NEURAL NETWORK DEEP LEARNING
ICP_2_SPRING24 ASSIGNMENT- 2

NAME : JAYADEEP NAGUBATHULA
UCM ID: 700747518

GITHUB LINK:
https://github.com/JayadeepNagubathula/ICP_2_SPRING24

1.Write a program that takes two strings from the user: first name, last name. Pass these variables to full name function that should return the (full name). o For example: ▪ First name = “your first name”, last_name = “your last name” ▪ Full_name = “your full name”
1a) Write function named “string_alternative” that returns every other char in the full_name string. Str = “Good evening” Output: Go vnn

Source code:

```
def fullname(first_name,last_name):
    z=first_name + " " + last_name
    return z
def string_alternative(x):
    return x[::2]

user_name=str(input("enter the first name:"))
user_name2=str(input("enter the last name:"))
user_full_name = fullname(user_name, user_name2)


print(user_full_name)

def main():
    alterstring = string_alternative(user_full_name)
    print(alterstring)

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

Output:

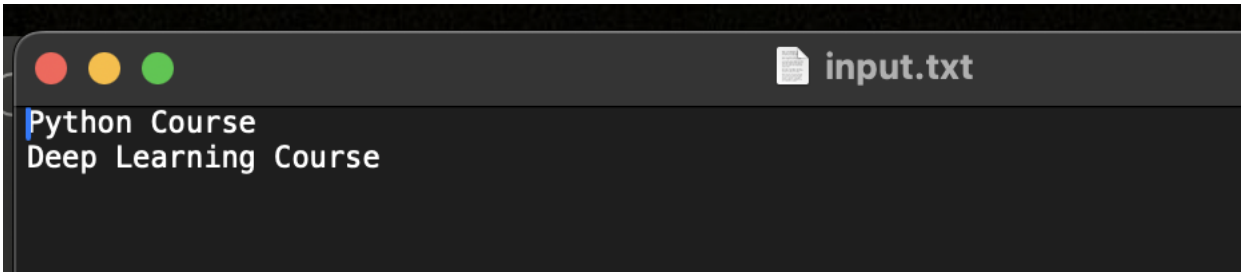
```
========================================
enter the first name:Good
enter the last name:Evening
Good Evening
Go vnn
```

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. Example: Input: a file includes two lines: Python Course Deep Learning Course Output: Python Course Deep Learning Course Word_Count: Python: 1 Course: 2 Deep: 1 Learning: 1

Source code:

```python
import os

def count_words(line):
    words = line.split()
    word_count = {}
    for word in words:
        word_count[word] = word_count.get(word, 0) + 1
    return word_count

# Get the current working directory
current_directory = os.getcwd()

# Set the input file path relative to the current directory
input_file_path = os.path.join(current_directory, 'input.txt')

with open(input_file_path, 'r') as file:
    content = file.read()
    Ae = [line.strip() for line in content.split('\n')]

# Concatenate stripped elements into a single string
concatenated_string = " ".join(element.strip() for element in Ae)

# Strip again and get word count
word_count = count_words(concatenated_string)

# Set the output file path relative to the current directory
output_file_path = os.path.join(current_directory, 'output.txt')

# Write the content and word count to an output file
with open(output_file_path, 'w') as output_file:
    output_file.write("Input Content:\n")
    output_file.write(content + '\n\nWord_Count:\n')
    for word, count in word_count.items():
        output_file.write(f'{word}: {count}\n')

# Print the content of the input file
print("Content in file:")
print(content)

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

print(f"\nOutput written to {output_file_path}")
```

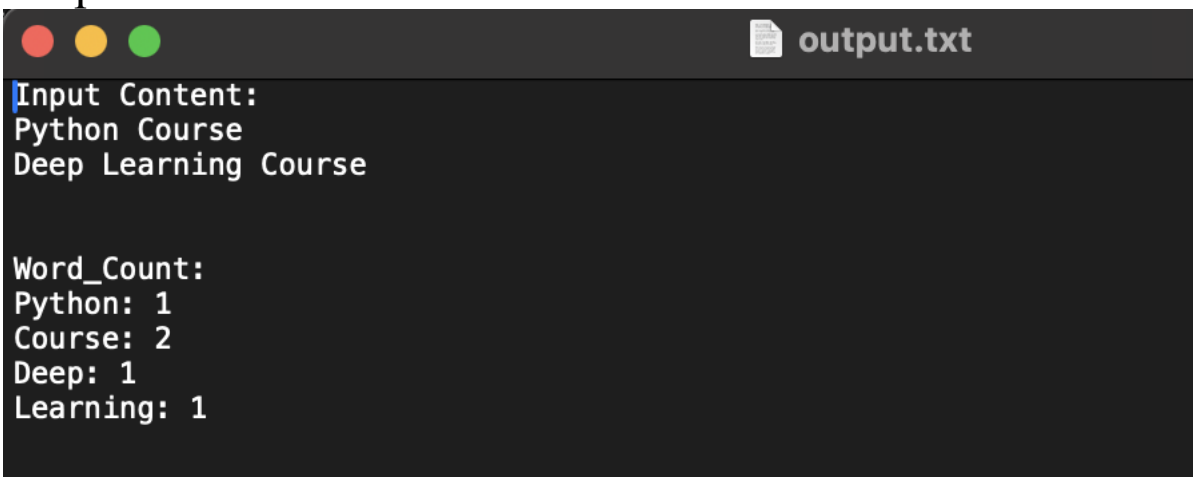
Input file:



Output:

```
========== RESTART: /Users/rev
Content in file:
Python Course
Deep Learning Course


Word_Count:
Python: 1
Course: 2
Deep: 1
Learning: 1

Output written to /Users/revanthjasti/Desktop/ICP_2_SPRING2024/output.txt
```

Output file:



3.Write a program, which reads heights (inches.) of customers into a list and convert these heights to centimeters in a separate list using: 1) Nested Interactive loop. 2) List comprehensions Example: L1: [150,155, 145, 148] Output: [68.03, 70.3, 65.77, 67.13]

Source code:

```
inches =[]
print ("Enter heights in inches. Type 'done' when finished.")
while True:
    x = input("Enter height in inches(or 'done' to finish): ")
    if x.lower() == 'done':
        break
    try:
        height_inch = float(x)
        inches.append(height_inch)
    except ValueError:
            print("Invalid input. Please enter a valid number or 'done'.")
centimeters = [height * 2.54 for height in inches]
print("Heights in Inches:", inches)
print("Heights in centimeters:", centimeters)
```

Output:

```
================================================================== RI
Enter heights in inches. Type 'done' when finished.
Enter height in inches(or 'done' to finish): 123.45
Enter height in inches(or 'done' to finish): 155.56
Enter height in inches(or 'done' to finish): 176.87
Enter height in inches(or 'done' to finish): done
Heights in Inches: [123.45, 155.56, 176.87]
Heights in centimeters: [313.563, 395.1224, 449.2498]
```