

Tugas UAS Bagian 1

Nama Kelompok:

1. Aditya Adiaksa-20922320
2. Ahmad Mushawir-20922307
3. Akhmad Faeda I.-20922313
4. Dwi Putra P.-20922312
5. Zainuddin-20922319

Bagian 1

1. Buat suatu dictionary dengan Python yang dapat menyimpan informasi 4-5 jenis kopi dengan terlebih dahulu melakukan kuantifikasi terhadap bahan-bahan yang belum memiliki nilai, dengan satuan untuk setiap bahan adalah sama. Cantumkan satuan yang digunakan untuk jumlah bahan. [Nilai: 20]

Satuan yang kami gunakan untuk jumlah bahan adalah Gram (gr). Harga pergram yang kita gunakan sudah dikonversi ke IDR.

```
import colorama
from colorama import Fore, Style

coffee_dict = {
    'material': ['Coffee', 'Milk', 'Sugar', 'Chocolate', 'Water', 'Cream', 'Ice Cream', 'Whiskey', 'Cube Ice'],
    'harga_pergram': [168, 17, 16, 133, 3, 73, 33, 1257, 1],
    'merek': ['Exelso', 'UHT Diamond', 'Gulaku', 'Milo', 'Le Mineralle', 'Max Creamer', 'Campina', 'Jameson', 'Atlas Ic
    'americano': [30, 0, 0, 0, 155, 0, 0, 0, 0],
    'cappucino': [30, 150, 0, 10, 0, 0, 0, 0, 0],
    'caffelatte': [30, 150, 0, 0, 0, 0, 0, 0, 0],
    'afogato': [0, 30, 0, 0, 0, 30, 0, 0, 0],
    'machiato': [30, 0, 0, 0, 0, 15, 0, 0, 0]
}

coffees = ['americano', 'cappucino', 'caffelatte', 'afogato', 'machiato']

colorama.init()

for coffee in coffees:
    print(Fore.BLUE + Style.BRIGHT + coffee.capitalize() + ":")
    for i in range(len(coffee_dict['material'])):
        material = coffee_dict['material'][i]
        quantity = coffee_dict[coffee][i]
        print("  - " + material + ": " + Fore.GREEN + str(quantity) + " gram")
    print(Style.RESET_ALL)
```

COFFE MENU :

Americano:

- Coffee: 30 gram
- Milk: 0 gram
- Sugar: 0 gram
- Chocolate: 0 gram
- Water: 155 gram
- Cream: 0 gram
- Ice Cream: 0 gram
- Whiskey: 0 gram
- Cube Ice: 0 gram

Cappucino:

- Coffee: 30 gram
- Milk: 150 gram
- Sugar: 0 gram
- Chocolate: 10 gram
- Water: 0 gram
- Cream: 0 gram
- Ice Cream: 0 gram
- Whiskey: 0 gram
- Cube Ice: 0 gram

Caffelatte:

- Coffee: 30 gram
- Milk: 150 gram
- Sugar: 0 gram
- Chocolate: 0 gram
- Water: 0 gram
- Cream: 0 gram
- Ice Cream: 0 gram
- Whiskey: 0 gram
- Cube Ice: 0 gram

Afogato:

- Coffee: 0 gram
- Milk: 30 gram
- Sugar: 0 gram
- Chocolate: 0 gram
- Water: 0 gram
- Cream: 30 gram
- Ice Cream: 0 gram
- Whiskey: 0 gram
- Cube Ice: 0 gram

Machiato:

- Coffee: 30 gram
- Milk: 0 gram
- Sugar: 0 gram
- Chocolate: 0 gram
- Water: 0 gram
- Cream: 15 gram
- Ice Cream: 0 gram
- Whiskey: 0 gram
- Cube Ice: 0 gram

2. Konversi data harga kopi yang semula dalam USD untuk setiap pound menjadi dalam IDR untuk per kilogram dan gambarkan grafiknya untuk suatu rentang waktu tertentu. [Nilai: 20]

