course_2_assessment_7

Due: 2018-11-25 01:33:00

Description: Assessment for Advanced Functions

Score: 0 of 6 = 0.0%

Questions

Not yet graded

Create a function called `mult` that has two parameters, the first is required and should be an integer, the second is an optional parameter that can either be a number or a string but whose default is 6. The function should return the first parameter multiplied by the second.

Save & Run

17/09/2022, 22:02:03 - 2 of 2

Original - 1 of 1

Show in CodeLens

```
1 def mult(req,op = 6):
2     return(req * op)
3
```

Activity: 15.7.1 ActiveCode (ac15_5_1)

Result	Actual Value	Expected Value	Notes
Pass	12	12	Testing that mult returns the correct value on input (2)
Pass	15	15	Testing that mult returns the correct value on input (3,5)
Pass	'hello...hello'	'hello...hello'	testing that mult returns the correct value on input (4, 'hello'

Expand Differences

You passed: 100.0% of the tests

Question in Context (/runestone/books/published/fopp/AdvancedFunctions/ChapterAssessment.html#ac15_5_1)

Not yet graded

The following function, `greeting` , does not work. Please fix the code so that it runs without error. This only requires one change in the definition of the function.

Save & Run 17/09/2022, 22:02:12 - 2 of 2 Show in CodeLens

```
1 def greeting(name,greeting="Hello ", excl="!"):
2     return greeting + name + excl
3
4 print(greeting("Bob"))
5 print(greeting(""))
6 print(greeting("Bob", excl="!!!"))
```

```
Hello Bob!
Hello !
Hello Bob!!!
```

Activity: 15.7.2 ActiveCode (ac15_5_2)

Result	Actual Value	Expected Value	Notes
Pass	'Hello Bob!'	'Hello Bob!'	Testing that greeting('Bob') returns 'Hello Bob!'.
Pass	'Hello !'	'Hello !'	Testing that greeting('') return 'Hello !'.

You passed: 100.0% of the tests

Question in Context (/runestone/books/published/fopp/AdvancedFunctions/ChapterAssessment.html#ac15_5_2)

Not yet graded

Below is a function, `sum` , that does not work. Change the function definition so the code works. The function should still have a required parameter, `intx` , and an optional parameter, `intz` with a defualt value of 5.

Save & Run 17/09/2022, 22:02:21 - 2 of 2 Show in CodeLens

```
1 def sum( intx, intz=5):
2     return intz + intx
```

Activity: 15.7.3 ActiveCode (ac15_5_3)

Result	Actual Value	Expected Value	Notes
Pass	10	10	Testing the function sum on inputs 8, 2.
Pass	17	17	Testing the function sum on input 12.

You passed: 100.0% of the tests

Question in Context (/runestone/books/published/fopp/AdvancedFunctions/ChapterAssessment.html#ac15_5_3)

Not yet graded

Write a function, `test` , that takes in three parameters: a required integer, an optional boolean whose default value is `True` , and an optional dictionary, called `dict1` , whose default value is `{2:3, 4:5, 6:8}` . If the boolean parameter is True, the function should test to see if the integer is a key in the dictionary. The value of that key should then be returned. If the boolean parameter is False, return the boolean value “False”.

Save & Run 17/09/2022, 22:02:36 - 3 of 3 17/09/2022, 22:02:35 - 2 of 2

Show in CodeLens

```
1 def test(intx,Bool = True , dict1 = {2:3,4:5,6:8}):
2     if Bool == True:
3         if intx in dict1:
4             return dict1.get(intx)
5     else:
6         return Bool
7 def test(intx,Bool = True , dict1 = {2:3,4:5,6:8}):
8     if Bool == True:
9         if intx in dict1:
```

```
            return dict1.get(intx)
    else:
        return Bool
```

Activity: 15.7.4 ActiveCode (ac15_5_4)

Result	Actual Value	Expected Value	Notes
Pass	3	3	Testing that test(2) returns 3
Pass	False	False	Testing that test(4, False) returns False
Pass	4	4	Testing that test(5, dict1 = {5:4, 2:1}) returns 4

