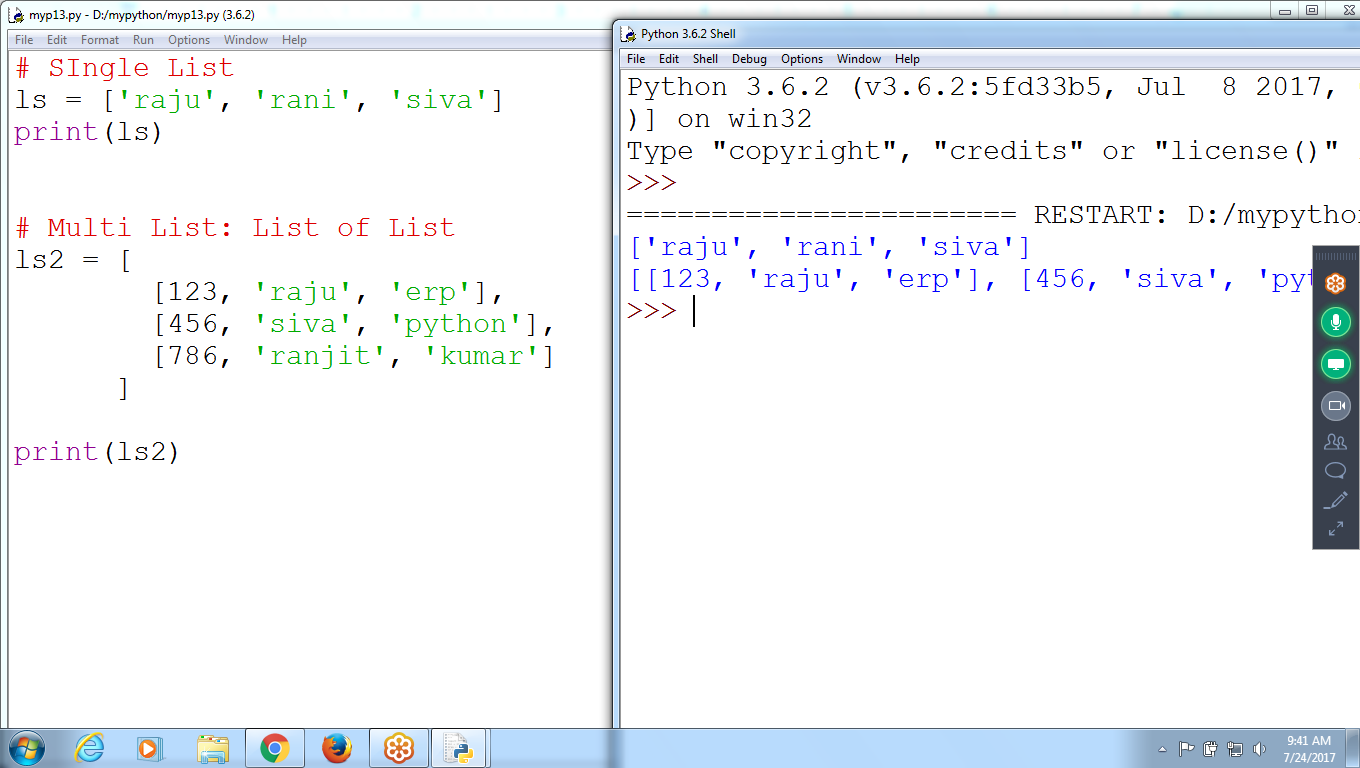
**Two Dimensional List :: in List**

****

**# SIngle List**

**ls = ['raju', 'rani', 'siva']**

**print(ls)**

**# Multi List: List of List**

**ls2 = [**

**[123, 'raju', 'erp'],**

**[456, 'siva', 'python'],**

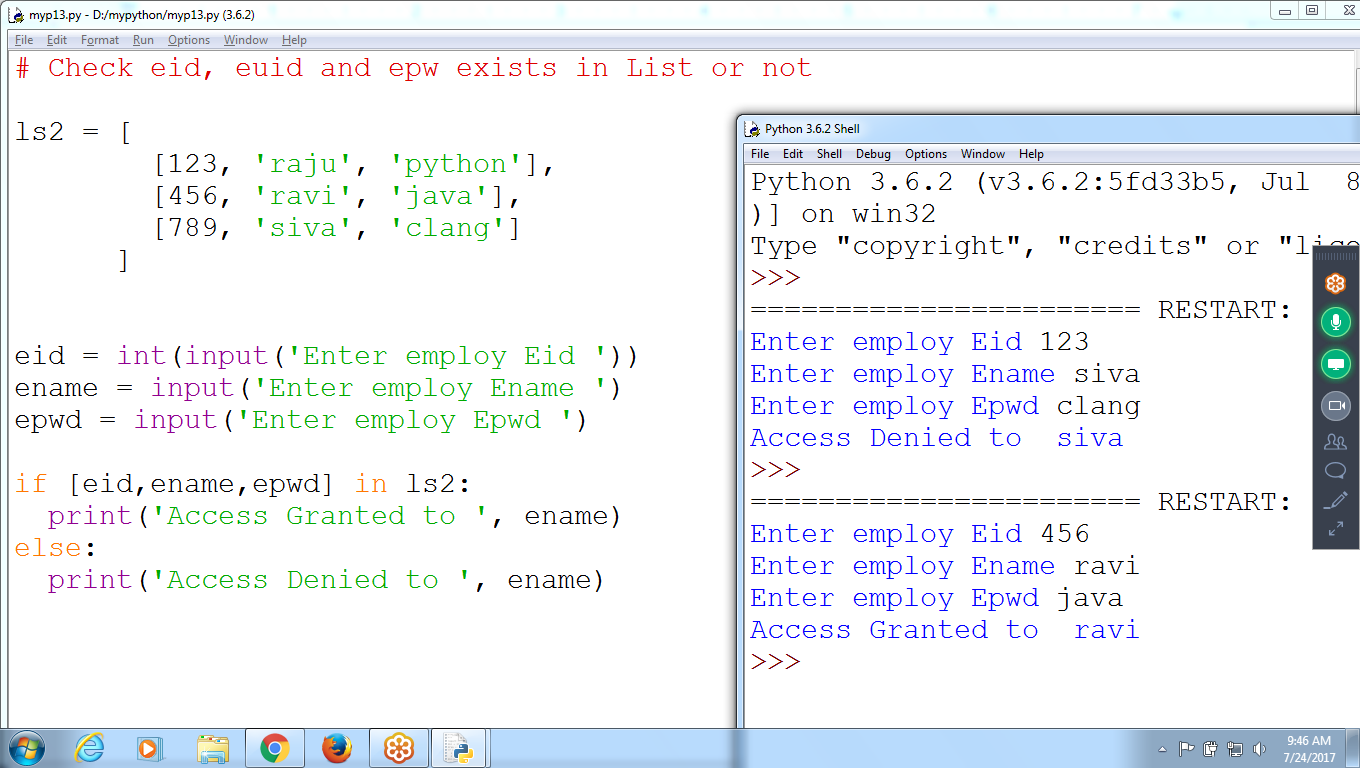
**[786, 'ranjit', 'kumar']**

**]**

**print(ls2)**

**Accept Eid, Uid,pwd**

**And check existing in list or not**

****

**# Check eid, euid and epwd exists in List or not**

**ls2 = [**

**[123, 'raju', 'python'],**

**[456, 'ravi', 'java'],**

**[789, 'siva', 'clang']**

**]**

**eid = int(input('Enter employ Eid '))**

**ename = input('Enter employ Ename ')**

**epwd = input('Enter employ Epwd ')**

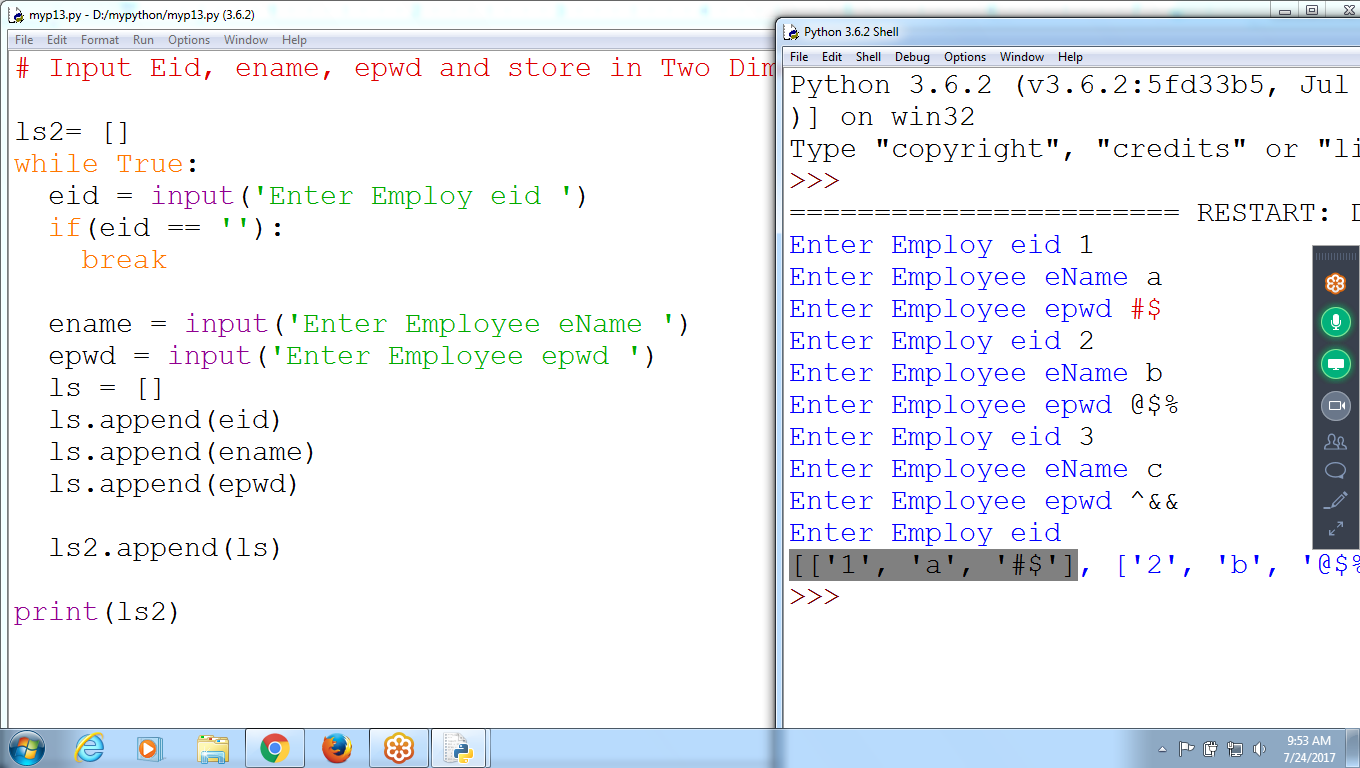
**if [eid,ename,epwd] in ls2:**

**print('Access Granted to ', ename)**

**else:**

**print('Access Denied to ', ename)**

**Accept 3 values and store onto LIST in TWO-dimensional list**



# Input Eid, ename, epwd and store in Two Dimensional List

ls2= []

while True:

eid = input('Enter Employ eid ')

if(eid == ''):

break

ename = input('Enter Employee eName ')

epwd = input('Enter Employee epwd ')

ls = []

ls.append(eid)

