

附件

JDK安裝與設定 (Windows系統)

JDK安裝與環境變數設定 (課堂實作)

- 下載JDK (本課程使用JDK 8，此版本也是目前企業使用率最高的穩定的LTS版本)
 - JDK可自行到相關網站下載，如Oracle JDK或是AdoptOpenJDK
 - 選擇對應的系統平台下載對應的JDK
- 安裝JDK
 - 請預先在C:/ (以WINDOWS系統)下建立資料夾，如jdk1.8.0_XX (XX為版本號碼，注意資料夾名稱勿有空白)
 - 執行JDK安裝程式，並將檔案安裝在剛建立的資料夾裡
- 環境變數設定 (我的電腦→右鍵→內容→進階→環境變數)，並於下方的系統變數→新增或編輯以下三行：
 - **JAVA_HOME** 如：**C:\jdk1.8.0_XX**
 - **Path** 如：**%JAVA_HOME%\bin**
 - **classpath** 如：**.;** 原因課程中說明

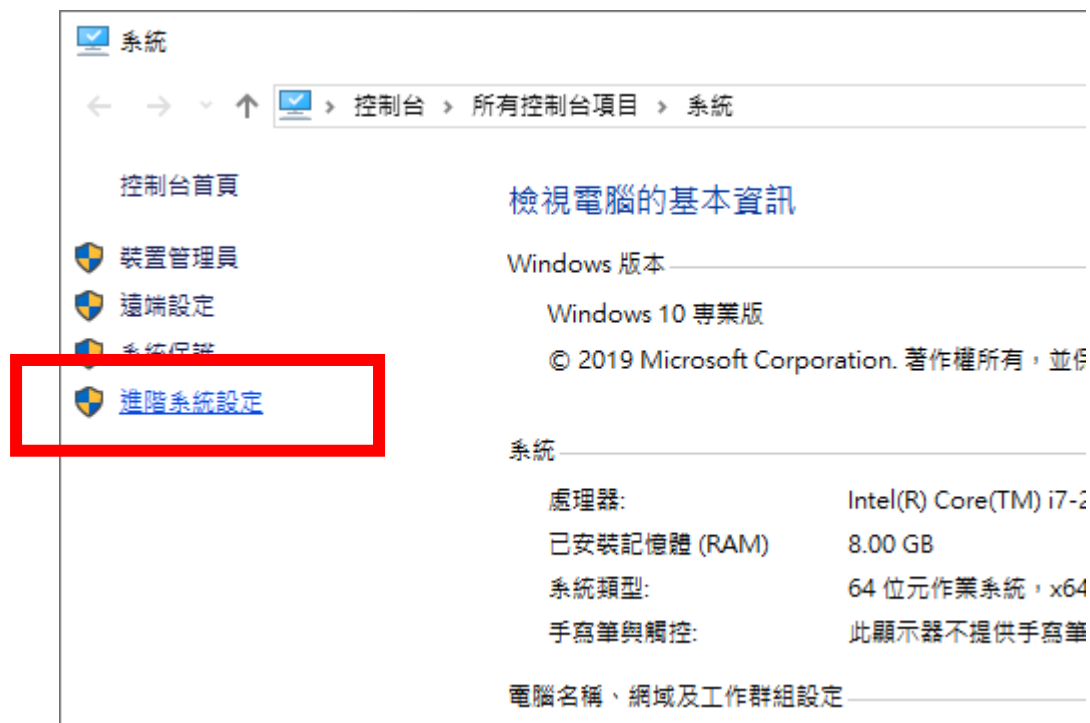
JDK安裝與環境變數設定 (圖解)

- 對著本機右鍵點選”內容”



JDK安裝與環境變數設定 (圖解)

- 點選”進階系統設定”



