

LAB3 – Distirbuted Systems Lab

Name: Sagnik Chatterjee

Reg:- 180905478

Roll No : 61

Batch : B2

SEC : B

Q1

'''

Author: Sagnik Chatterjee

Program to find the factos of a given number (input from user)
using for loop.

'''

```
def factors(n:int):
```

```
    list1=[]
```

```
    for i in range(2,n):
```

```
        if n%i==0:
```

```
            list1.append(i)
```

```
    return list1
```

```
def main():
```

```
    n=int(input("Enter the number: "))
```

```
    if n >=1:
```

```
        print("Factors of the number are: ")
```

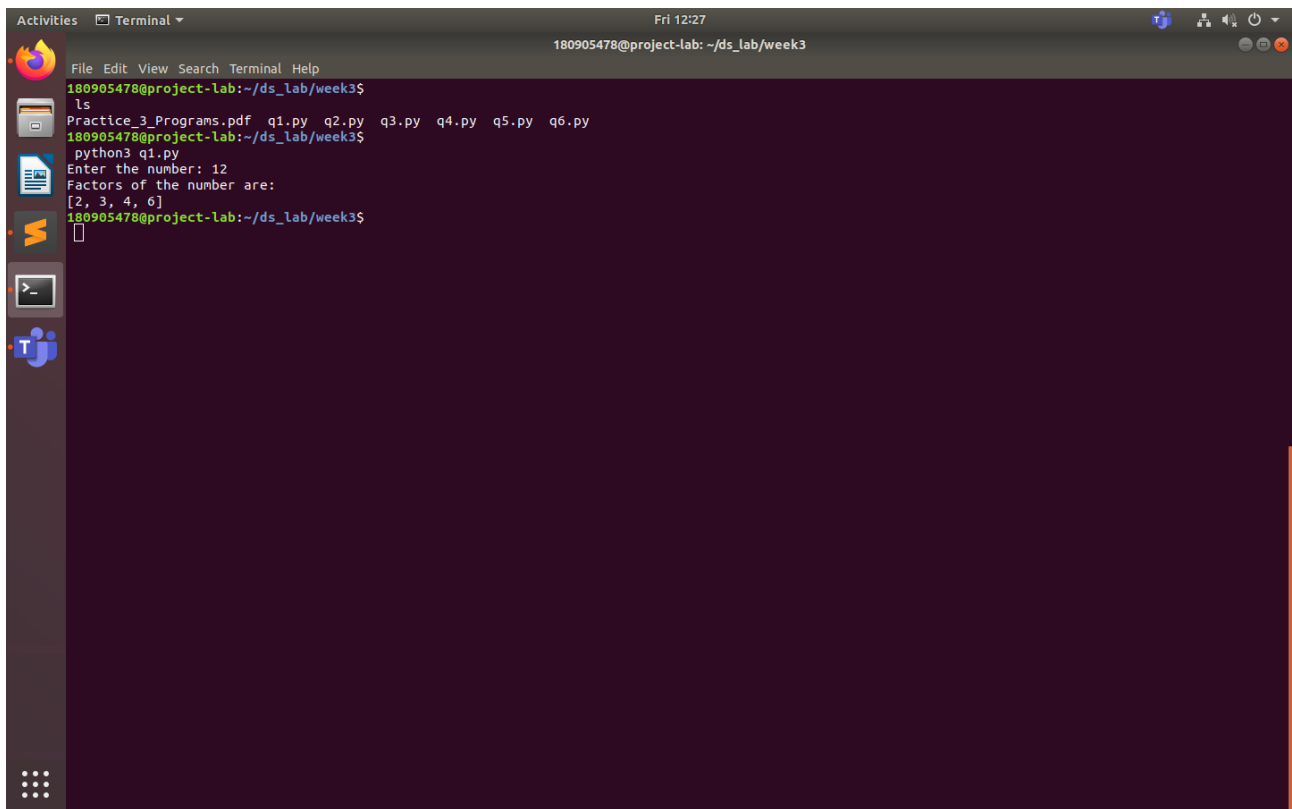
```
        print(factors(n))
```

```
    else:
```

```
        print("Please enter a positive natural number for checking tis factors.")
```

```
if __name__=='__main__':
```

```
    main()
```



