2. Python Syntax, Variables, and Data Types

PYTHON COURSE

SIN YONG TENG

1

Before we start:

1. Lets check if your PIP is working?

2. How to use Git?

2

PIP, is it working? Import a library!

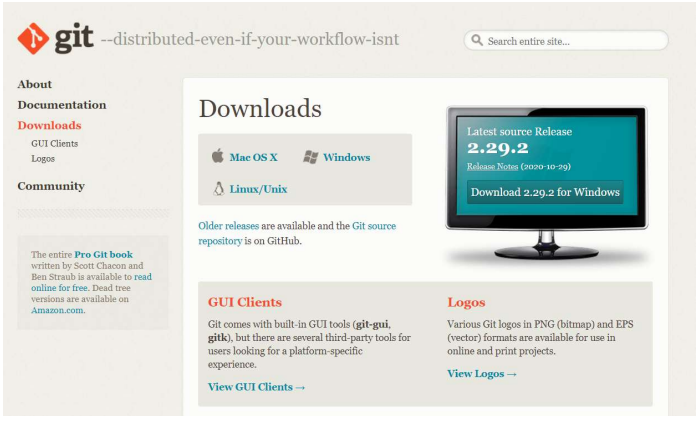
```
Microsoft Windows [Version 10.0.19041.630]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\User\Desktop\Class\ClassCode\P1>pip install pyttsx3
```

```
1  # pip install pyttsx3
2
3  import pyttsx3
4  engine = pyttsx3.init()
5
6
7  msg='Hello World!'
8  engine.say(msg)
9  engine.setProperty('rate', 20) #make it speak slow/fast
10 engine.runAndWait() #run and do not close immediately
```

3

Git/ Github

https://git-scm.com/downloads

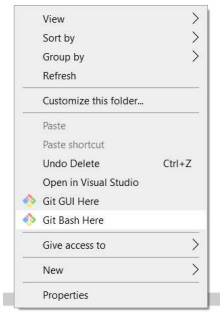


4

How to pull your class material?

1. Make a new folder anywhere

2. Right click -> Git Bash Here



3. Right click -> Git Bash Here

```
User@wa03-0815a MINGW64 ~/Desktop/Class (master)
$ git init
```

git pull https://github.com/tsyet12/ClassCode

5

Python Syntax

- Python is case sensitive

```
print('Hello World!')

Print('Hello World!')
```

```
x=100
print(X)
```

- Python is indentation sensitive

Error!

```
1   print('Hello World!')
2
3   print('Hello World!')
```

- Python comments

```
# This is a single line comment

'''
This is a multiple line comment
See?
'''
x='This is your code'
```

6

What are Python Data Types?

1. Data types are the classification or categorization of data items
2. Variables can store data of different types, and different types can do different things
3. Strictly speaking, data types are class and variables are objects.

```
<class 'str'>
<class 'int'>
<class 'float'>
<class 'int'>
<class 'dict'>
<class 'list'>
<class 'tuple'>
```

4. Python is a dynamically-typed language. Data types do not have to be predefined.

7

Python built-in data types

Category	Variable/Data Types	Example
Text	str	'Hello World' or "Hello World"
Numeric	int, float , complex	x=1, x=0.23, x=1j
Sequence	list, tuple, range	X=[1,2,3], x=(1,2,3), x=range(3)
Mapping	dict	X={"a":1, "b":2}
Set	set, frozenset	X={a,b,c}, x=frozenset({a,b,c})
Boolean	bool	X=True or X=False
Binary	bytes, bytearray, memoryview	b'Hello World', x=bytearray(5), x=memoryview(bytes(5))

8

Text: Strings in python

Texts in programming are called “strings”

```
1  x=input('Type in a string \n')
2
3  y=x[0]
4
5  z=x*10
6
7  e=x+x
8
```

9

Numeric: int, float, complex

```
1  import math
2
3
4  a=15
5  b=2+2j
6  c=1.53
7  d=math.pi
8  print(a+b+c+d)
```

Integers are whole number

Float are numbers with decimals

Complex are imaginary number

10

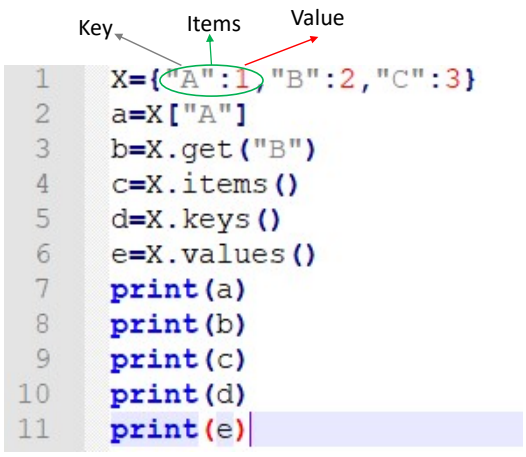
Sequence: List, Tuple, Range

```
1  a=[1,2,3]
2  b=(4,5,6)
3  c=range(6)
4  d=range(1,6)
5
6  print(a[1])
7  print(b[2])
8  print(c)
9  print(d)
```

Question: How to get [2,3] from a?

11

Mapping: Dict (Dictionary)



12

Sets

-Sets cannot have replicated values

```
1  Set={1,1,1,2,3,4,3,4}
2  List=[1,1,1,2,3,4,3,4]
3  Tuple=(1,1,1,2,3,4,3,4)
4  print(Set)
5  print(List)
6  print(Tuple)
```

13

Bool (Boolean)

True or False only

```
1  x=True
2  x=False
3
4  if x is True:
5      print("Hello World!")
6  else:
7      print("Hello Fake World!")
8
9
```

14

Challenge 1: Item to shop code

You are making an item to shop code converter. The customer will give you the name of the item and you must print the shop code:

Item name	Shop Code
Carrot	0001
Cabbage	0010
Potato	0011
Tomato	0100
Water	0101

15

Challenge 2: Make a sequence of the Fibonacci number in a list

E.g. [1,1,2,3,5,8,13,21,…] $F_n = F_{n-1} + F_{n-2}$. $\lim_{n\to\infty} \frac{F_{n+1}}{F_n} = \varphi.$

```
1  x=[1,1]
2
3  for i in range(100):
4          ????????
5
6  print(x)
```

16

Homework: Data type logic

You are given 2 input lists a=[a_1,a_2,a_3] and b=[b_1,b_2,b_3] where a1,a2,a3,b1,b2,b3 are integer lesser than 3.
Print list c=[c_1,c_2,c_3]

Corresponding element of a_n+b_n=	c_n
1	"ONE"
2	"TWO"
3	"THREE"
4	"FOUR"
5	"FIVE"
6	"SIX"

17

Conclusion

1. Checking PIP and import library
2. Simple Git protocol (init and pull)
3. Python Syntax
4. Python Data Type
5. Text: Strings
6. Numeric: int, float, complex
7. Sequence: list, tuples, range
8. Mapping: Dict
9. Set
10. Bool
11. Mapping Challenge using Dict
12. Using List for Fibonacci

18