Namespaces: Global vs. Local Variables

Before executing the function double(), variables x and y do not exist.

After executing the function double(), variables x and y still do not exist.

x and y exist only during the execution of the function call double(5).

They are said to be local variables of function double().

In [1]:
```
dir()
```

Out[1]:
```
['In',
 'Out',
 '_',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'exit',
 'get_ipython',
 'quit']
```

In [2]:
```
import random
```

In [3]:
```
dir()
```

Out[3]:
```
['In',
 'Out',
 '_',
 '_1',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i2',
 '_i3',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'exit',
 'get_ipython',
 'quit',
 'random']
```

In [4]:

```
dir(random)
```

Out[4]:

```
['BPF',
 'LOG4',
 'NV_MAGICCONST',
 'RECIP_BPF',
 'Random',
 'SG_MAGICCONST',
 'SystemRandom',
 'TWOPI',
 '_Sequence',
 '_Set',
 '__all__',
 '__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_accumulate',
 '_acos',
 '_bisect',
 '_ceil',
 '_cos',
 '_e',
 '_exp',
 '_inst',
 '_log',
 '_os',
 '_pi',
 '_random',
 '_repeat',
 '_sha512',
 '_sin',
 '_sqrt',
 '_test',
 '_test_generator',
 '_urandom',
 '_warn',
 'betavariate',
 'choice',
 'choices',
 'expovariate',
 'gammavariate',
 'gauss',
 'getrandbits',
 'getstate',
 'lognormvariate',
 'normalvariate',
 'paretovariate',
 'randint',
 'random',
 'randrange',
 'sample',
 'seed',
 'setstate',
 'shuffle',
 'triangular',
 'uniform',
 'vonmisesvariate',
 'weibullvariate']
```

In [4]:

```
random.randrange(1, 7)
```

Out[4]:

```
6
```

In [5]:

```
dir()
```

Out[5]:

```
['In',
 'Out',
 '_',
 '_1',
 '_3',
 '_4',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i2',
 '_i3',
 '_i4',
 '_i5',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'exit',
 'get_ipython',
 'quit',
 'random']
```

In [6]:

```
a = 100
```

In [7]:

```
dir()
```

Out[7]:

```
['In',
 'Out',
 '_',
 '_1',
 '_3',
 '_4',
 '_5',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i2',
 '_i3',
 '_i4',
 '_i5',
 '_i6',
 '_i7',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'a',
 'exit',
 'get_ipython',
 'quit',
 'random']
```

In [8]:

```
# Encapsulation Through Local Variables
def double(y):
    x = 2
    print('x={}, y={}'.format(x, y))
    return x * y

res = double(5)

print(x)
print(y)
```

```
x=2, y=5

---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-8-dc8377e9b63d> in <module>
      7 res = double(5)
      8 
----> 9 print(x)
     10 print(y)

NameError: name 'x' is not defined
```

In [9]:

```
dir()
```

Out[9]:

```
['In',
 'Out',
 '_',
 '_1',
 '_3',
 '_4',
 '_5',
 '_7',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i2',
 '_i3',
 '_i4',
 '_i5',
 '_i6',
 '_i7',
 '_i8',
 '_i9',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'a',
 'double',
 'exit',
 'get_ipython',
 'quit',
 'random',
 'res']
```

In [10]:

```
dir(f)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-10-0f29e100c1c1> in <module>
----> 1 dir(f)

NameError: name 'f' is not defined
```

Even during the execution of double(), local variables x and y are invisible outside of the function.

In [12]:

```
# Encapsulation Through Local Variables
x, y = 20, 50

def double(y):
    x = 2
    print('x = {}, y = {}'.format(x, y))
    return x * y

res = double(5)

print(x, y)
```

```
x = 2, y = 5
20 50
```

In [13]:

```
dir()
```

Out[13]:

```
['In',
 'Out',
 '_',
 '_1',
 '_3',
 '_4',
 '_5',
 '_7',
 '_9',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i10',
 '_i11',
 '_i12',
 '_i13',
 '_i2',
 '_i3',
 '_i4',
 '_i5',
 '_i6',
 '_i7',
 '_i8',
 '_i9',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'a',
 'double',
 'exit',
 'get_ipython',
 'quit',
 'random',
 'res',
 'x',
 'y']
```

In [14]:

```python
a = 10

def f(n):
    a = 5
    b = a + n
    return b
```

In [15]:

```python
f(5)
```

Out[15]:

```
10
```

In [16]:

```python
a
```

Out[16]:

```
10
```

In [17]:

```
b
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-17-89e6c98d9288> in <module>
----> 1 b

NameError: name 'b' is not defined
```

In [18]:

```
# Function Call Namespace

def h(n):
    print('Start h')
    print(1 / n)
    print(n)

def g(n):
    print('Start g')
    h(n - 1)
    print(n)

def f(n):
    print('Start f')
    g(n - 1)
    print(n)

f(4)
```

```
Start f
Start g
Start h
0.5
2
3
4
```

In [19]:

```
# Variable With a Local Scope

def f(b):           # f has global scope, b has local scope
    a = 6           # This a has scope local to function call f()
    return a * b    # This a is the local a

a = 0               # This a has global scope
print('f(3) = {}'.format(f(3)))
print('a is {}'.format(a))        # Global a is still 0
```

```
f(3) = 18
a is 0
```

