ITSE 1359 – Lab 7 Assignment (Refer to Ch. 7 as needed):

NOTE: Lab 7 requires you to work on and submit two programs. Read this document carefully and comply with submission and grading criteria.

Your first problem (Interest Due) is more prescriptive (use video provided), while the second program (Southern States with loops) is more open ended (solve as specified).

Program 1 – Interest Due (interest_due.py):

Note: Sublime Text does not handle user input, so you will have to use an editor that can like Idle or VS code – Unless you want to run your file from the command prompt – I show you how to do that in my YouTube videos for chapter 7.

Problem Definition: Create a Python program that prompts for loan amount, interest rate, and years of loan to find total interest due. The program should prompt for repeat. Name your file: interest_due.py. Do not forget general and specific comments.

Coding: Input for loan amount and interest rate should use the float() function, while input for years of loan should use the int() function. Your prompt for these three things must be in a loop which askes the user if they want to repeat. To find total interest due use this formula: total interest due = amount of loan * (interest rate/100) * years of loan.

Output: A program title and goodbye are required. Sentence structure for prompts and output should match screenshot provided. Dollars should be formatted as currency. Loop in this program is like that found in mountain_poll.py from your book.

Specific Comments/Pseudocode:

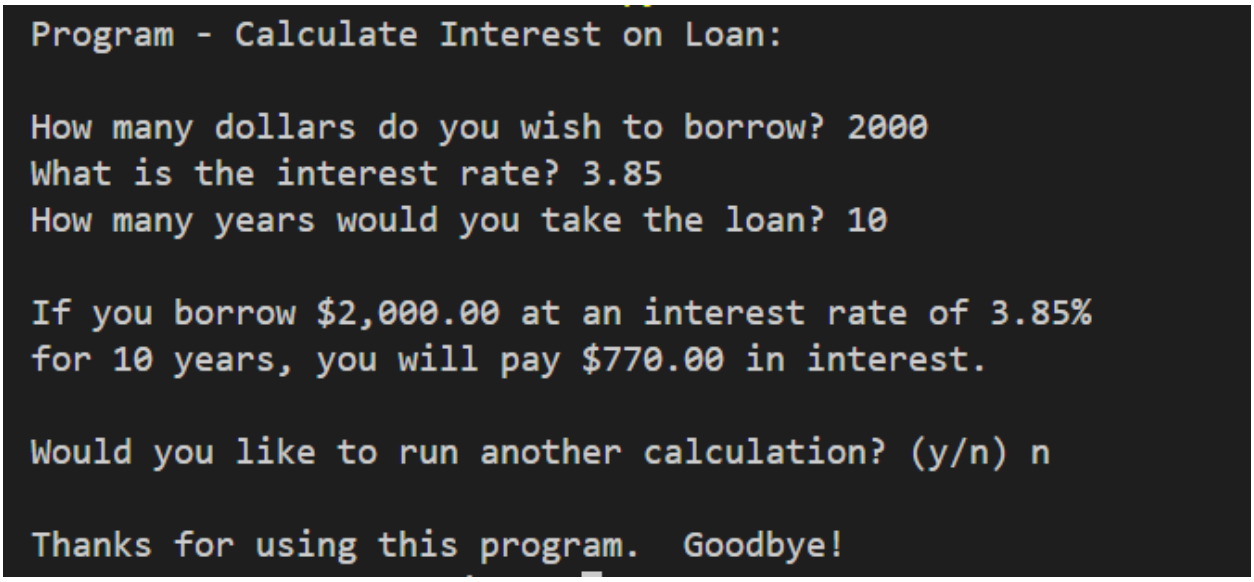
print the name of the program

set repeat flag to True

while repeat
    # prompt for loan amount and convert to num using float function
    # prompt for interest rate and convert to num using float function
    # prompt for years of loan and convert to num using int function
    # calc total interest as amount * (interest rate/100) * years
    # show output with currency formatted to 2 decimal places
    # test for again using if statement

print thanks and goodbye

Screenshot (Interest Due):



Code Examples (Interest Due):

How do you prompt in a loop?

```
while repeat:
    amount = input("\nHow many dollars do you wish to borrow? ")
    amount = float(amount)
```

How do you format for currency? (not in book – see lab 2):

```
f"\nIf you borrow ${amount:,.2f} at
```

In the while loop how do you test for again and set flag accordingly

```
    # prompt for again and set flag accordingly
    again = input("\nWould you like to run another calculation? (y/n) ")
    if again == 'n':
        repeat = False
```

VIP – Keep your Interest Due Solution: Keep your solution to this program in a safe place because you will need it again for an upcoming lab.

