Name: Pagadala Sahil Naidu

Subject: Neural Networks & Deep Learning

Course ID: CS-5720

Email: [sxp86940@ucmo.edu](mailto:sxp86940@ucmo.edu)

CRN: 22317

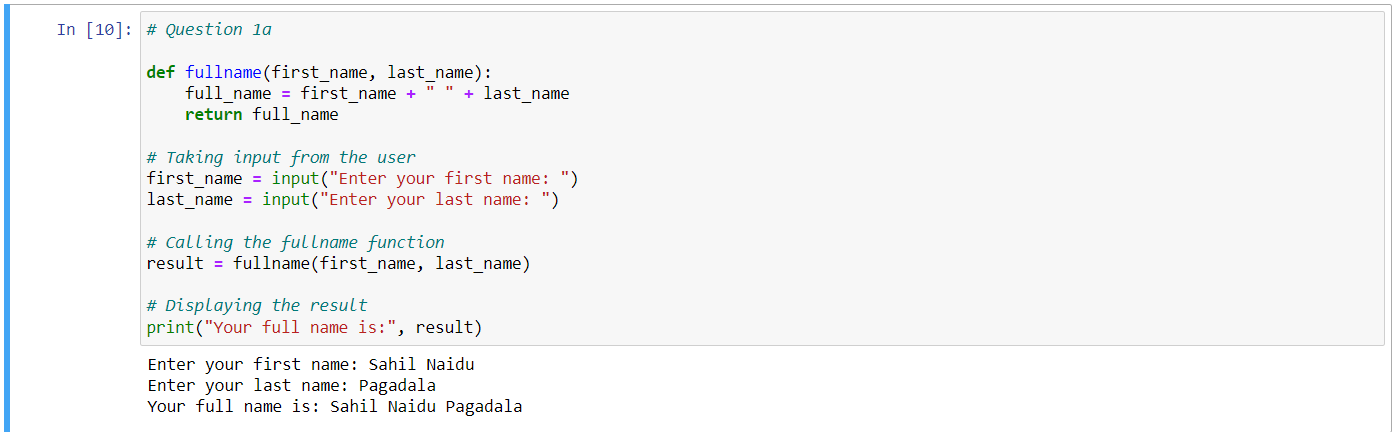
**Click the ICP2 link to access the assignment in GitHub.**

[**https://github.com/Sahilnaidupagadala03/Neural\_Networks\_Deeplearning**](https://github.com/Sahilnaidupagadala03/Neural_Networks_Deeplearning)

**Below is Voice over video.**

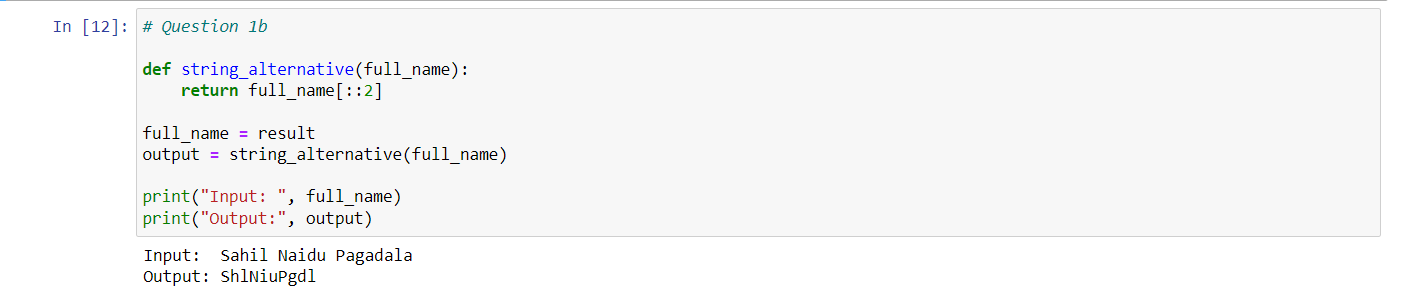
[**https://youtu.be/7FIybRw69Ds**](https://youtu.be/7FIybRw69Ds)

**Question 1-a**



The fullname method in this Python code accepts two inputs, first\_name and last\_name, and concatenates them with a space to get the full name. The entire name is then returned by the function. After that, the code uses the input function to ask the user to enter their first and last names. Using the supplied first and last names, it invokes the fullname function, storing the outcome in a variable named result. Lastly, it uses the print function to output the entire name.

