**Course**: CSC500\_1

**Module & Assignment**: 5 & Critical Thinking Assignment

**Student**: Jegan Palaniyandi

### Creating Python Programs

#### **Part 1:**

#### Write a program that **uses nested loops** to **collect data and calculate the average rainfall** over a period of years. The program should **first ask for the number of years**. The **outer loop will iterate once for each year**. The **inner loop will iterate twelve times**, once for each month. Each iteration of the inner loop will **ask the user for the inches of rainfall for that month**. After all iterations, the program should **display the number of months, the total inches of rainfall, and the average rainfall per month** for the entire period.

**Pseudocode**

# Pseudocode for calculating rainfall metrics

# Input: Total years for calculating metrics

# Recorded rainfall of each month for the total years entered above

# Initialize variables

SET tot\_years TO 0

SET curr\_year TO 2024

SET tot\_months TO 0

SET tot\_rainfall TO 0.00

SET avg\_rainfall TO 0.00

# Declare a list of month names

months = ["January", "February", "March", "April", "May", "June", "July", "August",

"September", "October", "November", "December"]

# Read input from user

INPUT tot\_years

# Outer loop - loops through the total number of years

FOR year FROM 1 TO tot\_years

PRINT "Enter the monthly rainfall for year -> " + (curr\_year - year)

# Inner loop - loops through the twelve months of a year

FOR index, month IN months

INPUT this\_month\_rainfall

tot\_months += 1

tot\_rainfall += this\_month\_rainfall

END FOR

END FOR

# Calculate average rainfall

avg\_rainfall = tot\_rainfall / tot\_months

# Output results

PRINT "Total months --> " + tot\_months

PRINT "Total rainfall recorded in " + tot\_years + " year(s) --> " + tot\_rainfall + " inches"

PRINT "Average rainfall in " + tot\_years + " year(s) --> " + avg\_rainfall + " inches"

#### **Code**

*"""*

*'rainfall\_metrics' method gets the following inputs from user,*

*- Total years for calculating metrics*

*- Recorded rainfall of each month of the total years entered above*

*Outputs,*

*- Total months*

*- Total rainfall recorded*

*- Average rainfall as 'Total rainfall recorded' divided by 'Total months'*

*"""*

def rainfall\_metrics():

# get the total number of years for which the metrics needs to be calculated

tot\_years = int(input("\nEnter the number of years for calculating rainfall metrics -> "))

# assumption is curr\_year = 2024 so that total number of years(n)

# is considered as recent past n years

curr\_year = 2024

# Initializing program variables

tot\_months = 0

tot\_rainfall = 0.00

avg\_rainfall = 0.00

# Declaring a months list which will be used in the loop while prompting the user

# to enter rainfall for each month

months = [

"January", "February", "March", "April",

"May", "June", "July", "August",

"September", "October", "November", "December"

]

# Outerloop - loops through the total number of years

# Innerloop - loops through the twelve months of a year

# calculates the total months and accumulation of rainfall at the end of each loop

for year in range(1, tot\_years + 1):

print('\nEnter the monthly rainfall for year -> ' + str(curr\_year - year) + '\n')

for index, month in enumerate(months):

this\_month\_rainfall = float(input('Enter the average rainfall(inches) of {} month,

{} --> '.format(month, curr\_year - year)))

tot\_months += 1

tot\_rainfall += this\_month\_rainfall

# Calculating average rainfall as 'Total rainfall recorded' divided by 'Total months'

avg\_rainfall = tot\_rainfall / tot\_months

print('\nTotal months --> {}'.format(tot\_months))

print('Total rainfall recorded in {} year(s) --> {:.2f} inches'

.format(tot\_years, tot\_rainfall))

print('Average rainfall in {} year(s) --> {:.2f} inches'

.format(tot\_years, avg\_rainfall))

if \_\_name\_\_ == "\_\_main\_\_":

print()