The screenshot shows a terminal window titled 'Terminal' with a menu bar (File, Edit, View, Search, Terminal, Help). The user is at the prompt '180905478@project-lab: ~/ds_lab/week3'. They run 'ls' showing files: 'Practice_3_Programs.pdf', 'q1.py', 'q2.py', 'q3.py', 'q4.py', 'q5.py', and 'q6.py'. Then they run 'python3 q1.py', which prompts 'Enter the number: 12' and outputs 'Factors of the number are: [2, 3, 4, 6]'.

```
180905478@project-lab:~/ds_lab/week3$ ls
Practice_3_Programs.pdf  q1.py  q2.py  q3.py  q4.py  q5.py  q6.py
180905478@project-lab:~/ds_lab/week3$ python3 q1.py
Enter the number: 12
Factors of the number are:
[2, 3, 4, 6]
180905478@project-lab:~/ds_lab/week3$
```

Q2

'''

Author : Sagnik Chatterjee

Program to find the sum of columns and rows using axis.

'''

```
import numpy as np
```

```
def main():
```

```
    print("Assuming a 2d matrix")
```

```
    print('Enter the dimension of the column and row,seperated using a single space on  
the same line')
```

```
    n,m=map(int,input().split())
```

```
    arr=[[int(input()) for x in range(n)] for y in range(m)]
```

```
    numpy_arr=np.array(arr)
```

```
    #original matrix
```

```
    print("Original matrix is :- ")
```

```
    print(numpy_arr)
```

```
    ##summing along all the rows
```

```
    print("The sum of all the rows:")
```

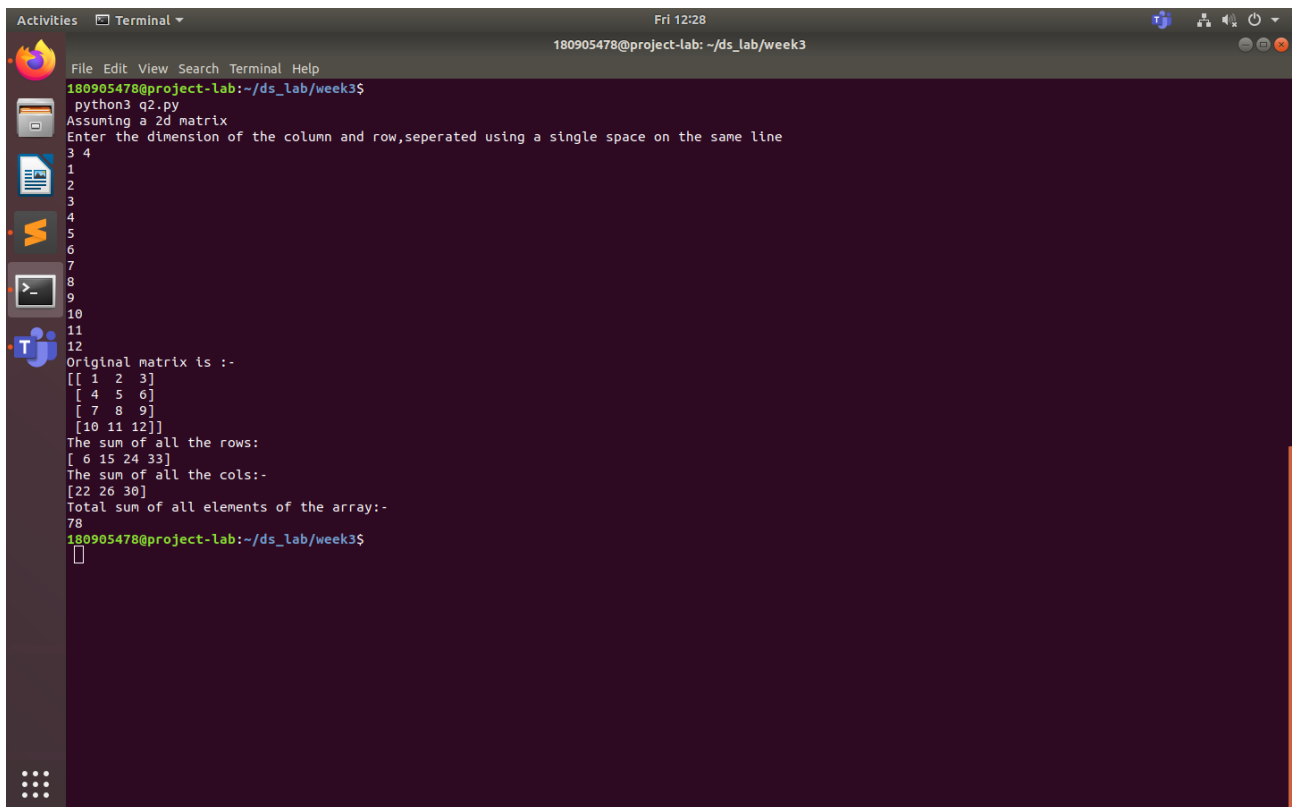
```
    print(np.sum(numpy_arr,axis=1))
```

```
    #summing along all the cols
```

```
print("The sum of all the cols:-")
print(np.sum(numpy_arr,axis=0))
```

```
#sum of matrix
print("Total sum of all elements of the array:- ")
print(np.sum(numpy_arr))
```

