DSA5101 Introduction to Big Data for Industry

Lecture 2 Primer to Python

LX Zhang

Department of Mathematics

National University of Singapore

Python

“It (Python) has

-- high-level data structures and

-- a simple but effective approach
to object-oriented programming”

- Python is a mix language
- Python documentation:
  https://docs.python.org/3/index.html

  https://www.jetbrains.com/lp/python-developers-survey-2019/

Why can you do with Python?

1. Scripting

    -- Python has a powerful library function for processing string.

2. Data processing in data science, machine learning and AI

    -- Process text data

    -- Machine learning application

    -- Visualization

3. Web development

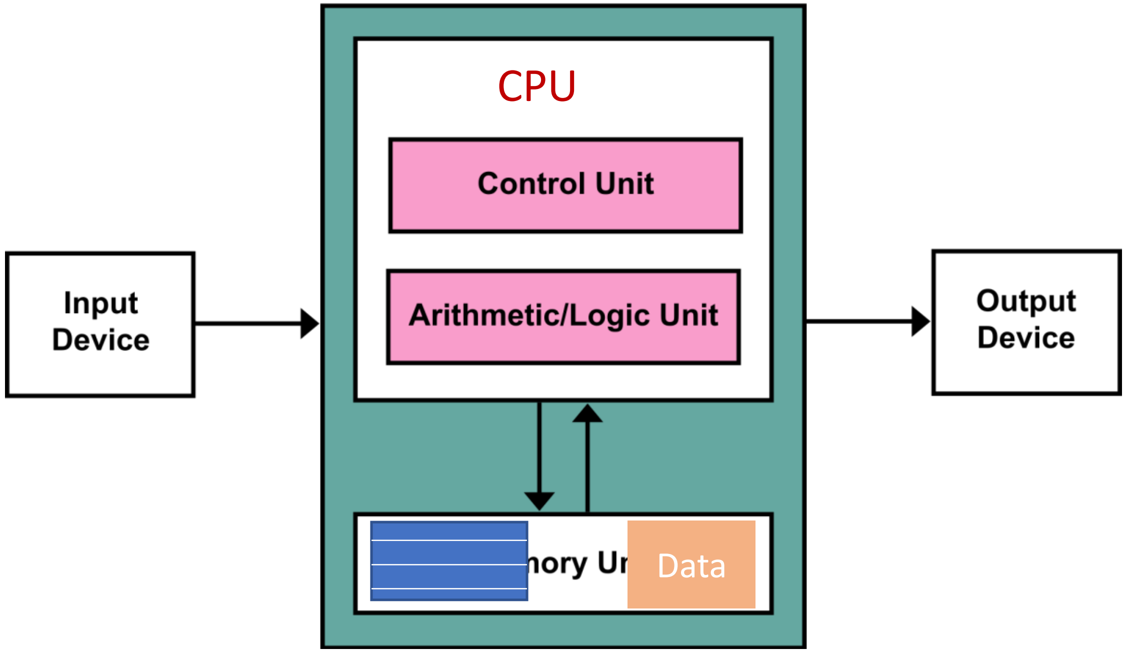
    -- Web development framework like Django and Flask are based on Python

    -- Python is used to support databases

    -- Python is also used to generate dynamic HTML page

Structure of Computer Programs

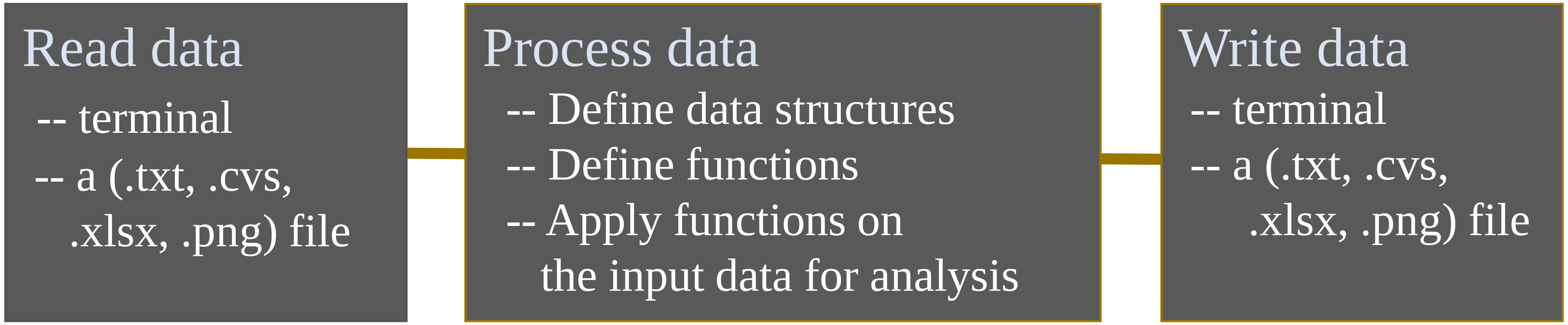
A program is a sequence of instructions that perform a specific function once it is executed