In [20]:

```
# Variable With a Global Scope

def f(b):           # f has global scope, b has local scope
    return a * b    # This a is the global a

a = 0               # This a has global scope
print('f(3) = {}'.format(f(3)))
print('a is {}'.format(a))        # Global a is still 0
```

```
f(3) = 0
a is 0
```

In [21]:

```python
# Modifying a Global Variable Inside a Function

def f(b):
    global a      # All references to a in f() are to the global a
    a = 6         # Global a is changed
    return a * b  # This a is the global a

a = 0             # This a has global scope
print('f(3) = {}'.format(f(3)))
print('a is {}'.format(a))        # Global a has been changed to 6
```

```
f(3) = 18
a is 6
```

In [22]:

```python
# Modifying a Global Variable Inside a Function

def f(b):
    global a      # All references to a in f() are to the global a
    a = 6         # Global a is changed
    return a * b  # This a is the global a

a = 0             # This a has global scope
print('a is {}'.format(a))
print('f(3) = {}'.format(f(3)))
print('a is {}'.format(a))        # Global a has been changed to 6
```

```
a is 0
f(3) = 18
a is 6
```

Exception Handling

In [23]:

```python
# Function Call Namespace

def h(n):
    print('Start h')
    print(1 / n)
    print(n)

def g(n):
    print('Start g')
    h(n - 1)
    print(n)

def f(n):
    print('Start f')
    g(n - 1)
    print(n)

f(2)
```

```
Start f
Start g
Start h
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-23-2824a6ec0033> in <module>
     16     print(n)
     17 
---> 18 f(2)

<ipython-input-23-2824a6ec0033> in f(n)
     13 def f(n):
     14     print('Start f')
---> 15     g(n - 1)
     16     print(n)
     17 

<ipython-input-23-2824a6ec0033> in g(n)
      8 def g(n):
      9     print('Start g')
---> 10     h(n - 1)
     11     print(n)
     12 

<ipython-input-23-2824a6ec0033> in h(n)
      3 def h(n):
      4     print('Start h')
----> 5     print(1 / n)
      6     print(n)
      7 

ZeroDivisionError: division by zero
```

In [24]:

```python
# Catching and Handling Exceptions

strAge = input('Enter your age: ')
intAge = int(strAge)
print('You are {} years old.'.format(intAge))
```

```
Enter your age: fifteen
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-24-047d04c7c26b> in <module>
      2 
      3 strAge = input('Enter your age: ')
----> 4 intAge = int(strAge)
      5 print('You are {} years old.'.format(intAge))

ValueError: invalid literal for int() with base 10: 'fifteen'
```

In [25]:

```python
# Catching and Handling Exceptions

while True:
    try:
        strAge = input('Enter your age: ')
        intAge = int(strAge)
        print('You are {} years old.'.format(intAge))
        break
    except:
        print('Enter your age using digits 0-9!')
```

```
Enter your age: s
Enter your age using digits 0-9!
Enter your age: 2
You are 2 years old.
```

In [26]:

```python
# Catching and Handling Exceptions

def readAge(filename):
    infile = open(filename)
    strAge = infile.readline()
    age = int(strAge)
    print('age is', age)
```

In [27]:

```python
readAge('age.txt')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-27-c7552c3d278a> in <module>
----> 1 readAge('age.txt')

<ipython-input-26-0ffefdef8912> in readAge(filename)
      4     infile = open(filename)
      5     strAge = infile.readline()
----> 6     age = int(strAge)
      7     print('age is', age)

ValueError: invalid literal for int() with base 10: 'fifteen\n'
```

In [28]:

```python
# Catching and Handling Exceptions

def readAge(filename):
    try:
        infile = open(filename)
        strAge = infile.readline()
        age = int(strAge)
        print('age is', age)
    except ValueError:
        print('Value cannot be converted to integer.')
```

In [29]:

```python
readAge('age.txt')
```

```
Value cannot be converted to integer.
```

In [30]:

```python
readAge('age.text')
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-30-04b68196c18e> in <module>
----> 1 readAge('age.text')

<ipython-input-28-ca34d15c9aea> in readAge(filename)
      3 def readAge(filename):
      4     try:
----> 5         infile = open(filename)
      6         strAge = infile.readline()
      7         age = int(strAge)

FileNotFoundError: [Errno 2] No such file or directory: 'age.text'
```

In [31]:

```python
def readAge(filename):
    try:
        infile = open(filename)
        strAge = infile.readline()
        age = int(strAge)
        print('age is', age)
    except FileNotFoundError:
        # Executed Only If an IOError Exception Is Raised
        print('Input/Output error.')
    except ValueError:
        # Executed Only If a ValueError Exception Is Raised
        print('Value cannot be converted to integer.')
    except:
        # Executed If an Exception Other Than IOError or ValueError Is Raised
        print('Other error.')
```

In [32]:

```python
readAge('age.txt')
```

```
Value cannot be converted to integer.
```

In [33]:

```python
readAge('age.text')
```

```
Input/Output error.
```

In [34]:

```python
def h(n):
    try:
        print('Start h')
        print(1 / n)
        print(n)
    except:
        print('!!!')

def g(n):
    print('Start g')
    h(n - 1)
    print(n)

def f(n):
    print('Start f')
    g(n - 1)
    print(n)
```

In [35]:

```python
f(5)
```

```
Start f
Start g
Start h
0.3333333333333333
3
4
5
```

