Microsoft Windows [Version 6.3.9600]

(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\gadiwsri>python --version

Python 3.6.5

C:\Users\gadiwsri>python

Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> name = "ravi"

>>> dir(name)

['\_\_add\_\_', '\_\_class\_\_', '\_\_contains\_\_', '\_\_delattr\_\_', '\_\_dir\_\_', '\_\_doc\_\_', '\_\_eq\_\_', '\_\_format\_\_', '\_\_ge\_\_', '\_\_getattribute\_\_', '\_\_getitem\_\_', '\_\_getnewargs\_\_', '\_\_gt\_\_', '\_\_hash\_\_', '\_\_init\_\_', '\_\_init\_subclass\_\_', '\_\_iter\_\_', '\_\_le\_\_', '\_\_len\_\_', '\_\_lt\_\_', '\_\_mod\_\_', '\_\_mul\_\_', '\_\_ne\_\_', '\_\_ne

w\_\_', '\_\_reduce\_\_', '\_\_reduce\_ex\_\_', '\_\_repr\_\_', '\_\_rmod\_\_', '\_\_rmul\_\_', '\_\_setattr\_\_', '\_\_sizeof\_\_', '\_\_str\_\_', '\_\_subclasshook\_\_', 'capitalize', 'casefold', 'center', 'count', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format\_map', 'index', 'isalnum', 'isalpha', 'isdecimal', 'isdigit',

'isidentifier', 'islower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join', 'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startswith', 'strip', 'swapcase', 'title', 'translat

e', 'upper', 'zfill']

>>> id(name)

18949920

>>> help(name.split)

Help on built-in function split:

split(...) method of builtins.str instance

S.split(sep=None, maxsplit=-1) -> list of strings

Return a list of the words in S, using sep as the

delimiter string. If maxsplit is given, at most maxsplit

splits are done. If sep is not specified or is None, any

whitespace string is a separator and empty strings are

removed from the result.

>>> marks = [90, 34, 12, 56]

>>> dir(marks)

['\_\_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']

>>> marks

[90, 34, 12, 56]

>>> marks.append(99)

>>> marks

[90, 34, 12, 56, 99]

>>> marks.append(1,2)

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: append() takes exactly one argument (2 given)

>>> marks.append([1,2])

>>> marks

[90, 34, 12, 56, 99, [1, 2]]

>>> marks.extend(12)

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: 'int' object is not iterable

>>> for i in 10000:

... print(i)

...

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: 'int' object is not iterable

>>> for i in 10.45:

... print(i)

...

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: 'float' object is not iterable

>>> for i in True:

... print(i)

...

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: 'bool' object is not iterable

>>> for i in "hi":

... print(i)

...

h

i

>>> dir("hi")

['\_\_add\_\_', '\_\_class\_\_', '\_\_contains\_\_', '\_\_delattr\_\_', '\_\_dir\_\_', '\_\_doc\_\_', '\_\_eq\_\_', '\_\_format\_\_', '\_\_ge\_\_', '\_\_getattribute\_\_', '\_\_getitem\_\_', '\_\_getnewargs\_\_', '\_\_gt\_\_', '\_\_hash\_\_', '\_\_init\_\_', '\_\_init\_subclass\_\_', '\_\_iter\_\_', '\_\_le\_\_', '\_\_len\_\_', '\_\_lt\_\_', '\_\_mod\_\_', '\_\_mul\_\_', '\_\_ne\_\_', '\_\_ne

w\_\_', '\_\_reduce\_\_', '\_\_reduce\_ex\_\_', '\_\_repr\_\_', '\_\_rmod\_\_', '\_\_rmul\_\_', '\_\_setattr\_\_', '\_\_sizeof\_\_', '\_\_str\_\_', '\_\_subclasshook\_\_', 'capitalize', 'casefold', 'center', 'count', 'encode', 'endswith', 'expandtabs', 'find', 'format', 'format\_map', 'index', 'isalnum', 'isalpha', 'isdecimal', 'isdigit',

'isidentifier', 'islower', 'isnumeric', 'isprintable', 'isspace', 'istitle', 'isupper', 'join', 'ljust', 'lower', 'lstrip', 'maketrans', 'partition', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstrip', 'split', 'splitlines', 'startswith', 'strip', 'swapcase', 'title', 'translat

e', 'upper', 'zfill']

