

การพัฒนาโปรแกรมประยุกต์และปัญญาประดิษฐ์ เพื่อการมองเห็นของเครื่องจักร Computer Programing and Artificial Intelligence in Machine Vision

1/4 – Start Python

- เริ่มต้นใช้งาน Python บน Jupyter Notebook Web App
- การเขียนโปรแกรมภาษา Python – การโปรแกรมอย่างสั้น
- คำถามท้ายบทเพื่อทดสอบความเข้าใจ

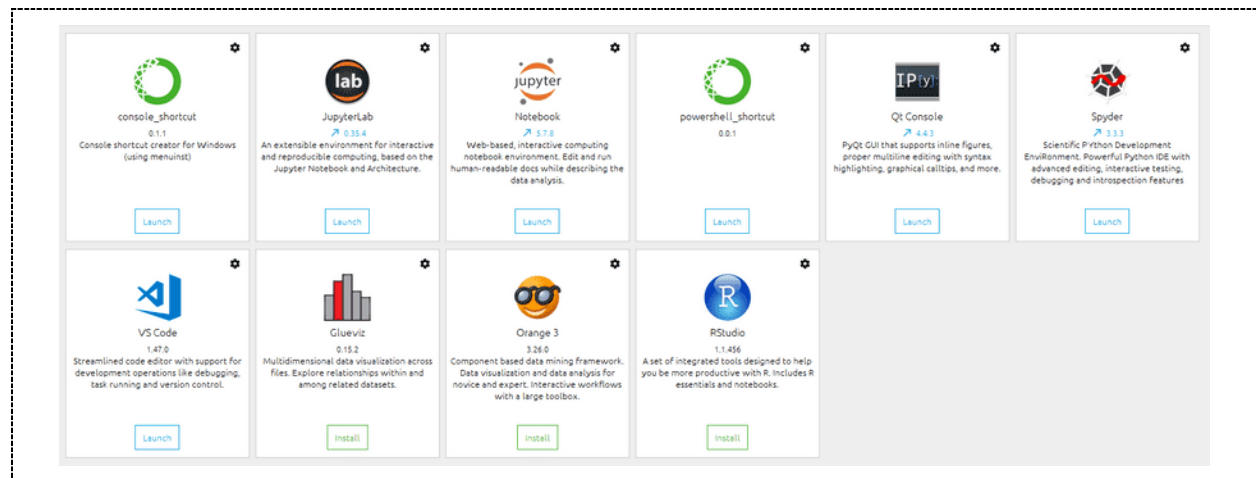
1/3 -- เริ่มต้นใช้งาน Python บน Jupyter Notebook Web App

1. การติดตั้งโปรแกรม Python ผ่าน Anaconda และ Jupyter

Anaconda คืออะไร

Anaconda เป็นเครื่องมือ หรือ open-source ที่จะช่วยให้เราพัฒนาโปรแกรมทางด้าน data science และ machine learning ได้ง่ายยิ่งขึ้น โดยจะมีการรวบรวมเครื่องมือต่างๆ ไว้ เช่น

- Code editor ที่ใช้สำหรับการเขียนโปรแกรม python ที่นิยมก็จะมี Spyder, VS Code
- Jupyter Notebook
- R studio สำหรับคนที่เขียนภาษา R



ภาพตัวอย่างเครื่องมือที่ Anaconda รวบรวมมาให้

การติดตั้ง mini-conda และตรวจสอบผลการติดตั้ง

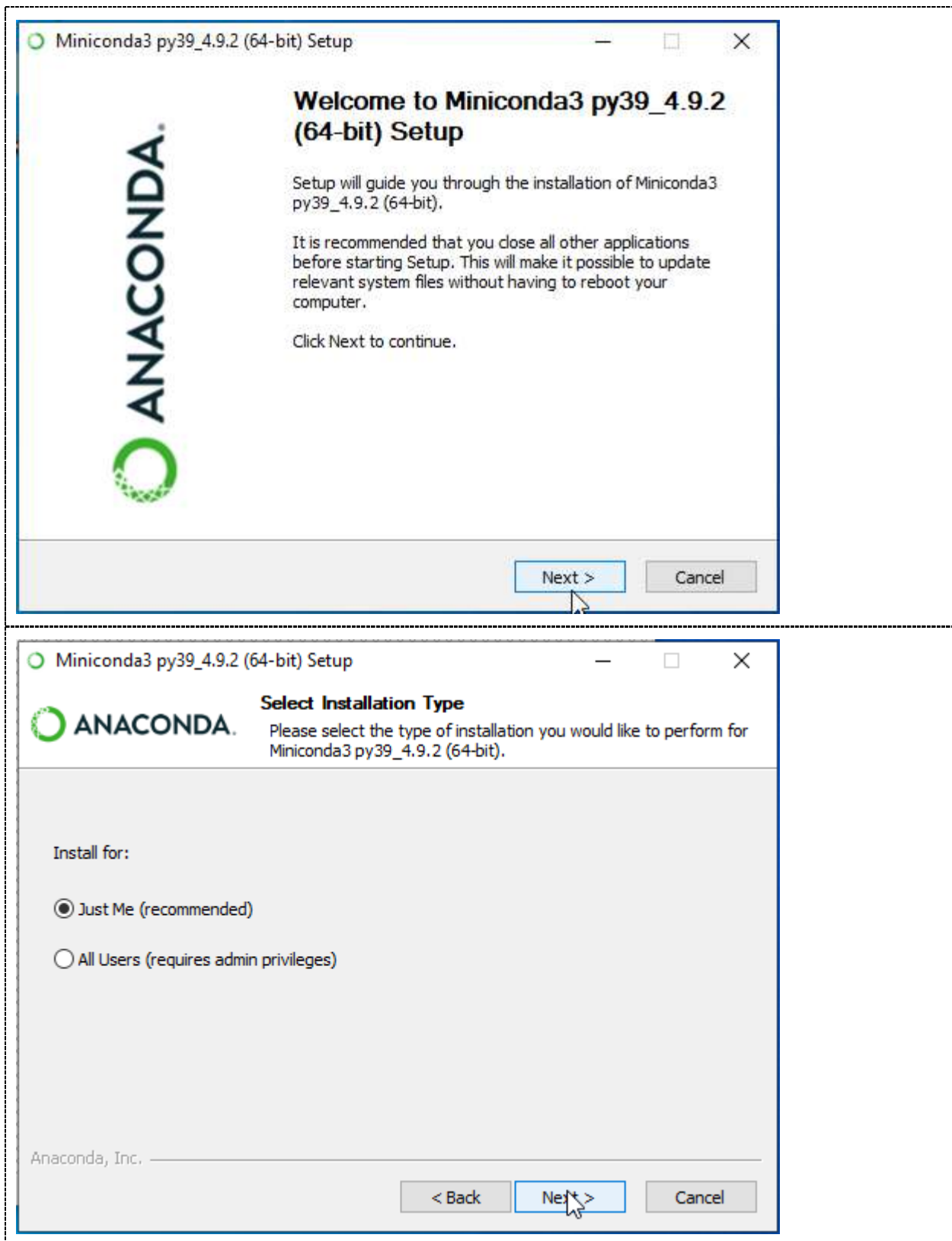
1. ดาวน์โหลด Anaconda จากเว็บนี้ anaconda จาก <https://docs.conda.io/en/latest/miniconda.html>
 - โดยให้เลือก Version 3.x (date 14-07-2021 version 3.9)
 - และให้เลือกขนาด bit ตาม OS Windows ที่เราลง