In [36]:

```python
f(2)
```

```
Start f
Start g
Start h
!!!
1
2
```

In [37]:

```python
def h(n):
    print('Start h')
    print(1 / n)
    print(n)

def g(n):
    print('Start g')
    h(n - 1)
    print(n)

def f(n):
    print('Start f')
    g(n - 1)
    print(n)
```

In [38]:

```python
f(2)
```

```
Start f
Start g
Start h
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-38-c510dc86724b> in <module>
----> 1 f(2)

<ipython-input-37-5105d77856d4> in f(n)
     11 def f(n):
     12     print('Start f')
---> 13     g(n - 1)
     14     print(n)

<ipython-input-37-5105d77856d4> in g(n)
      6 def g(n):
      7     print('Start g')
----> 8     h(n - 1)
      9     print(n)
     10 

<ipython-input-37-5105d77856d4> in h(n)
      1 def h(n):
      2     print('Start h')
----> 3     print(1 / n)
      4     print(n)
      5 

ZeroDivisionError: division by zero
```

In [39]:

```python
try:
    f(2)
except:
    print('!!!')
```

```
Start f
Start g
Start h
!!!
```

Modules, Revisited

In [40]:

```python
import example
```

In [41]:

```python
dir(example)
```

Out[41]:

```
['__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 'f',
 'g',
 'sys',
 'x']
```

In [42]:

```python
# Import "example.py" as a module

import example
```

In [ ]:

```python
# You May Have an Error because Python Couldn't Find an "example.py" In the Python Sear
```

In [ ]:

```python
# Python Search Path

import sys
sys.path
```

In [45]:

```python
sys.path.append('new_directory')
```

In [ ]:

```python
sys.path
```

In [47]:

```python
import example
```

In [48]:

```python
# Is It Working?
# What Are the Attributes (e.g. Functions, etc.) of the Module?

dir(example)
```

Out[48]:

```
['__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 'f',
 'g',
 'sys',
 'x']
```

In [49]:

```
import random
```

In [50]:

```
dir(random)
```

Out[50]:

```
['BPF',
 'LOG4',
 'NV_MAGICCONST',
 'RECIP_BPF',
 'Random',
 'SG_MAGICCONST',
 'SystemRandom',
 'TWOPI',
 '_Sequence',
 '_Set',
 '__all__',
 '__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_accumulate',
 '_acos',
 '_bisect',
 '_ceil',
 '_cos',
 '_e',
 '_exp',
 '_inst',
 '_log',
 '_os',
 '_pi',
 '_random',
 '_repeat',
 '_sha512',
 '_sin',
 '_sqrt',
 '_test',
 '_test_generator',
 '_urandom',
 '_warn',
 'betavariate',
 'choice',
 'choices',
 'expovariate',
 'gammavariate',
 'gauss',
 'getrandbits',
 'getstate',
 'lognormvariate',
 'normalvariate',
 'paretovariate',
 'randint',
 'random',
 'randrange',
 'sample',
 'seed',
 'setstate',
 'shuffle',
 'triangular',
 'uniform',
 'vonmisesvariate',
 'weibullvariate']
```

In [ ]:

```
# Attributes of a Module:
# The Names (e.g. Functions, Values, and Classes) Defined In the Module Are Called the
```

In [51]:

```
print(example.f)
print(example.x)
print(dir(example))
```

```
<function f at 0x000001D8CD7E9700>
100
['__builtins__', '__cached__', '__doc__', '__file__', '__loader__', '__name__', '__packa
ge__', '__spec__', 'f', 'g', 'sys', 'x']
```

In [52]:

```
import math
dir(math)
```

Out[52]:

```
['__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 'acos',
 'acosh',
 'asin',
 'asinh',
 'atan',
 'atan2',
 'atanh',
 'ceil',
 'comb',
 'copysign',
 'cos',
 'cosh',
 'degrees',
 'dist',
 'e',
 'erf',
 'erfc',
 'exp',
 'expm1',
 'fabs',
 'factorial',
 'floor',
 'fmod',
 'frexp',
 'fsum',
 'gamma',
 'gcd',
 'hypot',
 'inf',
 'isclose',
 'isfinite',
 'isinf',
 'isnan',
 'isqrt',
 'ldexp',
 'lgamma',
 'log',
 'log10',
 'log1p',
 'log2',
 'modf',
 'nan',
 'perm',
 'pi',
 'pow',
 'prod',
 'radians',
 'remainder',
 'sin',
 'sinh',
 'sqrt',
 'tan',
 'tanh',
 'tau',
 'trunc']
```

In [53]:

```
dir()
```

Out[53]:

```
['In',
 'Out',
 '_',
 '_1',
 '_13',
 '_15',
 '_16',
 '_3',
 '_4',
 '_41',
 '_43',
 '_44',
 '_46',
 '_48',
 '_5',
 '_50',
 '_52',
 '_7',
 '_9',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i10',
 '_i11',
 '_i12',
 '_i13',
 '_i14',
 '_i15',
 '_i16',
 '_i17',
 '_i18',
 '_i19',
 '_i2',
 '_i20',
 '_i21',
 '_i22',
 '_i23',
 '_i24',
 '_i25',
 '_i26',
 '_i27',
 '_i28',
 '_i29',
 '_i3',
 '_i30',
 '_i31',
 '_i32',
 '_i33',
 '_i34',
 '_i35',
 '_i36',
 '_i37',
 '_i38',
 '_i39',
 '_i4',
 '_i40',
 '_i41',
 '_i42',
 '_i43',
 '_i44',
 '_i45',
 '_i46',
 '_i47',
 '_i48',
 '_i49',
 '_i5',
 '_i50',
 '_i51',
 '_i52',
 '_i53',
 '_i6',
 '_i7',
 '_i8',
 '_i9',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'a',
 'double',
 'example',
 'exit',
 'f',
 'g',
 'get_ipython',
 'h',
 'intAge',
 'math',
 'quit',
 'random',
 'readAge',
 'res',
 'strAge',
 'sys',
 'x',
 'y']
```