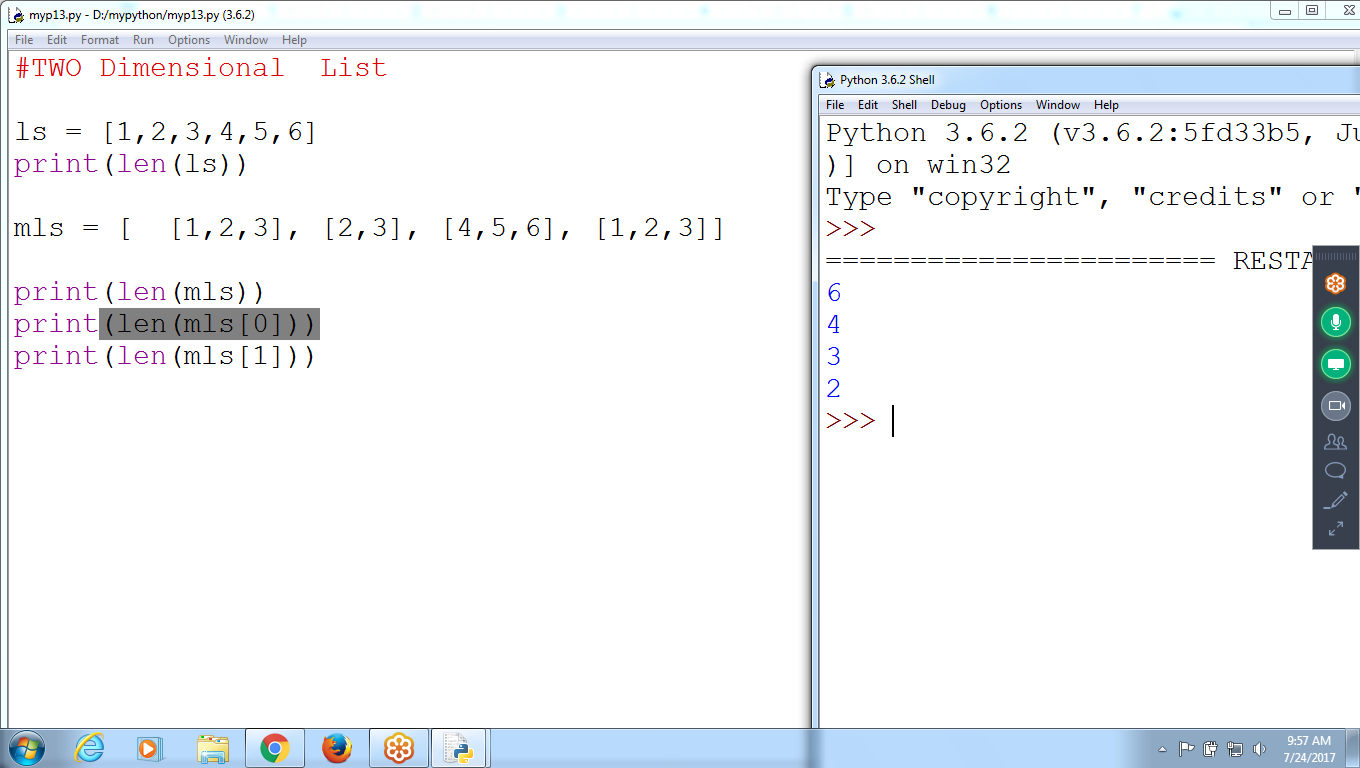
ls.append(ename)

ls.append(epwd)

ls2.append(ls)

print(ls2)

