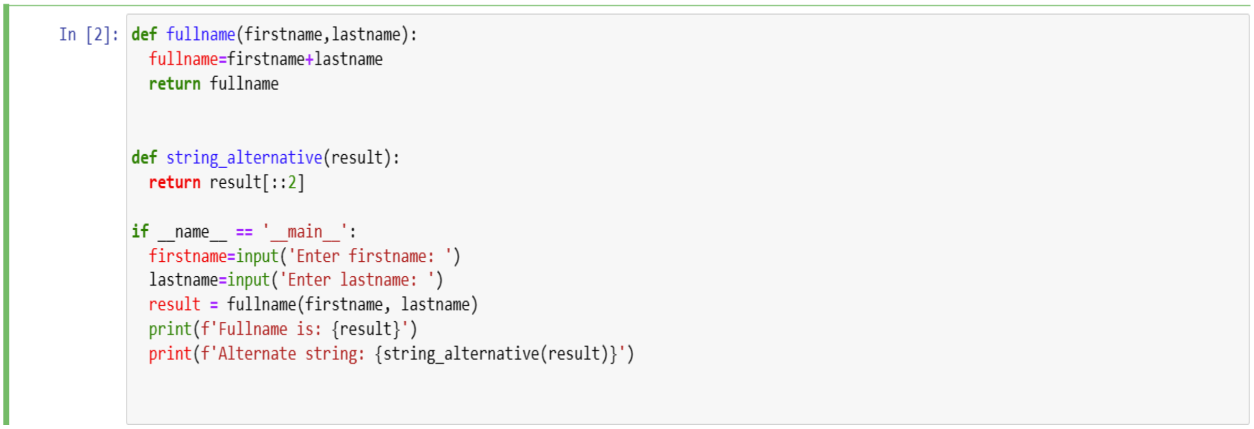
Neural Network Deep Learning (23442)

Assignment_2

https://github.com/niteesh0301/Assignment_2.git

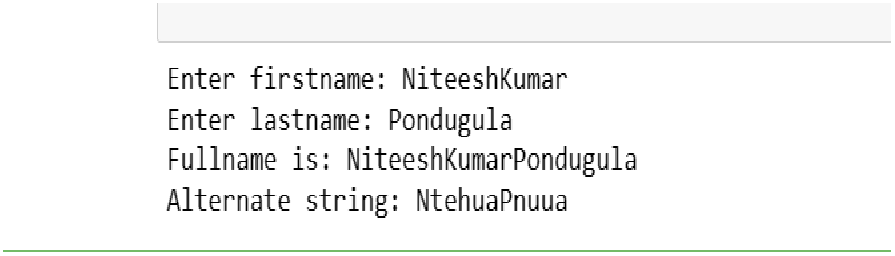
1(a) Solution:-

Code:-



```
In [2]: def fullname(firstname,lastname):
          fullname=firstname+lastname
          return fullname


        def string_alternative(result):
          return result[::2]

        if __name__ == '__main__':
          firstname=input('Enter firstname: ')
          lastname=input('Enter lastname: ')
          result = fullname(firstname, lastname)
          print(f'Fullname is: {result}')
          print(f'Alternate string: {string_alternative(result)}')
```