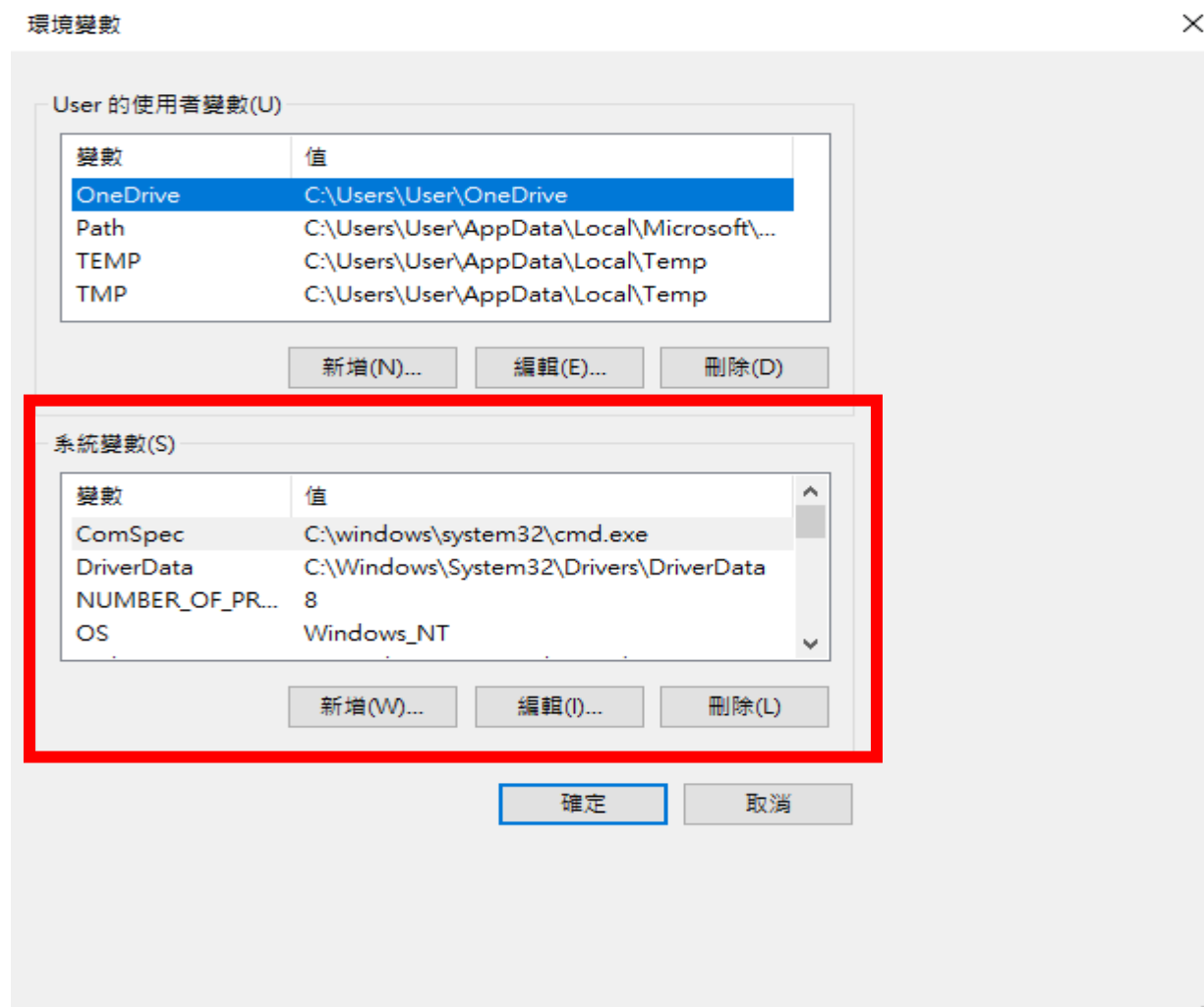
JDK安裝與環境變數設定 (圖解)

- 點選”環境變數”



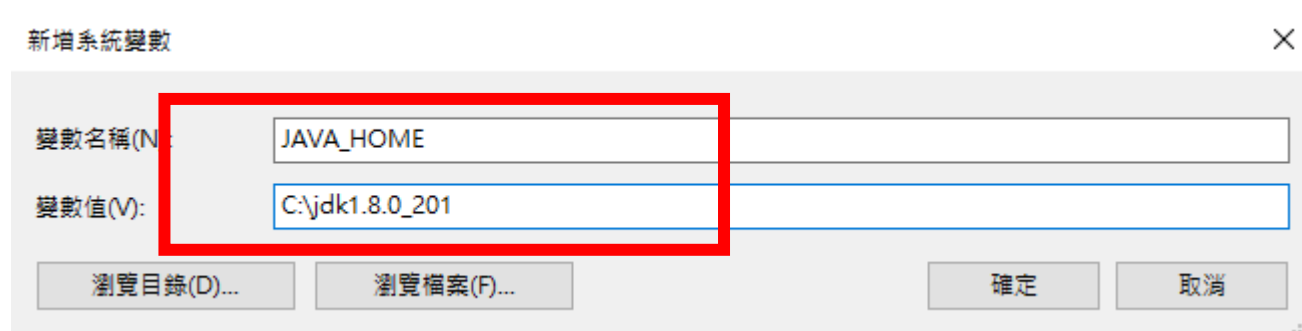
JDK安裝與環境變數設定 (圖解)

