

# 210411100085\_Modul7\_List2D

December 1, 2021

## 1 List 2 Dimensi

Tulis ringkasan atau penjelasan hal-hal berikut, dengan kata-kata kalian sendiri :

1. Jelaskan mengenai List 2D, Perbedaannya dengan List 1D
2. Berikan contoh data yang menggunakan representasi List2D

### 1. Jelaskan mengenai List 2D, Perbedaannya dengan List 1D

List 2D merupakan list yang mempunyai elemen list, ilustrasinya seperti ini :

```
# | 2 3 4 1 8 | \
# | 3 4 6 1 4 | ---- Ini merupakan list 2D karena mempunyai baris dan kolom
# | 3 4 5 8 3 | /
```

Dalam implementasinya di bahasa pemrograman python list 2D direpresentasikan list yang mempunyai elemen list dan untuk mengaksesnya bisa menggunakan 2 no index. Contohnya seperti ini :

```
listMahasiswa = [["Alexandra", "Joshua", "Steve"],[86,96,75]]

print("""
Nama Mahasiswa : %s
Nilai Mahasiswa : %s

Akses Index : %s,%s,%s
""") % (
    listMahasiswa[0],
    listMahasiswa[1],
    listMahasiswa[0][0],listMahasiswa[1][0],listMahasiswa[0][1:] #Mengakses list 2D
)
```

Perbedaan antara list 2D dan list 1D dapat terlihat dalam ilustrasi

```
# | 1 2 3 4 5 | ini adalah list 1 dimensi

# | 2 3 4 1 8 | \
# | 3 4 6 1 4 | ---- Ini merupakan list 2D karena mempunyai baris dan kolom
# | 3 4 5 8 3 | /
```

perbedaan dari ilustrasi tersebut adalah pada list 1 dimensi hanya memiliki 1 baris sedangkan pada list 2 dimensi mempunyai 3 baris dan kolom. Pada bahasa python list 1 dimensi adalah list itu

sendiri, sedangkan list 2 dimensi anggota pada element list adalah list jadi terdapat list di dalam list pada list 2 dimensi. Implementasi dalam bahasa pemrograman python :

```
namaMahasiswa = ["Alexandra", "Joshua", "Steve"] #Ini merupakan list 1D
nilaiMahasiswa = [86,96,75] #Ini merupakan list 1D

mahasiswa      = [ ["Alexandra", "Joshua", "Steve"],[86,96,75] ] #Ini merupakan list 2D
```

## 2. Berikan contoh data yang menggunakan representasi List2D

Untuk menyusun data agar dapat dilihat dan diproses dengan mudah kita perlu menyimpan data tersebut dalam sebuah wadah. Seperti di bahasa pemrograman python bila ingin menyimpan data lebih dari satu kita dapat menggunakan list yang dapat menampung banyak data, bagaimana bila data tersebut membutuhkan keteraturan solusinya kita bisa menggunakan tabel dan didalam tabel terdapat baris dan kolom sehingga implementasi dalam python menggunakan list 2D yang mempunyai baris dan kolom. Contoh data yang menggunakan representasi list 2D :

- Matrix

```
matrix = [[2,3,5],[2,7,8]]
for baris in matrix :
    print(baris)
```

- Tabel nama dan nilai mahasiswa

```
mahasiswa = [ ["Alexandra", "Joshua", "Steve"],[86,96,75] ]

for i in range(len(mahasiswa[0])) :
    print("%s = %d " % (mahasiswa[0][i],mahasiswa[1][i]) )
```

## 2 Implementasi-Modul Matriks

Buatlah modul matriks yang berisi proses pembuatan dan pengolahan data list 2D, sebagai berikut

