

## JAVA Lab0

### Step 1 : download java jdk

JDK9:

<http://www.oracle.com/technetwork/java/javase/downloads/jdk9-downloads-3848520.html>

- Click “Accept License Agreement”

### Java SE Development Kit 9.0.4

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

☒ Accept License Agreement ☐ Decline License Agreement

Product / File Description	File Size	Download
Linux	305.07 MB	<a href="#">jdk-9.0.4_linux-x64_bin.rpm</a>
Linux	338.21 MB	<a href="#">jdk-9.0.4_linux-x64_bin.tar.gz</a>
macOS	382.11 MB	<a href="#">jdk-9.0.4_osx-x64_bin.dmg</a>
Windows	375.56 MB	<a href="#">jdk-9.0.4_windows-x64_bin.exe</a>
Solaris SPARC	206.97 MB	<a href="#">jdk-9.0.4_solaris-sparcv9_bin.tar.gz</a>

- Choose your os (example: Windows)

### Java SE Development Kit 9.0.4

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download this software.

Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.

Product / File Description	File Size	Download
Linux	305.07 MB	<a href="#">jdk-9.0.4_linux-x64_bin.rpm</a>
Linux	338.21 MB	<a href="#">jdk-9.0.4_linux-x64_bin.tar.gz</a>
macOS	382.11 MB	<a href="#">jdk-9.0.4_osx-x64_bin.dmg</a>
Windows	375.56 MB	<a href="#">jdk-9.0.4_windows-x64_bin.exe</a>
Solaris SPARC	206.97 MB	<a href="#">jdk-9.0.4_solaris-sparcv9_bin.tar.gz</a>