Miniconda

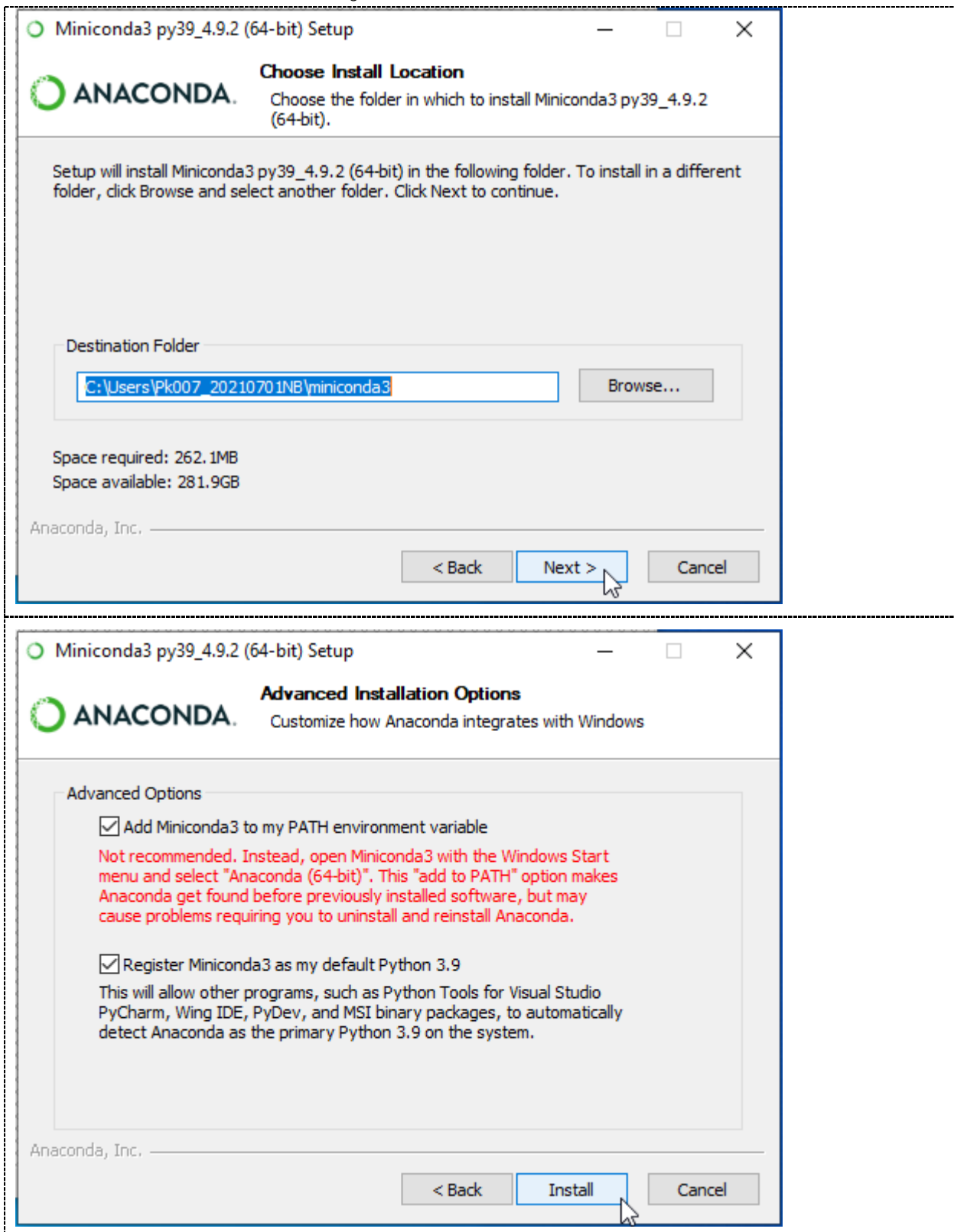
Windows installers

Python version	Name	Size	SHA256 hash
Python 3.9	Miniconda3 Windows 64-bit	57.7 MiB	c3a43d6bc4c4fa92454dbfa636ccb859a045d875df602b31ae71b9e0c3fec2b8
	Miniconda3 Windows 32-bit	54.9 MiB	5045fb9dc4485dbba21054262b7d1040a61a8739c1a56038ccb0258f233ad646

2. การติดตั้งโปรแกรม Mini Anaconda

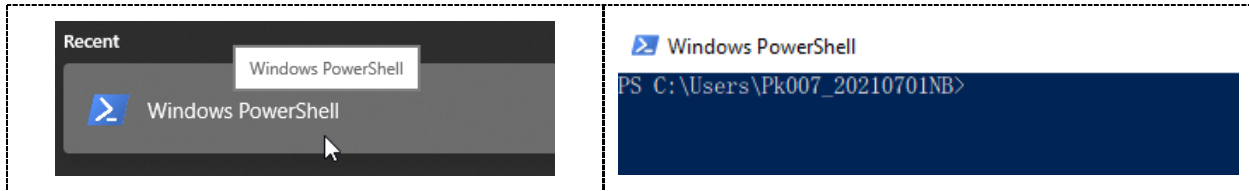


3. เลือก Folder ที่ติดตั้ง และที่สำคัญอย่าลืม **Add PATH**



4. Initialize conda

- เปิด Windows PowerShell ด้วยการพิมพ์ PowerShell ที่ Search
- ให้พิมพ์คำสั่ง **conda init**