- 對"系統變數"做設定



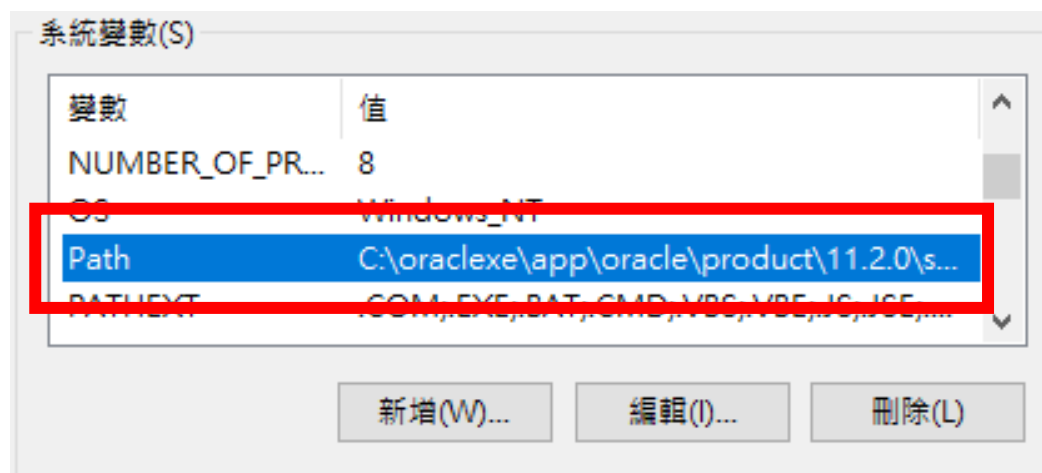
JDK安裝與環境變數設定 (圖解)

- 1. 點選”新增”
- 2. 變數名稱請輸入：JAVA_HOME
- 3. 變數值請輸入JDK安裝的資料夾路徑，如C:\jdk1.8.0_201



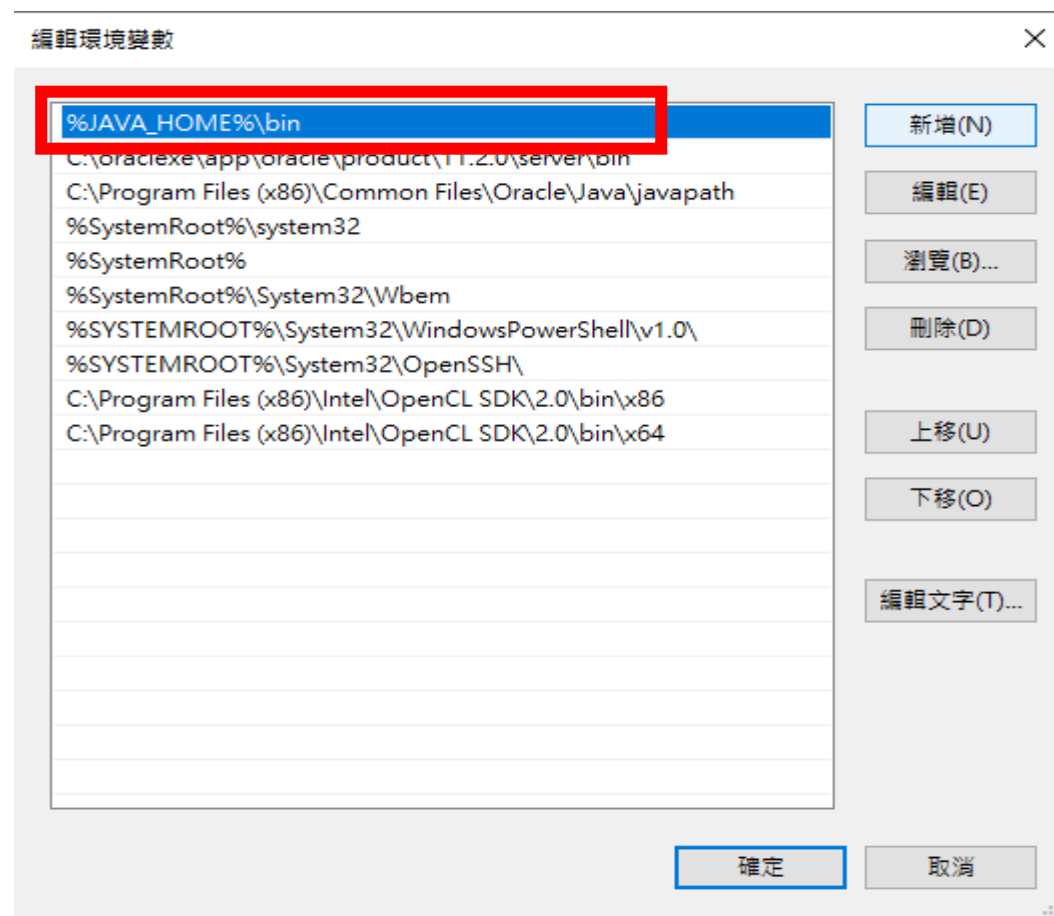
JDK安裝與環境變數設定 (圖解)

