

TUGAS LAB PEMBELAJARAN MESIN

Nama : Meina Lisa

NIM : 191402032

TUGAS 2

1. Kapan kita harus memakai tipe data String yang diapit oleh tanda petik tunggal (' ') dan tanda petik ganda (" ") !

Python dapat memanipulasi string atau teks yang diekspresikan dalam beberapa cara, dapat disertakan dalam tanda kutip tunggal ('...') atau tanda kutip ganda ("...") dengan hasil yang sama. Dalam interactive interpreter, output string yang diapit dengan tanda kutip dan karakter khusus dipisahkan dengan backslash (\), seperti \n, \t, dsb. Meskipun kadang terlihat beda dari input (tanda kutip terlampir dapat berubah), kedua string tersebut setara. String disertakan dalam tanda kutip ganda jika string berisi kutipan tunggal dan tidak ada tanda kutip ganda, jika tidak maka akan dilampirkan dalam tanda kutip tunggal (sebaliknya). Fungsi print() menghasilkan keluaran yang lebih mudah dibaca, dengan menghilangkan tanda kutip terlampir dan dengan mencetak karakter yang dipisahkan dan spesial:

Menggunakan petik tunggal karena di dalamnya ada petik ganda "Isn't,"

```
>>> "Isn't," they said.'
```

```
"Isn't," they said.'
```

Menggunakan \' untuk mencetak ' pada isn't

```
>>> print("Isn't," they said.)
```

```
"Isn't," they said.
```

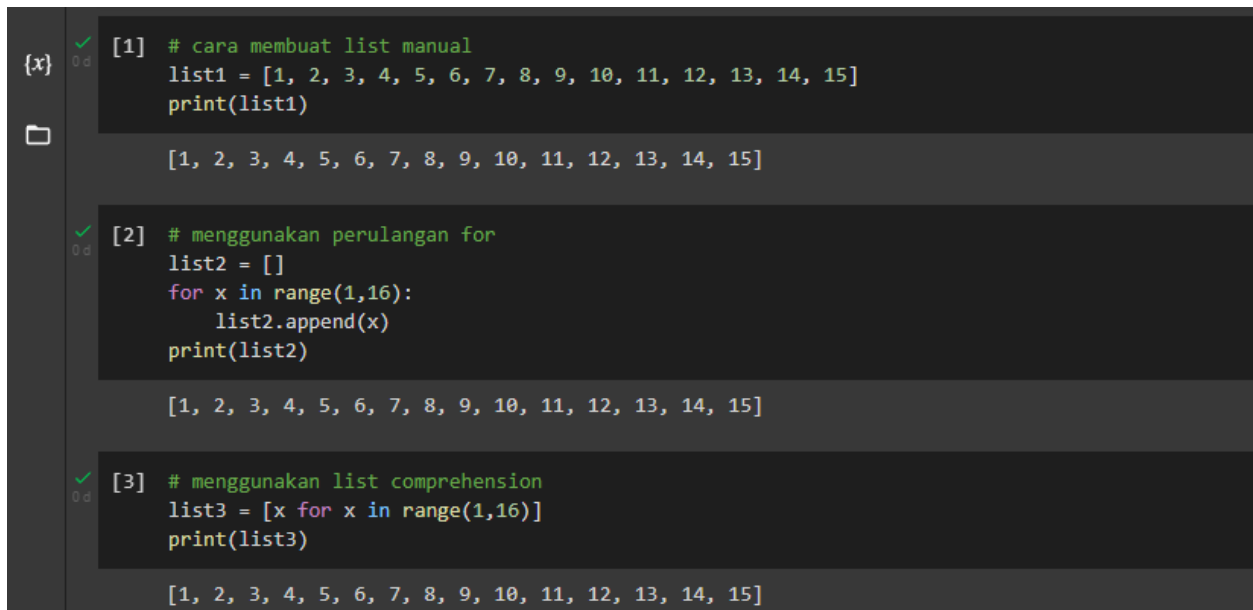
2. Jelaskan tentang indexing dan slicing dan berikan contohnya!

Array indexing adalah cara dalam mengakses elemen array, Anda dapat mengakses elemen pada array dengan Nomor Index, Contoh: variabel_i[index ke berapa] ATAU variabel_a[elemen ke berapa][index ke berapa].

Sedangkan Slicing adalah Memotong elemen Array menggunakan index yang diberikan. Contoh: [mulai:selesai] atau [mulai:] atau [:selesai] .

