Introduction to Python

Introduction Python & Tools

ทำไมต้องใช้ Python สำหรับ Data Science?

- **Easy to learn:** ไวยากรณ์อ่านง่าย เหมือนภาษาอังกฤษ
- **Readable:** โค้ดมีความสะอาด เป็นระเบียบ
- **Scalable:** รองรับการขยายระบบงานได้ดี
- **Libraries:** มีไลบรารีสำหรับ Data Science จำนวนมาก (Numpy, Pandas, etc.)
- **Community:** มีผู้ใช้งานทั่วโลก ช่วยเหลือและแบ่งปันความรู้

Python Libraries ที่สำคัญ

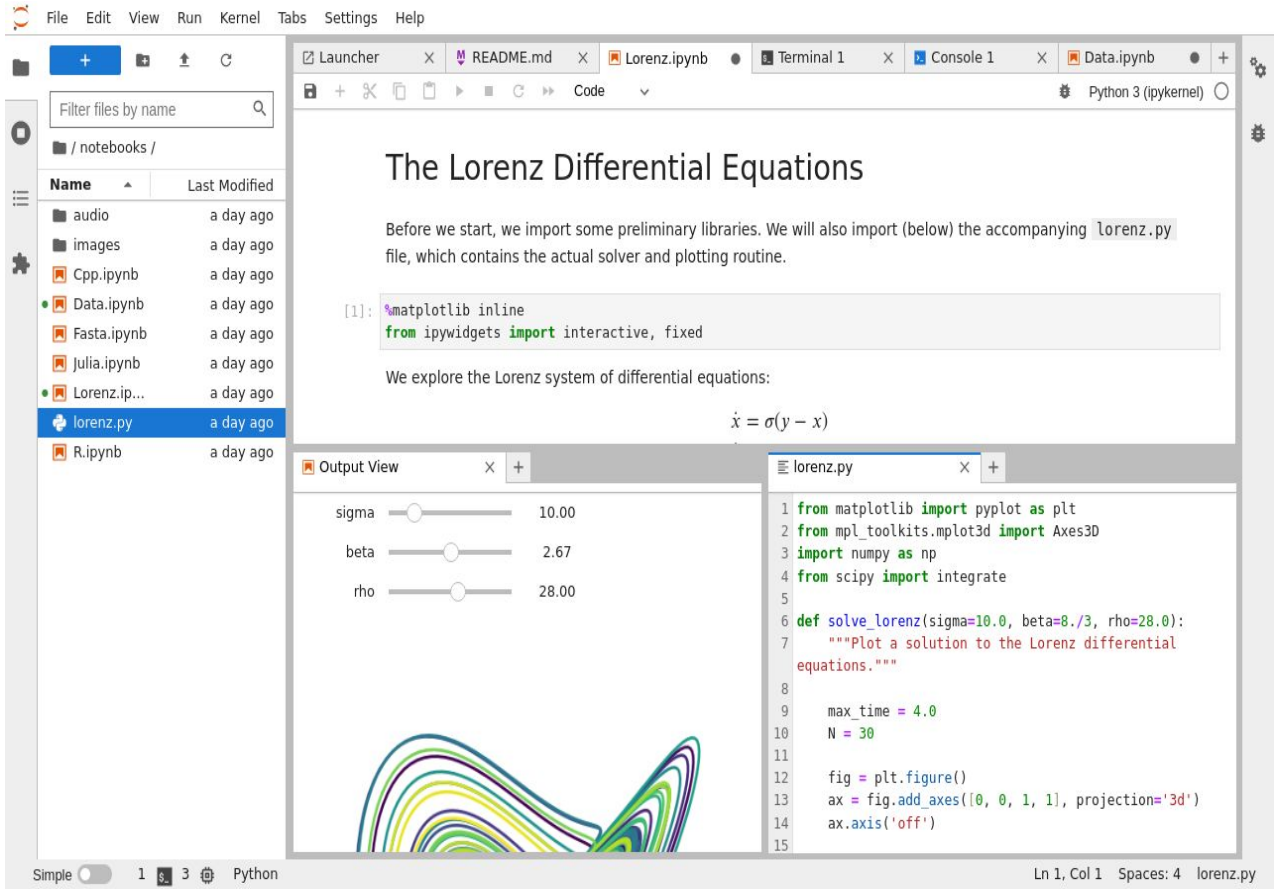
Library	หน้าที่หลัก (Usage)
NumPy, SciPy	การคำนวณทางวิทยาศาสตร์และคณิตศาสตร์ (Scientific Computing)
Pandas	จัดการและวิเคราะห์ข้อมูลตาราง (Data Manipulation)
Matplotlib, Seaborn	สร้างกราฟและนำเสนอข้อมูล (Visualization)
Scikit-learn	ทำ Machine Learning (Classification, Regression, Clustering)
TensorFlow, Keras	ทำ Deep Learning และ AI

