



北京航空航天大学
BEIHANG UNIVERSITY

Introduction to Computer Science and Programming

Lab Class 1

Pengfei Yi / Junyu Zhao

Beihang University

Outline

- Short guidance of installation of WinPython & Spyder
- Introduction to Python
 - Execution of Python programs
 - Output and input
 - Variable assignment
 - Commenting source code

WinPython download

What will happen on ~~Friday~~ Thursday?

- You will get used to the lab environment
- Understand some very basic concepts of Python
- The real introduction comes in the next **lecture**
 - If, until then, you want to prepare something, try to play around with Python a little bit
- Please download the following file beforehand:

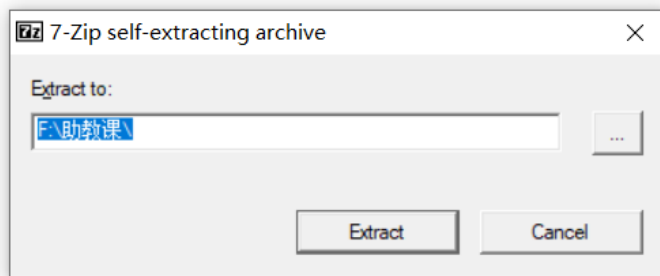
https://sourceforge.net/projects/winpython/files/WinPython_3.7/3.7.4.0/Winpython64-3.7.4.0.exe/download

- from Page 62 of the Lecture slide.
- <http://m3nets.de/teaching/SGE2019/>

WinPython Installation

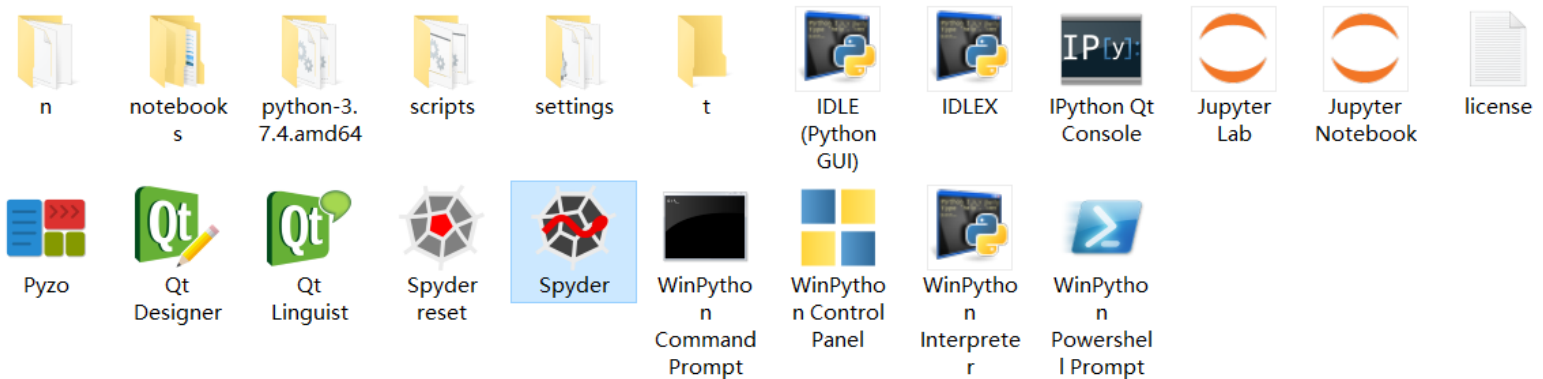
- Run the downloaded .exe file, you will extract it.

名称	修改日期	类型	大小
Winpython64-3.7.4.0	2019/7/19 6:29	应用程序	631,322 KB



WinPython Installation

- Double click on the program and you launch it



Introduction to Python

Execution of Python programs

Execution of Python programs

- Two methods to execute a Python program:
 - Terminal (i.e., cmd in windows).
 - `python3 abc.py`, here `abc.py` is the name of Python file.
 - Simplest way for running a program one time.
 - Integrated Development Environment (IDE).
 - You have a graphical user interface.
 - Many assistants for writing code.
 - We will use the IDE Spyder.

```
1 people = 20
2 cats = 30
3 dogs = 15
4
5
6 if people < cats:
7     print "Too many cats! The world is doomed!"
8
9
10 if people > cats:
11     print "Not many cats! The world is saved!"
12
13
14 if people < dogs:
15     print "The world is drooled on!"
16
17
18 if people > dogs:
19     print "The world is dry!"
20
21
```


The interface of spyder

The image shows the Spyder Python IDE interface. The main window is divided into several panes. The left pane is the Code Editor, which contains a Python script. The right pane is the Variable Explorer, which displays the current state of the program's variables. Below the Variable Explorer is the Command Window, which shows the output of the code execution. The top of the window features a menu bar and a toolbar. A red box highlights the Run button in the toolbar, which is labeled 'Run Button'. The current path of the file being edited is shown in the top right corner, labeled 'Current Path'.