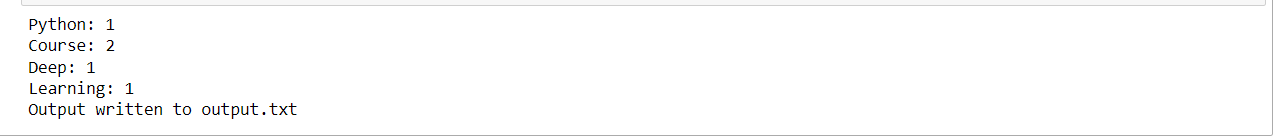
**Question 1-b**



string\_alternative, which accepts a whole name as input and outputs a different character set from the entire name. In particular, it includes every second character in the complete name using slicing with a step size of 2 (full\_name[::2]). The entire name from the preceding code, which was the result previously acquired, is then assigned to the variable full\_name by the code. It uses this complete name to invoke the string\_alternative function, and it saves the outcome in an output variable. Lastly, it uses the print function to print the result (alternative characters) together with the original complete name.

**Question 2**

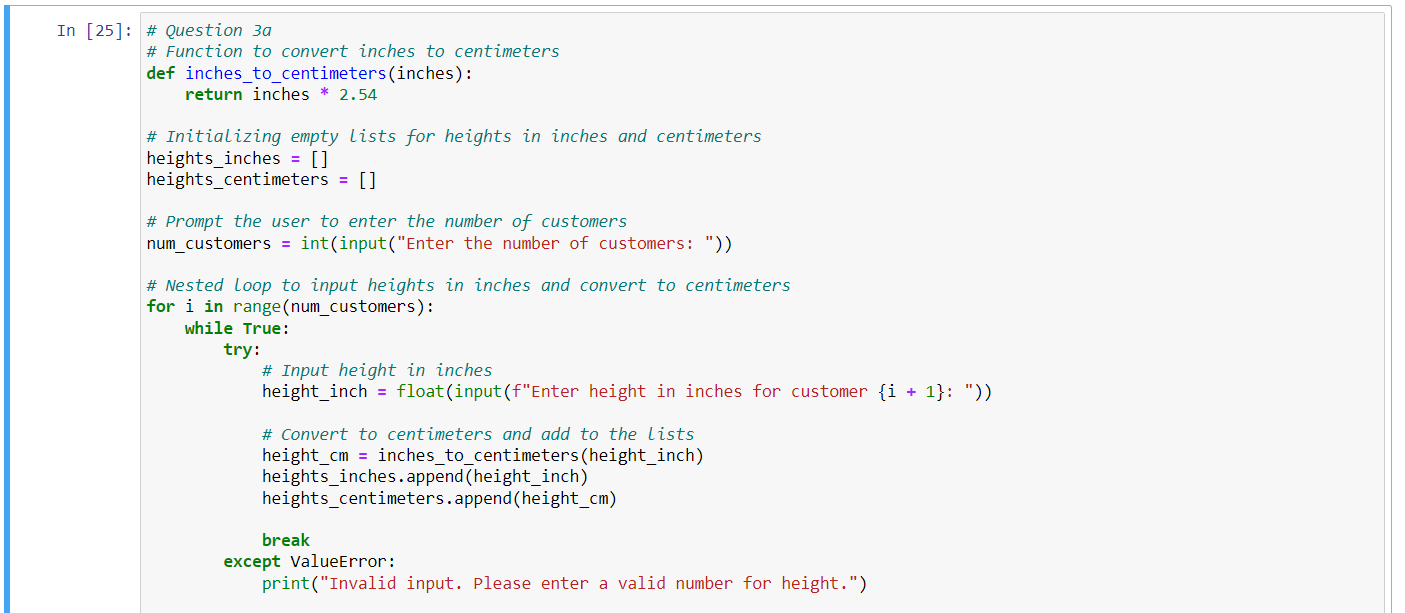
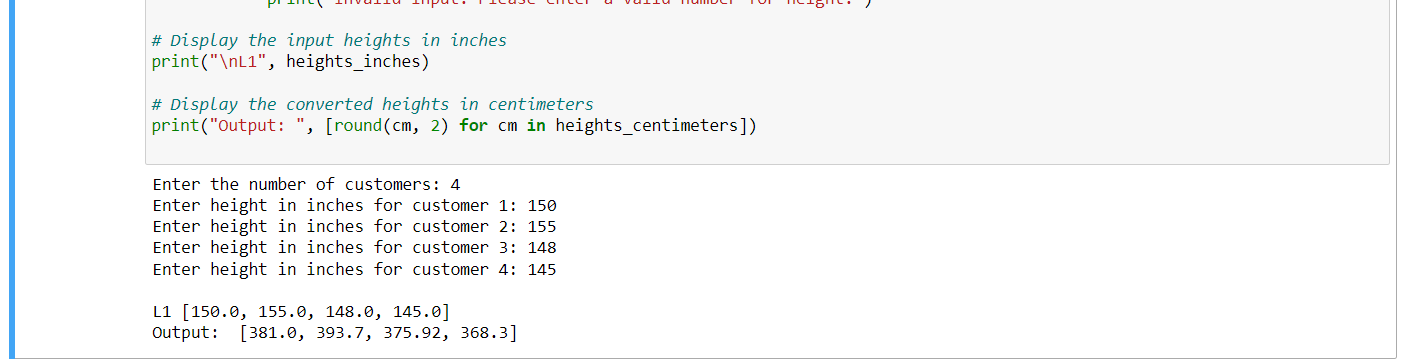