Jupyter Notebook

เครื่องมือหลักของ Data Scientist

Jupyter Notebook เป็น Interactive Computing Environment ที่นิยมที่สุด เพราะ:

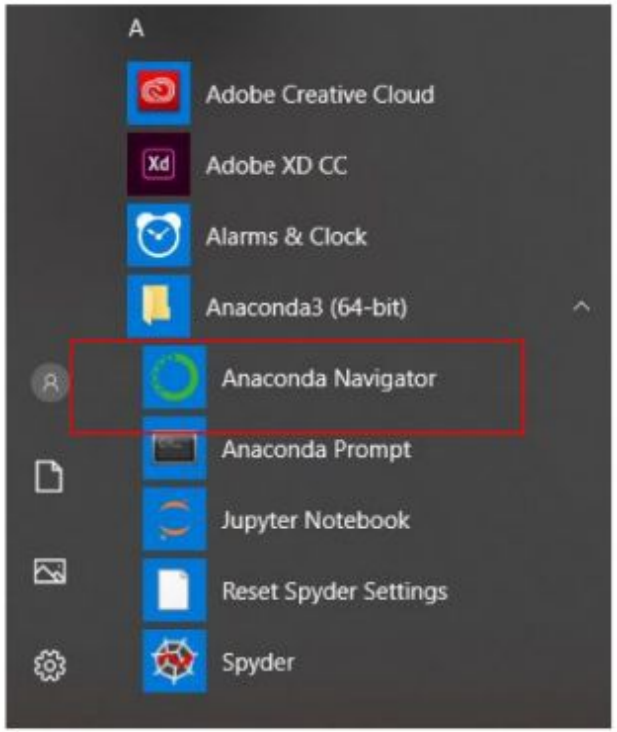
- **Interactive:** รันโค้ดทีละส่วน (Cell) และเห็นผลลัพธ์ทันที
- **Documentation:** เขียนคำอธิบายแทรกโค้ดได้ด้วย Markdown
- **Visualization:** แสดงกราฟรูปภาพในหน้าเดียวกันได้เลย
- **Easy to Share:** บันทึกเป็น HTML หรือ PDF เพื่อแชร์ได้ง่าย

