Book Buying Program – Using lists, functions, numeric variables, input, loop, and calculations.

1. Problem Definition: Develop a program that collects data about book purchases into a list and calculates the average cost of the books being purchased. Output of the total and average must be well formatted.
2. Problem Analysis: Create functions to be called by main() for the basic IPO activities. Input values for the number of books and the price of each book. Validate the numbers as whole numbers only. Append the book price into a list. Then calculate the total and average costs. Finally, output the total book cost and average book price.
3. Program Algorithm:

* Create a main() function that calls functions that:
  + displays a heading,
  + inputs and returns the number of books,
  + inputs, calculates, and returns the total cost and average price
  + accepts all the data and formats informational output
* In the main() function: Initialize variables for the number of books being purchased and the total and average costs. Then call the functions in turn. Finally, ask the user if they want to rerun the program.
* Display a program heading.
* In an inputs function - Ask the user for the number of books. Validate the input. Create an empty list. Create a loop that asks the user to input the price of each book and appends the price to a list. Call the input / validation function to verify each input.
* In a “get positive integer” function – perform the input and validation of each whole number.
* In a processing function - calculate the total from the list of prices. Also, calculate the average book price.
* In an outputs function - Display the total and average cost using nice formatting.

1. Program Code and Test:

*""" M. Bock 6/18/2019 this program summarizes cost of a list  
of books, giving each book a number. """*print(**'This program summarizes a book list.'**) *# print intro***def** main(): *# call functions and save results under key variable names.* **try**: *# generic exception handling - turn off during development* num\_books, price\_list = inputs()  
 total, average = processing(price\_list)  
 outputs(num\_books, price\_list, total, average)  
 restart = input(**'Need more books? Enter y or n: '**) *# restart feature* **if** restart == **'y'**:  
 print(**'OK, let\'s create a new list.'**)  
 main()  
 **else**:  
 print(**'Thanks for using the program.'**)  
 **except** Exception **as** err: *# turn off during development* print(err) *# turn off during development***def** inputs(): *# collect info needed from the user.* print(**'Enter the number of books that you need '**) *# user sets the list length/ repetitions of the for loop* num\_books = get\_pos\_int() *# call validation function to collect int > 0* price\_list = [] *# create list to save prices* **for** index **in** range(num\_books): *# for loop runs user-specified number of times & collects info on each book* print(**f'Enter the cost of book #{**index +1**}, to the nearest dollar: '**)  
 book\_cost = get\_pos\_int() *# call validation function to collect int > 0* price\_list.append(book\_cost) *# build price list* **return** num\_books, price\_list  
  
  
**def** get\_pos\_int(): *# collect and validate an int > 0* pos\_int = input(**'Please enter a whole number: '**)  
 **while** pos\_int.isnumeric() **is False or** int(pos\_int) == 0:  
 pos\_int = input(**'Enter a number greater than 0: '**)  
 pos\_int = int(pos\_int)  
 **return** pos\_int  
  
  
**def** processing(price\_list): *# use the list to calculate summary data* total = sum(price\_list)  
 average = round(total / len(price\_list), 2)  
 **return** total, average  
  
  
**def** outputs(num\_books, price\_list, total, average): *# display information about each book, and summary info* print(**f'Info on {**num\_books**} Books Needed'**)  
 print(**f'{"Book #":<10}{"Price":>10}'**)  
 **for** index **in** range(len(price\_list)):  
 print(**f'{**index + 1**:>2d}\t\t ${**price\_list[index]**:>8.2f}'**)  
 print(**f'{"Total":<10} ${**total**:>8.2f}'**)  
 print(**f'{"Average":<10} ${**average**:>8.2f}'**)  
  
  
main() *# call main to start or restart program.*

1. Program test output: \*\*\* TAKE A SCREEN SHOT OF THE OUTPUT IN THE SHELL \*\*\*

/usr/local/bin /lab4starter\_book\_list.py

This program summarizes a book list.

Enter the number of books that you need

Please enter a whole number: 5

Enter the cost of book #1, to the nearest dollar:

Please enter a whole number: 2

Enter the cost of book #2, to the nearest dollar:

Please enter a whole number: 3

Enter the cost of book #3, to the nearest dollar:

Please enter a whole number: 5

Enter the cost of book #4, to the nearest dollar:

Please enter a whole number: 6

Enter the cost of book #5, to the nearest dollar:

Please enter a whole number: 9

Info on 5 Books Needed

Book # Price

1 $ 2.00

2 $ 3.00

3 $ 5.00

4 $ 6.00

5 $ 9.00

Total $ 25.00

Average $ 5.00

Need more books? Enter y or n: n

Thanks for using the program.

Process finished with exit code 0>>>

\*\* RUN SEVERAL TIMES TO SHOW ERROR MESSAGES AND \*\*

\*\* WORKING WITH INCORRECT DATA \*\*

1. Maintenance: this phase of the program life cycle is where bugs are listed, before being fixed. Also, the programmer can list changes to make the code better in some way. Also, enhancements may be requested by the user and listed here for review.

This program works as requested.

Program Code ToDo list:

No planned changes required