wordcount\_per\_line is a function that accepts the names of an input file (input file) and an output file (output file). The function reads every line from the input file, counts how many times each word appears, and then copies the original lines to the output file along with the word counts. To guarantee correct file handling, the function makes an attempt to open and read the contents of the input file using a with statement. It produces an error message and raises the FileNotFoundError if the file cannot be located. The word frequencies are stored inside the function by initializing an empty dictionary word\_count. After that, it reads each line from the input file one at a time, separates it into words, and sends the line to the output file. Every word has no punctuation letters (',', ':', '.', '?', '!'). The code adds each word's occurrences to the word count dictionary. The function writes the word counts to the output file under the "Word\_Count" section after processing each line. The word counts are also printed to the console by the code. An error notice is printed if the input file cannot be located. The wordcount\_per\_line function is called by the script with the input and output file names ("input.txt" and "output.txt") specified. The output has been written to the designated output file, as shown by the message that is printed at the end.

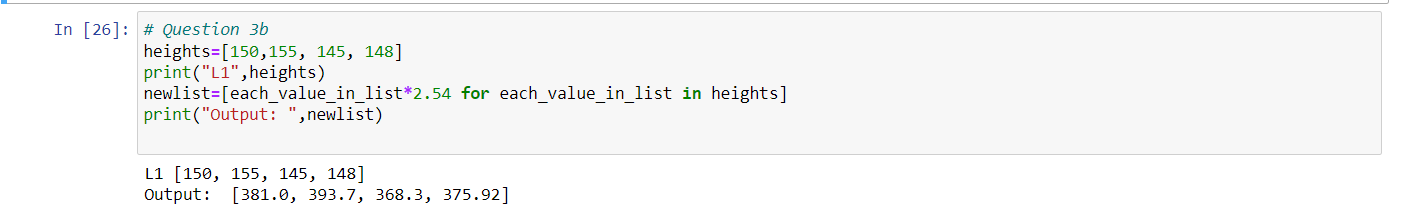


**Question 3-a**

**  
**

inches\_to\_centimeters function, returns the matching height in centimeters after receiving an input of height in inches. Multiplying the input by the conversion factor (2.54) completes the conversion. The heights in inches and centimeters are stored in the two empty lists, heights\_inches and heights\_centimeters, respectively. The user is asked to input the number of customers (num\_customers) that require the conversion of heights. Each customer's height in inches is requested to be entered by the user. An attempt is made to convert the input to a float using a try block. If successful, the height in inches and centimeters are added to the corresponding lists, and the height in centimeters is computed using the inches\_to\_centimeters function. A ValueError is raised and the user is asked to submit a valid number if the input is not a valid float. The code outputs the input heights in inches (heights\_inches) and the converted heights in centimeters (heights\_centimeters) once all input has been processed. When printing, the centimeter values are rounded to two decimal places.

**Question 3-b**

****

The code begins with a heights list ([150, 155, 145, 148]) with the values of the heights in inches. It uses the print function to print the original list of heights in inches, inserting the label "L1" before the list. The code generates a new list (newlist) by using a list comprehension. Every value in the old list (heights) is iterated over, multiplied by the conversion factor from inches to centimeters, which is 2.54, and the results are stored in the new list. It uses the print function to print the updated list of heights in cm, with the label "Output" appearing before the list. The conversion produced these numbers, which are rounded to two decimal places.