Output:-

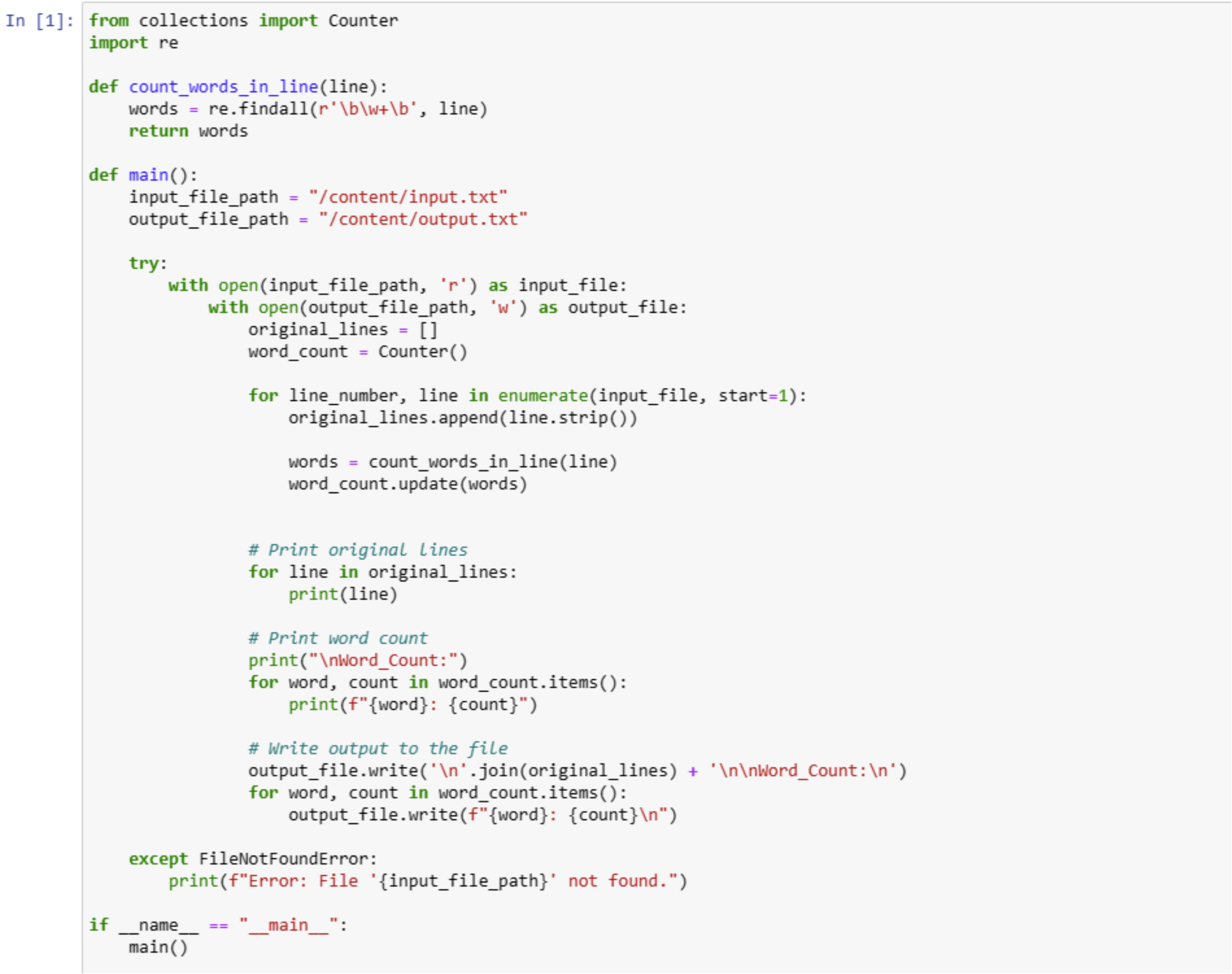


```
Enter firstname: NiteeshKumar
Enter lastname: Pondugula
Fullname is: NiteeshKumarPondugula
Alternate string: NtehuaPnuua
```

Explanation :- The actual output will depend on the user input. In this example, the script prompts the user to enter their first and last names, then prints the full name and an alternate string derived from it.