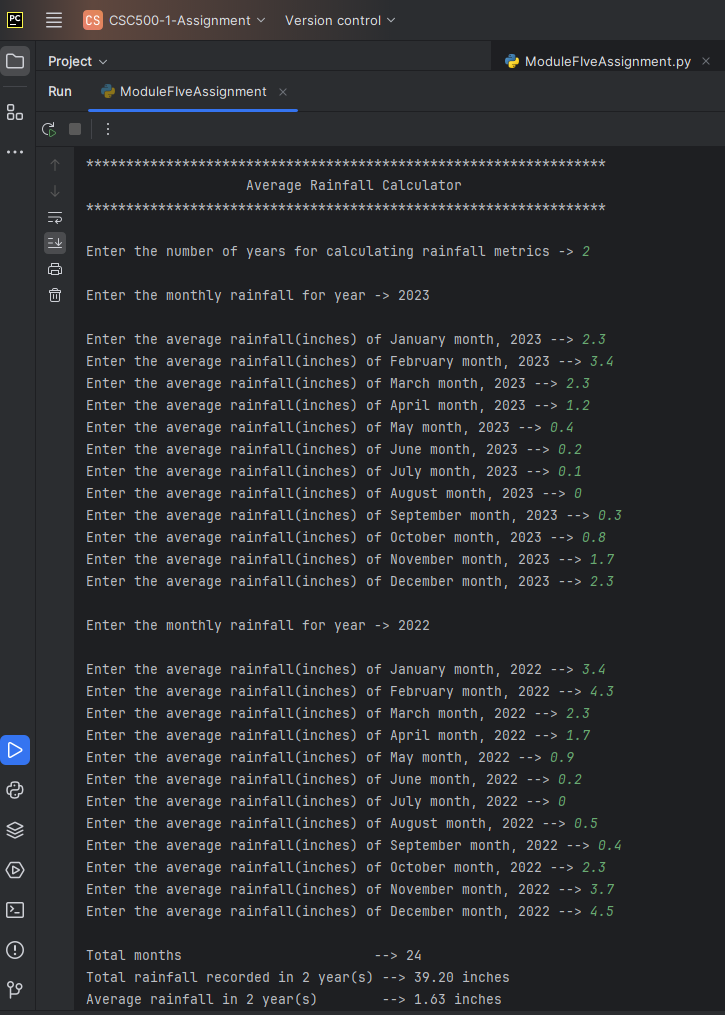
print('\*'\*65)

print(' '\*20 + 'Average Rainfall Calculator')

print('\*'\*65)

rainfall\_metrics()

**Output of code running from pycharm**



#### **Part 2:**

The CSU Global Bookstore has a book club that **awards points to its students based on the number of books purchased each month**. The points are awarded as follows:

* If a customer purchases 0 books, they earn 0 points.
* If a customer purchases 2 books, they earn 5 points.
* If a customer purchases 4 books, they earn 15 points.
* If a customer purchases 6 books, they earn 30 points.
* If a customer purchases 8 or more books, they earn 60 points.

Write a program that **asks the user to enter the number of books that they have purchased this month** and then **display the number of points** awarded.

**Pseudocode**

# Pseudocode for calculating award points based on books purchased

# Input: Total number of books purchased this month

# Output: Total points awarded for the books purchased

# Initialize variables

SET tot\_books\_purchased TO 0

SET tot\_points TO 0

# Read input from user

INPUT tot\_books\_purchased

# Check if the entered value is valid (0, 2, 4, 6, or 8)

IF tot\_books\_purchased NOT IN [0, 2, 4, 6, 8]

PRINT "Error -- Enter either 0, 2, 4, 6, or 8 as total books purchased"

EXIT

END IF

# Assign points based on the number of books purchased

IF tot\_books\_purchased == 2

tot\_points = 5

ELSE IF tot\_books\_purchased == 4

tot\_points = 15

ELSE IF tot\_books\_purchased == 6

tot\_points = 30

ELSE IF tot\_books\_purchased == 8

tot\_points = 60

END IF

# Output total points awarded

PRINT "Total points awarded for " + tot\_books\_purchased + " books --> " + tot\_points

