

The background of the slide is a light gray gradient. It is decorated with numerous realistic water droplets of various sizes. Some droplets are at the top left, some are in the middle right, and others are at the bottom. They have highlights and shadows, giving them a 3D appearance.

# **2022 LINEAR ALGEBRA**

## **PROJECT01**

2022/09/19

# OUTLINE

## 一、作業環境介紹與安裝

- IDE介紹
- Anaconda、Jupyter Notebook、numpy介紹
- Anaconda、Jupyter Notebook、numpy安裝及使用教學

## 二、Python基本語法說明

## 三、作業說明

# 作業環境介紹與安裝

## ● IDE ( Integrated Development Environment ) 整合開發環境

：一種輔助程式開發人員開發軟體的應用軟體，在開發工具內部就可以輔助編寫原始碼文字、并編譯打包成爲可用的程式，有些甚至可以設計圖形介面。

- Visual Studio Code (VS Code)
- PyCharm
- Spyder
- Jupyter Notebook

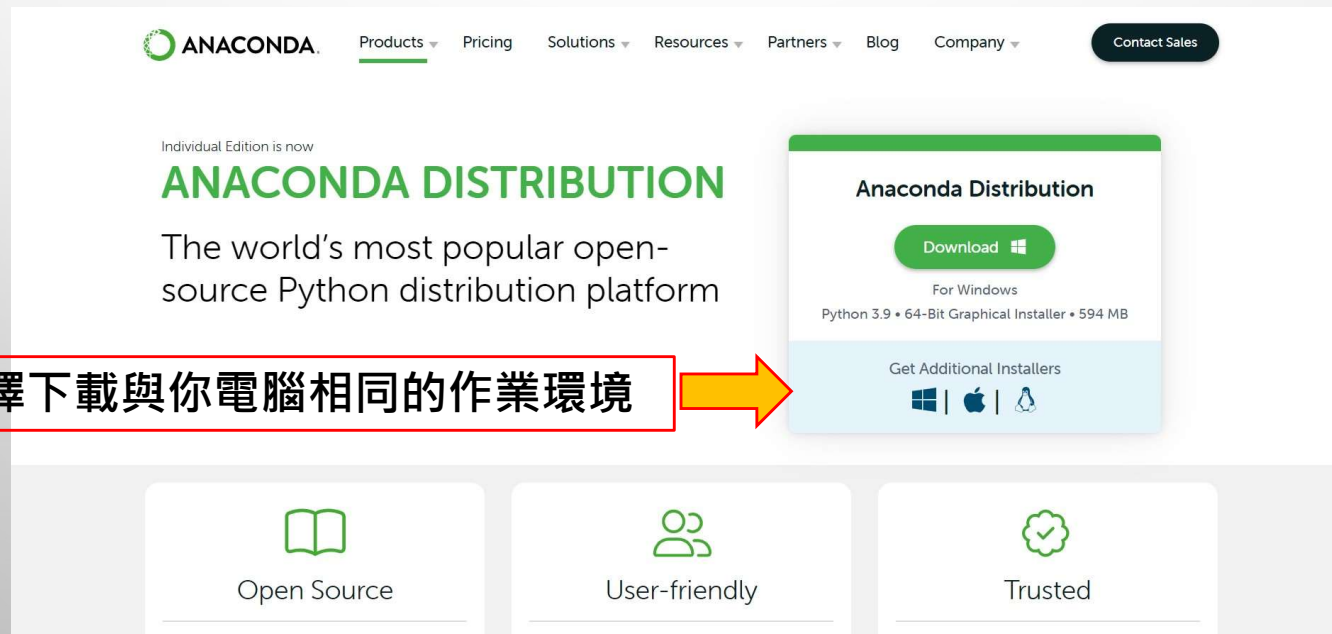
- **Anaconda**：是Python及R語言的一個免費開源發行版本，主要用於數據科學（Data Science），機器學習（Machine Learning），巨量數據處理（Large-Scale Data Processing）以及預測分析（Predictive Analytics），可Python資料科學（Data Science）平臺。
- **Jupyter Notebook**：是IPython + Notebook集成架構，介於編輯器和IDE之間的應用環境，除了Julia、的Python及R之外，也提供許多的kernels(例如Java、C#、Go、Ruby、JavaScript等超過50多種Jupyter kernels)。
- **Numpy套件**：使用者可以很容易地用此套件建立向量（Vector）、矩陣（Matrix）等進行高效率的大量資料運算，是許多 Python 資料科學套件的基礎。