Run Button

Current Path

Code Editor

Variable Explorer

Command Window

```
1 #-*- coding: utf-8 -*-
2 """
3 Spyder Editor
4 """
5 This is a temporary script file.
6 """
7
8 a = 1
9 b = 2
10 c = a+b
11 print('hello world.')
```

Name	Type	Size	Value
a	int	1	1
b	int	1	2
c	int	1	3

Variable explorer | File explorer | Help

IPython console

Console 1/A

Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 20:34:20) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.6.1 -- An enhanced Interactive Python.

In [1]: runfile('D:/WPY64-3740/settings/.spyder-py3/temp.py', wdir='D:/WPY64-3740/settings/.spyder-py3')
hello world.

In [2]:

IPython console | History log

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 10 Column: 8 Memory: 57%

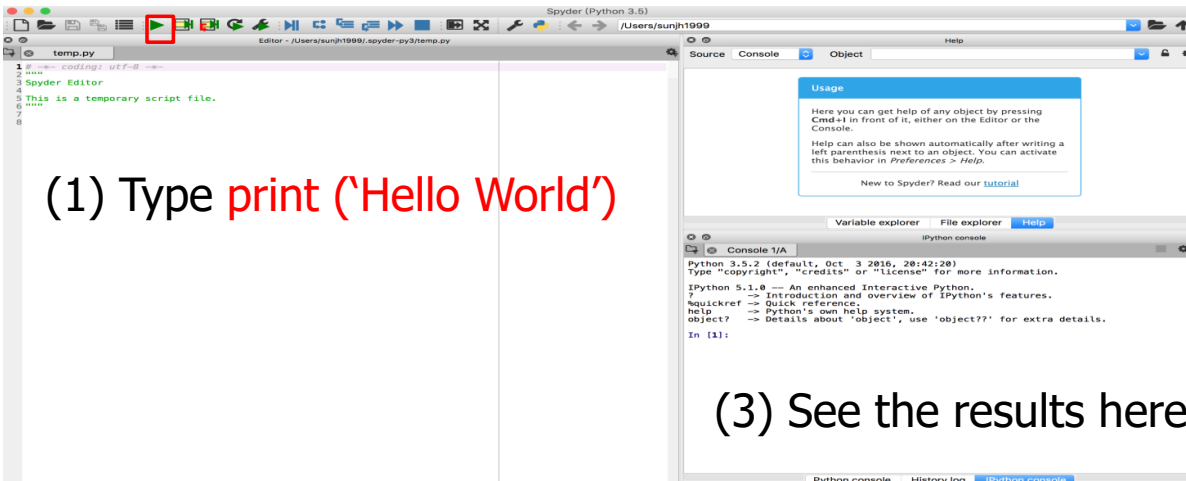
Output and input

Output: printing



- (1) Type `print ('Hello World')` in the **Code Editor**.
- (2) Press the **Run Button**, then save the file.
- (3) See the results in the **Console**.

(2) Press here and save the file



Input: read from keyboard

- The program will use the data which is input by you.
- Type the following statements line by line:
 `a= input('input a number:')`
 `print (a)`
- Run the program and type whatever you want in the command window.
- See the results.

```
input a number: b  
b
```



Variable assignments

Variable assignments

- Type the following statements line by line and see the difference.

```
abc='def'  
print ('first output:',abc)  
print ('second output:', 'abc')
```

```
first output: def  
second output: abc
```



Variable assignments

- Type the following statements line by line and see the results.

```
a=3  
b=7  
print ('a=',a,',b=',b)  
a-=1  
b+=1  
print ('a=',a,',b=',b)
```

```
a= 3 ,b= 7  
a= 2 ,b= 8
```



Commenting source code

Commenting source code

- It's a good habit to add some comment to explain your codes using `#`
- Cooperation between programmers are very important.
 - It is difficult for other people to maintain your code if they cannot understand the code.
- Of course, you do not need to write comments for those very simple code. Like this:

```
abc='def'          #variable assignment
print ('first output:',abc)      #print variable abc
print ('second output:', 'abc')  #print string 'abc'
```

Task: Print a 'C' using Python

- (1) Print an unlovely 'C' as follows:

```
print ("print C")
print ("*"*9)
print("*")
print("*")
print("*")
print("*")
print("*")
print("*")
print ("*"*9)
```

```
print C
*****
*
*
*
*
*
*
*****
```



Task: Print a 'C' using Python



- (1) Print an unlovely 'C' as follows:

```
print ("print C")
print ("*"*9)
print("*")
print("*")
print("*")
print("*")
print("*")
print("*")
print ("*"*9)
```

```
print C
*****
*
*
*
*
*
*
*****
```

- (2) How to print a beautiful 'C' like this?

```
print C
*****
*               *
*
*
*
*
*
*               *
*****
```

Open questions?

Suggestions from your side?



Thank you very much!

**If you have any questions,
please get in touch with us:
zhaojunyu@buaa.edu.cn**