# Simple functions in Python

```python
# use the math library
import math

def funcName(formalParameters):
    # body (defines value)
    return 'value'

# Examples

def f(x):
    return 2 * x

def double(x):
    return 2 * x

def remainder(x, y):
    """ Takes two argument x and y and returns the
    remainder after dividing x by y
    """
    value = x % y  # giving a name to our result
    return value

# or, equivalently,

def remainder(x, y):
    """ Takes two argument x and y and returns the
    remainder after dividing x by y
    """
    return x % y

# area for rectangles and circles

def circleArea(radius):
    return radius ** 2 * math.pi

def rectangleArea(width, length):
    return length * width

def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

def len(list):
    """ This function redefines len()
    ""
    if list == []:  # empty list
        return 0
    else:
        return 1 + len(list[1:])

# But that won't work for strings; try this:

def len1(listOrStr):
    if listOrStr == [] or listOrStr == '':
        return 0
    else:
        return 1 + len1(listOrStr[1:])
```
# functions

```python
>>> f
<function f at 0x1004e0cf8>
>>> f(3)
6
>>> f(0)
0
>>> double = f
>>> double(3)
6
```

```python
>>> double("Darn! ")
'Darn! Darn! '
>>> double(True)
2
>>> remainder(5,2)
1
>>> remainder(0,8)
0
>>> remainder(4,2)
0
>>> help(remainder)
Help on function remainder in module __main__:
remainder(x, y)
    Takes two argument x and y and returns the
    remainder after dividing x by y
```

```python
>>> circleArea(1)
3.141592653589793
>>> circleArea(10.5)
346.3605900582747
>>> circleArea(10)
314.1592653589793
>>> rectangleArea(4,5)
20
>>> rectangleArea(4,-5)
-20
>>> factorial(0)
1
>>> factorial(1)
1
>>> factorial(2)
2
>>> factorial(3)
6
>>> factorial(4)
24
>>> factorial(5)
120
>>> factorial(10)
3628800
>>> factorial(100)
93326215443944152681699233885626670049071596826438162146859296389521759999322991560894146397615651828625369792082722375825118521091686400000000000000000000000L
>>> (factorial(100) / factorial(99))
100L
```
Traceback (most recent call last):
  File "<pyshell#27>", line 1, in <module>
    factorial(-1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)
  File "/Users/wrc/Desktop/python/simpleFunctions.py", line 50, in factorial
    return n * factorial(n - 1)