# ANACONDA下載方法

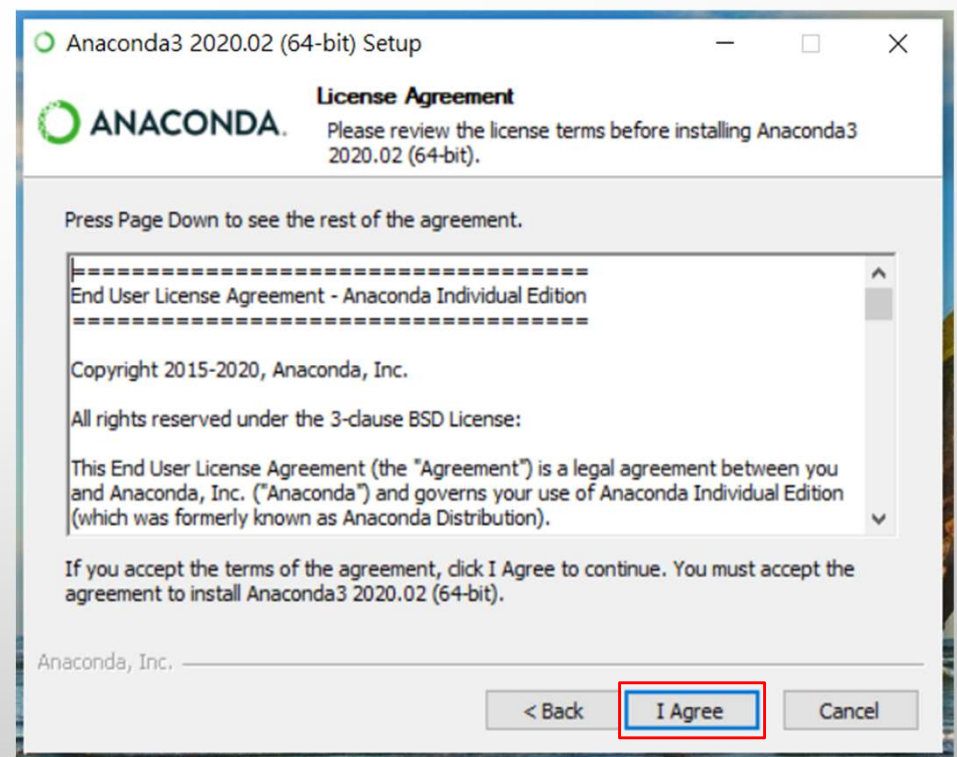
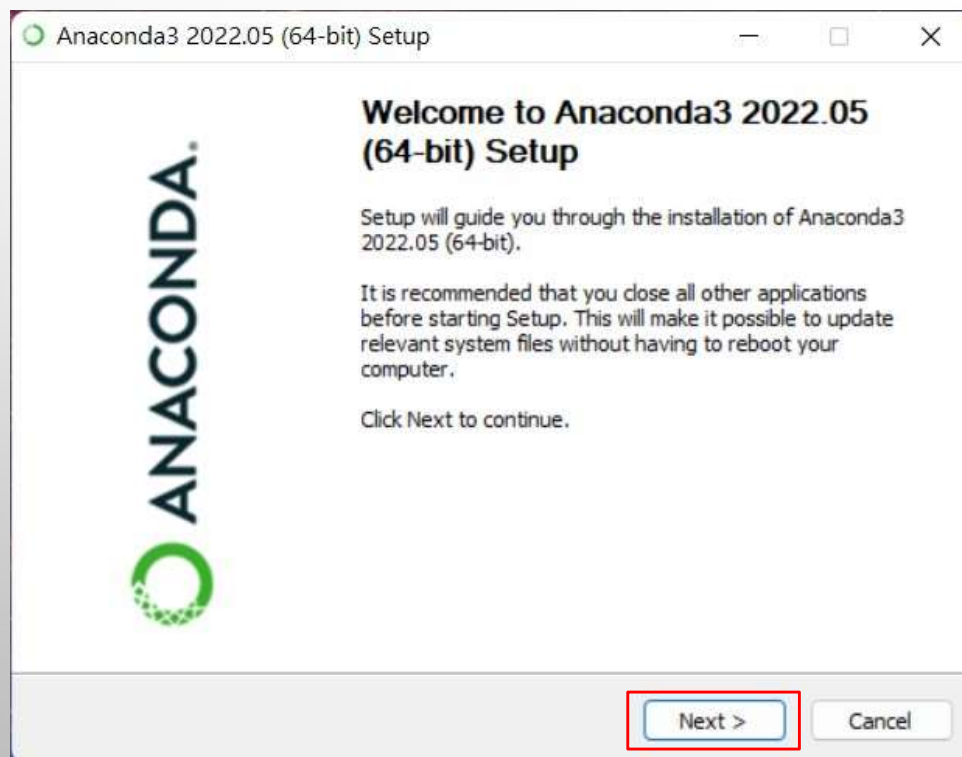
●進入ANACONDA官網:

<https://www.anaconda.com/products/distribution#windows>

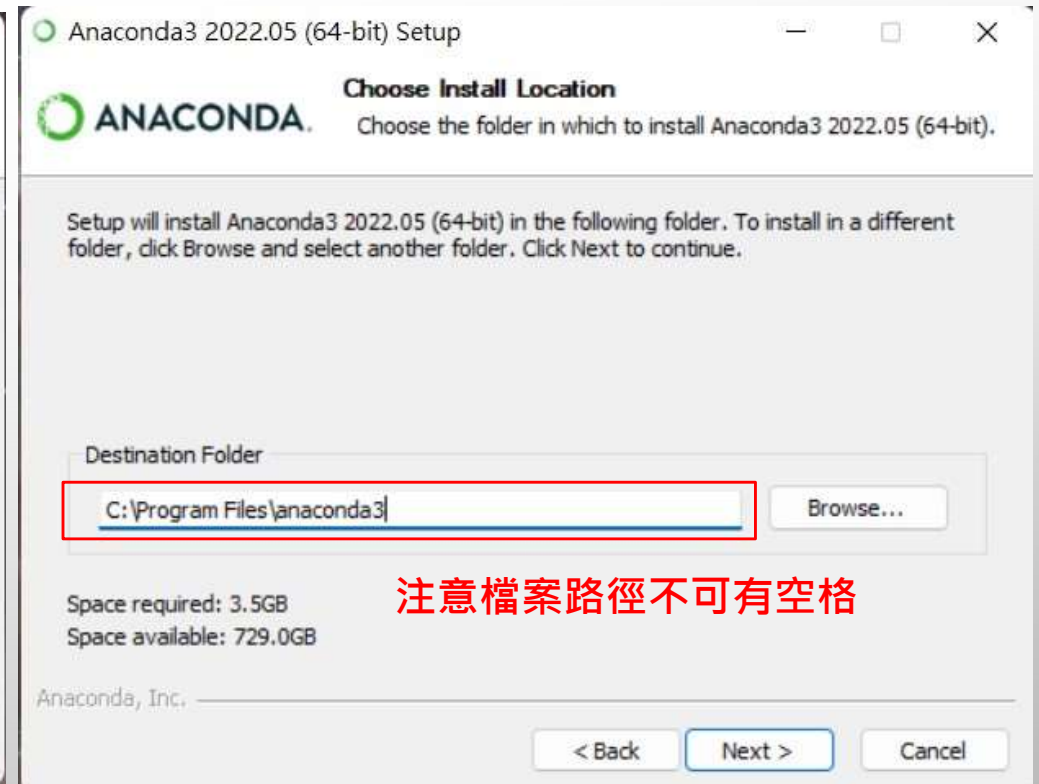
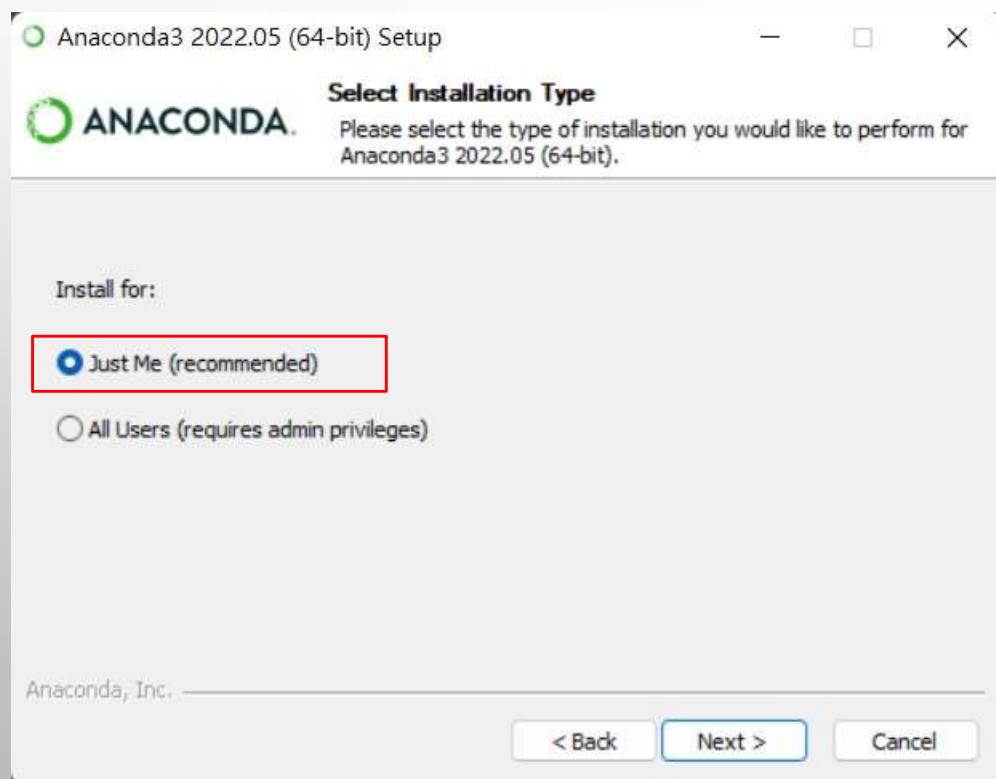
選擇下載與你電腦相同的作業環境