In [54]:

```
clemson = 'tigers'
```

In [55]:

```
dir()
```

Out[55]:

```
['In',
 'Out',
 '_',
 '_1',
 '_13',
 '_15',
 '_16',
 '_3',
 '_4',
 '_41',
 '_43',
 '_44',
 '_46',
 '_48',
 '_5',
 '_50',
 '_52',
 '_53',
 '_7',
 '_9',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i10',
 '_i11',
 '_i12',
 '_i13',
 '_i14',
 '_i15',
 '_i16',
 '_i17',
 '_i18',
 '_i19',
 '_i2',
 '_i20',
 '_i21',
 '_i22',
 '_i23',
 '_i24',
 '_i25',
 '_i26',
 '_i27',
 '_i28',
 '_i29',
 '_i3',
 '_i30',
 '_i31',
 '_i32',
 '_i33',
 '_i34',
 '_i35',
 '_i36',
 '_i37',
 '_i38',
 '_i39',
 '_i4',
 '_i40',
 '_i41',
 '_i42',
 '_i43',
 '_i44',
 '_i45',
 '_i46',
 '_i47',
 '_i48',
 '_i49',
 '_i5',
 '_i50',
 '_i51',
 '_i52',
 '_i53',
 '_i54',
 '_i55',
 '_i6',
 '_i7',
 '_i8',
 '_i9',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'a',
 'clemson',
 'double',
 'example',
 'exit',
 'f',
 'g',
 'get_ipython',
 'h',
 'intAge',
 'math',
 'quit',
 'random',
 'readAge',
 'res',
 'strAge',
 'sys',
 'x',
 'y']
```

In [ ]:

```
'''
Find the random module in one of the directories listed in sys.path, open it,
and find the implementations of functions randrange(), random(), and sample().
```

```
Then, import the module into the interpreter shell and view its attribute.
using the dir() function
'''
```

In [ ]:

```
sys.path
```

In [ ]:

```
'''
Bookkeeping Attributes / Special Variables (e.g. __doc__, __file__, __name__, __package_

These names exist for every imported module.
There names are defined by the Python interpreter at import time.
and are kept by the Python interpreter for bookkeeping purposes.

__name__: the name of the module
__file__: the absolute pathname of the file containing the module
__doc__: the module docstring
'''
```

In [58]:

```
import random
```

In [59]:

```
random.__name__
```

Out[59]:

```
'random'
```

In [ ]:

```
random.__file__
```

In [61]:

```
dir(random)
```

Out[61]:

```
['BPF',
 'LOG4',
 'NV_MAGICCONST',
 'RECIP_BPF',
 'Random',
 'SG_MAGICCONST',
 'SystemRandom',
 'TWOPI',
 '_Sequence',
 '_Set',
 '__all__',
 '__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_accumulate',
 '_acos',
 '_bisect',
 '_ceil',
 '_cos',
 '_e',
 '_exp',
 '_inst',
 '_log',
 '_os',
 '_pi',
 '_random',
 '_repeat',
 '_sha512',
 '_sin',
 '_sqrt',
 '_test',
 '_test_generator',
 '_urandom',
 '_warn',
 'betavariate',
 'choice',
 'choices',
 'expovariate',
 'gammavariate',
 'gauss',
 'getrandbits',
 'getstate',
 'lognormvariate',
 'normalvariate',
 'paretovariate',
 'randint',
 'random',
 'randrange',
 'sample',
 'seed',
 'setstate',
 'shuffle',
 'triangular',
 'uniform',
 'vonmisesvariate',
 'weibullvariate']
```

In [62]:

```
dir()
```

Out[62]:

```
['In',
 'Out',
 '_',
 '_1',
 '_13',
 '_15',
 '_16',
 '_3',
 '_4',
 '_41',
 '_43',
 '_44',
 '_46',
 '_48',
 '_5',
 '_50',
 '_52',
 '_53',
 '_55',
 '_56',
 '_57',
 '_59',
 '_60',
 '_61',
 '_7',
 '_9',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i10',
 '_i11',
 '_i12',
 '_i13',
 '_i14',
 '_i15',
 '_i16',
 '_i17',
 '_i18',
 '_i19',
 '_i2',
 '_i20',
 '_i21',
 '_i22',
 '_i23',
 '_i24',
 '_i25',
 '_i26',
 '_i27',
 '_i28',
 '_i29',
 '_i3',
 '_i30',
 '_i31',
 '_i32',
 '_i33',
 '_i34',
 '_i35',
 '_i36',
 '_i37',
 '_i38',
 '_i39',
 '_i4',
 '_i40',
 '_i41',
 '_i42',
 '_i43',
 '_i44',
 '_i45',
 '_i46',
 '_i47',
 '_i48',
 '_i49',
 '_i5',
 '_i50',
 '_i51',
 '_i52',
 '_i53',
 '_i54',
 '_i55',
 '_i56',
 '_i57',
 '_i58',
 '_i59',
 '_i6',
 '_i60',
 '_i61',
 '_i62',
 '_i7',
 '_i8',
 '_i9',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'a',
 'clemson',
 'double',
 'example',
 'exit',
 'f',
 'g',
 'get_ipython',
 'h',
 'intAge',
 'math',
 'quit',
 'random',
 'readAge',
 'res',
 'strAge',
 'sys',
 'x',
 'y']
```

