北京航空航天大學
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?

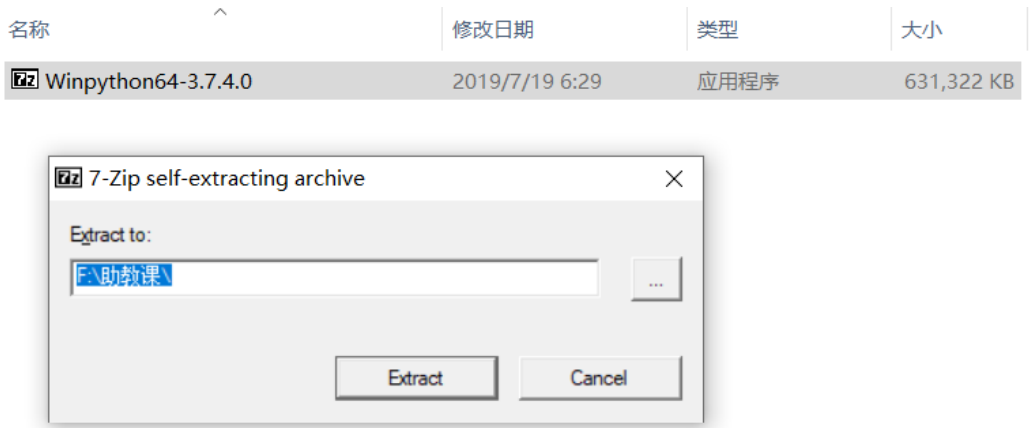
- You will get used to the lab environment
- Understand some very basic concepts of Python
- The real introduction comes in the next **lecture**
  - If, until then, you want to prepare something, try to play around with Python a little bit
- Please download the following file beforehand:

