

A p p e n d i x

# B

## JDK 安裝與 EJE 開發工具的使用

### 本章學習目標

B-1 建立 Java 語言的開發環境

B-2 使用 EJE 建立 Java 程式

B-3 EJE 的使用介面與基本使用

## B-1 建立 Java 語言的開發環境

Java 語言的開發環境需要安裝 JDK，然後配合編輯工具或整合開發環境來建立 Java 程式，在本書是使用 EJE 建立 Java 程式。

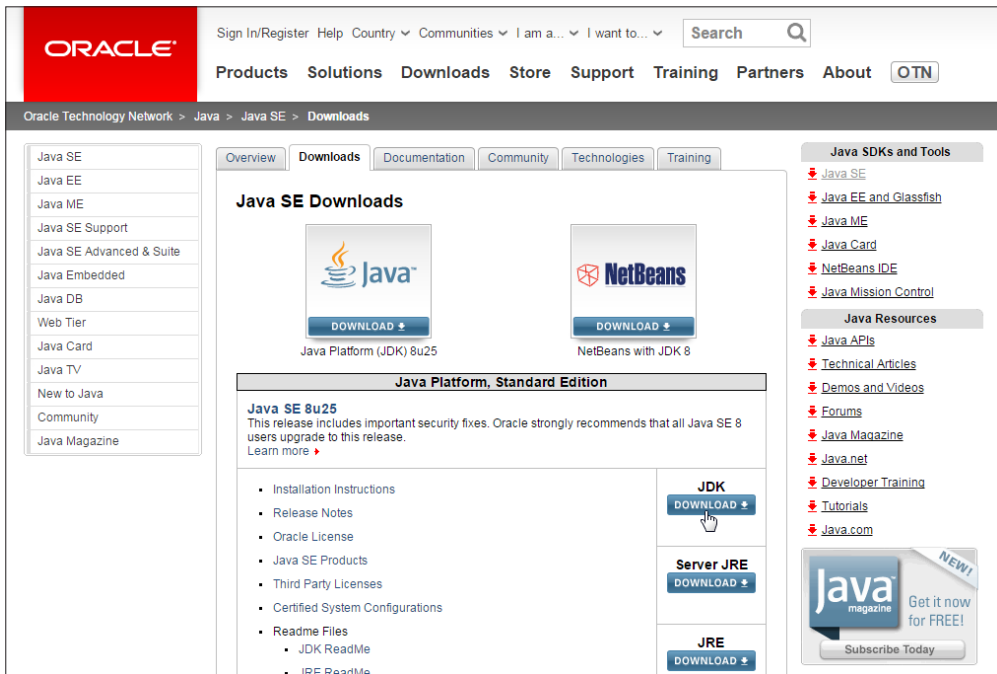
### B-1-1 下載與安裝 JDK

一般來說，部分 Java 整合開發環境會一併安裝 JDK，不過，大部分整合開發環境都需要自行安裝 JDK，在本書使用的版本是 JDK 8 (Java SE Development Kit 8)。












#### 下載 JDK

JDK 可以從網路上免費下載，其下載網址如下所示：

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>



按 JDK 下方游標所在的 **DOWNLOAD** 鈕，選 **Accept License Agreement** 後，可以在下方看到下載的超連結，如下圖所示：

Java SE Development Kit 8u25		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
<input checked="" type="radio"/> Accept License Agreement <input type="radio"/> Decline License Agreement		
Product / File Description	File Size	Download
Linux x86	135.24 MB	 <a href="#">jdk-8u25-linux-i586.rpm</a>
Linux x86	154.88 MB	 <a href="#">jdk-8u25-linux-i586.tar.gz</a>
Linux x64	135.6 MB	 <a href="#">jdk-8u25-linux-x64.rpm</a>
Linux x64	153.42 MB	 <a href="#">jdk-8u25-linux-x64.tar.gz</a>
Mac OS X x64	209.13 MB	 <a href="#">jdk-8u25-macosx-x64.dmg</a>
Solaris SPARC 64-bit (SVR4 package)	137.01 MB	 <a href="#">jdk-8u25-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	97.14 MB	 <a href="#">jdk-8u25-solaris-sparcv9.tar.gz</a>
Solaris x64 (SVR4 package)	137.11 MB	 <a href="#">jdk-8u25-solaris-x64.tar.Z</a>
Solaris x64	94.24 MB	 <a href="#">jdk-8u25-solaris-x64.tar.gz</a>
Windows x86	157.26 MB	 <a href="#">jdk-8u25-windows-i586.exe</a>
Windows x64	169.62 MB	 <a href="#">jdk-8u25-windows-x64.exe</a>

點選最下方 **jdk-8u25-windows-i586.exe** 超連結，可以下載 32 位元版本；**jdk-8u25-windows-x64.exe** 超連結是 64 位元版本的 JDK。