# ANACONDA下載方法

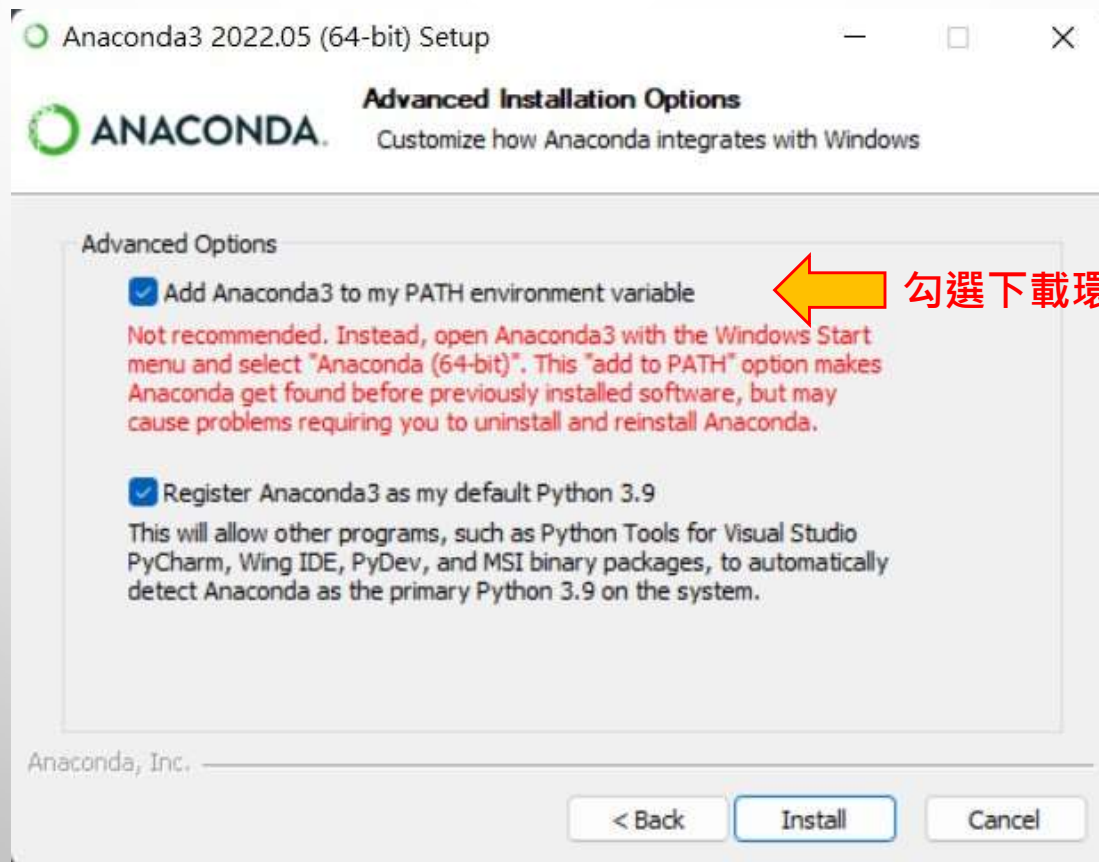


# ANACONDA下載方法



注意檔案路徑不可有空格

# ANACONDA下載方法



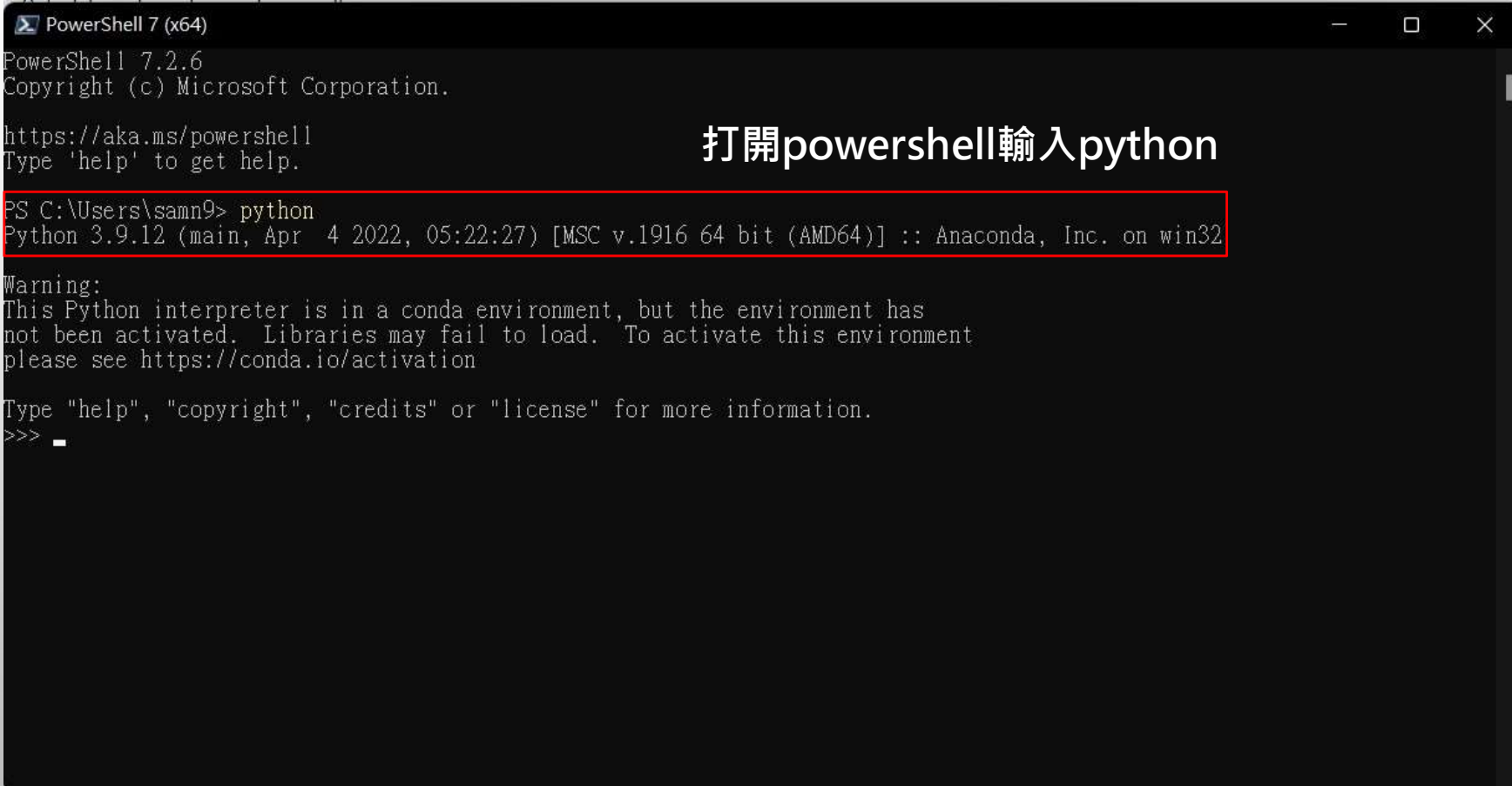
勾選下載環境變數的選項



# ANACONDA下載方法



# 確認ANACONDA下載成功



```
PowerShell 7 (x64)
PowerShell 7.2.6
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

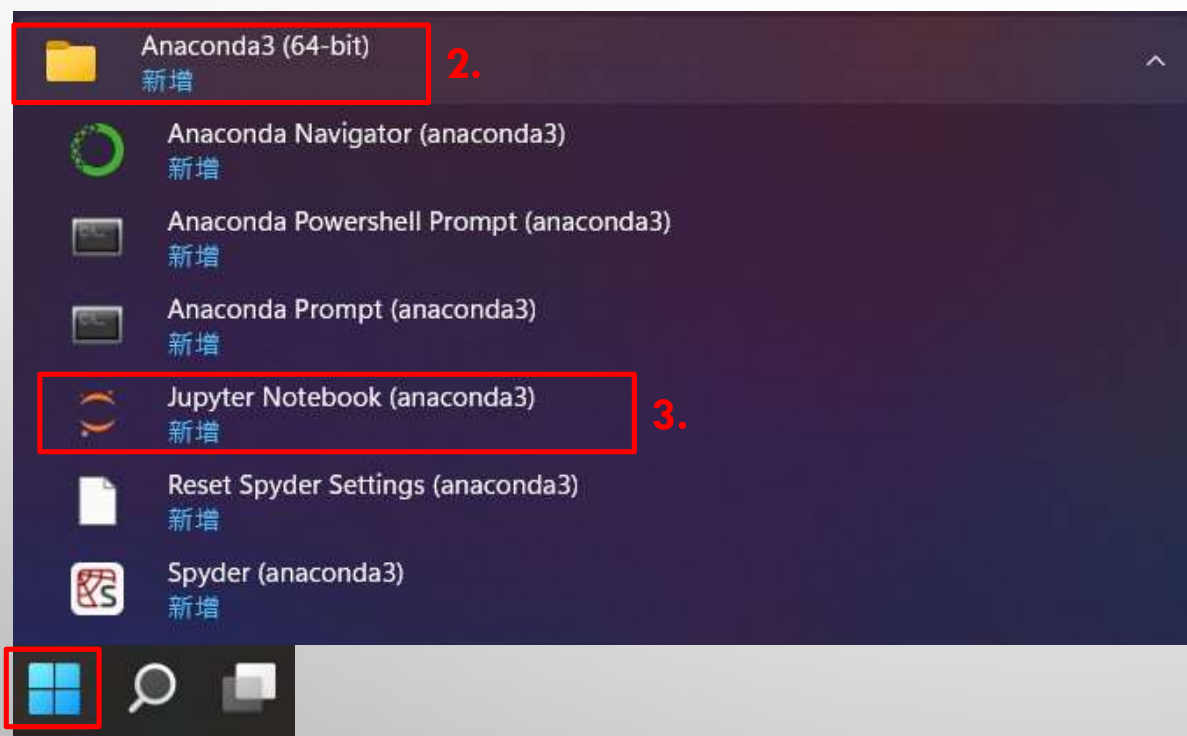
PS C:\Users\samn9> python
Python 3.9.12 (main, Apr  4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32


Warning:
This Python interpreter is in a conda environment, but the environment has
not been activated. Libraries may fail to load. To activate this environment
please see https://conda.io/activation

Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

打開powershell輸入python

# JUPYTER NOTEBOOK使用教學



1. 點擊  的圖示
2. 找到Anaconda3的資料夾
3. 點擊資料夾中的Jupyter Notebook

# JUPYTER NOTEBOOK使用教學