- 找到Path後, 點選"編輯"



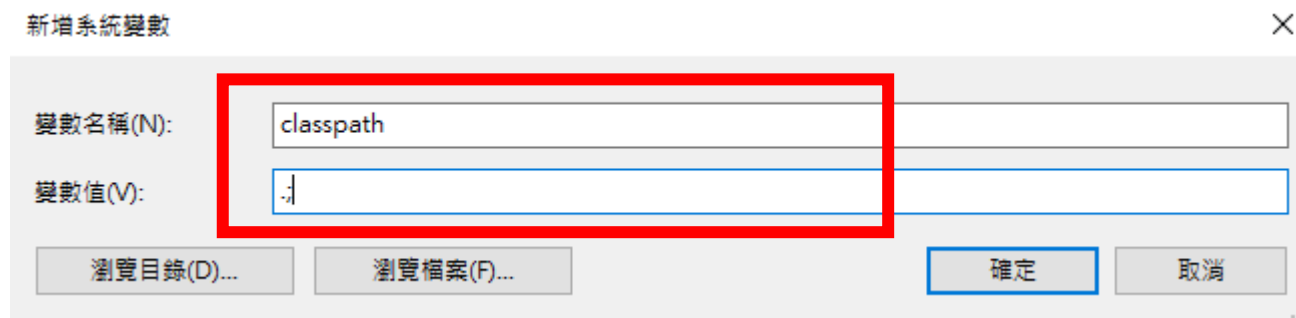
JDK安裝與環境變數設定 (圖解)