- ปิด - เปิด Windows PowerShell ใหม่
- ถ้าหากเกิด Error ให้ตั้งค่า Policy ของ Windows PowerShell
<https://www.skconan.com/powershell-set-policy/>

Policy ของ Windows PowerShell

สำหรับใครที่ใช้ Virtualenv, Anaconda หรือโปรแกรมอื่น แล้วติดปัญหาเกี่ยวกับ execution policies ของ Windows PowerShell ให้ลองทำตามบทความนี้ดูนพครับ

1. เปิด Windows PowerShell แบบ Administrator โดยการกด Shift ค้างไว้แล้วคลิกขวาที่ Icon windows ตรง Taskbar
2. เมื่อเปิดได้แล้วให้พิมพ์คำสั่ง
Set-ExecutionPolicy RemoteSigned
3. หลังจากนั้นจะมีคำถาม โดยจะเลือกแบบ Yes หรือ Yes to All ก็ได้ โดยพิมพ์ Y หรือ A แล้วกด Enter
4. ลองเปิด Windows PowerShell แบบปกติใหม่อีกครั้ง
ถ้ายังไม่หาย Error ให้ลองใช้คำสั่งด้านล่างแทนครับ
Set-ExecutionPolicy AllSigned

- ถ้าไม่ error เราจะเห็นคำว่า (base) อยู่ข้างหน้า

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

Loading personal and system profiles took 871ms.
(base) PS C:\Users\Pk007_20210701NB>
```

5. พิมพ์คำสั่ง **conda -V** จะแสดง version ของ conda {Uppcase V}

```
(base) PS C:\Windows\system32> conda -V
conda 4.9.2
(base) PS C:\Windows\system32>
```

6. พิมพ์คำสั่ง **python.exe** เพื่อทดลองใช้ python

```
(base) PS C:\Users\Pk007_20210701NB> python
Python 3.9.1 (default, Dec 11 2020, 09:29:25) [MSC v.1916 64 bit (AMD64)]
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

7. ให้ลองพิมพ์ `print('Hello')` และกด Enter ดูครับ จะได้ผลลัพธ์ประมาณนี้

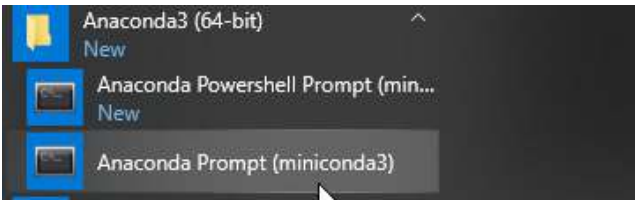
```
(base) PS C:\Users\Pk007_20210701NB> python
Python 3.9.1 (default, Dec 11 2020, 09:29:25) [MSC v.1916 64 bit (AMD64)]
Type "help", "copyright", "credits" or "license" for more information.
>>> print('Hello')
Hello
>>>
```

8. พิมพ์ **Ctrl + z** เพื่อออกจากโปรแกรม Python

9. การติดตั้งโปรแกรม Jupyter

- หลังจากทำการติดตั้งโปรแกรม Mini Anaconda เสร็จเรียบร้อยแล้ว ก็ทำการติดตั้งโปรแกรม jupyter ผ่านทางหน้าจอ Command Prompt ด้วยคำสั่ง → **conda install jupyter**

เรียก Anaconda Prompt



>> conda install jupyter

Anaconda Prompt (miniconda3) - conda install jupyter

```
(base) C:\Users\Pk007_20200917Bit64>conda install
CondaValueError: too few arguments, must supply command line package specs or --file

(base) C:\Users\Pk007_20200917Bit64>
(base) C:\Users\Pk007_20200917Bit64>conda install jupyter
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible solve.
Solving environment: failed with repodata from current_repodata.json, will retry with next rep
Collecting package metadata (repodata.json): done
Solving environment: /
```

Anaconda Prompt (miniconda3) - conda install jupyter

decorator-4.4.2	14 KB	#####
importlib-metadata-2	35 KB	#####
nbclient-0.5.1	58 KB	#####
packaging-20.8	36 KB	#####
python-gmp-6.1.0	689 KB	#####

10. การเรียกใช้งาน Jupyter

- ใช้งานด้วยคำสั่ง **jupyter notebook** โดยมีการเชื่อมต่อการทำงานกับเว็บเซิร์ฟเวอร์ การใช้งานโปรแกรม jupyter ผ่านทางเว็บเบราว์เซอร์
- See More >>