From-WikiPEDIA

A computer program consists of three components

- Input component
- Data analysis using an algorithm
- Output component



Read data

-- terminal

-- a (.txt, .cvs, .xlsx, .png) file

Process data

-- Define data structures

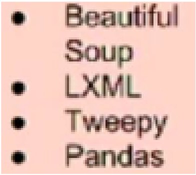
-- Define functions

-- Apply functions on the input data for analysis

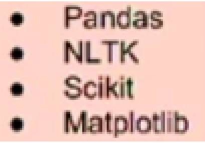
Write data

-- terminal

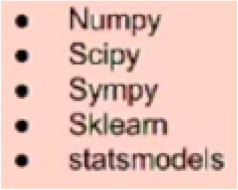
-- a (.txt, .cvs, .xlsx, .png) file



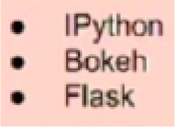
- Beautiful Soup
- LXML
- Tweepy
- Pandas



- Pandas
- NLTK
- Scikit
- Matplotlib



- Numpy
- Scipy
- Sympy
- Sklearn
- statsmodels



- IPython
- Bokeh
- Flask

Basic Data Types

- int {-maximum,…, -1, 0, 1, 2, …, maximum}
- float { …, -2.1, … 0, …, 0.2, …, 0.22222, … }
- Boolean {True, False}

-- For each data type, there are a set of built-in operations for computatin

Remarks: (1) A number cannot be too large or too small; what is the max integer in python?
(2) In Python, each line can contain only one statement

Quiz What operator is “//” ? Are emply lines allowed?

String Type

- A **string** is a sequence of symbols
- Single vs double vs triple quotes
- There are a lots of built-in operations for string manipulation, such as

  -- length function

  -- slicing functions

  -- upper and lower conversion

  -- substring (word) count

  -- replace a word with another
- Use dir(str) to check the methods and attributes associated with a type
- Use help(str) to check details of function and methods.

```
name = “Joe Smith”
college = ‘NUS’
fact = name + “ graduated from “ + college
print(fact)
print (f’{name} graduated from {college}’)
first_name=name[0:3]
reverse = name[-4:-1]
print(name.lower())
```

Warning: Anything input from terminal is a string

Quiz What is the function of triple quotes?

Homework Grasp the functions defined on strings

Advanced topic regular expression

Lists, Sets and Tuples

- (Ordered) List instances [1, 2, “tree”, 4, “john”, 2, 2.5]
- Slicing and other functions are associated with list
- Conversion between sentences and words lists:

    -- s.split(‘-’) splits s into a list of words not contains the parameter ‘-’

    -- ‘-’.join(L) to concatenate a list of words and ‘-’ into a string.

```
courses=[“math”, “science”, “history”, “chemistry”]
courses.append(“art”)
courses.insert(2, “art”)
course.extend([“art”, “education”])
my_module=course[2]
course.pop()
course.sort()
```

Quiz What is the value of my_module?

Homework Grasp functions defined on lists

Mathematics is all about the following objects

-- elements

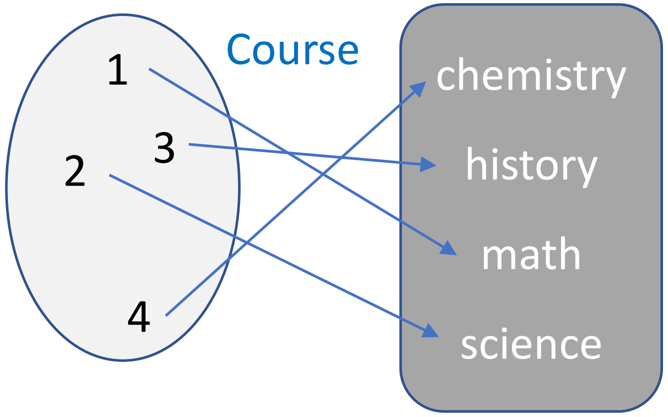
-- (finite) sets

-- (finite) functions and relations in general

- List, Set, Tuples are the mechanisms to definite

  -- a finite set, or

  -- a simple function on integers
- List of tuples defines a relation



```
courses=[“math”, “science”, “history”, “chemistry”]

Book_read=[(‘zhang’, book1), (‘zhang’, book2),
              (‘wang’, book3), (‘li’, book3)]
```