Food for Thought:

This lab asks you write that that accepts input from the user. What happens if you prompt for a number and the user enters a string? Try it. Your program crashes. Look at your Traceback. See the ValueError Exception? You will learn how to handle exceptions in an upcoming chapter. Hold on!

Program 2 – Southern States with Loop (southern_states_loop.py):

Problem Statement: This program will start with your lab 3 program – southern_states.py. Refer to that problem statement for a review of the logic and design of that program. Open that file and save it as southern_states_loop.py to begin your updates.

Coding: In this program you will use a for loop and a while loop to show the data in the list rather than index values.

Output: Your output should be the same as the original program. The only difference is that instead of using brute force to show the data in the list using the index values of the list, you will use loops. The first time you display the data in the list, use a for in loop. The second time you show the data, use a while loop.

Resources: Refer to chapter 4 for a discussion of the for loop and chapter 7 for discussion of the while loop. You may also want to refer to the internet for syntax examples. W3shools.com offers excellent examples of Python coding structures.

Specific Comments/Pseudocode:

initialize a list with names of the southern states all lowercase

print name of report

show unsorted list using for in loop - show title case

use neg index to access last element in list and show it in title case

sort the list using the sort method

show list again using while loop

use neg index to access last element in list

use len() function to show length of list

print credit line showing source of data

Screen shot (Southern States with Loops):

```
Report - Southern United States

UNSORTED:

Virginia
Tennessee
Arkansas
Louisiana
North Carolina
South Carolina
Mississippi
Alabama
Georgia
Florida
Texas

Last state on this unsorted list: Texas

SORTED:

Alabama
Arkansas
Florida
Georgia
Louisiana
Mississippi
North Carolina
South Carolina
Tennessee
Texas
Virginia

Last state in this ordered list: Virginia

Number of Southern States: 11

Source: simple.wikipedia.org/wiki/Southern_United_States

[Finished in 0.1s]
```

Coding Examples (Southern States with loops):

How to code a for in loop?

```
# show unsorted list using for in loop - show title case
print("UNSORTED:\n")
for state in southern_states:
    print(state.title())
```

How to code a while loop?

```
# show list again using while loop
print("SORTED: \n")
current_state = 0
while current_state < len(southern_states):
    print(f"{southern_states[current_state].title()}")
    current_state = current_state + 1
```

Submit your lab assignment:

Using the Canvas assignment tool, upload your completed work (2 files) to the lab 7 assignment. Attach the first file (interest_due.py) and then the second (southern_states_loop.py) and then submit.

All lab assignments must be submitted using the CANVAS assignments tool. Lab Assignments will not be accepted any other way**.** Make sure you submit your work to the right lab assignment number otherwise you will not get credit.

Grading Criteria:

- ✓ You must use the provided video to do the Interest Due program.
- ✓ Don't forget general comments.
- ✓ Specific comments are optional.
- ✓ Use white space to make your code easy to read.
- ✓ Comply with PEP-8 conventions for variable names, file names, etc.
- ✓ Satisfy the problem definition and other grading standards.
- ✓ Your work should not have syntax errors.
- ✓ Your work must be your own.
- ✓ Match your output screen to screenshot provided.
- ✓ Any deviations from lab specifications will result in points off.
- ✓ If you want to vary, do so on your own.

VIP Videos:

Interest Due program (step-by-step): Lab 7 Interest Due Video

Lectures and other videos: VIEW MY PYTHON PLAYLIST

Closing:

If you have questions about this lab send me a message using canvas inbox or attend zoom office hours. See Unit 0 for link for my office hours. Programming tutors are also available and listed in your Canvas class as announcements.

By-the-way, a great way to get ready for your lab assignments (and exams) is do the Try It Yourself problems in your book. Most of the solutions are on the authors website and I also discuss them in my YouTube video lectures. Another way to get ready for your lab assignments is to review my lecture notes for each chapter. In Unit 0, find the link for my chapter notes.

Warning: Labs will increase in complexity with each lab assignment. A beginner may need 5 to 15 hours to complete each lab. Please start early because there will be no extensions.

Have an exception free day! Prof. Benavides