TWO Dimensional Matrix

****

**#TWO Dimensional List**

**ls = [1,2,3,4,5,6]**

**print(len(ls))**

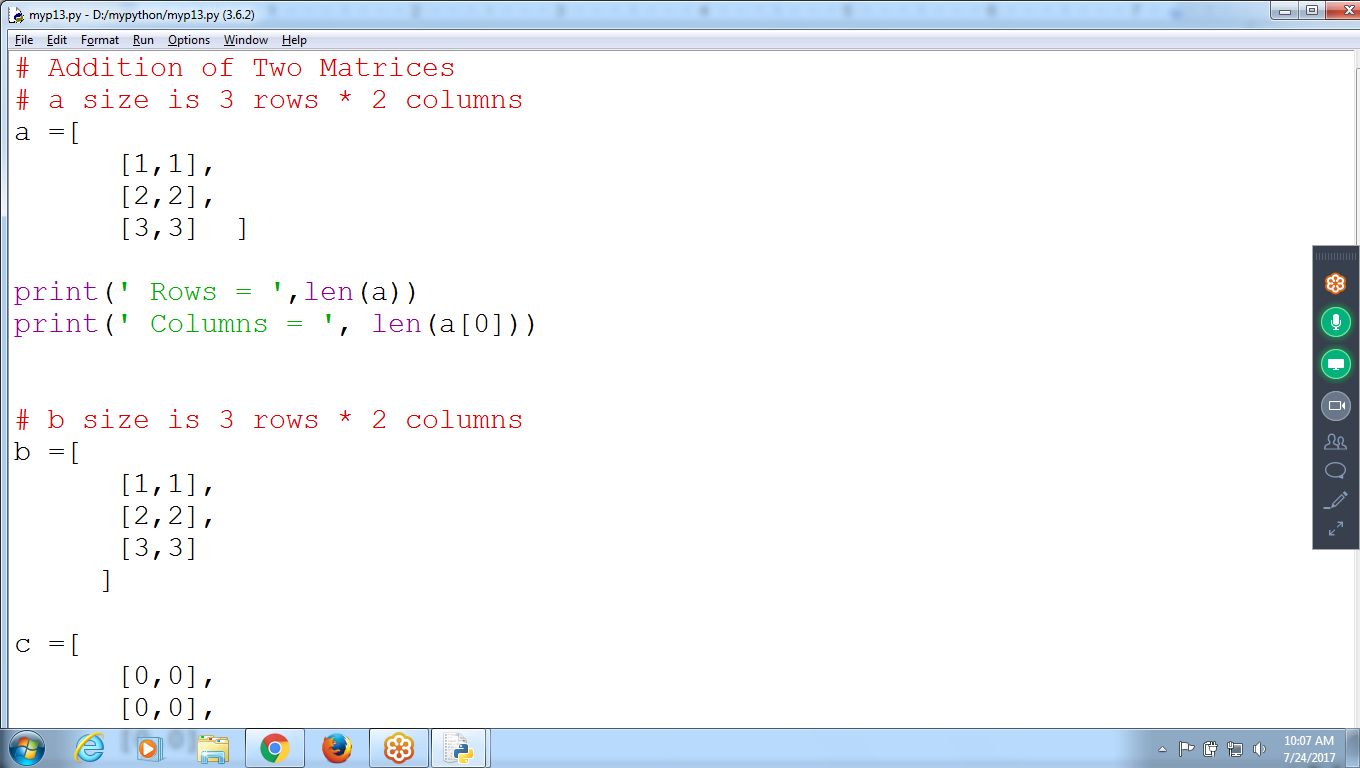
**mls = [ [1,2,3], [2,3], [4,5,6], [1,2,3]]**

