

# Installing Development kits

在開始開發程式之前，通常需要安裝程式開發環境，所謂的「開發環境」是一支程式，安裝後讓你可以執行你所撰寫的程式碼。然而，程式開發環境實際上非常多種，支援的功能差異也很大，因此有不少工程師均有他自己選擇或喜好的開發環境。為了各位學習方便，我們選擇整合兩套開發工具：**Canopy** 搭配 **ipython notebook**。前者整合了相當多套撰寫 Python 程式時可能會用到的第三方函式庫，且便利了第三方函式庫的安裝；後者則是便於初學者撰寫程式的互動介面。而日後會介紹一套 IDE 為 **LiClipse**，其提供了程式撰寫的便利功能，例如程式碼自動完成輸入、即時錯誤偵測、收合程式碼、以及物件函式建議等等，使用上十分便利。

## Install Python IDE: Canopy

### Downloading Canopy

1. 至以下鏈結下載並安裝 Python 的程式開發環境 Canopy，可以使用教育版下載則為免付費。<https://www.enthought.com/products/canopy/>
2. 點選「Get Canopy」進入 store。
3. 選擇「For Academics」

The screenshot shows a navigation bar with four tabs: 'For Individuals', 'For Groups', 'For Enterprises', and 'For Academics'. The 'For Academics' tab is highlighted with a thicker border.

4. 選擇「Request your license」。
- 
- The screenshot shows a dark blue card with the text 'CANOPY ACADEMIC WITH TRAINING ON DEMAND' at the top, followed by 'FREE' and 'For students and staff at degree-granting institutions'. At the bottom is a blue button labeled 'Request your license'.
5. 註冊：請用 academic e-mail，例如是 edu.tw 結尾的 E-mail。
  6. 切換至你的 E-mail 以啟動帳號
  7. 回到「Request your academic license」頁登入。

You have an academic license already!

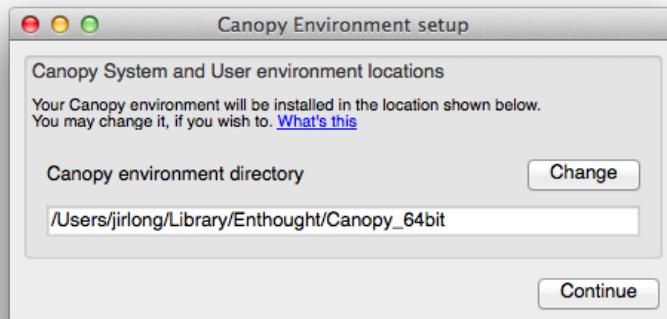
jirlong@ntnu.edu.tw has an active Academic License.

You can now download Enthought Canopy. Logging in as jirlong@ntnu.edu.tw within Canopy will give you access to all features available under Academic License.

8. 選擇「You can now download Enthought Canopy」
9. 選擇「Canopy Express Installer」或「Canopy Full Installer」，可以下載 Express 版本就好（請注意你所選擇的作業系統版本是 32 bits 或 64 bits）。

