CSE5APG
Week 1: Lab01

OBJECTIVES

- How to install Python.
- How to use the Python interpreter.
- How to implement basic Input, Process and Output.

1- Install Python

Before you can run Python programs on your computer, you need to download and install Python. Below is a description on how to install Python on Windows operating system. For Mac and Linux, you can follow the instructions on the Python download page.

Download and install Python (The description is based on a trial on 01 Feb 2020.)

1. Go to the Python download page: **http://python.org/download**

2. Click on button **Download Python 3.8.1** (or whatever the latest version is, depending on the date of downloading).

   Check that the download file (e.g., python-3.8.1.exe) is displayed at the bottom of the browser. Drag this file to a directory on your PC to save it.

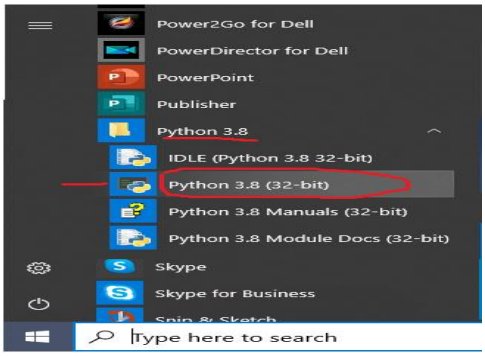
3. Double-click on the installation file to install Python.

   In the Setup window, select both options provided (Install Launcher for all users and Add Python 3.8.1 to PATH)

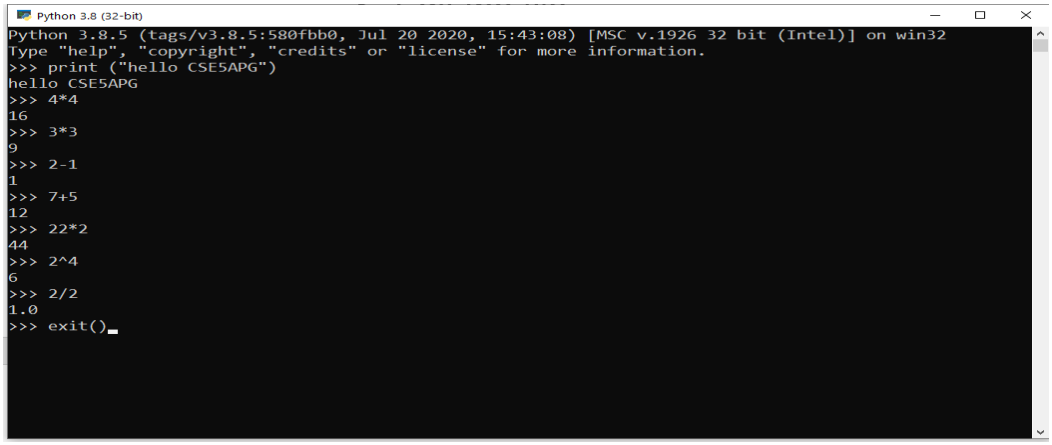
The installer will then complete the installation process.

Testing your installation:

- Open and work with the interactive shell

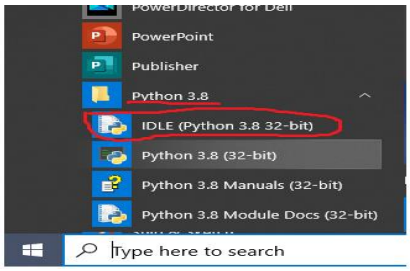


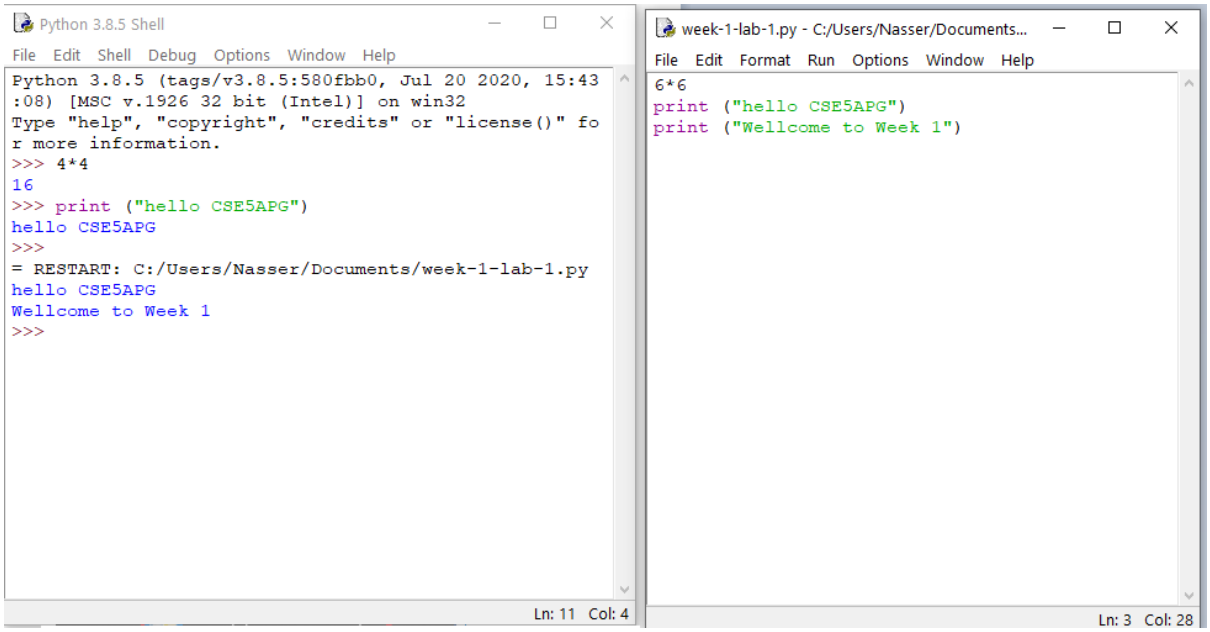
- At the interactive shell command (>>>), enter, for example, expression 4 * 4. You should see 16 displayed.
- Enter command exit() to leave the shell.



