

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).

For example:

■ First_name = "your first name", last_name = "your last name"

■ Full_name = "your full name"

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.

```
def fullname(first,last):  
    str=first+" "+last  
    return str  
def string_alternative(final):  
    newstr=final[::2]  
    return newstr  
first=input()  
last=input()  
final=fullname(first,last)  
print(final)  
alternate=string_alternative(final)  
print(alternate)
```

```
MaanyaRaj  
Yata
```

```
MaanyaRaj Yata  
MayRjYt
```

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. 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

```
file1 = open("C:/Users/ansan/.ipynb_checkpoints/input.txt","r")  
string=file1.read()  
file2 = open("C:/Users/ansan/.ipynb_checkpoints/output.txt","w+")  
words= dict()  
sep_words= string.split()
```

```

for i in sep_words:
    if i in words:
        words[i] += 1
    else:
        words[i] = 1
file2.write("Word_Count: "+str(words))

file2.close()

```

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]

1) Nested Interactive loop.

```

L=[]
O=[]
x=int(input("No of heights"))
for i in range(x):
    L.append(float(input()))
for j in range(x):
    O.append(L[j]/2.205)
print(O)

```

No of heights 4

150
155
145
148

```
[68.02721088435374, 70.29478458049887, 65.75963718820861,
67.12018140589569]
```

2) List comprehensions

```

x=int(input())
L=list(map(float, input().strip().split()))[:x]
O = [i /2.205 for i in L]
print(O)

```

4
150 155 145 148

```
[68.02721088435374, 70.29478458049887, 65.75963718820861,
67.12018140589569]
```