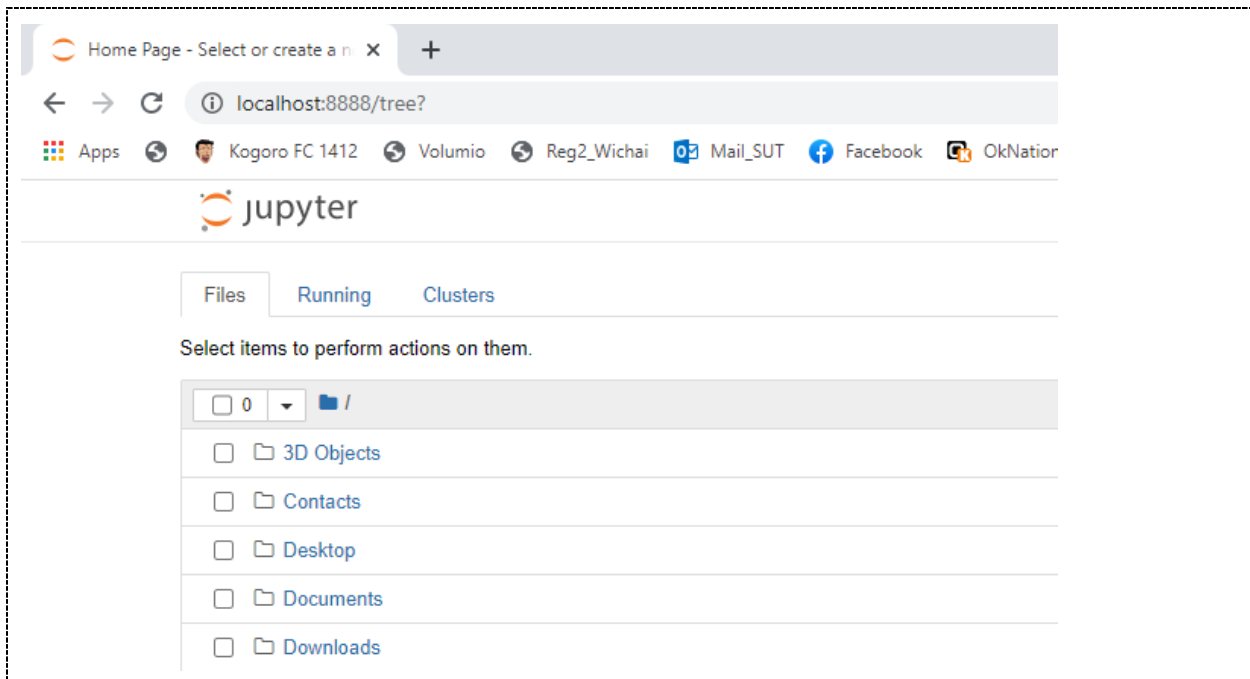
https://youtu.be/KKdItXMp09Q?list=PLoTScYm900GFvRk_MmZtovQXNIi36LUz

```

Anaconda Prompt (miniconda3)
(base) C:\Users\Pk007_20200917Bit64>jupyter notebook

Anaconda Prompt (miniconda3) - jupyter notebook
(base) C:\Users\Pk007_20200917Bit64>jupyter notebook
[I 22:38:14.258 NotebookApp] Writing notebook server cookie secret to C:\Users\Pk007_20200917Bit64\AppData\Roaming\jupyter\runtime\notebook_cookie_secret
[I 22:38:15.441 NotebookApp] Serving notebooks from local directory: C:\Users\Pk007_20200917Bit64\AppData\Local\jupyter-notebook\notebooks
[I 22:38:15.442 NotebookApp] Jupyter Notebook 6.1.5 is running at:
[I 22:38:15.442 NotebookApp] http://localhost:8888/?token=32d58eb1cf8d9bb65f07a0bd562894965c32765912d839ad
[I 22:38:15.442 NotebookApp] or http://127.0.0.1:8888/?token=32d58eb1cf8d9bb65f07a0bd562894965c32765912d839ad
[I 22:38:15.442 NotebookApp] Use Control-C to stop this server and shut down all kernels
[C 22:38:15.516 NotebookApp]

To access the notebook, open this file in a browser:
file:///C:/Users/Pk007_20200917Bit64/AppData/Roaming/jupyter/runtime/nbserver-12345/notebook.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=32d58eb1cf8d9bb65f07a0bd562894965c32765912d839ad
or http://127.0.0.1:8888/?token=32d58eb1cf8d9bb65f07a0bd562894965c32765912d839ad
  
```

11. การเรียกใช้งาน jupyter notebook ผ่าน Web browser เช่น chrome >> <http://localhost:8888/>

12. การกำหนดตำแหน่งแฟ้มข้อมูล

- เปิดแฟ้มข้อมูลที่ต้องการ จากนั้นกดปุ่ม Shift + Right Click ตรงที่ว่างของแฟ้มข้อมูล เลือกเมนู Open PowerShell window here
- ใช้คำสั่ง jupyter notebook เพื่อเปิดโปรแกรม jupyter ระหว่างใช้งานโปรแกรม ต้องเปิด Windows PowerShell ทิ้งไว้เพื่อการเชื่อมต่อการทำงานของระบบ

```

Anaconda Prompt (miniconda3) - jupyter notebook

(base) C:\Users\Pk007_20200917Bit64>jupyter notebook
[I 09:24:07.699 NotebookApp] Serving notebooks from local directory: C:\U
[I 09:24:07.699 NotebookApp] Jupyter Notebook 6.1.5 is running at:
[I 09:24:07.699 NotebookApp] http://localhost:8888/?token=07dbf07c5a217b9
[I 09:24:07.699 NotebookApp] or http://127.0.0.1:8888/?token=07dbf07c5a2
[I 09:24:07.699 NotebookApp] Use Control-C to stop this server and shut d

```

- การรันชุดคำสั่งในแต่ละเซลล์ ทำได้โดย กดปุ่ม Ctrl + Enter
- การรันพร้อมกันทุกเซลล์ให้ใช้เมนู Cell > Run All

1. พื้นที่สำหรับเขียนคำสั่งต่าง ๆ

2. ปุ่มคำสั่งที่ใช้ในการรันโปรแกรม

3. ปุ่มคำสั่งที่ใช้ในการบันทึกชุดคำสั่ง

4. ปุ่มคำสั่งที่ใช้ในการเพิ่มพื้นที่สำหรับเขียนคำสั่ง

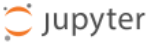
5. ปุ่มคำสั่งที่ใช้ในการลบพื้นที่สำหรับเขียนคำสั่ง

2. โปรแกรม Python - การดำเนินงานกับตัวแปร

1. ตัวแปร และการดำเนินการกับตัวแปร

- Firstname = 'Wichai' ชื่อตัวแปร Firstname ที่มีค่าตั้งต้นเป็นตัวอักษรเท่ากับ 'Wichai'
- age = 21 ชื่อตัวแปร age ที่มีค่าตั้งต้นเป็นเลขจำนวนเต็มมีค่าเท่ากับ 21
- Width = 2.75 ชื่อตัวแปร Width ที่มีค่าตั้งต้นเป็นเลขทศนิยมมีค่าเท่ากับ 2.75
- print (Firstname) แสดงค่าในตัวแปร Firstname

New → Python3

 Quit Logout

Files Running Clusters

Select items to perform actions on them. Upload New ↺











☐ 0 / Desktop / Week05Run Name Last Modified File size

.. seconds ago

The notebook list is empty.

Edit











File Edit View Insert Cell Kernel Widgets Help

         Code 

```
In [2]: Firstname = 'Wichai'
        age = 21
        Width = 2.75
        print (Firstname)
```

Run

File Edit View Insert Cell Kernel Widgets Help

         Code 

```
In [2]: Firstname = 'Wichai'
        age = 21
        Width = 2.75
        print (Firstname)

        Wichai

In [ ]:
```


2. การแจ้งเตือน การเฝ้าที่เกิตความผิดปกติของชุดคำสั่ง

```
In [3]: Firstname = 'Wichai'
age = 21x
Width = 2.75
print (Firstname)

File "<ipython-input-3-793a83a7366a>", line 2
    age = 21x
          ^
SyntaxError: invalid syntax
```

3. การดำเนินการกับตัวแปร

- Height = 3 สำหรับตัวแปร Height ที่มีค่าตั้งต้นเป็นเลขจำนวนเต็มเท่ากับ 3
- Area = Width * Height การดำเนินการทางคณิตศาสตร์โดย การนำตัวแปร Width มาคูณกับตัวแปร Height ผลลัพธ์ที่ได้คือตัวแปร Area มีค่าเท่า 8.25

```
In [6]: Width = 2.75
Height = 3
Area = Width * Height
print(Area)

8.25
```

