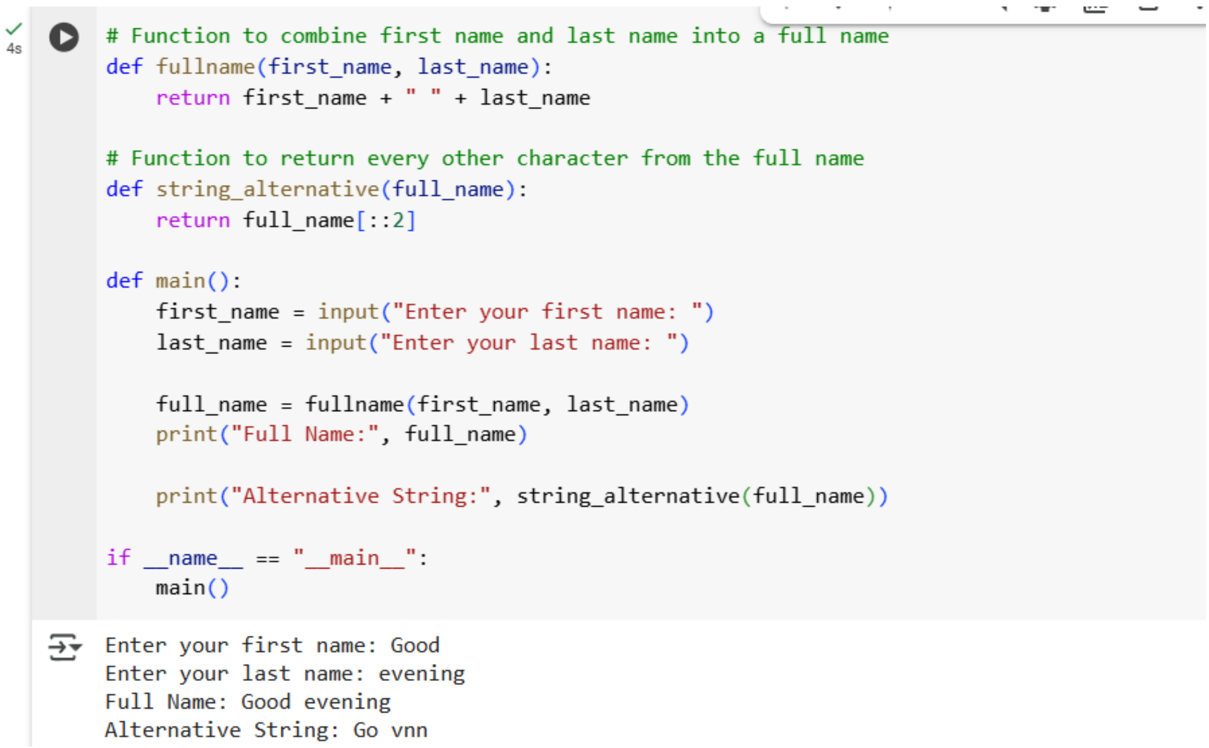
Neural Networks and Deep Learning - ICP-1

Name: Usha Kiran Reddy Patlolla
Student Id: 700764998

1. Write a program that takes two strings from the user: first_name, last_name. Pass these variables to fullname function that should return the (full name).