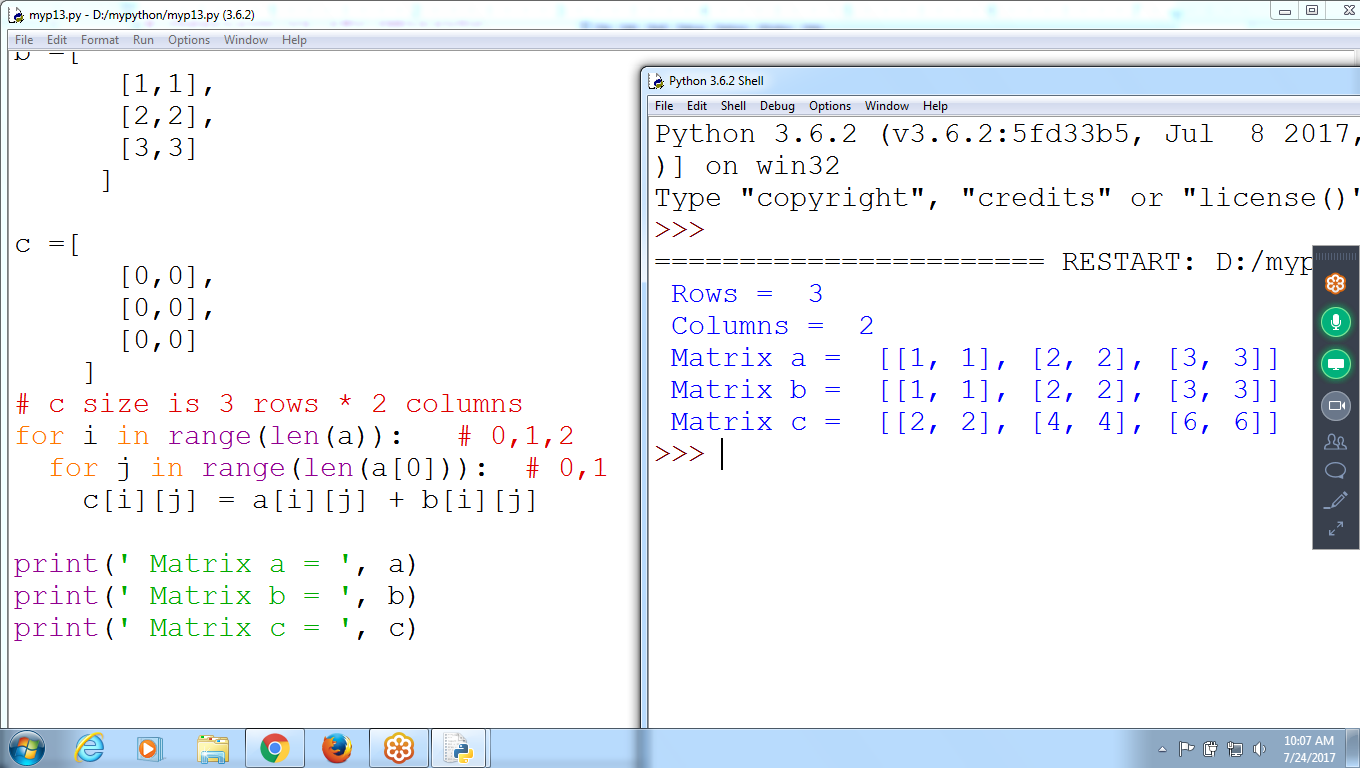
**print(len(mls))**

**print(len(mls[0]))**

**print(len(mls[1]))**

**Addition of TWO Matrices**

****



**# Addition of Two Matrices**

**a = [**

**[ 1, 1],**

**[ 2, 2],**

**[ 3, 3]**

**]**

**b = [**

**[ 1, 1],**

**[ 2, 2],**

**[ 3, 3]**

**]**

**c = [**

**[0,0],**

**[0,0],**

**[0,0]**

**]**

**for i in range(len(a)): # len(a) :: 3 :: i represents for rows**

**for j in range(len(a[0])): # len(a[0] ) :: 2 :: jis for columns**

**c[i][j] = a[i][j] + b[i][j]**

**print(a)**

**print(b)**

**print(c)**

**#c= a + b**

**# STring Concatenation**

**Matrix A Columns must equals to Matrix B Rows**

**2 \* 3 / 3 \* 2**

**A (2\*3) = [**

**1 2 1**

**1 2 1**

**B (3\*2) = [**

**1 1**

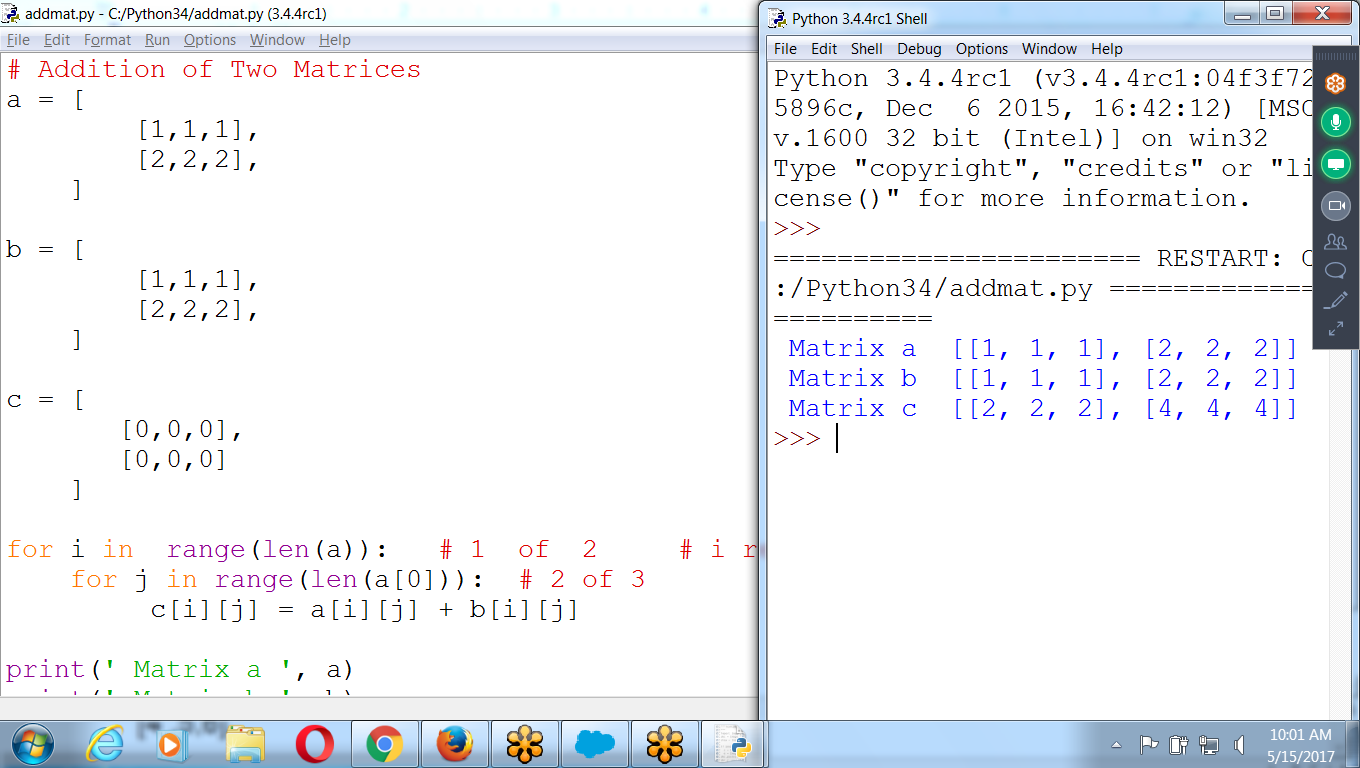
**2 1**

**1 2**

**C (2\*2) = 1+4+1 1+2+2**

**1+4+1 1+2+2**

**C = [ [ 6, 5] [6,5]]**



# Addition of Two Matrices

a = [

[1,1,1],

[2,2,2],

]

b = [

[1,1,1],

[2,2,2],

]

c = [

[0,0,0],

[0,0,0]

]

for i in range(len(a)): # 1 of 2 # i represents rows i.e 2

for j in range(len(a[0])): # 2 of 3

c[i][j] = a[i][j] + b[i][j]

print(' Matrix a ', a)

print(' Matrix b ', b)

print(' Matrix c ', c)