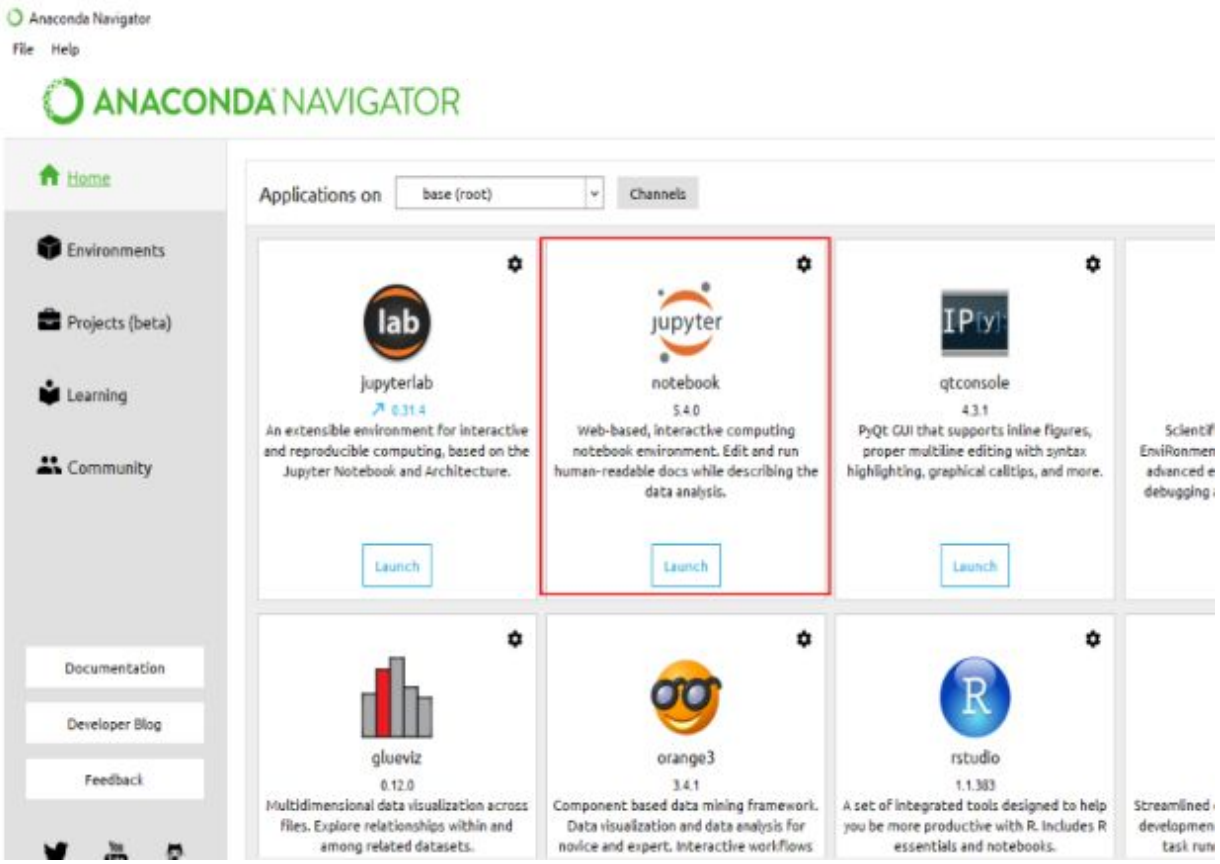


Anaconda Python environment manager

Download: https://www.anaconda.com/download/

Anaconda Python environment manager





Anaconda Python environment manager

Anaconda Python environment manager

องค์ประกอบของ Jupyter notebook

1. Cell สามารถใส่ code หรือ Markdown หรือ hyper link

- Code –ใส่ code
- Markdown –ใส่ข้อความหรือบรรยายสมการ

2. ในการเขียน python อาจจะใช้ตัว IDE คือ pycharm เหมาะสำหรับการสร้าง class หรือ package ขึ้นมาใหม่ ตัว ide ก็จะเหมาะกว่า ตัว IDE จะเก็บแค่ code แต่ไม่เก็บตัวผลลัพธ์ เราต้อง เขียนโปรแกรมไปเรื่อยๆ มันจะไม่ได้ interactive เหมือนตัว notebook

Markdown

```
# Heading 1
## Heading 2
```

Heading 1

Heading 2

##Comment

```
## plot graph
import matplotlib.pyplot as plt
x = 2,3,4,5
y = 1,2,3,4
```

Python Basic Data Types

int

จำนวนเต็ม

```
x = 10
```

float

ทศนิยม

```
y = 3.14
```

str

ข้อความ (String)

```
s = "Data"
```

bool

ค่าความจริง

```
ok = True
```

สามารถตรวจสอบชนิดข้อมูลได้ด้วยคำสั่ง type(variable)

Python Data Types

ชื่อย่อ	ความหมาย	ตัวอย่าง
int	จำนวนเต็ม	12345
float	จำนวนจริง	123.45
complex	จำนวนเชิงซ้อน	123+45j
bool	บูล	True
str	สายอักขระ	'12345'
list	ลิสต์	[1,2,3,4,5]
tuple	ทูเพิล	(1,2,3,4,5)
set	เซ็ต	{1,2,3,4,5}
dict	ดิกชันนารี	{'ก':1,'ข':2,'ค':3,'ง':4,'จ':5}
range	เรนจ์	range(1,6)

Python Basic Exercise (Data Type)

Exercise 1.1

ผลลัพธ์ a= 25 <class 'int'>

b= 35.5 <class 'float'>

c=25+35.50= 60.5 <class 'float'>

Exercise 1.2

ผลลัพธ์ แปลง ค่า 175 เป็นคำตอบข้างล่าง

- แปลงเลขฐานสิบ เป็น ฐานสอง 0b10101111
- แปลงเลขฐานสิบ เป็น ฐานแปด 0o257
- แปลงเลขฐานสิบ เป็น ฐานสิบหก 0xaf