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



```
Activities Terminal Fri 12:28
180905478@project-lab: ~/ds_lab/week3
File Edit View Search Terminal Help
180905478@project-lab:~/ds_lab/week3$ python3 q2.py
Assuming a 2d matrix
Enter the dimension of the column and row,seperated using a single space on the same line
3 4
1
2
3
4
5
6
7
8
9
10
11
12
Original matrix is :-
[[ 1  2  3]
 [ 4  5  6]
 [ 7  8  9]
 [10 11 12]]
The sum of all the rows:
[ 6 15 24 33]
The sum of all the cols:-
[22 26 30]
Total sum of all elements of the array:-
78
180905478@project-lab:~/ds_lab/week3$
```

Q3

'''

Author: Sagnik Chatterjee
Operations on arrays

'''

```
import numpy as np
```

```
#part a : Create array from list with type float
```

```
print("\nPart a')
print("\nEnter the size of the list ")
p=int(input())
arr1=[]
print('Enter the float values for the list')
while p>0:
    q=float(input())
```

```

        arr1.append(q)
        p=p-1
arr =np.array(arr1)
print("\nArray created from list :- \n")
print(arr1)

```

```

#part b: Create array from tuple
print('\nPart b')
tup1=(11,22,43,23523,1231213)
arr2 =np.array(tup1)
print("\nArray created from tuple:- ")
print(arr2)

```

```

#part c:- Creating a 3*4 array with all zeroes
print('\nPart c ')
a=(3,4)
print("\nThe 3*4 array created with all zeroes: - \n")
print(np.zeros(a))

```

```

#part d:- Creae a sequence of integers fom 0 to 20 with steps of 5
print('\n Part d ')
temp=0
sequence=[]
while temp<=20:
    sequence.append(temp)
    temp+=5

print('The sequence is :- \n')
print(sequence)

```

```

#part e:- Reshape 3X4 array to 2X2X3 array
print('Part e\n')
arr3=np.arange(12).reshape(3,4)
print("\nThe original array :- ")
print(arr3)
print('\nThe reshaped array:- ')
print(arr3.reshape(2,2,3))

```

```

""" part f:- Find maximum and minimum element of array, Row wise max and min, column
wise max
and min and sum of elements. (Use functions max(), min(), sum())
"""

```

```

print('Part f\n')
a=np.arange(12).reshape(2,6)
print('The array is :-\n ',a)
print("Full array Max = " , a.max() , " Min = " , a.min() , " Sum = " , a.sum())
print("Rowwise array Max = " , a.max(axis = 1) , " Min = " , a.min(axis = 1) , " Sum= " ,
a.sum(axis = 1))

```

```
print("Rowwise array Max = " , a.max(axis = 0) , " Min = " , a.min(axis = 0) , " Sum= " ,
a.sum(axis = 0))
```

```

180905478@project-lab:~/ds_lab/week3$ ls
Practice_3_Programs.pdf  q1.py  q2.py  q3.py  q4.py  q5.py  q6.py
180905478@project-lab:~/ds_lab/week3$ python3 q3.py
Part a
Enter the size of the list
3
Enter the float values for the list
1.56
7.89
3.678
Array created from list :-
[1.56, 7.89, 3.678]
Part b
Array created from tuple:-
[ 11 22 43 23523 1231213]
Part c
The 3*4 array created with all zeroes: -
[[0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]]
Part d
The sequence is :-
[0, 5, 10, 15, 20]
Part e
The original array :-
[[ 0 1 2 3]
 [ 4 5 6 7]
 [ 8 9 10 11]]
The reshaped array:-
[[ 0 1 2]
 [ 3 4 5]]

