


Nama: Nadhira Anindita Ralena NIM: 065002300021	 Praktikum Statistika	MODUL 2 Nama Dosen: Dedy Sugiarto
Hari/Tanggal: Rabu, 13 maret 2024		Nama Asisten Labratorium: 1. Tarum Widyasti 064002200027 2. Kharisma Maulida 064002200024

Tipe Data, Filter Data & Koneksi ke Database

1. Teori Singkat

Python memiliki beberapa tipe data dasar, di antaranya:

- Integer (int): Representasi bilangan bulat, misalnya: 5, -3, 100.
- Float (float): Representasi bilangan pecahan, misalnya: 3.14, 2.718.
- String (str): Urutan karakter, misalnya: 'hello', "world".
- Boolean (bool): Representasi nilai kebenaran, yaitu True atau False.
- List (list): Kumpulan elemen yang terurut dan dapat diubah, misalnya: [1, 2, 3, 4], ['apple', 'banana', 'cherry'].
- Tuple (tuple): Kumpulan elemen yang terurut dan tidak dapat diubah, misalnya: (1, 2, 3), ('red', 'green', 'blue').
- Dictionary (dict): Kumpulan pasangan kunci-nilai yang tidak terurut, misalnya: {'name': 'John', 'age': 30}.
- Set (set): Kumpulan elemen yang unik dan tidak terurut, misalnya: {1, 2, 3, 4}.

Filter Data dalam Python:

Untuk melakukan filter data dalam Python, Anda dapat menggunakan berbagai cara, tergantung pada struktur data yang Anda gunakan. Dalam konteks DataFrame, seperti yang digunakan dalam Pandas, Anda dapat menggunakan metode query() atau pengindeksan boolean.

2. Alat dan Bahan

Hardware : Laptop/PC

Software : R Studio

3. Elemen Kompetensi

Terdapat beberapa tipe data di Jupyter antara lain vektor, matriks dan data frame. Cantumkan setiap output yang dihasilkan dari console Jupyter, ke kolom yang sudah disediakan.

a. Latihan pertama – Vektor

1. Tuliskan Perintah berikut ini di jupyter notebook

```
a = [1, 2, -5, 0.3, 6, -2, 4] # numeric vector
b = ["one", "two", "three"] # character vector
c = [True, True, True, False, True] # logical vector
print(a)
print(b)
print(c)
```

Output:

```
In [1]: a = [1, 2, -5, 0.3, 6, -2, 4] # numeric vector
        b = ["one", "two", "three"] # character vector
        c = [True, True, True, False, True] # logical vector
        print(a)
        print(b)
        print(c)

[1, 2, -5, 0.3, 6, -2, 4]
['one', 'two', 'three']
[True, True, True, False, True]
```

b. Latihan Kedua – Matriks

1. Seluruh kolom dalam sebuah matriks harus memiliki tipe yang sama (numerik semua, karakter semua, dll) dan memiliki panjang yang sama.

**gunakan nama variable dengan nama anda masing-masing*

Output:

```
#MATRIKS
import numpy as np
cells = [3, 15, -27, 38]
r_nama = ["R1", "R2"]
c_nama = ["C1", "C2"]
nama_matrix = np.matrix(cells).reshape(2, 2)
print(nama_matrix)
```

```
In [2]: #MATRIKS
import numpy as np
cells = [3, 15, -27, 38]
r_nama = ["dhira1", "dhira2"]
c_nama = ["dhira1", "dhira2"]
nama_matrix = np.matrix(cells).reshape(2, 2)
print(nama_matrix)

[[ 3 15]
 [-27 38]]
```

c. Latihan Ketiga – Data Frame

1. Mengubah data input menjadi data frame

*gunakan nama variable dengan nama anda masing-masing

```
import pandas as pd
import numpy as np

nama1 = [1, 2, 3, 4]
nama2 = ["red", "white", "red", np.nan] # Menggunakan np.nan untuk
merepresentasikan NA
nama3 = [True, True, True, False]

dataku = pd.DataFrame({'ID': nama1, 'Color': nama2, 'Passed': nama3})
print(dataku)
```