```
Jupyter Notebook (anaconda3)
[I 22:39:42.294 NotebookApp] Writing notebook server cookie secret to C:\Users\samn9\AppData\Roaming\jupyter\runtime\notebook_cookie_secret
[W 2022-09-14 22:39:42.841 LabApp] 'notebook_dir' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to update your config before our next release.
[W 2022-09-14 22:39:42.841 LabApp] 'notebook_dir' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to update your config before our next release.
[I 2022-09-14 22:39:42.841 LabApp] JupyterLab extension loaded from C:\Users\samn9\anaconda3\lib\site-packages\jupyterlab
[I 2022-09-14 22:39:42.841 LabApp] JupyterLab application directory is C:\Users\samn9\anaconda3\share\jupyter\lab
[I 22:39:42.841 NotebookApp] Serving notebooks from local directory: C:\Users\samn9
[I 22:39:42.841 NotebookApp] Jupyter Notebook 6.4.8 is running at:
[I 22:39:42.841 NotebookApp] http://localhost:8888/?token=d6bab0a197cb1b3701dd762a3f92d1e71073c0188b8bd831
[I 22:39:42.841 NotebookApp] or http://127.0.0.1:8888/?token=d6bab0a197cb1b3701dd762a3f92d1e71073c0188b8bd831
[I 22:39:42.841 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 22:39:42.873 NotebookApp]
```



出現此視窗後，將會自動開啓jupyter

使用jupyter時請勿關閉該視窗，否則無法運作

To access the notebook, open this file in a browser:  
file:///C:/Users/samn9/AppData/Roaming/jupyter/runtime/notebook-  
Or copy and paste one of these URLs:  
http://localhost:8888/?token=d6bab0a197cb1b3701dd762a3f92d1e71073c0188b8bd831  