In [63]:

```
dir(__builtins__)
```

Out[63]:

```
['ArithmeticError',
 'AssertionError',
 'AttributeError',
 'BaseException',
 'BlockingIOError',
 'BrokenPipeError',
 'BufferError',
 'BytesWarning',
 'ChildProcessError',
 'ConnectionAbortedError',
 'ConnectionError',
 'ConnectionRefusedError',
 'ConnectionResetError',
 'DeprecationWarning',
 'EOFError',
 'Ellipsis',
 'EnvironmentError',
 'Exception',
 'False',
 'FileExistsError',
 'FileNotFoundError',
 'FloatingPointError',
 'FutureWarning',
 'GeneratorExit',
 'IOError',
 'ImportError',
 'ImportWarning',
 'IndentationError',
 'IndexError',
 'InterruptedError',
 'IsADirectoryError',
 'KeyError',
 'KeyboardInterrupt',
 'LookupError',
 'MemoryError',
 'ModuleNotFoundError',
 'NameError',
 'None',
 'NotADirectoryError',
 'NotImplemented',
 'NotImplementedError',
 'OSError',
 'OverflowError',
 'PendingDeprecationWarning',
 'PermissionError',
 'ProcessLookupError',
 'RecursionError',
 'ReferenceError',
 'ResourceWarning',
 'RuntimeError',
 'RuntimeWarning',
 'StopAsyncIteration',
 'StopIteration',
 'SyntaxError',
 'SyntaxWarning',
 'SystemError',
 'SystemExit',
 'TabError',
 'TimeoutError',
 'True',
 'TypeError',
 'UnboundLocalError',
 'UnicodeDecodeError',
 'UnicodeEncodeError',
 'UnicodeError',
 'UnicodeTranslateError',
 'UnicodeWarning',
 'UserWarning',
 'ValueError',
 'Warning',
 'WindowsError',
 'ZeroDivisionError',
 '__IPYTHON__',
 '__build_class__',
 '__debug__',
 '__doc__',
 '__import__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 'abs',
 'all',
 'any',
 'ascii',
 'bin',
 'bool',
 'breakpoint',
 'bytearray',
 'bytes',
 'callable',
 'chr',
 'classmethod',
 'compile',
 'complex',
 'copyright',
 'credits',
 'delattr',
 'dict',
 'dir',
 'display',
 'divmod',
 'enumerate',
 'eval',
 'exec',
 'filter',
 'float',
 'format',
 'frozenset',
 'get_ipython',
 'getattr',
 'globals',
 'hasattr',
 'hash',
 'help',
 'hex',
 'id',
 'input',
 'int',
 'isinstance',
 'issubclass',
 'iter',
 'len',
 'license',
 'list',
 'locals',
 'map',
 'max',
 'memoryview',
 'min',
 'next',
 'object',
 'oct',
 'open',
 'ord',
 'pow',
 'print',
 'property',
 'range',
 'repr',
 'reversed',
 'round',
 'set',
 'setattr',
 'slice',
 'sorted',
 'staticmethod',
 'str',
 'sum',
 'super',
 'tuple',
 'type',
 'vars',
 'zip']
```

Top-Level Module

In [ ]:

```
'''
Bookkeeping Attributes / Special Variables (e.g. __doc__, __file__, __name__, __packag

These names exist for every imported module.
There names are defined by the Python interpreter at import time
and are kept by the Python interpreter for bookkeeping purposes.

__name__: the name of the module
__file__: the absolute pathname of the file containing the module
__doc__: the module docstring
'''
```

In [ ]:

```
'''
1) If the module is being run as a top-level module, attribute __name__ is set to the s

2) If the file is being imported by another module, whether the top-level or other,
attribute __name__ is set to the module's name.
'''
```

In [64]:

```
# Let's Review the Code In "name.py".

!type name.py

# Windows users:
# !type name.py
```

```
print('My name is {}'.format(__name__))
```

In [65]:

```
# Let's Run "name.py" at the Command Line
# Now, the Module "name.py" Is the Top-Level Module

!python name.py
```

```
My name is __main__
```

In [66]:

```
# Let's Import "name.py"
# The Shell Is the Top-Level Program That Imports the Module "name.py"

import name
```

```
My name is name
```

In [67]:

```
# Let's Review the Code In "name_test.py"

!type name_test.py
```

```
if __name__ == '__main__':
        print('Now, I am the top-level module')
        print('Please import name_test to use grade function')

def grade(score):
        if score >= 90:
                print('Grade: A')
        elif score >= 80:
                print('Grade: B')
        elif score >= 70:
                print('Grade: C')
        elif score >= 60:
                print('Grade: D')
        else:
                print('Fail')
```

In [68]:

```
# Let's Run "name_test.py" at the Command Line
# Now, the Module "name_test.py" Is the Top-Level Module

!python name_test.py
```

```
Now, I am the top-level module
Please import name_test to use grade function
```