Working in IDLE Mode

- Open and work with the IDLE **mode**



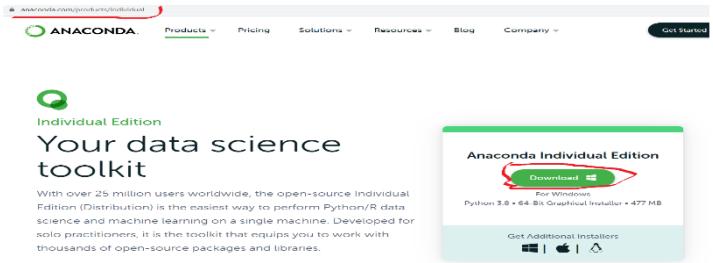


- When we run a program in IDLE, by choosing menu option Run > Run Module, after the execution of the program,

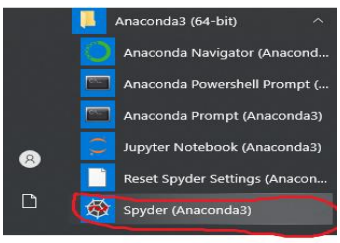
- An interactive shell will be available for further interaction.

Working in Spyder Anaconda

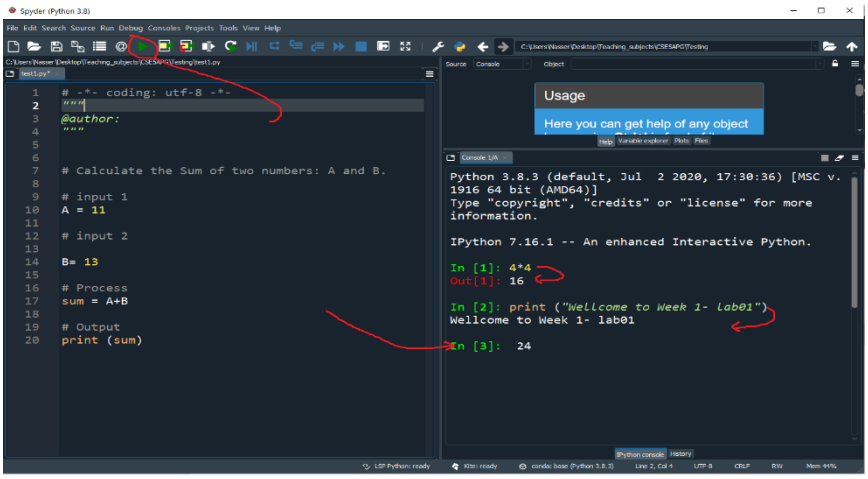
- **Download and install Anaconda** from **https://www.anaconda.com/products/individual**



- Open **Spyder**.

