def foo():
    m = 3
    def bar():
        n = 4
        print m + n
        print m
    bar()
foo()
print 'Goodbye!'
Nested Functions (2)

```python
m = 40

def foo():
    n = 10

    def bar():
        n = 5
        print m + n

    print m + n
    bar()

foo()

print 'Goodbye!'
```
# Function definition is here

def printinfo(name, age):
    "This prints a passed info into this function"
    print "Name:" , name
    print "Age:" , age

# Now you can call printinfo function
print 'Function call #1'
printinfo(50, "miki")

print 'Function call #2'
printinfo(age=50, name="miki")
# Function definition is here

```python
def printinfo(name, age=35):
    "This prints a passed info into this function"
    print "Name: ", name
    print "Age: ", age
```

# Now you can call printinfo function

```python
print 'Function call #1'
printinfo(age=50, name="miki")
```

```python
print '
Function call #2'
printinfo(name="miki")
```
# Function definition is here

def changeme(mylist):
    mylist.sort()
    print 'Values inside changeme():', mylist

def changeme2(mylist):
    mylist.append(12)  # same as mylist += [12]
    mylist.sort()
    print 'Values inside changeme2():', mylist

def changeme3(mylist):
    mylist = mylist + [12]
    mylist.sort()
    print 'Values inside changeme3():', mylist

# Now you can call changeme function
mylist = [19, 5, 87, 32]
Changing Parameters II

```python
mylist = [19, 5, 87, 32]
print '\nmylist values:', mylist
changeme(mylist)
print 'Values outside the function:', mylist

mylist = [19, 5, 87, 32]
print '\nmylist values:', mylist
changeme2(mylist)
print 'Values outside the function:', mylist

mylist = [19, 5, 87, 32]
print '\n'
print '\nmylist values:', mylist
changeme3(mylist)
print 'Values outside the function:', mylist
```