1 Pound = 0.453592 Kg

Asumsi kurs USD to IDR adalah 15.000

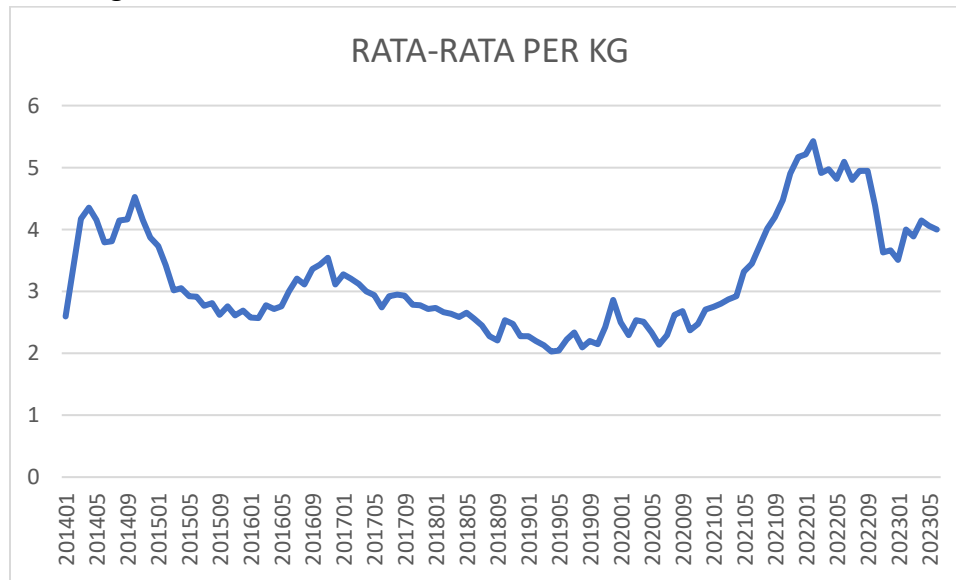
Data yang kami gunakan adalah data rata-rata harga perbulan sejak Januari 2014 hingga Juni 2023. Hasil data tersebut setelah dikonversi ke Kg dan harga IDR nya dapat dilihat pada table dibawah ini:

YEARMONTH	RATA-RATA PER KG	HARGA PER KG (IDR)
201401	2.593163229	38,897.45
201402	3.400331483	51,004.97
201403	4.166425208	62,496.38
201404	4.349335547	65,240.03
201405	4.156919554	62,353.79
201406	3.787576249	56,813.64
201407	3.805031434	57,075.47
201408	4.146520599	62,197.81
201409	4.161910977	62,428.66
201410	4.520678227	67,810.17
201411	4.150472347	62,257.09
201412	3.871270377	58,069.06
201501	3.731161484	55,967.42
201502	3.423712105	51,355.68
201503	3.019283249	45,289.25
201504	3.053100374	45,796.51
201505	2.923662675	43,854.94
201506	2.909011943	43,635.18
201507	2.769509323	41,542.64
201508	2.810591688	42,158.88
201509	2.62250573	39,337.59
201510	2.758536306	41,378.04
201511	2.610363498	39,155.45
201512	2.687988324	40,319.82
201601	2.579642637	38,694.64
201602	2.571837687	38,577.57
201603	2.772014578	41,580.22
201604	2.712296933	40,684.45
201605	2.755203124	41,328.05
201606	2.995673725	44,935.11
201607	3.206846682	48,102.70

YEARMONTH	RATA-RATA PER KG	HARGA PER KG (IDR)
201608	3.112718804	46,690.78
201609	3.357380537	50,360.71
201610	3.43050058	51,457.51
201611	3.544511157	53,167.67
201612	3.108940361	46,634.11
201701	3.278111166	49,171.67
201702	3.208285489	48,124.28
201703	3.11705137	46,755.77
201704	2.995956951	44,939.35
201705	2.936659938	44,049.90
201706	2.739145632	41,087.18
201707	2.921678513	43,825.18
201708	2.944074621	44,161.12
201709	2.927793721	43,916.91
201710	2.78048234	41,707.24
201711	2.776734955	41,651.02
201712	2.715876823	40,738.15
201801	2.735728941	41,035.93
201802	2.665414131	39,981.21
201803	2.640195216	39,602.93
201804	2.584985122	38,774.78
201805	2.652463808	39,786.96
201806	2.560545286	38,408.18
201807	2.448392892	36,725.89
201808	2.274779403	34,121.69
201809	2.203057977	33,045.87
201810	2.532011147	37,980.17
201811	2.47311618	37,096.74
201812	2.271645003	34,074.68
201901	2.278479338	34,177.19
201902	2.2006677	33,010.02
201903	2.126885164	31,903.28
201904	2.028138986	30,422.08
201905	2.042872085	30,643.08
201906	2.222713364	33,340.70
201907	2.339729556	35,095.94
201908	2.091857881	31,377.87
201909	2.19916535	32,987.48
201910	2.146585286	32,198.78
201911	2.425065866	36,375.99
201912	2.857455704	42,861.84
202001	2.502348927	37,535.23