```

```

180905478@project-lab:~/ds_lab/week3$ python3 q3.py
Array created from list :-
[1.56, 7.89, 3.678]
Part b
Array created from tuple:-
[ 11 22 43 23523 1231213]
Part c
The 3*4 array created with all zeroes: -
[[0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]]
Part d
The sequence is :-
[0, 5, 10, 15, 20]
Part e
The original array :-
[[ 0 1 2 3]
 [ 4 5 6 7]
 [ 8 9 10 11]]
The reshaped array:-
[[[ 0 1 2]
 [ 3 4 5]]
 [[ 6 7 8]
 [ 9 10 11]]]
Part f
The array is :-
[[ 0 1 2 3 4 5]
 [ 6 7 8 9 10 11]]
Full array Max = 11 Min = 0 Sum = 66
Rowwise array Max = [ 5 11] Min = [0 6] Sum= [15 51]
Rowwise array Max = [ 6 7 8 9 10 11] Min = [0 1 2 3 4 5] Sum= [ 6 8 10 12 14 16]
180905478@project-lab:~/ds_lab/week3$

```

Q4

'''

Author : Sagnik Chatterjee

Program to transpose a given matrix.

'''

```
import numpy as np
```

```
def main():
```

```
    print("Assuming a 2d matrix")
```

```
    print('Enter the dimension of the column and row,seperated using a single space on  
the same line')
```

```
    n,m=map(int,input().split())
```

```
    arr=[[int(input()) for x in range(n)] for y in range(m)]
```

```
    numpy_arr=np.array(arr)
```

```
    #original matrix
```

```
    print("Original matrix is :- ")
```

```
    print(numpy_arr)
```

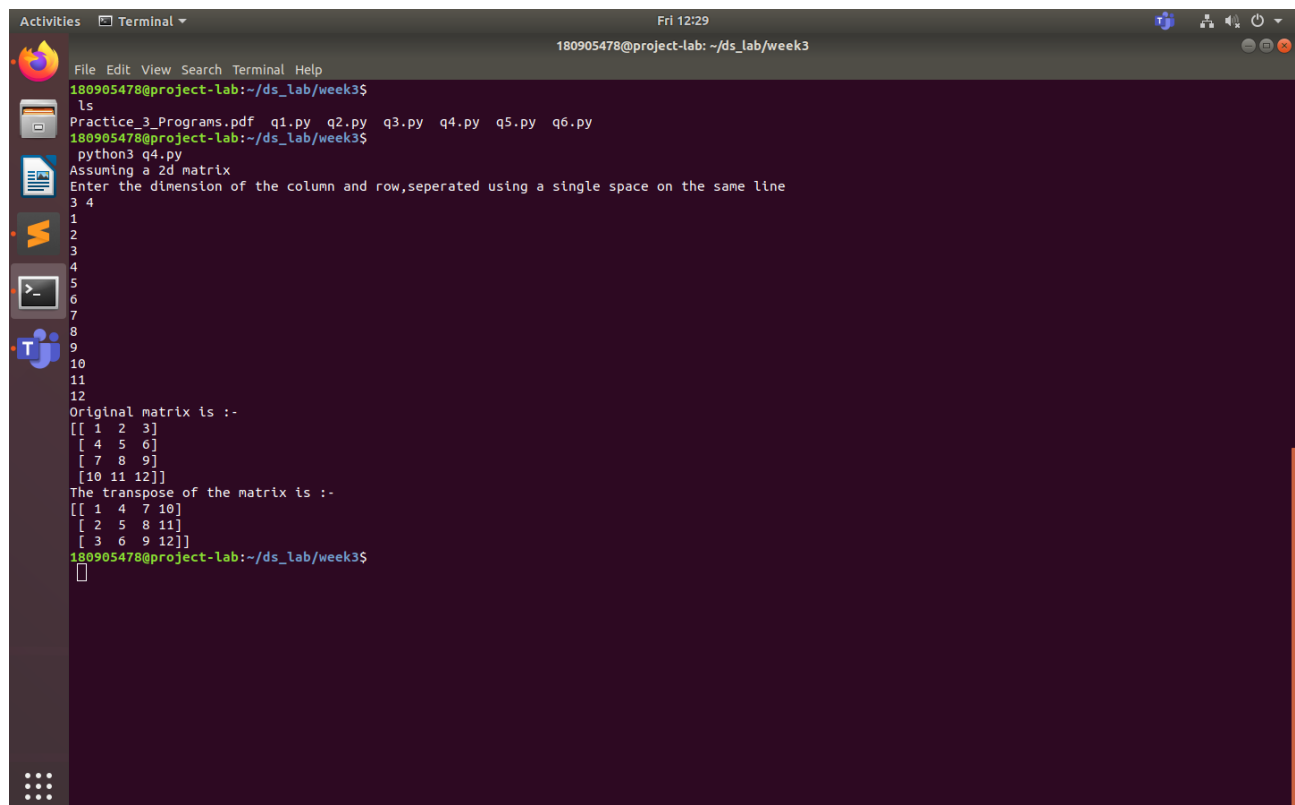
```
    # the transpose of the matrix is
```

```
    print("The transpose of the matrix is :-")
```

```
    print(numpy_arr.transpose())
```

```
if __name__=='__main__':
```

```
    main()
```