In [69]:

```
# Let's Import "name_test.py"
# The Shell Is the Top-Level Program That Imports the Module "name_test.py"

import name_test
```

In [70]:

```
grade(95)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-70-4490c6b69c3c> in <module>
----> 1 grade(95)

NameError: name 'grade' is not defined
```

In [71]:

```
name_test.grade(95)
```

```
Grade: A
```

In [72]:

```
name_test.grade(80)
```

```
Grade: B
```

In [73]:

```
name_test.grade(75)
```

```
Grade: C
```

In [74]:

```
name_test.grade(60)
```

```
Grade: D
```

There Are Three Ways to Import Module Attributes

In [75]:

```
!type example.py
```

```
'an example module'
import sys
```

```
def f():
    'function f'
    print('Executing f() from', __name__)

def g():
    'function g'
    print('Executing g() from', __name__)

x = 100  # global var
#print('example here')
```

In [76]:

```
# Three Ways to Import Module Attributes

# 1) Import the (Name of the) Module
import example

example.x
```

Out[76]:

```
100
```

In [77]:

```
example.f()
```

```
Executing f() from example
```

In [78]:

```
example.g()
```

```
Executing g() from example
```

In [79]:

```
dir(example)
```

Out[79]:

```
['__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 'f',
 'g',
 'sys',
 'x']
```

In [80]:

```
dir()
```

Out[80]:

```
['In',
 'Out',
 '_',
 '_1',
 '_13',
 '_15',
 '_16',
 '_3',
 '_4',
 '_41',
 '_43',
 '_44',
 '_46',
 '_48',
 '_5',
 '_50',
 '_52',
 '_53',
 '_55',
 '_56',
 '_57',
 '_59',
 '_60',
 '_61',
 '_62',
 '_63',
 '_7',
 '_76',
 '_79',
 '_9',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_exit_code',
 '_i',
 '_i1',
 '_i10',
 '_i11',
 '_i12',
 '_i13',
 '_i14',
 '_i15',
 '_i16',
 '_i17',
 '_i18',
 '_i19',
 '_i2',
 '_i20',
 '_i21',
 '_i22',
 '_i23',
 '_i24',
 '_i25',
 '_i26',
 '_i27',
 '_i28',
 '_i29',
 '_i3',
 '_i30',
 '_i31',
 '_i32',
 '_i33',
 '_i34',
 '_i35',
 '_i36',
 '_i37',
 '_i38',
 '_i39',
 '_i4',
 '_i40',
'_i41',
'_i42',
'_i43',
'_i44',
'_i45',
'_i46',
'_i47',
'_i48',
'_i49',
'_i5',
'_i50',
'_i51',
'_i52',
'_i53',
'_i54',
'_i55',
'_i56',
'_i57',
'_i58',
'_i59',
'_i6',
'_i60',
'_i61',
'_i62',
'_i63',
'_i64',
'_i65',
'_i66',
'_i67',
'_i68',
'_i69',
'_i7',
'_i70',
'_i71',
'_i72',
'_i73',
'_i74',
'_i75',
'_i76',
'_i77',
'_i78',
'_i79',
'_i8',
'_i80',
'_i9',
'_ih',
'_ii',
'_iii',
'_oh',
'a',
'clemson',
'double',
'example',
'exit',
'f',
'g',
'get_ipython',
'h',
'intAge',
'math',
'name',
'name_test',
'quit',
'random',
'readAge',
'res',
'strAge',
'sys',
'x',
'y']
```

In [ ]:

```
# 2) Import the Specific module attributes
```

In [81]:

```
# Restart the Kernel!
# Then, Import a Specific Module Attribute (e.g. "f") From the Module

from example import f
```

In [82]:

```
f()
```

```
Executing f() from example
```

In [83]:

```
g()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-83-5fd69ddb5074> in <module>
----> 1 g()

TypeError: g() missing 1 required positional argument: 'n'
```

In [84]:

```
x
```

Out[84]:

```
20
```

In [85]:

```
dir()
```

Out[85]:

```
['In',
 'Out',
 '_',
 '_1',
 '_13',
 '_15',
 '_16',
 '_3',
 '_4',
 '_41',
 '_43',
 '_44',
 '_46',
 '_48',
 '_5',
 '_50',
 '_52',
 '_53',
 '_55',
 '_56',
 '_57',
 '_59',
 '_60',
 '_61',
 '_62',
 '_63',
 '_7',
 '_76',
 '_79',
 '_80',
 '_84',
 '_9',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_exit_code',
 '_i',
 '_i1',
 '_i10',
 '_i11',
 '_i12',
 '_i13',
 '_i14',
 '_i15',
 '_i16',
 '_i17',
 '_i18',
 '_i19',
 '_i2',
 '_i20',
 '_i21',
 '_i22',
 '_i23',
 '_i24',
 '_i25',
 '_i26',
 '_i27',
 '_i28',
 '_i29',
 '_i3',
 '_i30',
 '_i31',
 '_i32',
 '_i33',
 '_i34',
 '_i35',
 '_i36',
 '_i37',
 '_i38',
 '_i39',
 '_i4',
 '_i40',
 '_i41',
 '_i42',
 '_i43',
 '_i44',
 '_i45',
 '_i46',
 '_i47',
 '_i48',
 '_i49',
 '_i5',
'_i50',
'_i51',
'_i52',
'_i53',
'_i54',
'_i55',
'_i56',
'_i57',
'_i58',
'_i59',
'_i6',
'_i60',
'_i61',
'_i62',
'_i63',
'_i64',
'_i65',
'_i66',
'_i67',
'_i68',
'_i69',
'_i7',
'_i70',
'_i71',
'_i72',
'_i73',
'_i74',
'_i75',
'_i76',
'_i77',
'_i78',
'_i79',
'_i8',
'_i80',
'_i81',
'_i82',
'_i83',
'_i84',
'_i85',
'_i9',
'_ih',
'_ii',
'_iii',
'_oh',
'a',
'clemson',
'double',
'example',
'exit',
'f',
'g',
'get_ipython',
'h',
'intAge',
'math',
'name',
'name_test',
'quit',
'random',
'readAge',
'res',
'strAge',
'sys',
'x',
'y']
```