- 1. 點選”新增”，並輸入：%JAVA_HOME%\bin
- 2. 把這個路徑透過”上移”，移到第一個位置



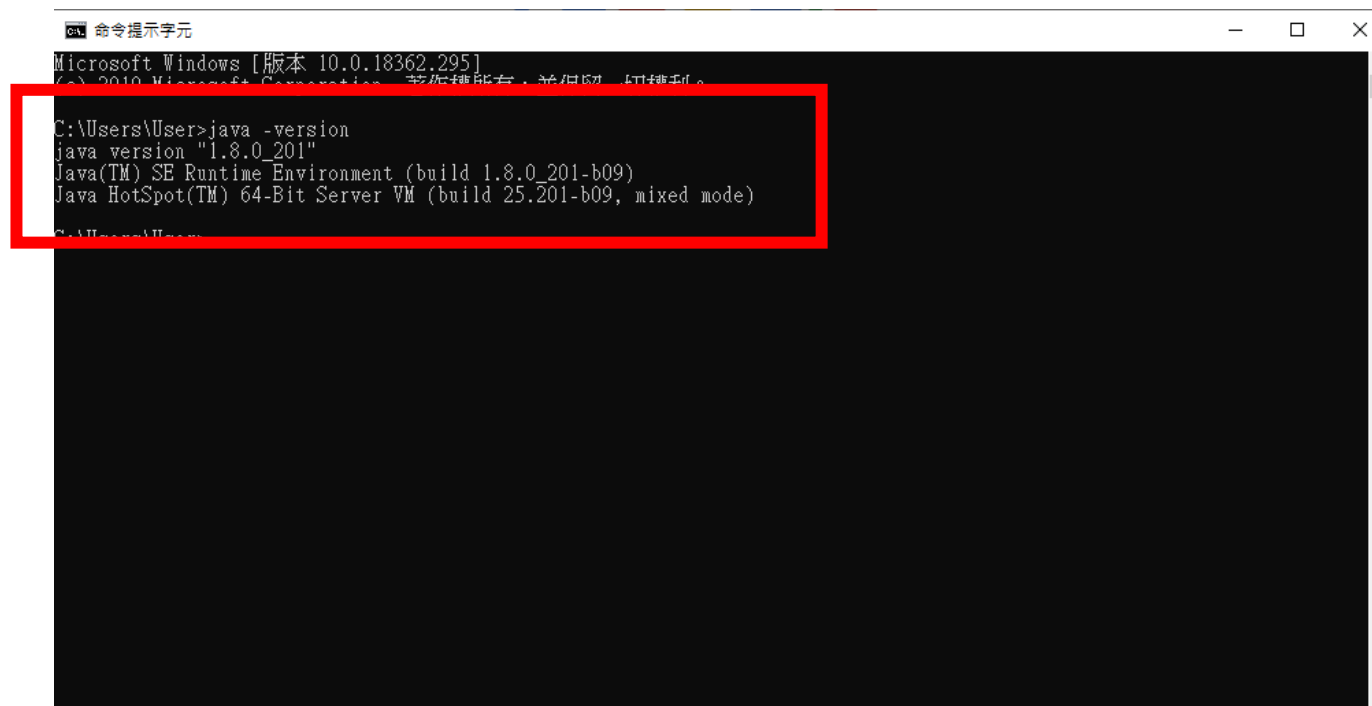
JDK安裝與環境變數設定 (圖解)

- 1. 點選”新增”
- 2. 變數名稱 : classpath
- 2. 變數值 : .; (於介紹類別路徑時說明)



JDK安裝與環境變數設定 (圖解)