The screenshot shows a terminal window with the following content:

```
180905478@project-lab: ~/ds_lab/week3
ls
Practice_3_Programs.pdf q1.py q2.py q3.py q4.py q5.py q6.py
180905478@project-lab:~/ds_lab/week3$ python3 q4.py
Assuming a 2d matrix
Enter the dimension of the column and row,seperated using a single space on the same line
3 4
1
2
3
4
5
6
7
8
9
10
11
12
Original matrix is :-
[[ 1  2  3]
 [ 4  5  6]
 [ 7  8  9]
 [10 11 12]]
The transpose of the matrix is :-
[[ 1  4  7 10]
 [ 2  5  8 11]
 [ 3  6  9 12]]
180905478@project-lab:~/ds_lab/week3$
```

Q5

'''

Author : Sagnik Chatterjee

Program to add two matrices.

'''

```
import numpy as np
```

```
def main():
```

```
    try:
```

```
        print("Assuming a 2d matrix")
```

```
        print('Enter the dimension of the 1st matrix(column and row),seperated using  
a single space on the same line')
```

```
        n,m=map(int,input().split())
```

```
        arr1=[[int(input()) for x in range(n)] for y in range(m)]
```

```
        print('Enter the dimension of the 2nd matrix(column and row),seperated  
using a single space on the same line')
```

```
        a,b=map(int,input().split())
```

```
        arr2=[[int(input()) for x in range(a)] for y in range(b)]
```

```
        if(n!=a or m!=b ):
```

```
            raise Exception("Dimension do not match for the two matrices;please  
check your values.")
```

```
        numpy_arr1=np.array(arr1)
```

```
        numpy_arr2=np.array(arr2)
```

```
        #original matrix
```

```
        print("Original matrices are :- ")
```

```
        print(numpy_arr1)
```

```
        print(numpy_arr2)
```

```
        ## the matrix after summing them up
```

```
        print("The matrix obtained after summing them up.")
```

```
        result_matrix= np.add(numpy_arr1,numpy_arr2)
```

```
        print(result_matrix)
```

```
    except Exception as e:
```

```
        print(e)
```

```
if __name__=='__main__':
```

```
    main()
```

```
Activities Terminal FRI 12:29 180905478@project-lab: ~/ds_lab/week3
File Edit View Search Terminal Help
180905478@project-lab:~/ds_lab/week3$ python3 q5.py
Assuming a 2d matrix
Enter the dimension of the 1st matrix(column and row),seperated using a single space on the same line
2 3
1
2
3
4
5
6
Enter the dimension of the 2nd matrix(column and row),seperated using a single space on the same line
2 3
7
8
9
10
11
12
Original matrices are :-
[[1 2]
 [3 4]
 [5 6]]
[[ 7 8]
 [ 9 10]
 [11 12]]
The matrix obtained after sunning them up.
[[ 8 10]
 [12 14]
 [16 18]]
180905478@project-lab:~/ds_lab/week3$
```

Q6

'''

Author : Sagnik Chatterjee

Program to find element wise product between 2 matrices

'''

```
import numpy as np
```

```
def main():
```

```
    try:
```

```
        print("Assuming a 2d matrix")
```

```
        print('Enter the dimension of the 1st matrix(column and row),seperated using  
a single space on the same line')
```

```
        n,m=map(int,input().split())
```

```
        arr1=[[int(input()) for x in range(n)] for y in range(m)]
```

```
        print('Enter the dimension of the 2nd matrix(column and row),seperated  
using a single space on the same line')
```

```
        a,b=map(int,input().split())
```

```
        arr2=[[int(input()) for x in range(a)] for y in range(b)]
```

```
        if(n!=a or m!=b ):
```

```
            raise Exception("Dimension do not match for the two matrices;please  
check your values.")
```

```
        numpy_arr1=np.array(arr1)
```