**Code**

***"""***

***Input:***

***- Total number of books purchased this month***

***Output***

***- Total points awarded for the books purchased***

***book\_club\_award\_program function displays the total***

***award points based on the following table,***

***| Total Books purchased | Award Points |***

***---------------------------------------***

***| 0 | 0 |***

***| 2 | 5 |***

***| 4 | 15 |***

***| 6 | 30 |***

***| 8 | 60 |***

***----------------------------------------***

***"""***

**def book\_club\_award\_program():**

**# Get the total number of books purchased this month as integer and initialize total points**

**tot\_books\_purchased = int(input("\nTotal number of books purchased this month --> "))**

**tot\_points = 0**

**# Assumption has been made a student can purchase books only as 0, 2, 4, 6, 8**

**# If the entered value from user is not one of the above value, the program exits**

**# after printing appropriate error message**

**if (not (tot\_books\_purchased == 0 or tot\_books\_purchased == 2 or**

**tot\_books\_purchased == 4 or tot\_books\_purchased == 6 or**

**tot\_books\_purchased == 8)):**

**print("\nError -- Enter either 0, 2, 4, 6, 8 as total books purchased")**

**exit(4)**

**# Following if loop assigns the total points based on the user input**

**if tot\_books\_purchased == 2:**

**tot\_points = 5**

**elif tot\_books\_purchased == 4:**

**tot\_points = 15**

**elif tot\_books\_purchased == 6:**

**tot\_points = 30**

**elif tot\_books\_purchased == 8:**

**tot\_points = 60**

**else:**

**tot\_points = 0**

**# prints the total points aw**

**print('Total points awarded for {} books --> {}'.format(tot\_books\_purchased, tot\_points))**

**if \_\_name\_\_ == "\_\_main\_\_":**

**print()**

**print('\*' \* 65)**

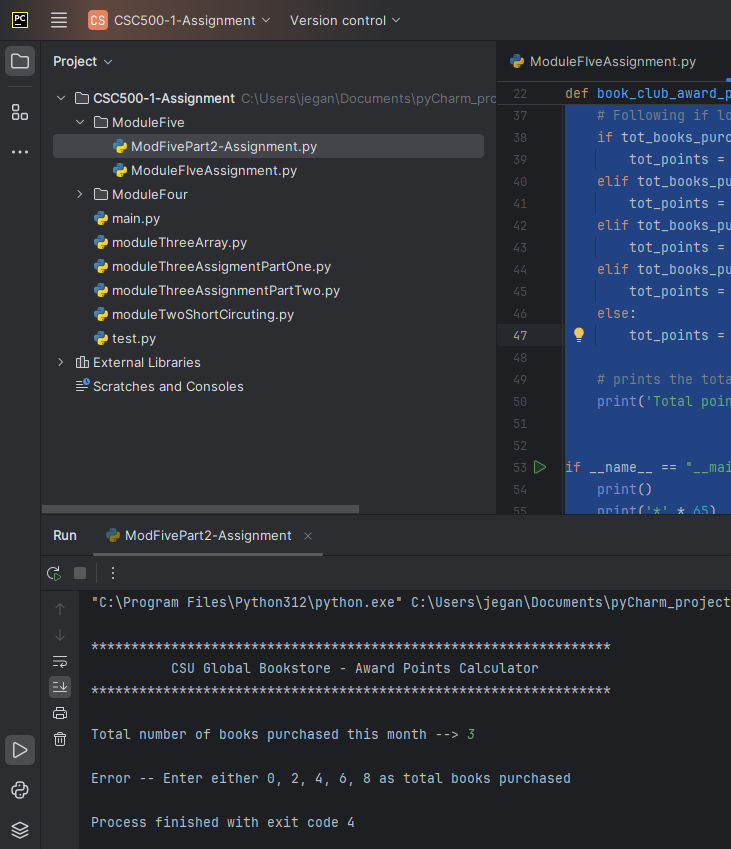
**print(' ' \* 10 + 'CSU Global Bookstore - Award Points Calculator')**

**print('\*' \* 65)**

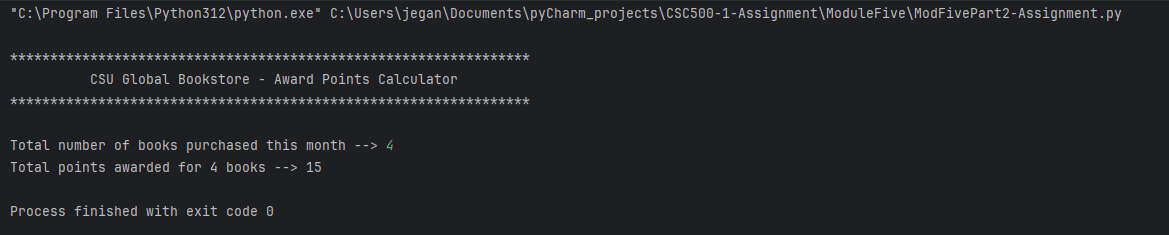
**book\_club\_award\_program()**

**Output of code**

**Invalid Input**

****

**Valid Input**

****

**GitHub Repository**

[**CSC500/moduleThree at main · jeganpalaniyandi/CSC500 (github.com)**](https://github.com/jeganpalaniyandi/CSC500/tree/main/moduleThree)