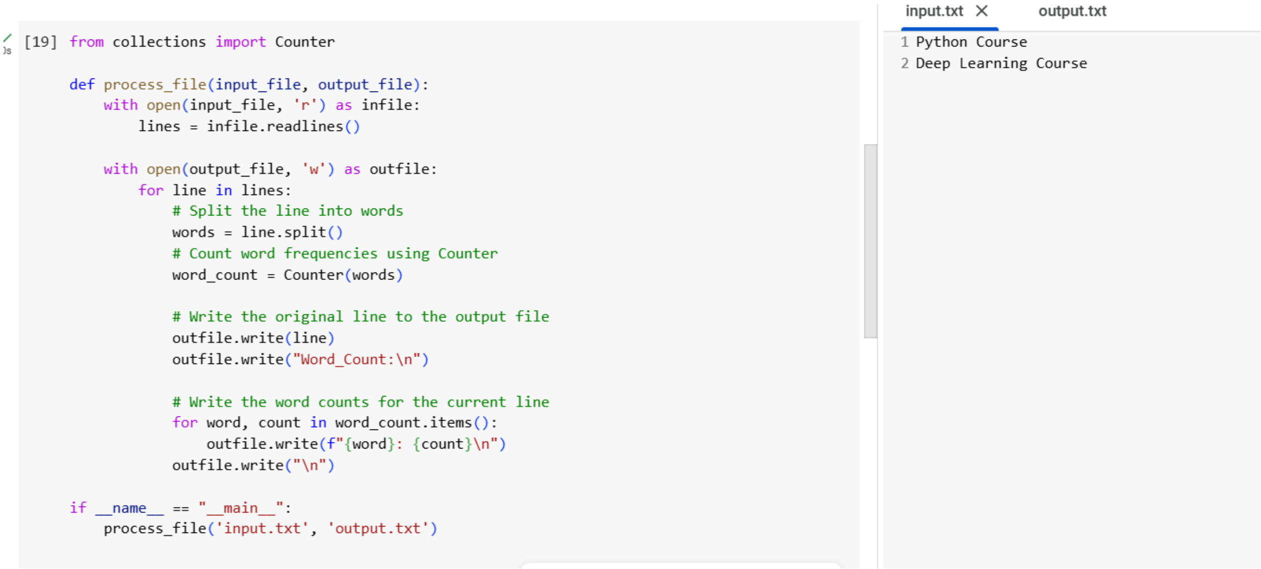
   Description: This program prompts the user to input their first and last names. It defines two functions: full name combines the first and last names into a full name, and string_alternative returns every other character of the full name using string slicing. The program displays both the full name and the alternative string with every other character removed. The logic is executed within the main function, which handles user input and calls the other functions to display the results.



```
# Function to combine first name and last name into a full name
def fullname(first_name, last_name):
    return first_name + " " + last_name

# Function to return every other character from the full name
def string_alternative(full_name):
    return full_name[::2]

def main():
    first_name = input("Enter your first name: ")
    last_name = input("Enter your last name: ")

    full_name = fullname(first_name, last_name)
    print("Full Name:", full_name)

    print("Alternative String:", string_alternative(full_name))

if __name__ == "__main__":
    main()
```

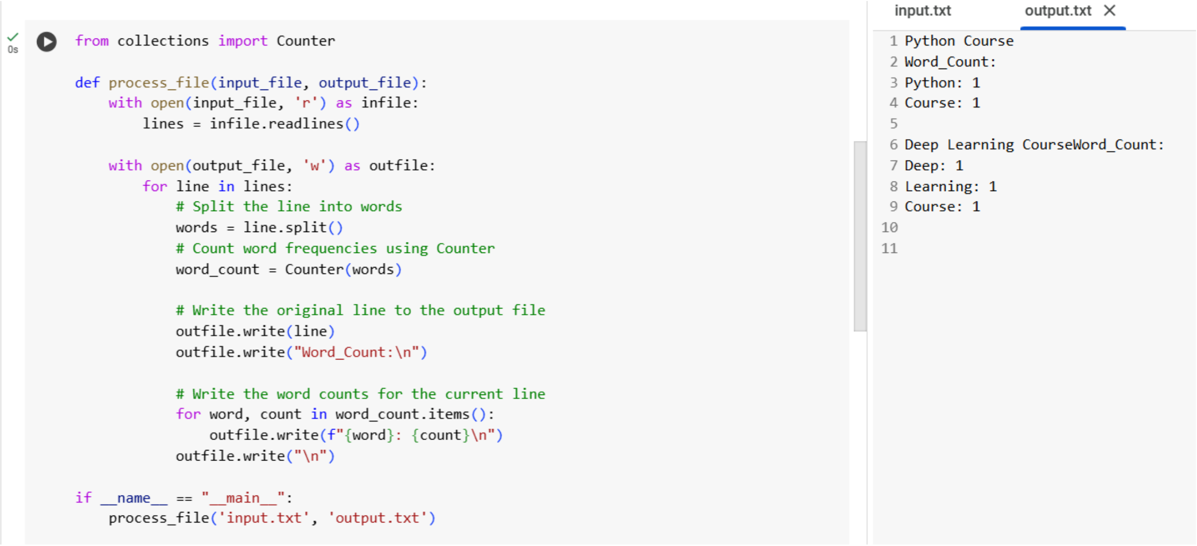
```
Enter your first name: Good
Enter your last name: evening
Full Name: Good evening
Alternative String: Go vnn
```

