

Exercise 10: Structure of program and functions

Look at the following piece of code – This was originally your exercise 2 challenge B.

```
#-----  
#sample structure.py  
#created by: d.payter  
#a program to calculate a new balance on a clothing store card  
#-----  
#In Python the starting point below is referred to as the driver module  
InitialBalance=35.00  
  
ClothingPurchase=float(input ("how much money did you spend? (£)"))  
  
NewCardBalance=(InitialBalance)-(ClothingPurchase)  
  
print("your new Card Balance is " +'£{0:.2f}'.format(NewCardBalance))
```

The issue was with the original code, there was no flexibility in the sense that the clothing purchase was fixed. We have now added an input statement to your code so that the user would define how much money they had spent.

Look at the structure of this code..... What would happen if we changed the order of all of the lines of code?

There is a set format to the structure of the code.... Your code must be robust enough to deal with a range of different input statements.

When we are dealing with only one problem this is bearable, but what happens if we have multiple problems to solve in one program?

We break it down into modules.... In other programming languages these are often referred to as "a function".

Enter in the following code and read the comments to see what the structure does....

```

Example –
#-----
#exercise 10b.py
#created by d.payter
#a program to show the structure of functions
#-----

#function definitions
def functionone():
    x = 8          #a new variable is created
    print ("x = ", x)  #accesses the local variable x, prints 8

def functiontwo():
    print ("x is now", x) #accesses the global variable x, prints 5

#main routine
x=5              #the global variable is created
print("x is",x)   #accesses the global variable and prints 5
functionone()     #calls functionone
functiontwo()     #calls functiontwo

```

It is now important that we understand how this could be used...

Challenge A

Your program has stored six integer values: 6,17,2,15,0,3

Write two functions in your program. One should calculate and display the average of the values, the other should calculate and display the total.

Call these functions from the main routine.

Use the following structure

```

# define your functions
    • def average():
    • def total():
# main routine, name all stored integers starting with N1.....N6
N1 = ?
N2 = ?
#call functions

```

The real power is a when we are able to apply a module repeatedly to different data.

Challenge B.

You want to send the same letter to 5 different people, but with their personal names at the top of the letter. Plan and write a modular program which can do this. The name of the person to whom the letter is addressed should be the input variable to a module which is called 5 times....

Dear John Smith
This is to inform you that you have been selected.....

Dear John Smith
This is to inform you that you have been selected.....

Dear John Smith
This is to inform you that you have been selected.....

Dear John Smith
This is to inform you that you have been selected.....

Dear John Smith
This is to inform you that you have been selected.....