## Step 2: install java jdk

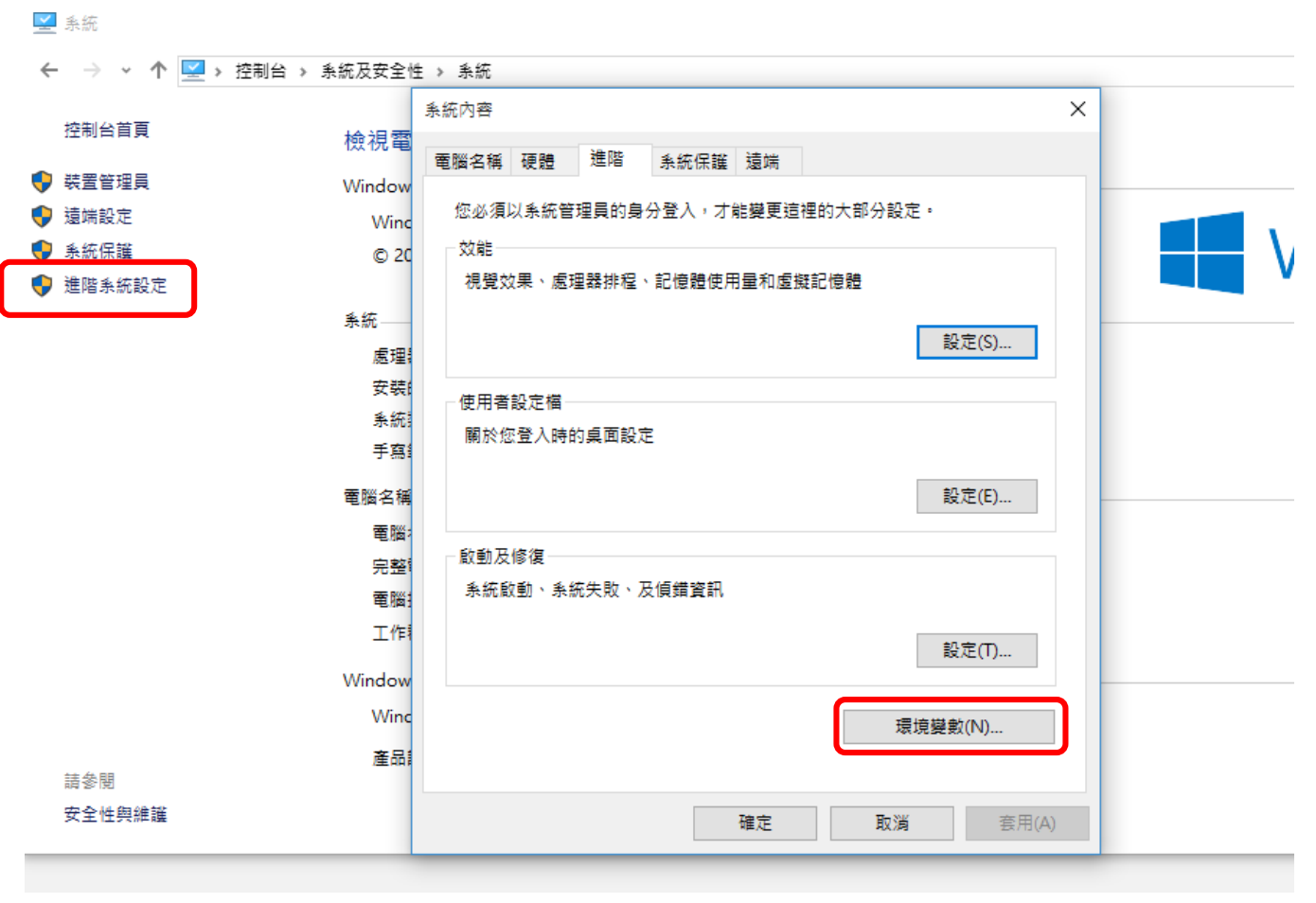
安裝完成後，請記得安裝的位置，然後找到 bin file

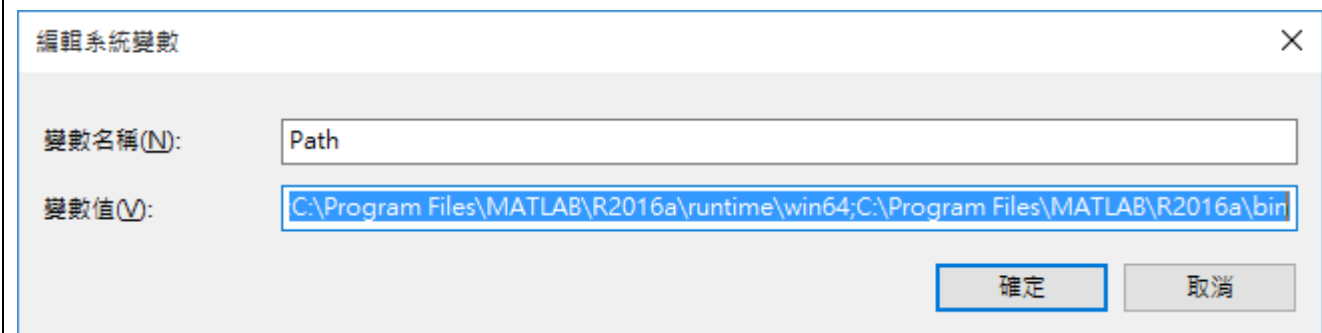
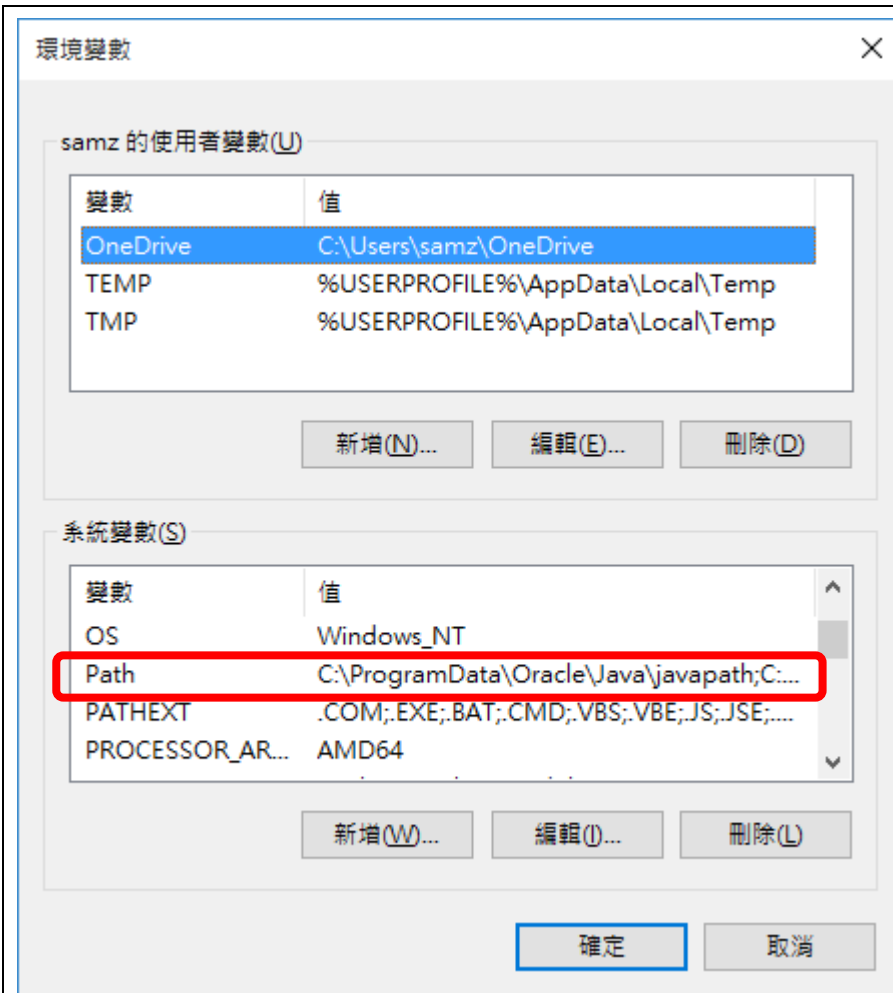
如: C:\Program Files\Java\jdk-9.0.4\bin