>>> l

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

NameError: name 'l' is not defined

>>> marks

[90, 34, 12, 56, 99, [1, 2]]

>>> marks.extend([90, 34])

>>> marks

[90, 34, 12, 56, 99, [1, 2], 90, 34]

>>> marks.extend([5, 78])

>>> marks

[90, 34, 12, 56, 99, [1, 2], 90, 34, 5, 78]

>>> marks.count(12)

1

>>> marks.count(34)

2

>>> marks.index(34)

1

>>> marks.index(34, 2)

7

>>> v = [9, 45, 12, 34]

>>> v.sort()

>>> v

[9, 12, 34, 45]

>>> v

[9, 12, 34, 45]

>>> k = [90, 45, 12, 34]

>>> j = k

>>> id(j)

50918984

>>> id(k)

50918984

>>> k[0]=100

>>> k

[100, 45, 12, 34]

>>> j

[100, 45, 12, 34]

>>> import copy

>>> a = [90, 45, 12, 34]

>>> b = copy.copy(a)

>>> b

[90, 45, 12, 34]

>>> a

[90, 45, 12, 34]

>>> id(a)

50919704

>>> id(b)

50919624

>>> a[0]=100

>>> a

[100, 45, 12, 34]

>>> b

[90, 45, 12, 34]

>>> k = [[90, 45, 12], [8, 4, 2]]

>>> k[0]

[90, 45, 12]

>>> k[0][0]

90

>>> m = copy.copy(k)

>>> id(m)

50919464

>>> id(k)

50919584

>>> id(k[0])

50919544

>>> id(m[0])

50919544

>>> k[0][0]=987

>>> k

[[987, 45, 12], [8, 4, 2]]

>>> m

[[987, 45, 12], [8, 4, 2]]

>>> a = [[9, 5, 1], [4, 11, 33]]

>>> b = copy.deepcopy(a)

>>> id(b)

50919704

>>> id(a)

50919384

>>> id(a[0])

50919424

>>> id(b[0])

50919304

>>> marks

[90, 34, 12, 56, 99, [1, 2], 90, 34, 5, 78]

>>> marks.insert(2, 987)

>>> marks

[90, 34, 987, 12, 56, 99, [1, 2], 90, 34, 5, 78]

>>> a = [90, 12, 34]

>>> b = [9, 4, 1]

>>> a+b

[90, 12, 34, 9, 4, 1]

>>> s = []

>>> for i in range(0, len(a)):

... s.append(a[i]+b[i])

...

>>> s

[99, 16, 35]

>>> w = [9, 12, 34, 54]

>>> w\_2 = [i+2 for i in w]

>>> w\_2

[11, 14, 36, 56]

>>>

Microsoft Windows [Version 6.3.9600]

(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\gadiwsri>python

Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> a = [9, 4, 12]

>>> b = [7, 3, 1]

>>> a+b

[9, 4, 12, 7, 3, 1]

>>> zip(a,b)

<zip object at 0x024F7800>

>>> list(zip(a,b))

[(9, 7), (4, 3), (12, 1)]

>>> s = [i+j for i,j in zip(a,b)]

>>> s

[16, 7, 13]

>>> words = ["hi", "how", "are", "you"]

>>> "".join(words)

'hihowareyou'

>>> "#".join(words)

'hi#how#are#you'

>>> "%".join(words)

'hi%how%are%you'

>>> marks = [90, 34, 12]

>>> "".join(marks)

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: sequence item 0: expected str instance, int found

>>> "".join([str(i) for i in marks])

'903412'

>>> "#".join([str(i) for i in marks])

'90#34#12'

>>> t = (9, 45, 12)

>>> type(t)

<class 'tuple'>

>>> k = 10, 20

>>> type(k)

<class 'tuple'>

>>> j = 10,

>>> type(j)

<class 'tuple'>

>>> l = 10

>>> type(l)

<class 'int'>

>>> t = (9, 45, 132, 33)

>>> t[0]

9

>>> t[0]=100

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

TypeError: 'tuple' object does not support item assignment

>>> marks = {"rakesh":100, "sandhya":99, "vinay":91, "bhaskar":77, "aditya":88, "hareesh":92, "ashish":81}