### 2.1 Create Matrix 2D

Buatlah fungsi create Matrix2D untuk membuat suatu matriks 2D dengan representasi list 2D (dengan argument atau parameter adalah jumlah baris dan jumlah kolom)

```
[1]: # Gunakan Import module from URL jika anda tidak ingin mendownload module
      ↪matrix.py
# Anda bisa mendownload di url :
# --- Github ---
# - https://raw.githubusercontent.com/satriaer77/alpro_praktikum/main/
      ↪Praktikum_Baru/Module/matrixLib.py (Versi OOP)
# - https://raw.githubusercontent.com/satriaer77/alpro_praktikum/main/
      ↪Praktikum_Baru/Module/matrix.py
#
# --- Google Drive ---
# - https://drive.google.com/drive/folders/1jf2w7p7480sl1yMQw9numNxxCoeQ1kAy?
      ↪usp=sharing
```

```

#----- Import from URL -----#
# Uncomment code dibawah ini bila akan menggunakan module lewat URL
# Setelah itu beri comment pada code from 'path' import matrix

import urllib.request

urlModule = urllib.request.urlopen('https://raw.githubusercontent.com/
↳satriaer77/alpro_praktikum/main/Praktikum_Baru/Module/matrixLib.py')
exec(urlModule.read()) #Menjalankan file py dari urlModule
matrix      = Matrix() #Inisialisasi class Matrix

#----- End Import from URL -----#

#from Module import matrix #Import dari disk folder Module

"""
# Karena module menggunakan script py terpisah jadi untuk codenya dicomment
# Ini adalah function code createMatrix2D (Membuat list matrix)

#----- Fungsi Membuat Matrix (createMatrix) -----#
# argument menggunakan variable baris dan kolom tipe <int>
# return value list matrix

def createMatrix2D(baris,kolom) :
    print("\nBuat Matrix Ordo %d x %d " % (baris,kolom) )
    matrix = []
    for i in range(baris) :
        bar = []
        for j in range(kolom) :
            bar.append(int(input("==> Mat[%d,%d] : " % (i,j) ))) #Menambahkan
↳element inputan user <int> ke list bar

            matrix.append(bar) #Add element list bar ke list matrix

    return matrix

#----- End Fungsi Membuat List Matrix (createMatrix) -----#

"""

```

```
mat1 = matrix.createMatrix2D(3,3)
mat2 = matrix.createMatrix2D(3,3)
mat3 = matrix.createMatrix2D(1,3)
mat4 = matrix.createMatrix2D(3,1)
```

Buat Matrix Ordo 3 x 3

```
==> Mat[0,0] : 3
==> Mat[0,1] : 1
==> Mat[0,2] : 6
==> Mat[1,0] : 2
==> Mat[1,1] : 10
==> Mat[1,2] : 3
==> Mat[2,0] : 4
==> Mat[2,1] : 1
==> Mat[2,2] : 5
```

Buat Matrix Ordo 3 x 3

```
==> Mat[0,0] : 3
==> Mat[0,1] : 4
==> Mat[0,2] : 0
==> Mat[1,0] : 9
==> Mat[1,1] : 10
==> Mat[1,2] : 11
==> Mat[2,0] : 3
==> Mat[2,1] : 12
==> Mat[2,2] : 7
```

Buat Matrix Ordo 1 x 3

```
==> Mat[0,0] : 3
==> Mat[0,1] : 1
==> Mat[0,2] : 5
```

Buat Matrix Ordo 3 x 1

```
==> Mat[0,0] : 7
==> Mat[1,0] : 6
==> Mat[2,0] : 4
```

## 2.2 Display Matrix

Buatlah fungsi display Matrix2D untuk menampilkan suatu matriks 2D (dengan argument atau parameter adalah matriks yang akan ditampilkan)

```
[2]: """
# Karena module menggunakan script py terpisah jadi untuk codenya dicomment
# Ini adalah function code dispMatrix2D (Menampilkan Matrix)

#----- Fungsi Menampilkan Matrix (dispMatrix2D) -----#
```

```

# argument menggunakan variable list matrix
# return value string

def dispMatrix2D(listMatrix) :
    strMatrix = ""
    pjgMax = 0 #Menampung nilai panjang karakter nilai terbesar

    #Mencari panjang karakter dari nilai terbesar
    for i in range(len(listMatrix)) :
        if pjgMax < len(str(max(listMatrix[i]))) :
            pjgMax = len(str(max(listMatrix[i])))

    #Menggabungkan string ke strMatrix
    for baris in listMatrix :
        i = 0 #Meghitung index
        for kolom in baris :

            if i == 0 : #cek bila index sama dengan 0
                strMatrix = strMatrix+"| "

            strMatrix = strMatrix+" %s%d " % ("┐
↳"*(pjgMax-len(str(kolom))),kolom)

            if i == len(baris)-1 : #Cek bila index sama dengan index terakhir┐
↳dari baris
                strMatrix = strMatrix+" |\n"

            i+=1

    return strMatrix

#----- End Fungsi Menampilkan Matrix (dispMatrix2D) -----#

"""

print("""
Matrix 1 :
%s

Matrix 2 :
%s

Matrix 3 :
%s

Matrix 4 :

```

```
%s

"""% (
    matrix dispMatrix2D(mat1),
    matrix dispMatrix2D(mat2),
    matrix dispMatrix2D(mat3),
    matrix dispMatrix2D(mat4)
) )
```