## Installing Canopy

1. Choose environment directory

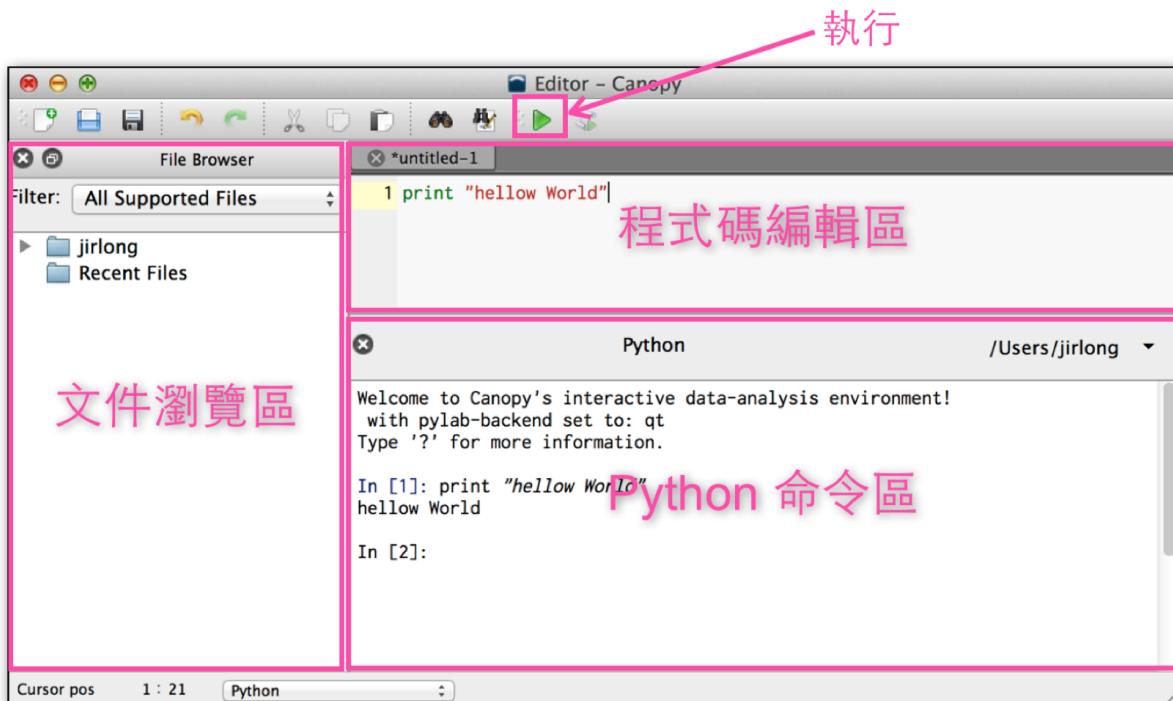


2. Set as default python editor



## Introduction to Canopy User Interface

Canopy 開發環境安裝完成後的界面情形，界面主要分為三大區塊：左方為程式與文件區，右上方用以編輯程式碼，右下方為 **python** 的命令列。



# Using ipython notebook

## 從命令列開啟 notebook

- Mac
  - 至你的/Users/[your user name]/帳號目錄底下新增 pss2015 資料夾



- 請在 Finder 搜尋「Terminal」。
- 輸入「cd /Users/[your user name]/pss2015/」用你的使用者名稱取代[your user name]
- 在命令列輸入 ipython notebook
- Windows
  - 至你的 C 槽底下新增 PSS2015 資料夾
  - 請在開始工具列或可以搜尋的地方找「cmd」或者「命令列執行視窗」。
  - 輸入「cd c:\pss2015\」
  - 在命令列輸入 ipython notebook

## 使用 ipython notebook

- 當你成功啟用時，你的命令列或者 Terminal 會看到以下訊息

```
[17]/Users/jirlong/pss2015$ cd ..
[18]/Users/jirlong$ cd /Users/jirlong/pss2015
[19]/Users/jirlong/pss2015$ ipython notebook
2015-02-14 10:43:26.213 [NotebookApp] Using existing profile dir: u'/Users/jirlong/.ipython/profile_default'
2015-02-14 10:43:26.217 [NotebookApp] Using MathJax from CDN: https://cdn.mathjax.org/mathjax/latest/MathJax.js
2015-02-14 10:43:26.245 [NotebookApp] Serving notebooks from local directory: /Users/jirlong/pss2015
2015-02-14 10:43:26.245 [NotebookApp] 0 active kernels
2015-02-14 10:43:26.245 [NotebookApp] The IPython Notebook is running at: http://localhost:8888/
2015-02-14 10:43:26.245 [NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
```

- 並且你的瀏覽器會被打開，同時出現這個頁面，恭喜你成功了。

- 從右上方「New Notebook」新增一個筆記本，建議命名為 helloWorld，或者 pss0227，注意命名，開頭不要大寫，中間沒有空白，第二個字開頭用大寫。
- 隨意輸入幾個指令並執行。

# “Hello World”

- 請在程式碼編輯區輸入以下指令，並執行程式，觀察輸出後的結果。

```

1  print "hellow world"
2  print 'hellow world'
3  name = 'jirlong'
4  print "my name is " + name
5  print I+2+4+5+6+7+8+9 ]
6  a = 5
7  print a+I+2+3+4+5+6+7+8+9
8  print a**4
9  print "a*a*a*a=" , a**4
IO  print ("a*a*a*a=" , a**4)
II  name = "jirlong "
12  print name*4