Output:

```
In [3]: import pandas as pd
import numpy as np

dhira1 = [1, 2, 3, 4]
dhira2 = ["red", "white", "red", np.nan] # Menggunakan np.nan untuk merepresentasikan NA
dhira3 = [True, True, True, False]

dataku = pd.DataFrame({'ID': dhira1, 'Color': dhira2, 'Passed': dhira3})
print(dataku)
```

	ID	Color	Passed
0	1	red	True
1	2	white	True
2	3	red	True
3	4	NaN	False

2. Selanjutnya ketikkan perintah dibawah ini

```
import pandas as pd

data_nama = pd.DataFrame({'id': list('abcdefghij'), 'x': list(range(1, 11)), 'y': list(range(11, 21))})
print(data_nama)
```

<https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html>

Output:

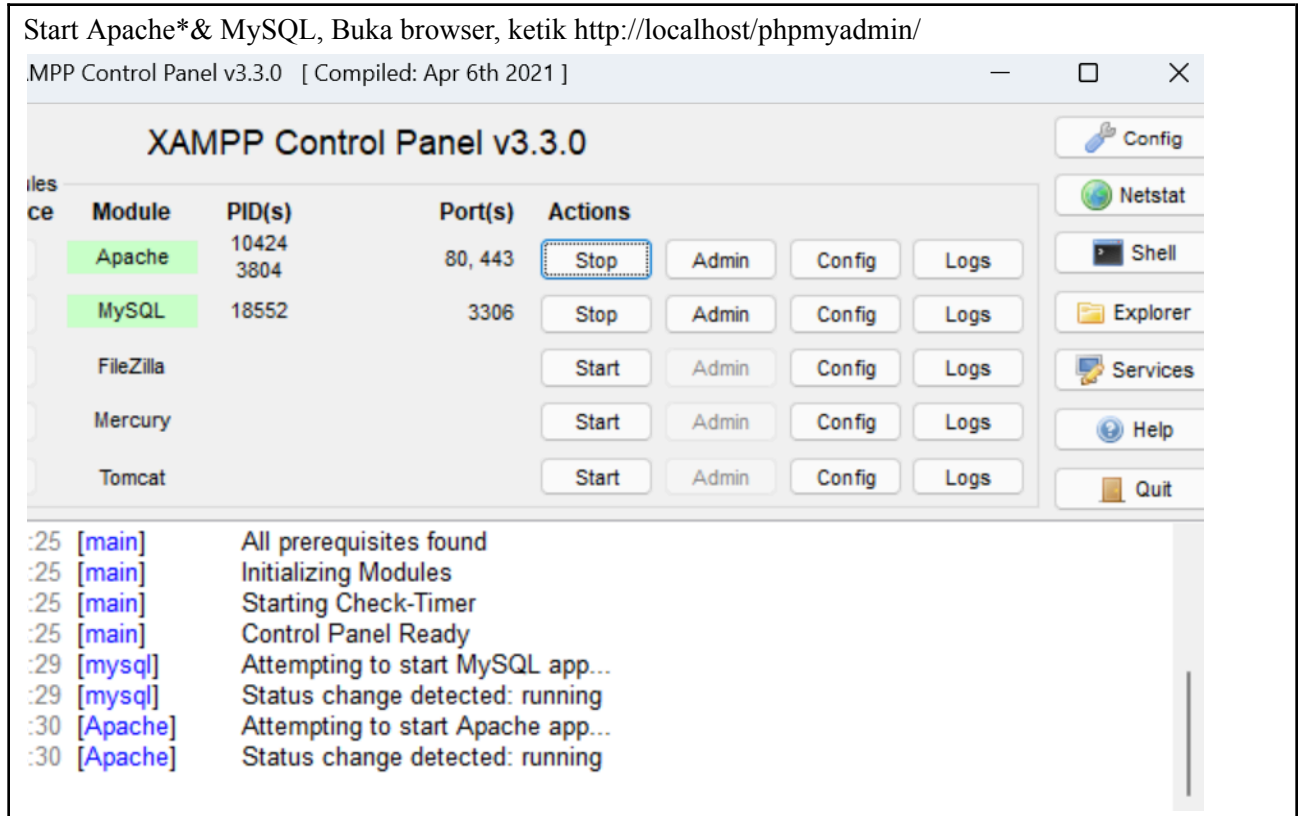
```
In [4]: import pandas as pd  
  
data_nama = pd.DataFrame({'id': list('abcdefghij'), 'x': list(range(1, 11)), 'y': list(range(11, 21))})  
print(data_nama)
```