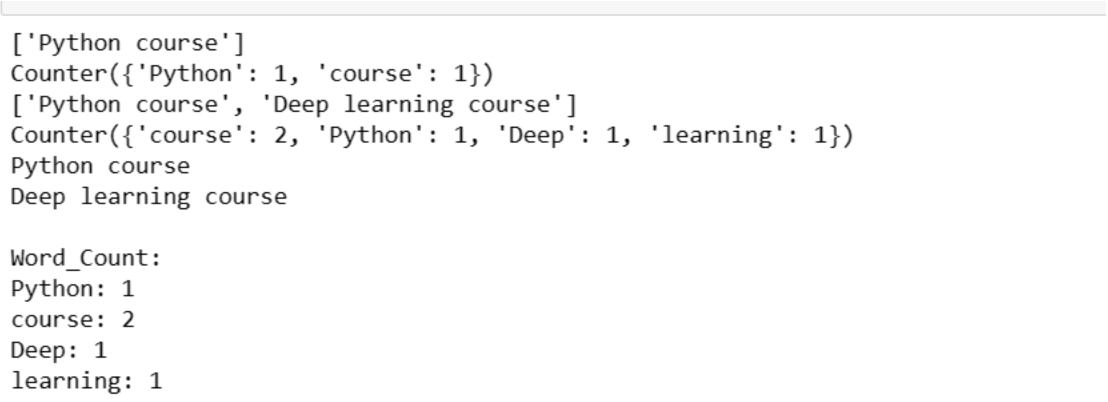
2(a) Solution:-

Code:-



```
In [1]: from collections import Counter
        import re

        def count_words_in_line(line):
            words = re.findall(r'\b\w+\b', line)
            return words

        def main():
            input_file_path = "/content/input.txt"
            output_file_path = "/content/output.txt"

            try:
                with open(input_file_path, 'r') as input_file:
                    with open(output_file_path, 'w') as output_file:
                        original_lines = []
                        word_count = Counter()

                        for line_number, line in enumerate(input_file, start=1):
                            original_lines.append(line.strip())

                            words = count_words_in_line(line)
                            word_count.update(words)


                        # Print original lines
                        for line in original_lines:
                            print(line)

                        # Print word count
                        print("\nWord_Count:")
                        for word, count in word_count.items():
                            print(f"{word}: {count}")

                        # Write output to the file
                        output_file.write('\n'.join(original_lines) + '\n\nWord_Count:\n')
                        for word, count in word_count.items():
                            output_file.write(f"{word}: {count}\n")

            except FileNotFoundError:
                print(f"Error: File '{input_file_path}' not found.")

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

Output:-

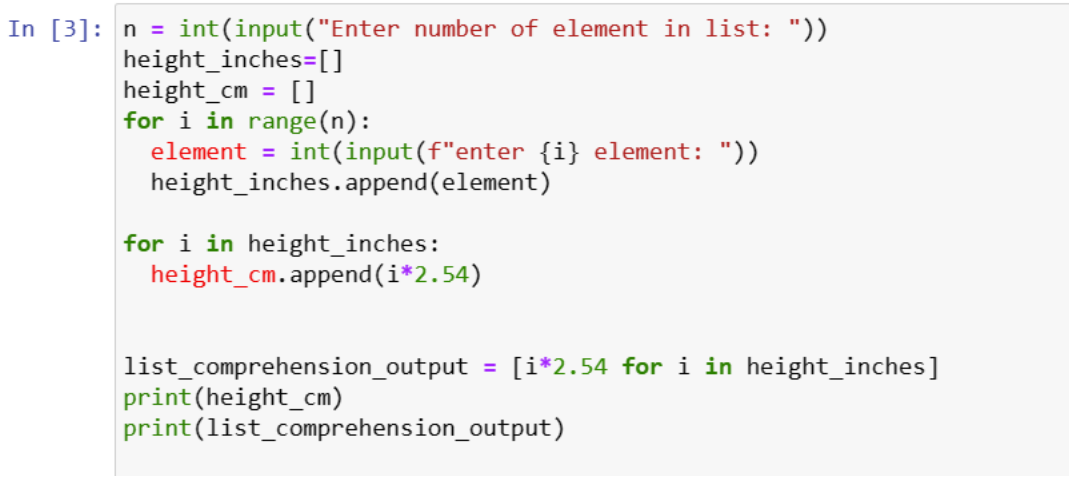


```
['Python course']
Counter({'Python': 1, 'course': 1})
['Python course', 'Deep learning course']
Counter({'course': 2, 'Python': 1, 'Deep': 1, 'learning': 1})
Python course
Deep learning course

Word_Count:
Python: 1
course: 2
Deep: 1
learning: 1
```

Explanation :- This Python script reads text from an input file (input.txt), tokenizes the words in each line, counts the occurrences of each word using the Counter class from the collections module, and then prints the original lines, word count, and writes the same information to an output file (output.txt). The script handles file errors and utilizes regular expressions (re) to extract words. The input and output file paths are specified, and the main functionality is encapsulated in the main() function, executed only if the script is run directly.

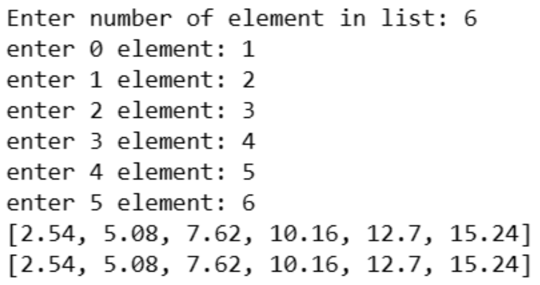
3(a) Solution:-

Code:-



```python
n = int(input("Enter number of element in list: "))
height_inches=[]
height_cm = []
for i in range(n):
  element = int(input(f"enter {i} element: "))
  height_inches.append(element)

for i in height_inches:
  height_cm.append(i*2.54)


list_comprehension_output = [i*2.54 for i in height_inches]
print(height_cm)
print(list_comprehension_output)
```

Output:-



```
Enter number of element in list: 6
enter 0 element: 1
enter 1 element: 2
enter 2 element: 3
enter 3 element: 4
enter 4 element: 5
enter 5 element: 6
[2.54, 5.08, 7.62, 10.16, 12.7, 15.24]
[2.54, 5.08, 7.62, 10.16, 12.7, 15.24]
```

Explanation :- This Python script takes user input for the number of elements in a list and then prompts the user to enter each element (assumed to be in inches). It calculates the corresponding heights in centimeters using a loop and list comprehension. The results are stored in two separate lists, height_cm using a traditional loop and list_comprehension_output using list comprehension. Finally, the script prints both lists containing the heights in centimeters. The conversion factor from inches to centimeters (2.54) is applied in both cases.