# Module 5: Critical Thinking

Maurice Henning

Colorado State University Global

CSC500-1: Principals of Programming

Dr. Steven Evans

April 23, 2024

**Online Shopping Cart**

In this two-part module, I am first asked to create a Python program that uses nested

loops to collect data and calculate average rainfall over the years. Loops such as “for” and

“while” in Python are blocks of code that repeat a sequence of commands as mentioned in

APMonitor.com (2021). I should program it so that the outer loop iterates once for each year,

the inner loop iterates twelve times for that year, and once for each month.

Additionally, I should program each iteration of the inner loop to request the user for inches of

rainfall for that month. Finally, the program should display the number of months, total

inches of rainfall, and the average rainfall per month for the entire period.

Part 1: The first request was to create a statement requesting the number of years from the user as

specified in the initial comment. I created the variable labeled it number\_years and enclosed the

statement to print the number of years between 1-15 via integer only.

Secondly, I created a conditional statement asking to verify input. In this instance the ‘if’

statement was used to check whether the number of years is less than zero or more than fifteen

then the program would print ‘try again between 1-15 years’. The ‘if’ statement causes one or

more statements to execute only when a Boolean expression is true Gaddis (2018).

In my next step, I created two variables to represent total rainfall and total number of months. I

initialized the variables at 0. The request was to create an outer loop to iterate once per year. I

used the ‘for’ statement to repeat the range of numbers input by the variable number\_years. The

range function creates a collection of numbers on the fly, 0,1,2,3.

The inner loop iterarates twelve times asking users for inches of rainfall and the average rainfall

per month. The variable labeled rainfall asks for user to input rainfall for the month in a year in

a decimal format. The loops will increase by adding a +1 inside the curly brackets of month and

year. Additionally, I created a conditional ‘if’ statement so that if a user enters anything less than

0.0 the return statement would be invalid output. This execution continues until all the numbers

for rainfall and the month and year have been entered.

The next comment creates a statement that adds the rainfall for that month's total. The following

statement increments the variable number\_months by 1. The final math takes place when I

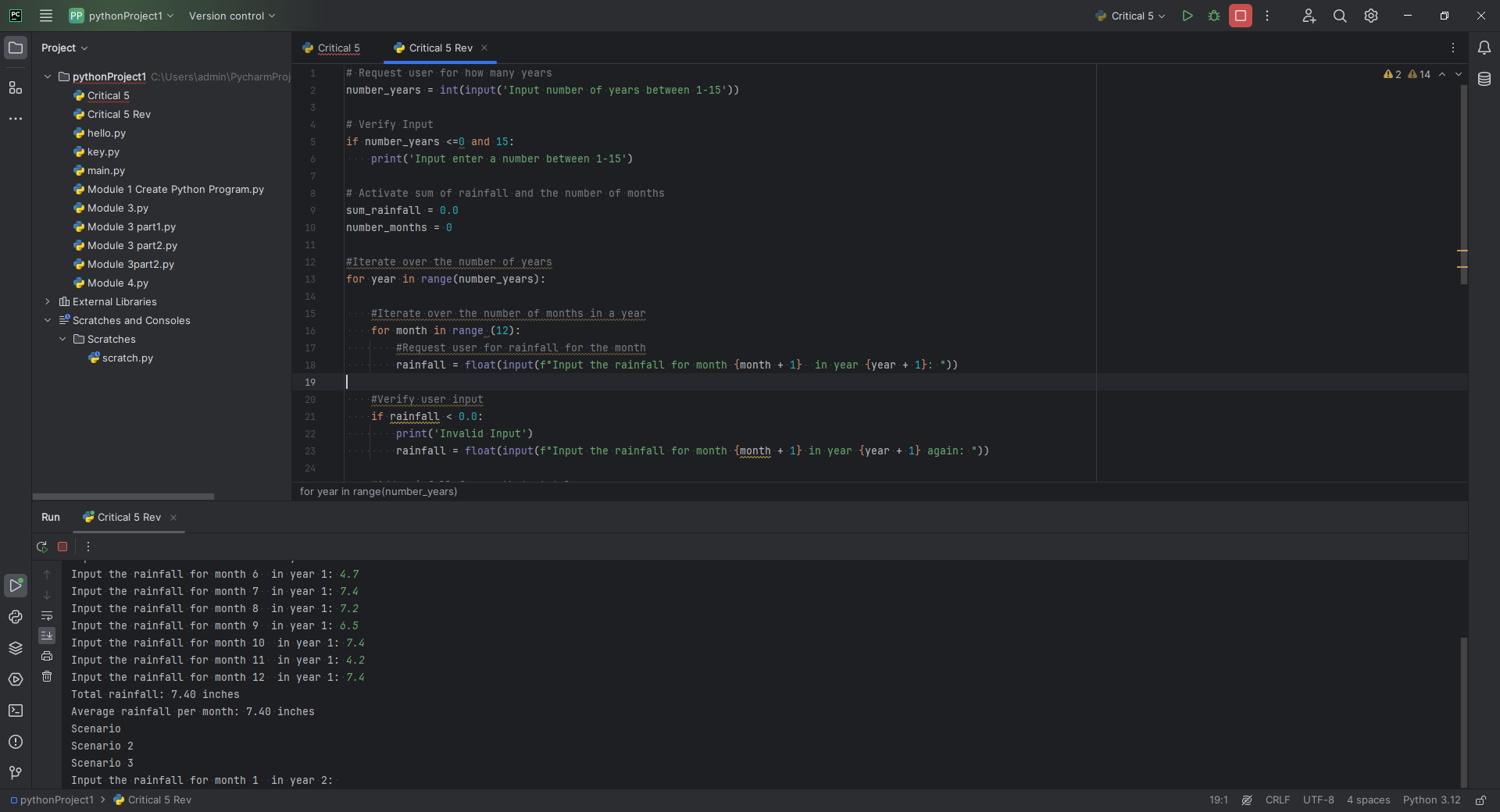
comment to calculate the average rainfall by the variable sum\_rainfall divided by