## Step 3: Setting Path

從控制台中找到系統

- 控制台\系統及安全性\系統 -> 進階系統設定 -> 環境參數





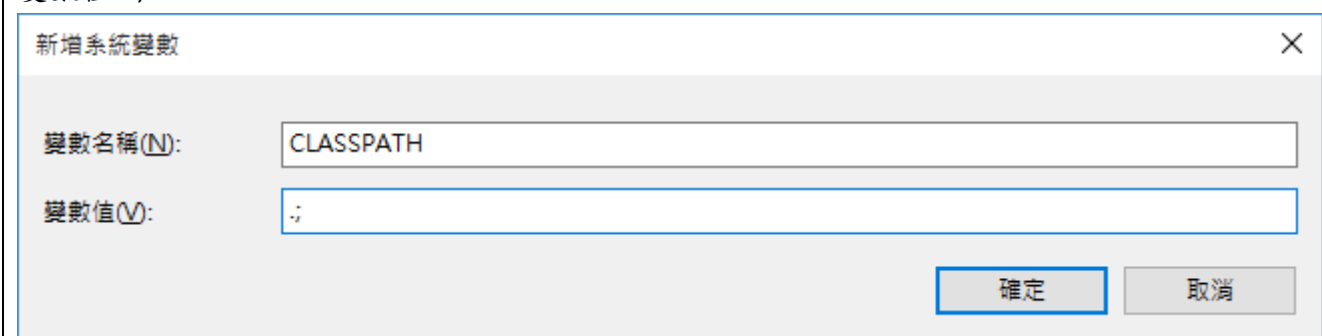
- 在變數值的最末端打上 C:\Program Files\Java\jdk-9.0.4\bin

注意:不要刪到其它的文字,然後如果原先最末端沒有";"記得打上";"(為了分隔其它程式的 path)

- 選定"新增" ->並加入下列數值

變數名稱: CLASSPATH

變數值: .;



## Step 4: testing

桌面：windows 鍵 + R -> 輸入 “cmd”

