CSC500

Module 5 Critical Thinking Assignment

GIT Repository: <https://github.com/focomapper/CSC500>

Introduction:

The module 5 Critical Thinking Assignment involves two parts.

Part 1 required creating an outer loop defined by user input number of years, and an inner loop that gathers user input for inches of rain for 12 months. After the loops, the program calculates an average rainfall per month and prints the results. I originally coded part 1 to use a class that contained a list of rainfall inches and the year number, and then added objects instantiated from that class, one of each year, to a list that could be iterated through. It worked but upon reviewing my code I realized I over-complicated the code and rewrote it without using the class. This reduced the number of lines of code substantially. I referenced python.org to figure out how to use the calendar library to translate the month number to the month name (docs.python.org).

Part 2 involved using branching to determine book club points awarded for a user input number of books. I found this task to be incredibly easy, to be honest, and completed the task in about 10 minutes, with testing.

Part 1, Python code calculating average rainfall per month for user input number of years:

Pseudo code-

1. Import calendar library
2. Initialize variables for years, total months and total rainfall
3. Get user input for number of years and validate
4. Outer loop through number of years
5. Inner loop through 12 months, get user input for inches rain per month and validate
6. Calculate average rainfall per month
7. Format and print total months, total and average rainfall

Source code-

import calendar  
  
# Initialize variables  
num\_years = 0  
num\_months = 0  
total\_rain = 0  
  
# Get user input for number of years  
try:  
 num\_years = int(input("Enter number of years: "))  
except ValueError:  
 print("Number of years must be a whole number. Run program again and enter valid number of years")  
 exit()  
  
# Outer loop through number of years  
for y in range(num\_years):  
 print(f"Year: {y + 1}")  
   
 # Inner loop through months in year  
 for m in range(12): # Loop through 12 times, one for each month in year  
 try: # Ensure inches are float  
 inches\_rain = float(input(f"Enter the inches of rain for month {calendar.month\_name[m+1]}: "))  
 num\_months += 1  
 total\_rain += inches\_rain  
 except ValueError:  
 print("Your input for inches of rain was not a number, run program again and enter valid inches")  
 exit()  
  
# Calculate average rain per month  
average\_rain = round(total\_rain/num\_months, 2)  
  
# Print required output  
print(f"\nTotal number of months: {num\_months}")  
print(f"Total inches of rain: {round(total\_rain, 2)} inches")  
print(f"Average rainfall per month for period of {num\_years} years: {average\_rain} inches")

Screenshot of code executing-

A screenshot of a computer

Description automatically generated

Part 2, Python code for calculating book club points based on user input number of books:

Pseudo code-

1. Initialize variable number of books and total points
2. Get number of books from user and validate
3. Determine points based on number of books
4. Format and print results

Source code-

# Initialize number of books, total points  
num\_books = 0  
total\_points = 0  
  
# Get user input for number of books purchased  
try:  
 num\_books = int(input("Enter number of books purchased this month: "))  
except ValueError:  
 print("Number of books must be a whole number. Run program again and enter valid number of books")  
  
# Determine points based on number of books  
if num\_books < 4:  
 total\_points = 5  
elif 4 <= num\_books < 6:  
 total\_points = 15  
elif 6 <= num\_books < 8:  
 total\_points = 30  
elif num\_books >= 8:  
 total\_points = 60  
  
# Output results  
print(f"\nTotal points for {num\_books} books = {total\_points} points.")

Screenshot of code executing-

A screenshot of a computer program

Description automatically generated

References:

Python.org. (n.d.). Calendar – General calendar-related functions. Retrieved February 18, 2024, from https://docs.python.org/3/library/calendar.html