

Spring 2024: CS5720

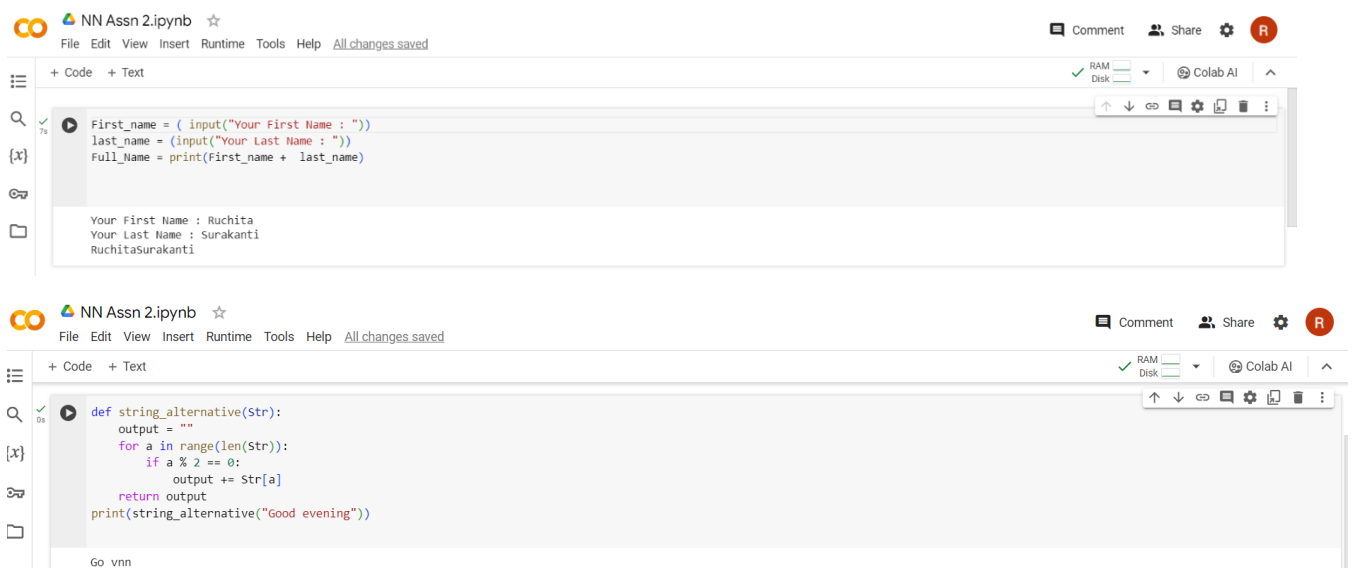
Neural Networks and Deep Learning - ICP-2

NAME: Ruchita Reddy Surakanti

700#: 700753219

Github link :

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). o For example: ▪ First_name = "your first name", last_name = "your last name" ▪ Full_name = "your full name" o Write function named "string_alternative" that returns every other char in the full_name string. Str = "Good evening" Output: Go vnn Note: You need to create a function named "string_alternative" for this program and call it from main function.



The image shows two screenshots of a Jupyter Notebook interface. The top screenshot shows a code cell with the following Python code:

```
First_name = (input("Your First Name : "))
last_name = (input("Your Last Name : "))
Full_Name = print(First_name + last_name)
```

The output of the code is:

```
Your First Name : Ruchita
Your Last Name : Surakanti
RuchitaSurakanti
```

The bottom screenshot shows a code cell with the following Python code:

```
def string_alternative(Str):
    output = ""
    for a in range(len(Str)):
        if a % 2 == 0:
            output += Str[a]
    return output
print(string_alternative("Good evening"))
```

The output of the code is:

```
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

File Edit View Insert Runtime Tools Help All changes saved

Files

- sample_data
 - README.md
 - anscombe.json
 - california_housing_test.csv
 - california_housing_train.csv
 - mnist_test.csv
 - mnist_train_small.csv
 - input.txt

```

file1 = open('input.txt', 'r')
counts = dict()
data = file1.read()
words = data.split()
for word in words:
    if word in counts:
        counts[word] += 1
    else:
        counts[word] = 1
print(counts)
f = open('output.txt', 'w')
f.write(data)
f.write('\nword_count:\n')
for key, value in counts.items():
    f.write(f'{key}: {value}\n')
f.close()

```

['Python': 1, 'Course': 2, 'Deep': 1, 'Learning': 1]

- 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]

File Edit View Insert Runtime Tools Help

Files

- sample_data
 - README.md
 - anscombe.json
 - california_housing_test.csv
 - california_housing_train.csv
 - mnist_test.csv
 - mnist_train_small.csv
 - input.txt
 - output.txt

```

L1=list(map(float,input().split()))
L2=[]
for x in L1:
    x=x*2.54
    L2.append(x)
print(L2)

3.2

L1=list(map(float,input().split()))
L2=[x*2.54 for x in L1]
print(L2)

```

150 155 145 148
[381.0, 393.7, 368.3, 375.92]