**# Multiplication of Two Matrices**

a = [

[1,1,1],

[2,2,2],

]

**# a (2\*3) b(3\*3)**

**# Result (2\*3)**

**For i in range(len(a)): #2 Rows**

**For j in range(len(b[0])) # 3 Columns**

**For k in range(len(a[0]) # 3 Multiplications**

**C[i][j] = c[i][j] + a[i] [k] \* b[k][j]**

b = [

[1,1,1],

[2,3,4],

[1,1,1]

]

**#b(3\*3)**

c = [

[1\*1+ 1\*2 +1\*1, 1\*1 +1\*3 +1\*1, 1\*1+1\*4+1\*1],

[2\*1+ 2\*2+ 2\*1, 2\*1 + 2\*3 +2\*1, 2\*1+2\*4+2\*1 ]

]

**#c(2\*3)**

c = [

[4, 5, 6],

[8, 10,12 ]

]

# a(2 \*3) = b( 3 \*3)

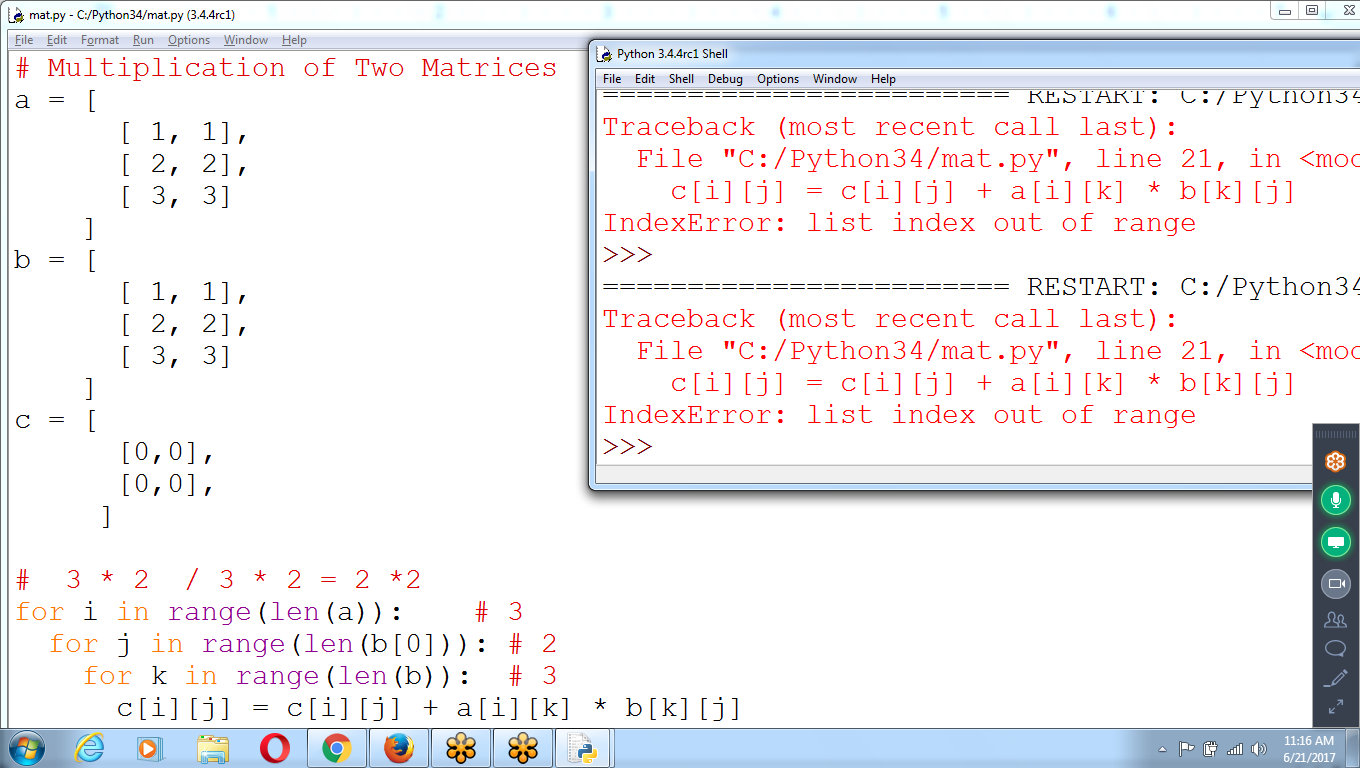
**COMMON: ‘a Matrix columns’ must be similar to ‘b matrix rows’**

