**Assignment 1**

Q1: Given two matrices please print the product of those two matrices

<https://snakify.org/en/lessons/two_dimensional_lists_arrays/>

for finding a length of list we use

len()

len(a)

a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]

s = 0

for i in range(len(a)):

    for j in range(len(a[i])):

        s += a[i][j]

print(s)

**using this I create this**

A   = [[1,3,4],[2,5],[5,9,6]]

for i in range(len(A)):

  for j in range(len(A[i])):

    print(i,j)

    print(A[i][j])

**counting R\*C**

A   = [[1,3,4,5],[2,5,6],[2,5,6],[2,5,6,5]]

countr = 0

countc = 0

for i in range(len(A)):

  for j in range(len(A[i])):

    countc = len(A[i])

  countr = len(A)

print(countr, countc)

**counting  and checking mul or not**

A   = [[1,2],[3,4]]

countr1 = 0

countc1 = 0

for i in range(len(A)):

  for j in range(len(A[i])):

    countc1 = len(A[i])

  countr1 = len(A)

print(countr1, countc1)

# counting for another R\*C

B   = [[1,2,3,4,5],[5,6,7,8,9]]

countr2 = 0

countc2 = 0

for i in range(len(B)):

  for j in range(len(B[i])):

    countc2 = len(B[i])

  countr2 = len(B)

print(countr2, countc2)

if countc1 == countr2:

  print("we can multiply this")

else:

    print("not multiplcable")

**use this method**

a\_dictionary = {}

for number in range(1,4):

    a\_dictionary["key%s" %number] = "abc"

print(a\_dictionary)

**create a seperate num from list and store it into dictionary**

globals() ['A'] = [[1,2,3,4,5],[5,6,7,8,9]]

globals() ['a']

def var(i,j):

  a = {}

  a["key{0}{1}".format(i,j)] = A[i][j]

  print(a)

for i in range(len(A)):

  for j in range(len(A[i])):

    var(i,j)

**seperated by rowise**

globals() ['A'] = [[1,2],[5,6]]

globals() ['a']

def var(i,j):

  a = {}

  if j == 0:

    a["r{0}{1}".format(i,j)] = A[i][j]

    print(a)

for i in range(len(A)):

  for j in range(len(A[i])):

    var(i,j)

**numbers by ROWS**

globals() ['A'] = [[1,2],

                   [5,6]]

def row(i,j):

  a = {}

  for z in range(len(A)):

    if i == z:

      a["{2}r{0}{1}".format(i,j,z)] = A[i][j]

  print(a)

def column(i,j):

  b = {}

for i in range(len(A)):

  for j in range(len(A[i])):

    print(i)

    for z in range(len(A)):

      if i == z:

        row(i,j)

# **Adding a new key value pair**

wordFreqDic.update( {'before' : 23} )

# **numbers by COLUMNS and rows**

globals() ['A'] = [[1,2],

                   [5,6]]

ar0 = {}

ar1 = {}

ac0 = {}

ac1 = {}

def row(i,j):

    if i == 0:

      ar0.update({"0r{0}{1}".format(i,j) : A[i][j]})

    elif i == 1:

      ar1.update({"1r{0}{1}".format(i,j) : A[i][j]})

def col(i,j):

  if j == 0:

    ac0.update({"0c{0}{1}".format(i,j) : A[i][j]})

  elif j == 1:

    ac1.update({"1c{0}{1}".format(i,j) : A[i][j]})

def column(i,j):

  b = {}

for i in range(len(A)):

  for j in range(len(A[i])):

    row(i,j)

    col(i,j)

print("columns0 = {0} columns1 = {1}".format(ac0, ac1))

print("row0 = {0} row1 = {1}".format(ar0, ar1))

output

columns0 = {'0c00': 1, '0c10': 5} columns1 = {'1c01': 2, '1c11': 6}

row0 = {'0r00': 1, '0r01': 2} row1 = {'1r10': 5, '1r11': 6}