	id	x	y
0	a	1	11
1	b	2	12
2	c	3	13
3	d	4	14
4	e	5	15
5	f	6	16
6	g	7	17
7	h	8	18
8	i	9	19
9	j	10	20

<https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html>

d. Latihan Keempat – Koneksi ke Database

1. Buat sebuah nama database terlebih dahulu dengan nama houseprices di phpmyadmin, Lalu klik menu import



Buat basis data

utf8mb4_general_ci

Buat

☐ Pilih Semua

Hapus

	Basis data	Penyortiran	Tindakan
<input type="checkbox"/>	dashboard_gugustugas	utf8mb4_general_ci	Check privileges
<input type="checkbox"/>	data_nama	utf8mb4_general_ci	Check privileges
<input type="checkbox"/>	houseprices1	utf8mb4_general_ci	Check privileges
<input type="checkbox"/>	information_schema	utf8_general_ci	Check privileges
<input type="checkbox"/>	lstm_beras	utf8mb4_general_ci	Check privileges
<input type="checkbox"/>	mysql	utf8mb4_general_ci	Check privileges
<input type="checkbox"/>	performance_schema	utf8_general_ci	Check privileges
<input type="checkbox"/>	phpmyadmin	utf8_bin	Check privileges
<input type="checkbox"/>	test	latin1_swedish_ci	Check privileges

total: 9

2. Pilih file yang ingin di import ke database (untuk file nama_excel.csv), Ceklis the first line of the file contains the table column name untuk membuat baris pertama pada file excel tersebut menjadi nama atributnya atau nama kolomnya pada database.

Status

Akun pengguna

Ekspor

Impor

Pengaturan

Replikasi

Variabel

Set Karakter

Mesin

Plugin

Mengimpor ke dalam server saat ini

Berkas untuk impor:

Dapat berupa berkas terkompresi (gzip, bzip2) atau tidak.
A compressed file's name must end in `[format].[compression]`. Example: `.sql.zip`

Telusuri komputer Anda: (Batas ukuran: 40MB)

Choose File

houseprices - houseprices.csv

Anda juga dapat drag dan drop sebuah file pada halaman manapun.

Set karakter berkas:

utf-8

Impor sebagian:

☒ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit.
This might be a good way to import large files, however it can break transactions.

Lewati beberapa kueri (untuk SQL) mulai dari yang pertama:

0

Other options

☒ Perbolehkan cek foreign key

Format

CSV

Catatan: Jika berkas mengandung banyak tabel, mereka akan digabungkan menjadi satu.

Impor sebagian:

☒ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit.
This might be a good way to import large files, however it can break transactions.

Lewati beberapa kueri (untuk SQL) mulai dari yang pertama:

0

Other options

☒ Perbolehkan cek foreign key

Format

CSV

Catatan: Jika berkas mengandung banyak tabel, mereka akan digabungkan menjadi satu.

Opsi format-spesifik:

Status

Akun pengguna

Ekspor

Impor

Pengaturan

Replikasi

Variabel

Set Karakter

Mesin

Plugin

Catatan: Jika berkas mengandung banyak tabel, mereka akan di gabungkan menjadi satu.