You passed: 100.0% of the tests

Question in Context (/runestone/books/published/fopp/AdvancedFunctions/ChapterAssessment.html#ac15_5_4)

Not yet graded

Write a function called `checkingIfIn` that takes three parameters. The first is a required parameter, which should be a string. The second is an optional parameter called `direction` with a default value of `True` . The third is an optional parameter called `d` that has a default value of `{'apple': 2, 'pear': 1, 'fruit': 19, 'orange': 5, 'banana': 3, 'grapes': 2, 'watermelon': 7}` . Write the function `checkingIfIn` so that when the second parameter is `True` , it checks to see if the first parameter is a key in the third parameter; if it is, return `True` , otherwise return `False` .

But if the second paramter is `False` , then the function should check to see if the first parameter is *not* a key of the third. If it's *not*, the function should return `True` in this case, and if it is, it should return `False` .

Save & Run 17/09/2022, 22:02:39 - 2 of 2 Show in CodeLens

```
def checkingIfIn(Str, direction = True, d = {'apple': 2, 'pear': 1, '
    if direction == True:
        if Str in d:
            return True
        else:
            return False
    else:
        if Str not in d:
```

```
9             return True
10        else:
11            return False
12
```

Activity: 15.7.5 ActiveCode (ac15_5_5)

Result	Actual Value	Expected Value	Notes
Pass	True	True	Testing that checkingIfIn returns the correct boolean on input 'grapes'
Pass	False	False	Testing that checkingIfIn returns the correct boolean on input 'carrots'
Pass	False	False	Testing that checkingIfIn returns the correct boolean on input ('grapes', False)
Pass	True	True	Testing that checkingIfIn returns the correct boolean on input ('carrots', False)
Pass	False	False	Testing that checkingIfIn returns the correct boolean on input ('grapes', d = {'carrots': 1, 'peas': 9, 'potatos': 8, 'corn': 32, 'beans': 1})
Pass	True	True	Testing that checkingIfIn returns the correct boolean on input ('peas', d = {'carrots': 1, 'peas': 9, 'potatos': 8, 'corn': 32, 'beans': 1})
Pass	False	False	Testing that checkingIfIn returns the correct boolean on input ('peas', False, {'carrots': 1, 'peas': 9, 'potatos': 8, 'corn': 32, 'beans': 1})
Pass	True	True	Testing that checkingIfIn returns the correct boolean on input ('apples', False, {'carrots': 1, 'peas': 9, 'potatos': 8, 'corn': 32, 'beans': 1})

You passed: 100.0% of the tests

Question in Context (/runestone/books/published/fopp/AdvancedFunctions/ChapterAssessment.html#ac15_5_5)

Not yet graded

We have provided the function `checkingIfIn` such that if the first input parameter is in the third, dictionary, input parameter, then the function returns that value, and otherwise, it returns `False`. Follow the instructions in the active code window for specific variable assignmemts.

Save & Run 17/09/2022, 22:02:47 - 2 of 2 Show in CodeLens

```
def checkingIfIn(a, direction = True, d = {'apple': 2, 'pear': 1, 'fru
    if direction == True:
        if a in d:
            return d[a]
        else:
            return False
    else:
        if a not in d:
            return True
        else:
            return d[a]
# Call the function so that it returns False and assign that function
c_false = checkingIfIn("")
# Call the fucntion so that it returns True and assign it to the varia
c_true = checkingIfIn("",False)
# Call the function so that the value of fruit is assigned to the vari
fruit_ans = checkingIfIn("fruit")
# Call the function using the first and third parameter so that the va
param_check = checkingIfIn("litchi",d = {"litchi":8})
```

Activity: 15.7.6 ActiveCode (ac15_5_6)

Result	Actual Value	Expected Value	Notes
Pass	8	8	Testing that param_check has the correct value
Pass	False	False	Testing that c_false has the correct value
Pass	19	19	Testing that fruit_ans has the correct value
Pass	True	True	Testing that c_true has the correct value

You passed: 100.0% of the tests

Question in Context (/runestone/books/published/fopp/AdvancedFunctions/ChapterAssessment.html#ac15_5_6)

Score Me