Mathematics is all about the following objects

-- elements

-- (finite) sets

-- (finite) functions and relations in general

- List, Set, Tuples are the mechanisms to definite

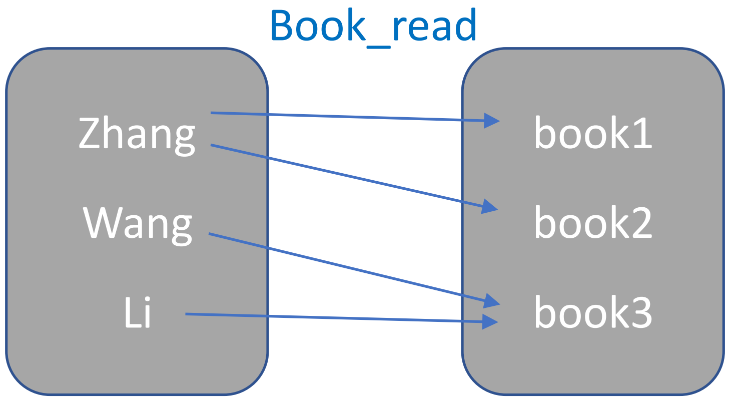
  -- a finite set, or

  -- a simple function on integers

- List of tuples defines a relation

```
courses=[“math”, “science”, “history”, “chemistry”]

Book_read=[(‘zhang’, book1), (‘zhang’, book2),
                (‘wang’, book3), (‘li’, book3)]
```



```
num=[1, 2, 3, 4]
num2=num.copy()
num.append(10)
print(num2)
```

```
num=[1, 2, 3, 4]
num2=num
num.append(10)
print(num2)
```

Question What is the value of num2 after each piece of code is executed?

Exercise: Write a short code to remove the duplicates in a list.

List of List

- Two dimensional arrays

  matrix=[[1, 2, 3], [3, 4, 5], [5, 6, 8]]

```
matrix=[ [1, 2, 3],
         [3, 4, 5],
         [5, 6, 8]]
matrix[1]
matrix[2]
```

Dictionaries

- A dictionary is a mapping or function from somethings to somethings.
- English dictionary is an example for dictionary data type

A list

0	item0
1	item1
2	item2
3	item3
4	item4

Natural index

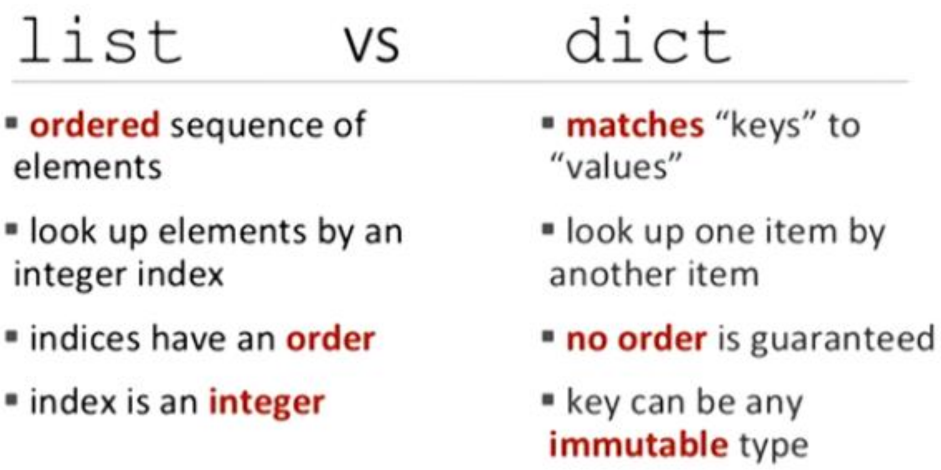
element

A dictionary

key0	value0
key1	vlaue1
key2	value2
key3	value3
key4	value4

Custom index

element



list	vs	dict
▪ **ordered** sequence of elements		▪ **matches** "keys" to "values"
▪ look up elements by an integer index		▪ look up one item by another item
▪ indices have an **order**		▪ **no order** is guaranteed
▪ index is an **integer**		▪ key can be any **immutable** type

Branching and Iteration

- Branching is supported by

```
“if <condition1>:
      <actions>
  elif <condition2>:
       <actions>
  else:
      <actions>”
```

Q: How to form a condition?

```
#  a code for defining a discrete function f
if  x in [1, 2, 3]:
     f=-1
elif x in [4, 5]:
     f=1
else:
    f=0
```

```
#  t_course denotes the course to take today
if  t_course== “math:
     print(“review math at 8am”)
elif t_course==“computer science”:
     print(“code at 8am”)
else:
     print(“sleep”)
```

Q: How to form a condition?