- 在cmd工具裡可以下此指令測試java版本資訊
指令 : `java -version`

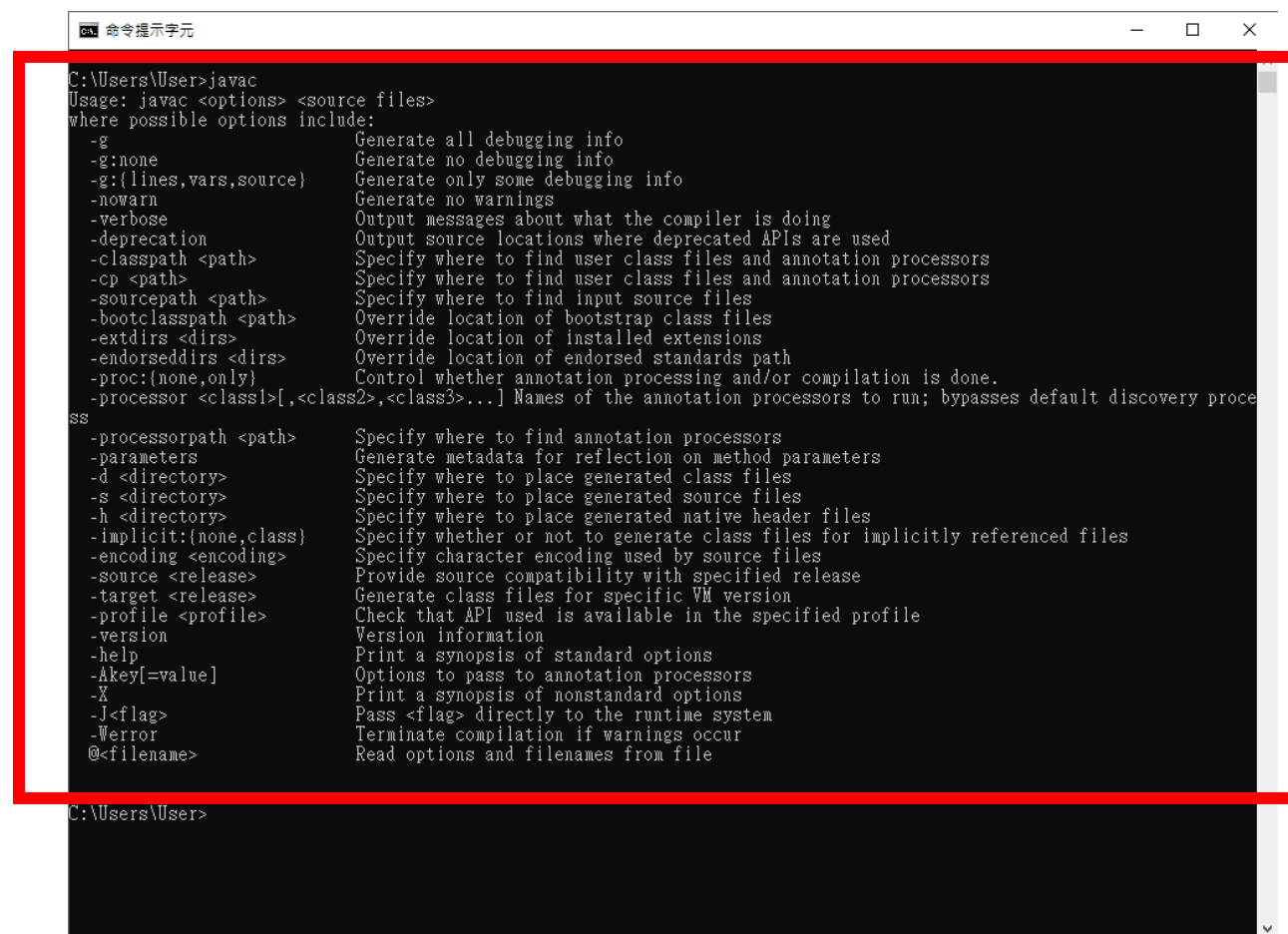


```
命令提示字元
Microsoft Windows [版本 10.0.18362.295]
(c) 2019 Microsoft Corporation。所有權保留。請勿複製。
C:\Users\User>java -version
java version "1.8.0_201"
Java(TM) SE Runtime Environment (build 1.8.0_201-b09)
Java HotSpot(TM) 64-Bit Server VM (build 25.201-b09, mixed mode)
```

JDK安裝與環境變數設定 (圖解)

- 接著下此指令測試環境變數是否設定成功

指令 : javac



```
C:\Users\User>javac
Usage: javac <options> <source files>
where possible options include:
  -g               Generate all debugging info
  -g:none          Generate no debugging info
  -g:{lines,vars,source}  Generate only some debugging info
  -nowarn          Generate no warnings
  -verbose         Output messages about what the compiler is doing
  -deprecation     Output source locations where deprecated APIs are used
  -classpath <path> Specify where to find user class files and annotation processors
  -cp <path>       Specify where to find user class files and annotation processors
  -sourcepath <path> Specify where to find input source files
  -bootclasspath <path> Override location of bootstrap class files
  -extdirs <dirs>   Override location of installed extensions
  -endorseddirs <dirs> Override location of endorsed standards path
  -proc:{none,only} Control whether annotation processing and/or compilation is done.
  -processor <class1>[,<class2>,<class3>...] Names of the annotation processors to run; bypasses default discovery process
  -processorpath <path> Specify where to find annotation processors
  -parameters     Generate metadata for reflection on method parameters
  -d <directory>   Specify where to place generated class files
  -s <directory>   Specify where to place generated source files
  -h <directory>   Specify where to place generated native header files
  -implicit:{none,class} Specify whether or not to generate class files for implicitly referenced files
  -encoding <encoding> Specify character encoding used by source files
  -source <release> Provide source compatibility with specified release
  -target <release> Generate class files for specific VM version
  -profile <profile> Check that API used is available in the specified profile
  -version         Version information
  -help            Print a synopsis of standard options
  -Akey[=value]    Options to pass to annotation processors
  -X              Print a synopsis of nonstandard options
  -J<flag>         Pass <flag> directly to the runtime system
  -Werror          Terminate compilation if warnings occur
  @<filename>      Read options and filenames from file
```