```

(Code can be found at <http://rextester.com/live/BWF89434>)

- 執行後的結果如下：

```

hellow world
hellow world
my name is jirlong
42
50
625
a*a*a*a= 625
('a*a*a*a=' , 625)
jirlong jirlong jirlong jirlong

```

# “Hellow World” 2

- 再輸入以下程式碼

```

1  a=3
2  b=2
3  print a/b
4  print a*b
5  print a**b
6  print a%b
7  print a/float(b)
8  print float(a)/b
9  print ((a+b)**2+(a-b)**2)/float(a**2+b**2)
10 print str(a)/b

```

(Code can be found at <http://rextester.com/TSH79118>)

- 得到以下的結果

```
Error(s), warning(s):
Traceback (most recent call last):
  File "source_file.py", line 10, in <module>
    print str(a)/b
TypeError: unsupported operand type(s) for /: 'str' and 'int'
1
6
9
1
1.5
1.5
2.0
```

# LiClipse as the Code Editor

- 由於 Canopy 的編輯器界面前目前缺乏甚多撰寫程式所需要的功能，所以建議安裝 LiClipse 來作為編輯器。但是 Canopy 和 Matplotlib 的結合真的較佳，安裝套件（Package）的方式也較為簡便，當需要安裝套件反而可以回到 Canopy 來執行。

## Downloading LiClipse

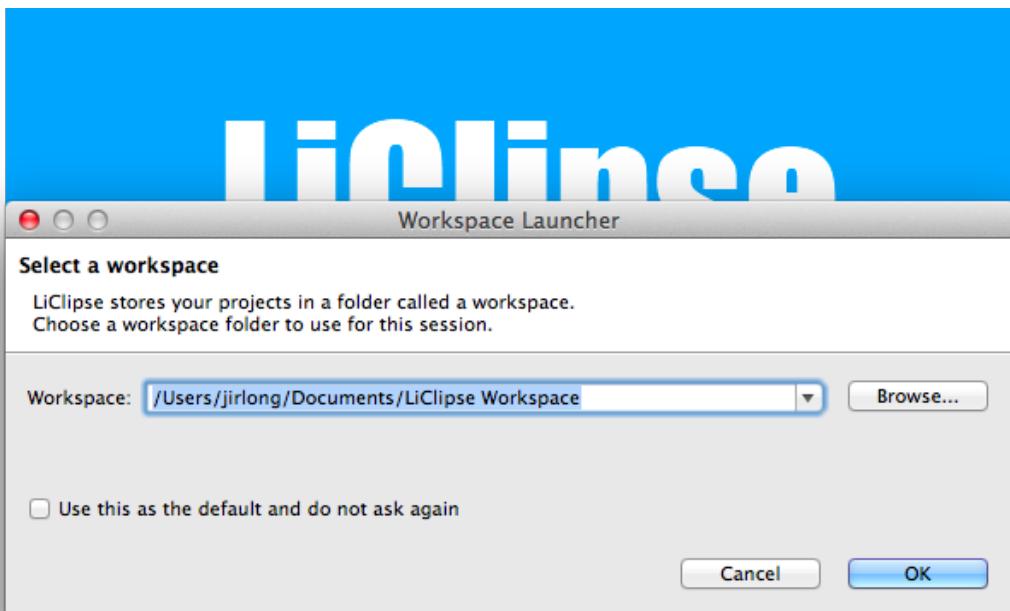
- 下載 LiClipse 並安裝 (<http://www.liclipse.com/download.html>)。
- 選擇相對應的作業系統版本（注意 WINx86 或 WIN64）。如果 canopy 是安裝在 32bits 的 Windows 上就要選 win32 x86 的版本，若是 64bits 就要選 win32\_64 版。

