Neural Network & Deep Learning ICP 2

Student name: SRINIVAS MUSINURI

Student Id: 700758813

Git Hub Link:

https://github.com/srinivasmusinuri/700758813_NNDL_ICP2

Video Link:

https://drive.google.com/file/d/1IazF25l0M76OdenpPaccMzT30mHTcoWr/view?usp=drive_link

1. a. 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).

```
# 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).

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

    # calling the fullname function here by passing inputs
    full_name = fullname(first_name, last_name)

    print("Full Name :", full_name)

main()
```

```
[51]
    def fullname(first_name, last_name):
        full_name = first_name + " " + last_name
        return full_name
```

Input & Output:

```
Enter your first name: srinivas
Enter your last name: musinuri
Full Name : srinivas musinuri
```

1.b. Write function named “string_alternative” that returns every other char in the full_name string.

```
[61] # 1.b  Write function named “string_alternative” that returns every other char in the full_name string.

    def main():

      # Enter fullname as input
        full_name = input("Enter your full name: ")

      # calling string_alternative function
        filteredStr = string_alternative(full_name)

      # print the result
        print("Alternative string :", filteredStr)

    main()
```

```
def string_alternative(inputStr):
    resultStr = ""

  # This for loop iterates through the indices of the input_string starting from index 0 and increasing by 2 in each step
    for i in range(0, len(inputStr), 2):
        resultStr += inputStr[i]
    return resultStr
```

Input & Output:

```
Enter your full name: srini
Alternative string : sii
```

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.

```
# 2 Write a python program to find the wordcount in a file (input.txt) for each line and then print the output.
    # Finally store the output in output.txt file.

  # give the path of both input and output text files

input_file = "input.txt"
output_file = "output.txt"
```

```
# the code opens the input file in read mode to read all lines,

with open(input_file, "r") as file:
    lines = file.readlines()

all_text = " ".join(lines)
word_count = count_words(all_text)
```

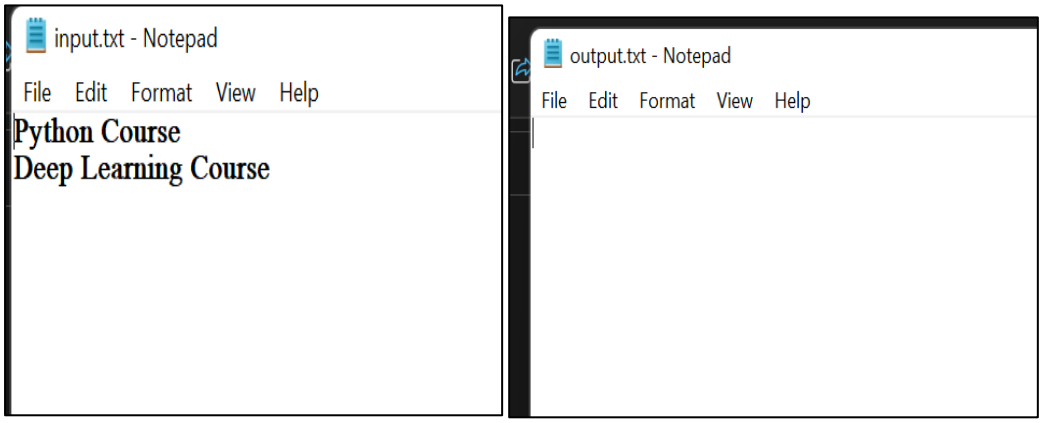
```
# 'count_words' function takes input string and returns a dictionary containing word and respective count
def count_words(input_txt):
    word_count = {}
    words = input_txt.split()
    for word in words:
        word = word.strip()
        if word:
            if word in word_count:
                word_count[word] += 1
            else:
                word_count[word] = 1
    return word_count
```

```
writing_lines = []
# Append individual input lines here, by removing the trailing spaces
for line in lines:
    writing_lines.append(line.strip())

writing_lines.append("Word_Count:")

# iterate through dictionary items and append word and respective count
for line, line_count in word_count.items():
    writing_lines.append(f"{line}: {line_count}")
```

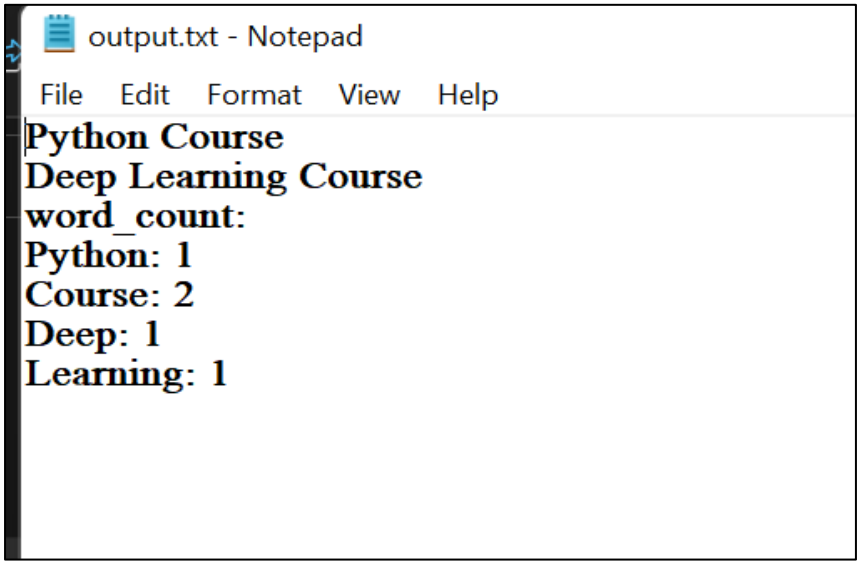
```
# the code opens the output file in write mode to write all lines,separated by newline

with open(output_file, "w") as file:
    file.write("\n".join(writing_lines))

print("Output has been written to 'output.txt' file.")
```

Input & Output:

Input file with text and empty output file before running the code:



Output file with result:



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

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

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

def main():

  # Enter customer count to take input
    cust_count = int(input("Enter the number of customers: "))
    inch_hyts = []

    # A.  Read heights in inches using nested loop
```

```
  # A.  Read heights in inches using nested loop
  for i in range(cust_count):
      hyt = float(input(f"Enter customer height {i+1} (in inches): "))
      inch_hyts.append(hyt)

  # Convert heights to centimeters using nested loop
  heights_cm = []
  for hyt in inch_hyts:

    # calling funtion for convertion
      cm_hyt = inches_to_cm(hyt)
      heights_cm.append(cm_hyt)
```

```
    # B.  Convert heights to centimeters using list comprehension
    heights_comp = [inches_to_cm(height) for height in inch_hyts]

  # print result
    print("customer heights  in centimeters (nested loop):", heights_cm)
    print("customer heights  in centimeters (list-comprehension):", heights_comp)

main()
```

Input & Output:

```
Enter the number of customers: 2
Enter customer height 1 (in inches): 55
Enter customer height 2 (in inches): 65
customer heights  in centimeters (nested loop): [139.7, 165.1]
customer heights  in centimeters (list-comprehension): [139.7, 165.1]
```