Opsi format-spesifik:

☐ Memperbarui data ketika duplikat kunci ditemukan saat impor (add ON DUPLICATE KEY UPDATE)

Kolom dipisahkan oleh:
','

Kolom diapit oleh:
'"'

Kolom dikembalikan oleh:
'"'

Baris diakhiri dengan:
auto

Nama dari tabel baru (tidak wajib):

Nama dari database baru (pilihan):

Impor seberapa banyak baris ini (opsional):

☒ The first line of the file contains the table column names (if this is unchecked, the first line will become part of the data)

☐ Jangan gugurkan sewaktu galat INSERT

Impor

localhost/phpmyadmin/index.php?route=/import

Servis: 527.0.0.1

Basis data

Status

Akun pengguna

Ekspor

Impor

Pengaturan

Replikasi

Variabel

Set Karakter

Mesin

Plugin

Impor telah selesai. 3 kueri dilakukan.

The following structures have either been created or altered. Here you can:

- View a structure's contents by clicking on its name.
- Change any of its settings by clicking the corresponding "Options" link.
- Edit struktur dengan mengklik link "Struktur".

CSV_DB_10 (Opsl)

houseprices__houseprices (Struktur) (Opsl)

(houseprices - houseprices.csv)

MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0.0011 detik.)

CREATE DATABASE IF NOT EXISTS `CSV_DB_10` DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;

Edit kotak [[Ubah]] [[Buat kode PHP]]

Error #1046 No database selected

MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0.0087 detik.)

CREATE TABLE IF NOT EXISTS `CSV_DB_10`.`houseprices__houseprices` (`Price` int(6), `Sqft` int(4), `Bedrooms` int(1), `Bathrooms` int(1), `Offers` int(1), `Brick` varchar(3), `Neighborhood` varchar(5)) DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;

Edit kotak [[Ubah]] [[Buat kode PHP]]

128 baris ditambahkan. (Pencarian dilakukan dalam 0.0053 detik.)

INSERT INTO `CSV_DB_10`.`houseprices__houseprices` (`Price`, `Sqft`, `Bedrooms`, `Bathrooms`, `Offers`, `Brick`, `Neighborhood`) VALUES (114200, 1790, 2, 2, 2, 'No', 'East'), (114200, 2030, 4, 2, 3, 'No', 'East'), (114800, 1740, 3, 2, 1, 'No', 'East'), (94700, 1980, 3, 2, 3, 'No', 'East'), (115800, 2130, 3, 3, 3, 'No', 'East'), (114600, 1780, 3, 2, 2, 'No', 'North'), (151600, 1830, 3, 3, 3, 'Yes', 'West'), (150700, 2160, 4, 2, 2, 'No', 'West'), (119200, 2110, 4, 2, 3, 'No', 'East'), (104000, 1730, 3, 3, 3, 'No', 'East'), (132500, 2030, 3, 2, 3, 'Yes', 'East'), (123000, 1870, 2, 2, 2, 'Yes', 'East'), (102600, 1910, 3, 2, 4, 'No', 'North'), (126300, 2150, 3, 3, 5, 'Yes', 'North'), (176800, 2590, 4, 3, 4, 'No', 'West'), (145800, 1780, 4, 2, 1, 'No', 'West'), (147100, 2190, 3, 3, 4, 'Yes', 'East'), (83600, 1990, 3, 3, 4, 'No', 'North'), (111400, 1700, 2, 2, 1, 'Yes', 'East'), (167200, 1920, 3, 3, 2, 'Yes', 'West'), (116200, 1790, 3, 2, 3, 'No', 'East'), (113800, 2000, [...])

Ubah

3. Klik go, Rename nama tabel sesuai dengan nama anda

Ubah nama basis data menjadi

nadhira_houseprices1

☒ Adjust privileges

phpMyAdmin

Server: 127.0.0.1 » Database: nadhira_houseprices1 » Tabel: houseprices__houseprices

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Pelacakan Trigger

Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

Menampilkan baris 0 - 24 (total 128, Pencarian dilakukan dalam 0,0004 detik.)