Python Basic Exercise (pandas)

```
import pandas as pd
data={
      'day':['1/1/2019','2/1/2019','3/1/2019'],
      'temp':[20,21,22],
      'event':['cold', 'hot','cool'],
      }
print(data)
df=pd.DataFrame(data)
print(df)
```

Python Basic Exercise (pandas)

Exercise 2.1

แสดงขนาดของข้อมูล

Exercise 2.1

แสดงข้อมูลค่าที่มากที่สุดใน column “temp”

Python Basic Exercise (Numpy)

```python
import numpy as np
a = np.array([1,2,3,4,5])
print(a)
```

Exercise 3

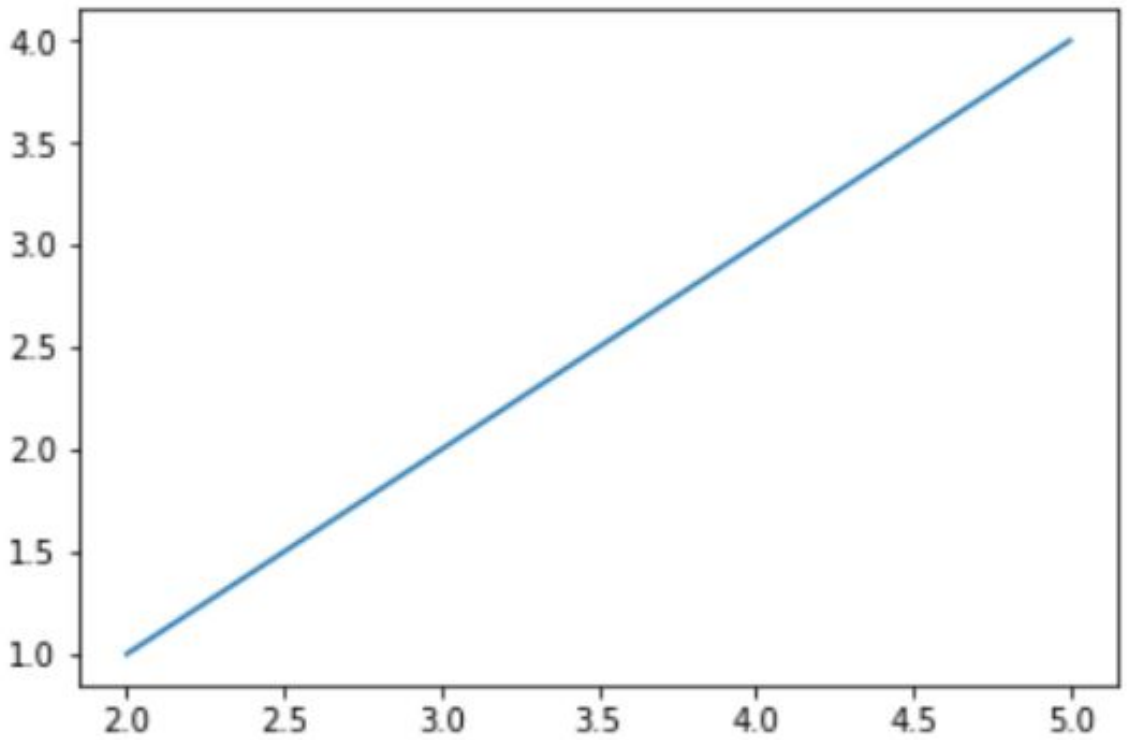
-สร้างอาร์เรย์ a (1 ชุด) คือ 24, 26, 23, 20, 38

-แสดงข้อมูลของอาร์เรย์ในตำแหน่งที่ 3

-แสดงค่าสูงสุดในอาร์เรย์

Python Basic Exercise (Matplotlib)

```
import matplotlib.pyplot as plt
x=2,3,4,5
y=1,2,3,4
plt.plot(x,y)
plt.show()
```

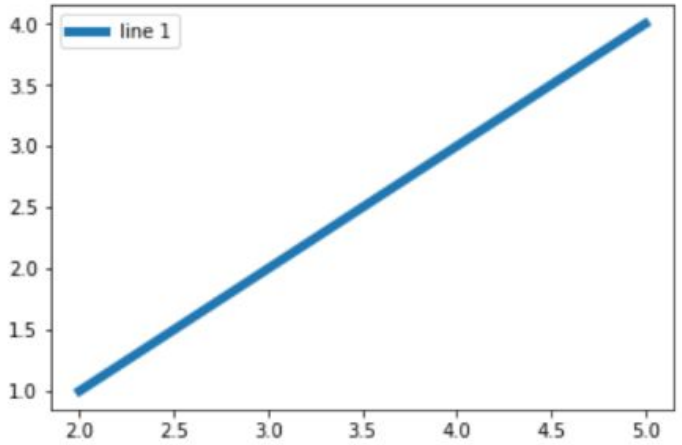


Python Basic Exercise (Matplotlib)