In [86]:

```
example.g()
```

```
Executing g() from example
```

In [87]:

```
example.x
```

Out[87]:

```
100
```

In [88]:

```
dir(example)
```

Out[88]:

```
['__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 'f',
 'g',
 'sys',
 'x']
```

In [89]:

```
# Restart the Kernel!
# 3) Then, Import All Module Attributes

from example import *
```

In [90]:

```
f()
```

```
Executing f() from example
```

In [91]:

```
g()
```

```
Executing g() from example
```

In [92]:

```
x
```

Out[92]:

```
100
```

In [93]:

```
dir()
```

Out[93]:

```
['In',
 'Out',
 '_',
 '_1',
 '_13',
 '_15',
 '_16',
 '_3',
 '_4',
 '_41',
 '_43',
 '_44',
 '_46',
 '_48',
 '_5',
 '_50',
 '_52',
 '_53',
 '_55',
 '_56',
 '_57',
 '_59',
 '_60',
 '_61',
 '_62',
 '_63',
 '_7',
 '_76',
 '_79',
 '_80',
 '_84',
 '_85',
 '_87',
 '_88',
 '_9',
 '_92',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_exit_code',
 '_i',
 '_i1',
 '_i10',
 '_i11',
 '_i12',
 '_i13',
 '_i14',
 '_i15',
 '_i16',
 '_i17',
 '_i18',
 '_i19',
 '_i2',
 '_i20',
 '_i21',
 '_i22',
 '_i23',
 '_i24',
 '_i25',
 '_i26',
 '_i27',
 '_i28',
 '_i29',
 '_i3',
 '_i30',
 '_i31',
 '_i32',
 '_i33',
 '_i34',
 '_i35',
 '_i36',
 '_i37',
 '_i38',
 '_i39',
 '_i4',
 '_i40',
 '_i41',
 '_i42',
 '_i43',
 '_i44',
 '_i45',
 '_i46',
 '_i47',
 '_i48',
 '_i49',
 '_i5',
 '_i50',
 '_i51',
 '_i52',
 '_i53',
 '_i54',
 '_i55',
 '_i56',
 '_i57',
 '_i58',
 '_i59',
 '_i6',
 '_i60',
 '_i61',
 '_i62',
 '_i63',
 '_i64',
 '_i65',
 '_i66',
 '_i67',
 '_i68',
 '_i69',
 '_i7',
 '_i70',
 '_i71',
 '_i72',
 '_i73',
 '_i74',
 '_i75',
 '_i76',
 '_i77',
 '_i78',
 '_i79',
 '_i8',
 '_i80',
 '_i81',
 '_i82',
 '_i83',
 '_i84',
 '_i85',
 '_i86',
 '_i87',
 '_i88',
 '_i89',
 '_i9',
 '_i90',
 '_i91',
 '_i92',
 '_i93',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'a',
 'clemson',
 'double',
 'example',
 'exit',
 'f',
 'g',
 'get_ipython',
 'h',
 'intAge',
 'math',
 'name',
 'name_test',
 'quit',
 'random',
 'readAge',
 'res',
 'strAge',
 'sys',
 'x',
 'y']
```

In [94]:

```python
import random
random.randrange(10)
```

Out[94]:

```
1
```

In [1]:

```python
# Restart the Kernel!
from random import randrange
randrange(10)
lst = [1, 2]
sample(lst, 1)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-1-e6df343f9dde> in <module>
      3 randrange(10)
      4 lst = [1, 2]
----> 5 sample(lst, 1)

NameError: name 'sample' is not defined
```

In [2]:

```python
# Restart the Kernel!
from random import *
randrange(10)
lst = [1, 2]
sample(lst, 1)
```

Out[2]:

```
[2]
```

A Class as a Namespace

A class (e.g. list, dict, tuple, etc.) is a code template for creating objects.

In Python, a namespace is associated with every class.

The name of the namespace is the name of the class and the names stored in the namespace are the class attributes (e.g., the class methods).