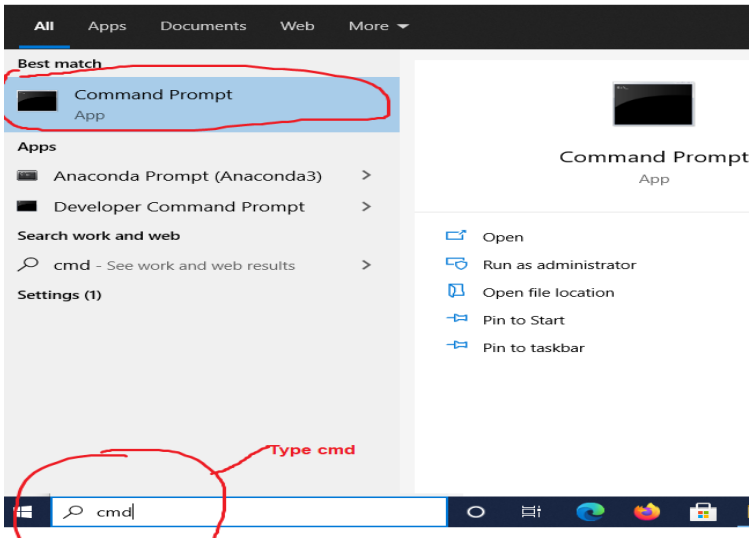


- Write your code in **Spyder editor**

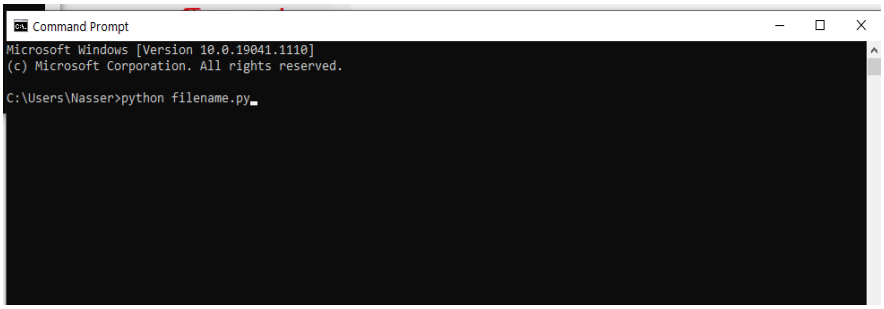


Working in Script mode

- Write your code in any text editor: **Notepad**, **Notepad++, WordPad**, …etc
- **Save your file as: filename.py**
- Open command prompt using **cmd** command as follows:



- Type: python filename.py and hit enter.



2- Implement basic Input, Process and Output

Example 1– Personal Welcome

In the program below, we get some **input** from the user:

```
#  Get the name from the user
name = input ("Please enter your name:  ")
#  Display the welcome message
print ("Welcome to CSE5APG Lab01,", name, "!")
```

Example 2– Adding Numbers

```
# Get two integers
n1 = 5
n2 = 3
#  Add the numbers and   display the result
print ("The total is ", n1+n2)
```

Example 3– Subtract Numbers

```
# Get two integers
n1 = 8
n2 = 10
#  Subtract the numbers and   display the result
print ("The value of n1-n2 is ", n1-n2)
```

Example 3– Arithmetic Operators

```
# Arithmetic Operators
x = 8
y = 4
print('x + y =', x+y)
print('x - y =', x-y)
print('x * y =', x*y)
print('x / y =', x/y)
print('x // y =', x//y)
print('x ** y =', x**y)
```

Example 4– Assignment Operators

```
# Assignment Operators
x = 4
print (x)
x += 2
print (x)
x -= 2
print (x)
x *= 2
print (x)
x /= 2
print (x)
x %= 2
print (x)
x //= 2
print (x)
```

```
x = 6
x **= 2
print (x)
x &= 2
print (x)
x |= 2
print (x)
x ^= 2
print (x)
x >>= 2
print (x)
x <<= 2
print (x)
```

Example 5– Comparison Operators

```
# Comparison Operators
x = 6
y= 8
print('x > y is', x>y)
print('x < y is', x<y)
print('x == y is', x==y)
print('x != y is', x!=y)
print('x >= y is', x>=y)
print('x <= y is', x<=y)
```

Example 6– Logical Operators

```
# Logical Operators
x = False
y = True
print('x and y is', x and y)
print('x or y is', x or y)
print('not x is', not x)
```

Example 7– Bitwise Operators

```
# Bitwise Operators
# 7 in binary is 111
x=7
# 3 in binary is 11
y=3
print('x & y is', x & y)
print('x | y is', x | y)
print('x ^ y is ', x ^ y)
```

Example 8– Special Operators

```
# Special Operators
x1 = 6
y1 = 6
x2 = 'Hi'
y2 = 'Hi'
x3="CSE5APG"
y3="CSE5APG"
print(x1 is not y1)
print(x2 is y2)
print(x3 is y3)
```

Example 9– Membership Operators

```
# Membership Operators
x = 'Hello CSE5APG Week 1 Lab01'
print('H' in x)
print('hello' not in x)
print('ello' not in x)
print('ello' in x)
y = {1,2,3,4,5,6}
print(1 in y)
print('a' in y)
print('C' in x)
print(7 in y)
```

Example 10– run and inspect the following code- try to change the values/data

```
# Example of data combination
x = 4               # integer number
f = 3.1415926       # floating point number
name = "Python CSE5APG"     # string

print(x)
print(f)
print(name)

# string + space + string
combination = name + " " + name
print(combination)

# float + float
sum = f + f
print(sum)

# integer + integer
sum = x + x
print(sum)

# integer + float
sum = x + f
print(sum)
```