在命令提示字元輸入 java 和 javac，有下列訊息代表成功

```
C:\WINDOWS\system32\cmd.exe
C:\Users\samz>java
用法: java [-options] class [args...]
      (執行類別)
或 java [-options] -jar jarfile [args...]
      (執行 jar 檔案)

選項包括:
  -d32          使用 32 位元資料模型 (如果有的話)
  -d64          使用 64 位元資料模型 (如果有的話)
  -server       選取 "server" VM
                預設的 VM 為 server.

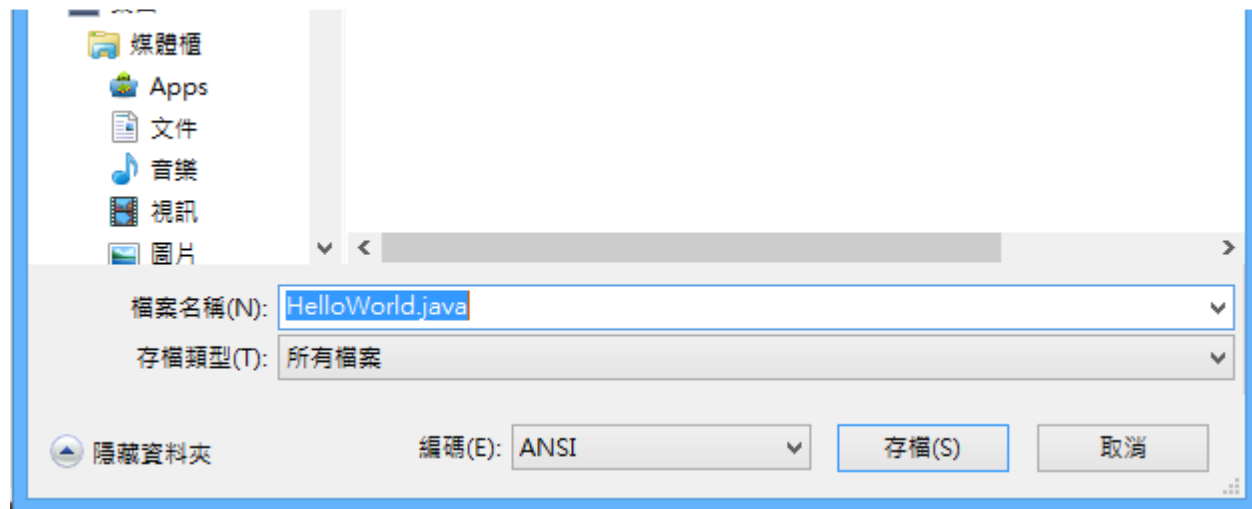
  -cp <目錄及 zip/jar 檔案的類別搜尋路徑>
  -classpath <目錄及 zip/jar 檔案的類別搜尋路徑>
                使用 ; 區隔的目錄、JAR 存檔以及 ZIP 存檔清單來搜尋類別檔案。
  -D<name>=<value>
                設定系統屬性
  -verbose:[class|gc|jni]
                啟用詳細資訊輸出
  -version      列印產品版本並結束
  -version:<value>
                警告: 此功能已不再使用，將會從未來版本中移除。
                需要指定的版本才能執行
  -showversion  列印產品版本並繼續
  -jre-restrict-search | -no-jre-restrict-search
                警告: 此功能已不再使用，將會從未來版本中移除。
                在版本搜尋中包括/排除使用者專用 JRE
  -? -help      列印此說明訊息
  -X            列印非標準選項的說明
  -ea[:<package>...[:<classname>]]
  -enableassertions[:<package>...[:<classname>]]
                啟用含指定詳細程度的宣告
  -da[:<package>...[:<classname>]]
  -disableassertions[:<package>...[:<classname>]]
                停用含指定詳細程度的宣告
  -esa | -enablesystemassertions
                啟用系統宣告
微軟注音 半 :
```

```
C:\WINDOWS\system32\cmd.exe
C:\Users\samz>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 <class>[,<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
微軟注音 半 :
```

## Step5: First JAVA Code

在一個資料夾底下建立一個檔案 HelloWorld.java

注意:如果是用新增文字的方式建立,請將存檔類型改成所有檔案



開啟它並打上：

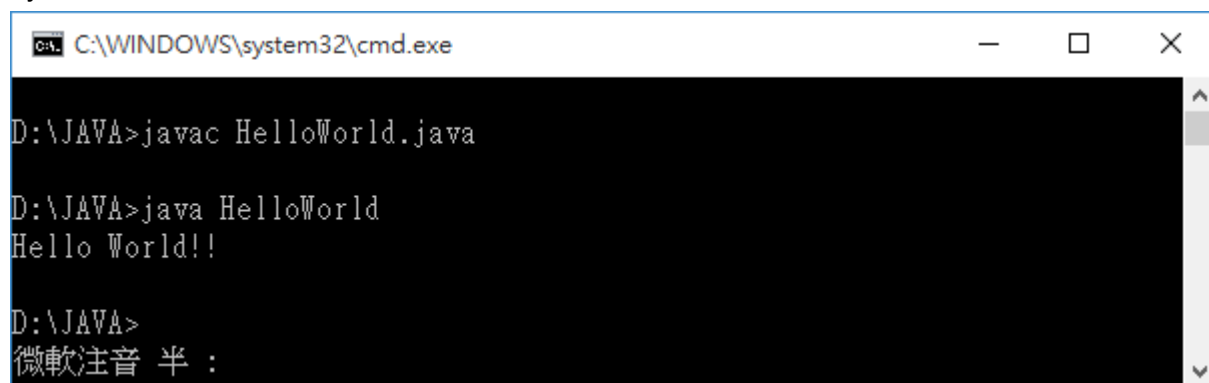
```
public class HelloWorld{  
    public static void main(String args[]){  
        System.out.println("Hello World!!");  
    }  
}
```