- An integer, a list, a dic

  - non-0 value is True
  - non-empty list is True
  - non-empty dic is True

- A logic sentence formed by logical operator

```
 #  x is an integer value

 if  x:
      print(“x is not zero”)
else:
     print(“x is zero”)
```

```
#  L is a list variable

if  L and x in L:
     print(“x in L”)
elif L:
     print(“x is not in L”)
else:
     print(“L is empty”)
```

Iterations

- Iteration is supported by

  “for <statement>:

  <actions> ”

  or

  “ while <condition>:

  <actions>”

- for loop iteration over some predefined values

- Iteration can be nested

```
#  a piece of code for search an item in a list
#  nums is a list

for   item in nums:
      if item==3:
            print(“3 is found”)
            break
```

Q: What are iterators?

- While loop performs dynamic iterations. Its outcome depends on computing within the loop condition

```
x=0
while x<=5:
      print(x*x)
      x +=1
```

- A “for loop” can be easily converted into a “while loop”

```
x=0
while True:
      if (x>5)
              break
      print(x*x)
      x +=1
```

Reading & Writing to Files

- Python functions supporting I/O are like C

  -- f=open(“test.txt”, ‘c’), c=r, w, a fore read, write and append, resp.

  -- f.read(size), f.readlines(), f.readline()

  -- f.write(“this is a test”).

```
#  open a file and read line by line

with open('test.txt', 'r') as f:
     for line in f:
         print(line, end='')
```

```
#  open a file and read chunk by chunk

with open('test.txt', 'r') as rf:
             with open('output.txt', 'w') as wf:
           chunk_size=4096
           rf_chunk=rf.read(chunk_size)
           while len(rf_chunk) > 0:
               wf.write(rf_chunk)
               rf_chunk=rf.read(chunk_size)
```

Question: How to read image files?

Advanced topic How to read zipped files?

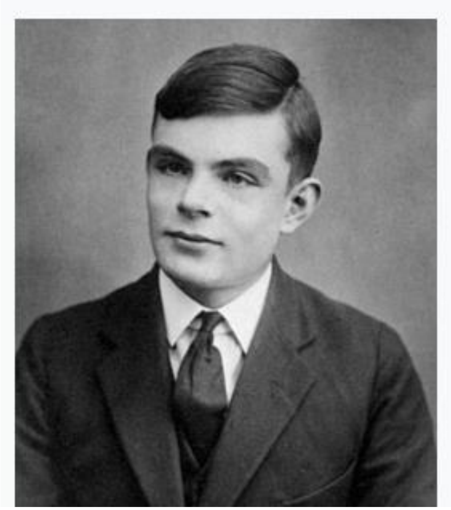
Summary

We have covered basic language mechanism

- Basic data types
  - -- int, float, Boolean
  - -- str
  - -- list, tuple and set
  - -- dictionary
- (branching) if statements
- (iteration) for loop and while loop
- Reading and writing to files

https://www.youtube.com/watch?v=kqtD5dpn9C8

History: Turing’s (图灵)Work

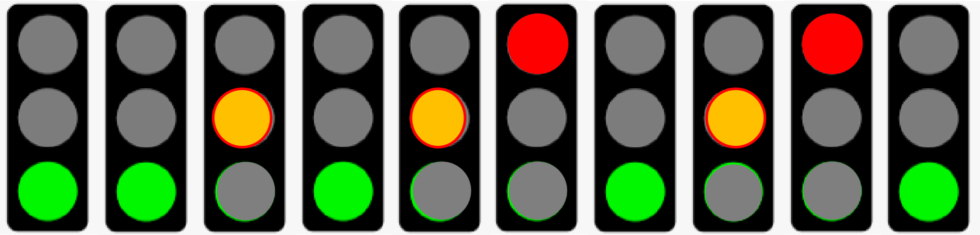


- A good carpenter need only a saw and a ruler
- 好厨子，一把盐
- In 1930’s, Turing proved a very elegant theorem that implying that

  “Any computation can be done with an array of memory in which 0, 1 can be written and six simply instructions including “read”, “write”, left move, right move etc.

In other words,

“Using all traffic lights in Singapore, one can compute everything.”



- 3A: Abstraction, Algorithm and Automation