SELECT * FROM `houseprices__houseprices`

☐ Profil ☐ [Edit kotak] ☐ [Ubah] ☐ [Jelaskan SQL] ☐ [Buat kode PHP] ☐ [Segarkan]

1 > >> ☐ Tampilkan semua Jumlah baris: 25 Saring baris: Cari di tabel ini

Extra options

Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood
114300	1790	2	2	2	No	East
114200	2030	4	2	3	No	East
114800	1740	3	2	1	No	East
94700	1980	3	2	3	No	East
119800	2130	3	3	3	No	East
114600	1780	3	2	2	No	North
151600	1830	3	3	3	Yes	West
150700	2160	4	2	2	No	West
119200	2110	4	2	3	No	East
104000	1730	3	3	3	No	East
132500	2030	3	2	3	Yes	East
123000	1870	2	2	2	Yes	East
102600	1910	3	2	4	No	North
126300	2150	3	3	5	Yes	North
176800	2590	4	3	4	No	West
145800	1780	4	2	1	No	West
147100	2190	3	3	4	Yes	East
83600	1990	3	3	4	No	North
111400	1700	2	2	1	Yes	East
167200	1920	3	3	2	Yes	West
116200	1790	3	2	3	No	East
113800	2000	3	2	4	No	North
91700	1690	3	2	3	No	North
106100	1820	3	2	3	Yes	North
Konsol	1770	4	3	2	Yes	East

4. Kembali ke jupyter notebook, lalu instal dahulu library yang dibutuhkan pada python. Jika belum tersedia, maka lakukan instruksi:

pip install mysql-connector-python

```
In [5]: pip install mysql-connector-python
```

```
Collecting mysql-connector-python
  Obtaining dependency information for mysql-connector-python from https://files.pythonhosted.org/packages/d9/91/007a0d60fee8db4f7385075dc50bf62d2d359b417b374ec06b06ce6c2d64/mysql\_connector\_python-8.3.0-cp311-cp311-win\_amd64.whl.metadata
  Downloading mysql_connector_python-8.3.0-cp311-cp311-win_amd64.whl.metadata (2.0 kB)
  Downloading mysql_connector_python-8.3.0-cp311-cp311-win_amd64.whl (15.4 MB)
----- 0.0/15.4 MB ? eta --:--
----- 0.0/15.4 MB ? eta --:--
----- 0.1/15.4 MB 1.7 MB/s eta 0:00:10
----- 0.1/15.4 MB 1.2 MB/s eta 0:00:13
----- 0.2/15.4 MB 1.2 MB/s eta 0:00:13
----- 0.2/15.4 MB 1.3 MB/s eta 0:00:13
----- 0.3/15.4 MB 1.1 MB/s eta 0:00:14
----- 0.3/15.4 MB 1.0 MB/s eta 0:00:15
----- 0.4/15.4 MB 1.2 MB/s eta 0:00:13
----- 0.4/15.4 MB 1.1 MB/s eta 0:00:14
----- 0.4/15.4 MB 1.1 MB/s eta 0:00:14
----- 0.6/15.4 MB 1.2 MB/s eta 0:00:13
----- 0.6/15.4 MB 1.1 MB/s eta 0:00:14
----- 0.7/15.4 MB 1.3 MB/s eta 0:00:12
```