3. Jelaskan dan Buatlah contoh penggunaan perulangan pada list menggunakan for dan list comprehension!

Untuk membuat list dalam Python, kita bisa menggunakan cara manual, dengan perulangan for, dan dengan list comprehension. List comprehension adalah cara mudah untuk mendefinisikan list secara otomatis dalam satu baris perintah. Ini sangat berguna jika anggota list yang hendak kita buat cukup banyak. List comprehension berbentuk perulangan for yang sedikit berbeda.



```
[x] ✓ [1] # cara membuat list manual
00 list1 = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]
print(list1)

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

✓ [2] # menggunakan perulangan for
00 list2 = []
for x in range(1,16):
    list2.append(x)
print(list2)

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]

✓ [3] # menggunakan list comprehension
00 list3 = [x for x in range(1,16)]
print(list3)

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]
```

4. Jelaskan perbedaan tipe data Set dan Tuple!

Tuple adalah tipe data kolektif yang bersifat immutable dan ordered. Dalam artian ia sama dengan list, hanya saja tuple tidak bisa diedit. Syntax: tuple(a, b, c)

Sedangkan Set adalah tipe data kolektif yang bersifat unordered (tidak bisa indexing dan slicing), unique (semua nilainya harus unik), dan changeable (bisa ditambah dan dihapus). Syntax: set{a, b, c}

5. Buatlah sebuah function yang mengembalikan sebuah list yang berisi nilai 'value' pada tipe data Dictionary! (input parameternya itu sebuah variable dictionary dan outputnya/return nya itu berupa list).

```
[4] def dictToList(dictionary):  
    listA = [dictionary[key] for key in dictionary]  
    return listA  
  
    # pemanggilan fungsi  
    diction = {"key1": "value1", "key2": "value2", "key3": "value3"}  
    print(dictToList(diction))  
  
['value1', 'value2', 'value3']
```

6. Buatlah sebuah folder bernama 'nimkalian' (pakai python, bukan dari file explorer), lalu buat sebuah file.txt yang berisi teks 'pesan kalian untuk lab kedepannya'.

```
[6] import os  
os.makedirs('191402032')  
  
with open('191402032/file.txt', 'w+') as c:  
    c.write('pesan kalian untuk lab kedepannya')  
    c.write('\nmaaf bang tolong tugasnya jangan banyak banget rasanya seperti lagi ngerjain ujian')  
  
with open('191402032/file.txt', 'r') as c:  
    print(c.read())  
  
pesan kalian untuk lab kedepannya  
maaf bang tolong tugasnya jangan banyak banget rasanya seperti lagi ngerjain ujian
```

7. Terdapat sebuah numpy array 2 dimensi sebagai berikut

4 columns

2	-5	-11	0
-9	4	6	13
4	7	12	-2

← 3 rows

Buatlah/Tampilkan:

a. Banyak Entry

b. Dimensi Array

c. Ukuran Array

d. Kodingan untuk menampilkan

e. Kodingan untuk menampilkan

-5	-11	0
4	6	13

-9	4	6
4	7	12

```
[1] import numpy as np
arrA = np.array([2, -5, -11, 0, -9, 4, 6, 13, 4, 7, 12, -2]).reshape(3, 4)
print(arrA)

print('banyak entry = ', arrA.size)
print('dimensi array = ', arrA.ndim)
print('ukuran array = ', arrA.shape)

print(arrA[0:2, 1:4])
print(arrA[1:4, 0:3])
```

```
[[ 2 -5 -11  0]
 [-9  4  6 13]
 [ 4  7 12 -2]]
banyak entry = 12
dimensi array = 2
ukuran array = (3, 4)
[[ -5 -11  0]
 [  4  6 13]]
[[-9  4  6]
 [ 4  7 12]]
```

8. Buatlah sebuah dataframe yang membaca url berikut:

<https://raw.githubusercontent.com/achmatim/data-mining/main/Dataset/iris.csv>

```
[2] import pandas as pd
url = 'https://raw.githubusercontent.com/achmatim/data-mining/main/Dataset/iris.csv'
dfa = pd.read_csv(url)

dfa
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Label
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...
145	6.7	3.0	5.2	2.3	Iris-virginica

a. Tampilkan statistik deskriptif untuk data numerik dan data non-numerik

The screenshot shows two Jupyter Notebook cells. The first cell contains the code `dfA.describe()` and displays a table of descriptive statistics for numerical data. The second cell contains the code `dfA.describe(include=["object", "bool"])` and displays a table of descriptive statistics for non-numerical data.

Cell 1: Numerical Data Statistics

```
[3] # data numerik
dfA.describe()
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

Cell 2: Non-numerical Data Statistics

```
# data non-numerik
dfA.describe(include=["object", "bool"])
```

	Label
count	150
unique	3
top	Iris-setosa
freq	50

b. Buatlah kodingan untuk memilih data dengan label 'Iris-setosa'

The screenshot shows a Jupyter Notebook cell with the code `dfA[dfA['Label']=='Iris-setosa']` and displays a table of the selected data rows.

```
[4] # pilih data yang label iris-setosa
dfA[dfA['Label']=='Iris-setosa']
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Label
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
5	5.4	3.9	1.7	0.4	Iris-setosa
6	4.6	3.4	1.4	0.3	Iris-setosa