Exercise 4

ปรับขนาดของแส้นให้ใหญ่กว่าเดิม 5 เท่า

อธิบาย เส้น line 1 และกำหนดให้ตำแหน่งที่แสดงให้อยู่มุมบนซ้าย ดังรูป



Data Type and Variable

Ex 1

ตัวแปร Integer, float, String

```
In [1]: ##Int, float, String
        a=3
        b=4.92
        c="Advance programming"

In [2]: print(a,type(a))

        3 <class 'int'>

In [3]: print(b,type(b))

        4.92 <class 'float'>

In [4]: print(c,type(c))

        Advance programming <class 'str'>
```

Data Type and Variable

Ex 2 การจัดรูปแบบการแสดงผลของ String

โดยการแทรกรูปแบบของการแสดงผลใน String literal

%d สำหรับการแสดงผล Integer

```
In [12]: a=7
         b=3
         c=a+b
         d=a/b

In [13]: print('a = %d' %a)

         a = 7

In [14]: print('b = %d' %b)

         b = 3

In [15]: print('c = %d' %c)

         c = 10

In [16]: print('d = ', d)

         d =  2.3333333333333335
```

%f สำหรับการแสดงผล float

```
speed = 80.53
pi=22/7
height=2.31E5
length = 1.3E-3

print('speed =%f' % speed)
print('pi =%f' % pi)
print('height =%f' % height)
print('length =%f' % length)
print(pi)

speed =80.530000
pi =3.142857
height =231000.000000
length =0.001300
3.142857142857143
```

Data Type and Variable

Ex 3 การแสดงผล string

```
In [19]: name = "Jane"
         country = "Thailand"
         language = 'Thai'
         interest = 'game'

In [20]: sentent1 = "what is your name?"
         sentent2 = 'I\'m Jane.'
         sentent3 = "He said \"I would learn python first\"."
         sentent4 = 'His teach replied "oh well"'

In [23]: print (sentent1)
         print (sentent2)
         print (sentent3)
         print (sentent4)

         what is your name?
         I'm Jane.
         He said "I would learn python first".
         His teach replied "oh well"
```

```
: site = 'computer science' + '.com'
  tutorial ='Python' 'Language'

: print (site)
  print (tutorial)

  computer science.com
  PythonLanguage
```

Data Type and Variable

Ex 3.1 การหาความยาว string

```
[22]: name = "Python"
      print(len(name))
```

```
6
```

Ex 3.2 การค้นหาตัวอักษรและกลุ่มตัวอักษรใน string

```
print(name[0])
print(name[-1])
print(name[2])
print(name[-2])
```

```
P
n
t
o
```

```
print(name[3:])
print(name[-5:])
print(name[:4])
```

```
hon
ython
Pyth
```

Data Type and Variable

Ex 4.1 การแสดงผลข้อมูลแบบลำดับ List

```
In [29]: Programing_languges = ["C", "C++", "Java", "Python", "PHP"]

In [30]: print("Index at 0 =", Programing_languges[0])
         print("Index at 3 =", Programing_languges[3])
         Programing_languges[0] = "scalar"

         Index at 0 = C
         Index at 3 = Python

In [31]: print("Index at 0 =", Programing_languges[0])

         Index at 0 = scalar
```

Data Type and Variable

Ex 4.2 การแสดงผลข้อมูลแบบลำดับ Tuple

```
In [32]: a = ('name', 'lastname', 'age')
         b = ('gender', 'school', 'age', 'python')

In [37]: print (a)
         print (b)

         ('name', 'lastname', 'age')
         ('gender', 'school', 'age', 'python')

In [41]: print (len(a))
         print (len(b))

         3
         4
```

