

Forum:[Hack Forums](#)

Thread: [Help python basic terminal menu.](#)

jmcatena91 - January 30, 2018, 8:13 pm

Basically I want to get to start and pick a number and it go to the function specified. I know I havent added mc or ex function yet but I can add them easier. I have tried many things but im starting to think its time to scrap. If anyone can help and tell what im doing wrong it would be greatly appreciated.

edit: To clarify further if I did do the functions correct Im not sure how I call them for the start function. So when I run for example it calls menu_choice function then goes to the appropriate function from the menu. So if I were to run it, It should pop up the menu asking "Choose your options..." I type in one for example "cc" it should then go to the cc function and let me finish on that function then when it spits out the cc print should ask if I would like to start again.

```
Code a = '''
*****
*      menu      *
*****
* CC - Character Count *
* RV - Reverse      *
* M5 - Multiply by 5  *
* MC - Middle Characters*
*****
* EX - Exit        *
*****
'''

print(a)

def menu_choice():
    while True:
        choice = input("Choose your Options, CC, RV, M5, MC, EX :)")
        if choice.lower() not in ('cc', 'rv', 'm5', 'mc', 'ex'):
            print("That is not a valid option.")
        else:
            return choice

def CC():
    if menu_choice == 'CC':
        cc = input('Enter string: ')
        print(""" + menu_choice + ""is", len(cc), "characters long")

def RV():
    if menu_choice == "RV":
        rv = input('Enter string: ')
        print(rv[::-1])
```

```
def M5():
    while True:
        if menu_choice == 'M5':
            m5 = input('choose "N" for number or "S" for string: ')
            if m5.lower() not in ('n', 's'):
                print('That is not a valid option.')
            elif m5.lower() == 'n':
                print(m5 * 5)
            elif m5.lower() == 's':
                print(str(m5) * 5)
            else:
                return m5
```

```
def start():
```

```
    while True:
        start()
        restart = input('Do you want to go again? (y/n) ')
        if restart.lower() != 'y':
            break
```

BottomNotch - February 2, 2018, 5:56 pm

I really don't know python too well, but from my understanding, this is wrong.

1st things first)

So what I see you doing is declaring a bunch of functions.... but not actually calling them. You must call a function in order for it to run, right now they ain't doin shit.

So if you want to run CC, just add 'CC()' to wherever you want it to be called. In your case when CC is input by the user.

2nd) You are returning the choice, which is the actual input (or choice) that the user indicated. See now my only confusion here is if you are returning 'choice' to the function 'menu_choice()', does that actually hold that input(choice) now? So that allows (if menu_choice == 'CC') to work?.. I just didn't think a function could be equal to some input you know? Like I think that's just suppose to be (if 'choice == 'CC')

3rd) I do know that it should not be written like that. Inside of your functions is where you put the condition (if choice = whatever), but the thing is, it's not ever getting to those functions because you never called them in the first place.

4th) Yeah your start runs cuz you call it at the end, but inside each function (if you want it to reset/ask u again or whatever), then just include Start() at the end of each function, after it prints what it has to

5) I think it should be altered to something like:::

```
Code def menu_choice():
    while True:
        choice = input("Choose your Options, CC, RV, M5, MC, EX :")
        if choice.lower() not in ('cc', 'rv', 'm5', 'mc', 'ex'):
            print("That is not a valid option.")
        else:
            return choice
//or just put the if, elif choices right into this function, no need to return choice then lolol
```

```
def choiceLoader(choice):
    if choice == 'CC':
        CC()
    elif choice == 'RV':
        RV()
    elif choice == 'M5':
        M5()
    elif choice == 'MC':
        MC()
    elif choice == 'EX':
        EX()
```

but like... i don't know the syntax, so that's just a basis. then u gotta choose where to call ur functions n shit. so i guess call choiceLoader() in the first function after you get the choice.... OR just get rid of the choiceLoader function I created and put the if,elif statement right into that first function once it grabs the choice!

fix ur shit <3

epicpeanut - February 10, 2018, 1:08 pm

If I'm understand the issue correctly, you want to be a start function which ask the user for input. Once input is given, to head the said command Function (i.e CC()). After running the CC(), it should return the input given. Then ask to continue (or go back the the start). In this case, you'd want to use a comparison. After running the whole function of CC(), declare a variable called option, in which option = input("Would you like to restart? y/n:"). Then use an if statement to compare if their choice (y or n) is equal to y or n. If option == y, call your start function and then a break. If option == n, end the process. If option != y and option != n: , option = input("Invalid Option, Would you like to restart? : y/n" and then a continue(I'm not at a computer right now so I can't test it, but the continue should work. If not, remove continue) . All of this should be within a while True loop. Hope this helps, if not, leave me a message or another post in this thread and I'll look more into it when I'm at a computer ?.

Also with the Start method, if you want it to open up a command line which asks the user to pick a command, I'm which you defined a function for each command, you'll use what my post said similarly with the option = input(""). Then compare the option given and follow through the calling the command, you call it simply by doing: CC(). And so on with the other functions ?

Just realised you posted your code. Your main problem is trying to compare their input in another function besides the Start function.

attr - February 10, 2018, 2:24 pm

you could use globals() to call the user supplied string as a function

Code

```
...
choice = input("Choose your Options, CC, RV, M5, MC, EX :")
if choice.lower() not in ('cc', 'rv', 'm5', 'mc', 'ex'):
    print("That is not a valid option.")
else:
    globals()[choice]()
...
but you really should have a second look at your code, it's full of errors
```

SerCoDev - February 12, 2018, 4:53 am

I may be mistaken but I am quite sure you should be defining the functions before you create the menu . I am still learning and may be wayyy off but it seems to me that this would cause the kind of problem you are experiencing.

vorticalbox - February 12, 2018, 10:34 am

(Feb 12, 2018 05:53 AM)SerCoDev Wrote: [\[CITE\]](#) I may be mistaken but I am quite sure you should be defining the functions before you create the menu . I am still learning and may be wayyy off but it seems to me that this would cause the kind of problem you are experiencing.

Python on Windows will throw a fix same when accessed global variables with the global flag. On Linux it's all good

But yes from a code reading point of view you should define from last to first in terms of functions