YEARMONTH	RATA-RATA PER KG	HARGA PER KG (IDR)
202002	2.296590328	34,448.85
202003	2.534817032	38,022.26
202004	2.508169708	37,622.55
202005	2.338266653	35,074.00
202006	2.134296901	32,014.45
202007	2.292759292	34,391.39
202008	2.620847013	39,312.71
202009	2.676814887	40,152.22
202010	2.367816732	35,517.25
202011	2.474082015	37,111.23
202012	2.709533518	40,643.00
202101	2.746686449	41,200.30
202102	2.802088661	42,031.33
202103	2.866730646	43,000.96
202104	2.923741411	43,856.12
202105	3.316909931	49,753.65
202106	3.446589566	51,698.84
202107	3.721255706	55,818.84
202108	4.015733402	60,236.00
202109	4.193997329	62,909.96
202110	4.47029594	67,054.44
202111	4.903926476	73,558.90
202112	5.167352081	77,510.28
202201	5.21178462	78,176.77
202202	5.425459444	81,381.89
202203	4.913197227	73,697.96
202204	4.97302642	74,595.40
202205	4.818807932	72,282.12
202206	5.089846462	76,347.70
202207	4.802564335	72,038.47
202208	4.94602696	74,190.40
202209	4.948379721	74,225.70
202210	4.385102954	65,776.54
202211	3.630615579	54,459.23
202212	3.66067864	54,910.18
202301	3.50419039	52,562.86
202302	3.996256548	59,943.85
202303	3.88800853	58,320.13
202304	4.142764864	62,141.47
202305	4.060390989	60,905.86
202306	3.994794147	59,921.91

Berikut adalah grafik dari data tersebut :



3. Buat fungsi-fungsi untuk menghasilkan elemen elemen matriks A dan B dan buatlah kedua matriks tersebut, [Nilai: 20]

Berikut adalah code python untuk menghasilkan elemen matriks A dan B:

```
import pandas as pd
import numpy as np
```

```
def elemA(p, q, x, y):
    sum = 0
    for xi in x:
        sum += xi ** (p + q - 2)
    return sum
```

```
def elemB(p, q, x, y):
    sum = 0
    N = len(x)
    for i in range(N):
        sum += y[i] * x[i] ** (p + q - 2)
    return sum
```

```
def createMatrixA(M, x, y):
    mat = []
    for p in range(M + 1):
        row = []
        for q in range(M + 1):
            apq = elemA(p + 1, q + 1, x, y)
            row.append(apq)
        mat.append(row)
    return mat
```

```
def createMatrixB(M, x, y):
    mat = []
    for p in range(M + 1):
        row = []
```

```

        bpq = elemB(p + 1, 1, x, y)
        row.append(bpq)
        mat.append(row)
    return mat

def printMatrix(mat):
    for row in mat:
        print(row)

df = pd.read_csv('D:\\Documents\\ITB\\SEMESTER 1\\Programming in Science\\UAS\\Kopi_v1.csv')
x = df['INDEX'].values
y = df['AVERAGE_KG'].values
M = 3
A = createMatrixA(M, x, y)
B = createMatrixB(M, x, y)

npA = np.array(A)
npB = np.array(B)

C = np.linalg.solve(npA, npB)

for i in range(len(C)):
    coeff = C[i][0]

def elemA(p, q, x, y):
    sum = 0
    for xi in x:
        sum += xi ** (p + q - 2)
    return sum

def elemB(p, q, x, y):
    sum = 0
    N = len(x)
    for i in range(N):
        sum += y[i] * x[i] ** (p + q - 2)
    return sum

def createMatrixA(M, x, y):
    mat = []
    for p in range(M + 1):
        row = []
        for q in range(M + 1):
            apq = elemA(p + 1, q + 1, x, y)
            row.append(apq)
        mat.append(row)
    return mat

def createMatrixB(M, x, y):
    mat = []
    for p in range(M + 1):
        row = []
        bpq = elemB(p + 1, 1, x, y)
        row.append(bpq)
        mat.append(row)

```

```
return mat

def printMatrix(mat):
    for row in mat:
        print(row)

df = pd.read_csv('D:\\Documents\\ITB\\SEMESTER 1\\Programming in Science\\UAS\\Kopi_v2_pred.csv')

x = df['INDEX'].values
y = df['AVERAGE_KG'].values
M = 3
A = createMatrixA(M, x, y)
B = createMatrixB(M, x, y)

print("Matriks A",A)
print("Matriks B",B)
```

Berikut adalah hasil dari matriks A dan B

```
PS C:\Users\PLN> & C:/Users/PLN/AppData/Local/Microsoft/WindowsApps/python3.11.exe "d:/Documents/ITB/SEMESTER 1/Programming in Science/UAS/Bagian1No3.py"
Matriks A [[18, 153, 1785, 23409], [153, 1785, 23409, 327369], [1785, 23409, 327369, 4767633], [23409, 327369, 4767633, 71397705]]
Matriks B [[93.7116], [835.9869], [9969.719700000001], [132481.2261]]
```