Python – Introduction

Python (基礎班)

Department of Engineering
IVE(Lee Wai Lee)
香港專業教育學院（李惠利）工程系

https://www.ive.edu.hk/lwl/ivesite/html/tc/campus_lwl_campusoffices_emt_overview.html

Revised: April 2025

Learning Outcomes 學習內容

After completion of the 2-hour taster programme, students should be able to:

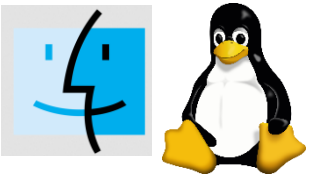
- Install Python and IDE on PC and use them to write codes
- Understand the Python basics (e.g. syntax, comments, variables, data types, numbers, Booleans, strings, lists, , standard libraries, Function)
- Understand simple conditionals (e.g. if statement, for and while loops)
- write a Python code to solve simple math problem

完成 2 小時的體驗課程後，學生應能：

- 在PC上安裝Python和IDE，並嘗試編寫程式碼
- Python 基礎知識 (e.g. 變數與資料類型, 函数, 標準函式庫)
- 選擇 (e.g. if statement, for and while 迴圈)
- 課堂作業 - 編寫 Python 程式碼來解決簡單的數學問題

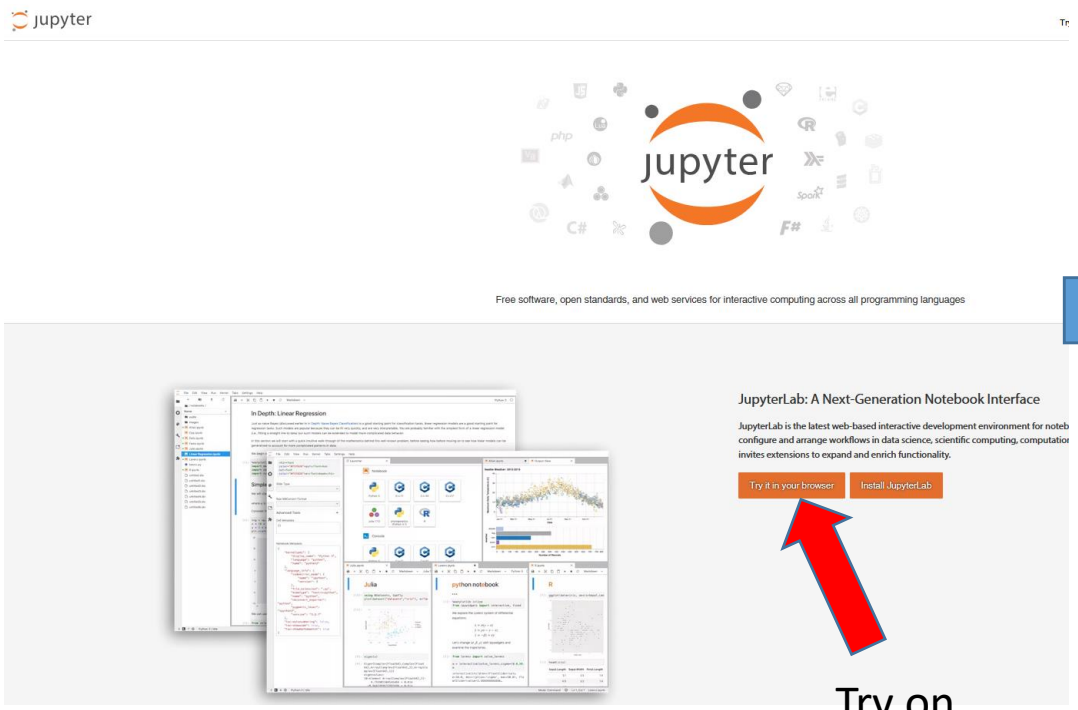
Why Python?

- Easy to use
- Active community
- Compatible with many platforms
- Excellent ranges of libraries, packages
- Data science
- Machine learning
- Web development
- Career opportunities

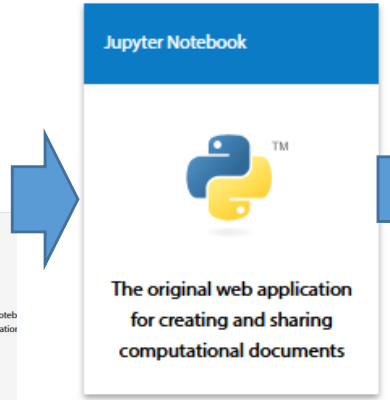


Just want to try Python without installing anything?

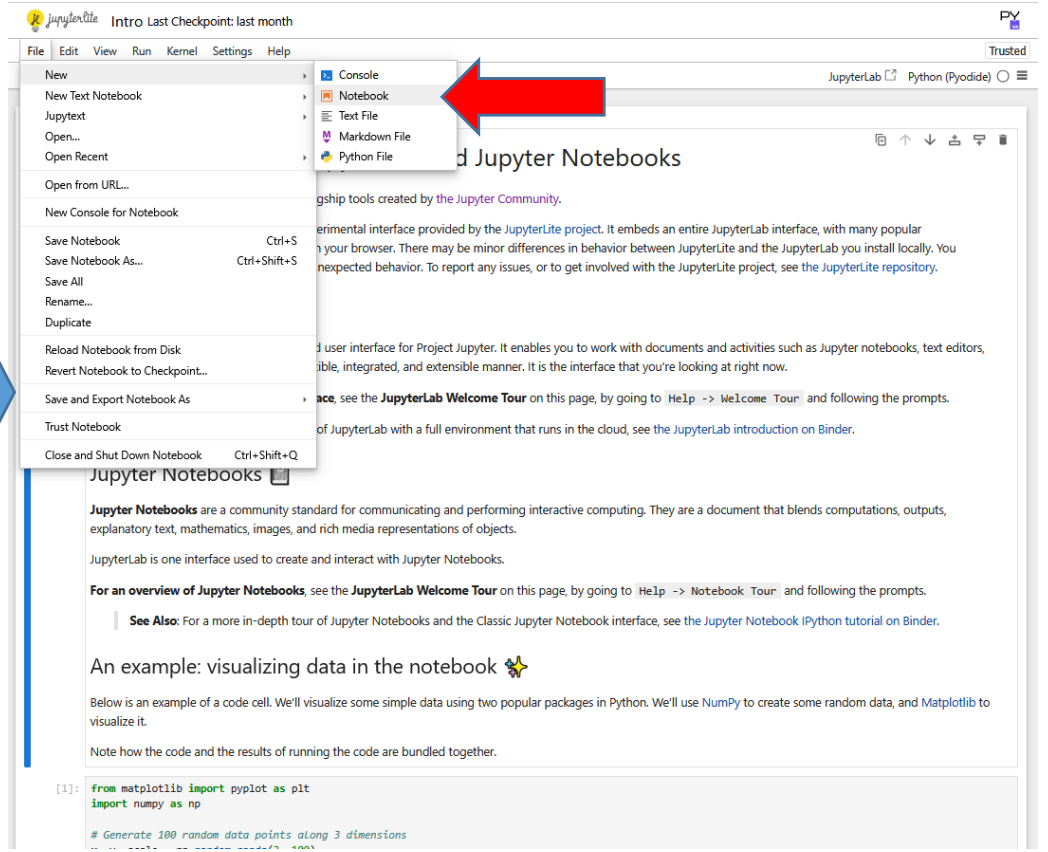
- Go to https://jupyter.org/



Try on browser

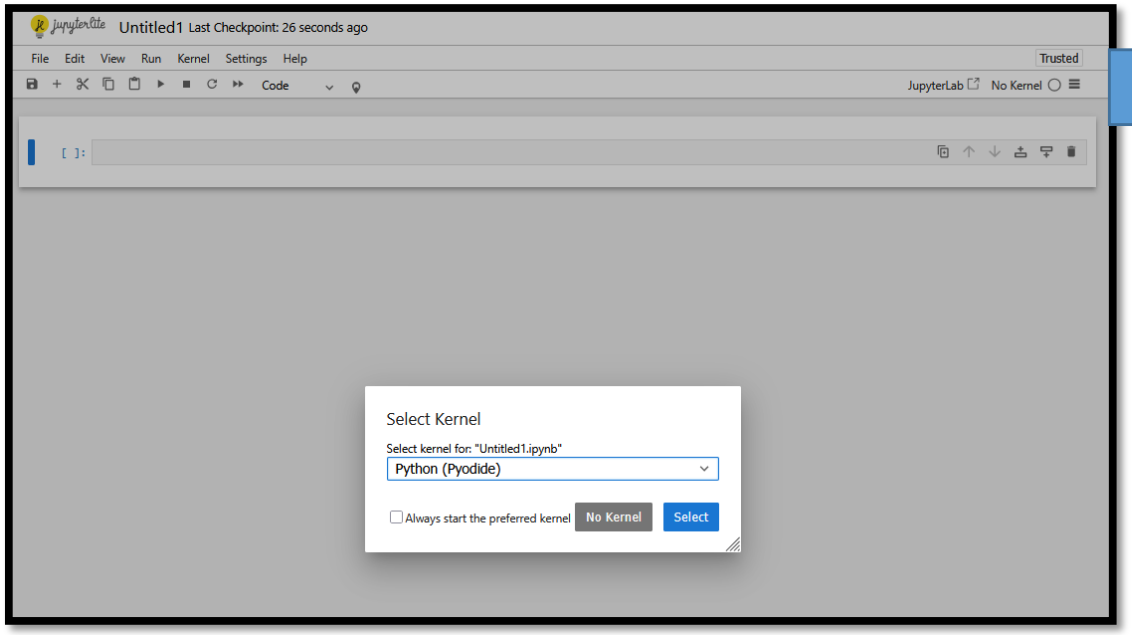


File → New → Notebook

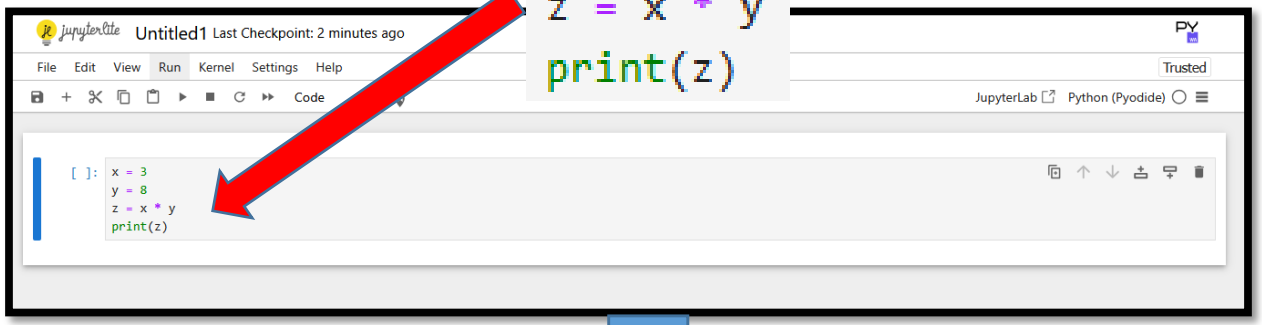


Using Jupyter Notebook online (no download)