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 91, 'bhaskar': 77, 'aditya': 88, 'hareesh': 92, 'ashish': 81}

>>> marks["sandhya"]

99

>>> marks.keys()

dict\_keys(['rakesh', 'sandhya', 'vinay', 'bhaskar', 'aditya', 'hareesh', 'ashish'])

>>> marks.values()

dict\_values([100, 99, 91, 77, 88, 92, 81])

>>> marks.items()

dict\_items([('rakesh', 100), ('sandhya', 99), ('vinay', 91), ('bhaskar', 77), ('aditya', 88), ('hareesh', 92), ('ashish', 81)])

>>> marks["vinay"]=98

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 98, 'bhaskar': 77, 'aditya': 88, 'hareesh': 92, 'ashish': 81}

>>> marks["bhaskar"]=marks["bhskar"]+4

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

KeyError: 'bhskar'

>>> marks["bhaskar"]=marks["bhaskar"]+4

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 98, 'bhaskar': 81, 'aditya': 88, 'hareesh': 92, 'ashish': 81}

>>> marks["kamala"]

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

KeyError: 'kamala'

>>> quit()

C:\Users\gadiwsri>notepad ke\_demo.py

C:\Users\gadiwsri>python ke\_demo.py

Traceback (most recent call last):

File "ke\_demo.py", line 3, in <module>

print("Marks of kamala:", marks["kamala"])

KeyError: 'kamala'

C:\Users\gadiwsri>python ke\_demo.py

Marks of sandhya: 99

remaining code execution

C:\Users\gadiwsri>python ke\_demo.py

Traceback (most recent call last):

File "ke\_demo.py", line 3, in <module>

print("Marks of kamal:", marks["kamal"])

KeyError: 'kamal'

C:\Users\gadiwsri>python

Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> marks = {"rakesh":100, "sandhya":99, "vinay":91, "bhaskar":77, "aditya":88, "hareesh":92, "ashish":81}

>>> marks["kamal"]

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

KeyError: 'kamal'

>>> marks.get("kamala")

>>> print(marks.get("kamala"))

None

>>> marks.get("kamala", 35)

35

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 91, 'bhaskar': 77, 'aditya': 88, 'hareesh': 92, 'ashish': 81}

>>> "sandhya" in marks

True

>>> "sandhya" not in marks

False

>>> if "sandhya" in marks:

... print(marks["sandhya"])

... else:

... print(35)

...

99

>>> if "kamala" in marks:

... print(marks["kamala"])

... else:

... print(35)

...

35

>>> marks.get("kamala", 35)

35

>>> marks.get("sandhya", 35)

99

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 91, 'bhaskar': 77, 'aditya': 88, 'hareesh': 92, 'ashish': 81}

>>> marks.setdefault(

... "kamala", 35)

35

>>>

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 91, 'bhaskar': 77, 'aditya': 88, 'hareesh': 92, 'ashish': 81, 'kamala': 35}

>>> marks.setdefault("lakshmi", 45)

45

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 91, 'bhaskar': 77, 'aditya': 88, 'hareesh': 92, 'ashish': 81, 'kamala': 35, 'lakshmi': 45}

>>> marks.setdefault("chaitanya", 100)

100

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 91, 'bhaskar': 77, 'aditya': 88, 'hareesh': 92, 'ashish': 81, 'kamala': 35, 'lakshmi': 45, 'chaitanya': 100}

>>> marks.setdefault("sandhya", 35)

99

>>> marks

{'rakesh': 100, 'sandhya': 99, 'vinay': 91, 'bhaskar': 77, 'aditya': 88, 'hareesh': 92, 'ashish': 81, 'kamala': 35, 'lakshmi': 45, 'chaitanya': 100}

>>> quit()

C:\Users\gadiwsri>notepad count\_words.py

C:\Users\gadiwsri>python count\_words.py

Occurrence of words:

{'hi': 1, 'how': 1, 'are': 3, 'you': 3, 'where': 1, 'what': 1, 'doing': 1}

C:\Users\gadiwsri>python count\_words.py

Occurrence of words:

{'hi': 1, 'how': 1, 'are': 3, 'you': 3, 'where': 1, 'what': 1, 'doing': 1}

C:\Users\gadiwsri>python count\_words.py

C:\Users\gadiwsri>python

Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> l = [90, 45, 12, 34, 90]