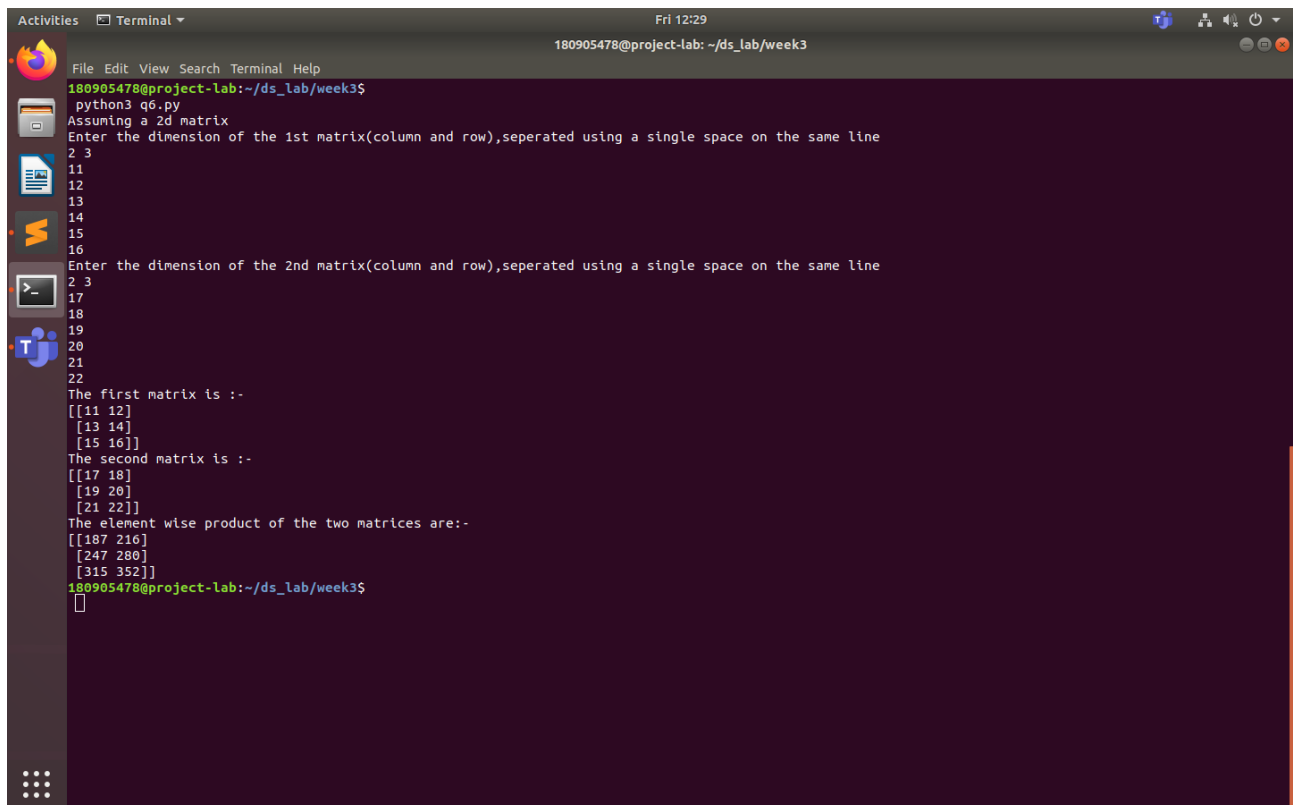
```
numpy_arr2=np.array(arr2)
```

```
##the original matrices  
print("The first matrix is :- ")  
print(numpy_arr1)  
print("The second matrix is :- ")  
print(numpy_arr2)
```

```
## the element wise product of the 2 matrices  
print("The element wise product of the two matrices are:- ")  
print(np.multiply(numpy_arr1,numpy_arr2))
```

```
except Exception as e:  
    print(e)
```

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



```
Activities Terminal Fri 12:29  
180905478@project-lab: ~/ds_lab/week3  
File Edit View Search Terminal Help  
180905478@project-lab:~/ds_lab/week3$ python3 q6.py  
Assuming a 2d matrix  
Enter the dimension of the 1st matrix(column and row),seperated using a single space on the same line  
2 3  
11  
12  
13  
14  
15  
16  
Enter the dimension of the 2nd matrix(column and row),seperated using a single space on the same line  
2 3  
17  
18  
19  
20  
21  
22  
The first matrix is :-  
[[11 12]  
 [13 14]  
 [15 16]]  
The second matrix is :-  
[[17 18]  
 [19 20]  
 [21 22]]  
The element wise product of the two matrices are:-  
[[187 216]  
 [247 280]  
 [315 352]]  
180905478@project-lab:~/ds_lab/week3$
```