or http://127.0.0.1:8888/?token=d6bab0a197cb1b3701dd762a3f92d1e71073c0188b8bd831

jupyter

Quit Logout

Files Running Clusters

Select items to perform actions on them.

Upload New ↻

	Name	Last Modified	File size
<input type="checkbox"/>	0		
<input type="checkbox"/>	anaconda3	23 分鐘前	
<input type="checkbox"/>	ansel	1 個月前	

# JUPYTER NOTEBOOK使用教學

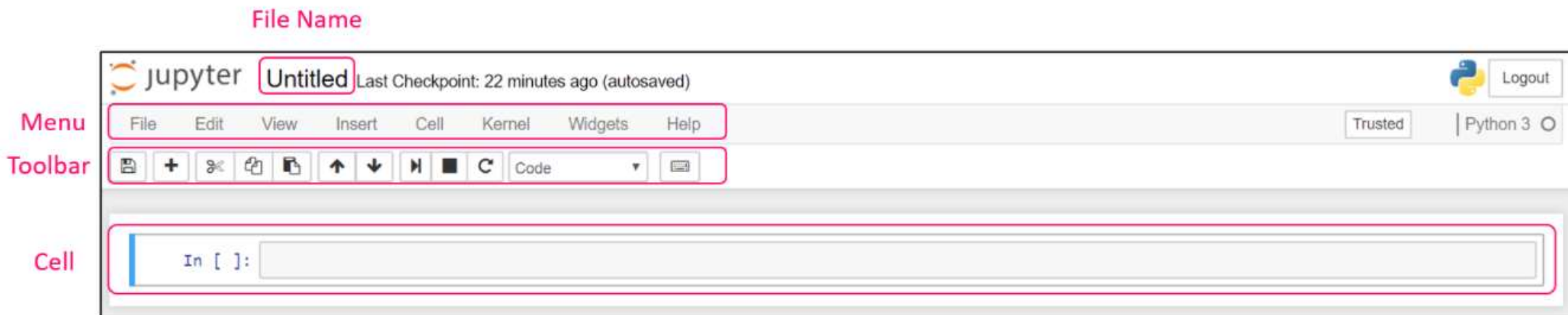
1.點選右上角的new

2.之後選python 3

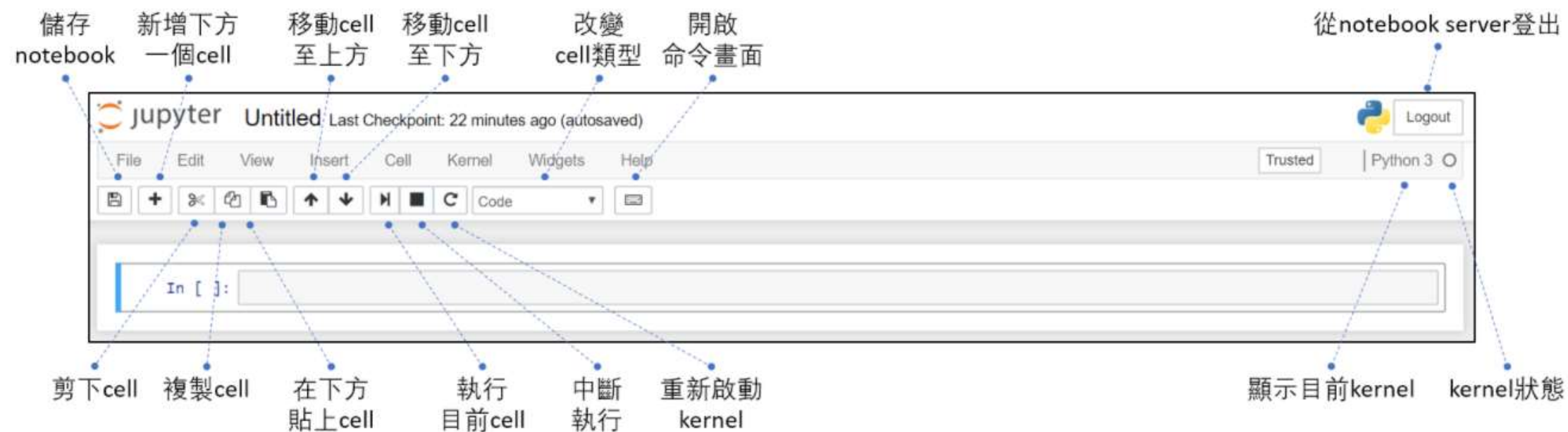


# JUPYTER NOTEBOOK使用教學

3.步驟2.執行完後, 會來到此畫面

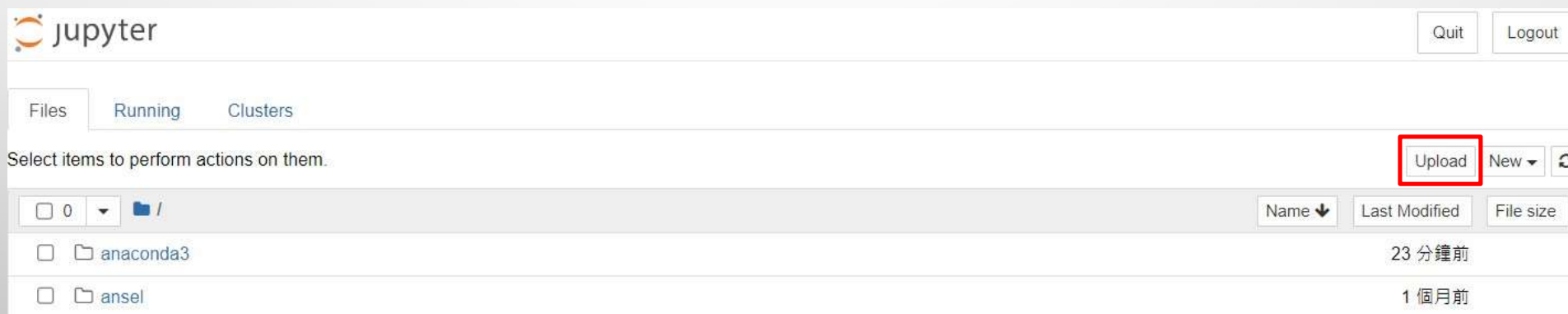


# JUPYTER NOTEBOOK使用教學



# JUPYTER NOTEBOOK使用教學

## 5.檔案匯入



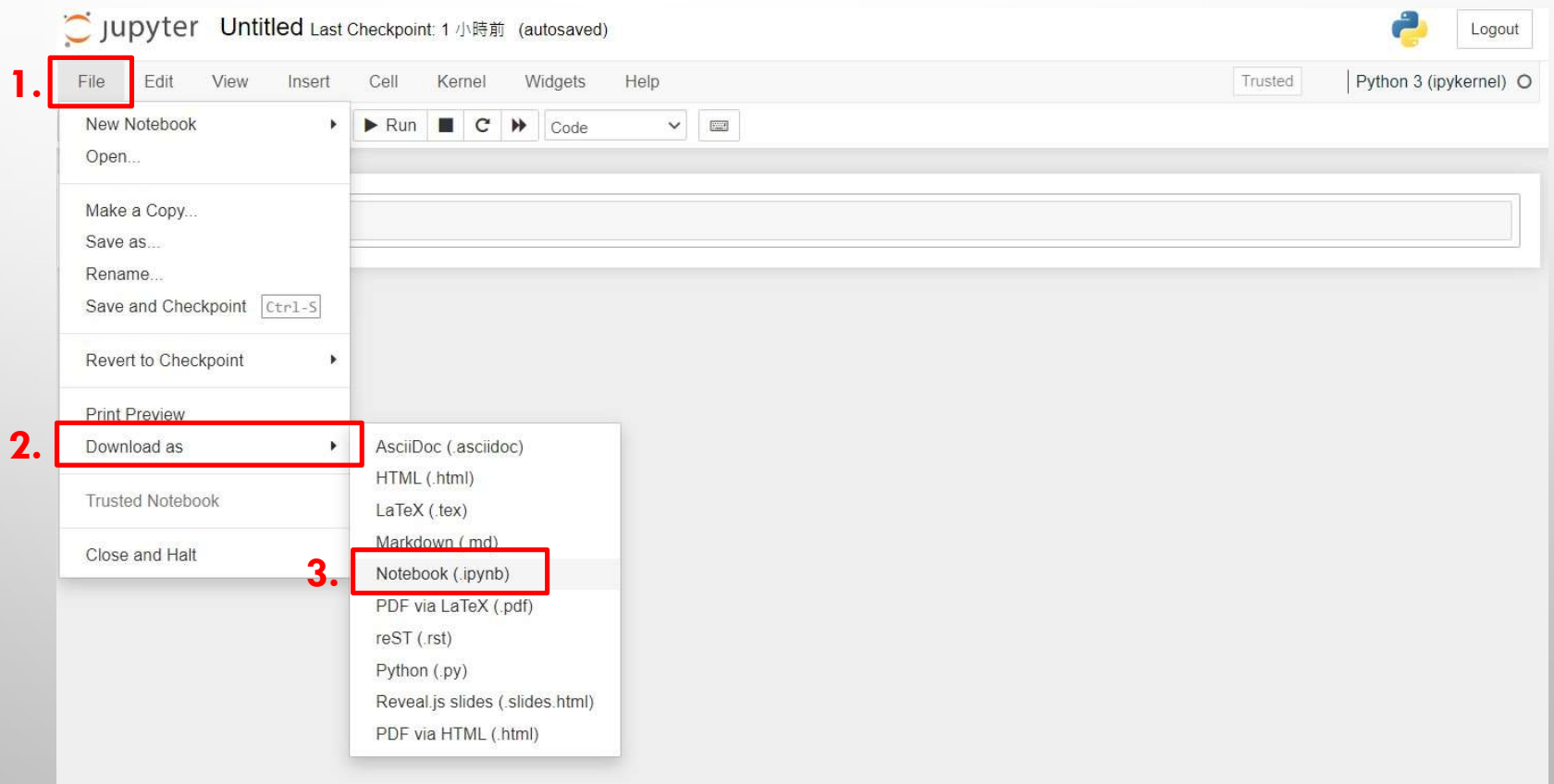
The image shows the Jupyter Notebook interface with the 'Files' tab selected. At the top right, there are 'Quit' and 'Logout' buttons. Below the tabs, there is a message 'Select items to perform actions on them.' and an 'Upload' button, which is highlighted with a red rectangle. To the right of the 'Upload' button are 'New' and a refresh icon. Below this, there is a table with columns 'Name', 'Last Modified', and 'File size'. The table lists two folders: 'anaconda3' and 'ansel'.

	Name	Last Modified	File size
<input type="checkbox"/>	anaconda3	23 分鐘前	
<input type="checkbox"/>	ansel	1 個月前	

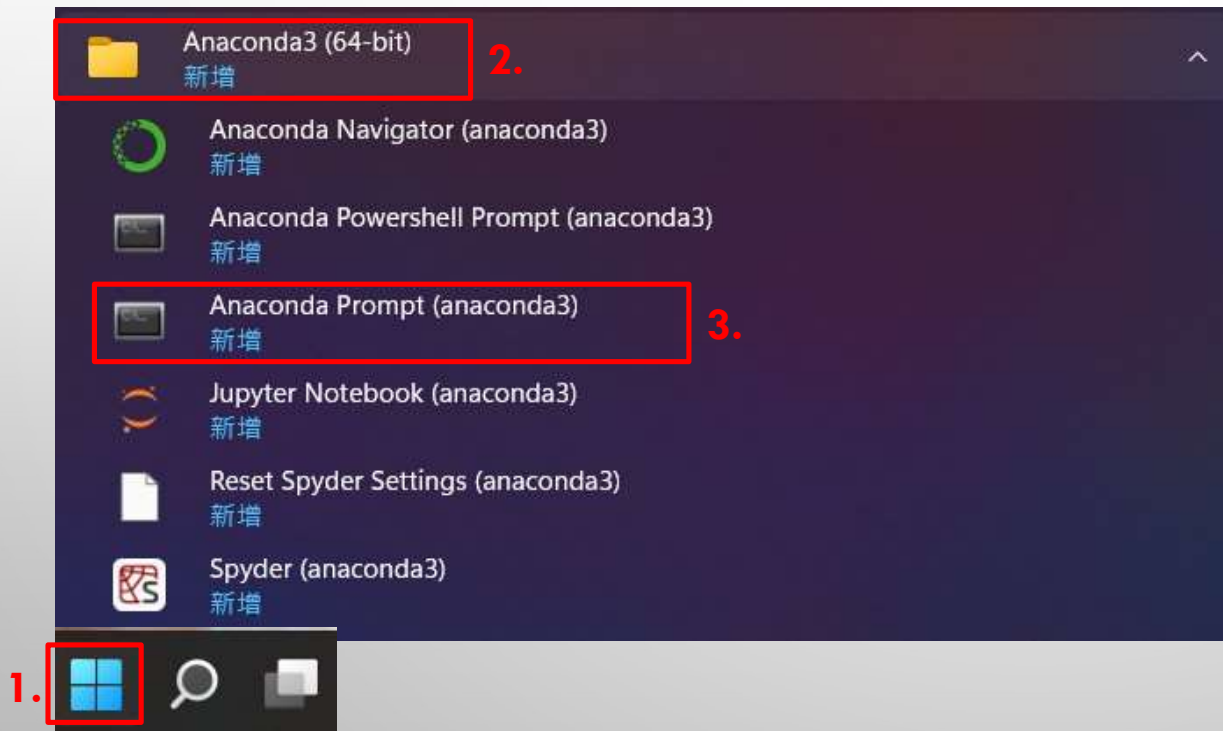



# JUPYTER NOTEBOOK使用教學

## 5.檔案匯出



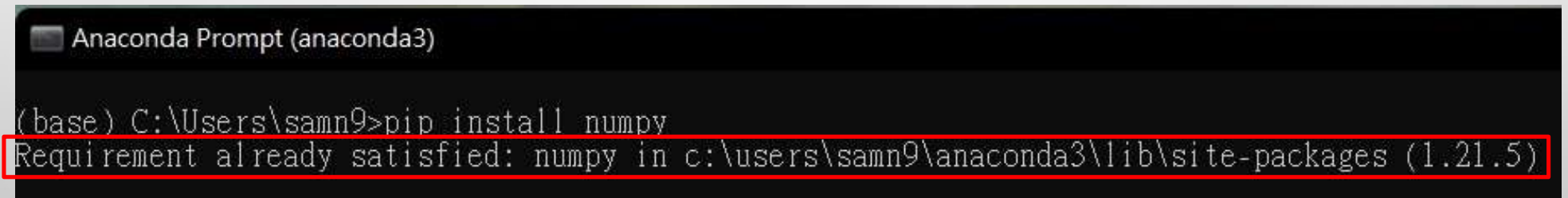
# NUMPY安裝教學



1. 點擊  的圖示
2. 找到ANACONDA3的資料夾
3. 點擊資料夾中的  
ANACONDA PROMPT

# NUMPY安裝教學

- 輸入指令 `pip install package` (package 是要安裝的套件名稱)
- 確認出現成功安裝的訊息