Python Data Structures

List & Tuple ข้อแตกต่าง

List: แก้ไขข้อมูลได้ ใช้เครื่องหมาย [ ]

```
my_list = [1, 2, "a"]
```

Tuple: แก้ไขไม่ได้ ใช้เครื่องหมาย ( )

```
my_tuple = (1, 2, "a")
```

Dictionary & Set

Dictionary: เก็บแบบ Key:Value ใช้ { }

```
my_dict = {"name": "John", "age": 25}
```

Set: ข้อมูลไม่ซ้ำกัน ไม่มีลำดับ

```
my_set = {1, 2, 3}
```

Data Type and Variable

Ex 5 การแสดงผลข้อมูลแบบ Boolean

```
a = 4 < 2
b = 6 > 5
print(a, type(a))
print(b, type(b))
```

```
False <class 'bool'>
True <class 'bool'>
```

Data Type and Variable

Ex 6 การแสดงผลด้วยฟังก์ชัน print()

```
lang = "Python"
version = 3.6
print("%s langaung" %lang)
print("version %f" %version)
print("%d" %234)
print("%s %f %d" %(lang, version, 234))
```

```
Python langaung
version 3.600000
234
Python 3.600000 234
```

```
print("tree", "soil", "water", sep=',')
print("one", end =' ')
print("two", end =' ')
print("three", end =' ')
```

```
tree,soil,water
one two three
```

Data Type and Variable

Ex 7 การอ่านข้อมูลจากแป้นพิมพ์- String

```
name = input("Enter your name:")
print("hello"+name)
```

```
Enter your name:tom
hellotom
```

```
name = input("Enter your name:")
print("hello"+'\t'+name)
```

```
Enter your name:tom
hello	tom
```

Expression

ตัวดำเนินการที่ใช้กับข้อมูล numeric

ตัวดำเนินการ	ความหมาย	ตัวอย่าง	ผลลัพธ์
+	บวก	5+2	7
-	ลบ	5-2	3
*	คูณ	5*2	10
/	หาร	5/2	2.5
%	หารเอาเศษ	5%2	1
**	ยกกำลัง	5**2	25
//	หารไม่เอาเศษ	5//2	2

Expression

ตัวดำเนินการที่ใช้กับข้อมูล Boolean (True and False)

&=and, |=or, ^=xor, ~=not

A	B	A and B	A or B	Not A	Not B
True	True	True	True	False	False
True	False	False	True	False	True
False	True	False	True	True	False
False	False	False	False	True	True

Pivot table

รายงานสรุปที่อยู่ในรูปแบบของตาราง

ตัวอย่าง การสร้าง Pivot table โดยใช้ Excel
Source dataset : https://archive.ics.uci.edu/dataset/352/online+retail

[79]:
```
data2 = pd.read_csv("data/OnlineRetail.csv", encoding='unicode_escape')
```

[80]:
```
data2
```

[80]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	1/12/2010 8:26	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	1/12/2010 8:26	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	1/12/2010 8:26	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	1/12/2010 8:26	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	1/12/2010 8:26	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	9/12/2011 12:50	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	9/12/2011 12:50	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	9/12/2011 12:50	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	9/12/2011 12:50	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	9/12/2011 12:50	4.95	12680.0	France

541909 rows × 8 columns

Pivot table

Ex 1 Pivot table

[99]:
```
data2.head()
```

[99]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	total	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	1/12/2010 8:26	2.55	520	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	1/12/2010 8:26	3.39	692	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	1/12/2010 8:26	2.75	748	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	1/12/2010 8:26	3.39	692	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	1/12/2010 8:26	3.39	692	17850.0	United Kingdom

[101]:
```
data2.tail()
```

[101]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	total	CustomerID	Country
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	9/12/2011 12:50	0.85	347	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	9/12/2011 12:50	2.10	428	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	9/12/2011 12:50	4.15	564	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	9/12/2011 12:50	4.15	564	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	9/12/2011 12:50	4.95	505	12680.0	France

Pivot table

Ex 2.1 Create Pivot table

```
[111]: pd.pivot_table(data2, values="total", index="Country").head()
```

[111]:

Country	total
Australia	3768.853058
Austria	861.493766
Bahrain	981.368421
Belgium	679.795070
Brazil	1215.156250

Pivot table

Ex 2.2 Create Pivot table with 2 indexes

```
[123]: pd.pivot_table(data2, values="total", index=["Country","CustomerID"], aggfunc="sum").head().sort_values(by="total", ascending=False)
```