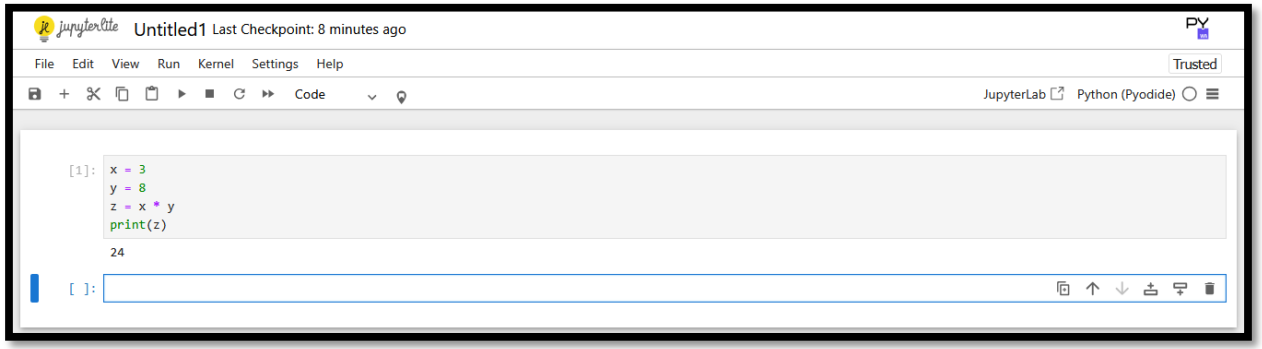
- Select the right kernel




```
x = 3
y = 8
z = x * y
print(z)
```



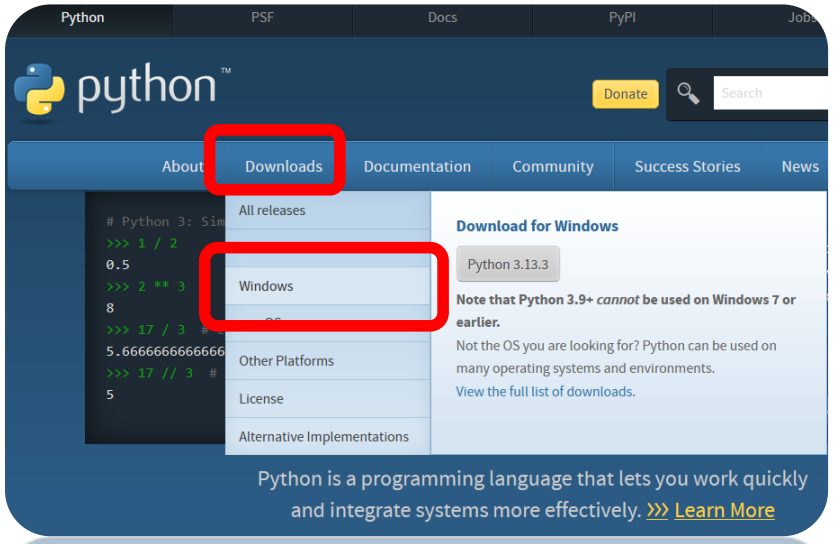
Run the selected cell by pressing
"Shift" + "Enter" or ▶



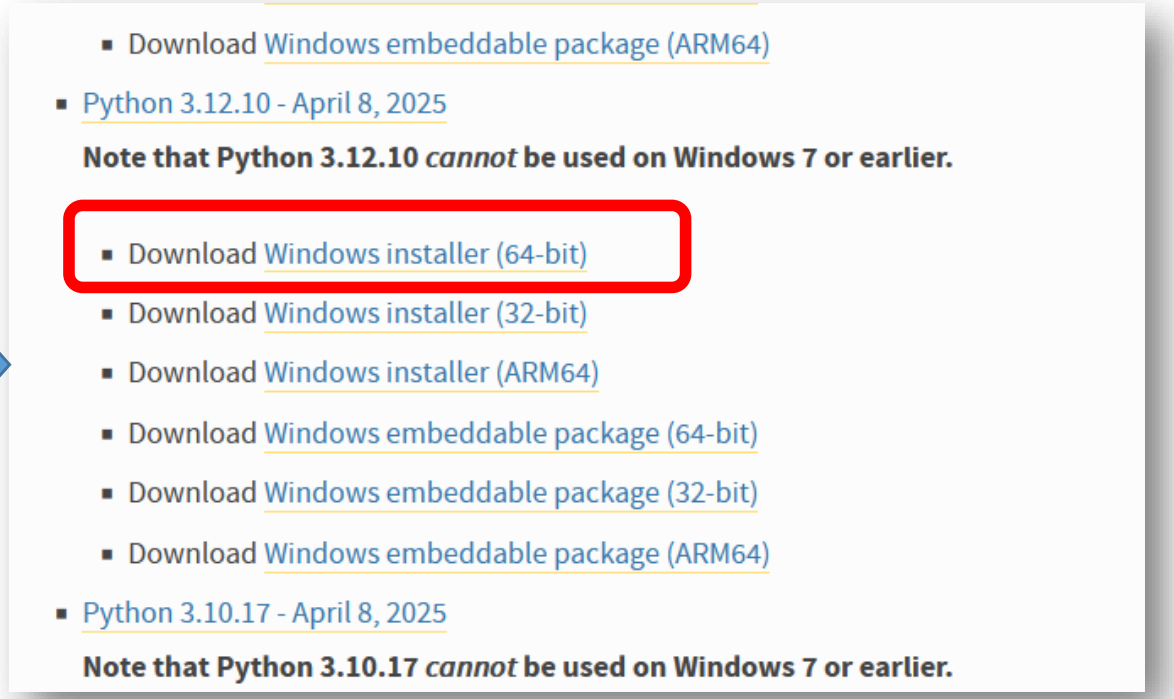
Python on your PC |安装 **Python**

- https://www.python.org/

1. Downloads → Windows



2. Scroll down and find Python 3.12.10, download the installer for Windows (64-bit)

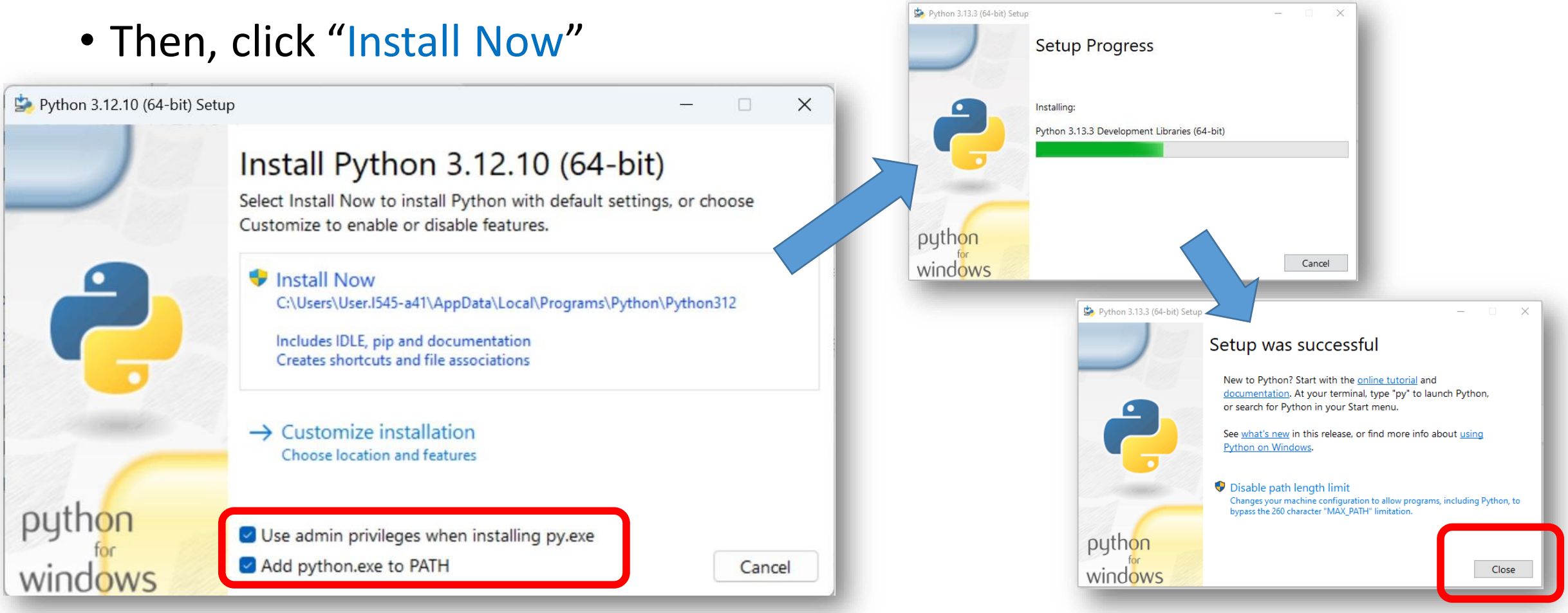


3. From your Download folder, double click the application file (python-3.12.10-amd64.exe) that you just downloaded

Install Python 3.12.10 (64-bit)

Python 最新的版本是 3.13.3 ，但有些 Framework (如 TensorFlow) 暫時未支援此版本，所以本課程是使用 3.12.10 版本。

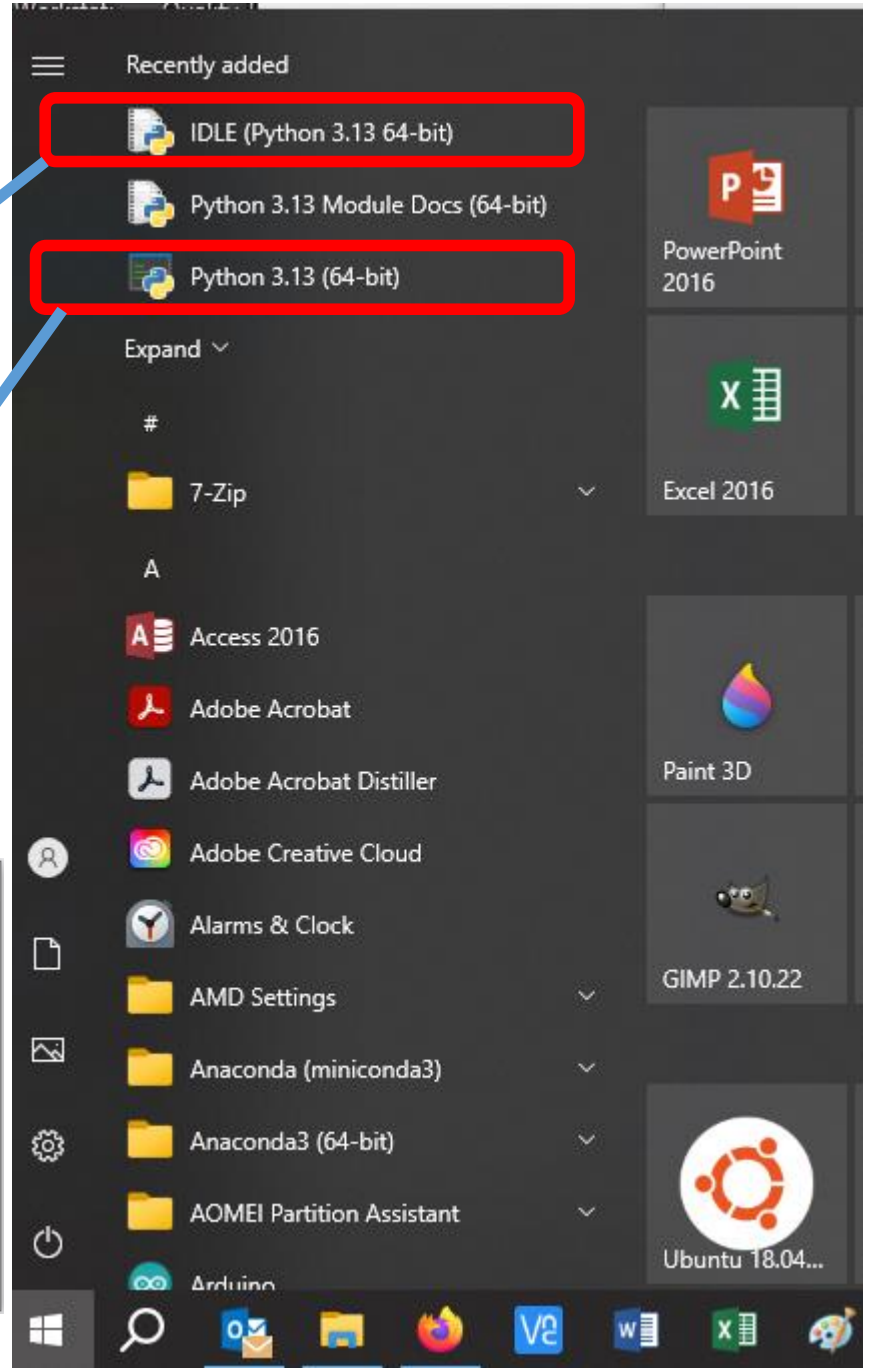
- Make sure “Add python.exe to PATH” is checked ☑
- Then, click “Install Now”

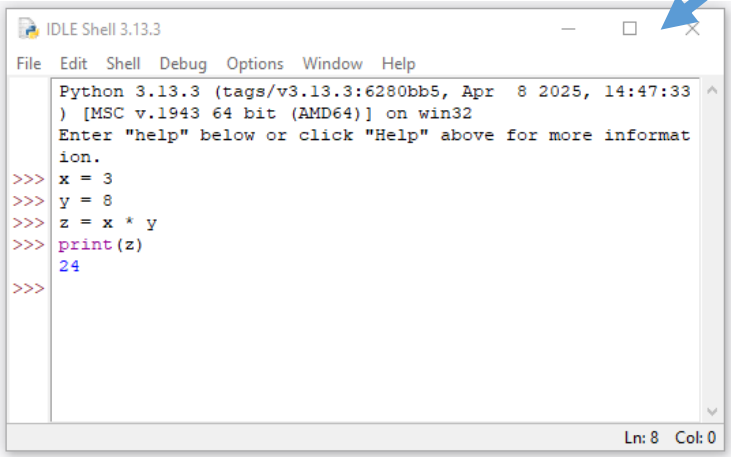


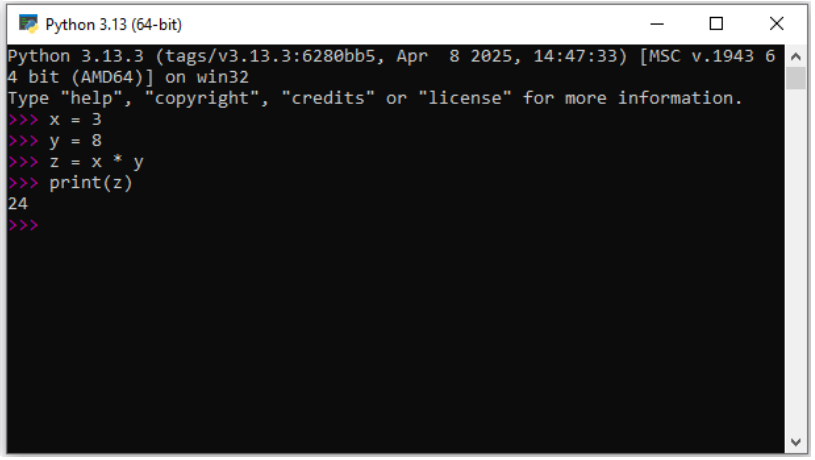
Python installed

For Windows 10

- Try the **IDLE** and Python Shell that are installed together
- **IDLE** is Python's **I**ntegrated **D**evelopment and **L**earning **E**nvironment
- Python Shell window



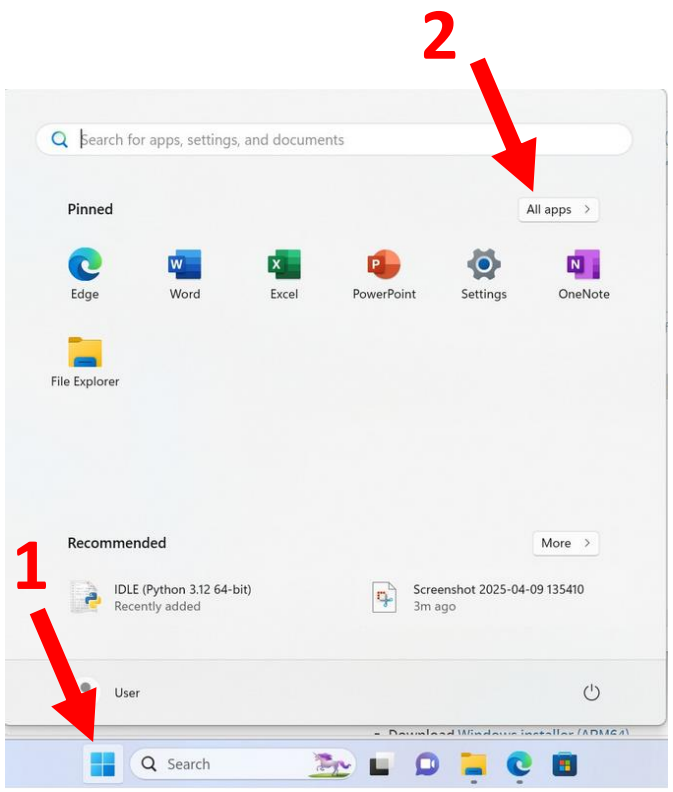


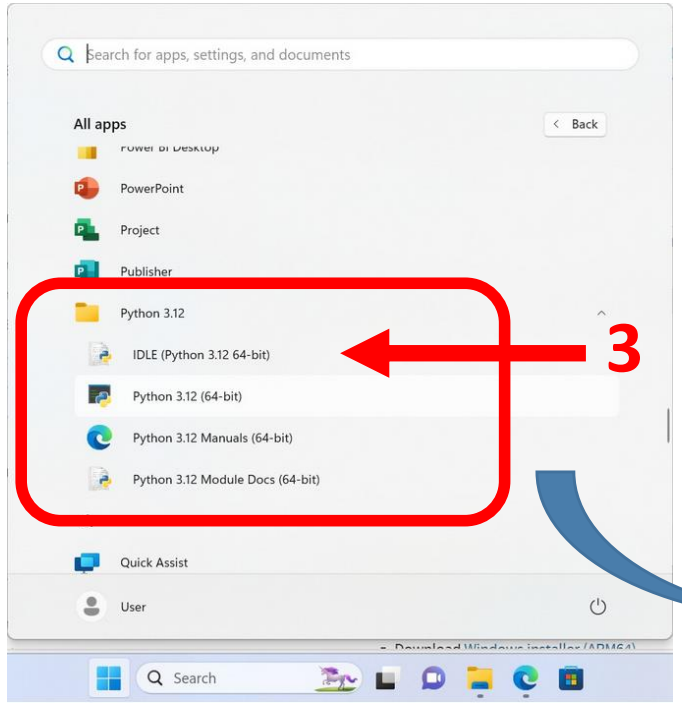


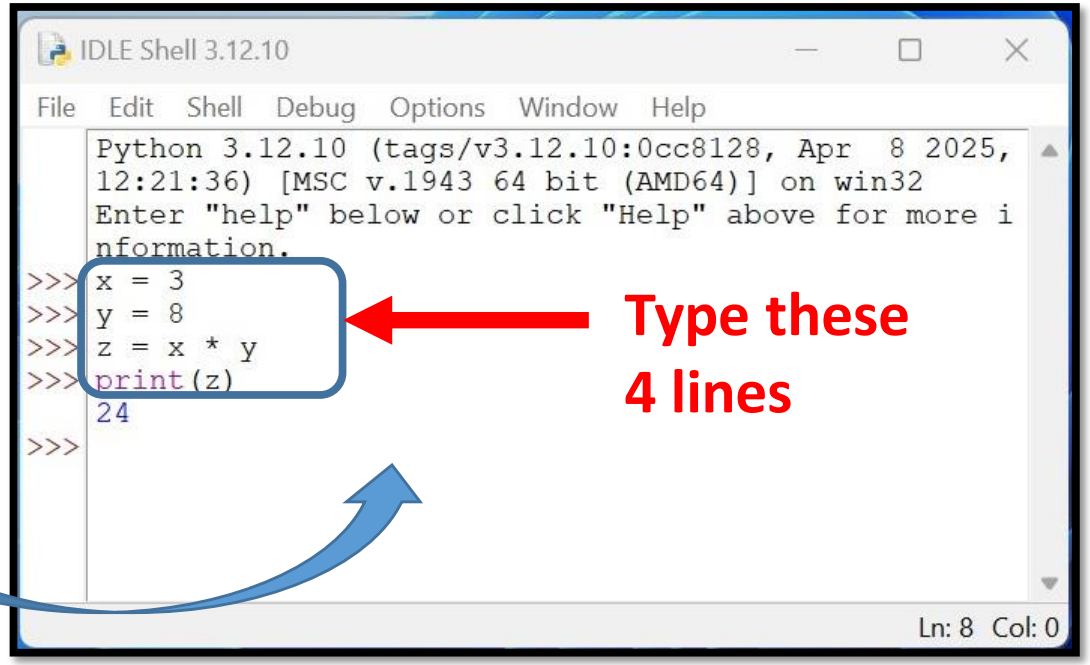
Run Python

For Windows 11

- Run the Python IDLE Shell after Python is installed





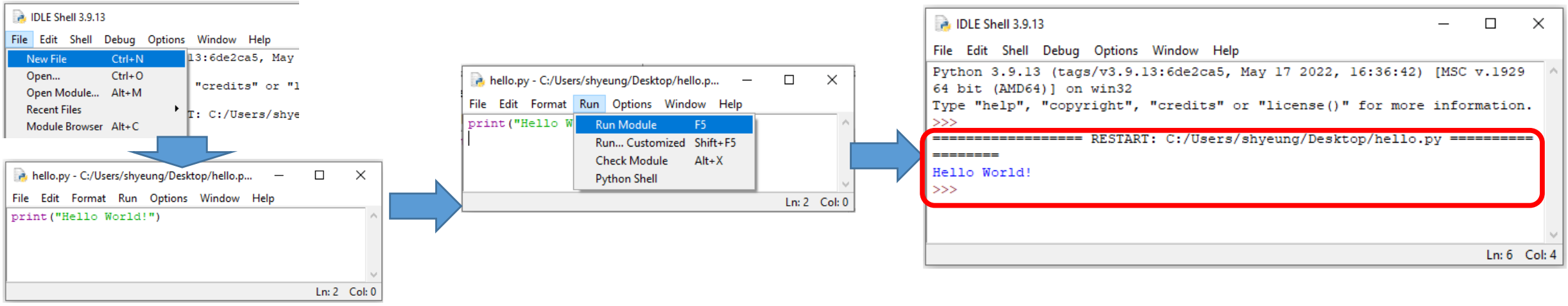


Further use of Python IDLE

- 在**IDLE** 程式，選取 “**File**” => “**New File**”
- 鍵入以下程式：

```
print ("Hello World")
```

- 將程式儲存為 “**hello.py**”。選取 “**Run**” => “**Run Module**” 或按 **F5** 鍵去運行程式。



Some good IDE (Integrated Development Environment)

- There are several popular Python IDEs that your can use for free

PyCharm

https://www.jetbrains.com/pycharm/

VS Code

https://code.visualstudio.com/

Sublime Text

https://www.sublimetext.com/

Atom

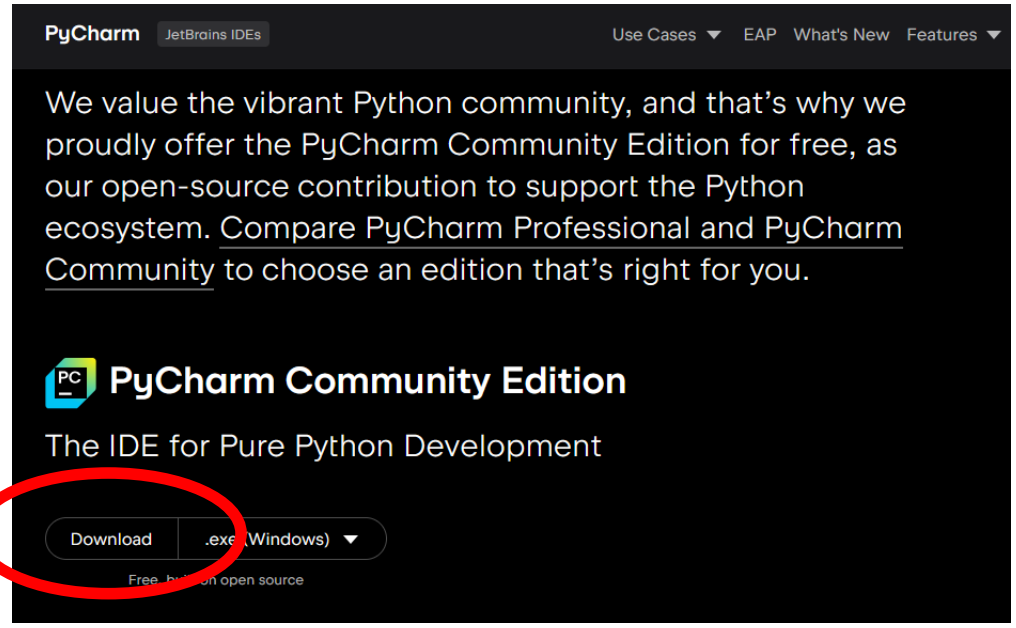
https://atom-editor.cc/

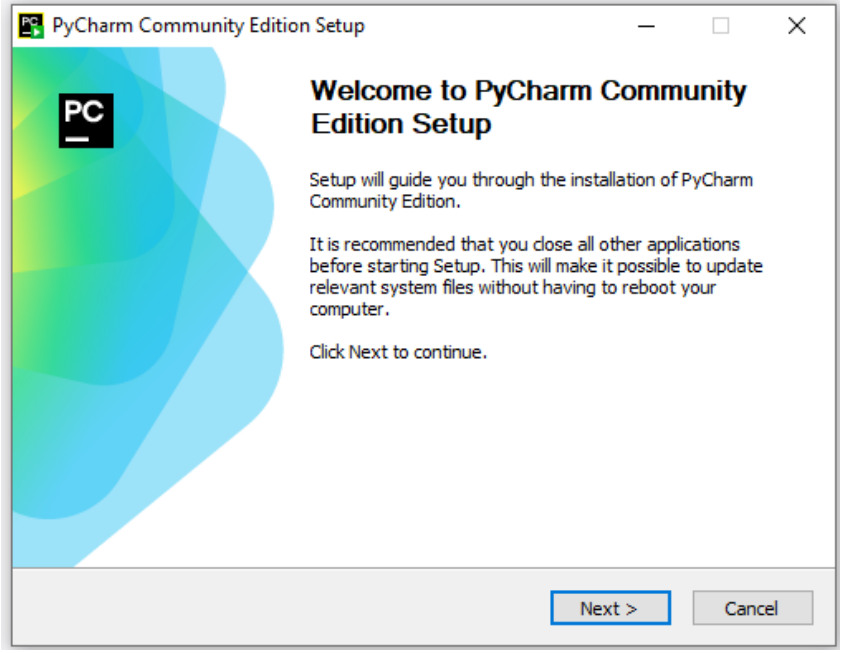
Thonny

PyCharm

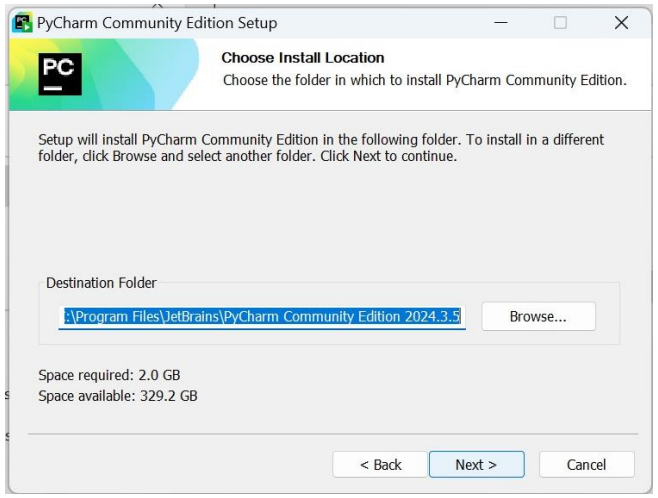
- Download PyCharm Community Edition

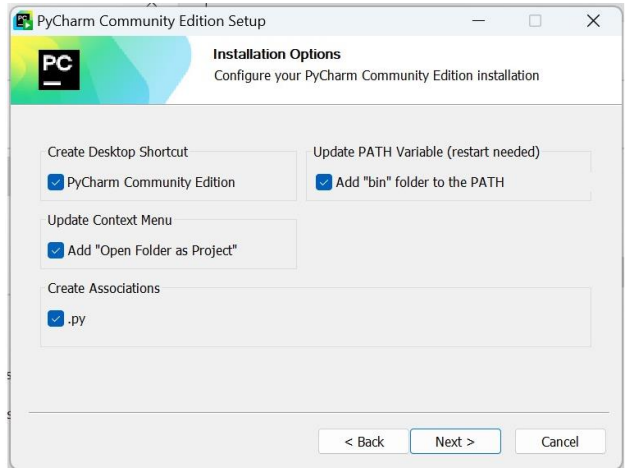
1. Scroll down and find the PyCharm Community Edition. It is a free open source software. Click "Download"
2. Go to Downloads folder and double click the pycharm-community-2024.3.5.exe file to start installation

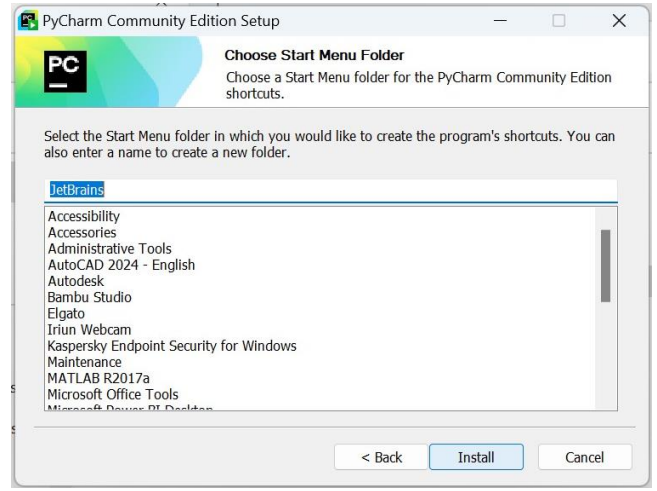


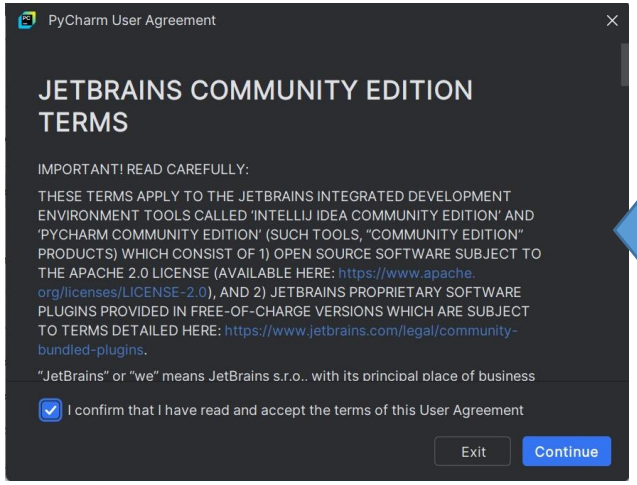


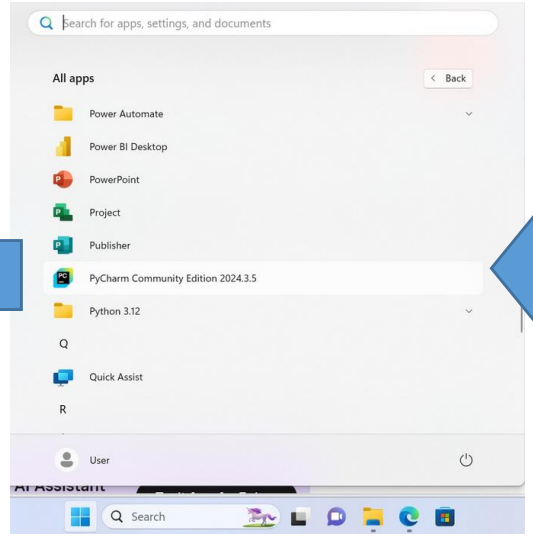
Install and run PyCharm (1)



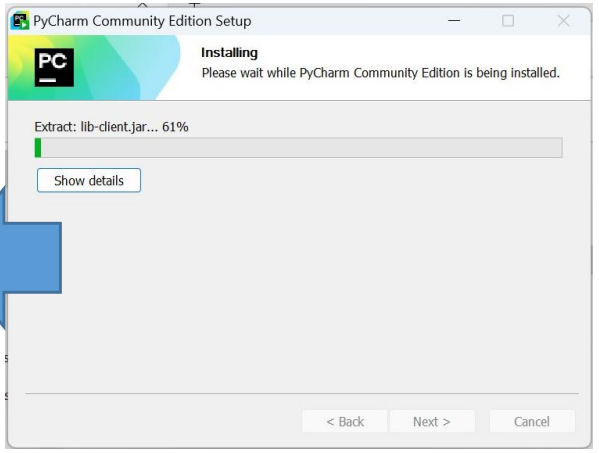




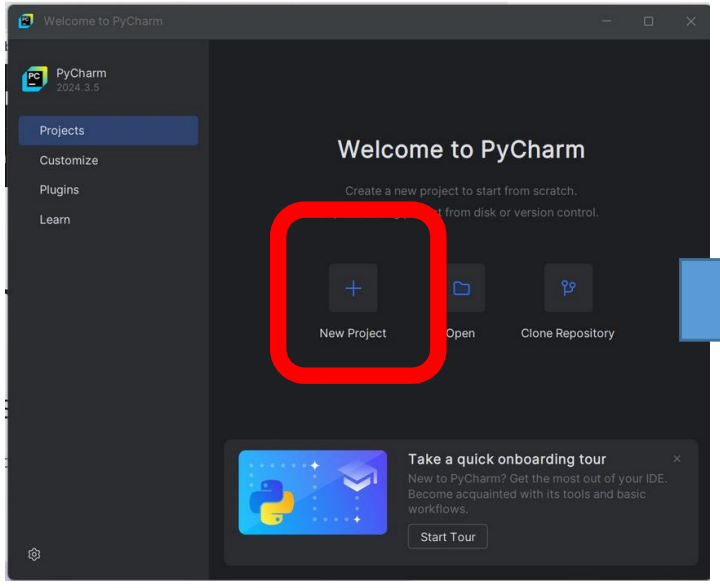


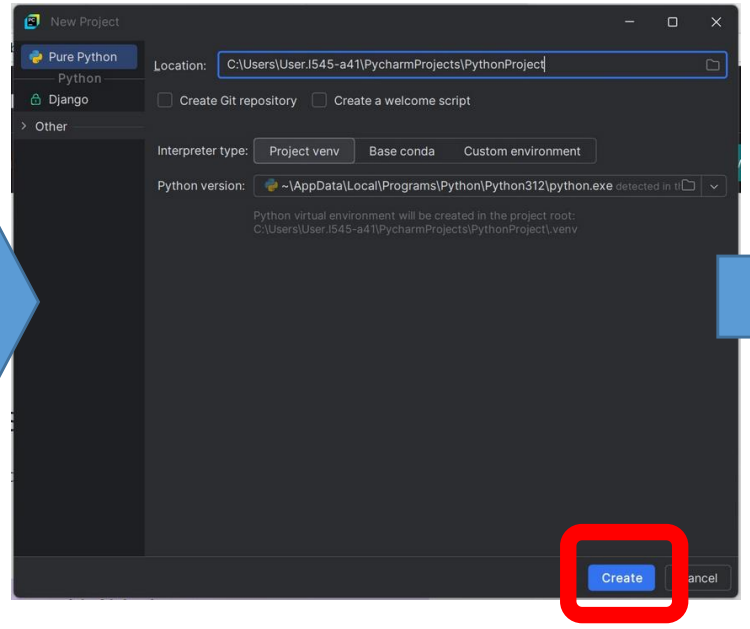


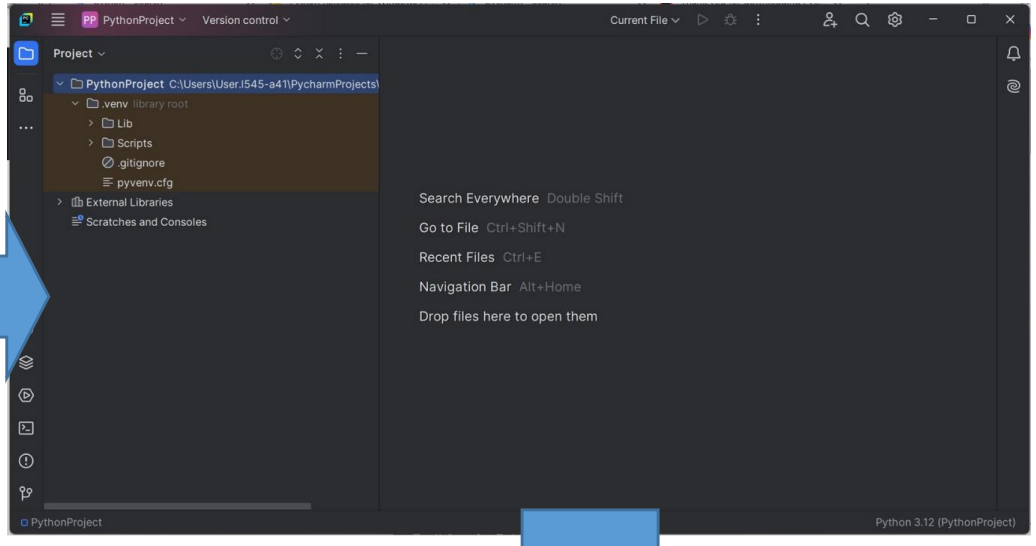
DO NOT RESTART YOUR COMPUTER

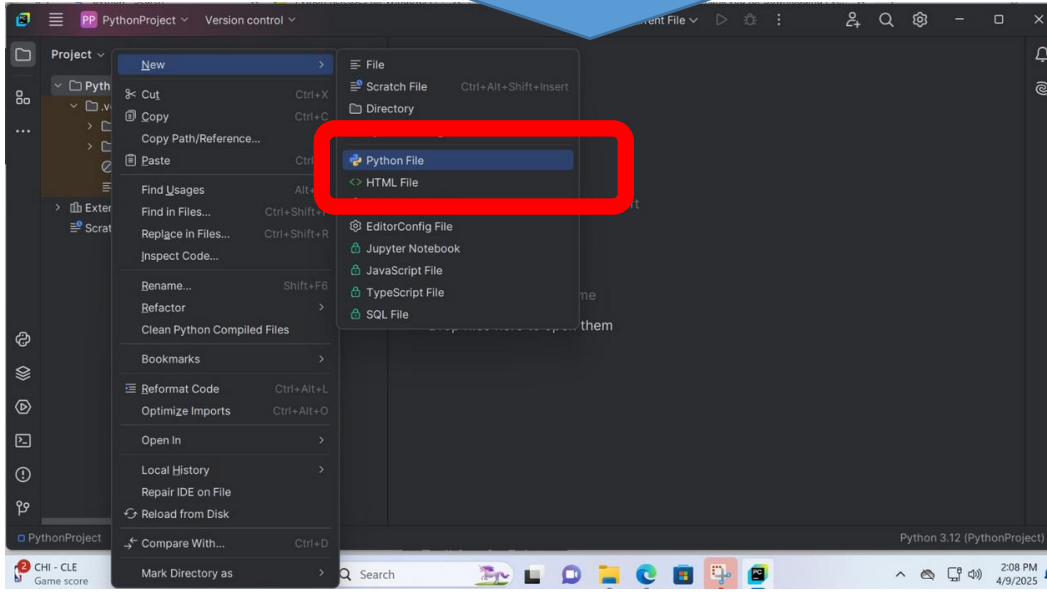


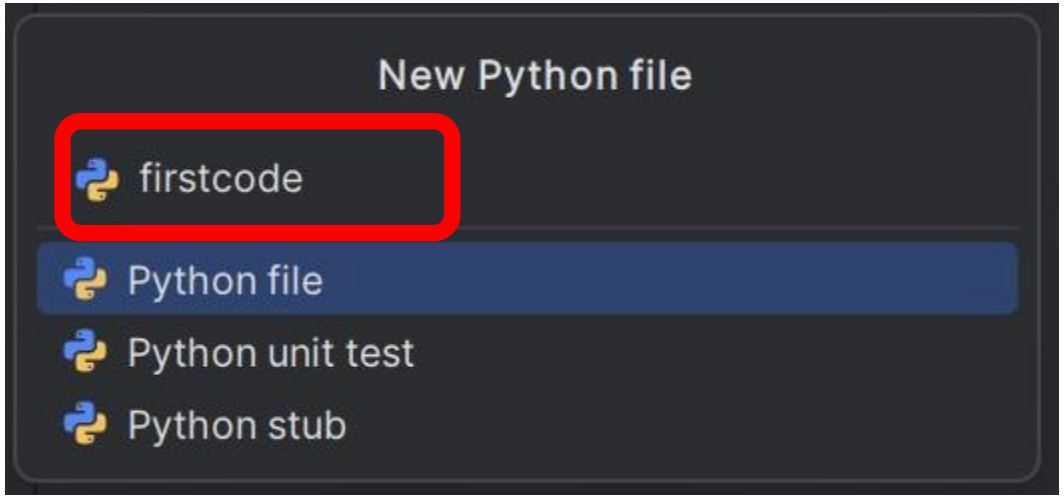
Install and run PyCharm (2)



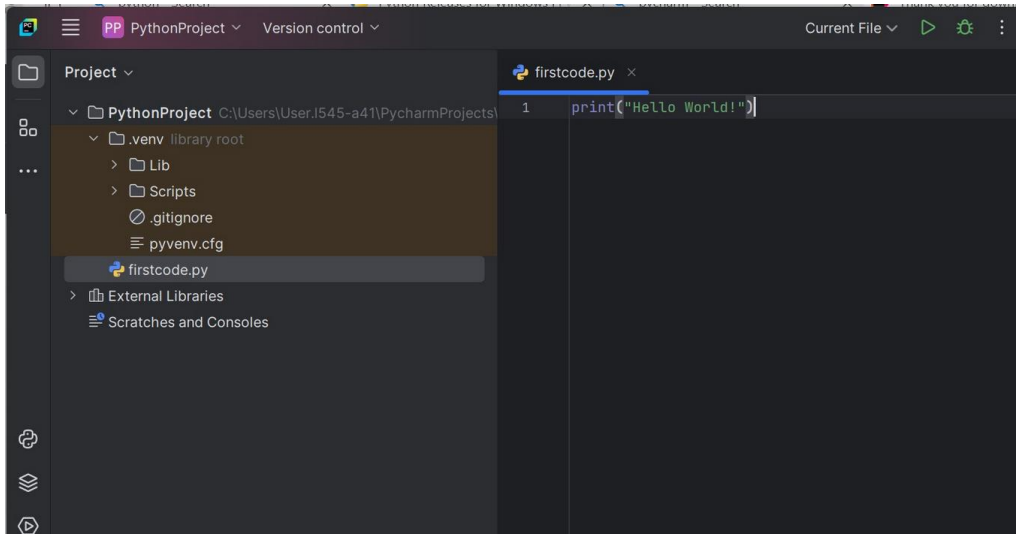


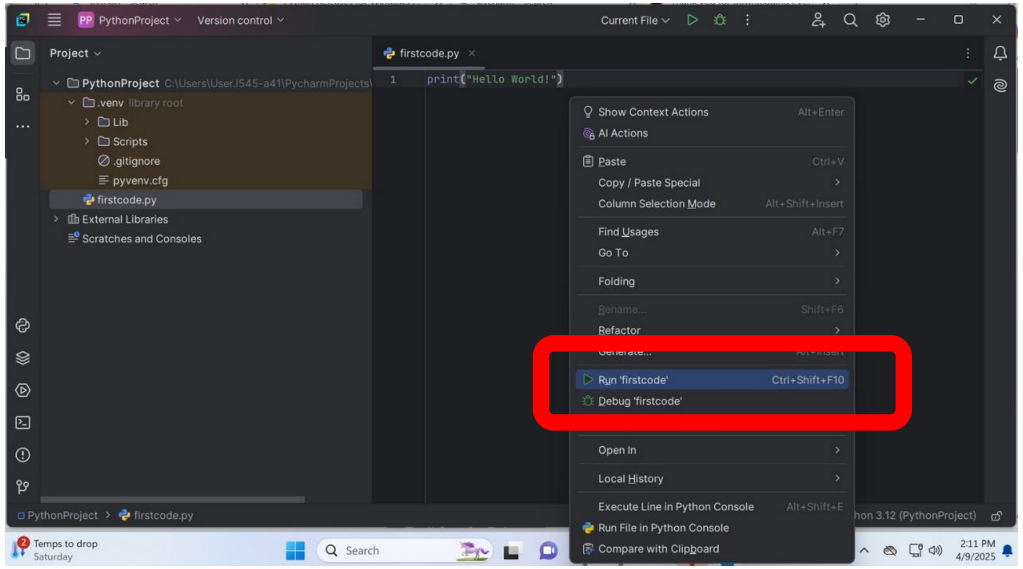


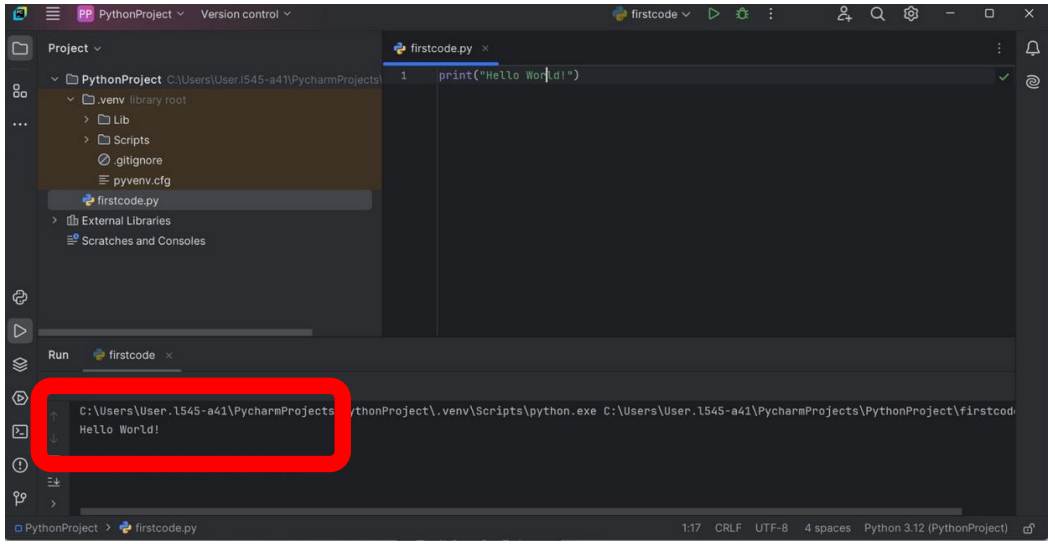




Install and run PyCharm (3)

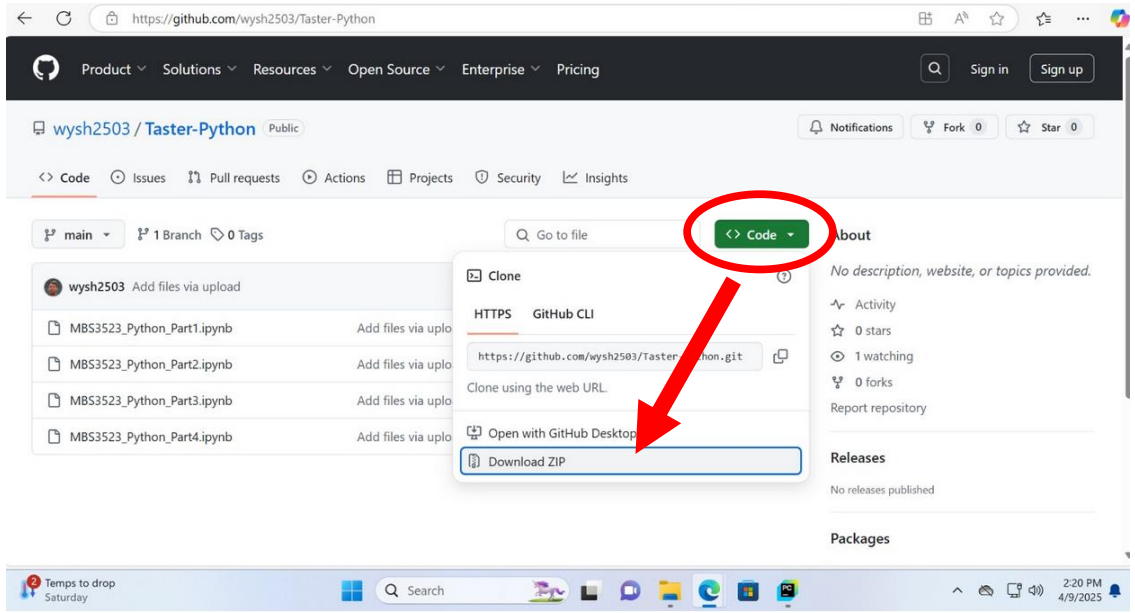


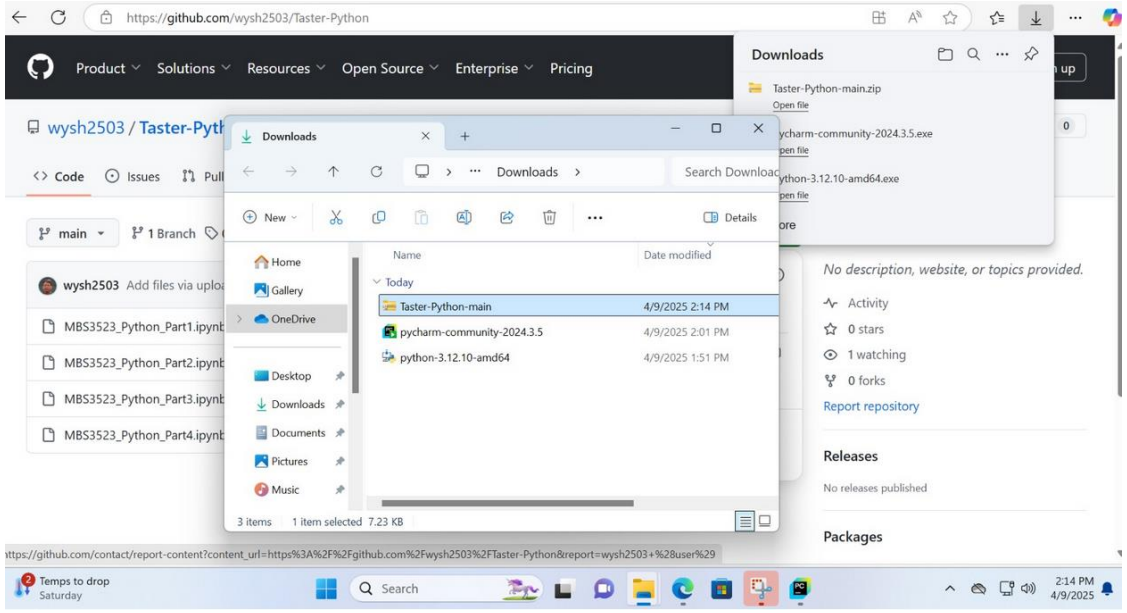




Time for practice |練習時間

- Get the file from https://github.com/wysh2503/Taster-Python
- Download the files (Code → Download ZIP), open the Download folder, right click the Taster-Python-main ZIP file and select “Extract All…” to unzip the folder, save them on your Desktop
- Drag and drop to your Jupyter Notebook





[1] 變數（variable）與資料類型（type）

- 變數是程式中用來暫存數據以作計算、輸入和輸出之用。
- 鍵入以下程式及運行：

```
strName = input ("Please enter your name:")
print ("Hello " + strName)
```

輸出結果為：

```
Please enter your name: IVE(LEE WAI LEE) - ENGINEERING
Hello  IVE(LEE WAI LEE) - ENGINEERING
>>>
```

- **input()** 函数是讀取使用者的數據。
- **strName** 是一個變數，用來暫存使用者的名字 (例子中的 **IVE(LEE WAI LEE) - ENGINEERING**)。而它的類型 (type) 是字串 (string)。

[1] 變數（variable）與資料類型（type）

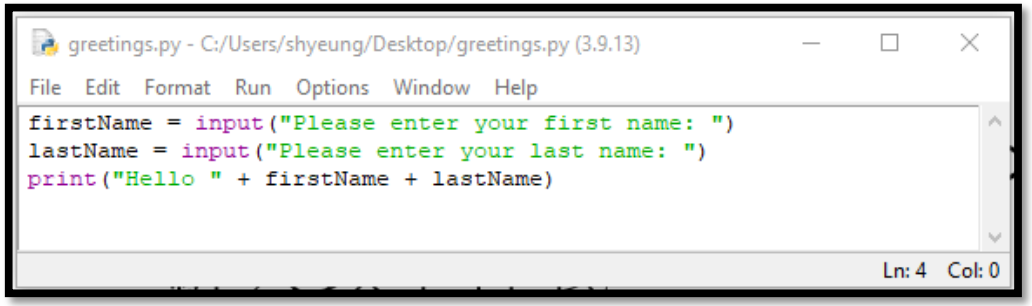
練習一

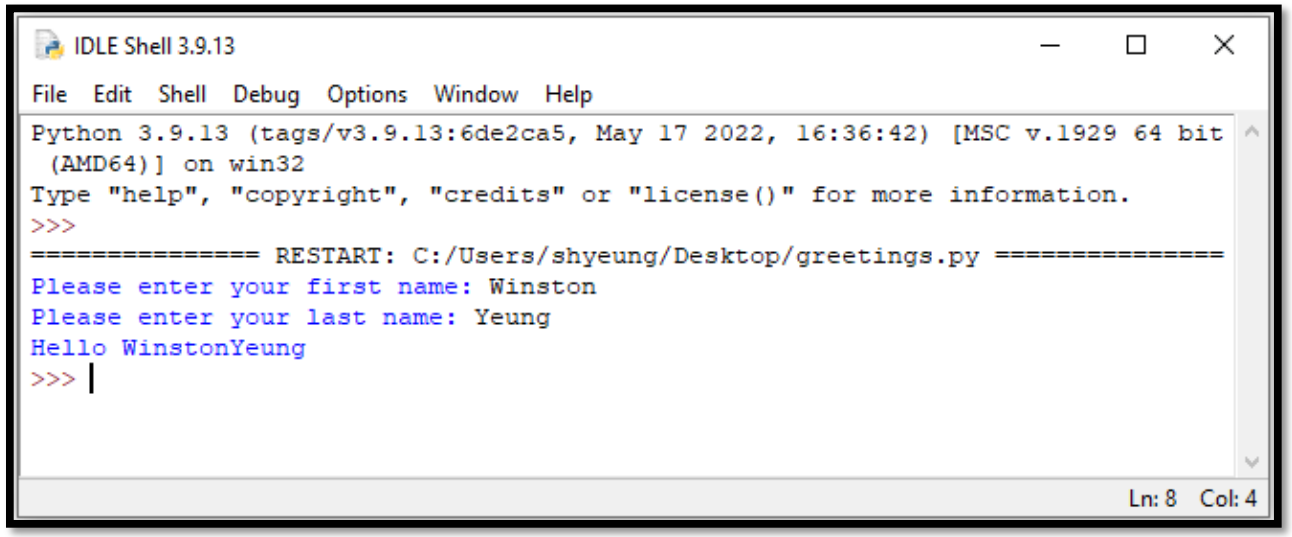
寫一個程式 (greetings.py)：讀取使用者的**名字(first name)**及**姓氏(last name)**，然後輸出 Hello [名字]及[姓氏]

```
Please enter your first name: Winston
Please enter your last name: Yeung
Hello WinstonYeung
>>>
```

練習一 (答案)

- 使用**Python IDLE**





[1] 變數（variable）與資料類型（type）

- 鍵入以下程式及運行 (add.py)：

```
num1 = int(input ("Please enter a number: "))
num2 = int(input ("Please enter another number: "))
add = num1 + num2
print (num1, "+", num2, " = " + str(add))
```

輸出結果為：

```
Please enter a number:1
Please enter another number:3
1 + 3  =  4
>>>
```

- **int** () 函数是將字串轉換為整數。
- **str** () 函数是將整數轉換為字串。
- **num1**和 **num2**是整數類型的變數，用來暫存使用者輸入的兩個整數。**add** 是用來暫存運算的結果及輸出時使用。

[1] 變數（variable）與資料類型（type）

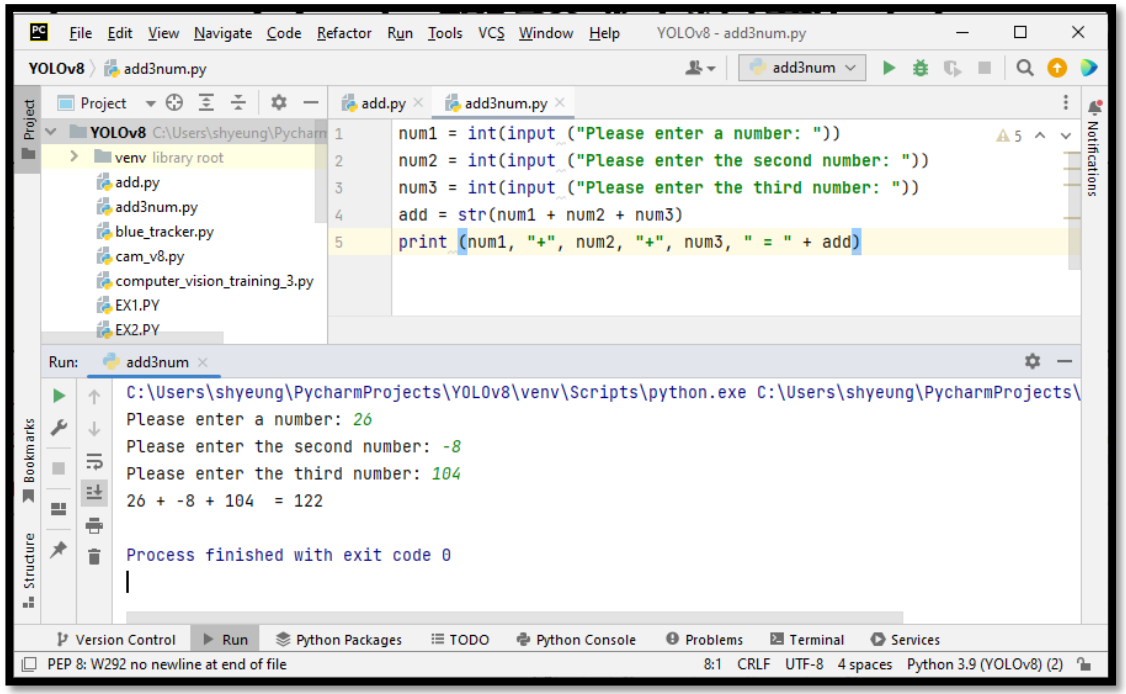
練習二

寫一個程式 (add3numbers.py)：讀取 3 個數字(num1, num2, num3)並將它們相加(add)

```
Please enter a number: 26
Please enter the second number: -8
Please enter the third number: 104
26 + -8 + 104  = 122
```

- 練習二(答案)
- 使用**PyCharm**
- Run the code and test with any 3 numbers

```
num1 = int(input ("Please enter a number: "))
num2 = int(input ("Please enter the second number: "))
num3 = int(input ("Please enter the third number: "))
add = str(num1 + num2 + num3)
print (num1, "+", num2, "+", num3, " = " + add)
```



[2] Conditionals | 選擇 (if … else …/ if … elif … else)

- 寫程式時，很多時需要作分流處理。例如測驗成績達40分為合格，否則為不合格；使用者為男或女，會有不同的處理。這時可以使用 if - else 敘述。鍵入以下程式及運行 (M_F_selection.py)：

```
sex = str(input ("Your test sex (male/female): "))
# check the test mark {# 符號是用來加入注解，編程器會跳過這部份。}
if sex == "male":
   print("Go to the LEFT toilet!")
else:
   print("Go to the RIGHT toilet!")
```

輸出結果為：

```
Your test sex (male/female): male
Go to the LEFT toilet!
```

```
Your test sex (male/female): female
Go to the RIGHT toilet!
```

如果 sex == "male"的檢查結果為真(True)時，程式會執行 if 以下的敘述，上例是： print("Go to the LEFT toilet!") ，否則會執行else以下的敘述，上例是： print("Go to the RIGHT toilet!")。
注意：程式中的 ：及 縮排(Indentation)。

[2] Conditionals | 選擇 (if … else …/ if … elif … else)

練習三

寫一個程式 (checkGrade.py)：讀取測驗成績，然後根據下列表格輸出適當的等級：

測驗成績	等級
>=80	Pass with Distinction
60 - <80	Pass with Credit
40 - <60	Pass
< 40	Fail

- 練習三(答案)
- 使用**PyCharm**
- Run the code and test with the following marks:
  - 85, 67, 42.5, 30

```
testMark = float(input ("Your test mark: "))
# check the test mark
if testMark >= 80:
    print("Pass with Distinction!!")
elif testMark >=60 and testMark<80:
    print("Pass with Credit!")
elif testMark >=40 and testMark < 60:
    print("Pass!")
else:
    print("Fail!")
```

[3] Loop | 迴圈 (for, while)

- 編寫程式時，有時需要重複或搜索某一記錄，可以用 **for** 或 **while**。例如要列印 5 行 Hello World，可以執行以下敘述：

```
print ("Hello World")
print ("Hello World")
print ("Hello World")
print ("Hello World")
print ("Hello World")
```

- 但如果要列印 100 行，以上方法便不太方便。
- 使用迴圈 **for**：鍵入以下程式及運行 (forLoopHello.py)：

```
for i in range (0,5):
    print ("Hello World")
```

- 或者
- 使用迴圈 **while**：鍵入以下程式及運行 (whileLoopHello.py)：

```
i = 0
while i < 5:
    print ("Hello World")
    i = i + 1
```

[3] Loop |迴圈 - Nested Loops 巢狀迴圈

- 鍵入以下程式及運行 (printStar.py)：

```
n = int (input("Enter a number n:"))
for i in range (0, n):
    print("* ", end="")
```

- 輸出結果為：

```
Enter a number n:5
* * * * *
>>>
```

- In Python, **a loop inside a loop is known as a nested loop (**巢狀迴圈**)**.
- 加多一個迴圈 (printStar2loop.py)

```
n = int (input("Enter a number n:"))
for j in range (0, n):
    for i in range (0, n):
        print("* ", end="")
    print ("")
```

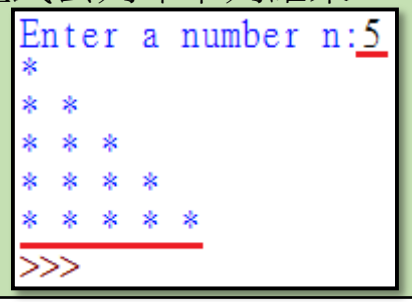
- 輸出結果為：

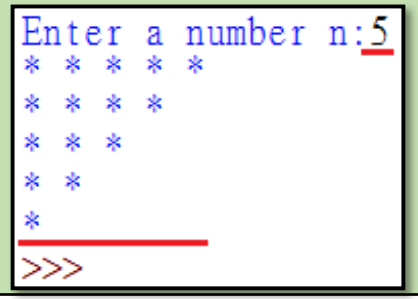
```
Enter a number n:5
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
>>>
```

[3] Loop |迴圈 - Nested Loops 巢狀迴圈

練習四

修改以上程式去列印下列結果：





- 練習四(答案)
- 使用**PyCharm**

```
n = int (input("Enter a number n:"))
for j in range (0, n):
    for i in range (0,     ):
        print("* ", end="")
    print ("")
```

```
n = int (input("Enter a number n:"))
for j in range (0, n):
    for i in range (0,     ):
        print("* ", end="")
    print ("")
```

[4] Library/Standard Library | 函式庫

- 鍵入以下程式 (pythagoras.py)

```
import math
a = float(input("a= "))
b = float(input("b= "))
c = math.sqrt(a*a + b*b)
print ("c= %.2f" % c)
```

- **import math** 載入函式庫 **math**，使用**math.sqrt** 來計算開方。
- Try and run the code

```
a= 21
b= 8
c= 22.47
```

```
a= 3
b= 4
c= 5.00
```

[5] Python Functions |函数

- A function is a block of code which only runs when it is called.
- You can pass data, known as parameters, into a function.
- A function can return data as a result.

```
def printString( strPrint, n):
    for i in range (0, n):
        print (strPrint)

printString ("This Taster Programme is interesting", 6)
```

- Result

```
C:\Users\shyeung\PycharmProjects\YOLOv8\ver
This Taster Programme is interesting
This Taster Programme is interesting
This Taster Programme is interesting
This Taster Programme is interesting
This Taster Programme is interesting
This Taster Programme is interesting

Process finished with exit code 0
```

[5] Python Functions |函数

- Sometimes you might want to return the calculated/manipulated result from the function:

```
def addition(num1, num2):
    answer = num1 + num2
    return answer
a = float(input("a="))
b = float(input("b="))
print (a, "+", b, "= %.2f" % addition(a, b))
```

- Result

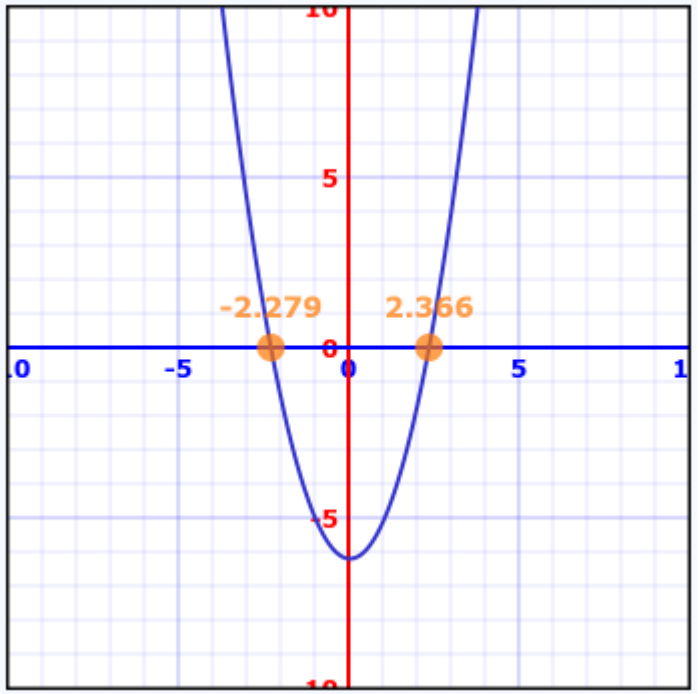
```
a=3.8
b=6.9
3.8 + 6.9 = 10.70
```

Inclass Exercise 1

- 寫 一個Python 程式碼(QuadraticEquation.py)來找出二次方程式的根（ 當 y = 0 時 ）

- Given: $y = 1.15x^2 + 0.1x - 6.2$
- Write a Python code to find the roots (when y = 0) of the above quadratic equation

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

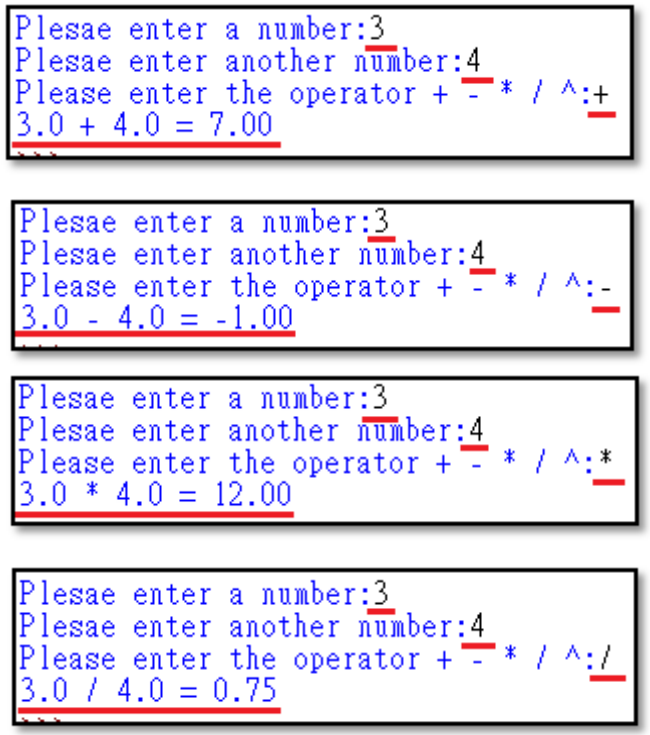


Reference:
https://www.mathsisfun.com/algebra/quadratic-equation-graph.html

Inclass Exercise 2

- 寫一個程式 (calculator.py)：讀取兩個實數及一個運算符號，然後輸出其運算結果。[注意：程式中需要使用函数。]

實數1	實數2	運算符號	計算
a	b	+ (加)	a + b
a	b	- (減)	a - b
a	b	* (乘)	a * b
a	b	/ (除)	a / b
a	b	^ (次方)	a ^ b



Suggested answer for Inclass Exercise 1

```python
import math

# Define coefficients
a = 1.15  # coefficient of x²
b = 0.1   # coefficient of x
c = -6.2  # constant term

# Method 1: Using quadratic formula
# Quadratic formula: x = (-b ±√(b² - 4ac)) / (2a)
discriminant = b**2 - 4*a*c

if discriminant < 0:
    print("No real roots exist (discriminant < 0)")
elif discriminant == 0:
    x = -b / (2*a)
    print(f"One real root: x = {x:.4f}")
else:
    x1 = (-b + math.sqrt(discriminant)) / (2*a)
    x2 = (-b - math.sqrt(discriminant)) / (2*a)
    print("Two real roots:")
    print(f"x1 = {x1:.4f}")
    print(f"x2 = {x2:.4f}")
```


Thank You!
alt