In [3]:

```
dir()
```

Out[3]:

```
['In',
 'Out',
 'Random',
 'SystemRandom',
 '_',
 '_2',
 '__',
 '___',
 '__builtin__',
 '__builtins__',
 '__doc__',
 '__loader__',
 '__name__',
 '__package__',
 '__spec__',
 '_dh',
 '_i',
 '_i1',
 '_i2',
 '_i3',
 '_ih',
 '_ii',
 '_iii',
 '_oh',
 'betavariate',
 'choice',
 'choices',
 'exit',
 'expovariate',
 'gammavariate',
 'gauss',
 'get_ipython',
 'getrandbits',
 'getstate',
 'lognormvariate',
 'lst',
 'normalvariate',
 'paretovariate',
 'quit',
 'randint',
 'random',
 'randrange',
 'sample',
 'seed',
 'setstate',
 'shuffle',
 'triangular',
 'uniform',
 'vonmisesvariate',
 'weibullvariate']
```

In [4]:

```
dir(list)
```

Out[4]:

```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__iadd__',
 '__imul__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__reversed__',
 '__rmul__',
 '__setattr__',
 '__setitem__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'append',
 'clear',
 'copy',
 'count',
 'extend',
 'index',
 'insert',
 'pop',
 'remove',
 'reverse',
 'sort']
```

In [5]:

```
lst = []
```

In [6]:

```
type(lst)
```

Out[6]:

```
list
```

In [7]:

```
lst_1 = []
```

In [8]:

```
lst_2 = []
```

In [9]:

```
dir(lst)
```

Out[9]:

```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__iadd__',
 '__imul__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__reversed__',
 '__rmul__',
 '__setattr__',
 '__setitem__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'append',
 'clear',
 'copy',
 'count',
 'extend',
 'index',
 'insert',
 'pop',
 'remove',
 'reverse',
 'sort']
```

In [10]:

```
dir(lst_2)
```

Out[10]:

```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__iadd__',
 '__imul__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__reversed__',
 '__rmul__',
 '__setattr__',
 '__setitem__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'append',
 'clear',
 'copy',
 'count',
 'extend',
 'index',
 'insert',
 'pop',
 'remove',
 'reverse',
 'sort']
```

In [11]:

```
dir(lst_1)
```

Out[11]:

```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__iadd__',
 '__imul__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__reversed__',
 '__rmul__',
 '__setattr__',
 '__setitem__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'append',
 'clear',
 'copy',
 'count',
 'extend',
 'index',
 'insert',
 'pop',
 'remove',
 'reverse',
 'sort']
```

In [12]:

```
# Let's Think About List Methods 'pop, sort'

lst = ['pear', 'apple', 'strawberry']
lst.pop(2)
lst
```

Out[12]:

```
['pear', 'apple']
```

In [13]:

```
lst.sort()
lst
```

Out[13]:

```
['apple', 'pear']
```

In [14]:

```
import math
print(math.sqrt)

print(list.pop)
print(list.sort)
dir(list)

# Methods Are AttributeS of the List Class
```

```
<built-in function sqrt>
<method 'pop' of 'list' objects>
<method 'sort' of 'list' objects>
```

Out[14]:

```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__iadd__',
 '__imul__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__reversed__',
 '__rmul__',
 '__setattr__',
 '__setitem__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'append',
 'clear',
 'copy',
 'count',
 'extend',
 'index',
 'insert',
 'pop',
 'remove',
 'reverse',
 'sort']
```

In [15]:

```
dir(dict)
```

Out[15]:

```
['__class__',
 '__contains__',
 '__delattr__',
 '__delitem__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__reversed__',
 '__setattr__',
 '__setitem__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'clear',
 'copy',
 'fromkeys',
 'get',
 'items',
 'keys',
 'pop',
 'popitem',
 'setdefault',
 'update',
 'values']
```

In [16]:

```
dir(tuple)
```

Out[16]:

```
['__add__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__getnewargs__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__rmul__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'count',
 'index']
```

Method Invocations

Rewrite the below Python statement so that instead of making the usual method invocations:

instance.method(arg1, arg2, ...),

you use the notation:

class.method(instance, arg1, arg2, ...)

In [17]:

```
s = 'hello'
print(s.upper())
```

```
HELLO
```

In [18]:

```
s = 'hello'
print(str.upper(s))
```

```
HELLO
```

In [19]:

```
lst = [9, 1, 8, 2, 7, 3]

print(lst)
lst.sort()
print(lst)
```

```
[9, 1, 8, 2, 7, 3]
[1, 2, 3, 7, 8, 9]
```

In [20]:

```
lst = [9, 1, 8, 2, 7, 3]

print(lst)
list.sort(lst)
print(lst)
```

```
[9, 1, 8, 2, 7, 3]
[1, 2, 3, 7, 8, 9]
```

In [21]:

```
lst2 = [9, 1, 8, 2, 7, 3]
print(lst2)

lst2.append(6)
print(lst2)
```

```
[9, 1, 8, 2, 7, 3]
[9, 1, 8, 2, 7, 3, 6]
```

In [22]:

```
lst2 = [9, 1, 8, 2, 7, 3]
print(lst2)

list.append(lst2,6)
print(lst2)
```

```
[9, 1, 8, 2, 7, 3]
[9, 1, 8, 2, 7, 3, 6]
```

In [23]:

```
s = 'ACM'
```

In [24]:

```
s.lower()
```

Out[24]:

```
'acm'
```

In [25]:

```
s = 'ACM'
str.lower(s)
```

Out[25]:

```
'acm'
```

In [26]:

```
s.find('C')
```

Out[26]:

```
1
```

In [27]:

```
str.find(s, 'C')
```

Out[27]:

```
1
```

In [28]:

```
s = 'ACM'
s.replace('AC', 'IB')
```

Out[28]:

```
'IBM'
```

In [29]:

```
s = 'ACM'
str.replace(s, 'AC', 'IB')
```

Out[29]:

```
'IBM'
```