>>> v = set(l)

>>> v

{90, 34, 12, 45}

>>> k = {34, 12, 34, 56, 778}

>>> k

{56, 34, 778, 12}

>>> k.add(12)

>>> k

{56, 34, 778, 12}

>>> k.remove(778)

>>> k

{56, 34, 12}

>>> type(k)

<class 'set'>

>>> dir(k)

['\_\_and\_\_', '\_\_class\_\_', '\_\_contains\_\_', '\_\_delattr\_\_', '\_\_dir\_\_', '\_\_doc\_\_', '\_\_eq\_\_', '\_\_format\_\_', '\_\_ge\_\_', '\_\_getattribute\_\_', '\_\_gt\_\_', '\_\_hash\_\_', '\_\_iand\_\_', '\_\_init\_\_', '\_\_init\_subclass\_\_', '\_\_ior\_\_', '\_\_isub\_\_', '\_\_iter\_\_', '\_\_ixor\_\_', '\_\_le\_\_', '\_\_len\_\_', '\_\_lt\_\_', '\_\_ne\_\_', '\_\_new\_\_', '\_

\_or\_\_', '\_\_rand\_\_', '\_\_reduce\_\_', '\_\_reduce\_ex\_\_', '\_\_repr\_\_', '\_\_ror\_\_', '\_\_rsub\_\_', '\_\_rxor\_\_', '\_\_setattr\_\_', '\_\_sizeof\_\_', '\_\_str\_\_', '\_\_sub\_\_', '\_\_subclasshook\_\_', '\_\_xor\_\_', 'add', 'clear', 'copy', 'difference', 'difference\_update', 'discard', 'intersection', 'intersection\_update', 'isdisjoint

', 'issubset', 'issuperset', 'pop', 'remove', 'symmetric\_difference', 'symmetric\_difference\_update', 'union', 'update']

>>> help(k.remove)

Help on built-in function remove:

remove(...) method of builtins.set instance

Remove an element from a set; it must be a member.

If the element is not a member, raise a KeyError.

>>> def print\_text():

... print("hi there")

...

>>> print\_text()

hi there

>>> print\_text()

hi there

>>> print\_text()

hi there

>>> def print\_val(x):

... print(x)

...

>>> print\_val(100)

100

>>> quit()

C:\Users\gadiwsri>notepad fundemo1.py

C:\Users\gadiwsri>python fundemo1.py

<class 'tuple'>

(10, 20, 3, 45, 22)

C:\Users\gadiwsri>notepad fundemo2.py

C:\Users\gadiwsri>copy fundemo1.py fundemo2.py

Overwrite fundemo2.py? (Yes/No/All): Yes

1 file(s) copied.

C:\Users\gadiwsri>notepad fundemo2.py

C:\Users\gadiwsri>python fundemo2.py

<class 'dict'>

{'a': 10, 'b': 20}

C:\Users\gadiwsri>copy fundemo1.py fundemo3.py

1 file(s) copied.

C:\Users\gadiwsri>notepad fundemo3.py

C:\Users\gadiwsri>python fundemo3.py

10 100

C:\Users\gadiwsri>python fundemo3.py

10 20

C:\Users\gadiwsri>copy fundemo1.py fundemo4.py

1 file(s) copied.

C:\Users\gadiwsri>notepad fundemo4.py

C:\Users\gadiwsri>notepad fundemo4.py

C:\Users\gadiwsri>python fundemo4.py

a: 10 b: 20 args: (3, 45, 22) kwargs: {'x': 9, 'y': 3}

C:\Users\gadiwsri>python fundemo4.py

File "fundemo4.py", line 1

def print\_vals(b=100,a, \*args, \*\*kwargs):

^

SyntaxError: non-default argument follows default argument

C:\Users\gadiwsri>python

Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> import random

>>> dir(random)

['BPF', 'LOG4', 'NV\_MAGICCONST', 'RECIP\_BPF', 'Random', 'SG\_MAGICCONST', 'SystemRandom', 'TWOPI', '\_BuiltinMethodType', '\_MethodType', '\_Sequence', '\_Set', '\_\_all\_\_', '\_\_builtins\_\_', '\_\_cached\_\_', '\_\_doc\_\_', '\_\_file\_\_', '\_\_loader\_\_', '\_\_name\_\_', '\_\_package\_\_', '\_\_spec\_\_', '\_acos', '\_bisect', '\_ceil'