Matrix 1 :

```
| 3 1 6 |
| 2 10 3 |
| 4 1 5 |
```

Matrix 2 :

```
| 3 4 0 |
| 9 10 11 |
| 3 12 7 |
```

Matrix 3 :

```
| 3 1 5 |
```

Matrix 4 :

```
| 7 |
| 6 |
| 4 |
```

## 2.3 Square Matrix

Buatlah fungsi is square Matrix2D untuk mengecek apakah suatu matrix 2D adalah matriks square ? (dengan argument atau parameter adalah matriks yang akan dicek)

```
[3]: """
# Karena module menggunakan script py terpisah jadi untuk codenya dicomment
# Ini adalah function code isSquare (Mengecek apakah matrix memiliki panjang_
↳ baris dan kolom sama)

#----- Fungsi Mengecek Apakah Panjang Baris dan Kolom Matrix Sama -----#
# argument menggunakan variable list matrix
# return value boolean
```

```

def isSquare(listMatrix) :
    square = True
    if len(listMatrix) != len(listMatrix[0]) :
        square = False

    return square

#----- End Fungsi Mengecek Apakah Panjang Baris dan Kolom Matrix Sama_
↪(isSquare) -----#

"""

print("""
Matrix 1 :
%s
is Square ? %s

Matrix 2 :
%s
is Square ? %s

Matrix 3 :
%s
is Square ? %s

Matrix 4 :
%s
is Square ? %s
""") % (
    matrix.dispMatrix2D(mat1),matrix.isSquare(mat1),
    matrix.dispMatrix2D(mat2),matrix.isSquare(mat2),
    matrix.dispMatrix2D(mat3),matrix.isSquare(mat3),
    matrix.dispMatrix2D(mat4),matrix.isSquare(mat4)
) )

```

Matrix 1 :

	3	1	6	
	2	10	3	
	4	1	5	

is Square ? True

Matrix 2 :

```
| 3 4 0 |  
| 9 10 11 |  
| 3 12 7 |
```

is Square ? True

Matrix 3 :

```
| 3 1 5 |
```

is Square ? False

Matrix 4 :

```
| 7 |  
| 6 |  
| 4 |
```

is Square ? False

## 2.4 Penjumlahan Matriks

Buatlah fungsi Penjumlahan Matrix2D untuk menjumlahkan dua buah matriks (dengan argument atau parameter adalah matriks yang akan dijumlahkan). Di dalam fungsi tersebut dilakukan pengecekan apakah dua buah matriks dapat dijumlahkan

```
[4]: """  
# Karena module menggunakan script py terpisah jadi untuk codenya dicomment  
# Ini adalah function code addList (Menjumlahkan 2 matrix)  
  
#----- Fungsi Menjumlahkan 2 Matrix Sama Ordo (addList) -----#  
# argument menggunakan variable 2 list matrix  
# return value hasil penjumlahan matrix <list>  
  
def addList(listMatrix1, listMatrix2) :  
    pjgBaris = len(listMatrix1) == len(listMatrix2) #Membandingkan panjang  
    ↪ baris kedua matrix  
    pjgKolom = len(listMatrix1[0]) == len(listMatrix2[0]) #Membandingkan  
    ↪ panjang kolom kedua matrix  
  
    if pjgBaris and pjgKolom : #Jika Kedua Panjang baris dan kolom matrix sama  
        hasil = []  
        for i in range(len(listMatrix1)) :  
            tmpHasil = []
```



```

        for j in range(len(listMatrix1[i])) :
            tmpHasil.append(listMatrix1[i][j]+listMatrix2[i][j])
        ↪ #Menambahkan hasil penjumlahan kedua element matrix ke list tmpHasil

        hasil.append(tmpHasil) #Menambahkan element tmpHasil ke list hasil

    return hasil

else :
    print("'Ordo matrix tidak sama'")

#----- End Fungsi Menjumlahkan 2 Matrix Sama Ordo (addList) -----#

"""

jumlahMatrix = matrix.addList(mat1,mat2)

print("""
Matrix1 :
%s

Matrix2 :
%s

Jumlah :
%s

""" % (
    matrix.dispMatrix2D(mat1),
    matrix.dispMatrix2D(mat2),
    matrix.dispMatrix2D(jumlahMatrix)
) )

```

Matrix1 :

	3	1	6	
	2	10	3	
	4	1	5	

Matrix2 :

	3	4	0	
	9	10	11	
	3	12	7	

Jumlah :

	6	5	6	
	11	20	14	
	7	13	12	