[123]:

		total
Country	**CustomerID**	
Australia	**12415.0**	4278713
	12388.0	94551
	12393.0	53811
	12386.0	13665
	12422.0	13132

Pivot table

Ex 2.3 Create Pivot table and add columns

```
[125]: pd.pivot_table(data2, values="total", index=["Country","CustomerID"], columns="InvoiceDate", aggfunc="sum").head()
```

[125]:

	InvoiceDate	1/11/2011 10:01	1/11/2011 10:02	1/11/2011 10:04	1/11/2011 10:12	1/11/2011 10:13	1/11/2011 10:16	1/11/2011 10:21	1/11/2011 10:27	1/11/2011 10:30	1/11/2011 10:32	...	9/9/2011 15:25
Country	**CustomerID**												
Australia	**12386.0**	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN
	12388.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN
	12393.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN
	12415.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN
	12422.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN

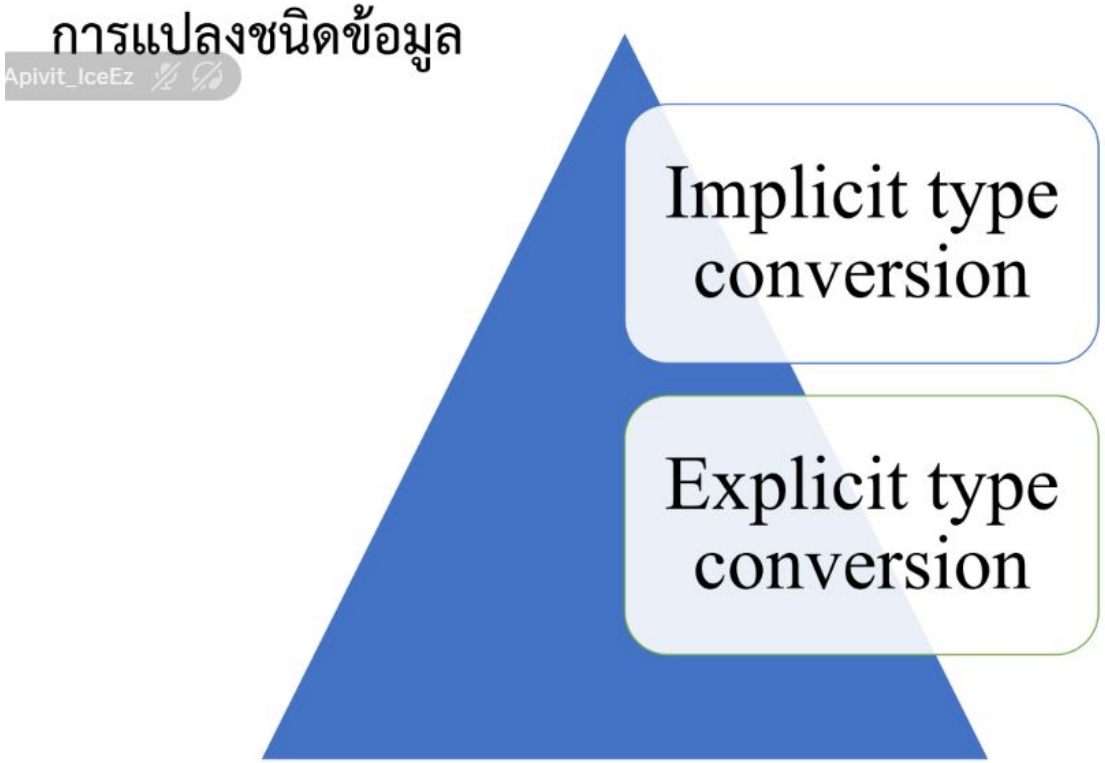
5 rows × 20460 columns

Python Basic Practice

https://iceez555.github.io/python-quiz/

Type Conversion

การแปลงชนิดข้อมูล



Type Conversion

Ex 1 Implicit type conversion –> python interpreter

```
[199]: a = 6        # int (จำนวนเต็ม)
       b = 15.5     # float (ทศนิยม)
       c = a + b    # Python จะแปลงผลลัพธ์เป็น float ให้อัตโนมัติ

       print("a = " , a , "and Type of a = " , type(a))
       print(f"b = {b} and Type of b = {type(b)}")
       print("result c = ", c)
       print(type(c))
```

```
a =  6 and Type of a =  <class 'int'>
b = 15.5 and Type of b = <class 'float'>
result c =  21.5
<class 'float'>
```

