CH3: Introduction to interactive mode

As we discussed before, python offers interactive modes, that means, commands are read from terminal (python shell) , executes and print the result. Let start with the python interactive modes:

3.1 Python as a calculator:

Open the terminal and type python to go to the python terminal. After typing python it will open a python shell, each line start with “>>>” three greater-than sign, where we can type the command. This prompt is called primary prompt:. ‘

$python

Python 2.7.6 (default, Nov 10 2013, 19:24:18) [MSC v.1500 32 bit (Intel)] on win32

Type "copyright", "credits" or "license()" for more information.

>>>

First it will show the Python version number (Here it is 2.7.6) build date and machine type.

Now type the following and Notice the output:

1. 1+2
2. (30-20)/3\*7
3. 10/-3
4. (1+2+4+6+9+10)/10

The output is displayed as below:

>>>

>>>

>>> 1+2

3

>>> (30-20)\*3/7

4

>>> 10/-3

-4

>>> (1+2+4+6+9+10)/10

3

>>>

As you can see, you can do all kind of Arithmetic operation (like +, -,\*, /, %) with this easy calculator. Cool! But Python can do more than what a simple calculator can do.

1. **Say hello to the world** 
   * **Write a program to print “hello World”**
   * **Print another string “Welcome to the world of python”**
   * **Print multiline string using python**

>>> print 'hello World'

**hello World**

* >>> print "Welcome to the world of python "

**Welcome to the world of python**

>>> print """ Welcome to

1. Python

2. Django

"""

**Welcome to**

**1. Python**

**2. Django**

1. **Assign value and do some arithmetic’s**
   * Define two variable width and height and find the area and printing it.
   * Assign a, b, c, d to 100 by chain assignment
   * Show that arithmetic of integers return integers but If any of the operand is float, returns a float results

>>> width =10

>>> height = 10\*30

>>> area = width \* height

>>> area

3000

>>> width

10

>>> a = b = c = d =100

>>> a

100

>>> b

100

>>> c

100

>>> d

100

>>> 3/2

1

>>> 3/2.0

1.5

>>> 3.0/2.0

1.5

>>> 3.0 / 2

1.5

1. Do some operation on strings:
   * Create two variables a and b contains ‘python’ and ‘hello’, do the concatenation and save the result into another variable c - printing it.
   * Also print first 5 character from ‘hellopython’
   * Now Reverse “hellopython”

>>> a = 'Python'

>>> a

'Python'

>>> b = 'hello'

>>> c = b + a

>>> c

'helloPython'

>>> print c[:5]

hello

>>> c[::-1] 🡨 It will Reverse

'nohtypolleh'

>>>

1. Play with the simple list:
   * Here create a list of number 1 to 6 and find the sum.
   * Create a list containing 3 word and join them with space in between
   * Create a list of unsorted list and print in sorted order
   * Print the sum of 1 to 10,000
   * Print 20 to 1 in reverse direction

>>> a = [1,2,3,4,5,6]

>>> sum(a)

21

>>> b = ['I','Love','python']

>>> b

['I', 'Love', 'python']

>>> ' '.join(b)

'I Love python'

>>> c = [3,4,10,2,1,9]

>>> sorted(c)

[1, 2, 3, 4, 9, 10]

>>> sum(range(1,10000))

49995000

>>> range(1,21)

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,20]

>>> range(1,21)[::-1]

[20,19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1]

>>>

1. Do some error and see how python interacts
   * Try to divide a number by zero
   * Try to concatenate a integer with string
   * Print a variable without defining it

>>> 10/0

Traceback (most recent call last):

File "<pyshell#54>", line 1, in <module>

10/0

ZeroDivisionError: integer division or modulo by zero

>>> a = 15

>>> a +'hello'

Traceback (most recent call last):

File "<pyshell#56>", line 1, in <module>

a+'hello'

TypeError: unsupported operand type(s) for +: 'int' and 'str'

>>> str(a) + 'hello'

'15hello'

>>> print y

Traceback (most recent call last):

File "<pyshell#58>", line 1, in <module>

print y

NameError: name 'y' is not defined

>>>