# Result a rows and b colus

c = 2\*3

A 0th rows, multiply with b0th colu

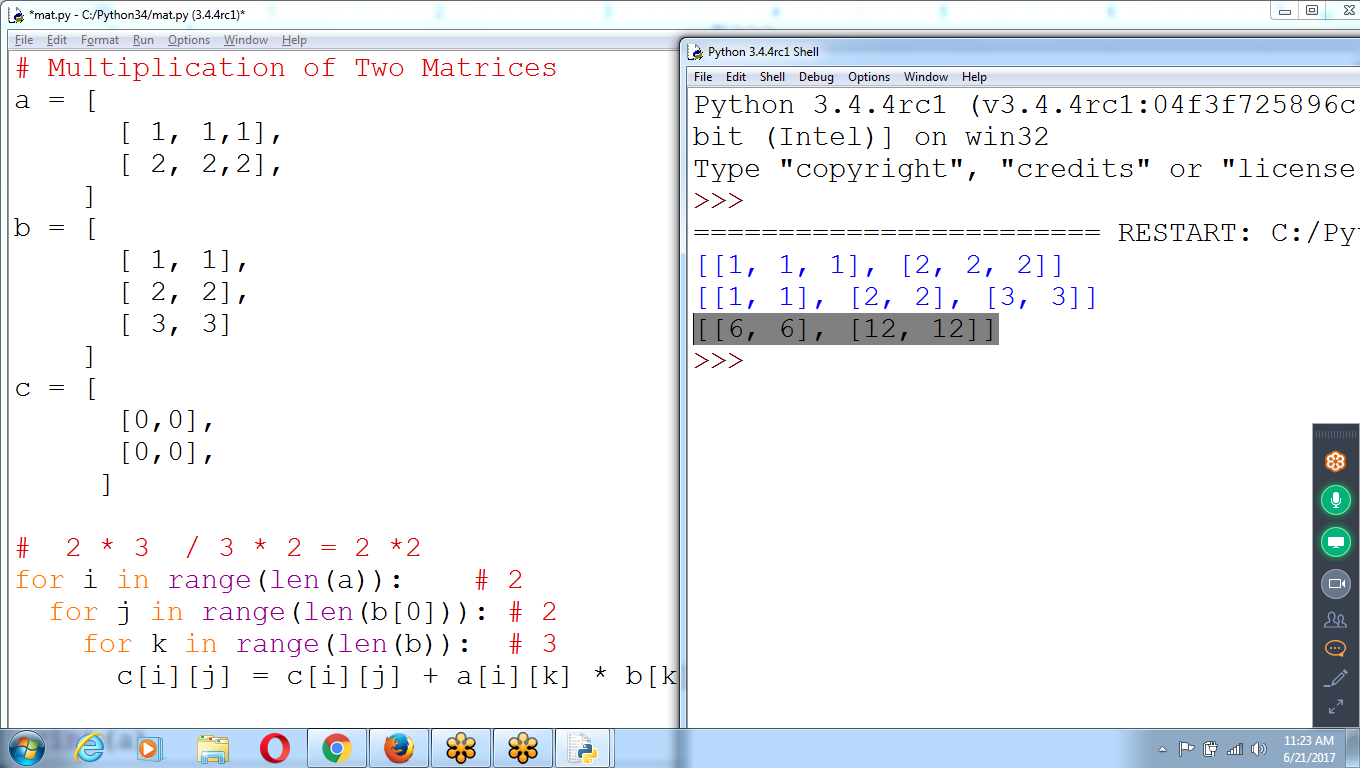
A : 3\*2 and B : 3 \*2 NOT possible



A : 2\*3

B: 3 \*2

C : 2 \*2 # Valid matrix



# Multiplication of Two Matrices

a = [

[ 1, 1,1],

[ 2, 2,2],

]

b = [

[ 1, 1],

[ 2, 2],

[ 3, 3]

]

c = [

[0,0],

[0,0],

]

# 2 \* 3 / 3 \* 2 = 2 \*2

for i in range(len(a)): # 2

for j in range(len(b[0])): # 2

for k in range(len(b)): # 3

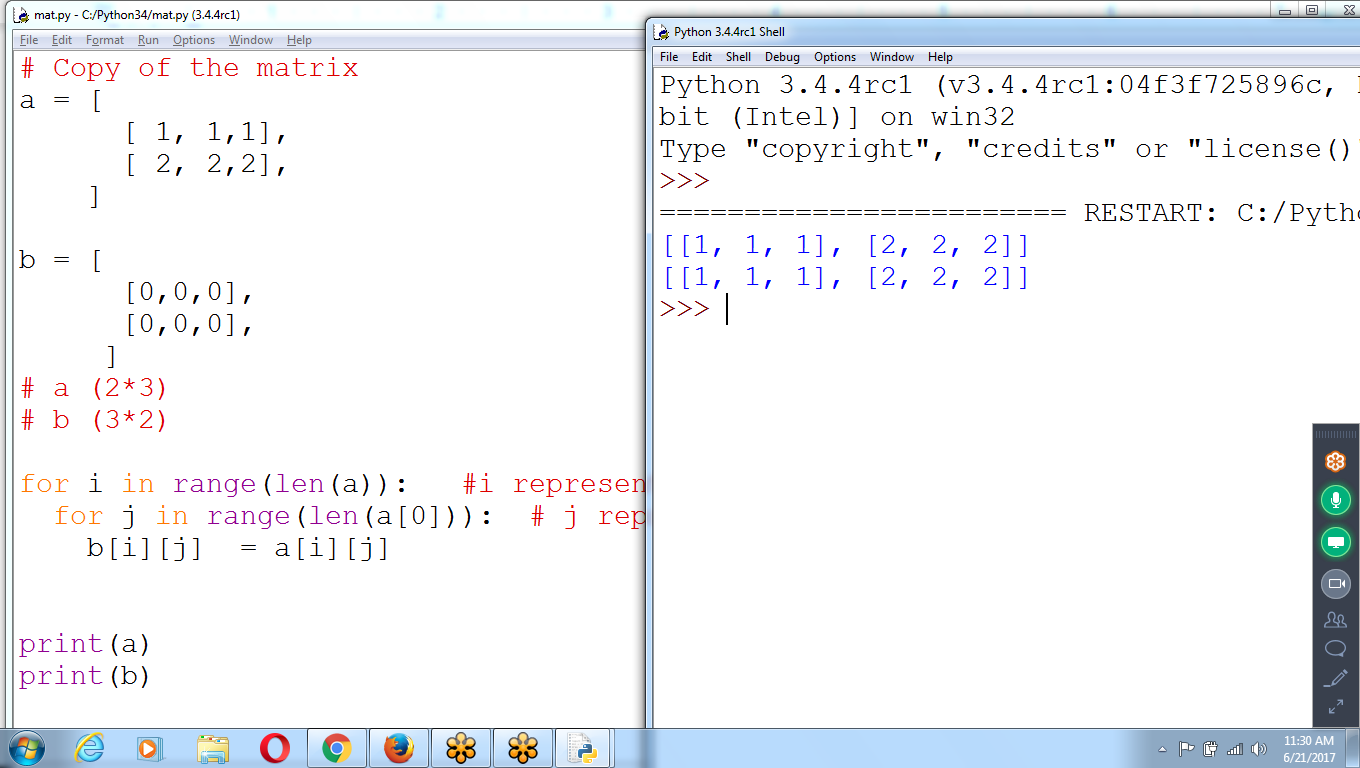
c[i][j] = c[i][j] + a[i][k] \* b[k][j]