```
Anaconda Prompt (anaconda3)

(base) C:\Users\samn9>pip install numpy
Requirement already satisfied: numpy in c:\users\samn9\anaconda3\lib\site-packages (1.21.5)
```

# OUTLINE

## 一、作業環境介紹與安裝

- IDE介紹
- Anaconda、Jupyter Notebook、numpy介紹
- Anaconda、Jupyter Notebook、numpy安裝及使用教學

## 二、Python基本語法說明

## 三、作業說明

# PYTHON基本語法說明

- `import ... as ...`
- `import` 代表的是引入一個package(套件)
- 而`as` 代表的是取一個簡易的名稱避免原名稱過于冗長。

- EX：以引入作業要用的NUMPY為例子：

```
import numpy as np
```

# PYTHON基本語法說明

- **型態**
- **String(字串)**：把字元串起來，(字母、數字、符號、空格或是換行都是字元)。
- **int(整數)**，用來表示整數值。
- **float(浮點數)**，用來表示小數值。
- **bool(布林值)**，代表的是**1(true)**與**0(false)**，用來判斷語句的運算值為真或假。

# PYTHON基本語法說明

- 基礎語法
- PYTHON 在輸入時input(輸入)皆默認為string(字串) 的型態。
- split() 用來把字串透過指定的分隔符號對字串進行分割，  
不給入參數的話，預設會以 空白作為分隔符號做切割。
- 型態()，代表的是強制轉換型態。
- type() 為用來查看資料型態的方法，class可視作一個類別的意思。
- input() 為輸入，print() 為輸出。

# PYTHON基本語法說明

- **運算子**
- `'/'` 為一般的除法概念，而 `'//'` 為整數除法也就是忽略小數點。
- `'%'` 為求餘數。
- `'**'` 為次方意義，即a的b次方。

```
a=1
b=2
print(type(a),type(b))
print(a+b,type(a+b))
print(a-b,type(a-b))
print(a*b,type(a*b))
print(a/b,type(a/b))
print(a//b,type(a//b))
print(a%b,type(a%b))
print(a**b,type(a**b))
print(float(a),type(float(a)))
```

```
<class 'int'> <class 'int'>
3 <class 'int'>
-1 <class 'int'>
2 <class 'int'>
0.5 <class 'float'>
0 <class 'int'>
1 <class 'int'>
1 <class 'int'>
1.0 <class 'float'>
```



# PYTHON基本語法說明

- 基礎語法
- if ... else
- if '條件句' :  
    '執行句'  
    elif '條件句' :  
    '執行句'  
    else '條件句' :  
    '執行句'

```
a=0  
if a>0:  
    print('a>0')  
elif a<0:  
    print('a<0')  
else:  
    print('a=0')
```

a=0

(Python會自動偵測型態，所以這裡的a,b 便會被判斷為int的型態。)

# PYTHON基本語法說明

- **NUMPY 套件**
- `np.array()`：建成一個矩陣，有矩陣加減乘除的應用。
- 後項的FLOAT是將每一個ELEMENT強制轉換的意思。
- `dtype`:返回陣列中元素的資料型態。
- `np.astype()`：轉換陣列/矩陣的型態(EX: INT-FLOAT)
- 這邊的float64代表有8個bytes， $8*8=64$ 個bits。
- **基礎語法**
- 可直接寫成A，或`print(A)`。

```
A=np.array(([1,2,3],[4,5,6]),float)
#A
print(A,type(A),A.dtype)

[[1.  2.  3.]
 [4.  5.  6.]] <class 'numpy.ndarray'> float64
```

# PYTHON基本語法說明

- **NUMPY 套件**
- `np.zeros((m,n))`:建成一個M個ROW、N個COLUMN，ELEMENT全為0的矩陣。
- `shape`:查看矩陣的ROW跟COLUMN。
- **基礎語法**
- `map(function, iterable, ...)`:定義一個FUNCTION。  
接著用這個function來對一個iterable 的物件內每一個元素做處理。

# PYTHON基本語法說明

- 在這邊，使用map函式第一個參數int，將第二個參數轉型成整數(int)。

```
m,n=map(int,input().split())  
matrix=np.zeros((m,n))  
print(matrix,matrix.shape)
```

```
3 5  
[[0. 0. 0. 0. 0.]  
 [0. 0. 0. 0. 0.]  
 [0. 0. 0. 0. 0.]] (3, 5)
```

# PYTHON基本語法說明

- **for迴圈**
- i是從0開始到num-1，而不是到num。
- len:返回的值為長度或項的個數
- += ex: (num+=1) == (num=num+1)

```
num=10  
for i in range(num):  
    print(i)
```

0  
1  
2  
3  
4  
5  
6  
7  
8  
9

```
B=np.array([1,2,3])  
for i in range(len(B)):  
    print(i)  
    B[i]=2*B[i]
```

B

0  
1  
2

array([2, 4, 6])

# PYTHON基本語法說明

- 矩陣乘法
- “使用 ‘\*’ ，表示矩陣內的每個ELEMENT相乘。（ ‘/’ 用法相同 ）