- Name = 'Wichai' สำหรับตัวแปร Name ที่มีค่าตั้งต้นเป็นตัวอักษรเท่ากับ "Wichai"
- Char = Name [0] การเลือกตัวอักษรจากตัวแปร Name จากตำแหน่งแรก กรณีนี้จะเริ่มที่ค่า 0 ผลลัพธ์ที่ได้คือตัวแปร Char มีค่าเท่า 'W'
- Length = len(Name) การนับจำนวนตัวอักษรด้วยฟังก์ชัน len() ที่อยู่ในตัวแปร Name ผลลัพธ์ที่ได้คือตัวแปร Length มีค่าเท่า 6

```
In [9]: Name = 'Wichai'
Char = Name [0]
Length = len(Name)
print(Char)
print(Length)

W
6
```

3. โปรแกรม Python - การแสดงผล และการรับค่าข้อมูลทางแป้นพิมพ์

กิจกรรม 1: จงเขียนโปรแกรมหาค่า BMI โดยมีสูตรดังต่อไปนี้ $BMI = \frac{Weight(kg)}{[Height(m)]^2}$

h = 165

w = 59

bmi = ?

1.1 การแสดงผล และการรับค่าข้อมูลทางแป้นพิมพ์

- การแสดงผลด้วยคำสั่ง print() โดยมีการระบุข้อความ หรือข้อมูลที่ต้องการแสดงผล
- ตัวอย่างเช่น print("การเขียนโปรแกรมภาษา Python")

```
In [10]: print("การเขียนโปรแกรมภาษา Python")
          การเขียนโปรแกรมภาษา Python
```

1.2 การรับค่าข้อมูลทางแป้นพิมพ์

- ชุดคำสั่งที่ใช้ในการรับข้อมูล คือ ฟังก์ชัน input() โดยสามารถระบุข้อความ 'Please input your Weight' เพื่ออธิบายถึงข้อมูลที่ใช้ควรที่จะกำหนดค่าข้อมูลให้กับตัวแปร Weight

```
In [*]: Weight = input("Please input your Weight")
          Please input your Weight 
```

1.3 การนำผลลัพธ์ที่ได้ออกมาแสดงผล

```
In [13]: Weight = 60
          Height = 1.7
          BMI = Weight / (Height**2)
          print("Body Mass Index = ",BMI)

          Body Mass Index = 20.761245674740486
```

1.4 ตัวอย่าง : การรับค่าข้อมูลทางแป้นพิมพ์

```
In [16]: Weight = input("น้ำหนัก [kg]: ")
Height = input("ส่วนสูง [m]: ")
BMI = float(Weight) / (float(Height) * float(Height))
print("Body Mass Index = %.3f" %BMI)
```

```
น้ำหนัก [kg]: 70
ส่วนสูง [m]: 1.7
Body Mass Index = 24.221
```

กิจกรรม 2: จงเขียนโปรแกรมสำหรับคำนวณหาค่าเงินต้นก่อนคำนวณ VAT 7%

- โดยมีผลลัพธ์ดังตัวอย่างต่อไปนี้

ยอดรวม : 107

100.0

ยอดรวม : 53.5

50.0

4. โปรแกรม Python - การควบคุมการดำเนินการของโปรแกรม

1. การตรวจสอบเงื่อนไข if

- กรณีที่ต้องการตัดสินใจว่าจะดำเนินการ หรือไม่ดำเนินการชุดคำสั่งนั้น ๆ
- คำสั่งที่ต้องการให้อยู่ภายใต้การทำงานของคำสั่งก่อนหน้าจะต้องมีการเว้นระยะด้านหน้าเสมอ

```
In [17]: password = input("Password : ")
         if password == '1234':
             print('Your Password id Correct')

Password : 1234
Your Password id Correct
```

2. การตรวจสอบเงื่อนไข if else

- กรณีที่ต้องการตัดสินใจว่าจะดำเนินการ หรือไม่ดำเนินการชุดคำสั่งนั้นๆ และหากต้องการให้มีการตอบสนองต่อเหตุการณ์ที่ไม่ถูกต้อง

```
In [18]: password = input("Password : ")
         if password == '1234':
             print('Your Password id Correct')
         else:
             print('Your Password id incorrect')

Password : 1345
Your Password id incorrect
```

3. การตรวจสอบหลายเงื่อนไข

- กรณีที่ต้องการตัดสินใจว่าจะดำเนินการ หรือไม่ดำเนินการจากหลายเงื่อนไข สามารถเชื่อมโยงเงื่อนไขด้วย and หรือ or

```
In [19]: user = input("ชื่อ : ")
         password = input("รหัสผ่าน : ")
         if user == 'wichai' and password == '1234':
             print('การยืนยันผู้ใช้งานถูกต้อง')
         else:
             print('การยืนยันผู้ใช้งานไม่ถูกต้อง')

ชื่อ : wichai
รหัสผ่าน : 1234
การยืนยันผู้ใช้งานถูกต้อง
```

กิจกรรม 3: จงเขียนโปรแกรมสำหรับคำนวณภาษีเงินได้ โดยมีรายละเอียดดังต่อไปนี้

เงินได้สุทธิ (บาท)	อัตราภาษี
150,000 บาทแรก	ยกเว้น
150,001 - 300,000	5
300,001 บาทขึ้นไป	10

ยอดเงิน: 1000

ยอดเงิน: 200000

ยอดเงิน: 400000

0.0

2500.0

17500.0

Capture Code โปรแกรม		
ผลการทำงาน เมื่อยอดเงิน: 1,000+รหัสนักศึกษา 3 ตัวท้าย	ผลการทำงาน เมื่อยอดเงิน: 200,000+รหัสนักศึกษา 4 ตัวท้าย	ผลการทำงาน เมื่อยอดเงิน: 400,000+รหัสนักศึกษา 5 ตัวท้าย

4. การดำเนินการซ้ำ

- การทำงานของคำสั่งที่มีความต่อเนื่องกันหลาย ๆ ครั้ง เช่น การพิมพ์ข้อความ การรับค่าข้อมูลทางแป้นพิมพ์ การคำนวณค่าผลรวมของค่าตัวเลข

```
In [7]: ➤ Cars = ['Honda', 'Toyota', 'Mazda', 'Nissan']  
print(Cars[0])  
print(Cars[1])  
print(Cars[2])  
print(Cars[3])  
  
Honda  
Toyota  
Mazda  
Nissan
```