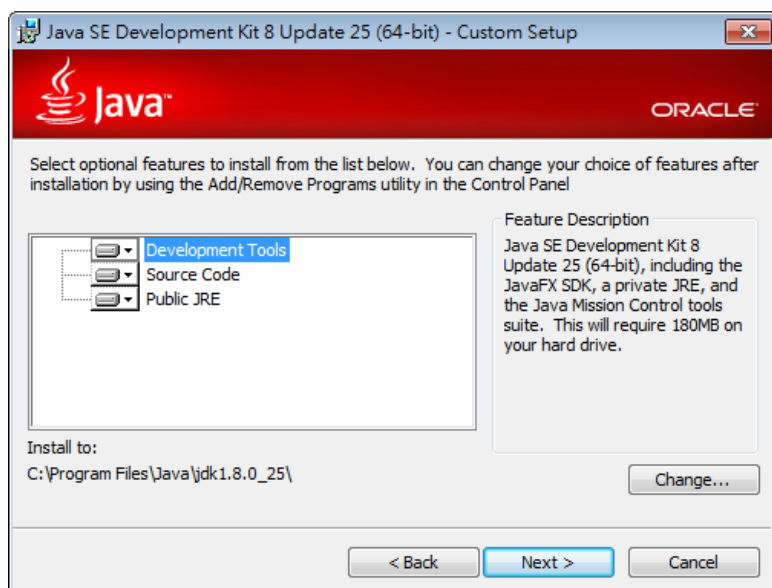
## 安裝 JDK

JDK 8 更新版本並不是使用「.」後的次版號來表示，而是改為更新 Update 來標示，以本書為例是使用 JDK 8 Update 25，其下載 Windows 版 JDK 的檔名為 **jdk-8u25-windows-i586.exe**；64 位元是 **jdk-8u25-windows-x64.exe**，在

**Step 1** 在本書是以 64 位元版本和 Windows 7 為例，請按二下 **jdk-8u25-windows-x64.exe** 程式檔案，如果看到「使用者帳戶控制」對話方塊，請按是鈕繼續，可以看到歡迎安裝的精靈畫面。



**Step 2** 按 **Next** 鈕，選擇 JDK 安裝元件和路徑。



**Step 3** 預設安裝路徑是「C:\Program Files\Java\jdk1.8.0\_25\」，不用更改，請按 **Next** 鈕開始安裝 JDK，可以看到目前的安裝進度。

**Step 4** 請稍等數分鐘，等到 JDK 安裝完成，我們需要選擇安裝 Java SE Runtime Environment (即 JVM) 的目的路徑。



**Step 5** 不用更改，按下一步鈕繼續。稍等一下，等到完成安裝和設定 JVM 後，可以看到完成安裝的精靈畫面。



**Step 6** 請按 Close 鈕完成 JDK 的安裝。

## B-1-2 Java 程式碼編輯器 - EJE

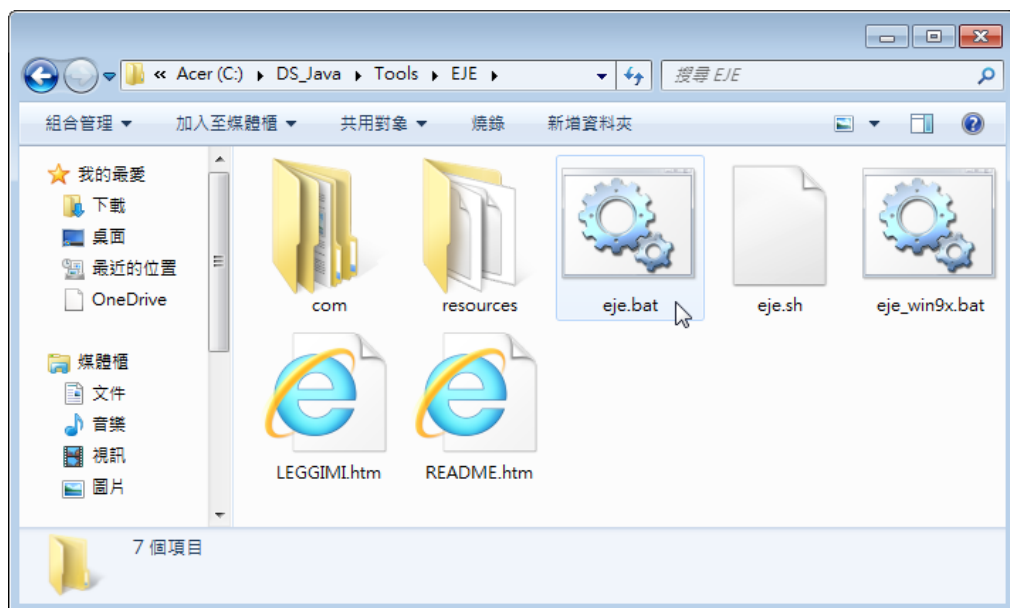
EJE (Everyone's Java Editor) 是由 Claudio De Sio Cesari 開發，一套輕量且簡單好用的 Java 程式碼編輯器，不同於其他重量級 Java 整合開發環境，例如：Eclipse IDE、NetBean、IntelliJ IDEA，使用者需要花費相當時間先學習工具的使用，才能開始撰寫 Java 程式；EJE 介面簡單且直覺，幾乎不用學習，一啟動就可以馬上開始撰寫 Java 程式。

因為 EJE 本身是使用 Java 開發，可以跨平台支援 Windows 和 Linux 等作業系統，其高親和力的使用介面，非常適合開始學習 Java 語言的使用者或學生來使用。再加上筆者已經將使用介面完全中文化，EJE 可以說是目前有史以來最簡單的中文版 Java 程式碼編輯器和開發環境，其主要特點如下所示：

