

# Basic Python

1.1 (c) 2014 Maker Works  
→ means *will produce* or *yields* and is not part of Python

## Basic Syntax

# Comment to end of line  
Indentation by same amount groups statements together

Blank lines are okay  
Okay to split lines inside open ( [ {  
Separate simple statements with ;  
a=1; b=2

## Assignments

```
a=1
b="a string"
c=["a","list"]
a += 1 same as a=a+1
+=, -=, /=, *=, etc.
[a,b]=[3,7]
a → 3, b → 7
```

## Comparisons

```
1 < 2, 1 <= 2, 2 == 2, 3 >= 2, 3 > 2
1 != 2
True
False
```

## Boolean

A and B, A or B, not B

## Math Operators

+, -, \*, /, \*\* (power)

## Strings

“single 'and' double”  
'quotes “work” the same'  
' ' 'Use three quotes for multi-line strings' ' '  
“A” “B” “C” → “ABC”

```
“A” + “B” → “AB”
“Is %i it?” % 5 → “Is 5 it?”
“%i %i” % (2,3) → “2 3”
(Note tuple above! No lists.)
Types:
%s: string
%d, %i: integer
%f: floating point number
%<width>.<precision><type>
“X %5s X” % “Y” → “X   Y X”
“%.2f” % 3.14159 → “3.14”
```

## Lists

```
a=[1,2,3]
a[2] → 3
a[0] → 1 (index starts at 0)
```

## Tuples

A tuple is an unmodifiable list  
a=(1,2,3)  
a[2] → 3 (yes, square brackets)

## Dictionaries

```
b={'name':'bob', 'age':42}
b['name'] → 'bob'
b['age'] → 42
b.keys() → ['age','name']
```

## Variable and Function Names

Must start with a letter  
Contains letters, digits, \_underscores  
Any length  
Case matters  
Can not be a reserved word (for, if, etc.)  
from time import sleep  
sleep(1)

## Print

```
print 1, “this”, 2 → 1 this 2
```

## Input

```
a=input(“Enter first number”)
```

## If

```
if a>b:
    print “a>b”
elif a<b:
    print “a<b”
else:
    print “a==b”
```

## While

```
a=1
while a<5:
    print a, # comma for no new line
    a = a + 1
```

## For

```
a=[1,3,2,7]
for b in a:
    print b,
→ 1 3 2 7
for b in range(10):
    print b,
→ 0 1 2 3 4 5 6 7 8 9
```

## Functions

```
def addOne(x):
    y=x+1
    return y
addOne(3) → 4
```

## Import

```
import time
time.sleep(1)
from time import sleep
sleep(1)
```

## Try

```
try:
    a= 5/0
except:
    print “divide error”
→ divide error
```

## Built-in Functions

```
abs(-4) → 4
float(“3.14”) → 3.14 (string to number)
help(“for”) → brief help on functions, statements, etc.
int(3.14) or int(“3”) → 3
len([1,2,3,4]) → 4
max([3,4,2]) → 4
min([3,4,2]) → 2
range(start, stop, step)
range(1,4) → [1, 2, 3]
range(1,10,2) → [1,3,5,7,9]
sum([2,3,4]) → 9
```

## Other

global: use value of global variable

## RPi.GPIO

```
setmode(BCM)
setup(pin, OUT)
setup(pin, IN, PUD_UP/PUD_DOWN)
input(pin)
output(pin, LOW/HIGH)
```

## Command Line

Stop program: control-C (^C)  
Run Python as superuser:  
sudo python <progname>.py  
Change directory: cd <dirname>  
List files: ls (lower-case L)  
List files w/details: ls -l (lower-case L)  
Find a file:  
sudo find / -name <filename.ext>  
Shutdown the Raspberry Pi from command line: sudo shutdown -h 0  
Reboot: sudo shutdown -r 0