

Python: Basics

FOSSEE

1 Data types

1.1 int and float

A whole number is a `int` variable.

```
In []: a = 13
In []: type(a)
Out []: <type 'int'>
In []: b = -2
In []: type(b)
Out []: <type 'int'>
In []: c = 500000000
In []: type(c)
Out []: <type 'int'>
```

A number with decimal is a `float`.

```
In []: p = 3.141592
In []: type(p)
Out []: <type 'float'>
```

1.2 Complex Numbers

```
In []: c = 3+4j #coeff of j specifies imaginary part
In []: abs(c) #absolute value of complex number
Out []: 5.0
In []: c.imag #accessing imaginary part of c
Out []: 4.0
In []: c.real #accessing real part of c
Out []: 3.0
```

1.3 Boolean

```
In []: a = False
In []: b = True
In []: c = True
In []: (a and b) or c #Boolean operations
Out []: True
```

Note: Python is case sensitive language, True is bool type, but true would be a variable. and hence following assignment fails:

```
In []: a = true
```

1.4 Strings

```
In []: w = "hello" #w is string variable
In []: print w[1]
Out []: e
In []: print w[-1] #last character of string
Out []: o
```

Note: For a string variable, individual elements can be accessed using indices.

Note: All slicing and striding operations works with strings as well.

```
In []: len(w) #function to calculate length of string
Out []: 5
In []: w[0] = 'H' # ERROR: Strings are immutable
```

1.5 String methods

```
In []: a = 'Hello World'
In []: a.startswith('Hell') # 'a' starts with 'Hell'
Out []: True
In []: a.endswith('ld') # 'a' ends with 'ld'
Out []: True
In []: a.upper() # all characters to upper case
Out []: 'HELLO WORLD'
In []: a.lower() # all characters to lower case
Out []: 'hello world'
In []: ''.join(['a', 'b', 'c'])
```

```
Out []: 'abc'
```

join function joins all the list member passed as argument with the string it is called upon. In above case it is empty string.

```
In []: ' '.join(['a','b','c'])
```

```
Out []: 'a b c'
```

```
In []: ','.join(['a','b','c'])
```

```
Out []: 'a,b,c'
```

1.6 String formatting

```
In []: x, y = 1, 1.234 #initializing two variables
```

```
In []: 'x is %s, y is %s' %(x, y)
```

```
Out []: 'x is 1, y is 1.234'
```

Note: %s used in above formatting specifies 'str' representation of variables. One can also try:

%d for int representation

%f for float representation

```
In []: 'x is %f, y is %f' %(x, y)
```

```
Out []: 'x is 1.000000, y is 1.234000'
```

```
In []: 'x is %d, y is %d' %(x, y)
```

```
Out []: 'x is 1, y is 1'
```

1.7 Arithmetic Operators

```
In []: 45 % 2 # Modulo operator
```

```
Out []: 1
```

```
In []: 5 ** 3 # Power
```

```
Out []: 125
```

```
In []: a = 5
```

```
In []: a += 1 #increment by 1, translates to a = a + 1
```

```
In []: a *= 2
```

1.8 String Operations

```
In []: s = 'Hello'
```

```
In []: p = 'World'
```

```
In []: s + p #concatenating two strings
Out[]: 'HelloWorld'
In []: s * 4 #repeat string for given number of times
Out[]: 'HelloHelloHelloHello'
```

1.9 Relational and Logical Operators

```
In []: p, z, n = 1, 0, -1 #initializing three variables
In []: p == n #equivalency check
Out[]: False
In []: p >= n
Out[]: True
In []: n < z < p #finding largest number among three
Out[]: True
In []: p + n != z
Out[]: False
```

1.10 Built-ins

```
In []: int(17 / 2.0) #converts arguments to integer
Out[]: 8
In []: float(17 / 2) #argument is already integer(17 / 2 = 8)
Out[]: 8.0
In []: str(17 / 2.0) #converts to string
Out[]: '8.5'
In []: round( 7.5 )
Out[]: 8.0
```

1.11 Console Input

```
In []: a = raw_input('Enter a value: ')
Enter a value: 5
```

Note: `raw_input` always returns string representation of user input and hence:

```
In []: type(a)
Out[]: <type 'str'>
```

To get integer or floating point of this input, one has to perform type conversion:

```
In []: b = int(a)
```

2 Conditionals

if

```
In []: x = int(raw_input("Enter an integer:"))
In []: if x < 0:
...:     print 'Be positive!'
...: elif x == 0:
...:     print 'Zero'
...: elif x == 1:
...:     print 'Single'
...: else:
...:     print 'More'
```

Ternary Operator

```
In []: a = raw_input('Enter number(Q to quit):')
In []: num = int(a) if a != 'Q' else 0
```

Above statement can be read as “num is int of a, if a is not equal to ‘Q’, otherwise 0”

3 Links and References

- Reference manual to describe the standard libraries that are distributed with Python is available at <http://docs.python.org/library/>
- To read more on strings refer to:
<http://docs.python.org/library/stdtypes.html#string-methods>