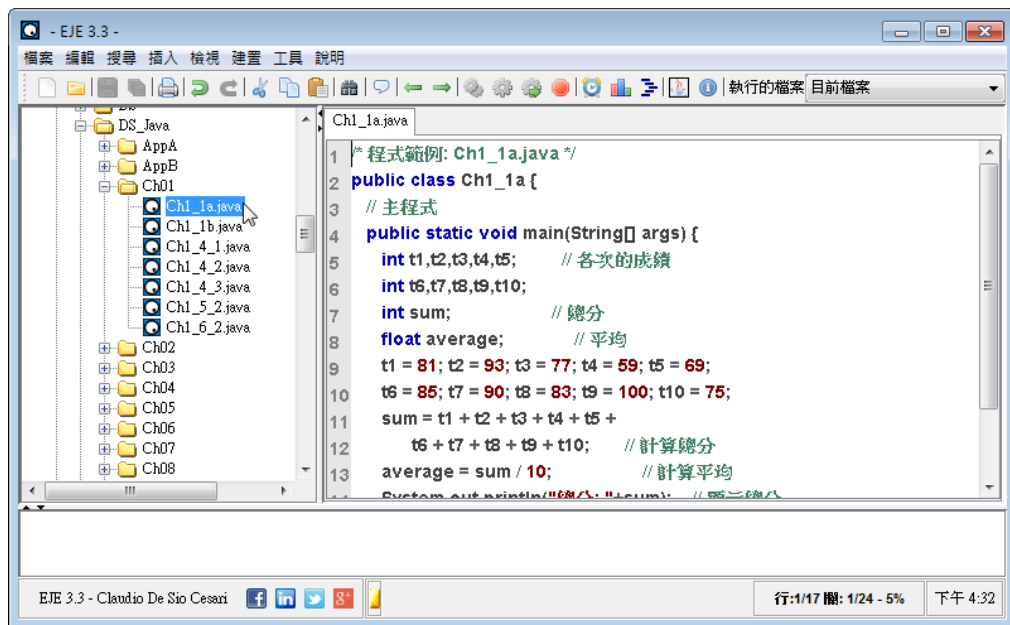
- EJE 不只是程式碼編輯器，更可以在同一工具編譯和執行 Java 程式，和使用參數來執行程式，並且支援最新版 Java 8。
- EJE 是使用色彩標示程式語法，可以使用標籤頁來同時編輯多個程式檔案。
- EJE 提供檔案總管視窗可以快速找到和開啟 Java 程式檔案，和支援多個工作目錄（擁有 Java 程式的目錄）。
- EJE 提供功能表指令來插入程式片段的範本，例如：條件、迴圈、主程式、類別、介面、建構子和註解等。
- EJE 支援自動程式語法提示功能，可以彈出視窗來顯示相關類別、物件、方法和屬性清單。

### 啟動 Java 編輯器 EJE

EJE 並不需要安裝，請直接解壓縮和複製至指定目錄即可，例如：「C:\ADS\_Java\Tools\EJE」，如下圖所示：



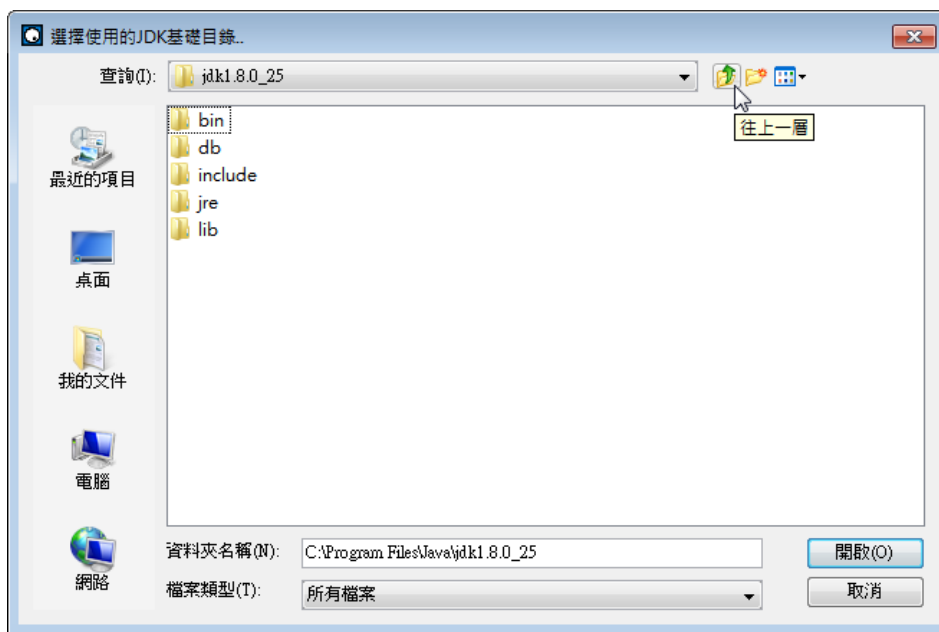
請按二下 **eje.bat** 啟動 Java 編輯器 EJE，稍等一下，可以看到 EJE 編輯器的執行畫面，如下圖所示：



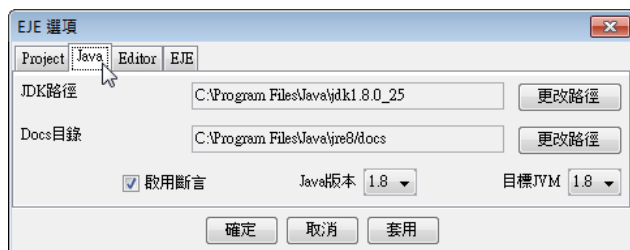
上述使用介面非常簡潔，上方是功能表和工具列，左邊是檔案總管視窗，可以用來檢視擁有 Java 程式的目錄，例如：展開「D:\ADS\_Java\Ch01」目錄，可以看到 Java 程式檔，點選檔名即可開啟 Java 程式來編輯、編譯和執行。結束 EJE 請執行「檔案/結束」指令。

## 設定 EJE 的 JDK 安裝路徑

EJE 在第一次啟動時，如果沒有指定 JDK 路徑，預設會開啟「選擇使用的 JDK 基礎目錄」對話方塊來指定 JDK 安裝路徑，如下圖所示：



請切換至 JDK 安裝路徑「C:\Program Files\Java\jdk1.8.0\_25\」後，按開啟鈕設定 JDK。如果已經啟動 EJE，我們可以執行「檔案/選項設定」指令來設定 JDK，可以看到「EJE 選項」對話方塊。





選 **Java** 標籤，可以在上方更改 JDK 路徑和 Docs 目錄，下方選擇 Java 版本和目標 JVM 版本，1.8 是 Java 8。請注意！EJE 必須正確指定 Docs 目錄，才能在輸入程式碼時顯示物件屬性和方法清單的提示訊息。

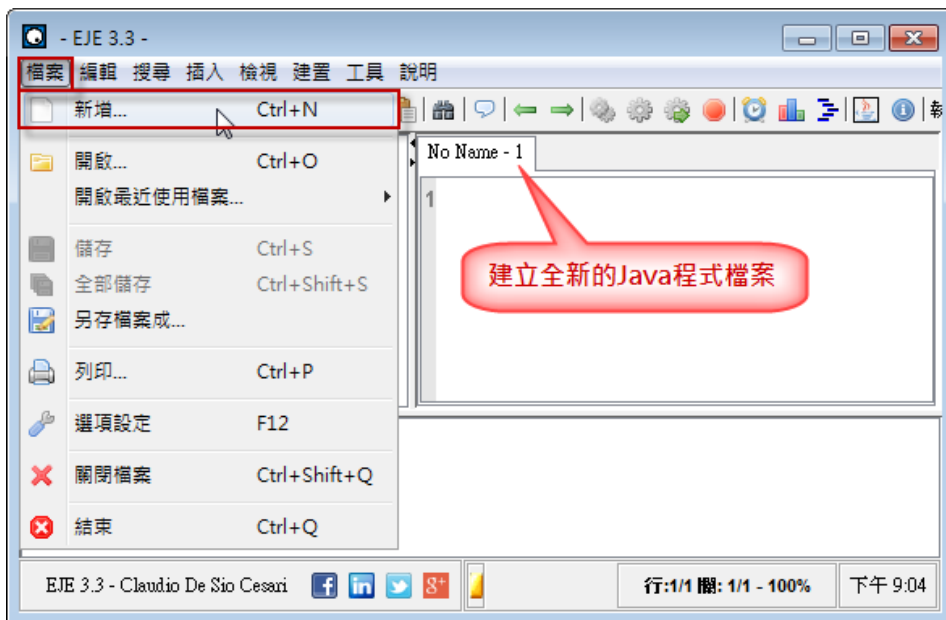
## B-2 使用 EJE 建立 Java 程式

EJE (Everyone's Java Editor) 是一套簡單但功能足夠的 Java 程式碼編輯器，可以在同一工具編輯、編譯和執行 Java 程式，再加上 EJE 支援功能表指令來插入程式片段，幾乎不用打幾個字，就可以快速建立第一個 Java 程式。

### 步驟一：啟動 EJE 新增 Java 程式

EJE 是以擁有 Java 程式檔的目錄作為工作目錄來管理 Java 程式，在啟動 EJE 後，就可以直接新增空白內容的全新 Java 程式檔，其步驟如下所示：

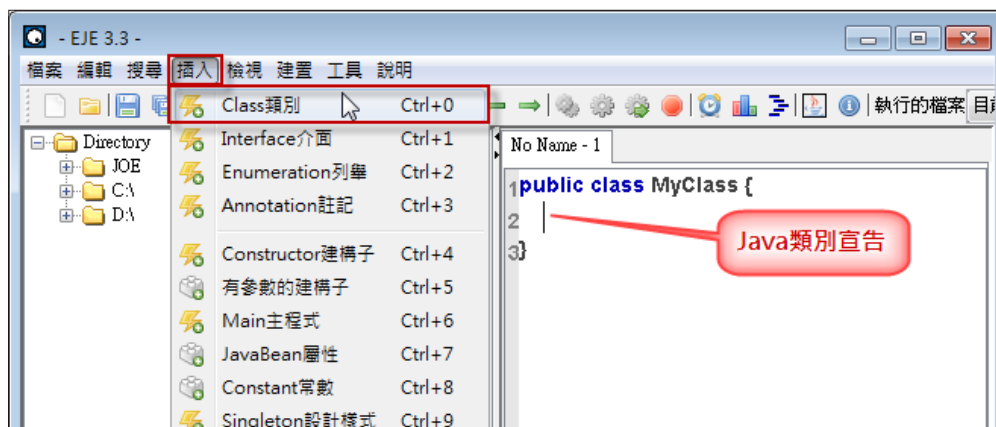
**Step 1** 請啟動 EJE，執行「檔案/新增」指令建立全新 Java 程式檔案，因為尚未儲存，所以標籤是 No Name。



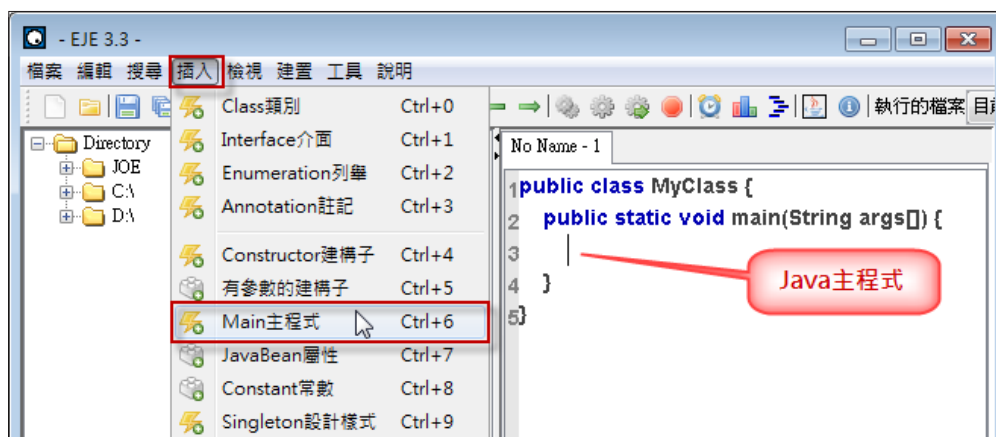
## 步驟二：在 EJE 編輯 Java 程式碼

現在，我們可以在 EJE 右邊標籤頁編輯 Java 程式碼，讀者可以自行使用鍵盤輸入程式碼，或使用功能表指令來快速插入 Java 程式片段。請繼續上面步驟，如下所示：

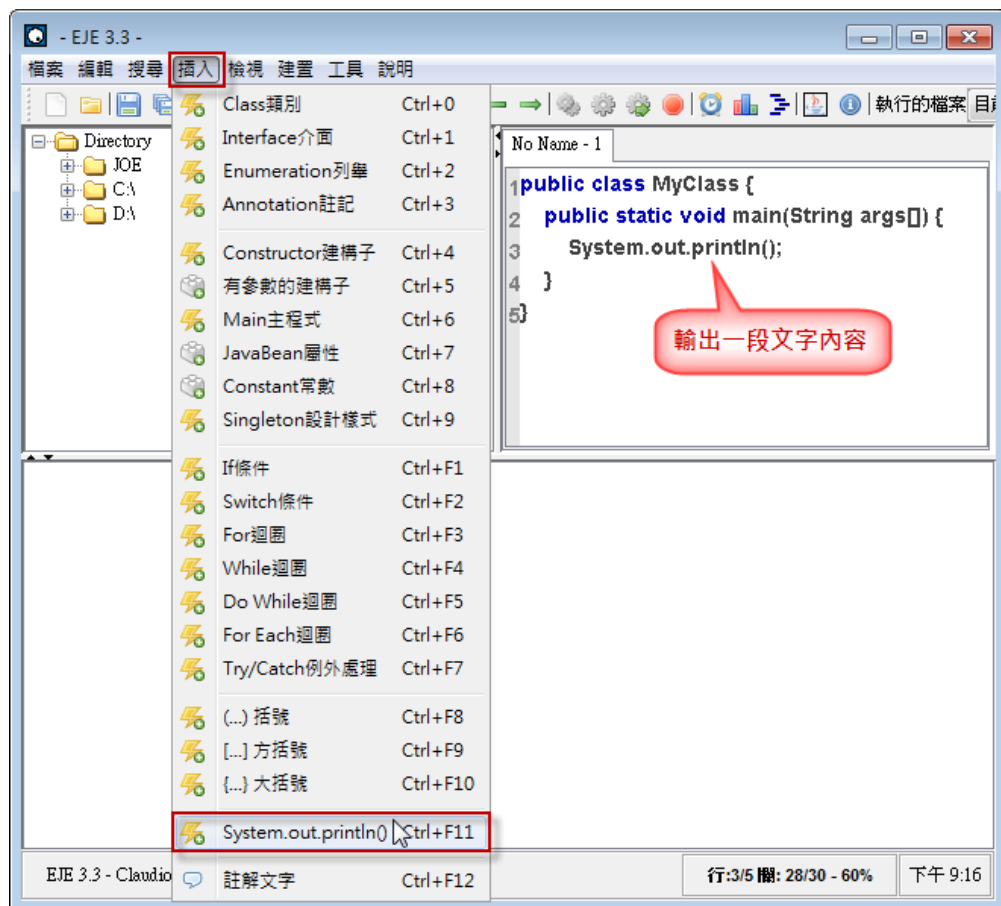
**Step 1** 請執行「插入/Class 類別」指令，可以插入 Java 類別宣告，此時游標插入點是位在類別宣告之中。



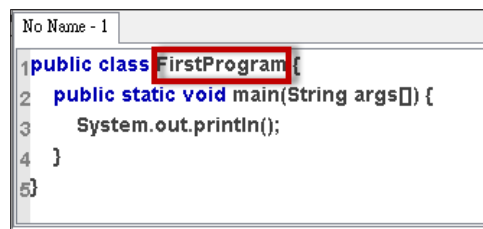
**Step 2** 執行「插入/Main 主程式」指令插入 Java 主程式 main()，可以看到現在游標插入點是位在主程式的程式區塊中。



**Step 3** 執行「插入/System.out.println()」指令插入方法來輸出字串內容。



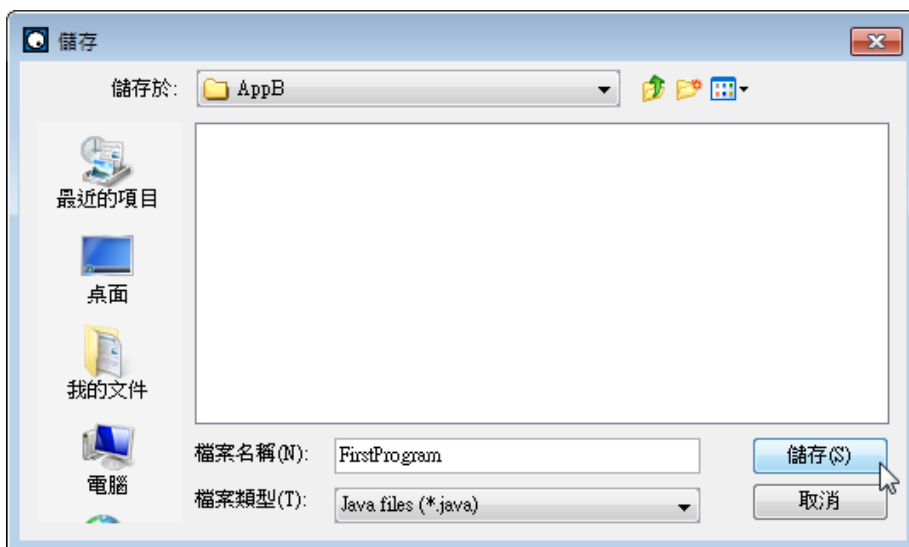
**Step 4** 請更改類別名稱，從 MyClass 改為 FirstProgram。



**Step 5** 在 `System.out.println()` 鍵入參數的輸出字串內容「第一個 Java 程式」。

```
No Name - 1
1 public class FirstProgram {
2     public static void main(String args[]) {
3         System.out.println("第一個Java程式");
4     }
5 }
```

**Step 6** 執行「檔案/儲存」指令，可以開啟「儲存」對話方塊。

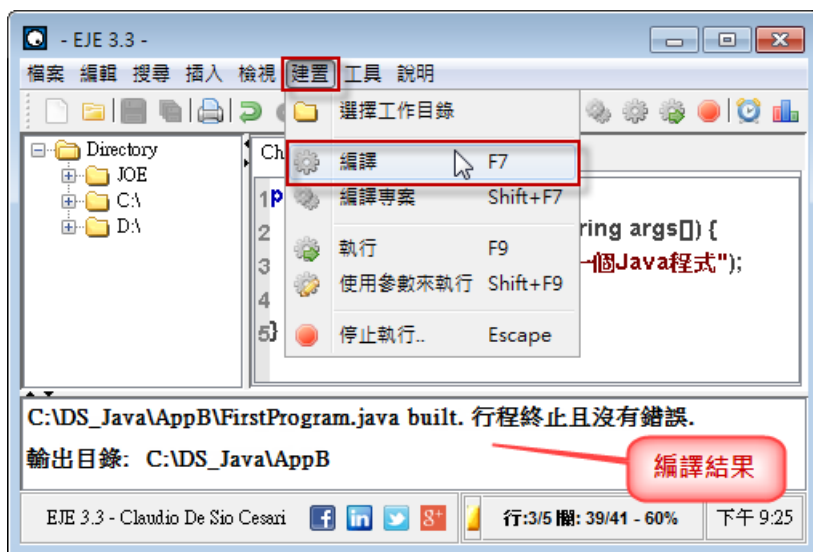


**Step 7** 請切換至儲存的目錄後，在**檔案名稱**欄輸入檔案名稱 `FirstProgram`（預設自動填入和類別相同的名稱），按**儲存**鈕儲存 Java 程式檔。

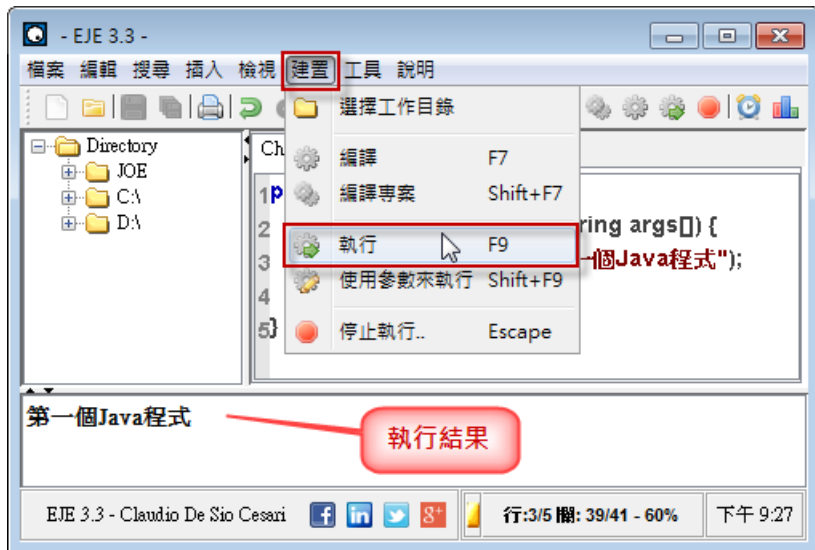
### 步驟三：編譯和執行 Java 程式

在完成 Java 程式編輯後，就可以在 EJE 編譯和執行 Java 程式。請繼續上面步驟，如下所示：

**Step 1** 請執行「**建置/編譯**」指令或按 **F7** 鍵編譯 Java 程式，如果沒有錯誤，可以在下方顯示成功編譯的訊息文字。



**Step 2** 執行「建置/執行」指令或按 **F9** 鍵執行 Java 程式，可以在下方看到執行結果顯示的字串內容。



如果 EJE 在執行過程出現問題，一直看不到執行結果且沒有回應，請執行「建置/停止執行」指令停止程式執行後，再試一次。

## 說明

請注意！EJE 可以在下方視窗輸入 Java 程式所需的資料，不過，輸入資料並不支援中文內容，而且只允許輸入英文小寫字母，如果輸入英文大寫字母或是部分符號字元，例如：「+」號時，輸入的字串會多出「?」符號。

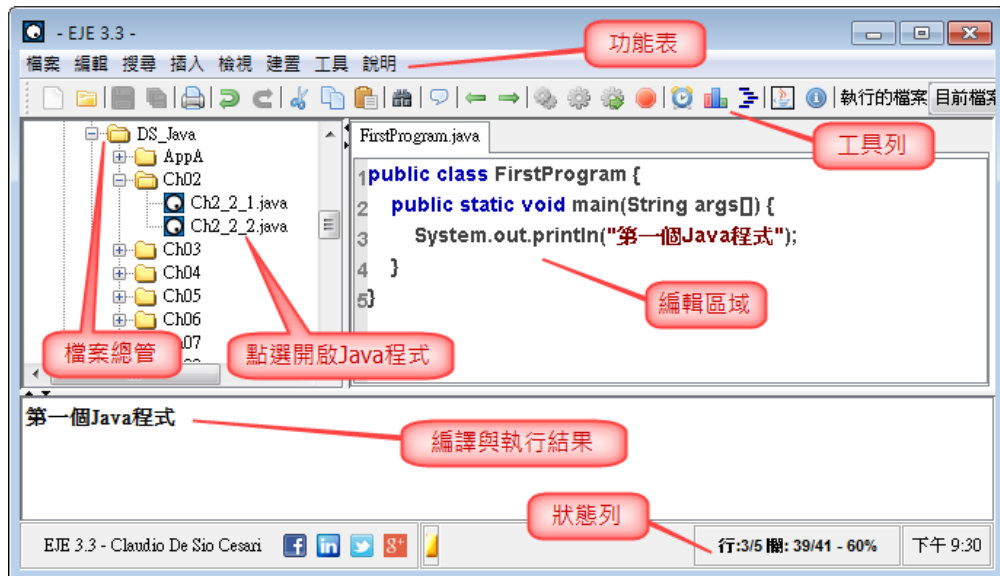
如果在輸入資料時發生問題，而且執行「**建置/停止執行**」指令仍然無法解決時，請結束 EJE，重新啟動 EJE 來解決此問題。

## B-3 EJE 的操作介面與基本使用

Java 編輯器 EJE 的主要目的是幫助我們學習 Java 程式設計，可以讓開發者專注於程式語法的學習，而不用學習複雜的 IDE，所以使用介面十分簡潔，不過，EJE 仍然提供開發 Java 程式所需的主要功能。

### B-3-1 EJE 的使用介面

EJE 使用介面十分簡潔，並沒有多餘的視窗，在啟動 EJE 後，就會進入 EJE 主視窗的 Java 程式開發介面，如下圖所示：



上述使用介面的上方是功能表，之下是工具列，左邊是檔案總管視窗，可以切換路徑顯示 Java 檔案清單，點選檔案名稱，就可以在右邊編輯區域，以標籤頁方式來開啟 Java 檔案。

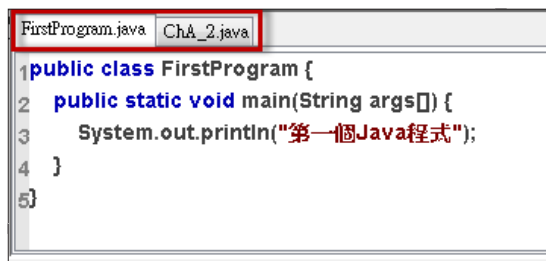
在下方區域可以顯示編譯和執行結果，如果編譯有錯誤，錯誤或警告訊息也都是顯示在此區域，最下方是狀態列，可以看到右下角的行列號和時間。