2. Write a python program to find the wordcount in a file (input.txt) for each line and then print the output. o Finally store the output in output.txt file.
   Description: This Python program processes a text file by reading each line, counting the frequency of each word using the Counter class from the collection’s module, and then writing both the original line and the word frequencies to a new output file. It first opens the input file, reads the lines, and splits each line into words. The Counter is used to tally the occurrences of each word in the line. Then, the program writes the original line to the output file, followed by the word counts for that line, formatted as "word: count". Finally, the output file contains the original text along with word frequencies for each line. The program is designed to handle multiple lines in the input file and produce a corresponding word count summary for each.



```python
from collections import Counter

def process_file(input_file, output_file):
    with open(input_file, 'r') as infile:
        lines = infile.readlines()

    with open(output_file, 'w') as outfile:
        for line in lines:
            # Split the line into words
            words = line.split()
            # Count word frequencies using Counter
            word_count = Counter(words)

            # Write the original line to the output file
            outfile.write(line)
            outfile.write("Word_Count:\n")

            # Write the word counts for the current line
            for word, count in word_count.items():
                outfile.write(f"{word}: {count}\n")
            outfile.write("\n")

if __name__ == "__main__":
    process_file('input.txt', 'output.txt')
```

```
Python Course
Deep Learning Course
```