```
A=np.array([1,2,3])
B=np.array([4,5,6])
print(A*B,type(A*B),(A*B).dtype)

[ 4 10 18] <class 'numpy.ndarray'> int32
```

```
C=A*B
print(C.astype(float),(C.astype(float)).dtype)

[ 4. 10. 18.] float64
```

# PYTHON基本語法說明

- **NUMPY 套件**
- **A.dot(B)**:代表矩陣的內積，亦可寫成**np.dot(A,B)**。

```
A.dot(B)  
print(A.dot(B),type(A.dot(B)),(A.dot(B)).dtype)  
32 <class 'numpy.int32'> int32
```

# PYTHON基本語法說明

- **NUMPY 套件**
- `np.diag()`:將原本1\*3的A矩陣，形成3\*3以一維矩陣A的元素作為對角線上的矩陣。

```
np.diag(A)
```

```
array([[1, 0, 0],  
       [0, 2, 0],  
       [0, 0, 3]])
```



# PYTHON基本語法說明

- **NUMPY 套件**
- **reshape**:對有n個element的矩陣，重塑陣列形狀，需化為擁有相同數量(即row\*column=n)的element的矩陣。
- **np.diag()**:若對一個不為一維矩陣的使用np.diag()，會變成提取對角線上的元素。

```
D=np.array([1,2,3,4,5,6,7,8,9])  
print(D, '\n', D.reshape(3,3), '\n', np.diag(D.reshape(3,3)))
```

```
[1 2 3 4 5 6 7 8 9]  
[[1 2 3]  
 [4 5 6]  
 [7 8 9]]  
[1 5 9]
```

# OUTLINE

## 一、作業環境介紹與安裝

- IDE介紹
- ANACONDA、JUPYTER NOTEBOOK、NUMPY介紹
- ANACONDA、JUPYTER NOTEBOOK、NUMPY安裝及使用教學

## 二、PYTHON基本語法說明

## 三、題目要求

## 作業説明

本次project提供給同學的檔案（共三份）：

- 1.教學PPT(即本檔案)
- 2.與教學PPT上相同的範例code [LA\_exp\_proj01.ipynb]
- 3.需要學生實作的作業檔案 [LA\_project01\_學生\_v1.ipynb]

**1跟2請學生自主學習參考，也是本次作業的考察範圍！**

!!!!!!!!!!!!!!! 本次project需要學生提交的檔案（共兩份）：!!!!!!!!!!!!!!!

- 1.程式作業：實作[LA\_project01\_學生\_v1.ipynb],並將檔案名稱改爲[LA\_proj01\_學號\_version.ipynb] .(version 如果是第一次繳交就是v1,第二次就是v2,以此類推！)

## 2.Report

檔案格式：PDF

檔案名稱：[LA\_Report01\_學號\_version]

將兩份檔案一同壓縮後命名[LA\_proj01\_學號\_version]于作業區繳交(檔案名稱不對本次作業成績-10！)

繳交期限：2022/9/26 0:00. ( 7天 )

# 作業說明-程式作業(0-1)

- 題目0：請同學按照作業環境安裝的步驟分別下載安裝anaconda與numpy套件，並截圖證明已經下載成功的畫面（ anaconda一張， numpy一張， 如何證明前面PPT有寫）並將圖片插入到report中。

# 作業說明-程式作業(1-1)

題目1: 構建3x5的full zeros matrix : zero\_A. 並按範例output一樣的內容 (20pt)

Output:

```
int zero_A:
[[0 0 0 0 0]
 [0 0 0 0 0]
 [0 0 0 0 0]]
type of zero_A : <class 'numpy.ndarray'>
dtype of zero_A : int32
float zero_A:
[[0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0.]]
```

考察內容：

1.矩陣的建立2.檢查變數型態等基本資訊3.型態的轉換

Function : np.zeros()      type()   np.dtype()   np.astype()      print()

評分規則：

一共4個output，每個5 point，成功輸出即得分。

## 作業說明-程式作業(2-1)

題目2: 請用以下2種方法得到矩陣A 並printA的size 大小.(30pt)

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \end{bmatrix} \leftarrow$$

i.e. 宣告一個形態為ndarray的一維陣列Element: [1 0 0 0 0 0 3 0 0 0 0 5 0 0]

方法1.請以for迴圈將zero\_A內的元素依次替換成element的元素。(15pt)

方法2.請利用教學中的numpy 套件中function將Element直接reshape成A(15pt)

考察內容：

1.構建矩陣的多種方法 2.for 回圈 3.檢查矩陣大小4.矩陣元素的表示方法

Function : np.array()      np.shape() A\_ij in python :A[i,j]      np.reshape()

## 作業說明-程式作業(2-2)

by method 1:

Output:

```
Element =: [1 0 0 0 0 0 3 0 0 0 0 0 5 0 0]
matrix A by method 1:
[[1 0 0 0 0]
 [0 3 0 0 0]
 [0 0 5 0 0]]
A's shape is : (3, 5)
```

by method2:

Output:

```
matrix A by method 2 :
[[1 0 0 0 0]
 [0 3 0 0 0]
 [0 0 5 0 0]]
```

提示：

方法一需要寫巢狀for回圈，即兩個for回圈，實作時請注意縮排！

評分規則：**output需要與範例完全相同，包括型態！**

方法一15 point，3個output，每個output 5 point，

方法二15 point 1個output，成功輸出即得分。

# 作業說明-程式作業 ( 3-1 )

題目3:根據提供的算法和步驟Solve diagonal linear system ! (40pt)

Note:pseducode 中陣列的index從1開始, python從0開始, 注意區分!!!

Algorithm of diagonal system by for 迴圈:

```
given A, b
for i = 1 . . . n
    x_i = b_i / a_i,i
end
```

step 1:given A,b:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{bmatrix} \quad b = \begin{bmatrix} -1 \\ 6 \\ -15 \end{bmatrix}$$

考察內容：

1.矩陣運算  $Ax=b$  2.for回圈 3.矩陣運算符號的應用  $'/'$

Function : Np.zeros() np.array() Np.diag() np.astype()



## 作業說明-程式作業 ( 3-2 )

step 2\_v1:by for 迴圈 (20pt)

**Output: x= [-1 2 -3]**

Note:The type of x'output must be int!

step 2\_v2:by vector compute (利用 operator “/” and np.diag())(20pt)

**x=b/diag(A)**

**Output: x= [-1 2 -3]**

Note:The type of x'output must be int!

評分規則：**output**需要與範例完全相同，包括型態！

方法一20 point，1個output

方法二20 point，1個output，成功輸出即得分。

# 作業說明-REPORT

Report 需包括：

1.姓名，系級，學號

2.題目0

3.心得（本次學習的收穫或建議）

4.分數：心得與題目0共同占本次project 10分，缺任一樣10分都拿不到！