## Installing LiClipse

- 若電腦中沒有 JAVA SE 6 runtime 的話則會被提醒安裝。

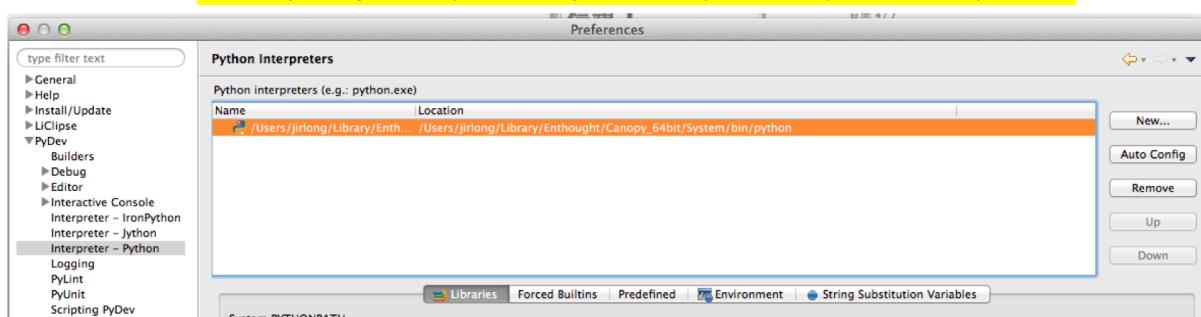


- 選擇 Workspace，程式碼擺放的位置。

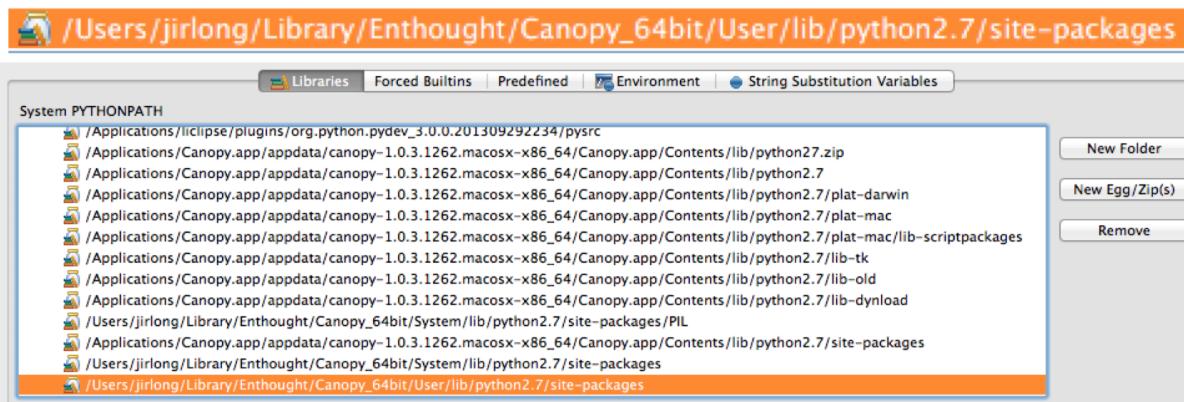


### 3. 設定開發路徑（將 PyDev 指到 Canopy）

- 安裝完畢後開啓 LiClipse 的偏好設定 (window/preferences)
- 選擇 PyDev 的 Interpreter/python interpreter 以設定 Canopy 的 Python 所在位置
- Windows 所在位置為 C:\Users\NTNU-GLIS\AppData\Local\Enthought\Canopy\User\python，  
如果以 Mac 來說的話就是  
`/Users/jirlong/Library/Enthought/Canopy_64bit/System/bin/python`。



- ### 4. 除此之外還可能需要增加 Library 所在位置，以 Mac 來說是在下面的位置(□一定會放在某個 python2.7 底下的 site-packages 資料夾)，記得按 Apply，他會執□行匯□入的動作□一陣□子，等跑完後再按 OK。



5. 之後重新開啓 LiClipse 程式，就可以開始撰寫。□後若需要新增新的 Package，就回到 Canopy 去新增，撰寫程式則在 LiClipse 寫較為□方便。
6. 新增專案 File>New>PyDev Project
  - 如果沒有的話，代表沒有切換到 PyDev Perspective，請找到 Window > Open Perspectives，選擇 PyDev 並套用即可。

## Problem shooting

- Mac 的使用者如果發現再也無法 import 一些來自 canopy 的 library 時，請讓 python 自動偵測並匯入 python 後，利用下圖 New folder 的功能，加入最後兩行函式庫（也是上圖的最後兩個函式庫）並點選 apply 就可以接下去使用。