print(a)

print(b)

print(c)

**Copy of the Matrix**



# Copy of the matrix

a = [

[ 1, 1,1],

[ 2, 2,2],

]

b = [

[0,0,0],

[0,0,0],

]

# a (2\*3)

# b (3\*2)

for i in range(len(a)): #i represents rows :: 2

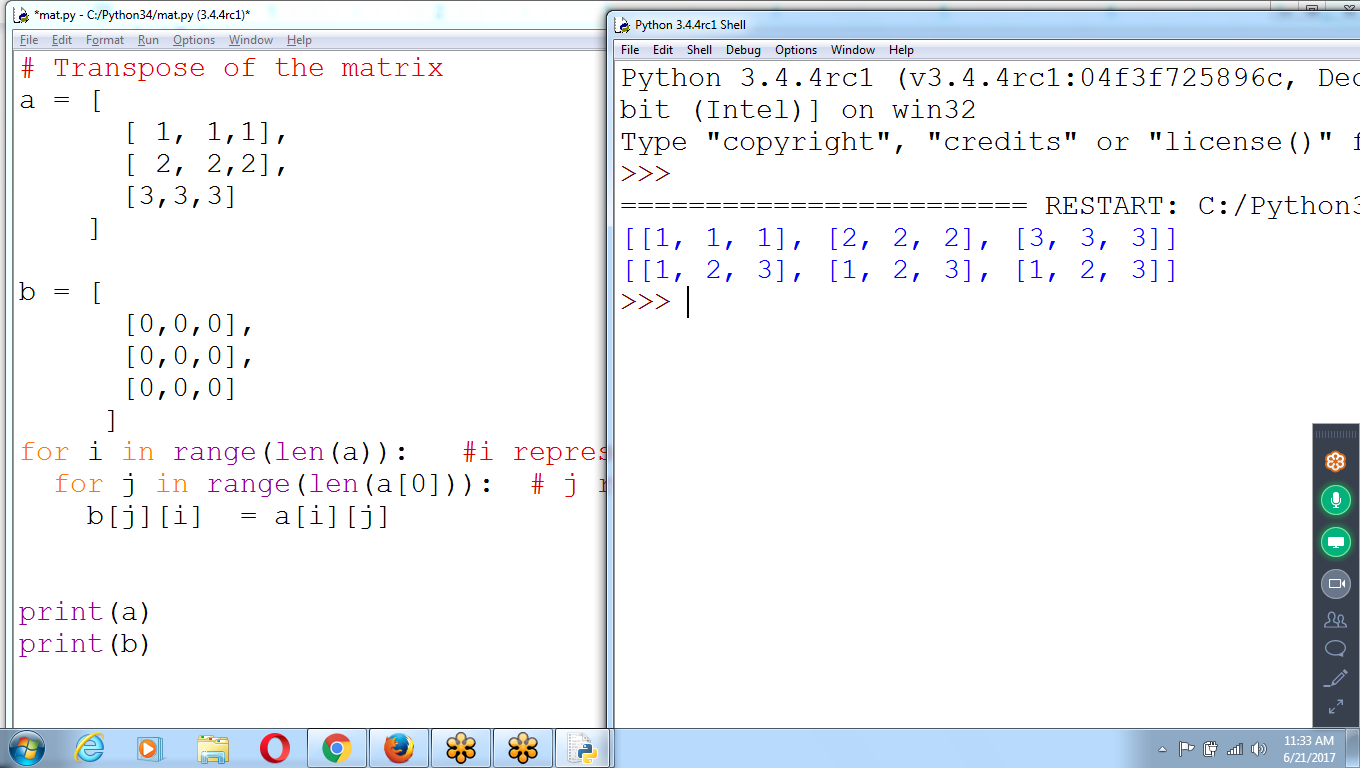
for j in range(len(a[0])): # j represents Columns :: 3

b[i][j] = a[i][j]

print(a)

print(b)

**Transpose of Given matrix**



# Transpose of the matrix

a = [

[ 1, 1,1],

[ 2, 2,2],

[3,3,3]

]

b = [

[0,0,0],

[0,0,0],

[0,0,0]

]

for i in range(len(a)): #i represents rows :: 2

for j in range(len(a[0])): # j represents Columns :: 3

b[j][i] = a[i][j]

print(a)

print(b)