Type Conversion

Ex 2.1 Explicit type conversion -> type casting

```
[223]: Pork_price = input("กรุณากรอกราคาหมู: ")
Chicken_price = input("กรุณากรอกราคาไก่: ")
Total_price=int(Pork_price)+int(Chicken_price)
print("ราคาหมู: " + Pork_price + " บาท")
print("ราคาไก่: " + Chicken_price + " บาท")
print("ราคาทั้งหมด " + str(Total_price) + " บาท")
```

```
กรุณากรอกราคาหมู:  25
กรุณากรอกราคาไก่:  15
ราคาหมู: 25 บาท
ราคาไก่: 15 บาท
ราคาทั้งหมด 40 บาท
```

Type Conversion

Ex 2.2 การอ่านข้อมูลจากแป้นพิมพ์-Integer

```
[225]: a = int(input("First num: "))
       b = int(input("Second number: "))
       print("a+b = %d" %(a+b))

       First num:  25
       Second number:  71
       a+b = 96
```

Math (round and sum)

Ex 1 การปัดเศษ (Round)

```
[227]: print(round(4))
       print(round(4.5))
       print(round(4.6))
       print(round(-0.1))
       print(round(-0.6))

       4
       4
       5
       0
       -1
```

Ex 2 ผลรวม (Sum)

```
[231]: print(sum([1,2,3]))
       print(sum([1,2,3],10))

       6
       16
```

ตัด space (lstrip, rstrip, strip)

Ex 1 การตัดช่องว่าง (Strip)

```
[239]: text = "   a line   "
       print(f"Original: '{text}'")
       print(f"Lstrip  : '{text.lstrip()}'")  # ตัดซ้าย
       print(f"Rstrip  : '{text.rstrip()}'")  # ตัดขวา
       print(f"Strip   : '{text.strip()}'")   # ตัดทั้งคู่

       print("-" * 20)
```

```
Original: '   a line   '
Lstrip  : 'a line   '
Rstrip  : '   a line'
Strip   : 'a line'
--------------------
```

ตัวดำเนินการเปรียบเทียบ

Operator	Description	Example	result
==	equal	a == b	False
!=	not equal	a != b	True
>	greater than	a > b	False
<	less than	a < b	True
>=	greater than or equal	a >= b	False
<=	less than or equal	a <= b	True

Selection (if-elif-else)

Ex 1 Python if Statement Syntax

```
if test expression:
    statement(s)
```

[255]:
```
num = 3
if num > 0:
    print(num, "is a positive number.")
print("This is also always printed")

num1 = -1
if num1 > 0:
    print(num, "is a positive number.")
print("This is also always printed")
```

```
3 is a positive number.
This is also always printed
This is also always printed
```

Selection (if-elif-else)

Ex 2 Syntax of if...else

```
if test expression:
     Body of if
else:
     Body of else
```

```
[261]: num1 = -1
       if num1 > 0:
           print(num, "is a positive number.")
       else:
           print(num1,"Negative number")
```

```
-1 Negative number
```

Selection (if-elif-else)

Ex 3 Syntax of if...elif...else

```
if test expression:
     Body of if
elif test expression:
     Body of elif
else:
     Body of else
```

```
[267]: num = 3.5
if num > 0:
    print("Positive number")
elif num == 0:
    print("Zero")
else:
    print("Negeative number")
```

```
Positive number
```

Repetition (while, for)

Ex 1 Syntax of while loop

```
while expression:
    statement(s)
```

```
[3]: count = 0
     print("Start While Loop:")
     while count < 3:  # ทำตราบเท่าที่ count น้อยกว่า 3
         print(f"The count is: {count}")
         count = count + 1
```

```
Start While Loop:
The count is: 0
The count is: 1
The count is: 2
```

Repetition (while, for)

Ex 2 Syntax of for Loop

```
for iterating_var in sequence:
    statements(s)
```

```
[13]: fruits = ['banana', 'apple', 'mango']
      print("Start For Loop:")
      for fruit in fruits:
          print(f"Current fruit: {fruit}")
```

```
Start For Loop:
Current fruit: banana
Current fruit: apple
Current fruit: mango
```

```
[15]: for letter in 'Python':
          print('Current Letter :', letter)
```

```
Current Letter : P
Current Letter : y
Current Letter : t
Current Letter : h
Current Letter : o
Current Letter : n
```