, '\_cos', '\_e', '\_exp', '\_inst', '\_itertools', '\_log', '\_pi', '\_random', '\_sha512', '\_sin', '\_sqrt', '\_test', '\_test\_generator', '\_urandom', '\_warn', 'betavariate', 'choice', 'choices', 'expovariate', 'gammavariate', 'gauss', 'getrandbits', 'getstate', 'lognormvariate', 'normalvariate', 'paretovaria

te', 'randint', 'random', 'randrange', 'sample', 'seed', 'setstate', 'shuffle', 'triangular', 'uniform', 'vonmisesvariate', 'weibullvariate']

>>> help(random.randint)

Help on method randint in module random:

randint(a, b) method of random.Random instance

Return random integer in range [a, b], including both end points.

>>> quit()

C:\Users\gadiwsri>notepad arith.py

C:\Users\gadiwsri>python

Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> import arith

>>> arith.my\_add(1,2)

3

>>> arith.my\_diff(7,6)

1

>>> from arith import my\_add

>>> my\_add(9, 3)

12

>>> my\_diff(4,3)

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

NameError: name 'my\_diff' is not defined

>>> arith.my\_diff(9,4)

5

>>> from arith import my\_diff

>>> my\_diff(4,3)

1

>>> help(arith.my\_add)

Help on function my\_add in module arith:

my\_add(a, b)

>>> help(arith.my\_add)

Help on function my\_add in module arith:

my\_add(a, b)

>>> quit()

C:\Users\gadiwsri>python

Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> import arith

>>> help(arith.my\_add)

Help on function my\_add in module arith:

my\_add(a, b)

this function takes two arguments

and returns the sum of it

>>> quit()

C:\Users\gadiwsri>dir arith\*

Volume in drive C is Windows

Volume Serial Number is 38F5-5B11

Directory of C:\Users\gadiwsri

11-04-2018 18:39 167 arith.py

1 File(s) 167 bytes

0 Dir(s) 107,833,573,376 bytes free

C:\Users\gadiwsri>mkdir sample

C:\Users\gadiwsri>move arith.py sample\

1 file(s) moved.

C:\Users\gadiwsri>dir arith\*

Volume in drive C is Windows

Volume Serial Number is 38F5-5B11

Directory of C:\Users\gadiwsri

File Not Found

C:\Users\gadiwsri>python

Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 16:07:46) [MSC v.1900 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>> import arith

Traceback (most recent call last):

File "<stdin>", line 1, in <module>

ModuleNotFoundError: No module named 'arith'

>>> import sys

>>> sys.path

['', 'C:\\Python36\\python36.zip', 'C:\\Python36\\DLLs', 'C:\\Python36\\lib', 'C:\\Python36', 'C:\\Python36\\lib\\site-packages']

>>> sys.path.append("C:\\Users\\gadiwsri\\sample")

>>> sys.path

['', 'C:\\Python36\\python36.zip', 'C:\\Python36\\DLLs', 'C:\\Python36\\lib', 'C:\\Python36', 'C:\\Python36\\lib\\site-packages', 'C:\\Users\\gadiwsri\\sample']

>>> import arith

>>> arith.my\_add(9, 45)

54

>>> arith.my\_diff(89, 34)

55

>>> quit()

C:\Users\gadiwsri>notepad use\_arith.py

C:\Users\gadiwsri>python use\_arith.py

12

5

C:\Users\gadiwsri>type use\_arith.py

import sys

sys.path.append("C:\\Users\\gadiwsri\\sample")

import arith

print(arith.my\_add(9, 3))

print(arith.my\_diff(8, 3))

C:\Users\gadiwsri>type sample\\arith.py

def my\_add(a,b):

"""this function takes two arguments

and returns the sum of it"""

return a+b

def my\_diff(a,b):

if a>b:

return a-b

else:

return b-a

C:\Users\gadiwsri>python fundemo4.py

a: 10 args: (20, 3, 45, 22) kwargs: {'x': 9, 'y': 3}

C:\Users\gadiwsri>python fundemo4.py

a: 10 b: 20 args: (3, 45, 22) kwargs: {'x': 9, 'y': 3}

C:\Users\gadiwsri>