

All MethodsStatic MethodsConcrete Methods

Modifier and Type	Method	Description
static int	counter(int c)	Title: counter
static void	escapeMaze(Athlete K)	Title: escapeMaze
static void	main(String <sup>Ⓔ</sup> [] args)	

Methods inherited from class java.lang.Object<sup>Ⓔ</sup>

equals<sup>Ⓔ</sup>, getClass<sup>Ⓔ</sup>, hashCode<sup>Ⓔ</sup>, notify<sup>Ⓔ</sup>, notifyAll<sup>Ⓔ</sup>, toString<sup>Ⓔ</sup>, wait<sup>Ⓔ</sup>, wait<sup>Ⓔ</sup>, wait<sup>Ⓔ</sup>

Constructor Details

Lab00

public Lab00()

Method Details

main

public static void main(String<sup>Ⓔ</sup>[] args)

Parameters:

args -

escapeMaze

public static void escapeMaze(Athlete K)

Title: escapeMaze

Description: try to escape the maze

Parameters:

K -

counter

public static int counter(int c)

Title: counter

Description: return c+1

Parameters:

c -

Returns: