

3. Hello World!您的第一隻Java程式

本章將先為您示範如何使用NetBeans來開發一個Java的Application。不論您是否已有Java程式設計的經驗，筆者都建議您仔細地閱讀本章，以瞭解在NetBeans中，是如何進行這些設計的工作。我們將以一個十分簡單的程式—「Hello World」做為示範，讓您瞭解NetBeans開發程式的細節。

3.1 使用JDK開發

在開始使用NetBeans進行Java程式開發前，我們先以一個簡單的範例說明如何以JDK完成一個Java程式的開發。我們所要開發的程式，是一個在Console模式輸出「Hello World!」字串的簡單下執行的應用。

- 首先，確認你的環境中Java是否有正確地被安裝。
 - 打開“命令提示字元”(也就是terminal)
 - 輸入java及javac
 - 檢視是否正確

Windows平台請確定有將jdk目錄下的bin目錄加入到Path環境變數的設定中。

- 開啟記事本或其它你慣用的文字檔編輯程式
- 輸入以下的程式碼

HelloWorld.java

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

- 完成後，請輸入以下指令

```
javac HelloWorld.java
```

- 如果沒有錯誤，則已完成編譯產生了HelloWorld.class的byte code。再輸入以下指令：

```
java HelloWorld
```

- 你應該會看到成功地輸出Hello World!

關於這個程式的說明，我們將於後續內容中說明。

課堂練習1

3.2 程式說明

前述的HelloWorld.java程式，定義了一個class - 名稱為HelloWorld[]以下為您說明相關的細節：

- Java語言規定，用以定義class的原始程式，其主要檔案名稱必須與class名稱一致。



注意：有些書籍將「class定義」稱為「class宣告」，本書將視情況交替使用這兩個名詞。



Java是case sensitive的語言，在為識別字(包含變數名稱、類別名稱等)命名時，大小寫視為不同字元。因此HelloWorld與HELLOWorld[]HelloWORLD[]helloworld[]helloWorld等會被視為不同的識別字。

- 第5行的package myfirstproject;[]是用以指定此程式所定義的class屬於myfirstproject這個package(套件)




Java是一個物件導向的程式語言，每個使用Java開發的應用程式通常會由多個class(類別)組成。為了方便管理大量的class[]Java使用了Package的概念 - 「一個應用程式可以擁有多個packages[]而每個package又可包含一個或一個以上的class[]回顧我們以NetBeans所開發完成的第一個專案MyFirstProject[]其中就包含有一個package[]名為myfirstproject[]


在學習Java語言的初期，您應該不會在一個專案中包含多個packages[]所以您可以暫時把package管理的問題交由NetBeans負責。您只要先專注於class的部份即可。


- HelloWorld.java的第11-20行定義了HelloWorld class
 - class的定義包含其存取子[]class名稱、傳入的參數以及class內容定義
 - 第11行的public為存取子，表示任何人都可以使用這個class
 - 第11行也定義了class名稱，即為前述的HelloWorld
 - 在class名稱後的是一組由大括號包裹起來的class內容(從第12行到第19行)
- 第1-4行、7-10行、13-15行以及第17行，為程式碼註解(comments)
 - Java語言的註解與C/C++相同，提供兩種註解型式
 - 第一種是單行註解，以「//」開頭

- 第二種是段落註解，以「/*」開頭，以「*/」結尾
- 此處看到的是第二種型式，但稍做修改為以「/**」開頭，以「*/」結尾，此種型式主要是配合JavaDoc使用


 JavaDoc是JDK用以製作程式說明文件的一種工具，我們只要在撰寫程式時，配合JavaDoc規範編寫註解，就可以自動產生程式的說明文件，詳見 [JavaDoc](#)

- 在第12-19行的class內容定義僅包含一個名為main的method
- main method即為程式的進入點
- Java語言規定程式進入點的型式必須為 `public static void main(String[] args)`

 在C/C++語言中的function在Java語言中稱為method(方法)。

 進入點(entry point)係指應用程式啟始後，首先被系統呼叫加以執行之處。一個使用Java開發的應用程式，可以包含多個class但只能有一個class具有main method也就是只允許一個進入點。

- 第18行是main method中唯一一行的java敘述
- 呼叫System.out.println() method將“Hello World!”字串輸出

 System.out.println()的作用就如同C語言中的printf()用以輸出字串或資料到標準輸出(也就是console)

最後，讓我們將一個最簡單的Java語言的程式架構列示於下：

```
class className
{
    public static void main(String[] args)
    {
        ... 略 ...
    }
}
```

我們在學習java語言的初期，大部份撰寫的程式都會符合以上的架構。

From:

<https://junwu.nptu.edu.tw/dokuwiki/> - **Jun Wu**的教學網頁

國立屏東大學資訊工程學系

CSIE, NPTU

Total: 295523



Permanent link:

<https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=java:helloworld>

Last update: **2025/09/04 12:57**