5. Lalu jalankan perintah dibawah ini

Output:

```
import mysql.connector

# Membuat koneksi ke MySQL
connection = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="houseprices"
)

# Membuat objek cursor untuk mengeksekusi kueri
cursor = connection.cursor()

try:
    # Mengeksekusi kueri SQL
    my_query = "SELECT * FROM nama;"
    cursor.execute(my_query)

    # Mengambil semua hasil kueri
    result = cursor.fetchall()

    # Menampilkan hasil kueri
    print("\nHasil Kueri:")
    for row in result:
        print(row)

finally:
    # Menutup kursor dan koneksi
    cursor.close()
    connection.close()
```

```
In [24]: import mysql.connector

# Membuat koneksi ke MySQL
connection = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="nadhira_houseprices1"
)

# Membuat objek cursor untuk mengeksekusi kueri
cursor = connection.cursor()

try:
    # Mengeksekusi kueri SQL
    my_query = "SELECT * FROM houseprices__houseprices;"
    cursor.execute(my_query)

    # Mengambil semua hasil kueri
    result = cursor.fetchall()

    # Menampilkan hasil kueri
    print("\nHasil Kueri:")
    for row in result:
        print(row)

finally:
    # Menutup cursor dan koneksi
    cursor.close()
    connection.close()
```

```
Hasil Kueri:
(114300, 1790, 2, 2, 2, 'No', 'East')
(114200, 2030, 4, 2, 3, 'No', 'East')
(114800, 1740, 3, 2, 1, 'No', 'East')
(94700, 1980, 3, 2, 3, 'No', 'East')
(119800, 2130, 3, 3, 3, 'No', 'East')
(114600, 1780, 3, 2, 2, 'No', 'North')
(151600, 1830, 3, 3, 3, 'Yes', 'West')
(150700, 2160, 4, 2, 2, 'No', 'West')
(119200, 2110, 4, 2, 3, 'No', 'East')
(104000, 1730, 3, 3, 3, 'No', 'East')
(132500, 2030, 3, 2, 3, 'Yes', 'East')
(123000, 1870, 2, 2, 2, 'Yes', 'East')
(102600, 1910, 3, 2, 4, 'No', 'North')
(126300, 2150, 3, 3, 5, 'Yes', 'North')
(176800, 2590, 4, 3, 4, 'No', 'West')
(145800, 1780, 4, 2, 1, 'No', 'West')
(147100, 2190, 3, 3, 4, 'Yes', 'East')
(13700, 1000, 2, 2, 2, 'No', 'West')
```

6. Jalankan perintah dibawah ini:

*Perintah ini akan menampilkan 86 baris data hasil filter.

Output:

```
In [27]: import pandas as pd
#mengkonvers hasil kueri ke dataframe pandas
df = pd.DataFrame(result, columns=[desc[0] for desc in cursor.description])

#filter data berdasarkan kolom "brick" yg bernilai 'no'
df_filtered = df[df['Brick']=='No']

#menampilkan hasil filter
print("\nHasil Fokter:")
print(df_filtered)
```

Hasil Fokter:

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood
0	114300	1790	2	2	2	No	East
1	114200	2030	4	2	3	No	East
2	114800	1740	3	2	1	No	East
3	94700	1980	3	2	3	No	East
4	119800	2130	3	3	3	No	East
..
120	110400	1930	2	3	3	No	North
121	105600	1930	3	3	3	No	East
125	113500	2070	2	2	2	No	North
126	149900	2020	3	3	1	No	West
127	124600	2250	3	3	4	No	North

[86 rows x 7 columns]

In [27]:

7. Jalankan perintah dibawah ini:

*Perintah ini akan menampilkan 105 baris data hasil filter.

Output:

```
In [31]: import pandas as pd
#mengkonvers hasil kueri ke dataframe pandas
df = pd.DataFrame(result, columns=[desc[0] for desc in cursor.description])

#filter data berdasarkan kolom "brick" yg bernilai 'no'
df_filtered = df[(df['Brick'] == 'No') | (df['Neighborhood'] == 'East')]

#menampilkan hasil filter
print(df_filtered)
```

	Price	SqFt	Bedrooms	Bathrooms	Offers	Brick	Neighborhood
0	114300	1790	2	2	2	No	East
1	114200	2030	4	2	3	No	East
2	114800	1740	3	2	1	No	East
3	94700	1980	3	2	3	No	East
4	119800	2130	3	3	3	No	East
..
123	119700	1900	3	3	3	Yes	East
124	147900	2160	4	3	3	Yes	East
125	113500	2070	2	2	2	No	North
126	149900	2020	3	3	1	No	West
127	124600	2250	3	3	4	No	North



[105 rows x 7 columns]

e. Latihan Keenam – Tugas












Buat sebuah database serta tabel di dalamnya (bisa gunakan data teman dipraktikum ke 1). Lakukan koneksi python ke database serta berikan beberapa filter data sesuai yang anda inginkan. Tampilkan data tersebut