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/**

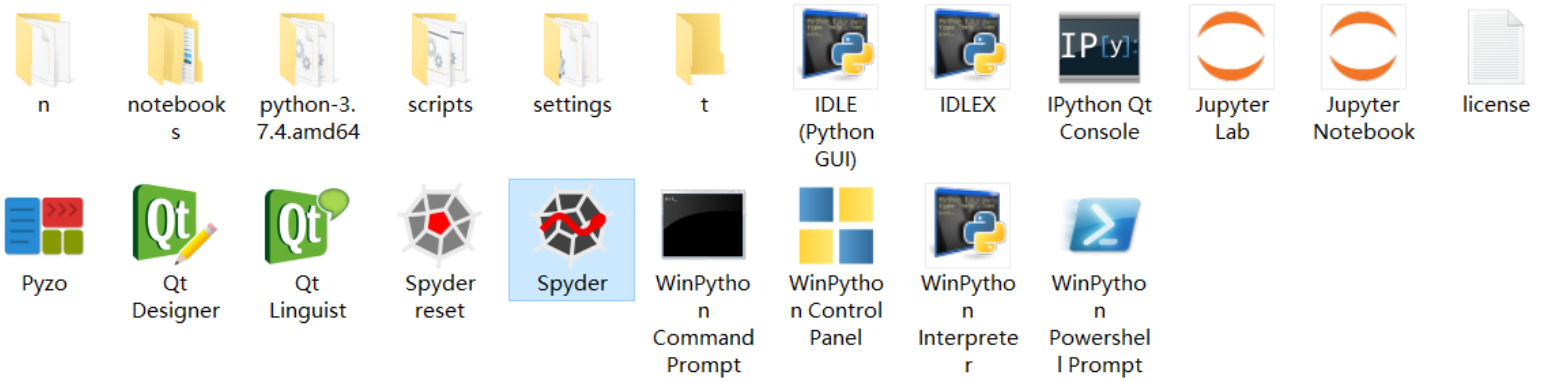
WinPython Installation

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



WinPython Installation

- Double click on the program and you launch it



Introduction to Python

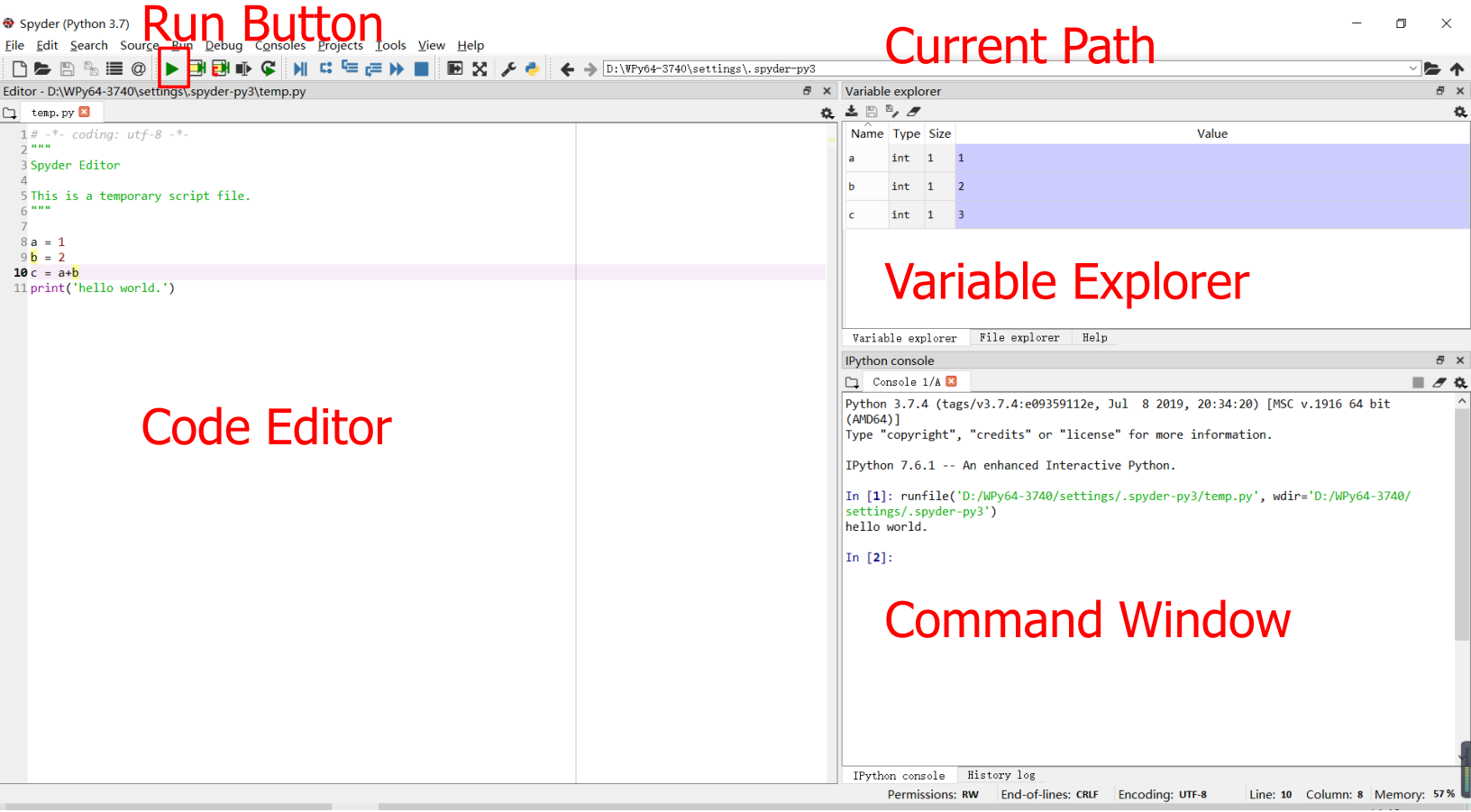
Execution of Python programs

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.

```
people = 20
cats = 30
dogs = 15


if people < cats:
  print "Too many cats! The world is doomed!"


if people > cats:
  print "Not many cats! The world is saved!"


if people < dogs:
  print "The world is drooled on!"


if people > dogs:
  print "The world is dry!"

```

The interface of spyder

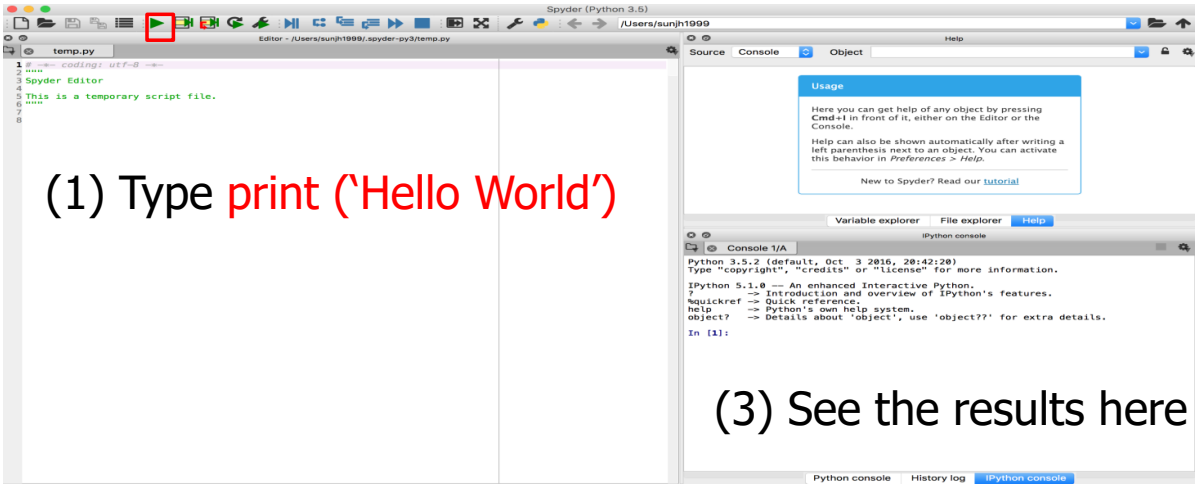


Output and input

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