number\_months.

After all calculations have taken place the program outputs messages showing the total rainfall

formatted with 2 spaces in inches while also illustrating average rainfall per month. The tally is

printed as scenarios. See figure 1:



Figure

This was an extremely challenging program because creating nested loops requires that your

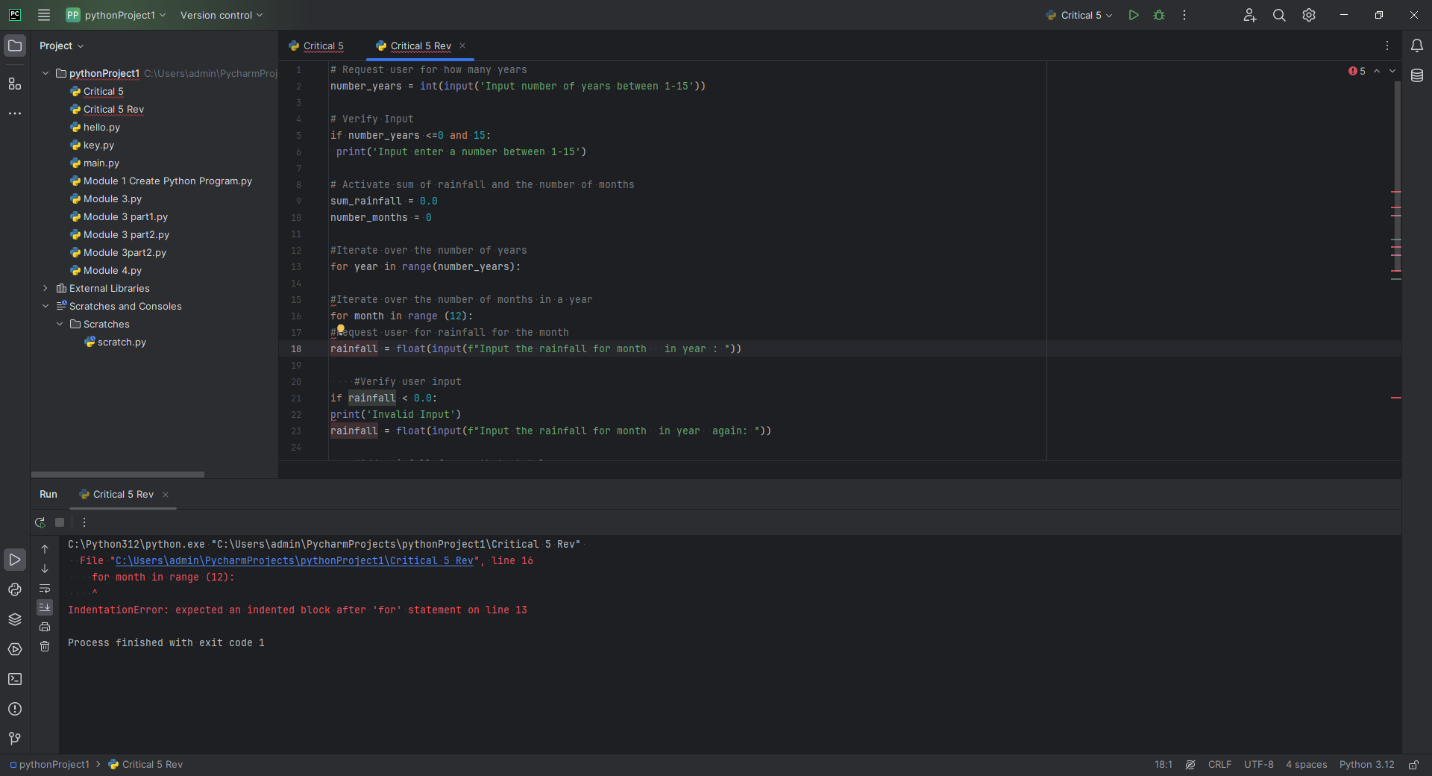
code is created with the proper indentation spacing. This is vitally important as the program will

not execute even if the code is accurate in syntax. An indentation in Python is used to segregate a

singular code into identifiable groups of functionally similar statements. The error would surface

in the execution. I would consistently receive code error: IndentationError: expected an indented

block after ‘for’ statement on line 13 See figure 2:



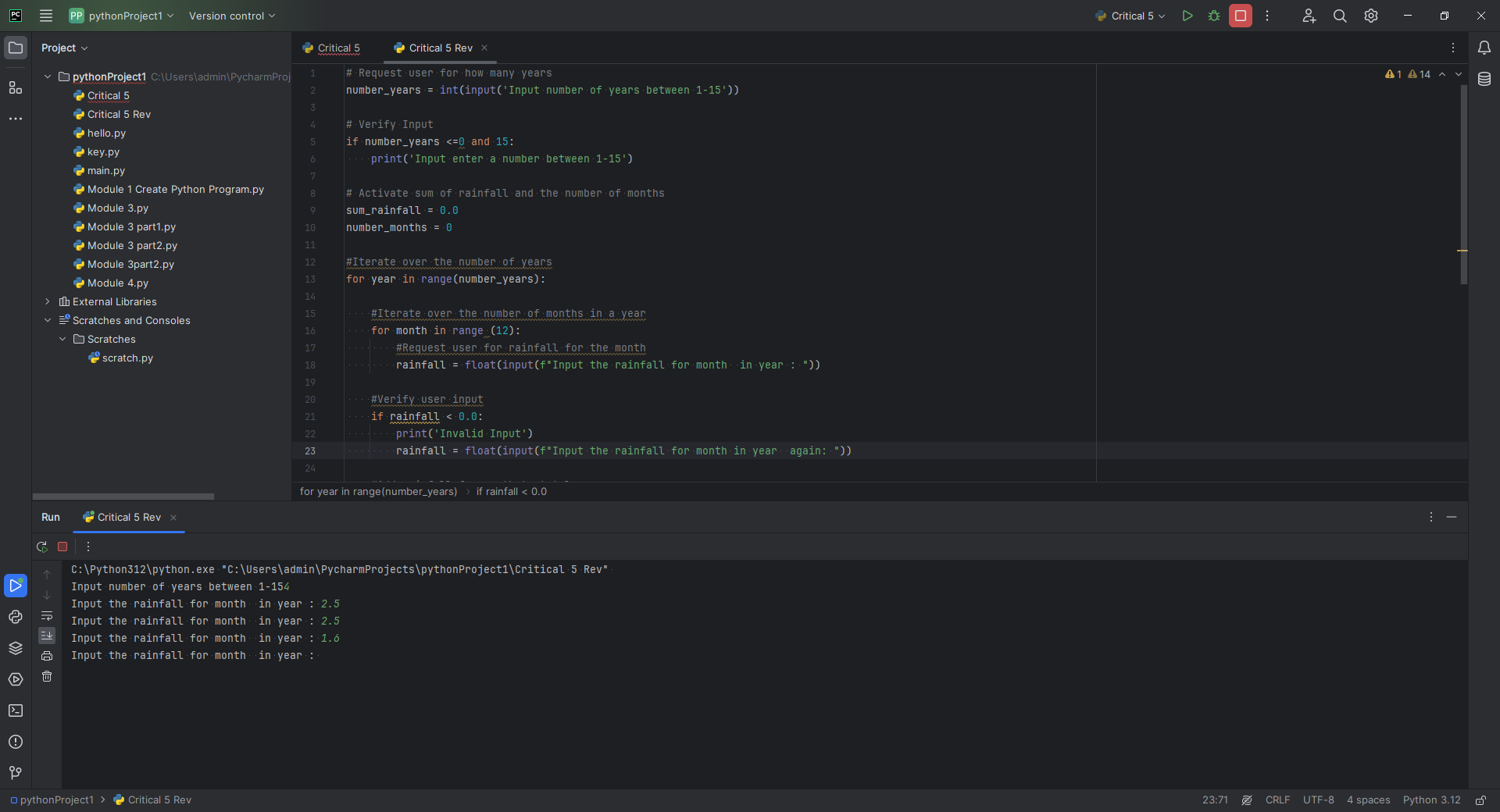
Figure

Furthermore, initially, I was unable to get the month to increase in range. My output always

printed month in year not exhibiting the number in month. After further research I discovered

adding a +1 to the month array increases the numerals. Although this did not provide errors in

the program output it defeated its operational purposes. See figure 3:



Figure

**PSEUDOCODE:**

**OBTAIN number of years  
IF condition is less than 0 or more than15 THEN print INPUT number between 1-15**

**INIT rainfall and number of months**

**FOR year in range  
FOR month in range  
OBTAIN rainfall month and year  
IF rainfall is less than 0 print invalid input ELSE output rainfall   
COMPUTE sum of rainfall  
COMPUTE number months  
COMPUTE average rainfall  
DISPLAY Total rainfall and average in scenario**

Part 2: of the module requests the creation of a program for the CSU bookstore that rewards

students based on the number of books purchased each month. The students are awarded points if

they purchase a certain number of books. The program asks the user how many books then

returns what they have gained as a result of the purchase.

I began the program with a comment statement creating the variable I labeled number\_books

followed by the statement to ‘enter the number of books purchased this month.’