1. Buatlah terlebih dahulu Database baru dengan format PS2[NamaAnda]


Basis data

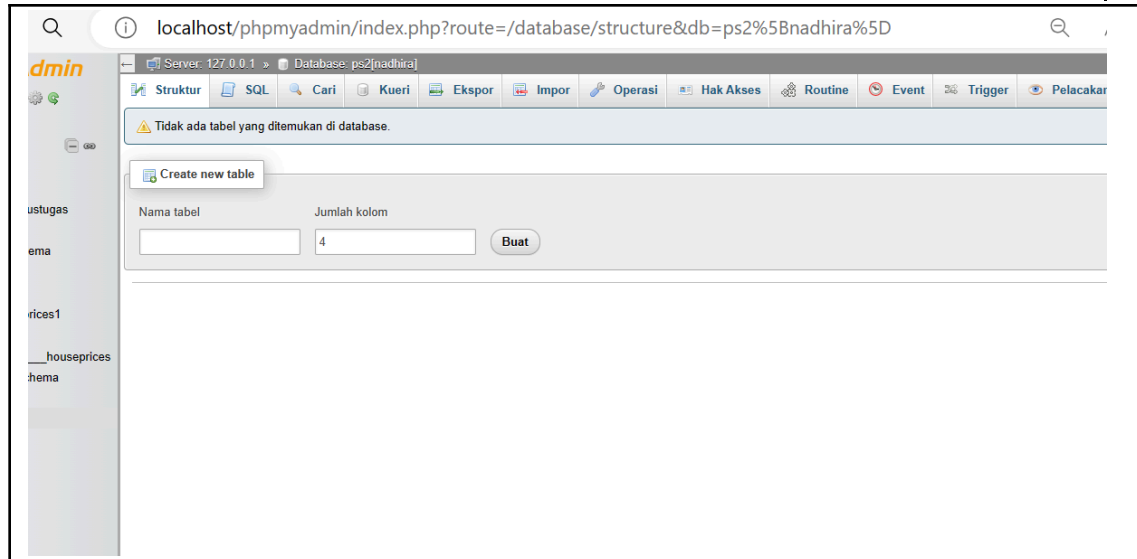
 **Buat basis data** 

☐ **Pilih Semua**

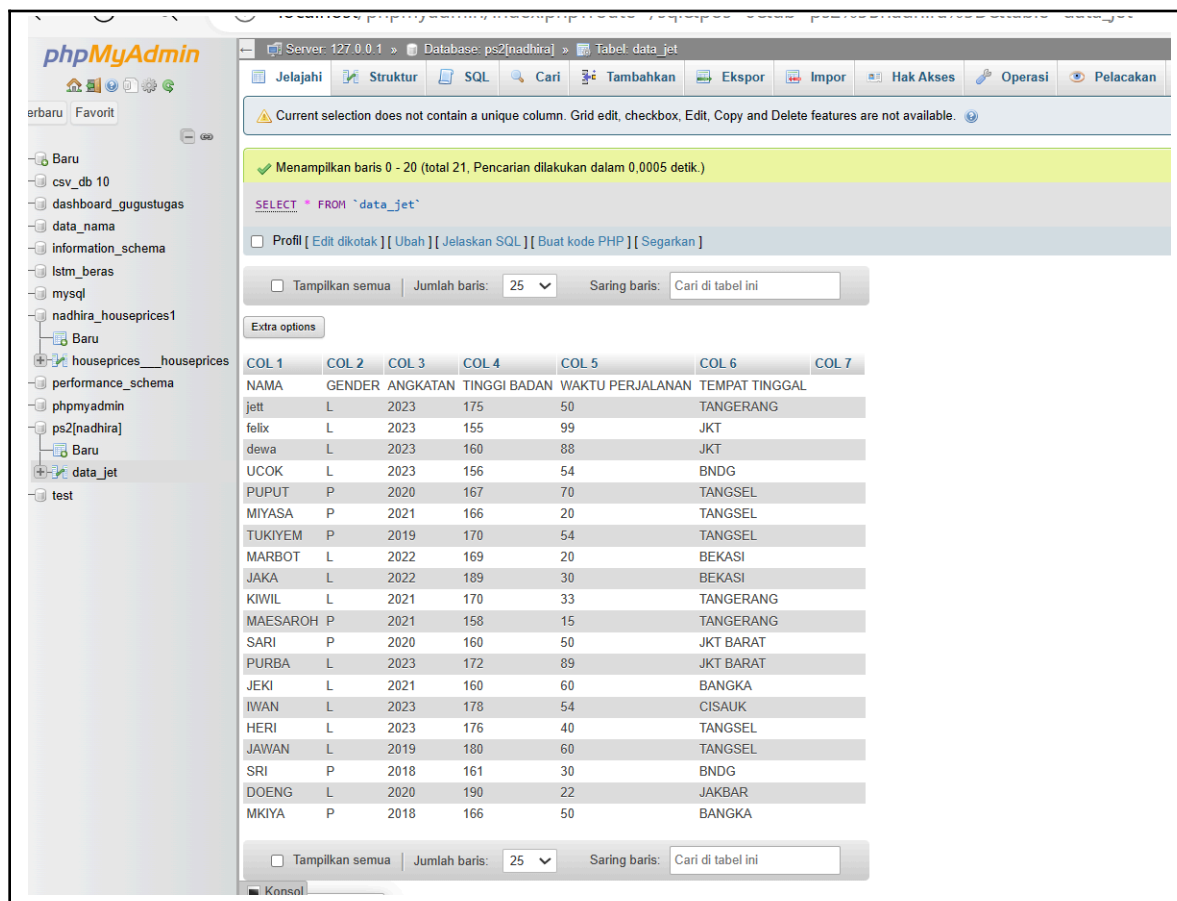
Basis data	Penyortiran	Tindakan
<input type="checkbox"/> csv_db 10	utf8_general_ci	 Check privileges
<input type="checkbox"/> dashboard_gugustugas	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/> data_nama	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/> information_schema	utf8_general_ci	 Check privileges
<input type="checkbox"/> lstm_beras	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/> mysql	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/> nadhira_houseprices1	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/> performance_schema	utf8_general_ci	 Check privileges
<input type="checkbox"/> phpmyadmin	utf8_bin	 Check privileges
<input type="checkbox"/> ps2[nadhira]	utf8mb4_general_ci	 Check privileges
<input type="checkbox"/> test	latin1_swedish_ci	 Check privileges

Total: 11

 **Perhatian:** Pengaktifan statistik basis data dapat mengakibatkan lalu lintas besar antara webserver dan server MySQL.



2. Import file berformat csv yang telah anda buat sebelumnya di Praktikum 1, dimana file tersebut berisi 6 kolom dan 20 baris data.