接著在當下的資料匣內的空白處按住shift加上滑鼠右鍵，點選在此處開啟命令提示視窗。

- javac HelloWorld.java

此時你會發現當下的資料匣出現HelloWorld.class檔

- java HelloWorld



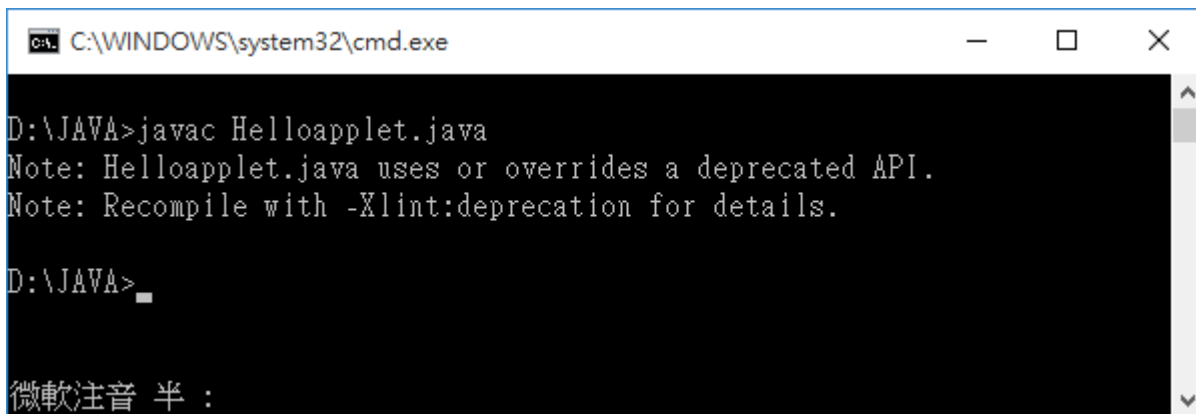
## JAVA Applet code

- 和剛剛一樣，在同一個資料夾底下建立一個叫HelloApplet.java的檔案

```
import java.applet.Applet;
import java.awt.*;

public class HelloApplet extends Applet{
    public void paint(Graphics g){
        g.drawString("Hello Applet",40,10);
    }
}
```

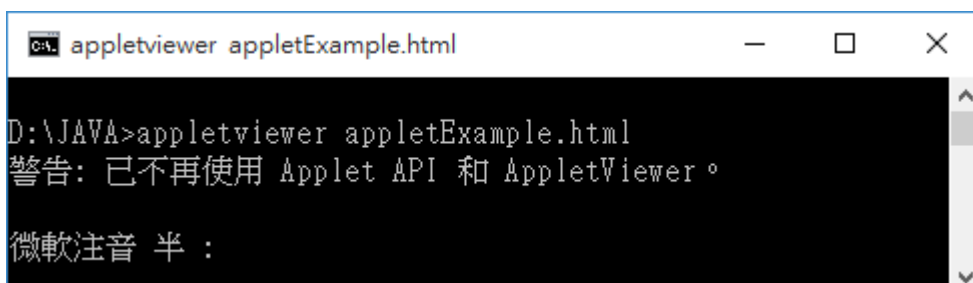
- 接著自己編譯成class檔



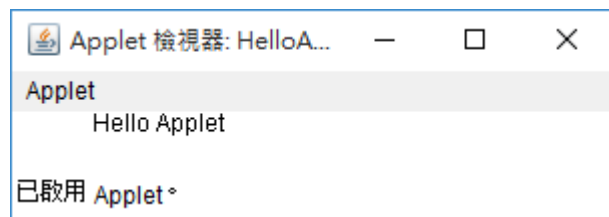
- 在同一個資料匣底下建立一個叫appletEaxmple.html的檔案

```
<html>
<body>
<applet code="HelloApplet.class" height="30" width="300">
</applet>
</body>
</html>
```

在命令提示視窗下執行：



產生下面結果則成功



這是 java 所有官方的 class，以後有想用的 class 不知道怎麼用，請上這個

**JDK8:**

<https://docs.oracle.com/javase/8/docs/api/>

**JDK9:**

<https://docs.oracle.com/javase/9/docs/api/overview-summary.html>