## B-3-2 EJE 的基本使用

EJE 在使用上非常直覺，這一節筆者準備簡單說明 EJE 的基本使用，這是一些 Java 程式開發時的常用功能。

### 程式碼編輯器

EJE 右邊是標籤頁的程式碼編輯器，一個比 Windows 記事本功能更強大的程式碼編輯視窗，如右圖所示：



選上方檔案名稱標籤，可以切換編輯的 Java 程式檔案，在標籤上，執行右鍵快顯功能表的**關閉檔案**指令，可以關閉標籤頁編輯的程式檔案（或執行「**檔案/關閉檔案**」指令）。在程式碼部分的關鍵字是藍色；註解為綠色；字串是紅色；其他程式碼是黑色。

在程式碼編輯視窗輸入文字內容時，鍵盤主要編輯按鍵的說明，如下表所示：

鍵盤按鍵	說明
Insert	切換插入字元或是取代字元
Caps Lock	切換英文字母的大小寫
Delete	刪除後面的一個字元
←Backspace	刪除前面的一個字元
Enter	新增一列程式碼

## 顯示程式碼前的行列號

EJE 標籤頁的程式碼編輯器預設在 Java 程式碼前顯示行列號，如果不想顯示行列號，請執行「檔案/選項設定」指令開啟「ELE 選項」對話方塊。

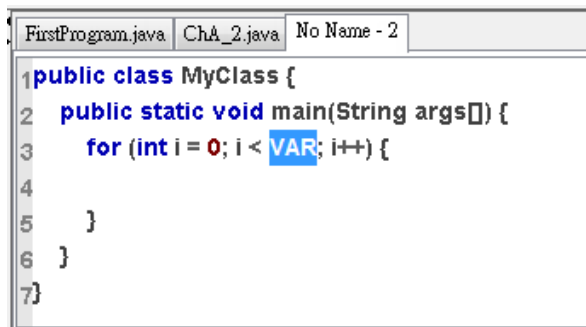


選 **EJE** 標籤，取消勾選顯示行號，按確定鈕隱藏行列號。

## 插入 Java 程式碼片段

EJE 在編輯 Java 程式碼時，可以執行**插入**功能表指令來插入常用 Java 程式片段，讓我們使用最少的鍵盤輸入來快速建立 Java 程式，對於 Java 語言的初學者來說，此功能可以大幅降低輸入程式碼拼字錯誤所造成的語法錯誤。

例如：執行「**插入/For 迴圈**」指令，可以插入 Java 語言的 for 迴圈，如下圖所示：

The image shows a code editor window with three tabs: "FirstProgram.java", "ChA\_2.java", and "No Name - 2". The "No Name - 2" tab is active. The code in the editor is as follows:

```
1 public class MyClass {  
2     public static void main(String args[]) {  
3         for (int i = 0; i < VAR; i++) {  
4               
5         }  
6     }  
7 }
```

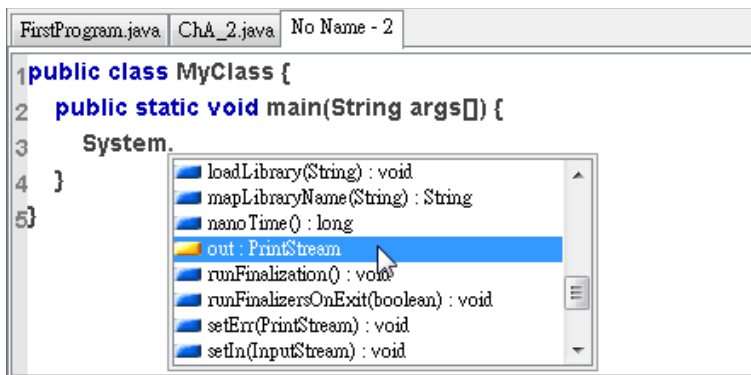
The word "VAR" in the for loop is highlighted in blue.



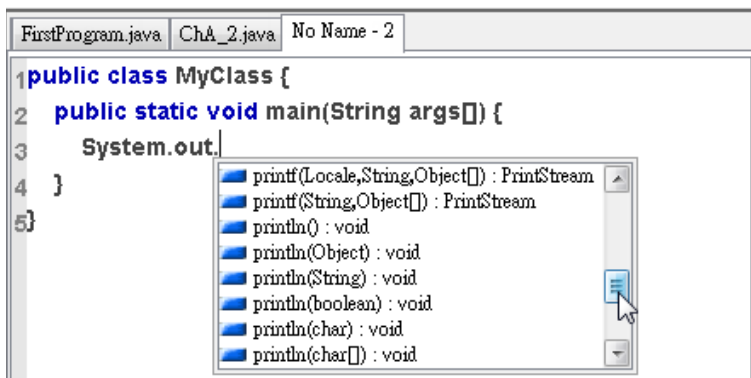
上述程式區塊是 for 迴圈。EJE 目前支援的 Java 程式片段包含：物件導向的類別宣告、介面宣告、列舉、類別建構子和 main()主程式等，流程控制的 If、Switch、For、While、Do While、For Each 和 Try/Catch 程式敘述，和各種括號與註解文字等。

## 自動 Java 程式語法提示

EJE 程式碼編輯器支援自動程式語法提示功能，只需輸入名稱後，輸入「.」符號，就可以彈出視窗顯示相關類別、物件、方法和屬性清單。例如：輸入 System，再輸入「.」就會顯示清單，如下圖所示：



找到和點選 **out** 輸入程式碼後，輸入「.」就會再次顯示清單來選擇方法或屬性，如下圖所示：



## MEMO

A large rectangular area for writing a memo, with a rounded bottom-right corner. The area is empty and intended for handwritten or typed notes.