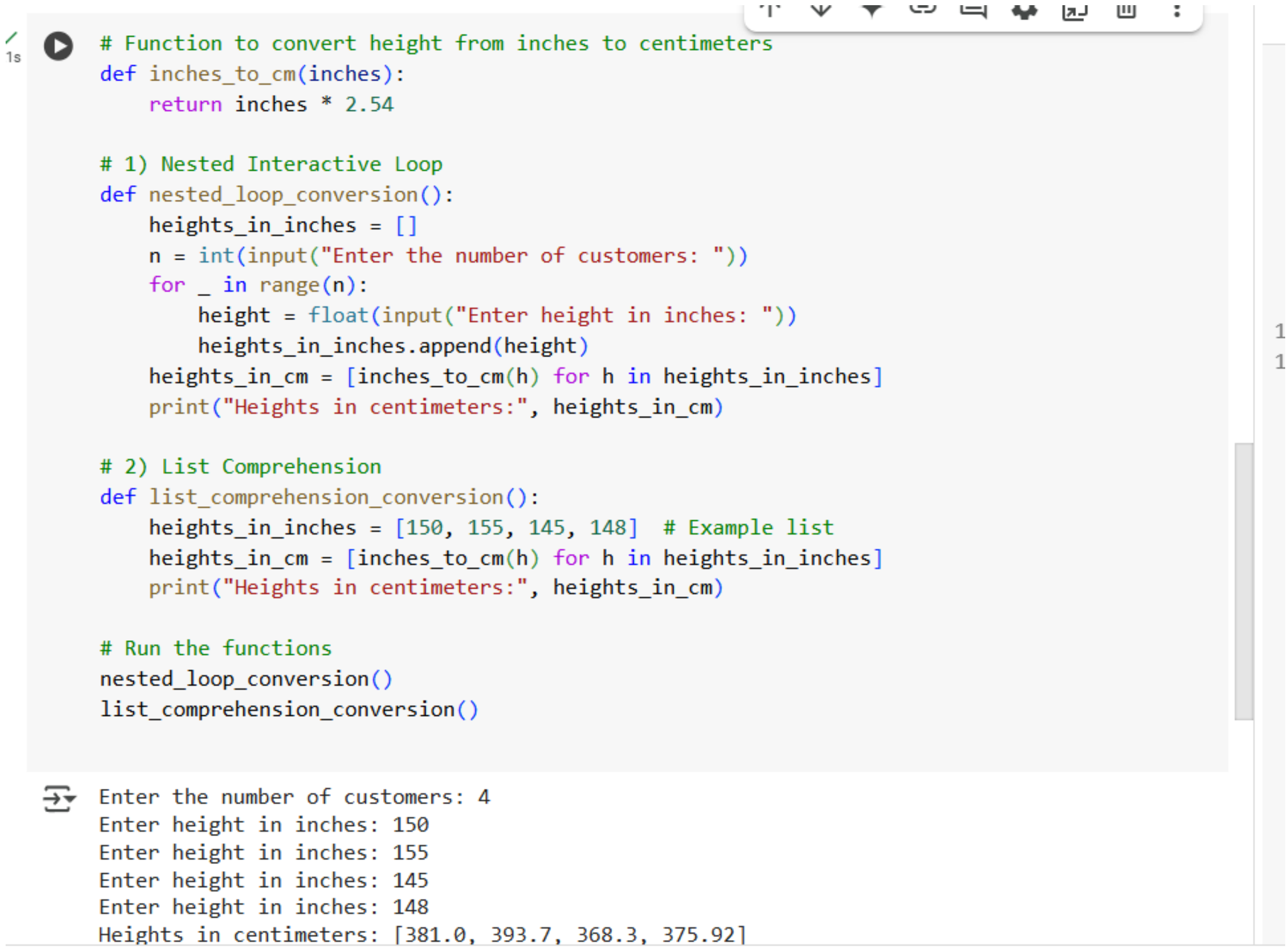


```python
from collections import Counter

def process_file(input_file, output_file):
    with open(input_file, 'r') as infile:
        lines = infile.readlines()

    with open(output_file, 'w') as outfile:
        for line in lines:
            # Split the line into words
            words = line.split()
            # Count word frequencies using Counter
            word_count = Counter(words)

            # Write the original line to the output file
            outfile.write(line)
            outfile.write("Word_Count:\n")

            # Write the word counts for the current line
            for word, count in word_count.items():
                outfile.write(f"{word}: {count}\n")
            outfile.write("\n")

if __name__ == "__main__":
    process_file('input.txt', 'output.txt')
```

```
Python Course
Word_Count:
Python: 1
Course: 1

Deep Learning CourseWord_Count:
Deep: 1
Learning: 1
Course: 1


```

3. Write a program, which reads heights (inches.) of customers into a list and convert these heights to centimetres in a separate list using: 1) Nested Interactive loop. 2) List comprehensions.

Description: This Python program provides two methods to convert customer heights from inches to centimetres. The first method, **Nested Interactive Loop**, allows the user to input the number of customers and their corresponding heights in inches. It then converts these heights to centimetres and displays the results. The second method, **List Comprehension**, uses a predefined list of heights in inches and applies a more concise approach to convert the heights to centimetres using list comprehension. In both methods, the inches_to_cm function is used to perform the conversion by multiplying the given height in inches by the conversion factor (2.54). The program then prints the converted heights in centimetres for each method.



```python
# Function to convert height from inches to centimeters
def inches_to_cm(inches):
    return inches * 2.54

# 1) Nested Interactive Loop
def nested_loop_conversion():
    heights_in_inches = []
    n = int(input("Enter the number of customers: "))
    for _ in range(n):
        height = float(input("Enter height in inches: "))
        heights_in_inches.append(height)
    heights_in_cm = [inches_to_cm(h) for h in heights_in_inches]
    print("Heights in centimeters:", heights_in_cm)

# 2) List Comprehension
def list_comprehension_conversion():
    heights_in_inches = [150, 155, 145, 148]  # Example list
    heights_in_cm = [inches_to_cm(h) for h in heights_in_inches]
    print("Heights in centimeters:", heights_in_cm)

# Run the functions
nested_loop_conversion()
list_comprehension_conversion()
```

```
Enter the number of customers: 4
Enter height in inches: 150
Enter height in inches: 155
Enter height in inches: 145
Enter height in inches: 148
Heights in centimeters: [381.0, 393.7, 368.3, 375.92]
```