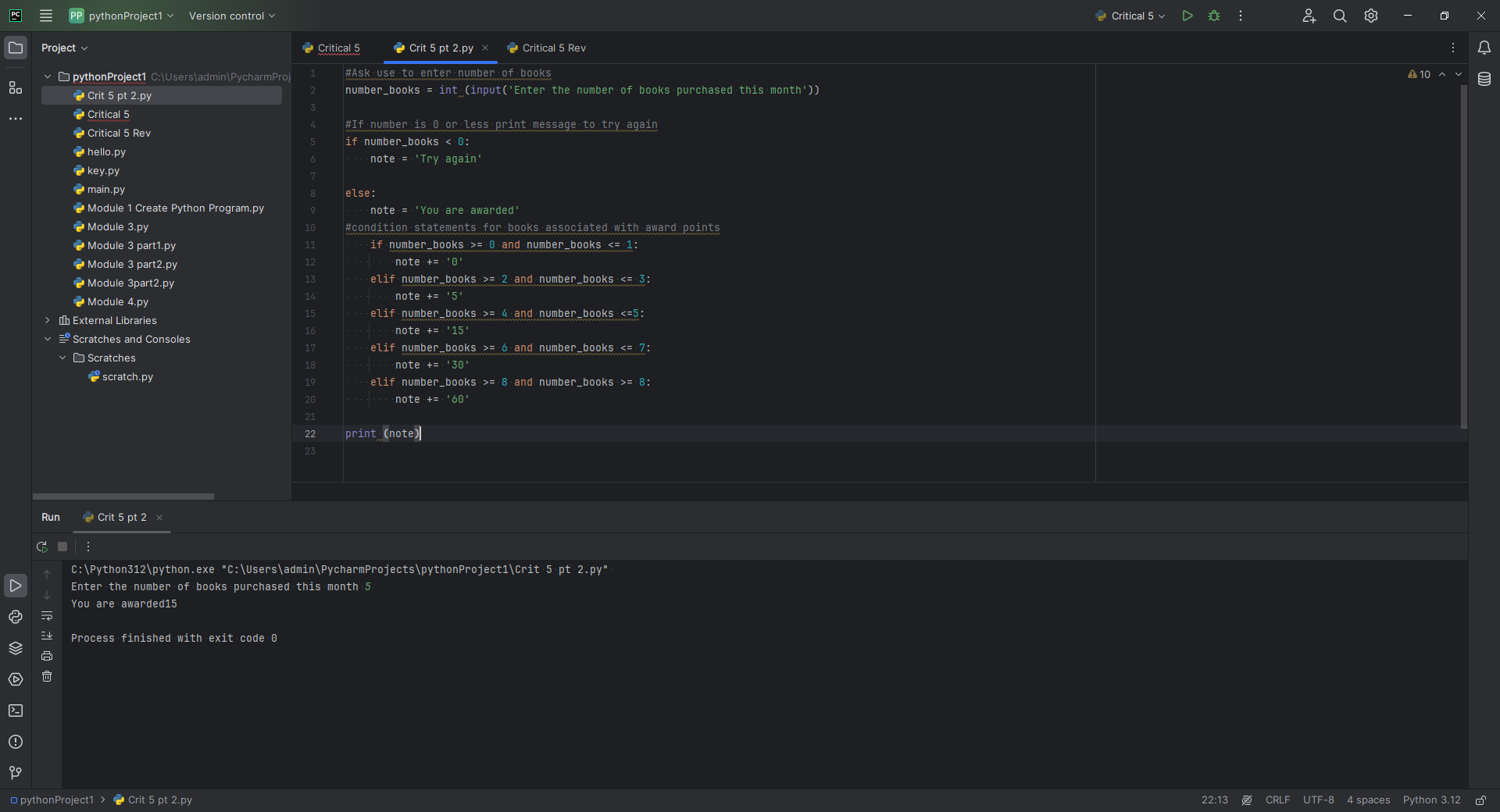
I create a conditional ‘if’ statement that states if the user types any number less than zero to print

‘try again’ otherwise print the variable I labeled note which equals the statement ‘you are

awarded.’ This display statement is tied in with the following conditional ‘if’ and ‘elif’

statements, whereas whatever number the user enters will print that note variable followed by the

corresponding conditional statement that coincides with the number entered. See figure 4 :