5. การวนซ้ำด้วยคำสั่ง for

- การวนซ้ำด้วยคำสั่ง for จะใช้สำหรับกรณีที่สามารถควบคุมจำนวนการดำเนินการซ้ำ

```
In [6]: ➤ Cars = ['Honda', 'Toyota', 'Mazda', 'Nissan']  
  
for Car in Cars:  
    print(Car)  
  
Honda  
Toyota  
Mazda  
Nissan
```

6. ตัวอย่างการประยุกต์ การวนซ้ำด้วยคำสั่ง for

- ตัวอย่างการตรวจสอบรหัสผ่าน ถ้าเกิน 3 ครั้งให้ติดต่อธนาคาร

In [6]:

```
for i in range(1,4):
    password = input('Password : ')
    if password == '2134' :
        print('Start your transection ')
        break
if i == 3 :
    print('Please contact the Bank. ')
```

```
Password : 111
Password : 111
Password : 111
Please contact the Bank.
```

7. การวนซ้ำด้วยคำสั่ง while

- คำสั่ง while จะช่วยจัดการปัญหาจากกรณีที่มีการดำเนินการซ้ำที่เราไม่รู้จำนวนครั้ง หรือ ช่วงของการดำเนินการซ้ำ ด้วยการตรวจสอบเงื่อนไข

In [6]:

```
Score = 0
while Score <= 0 or Score > 100 :
    Score = int ( input('Input your score : ') )

if Score >= 50 :
    print('Pass')
else:
    print('Not Pass')
```

```
Input your score : 120
Input your score : 0
Input your score : -5
Input your score : 65
Pass
```

กิจกรรม 4: คำนวณผลรวมดอกเบี้ย โดยกำหนด เงินต้น ดอกเบี้ย และจำนวนปี โดยมีตัวอย่างผลลัพธ์ดังต่อไปนี้

Capture Code โปรแกรม	
เงินต้น: 100 ดอกเบี้ย: 3 จำนวนปี: 2	เงินต้น: 200 ดอกเบี้ย: 3 จำนวนปี: 10
106.09	268.7832758688244

5. โปรแกรม Python - การเรียกใช้ส่วนขยาย

1. การติดตั้งส่วนขยาย และยกเลิกการติดตั้ง
 - เมื่อผู้พัฒนาจะต้องทำการดำเนินการที่มีความซับซ้อนเพิ่มขึ้น การโปรแกรมก็จำเป็นที่จะต้องมีการเรียกใช้คำสั่งพิเศษ ซึ่งอยู่ในส่วนขยายต่าง ๆ โดยใช้คำสั่ง
 - การติดตั้งส่วนขยายทำได้ด้วยคำสั่ง → `conda install extension-name`
 - การยกเลิกการติดตั้งส่วนขยายทำได้ด้วยคำสั่ง → `conda uninstall extension-name`
2. ตัวอย่าง การติดตั้งและยกเลิกการติดตั้งส่วนขยาย
 - บรรทัด `In [*]` หมายถึงกำลังทำงานให้รอ อาจบังคับให้หยุดทำงานด้วย `Ctrl+C`

```
In [4]: conda install pandas
```

```
Collecting package metadata (current_repodata.json): ...working... done
Solving environment: ...working... done

Note: you may need to restart the kernel to use updated packages.
# All requested packages already installed.
```

```
In [5]: conda uninstall pandas
```

```
Collecting package metadata (repodata.json): ...working... done
Note: you may need to restart the kernel to use updated packages.

Solving environment: ...working...
Warning: 2 possible package resolutions (only showing differing packages):
  - defaults/noarch::parso-0.8.1-pyhd3eb1b0_0, defaults/win-64::jedi-0.17.0-py38_0
  - defaults/noarch::parso-0.7.0-py_0, defaults/win-64::jedi-0.17.2-py38haa95532_1done

## Package Plan ##

environment location: C:\Users\Pk007_20200917Bit64\miniconda3

removed specs:
  - pandas
```

3. การประกาศเพื่อเรียกใช้ส่วนขยาย

- การประกาศเพื่อเรียกใช้ส่วนขยาย → `import extension-name`
- ตัวอย่างการเรียกใช้ส่วนขยาย `datetime` เพื่อการทำงานกับข้อมูลวันและเวลา

```
In [7]: import datetime
```

```
In [10]: x = datetime.datetime.now()
print(x)
print("Date = ",x.strftime('%d/%m/%y'))
print("Time = ",x.strftime('%H/%M/%S'))
```

```
2020-12-25 10:35:30.186688
```

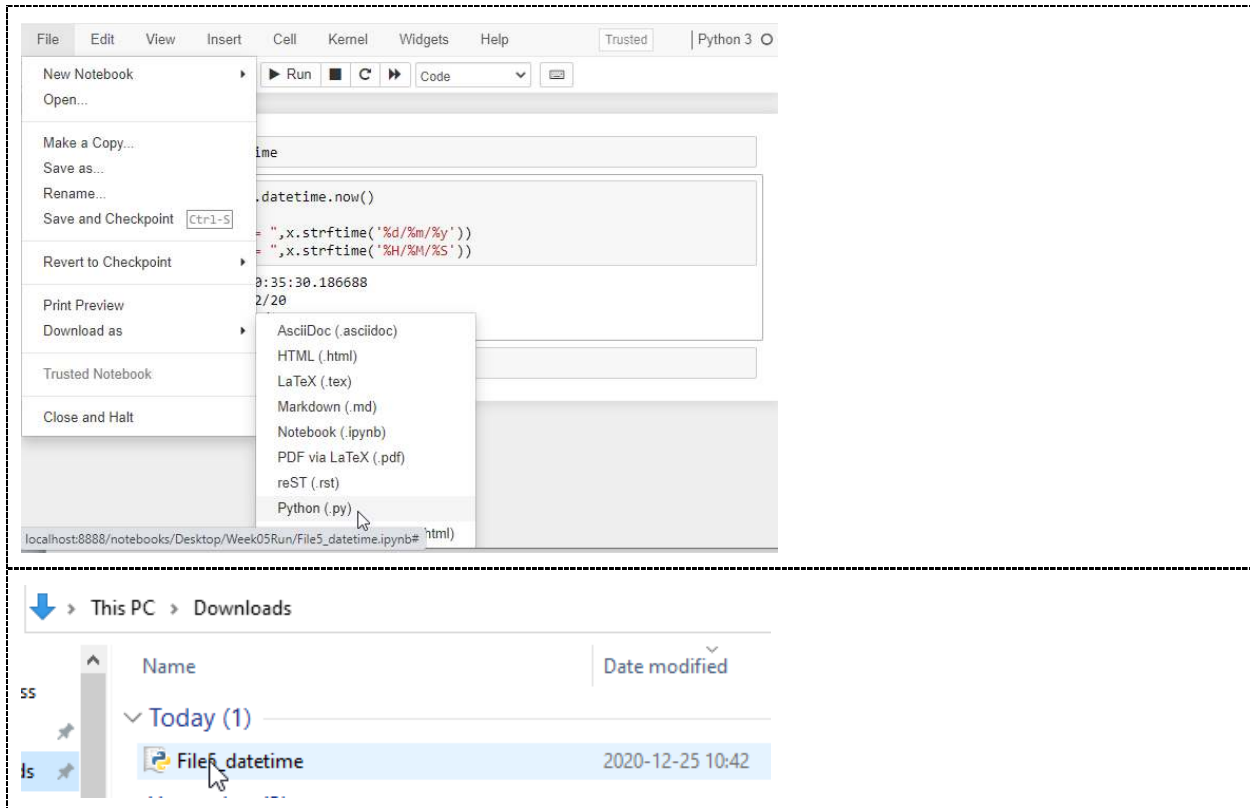
```
Date = 25/12/20
```

```
Time = 10/35/30
```

