


# Lab 9: loops continued, writing to a file

## FEEDBACK SURVEY

<https://vevox.app/#/m/147850478>  <https://vevox.app/#/m/147850478> Please take this survey based on the feedback you received from your week 7 in-lab exam. Be sure to review your feedback before taking the survey.

## STRING ITERATOR

Write a program to read a sentence from the user.

- I. Write a for loop to print the sentence but with the punctuation marks changed to 'X'.
- II. Add code to print the uppercase letters as ASCII codes (use `ord(character)`)

*note: `ord(character)` return an integer representing the Unicode code of that character, `chr(i)` returns the character (glyph) whose Unicode is the integer *i*.*

## NESTED LOOP

Use a nested loop structure to print the following stair step pattern:

```
#
#
#
#
#
#
```

*note: the amount of white space before each # symbol increases on each step.*

## WRITING TO A FILE

1. Write code to save the numbers 1 to 100 to a file called "numbers.txt".
2. Modify the program so that you ask for the file's name rather than always calling it "numbers.txt".
3. Write another program to add the numbers 101 to 200 to the same file.

## GATHERING INFORMATION IN A LOOP

1. Write a program to ask the user for the number of people for whom they must buy presents for Christmas.
  1. Add code to read the name of each person and the budget they have allocated.
  2. Use a string accumulator to save the information about each person's present
2. At the end of the program print out the present list ONCE.

## WRITING TO A FILE IN A LOOP

1. Write a program to ask the user for the number of people for whom they must buy presents for Christmas.
  1. Add code to read the name of each person and the budget they have allocated.
  2. Write this information to presents.txt.
2. Add validation code for the number of people.
3. Add a try/except to deal with issues relating to the file.

## TRY/EXCEPT

1. Create a Python file and add the following code:

```
DIVISION_SYMBOL = chr(247)
x = float(input("Numerator >>> "))
y = float(input("Denominator >>> "))
division = x / y
print(f"{x:.2f} {DIVISION_SYMBOL} {y:.2f} = {division:.4f}")
```

2. Run the program for the following test inputs:
  1. Input 5 and 6
  2. Input 4.5 and -9.8
  3. Input 5 and 0
  4. Input string instead of a number
3. Add a try/except to deal with:
  1. Division by zero error using ZeroDivisionError exception. Test with Input 5 and 0
  2. ValueError error which will be triggered if the user types a letter instead of a number. Test by typing a string instead of floating-point number
  3. Add code to write the mathematical expression to a file as well as to the console. The name of the file is your choice, adding extra exceptions as needed e.g. IOError

## EXTRA CHALLENGE

Write code to ask the user for the number of people.

- a. Add a for loop to iterate through that number of people, asking for the name of each person.

- b. Add code to also ask the user for the age of each person.
- c. The program should display all the data read in a neat table.

The following is a skeleton for this, some of which is Python but some of which is not

```
table = f'{Name:20}{Age:10}\n'
Ask for the number of people and store in number_of_people
For loop going from 1 to number_of_people (inclusive):
    Ask for and read the name
    Ask for and read the age
    table += f'{name:20}{age:>10}\n'
print(table)
```

Add code to calculate and display the average age of the people.

- d. Add code to determine the oldest age.
- e. Add code to also display the name of the oldest person as well as their age. This requires another variable which will receive a value in the if statement checking the oldest age.
- f. Add code to deal with two people who might have the same age (Hint: consider using a string accumulator to store the names of all those who are the oldest and add to this string in an if statement that determines if the persons\_age == oldest\_age).
- g. Add code to the previous program to find the youngest person.
- h. Add code to validate the number of people as a positive integer. Recall that this cannot be done using a for loop – use a while loop like that from last week.