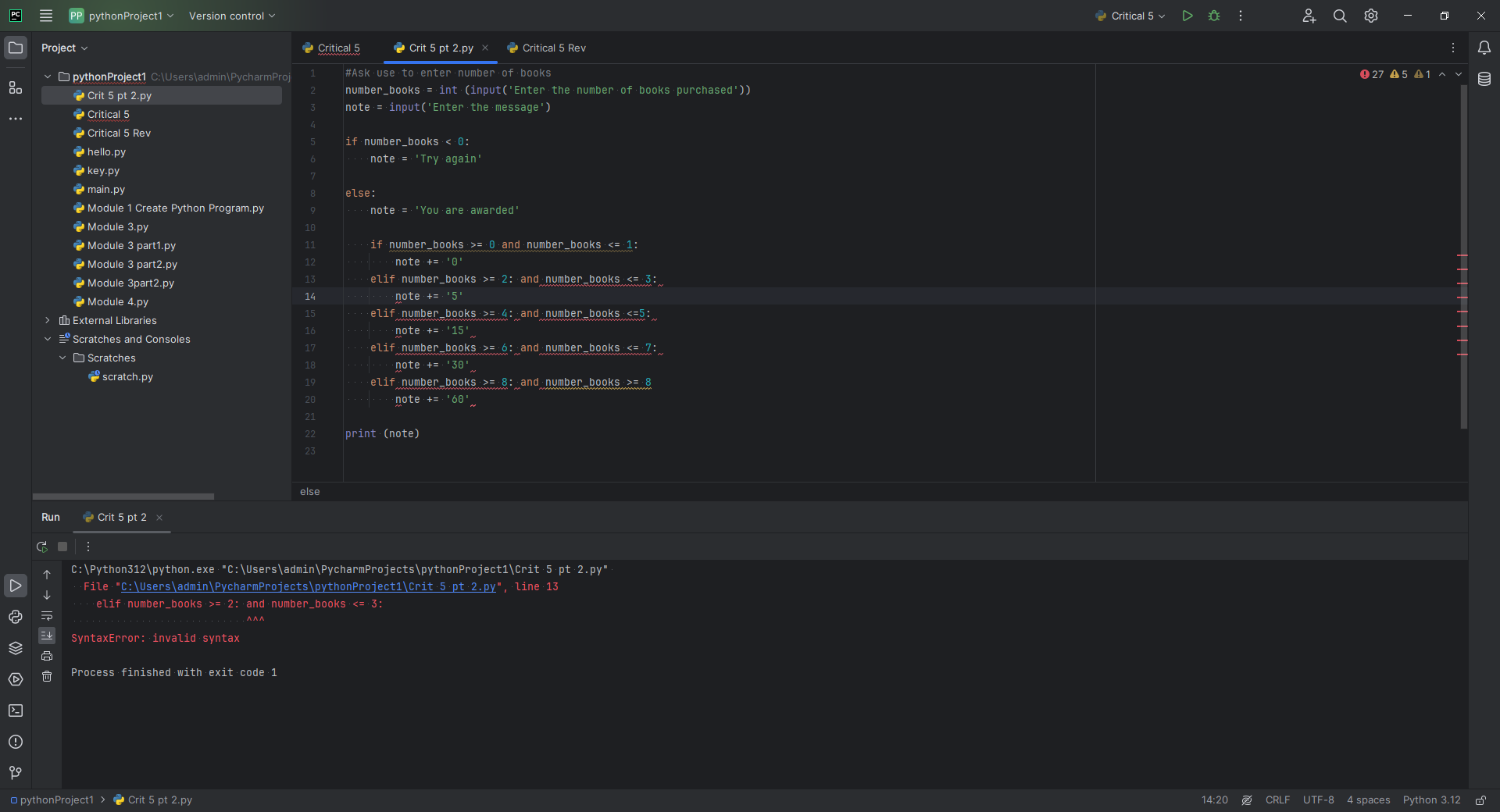


Figure

Most of the errors I experienced while building this program were simple syntax errors. Such as

forgetting to put colons or semicolons after a statement. Otherwise pretty straightforward. See

figure 5.



Figure

**PSEUDOCODE**

**OBTAIN number of books  
IF number books is less than 0 then print try again  
ELSE output You are awarded  
IF number of books equal to or less than 1  
ESLSEIF number of books equal to or less than 3  
ESLSEIF number of books equal to or less than 5  
ESLSEIF number of books equal to or less than 7  
ESLSEIF number of books equal to or more than 8  
DISPLAY answer**

**References**

(2020, January 22). Python Programming Basics   
<https://apmonitor.com/che263/index.php/Main/PythonBasics>

Gaddis, T (2018). Starting out with Python   
Fourth edition. Boston: Pearson

Loops in Python. (2021).[Video/DVD] APMonitor.com. Retrieved from <https://video.alexanderstreet.com/watch/loops-in-python>