6. โปรแกรม Python - การนำไปใช้งาน

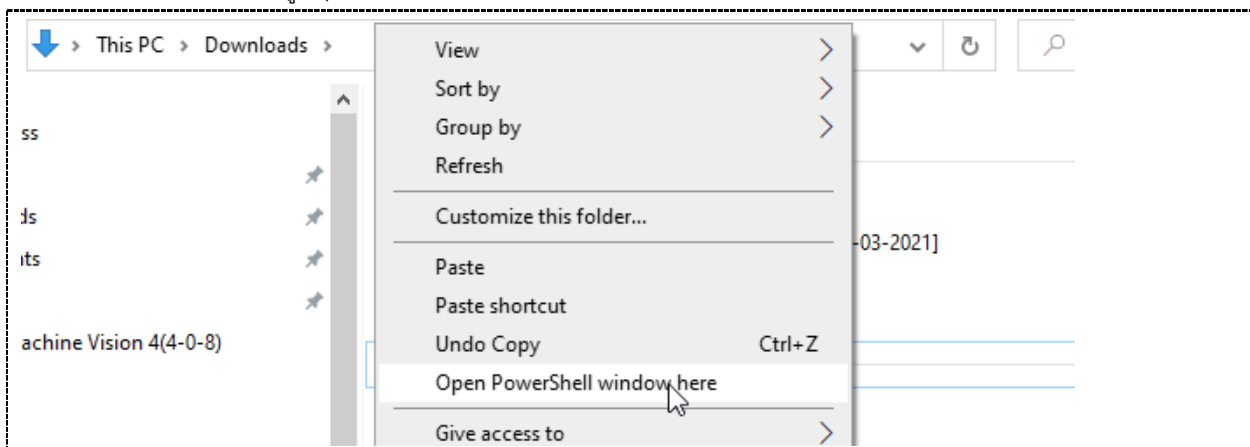
1. การดาวน์โหลดเพื่อนำไปใช้งาน

- File → Download as → Python(.py)
- ไฟล์ที่ได้จะจัดเก็บอยู่ในแฟ้ม Downloads



2. การเรียกใช้งาน

- กดปุ่ม Shift + Right Click ตรงที่ว่างของแฟ้มข้อมูล
- เลือกเมนู Open PowerShell window here



3. การเรียกใช้งาน ด้วยคำสั่ง python

- C:\Users\bezr\Downloads> **python filename.py**



The screenshot shows a Windows Command Prompt window titled "Command Prompt". The command prompt is at the directory C:\Users\Pk007_20200917Bit64\Downloads. The user has entered the command `python File5_datetime.py`. The output of the script is displayed on the next three lines: `2020-12-25 10:54:36.213000`, `('Date = ', '25/12/20')`, and `('Time = ', '10/54/36')`. The prompt then returns to the command line, showing `C:\Users\Pk007_20200917Bit64\Downloads>_`.

```
Command Prompt
C:\Users\Pk007_20200917Bit64\Downloads>python File5_datetime.py
2020-12-25 10:54:36.213000
('Date = ', '25/12/20')
('Time = ', '10/54/36')
C:\Users\Pk007_20200917Bit64\Downloads>_
```

2/3 -- การเขียนโปรแกรมภาษา Python – การโปรแกรมอย่างสั้น

<https://www.facebook.com/learntodeveloper>

1. จากโปรแกรม Ex1 ถึง Ex10 ให้ทดสอบโปรแกรมบน Jupyter แล้วอธิบายการทำงาน ผลการทำงานของโปรแกรม

<pre> 1 x,y = 30, "TEST" 2 print (x,y) 3 x,y = y,x 4 print (x,y) 5 </pre> <p>30 TEST TEST 30</p>	<p><u>1. อธิบายการทำงานของโปรแกรม</u></p> <ul style="list-style-type: none"> • การประกาศ ค่าตัวแปร X=30 และ Y="TEST" • การสลับค่าของตัวแปร



ex.1





เราสามารถสลับตัวแปรได้อย่างง่าย



ex.2



```
10 Essential Python .py •  
10 Essential Python .py > ...  
1 myname = "ABCDEFGFGHI"  
2 print("Reverse is", myname[::-1])  
3  
4 # ผลลัพธ์  
5 # Reverse is IHGFGE DCBA
```

เราสามารถแสดงข้อความกลับหลัง
โดยใช้แค่ code ดังภาพ 1 บรรทัด
โดยไม่ต้อง loop ให้เสียเวลา
รูปแบบ `[::-1]` คือการ Reverse



ex.3



```
10 Essential Python .py x
10 Essential Python .py > ...
1 mylist = ["HI", "MY", "NAME", "IS", "BILL"]
2 print(" ".join(mylist))
3
4 # ผลลัพธ์
5 # HI MY NAME IS BILL
```

**เราสามารถดึงข้อความใน List
มาแสดงรวมกันโดยใช้การ Join
ไม่ต้องเขียน Loop ให้เสียเวลาเหมือนเคย**



ex.4



```
10 Essential Python .py •  
10 Essential Python .py > ...  
1 mynum = 5  
2 result = 2 < mynum == 5  
3 print(result)  
4 result = 6 > mynum < 2  
5 print(result)  
6 # ผลลัพธ์  
7 # True  
8 # False
```

เราสามารถเปรียบเทียบ operator

ได้ใน 1 บรรทัด

เช่นบรรทัดที่ 2 คือ

2 น้อยกว่า mynum และ mynum ต้องเท่ากับ 5

จะได้ผลลัพธ์ = True [เป็นจริง]



python™

ex.5



python™

```
10 Essential Python .py •
10 Essential Python .py
1 import os
2 import pymysql
3
4 print(os)
5 print(pymysql)
6
7 # ผลลัพธ์
8 #<module 'os' from
9 # 'C:\\Users\\nara-\\AppData\\Local\\
10 # Programs\\Python\\Python36-32\\lib\\os.py'>
11
12 #<module 'pymysql' from
13 # 'C:\\Users\\nara-\\AppData\\Local\\Programs\\Python
14 # \\Python36-32\\lib\\site-packages\\pymysql\\__init__.py'>
```

**เราสามารถดูที่อยู่ไฟล์ ของ lib นั้นๆได้
แค่พิมพ์ดั่งภาพ**



ex.6



```
10 Essential Python .py •  
10 Essential Python .py > ...  
1 def mystring():  
2     return "str1","str2","str3"  
3  
4 myname1,myname2,myname3 = mystring()  
5 print(f"{myname1},{myname2},{myname3}")  
6  
7 # ผลลัพธ์  
8 # str1,str2,str3
```

**1 function สามารถ
รีเทินค่าออกมาได้มากกว่า 1 ค่านะ**



ex.7



```
10 Essential Python .py •
10 Essential Python .py > ...
1  mylist = [
2      1,2,3,4,
3      2,2,3,1,
4      4,4,4,5,
5      5,5,5,5
6  ]
7  print(max(set(mylist), key = mylist.count))
8
9  # ผลลัพธ์
10 # 5
```

เราสามารถหาตัวเลขที่อยู่ในลิสต์
ว่าตัวเลขไหนมีความถี่เยอะสุด
ดังภาพจะมีเลข 5 ด้วยกัน 5 ตัวเลข



Ex.8



```
10 Essential Python .py •
10 Essential Python .py > ...
1 import sys
2 mystring = "hello world"
3 print(sys.getsizeof(mystring))
4
5 # ผลลัพธ์
6 # 36
```

เราสามารถดูได้ว่าตัวแปรที่ใช้
ใช้ memory ไปเท่าไหร่
อ่านข้อมูลเพิ่มได้ที่



Ex.9



```
10 Essential Python .py •
10 Essential Python .py > ...
1 mystring = "HI "
2 print(mystring * 10)
3
4 # ผลลัพธ์
5 # HI HI HI HI HI HI HI HI HI HI
```

เราสามารถป้อนค่าออกมาดังภาพ
โดยไม่ต้อง loop ให้เสียเวลา



ex.10



```
10 Essential Python .py •  
10 Essential Python .py > ...  
1 mystrig = "Messi"  
2 Result = "es" in mystrig  
3 print(Result)  
4  
5 # ผลลัพธ์  
6 # True
```

เราสามารถเช็คค่าบางค่าในตัวแปรได้
แค่ 1 บรรทัดดังภาพ

การพัฒนาโปรแกรมประยุกต์และปัญญาประดิษฐ์ เพื่อการมองเห็นของเครื่องจักร
Computer Programing and Artificial Intelligence in Machine Vision

ชื่อ-สกุล :

3/3 -- คำถามท้ายบทเพื่อทดสอบความเข้าใจ

Quiz_101 – ทดสอบการทำงานของโปรแกรมและ Capture Jupyter โค้ดโปรแกรม

กิจกรรม 1: BMI ใช้น้ำหนัก ส่วนสูงของตัวเอง

```
In [16]: Weight = input("น้ำหนัก [kg]: ")
Height = input("ส่วนสูง [m]: ")
BMI = float(Weight) / (float(Height) * float(Height))
print("Body Mass Index = %.3f" %BMI)
```

น้ำหนัก [kg]: 70
ส่วนสูง [m]: 1.7
Body Mass Index = 24.221

กิจกรรม 2: คำนวณค่าเงินต้นก่อนคำนวณ VAT 7% ทดสอบที่ราคาสินค้า = 4,321.00 บาท

Capture Code โปรแกรม และผลการทำงานของโปรแกรม

กิจกรรม 3: จงเขียนโปรแกรมสำหรับคำนวณภาษีเงินได้

Capture Code โปรแกรม

ผลการทำงาน เมื่อยอดเงิน: 1,234.00	ผลการทำงาน เมื่อยอดเงิน: 201,234.00	ผลการทำงาน เมื่อยอดเงิน: 456,789.00

กิจกรรม 4: คำนวณผลรวมดอกเบี้ยย

Capture Code โปรแกรม

เงินต้น: 1,234.00 ดอกเบี้ย: 3 จำนวนปี: 2 ???.??	เงินต้น: 1,234.00 ดอกเบี้ย: 3 จำนวนปี: 10 ???.??
--	---

Quiz_102 – ทดสอบการทำงานและอธิบายการทำงานของโปรแกรม

<p>In [3]:</p> <pre> 1 x,y = 1929, "WICHAI" 2 print (x,y) 3 x,y = y,x 4 print (x,y) 5 1929 WICHAI WICHAI 1929 </pre>	<p><u>1. อธิบายการทำงานของโปรแกรม</u></p> <ul style="list-style-type: none"> • การประกาศ ค่าตัวแปร X=1929 และ Y="WICHAI" • การสลับค่าของตัวแปร
	<p><u>2. อธิบายการทำงานของโปรแกรม</u></p>
	<p><u>3. อธิบายการทำงานของโปรแกรม</u></p>
	<p><u>4. อธิบายการทำงานของโปรแกรม</u></p>
	<p><u>5. อธิบายการทำงานของโปรแกรม</u></p>
	<p><u>6. อธิบายการทำงานของโปรแกรม</u></p>
	<p><u>7. อธิบายการทำงานของโปรแกรม</u></p>
	<p><u>8. อธิบายการทำงานของโปรแกรม</u></p>
	<p><u>9. อธิบายการทำงานของโปรแกรม</u></p>
	<p><u>10. อธิบายการทำงานของโปรแกรม</u></p>