3. Koneksikan Python ke Database tersebut sebagaimana yang telah anda lakukan pada Elemen Kompetensi 1 dimodul kedua ini dengan menyesuaikan kembali nama Database baru yang sudah dibuat.

```
In [34]: import mysql.connector

# Membuat koneksi ke MySQL
connection = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="ps2[nadhira]"
)

# Membuat objek cursor untuk mengeksekusi kueri
cursor = connection.cursor()

try:
    # Mengeksekusi kueri SQL
    my_query = "SELECT * FROM data_jet;"
    cursor.execute(my_query)

    # Mengambil semua hasil kueri
    result = cursor.fetchall()

    # Menampilkan hasil kueri
    print("\nHasil Kueri:")
    for row in result:
        print(row)

finally:
    # Menutup kursor dan koneksi
    cursor.close()
    connection.close()

Hasil Kueri:
('NAMA', 'GENDER', 'ANGKATAN', 'TINGGI BADAN', 'WAKTU PERJALANAN', 'TEMPAT TINGGAL', '')
('Jett', 'L', '2023', '175', '50', 'TANGERANG', '')
('Felix', 'L', '2023', '155', '99', 'JKT', '')
('dewa', 'L', '2023', '160', '88', 'JKT', '')
('UCOK', 'L', '2023', '156', '54', 'BNDG', '')
('PUPUT', 'P', '2020', '167', '70', 'TANGSEL', '')
('MIYASA', 'P', '2021', '166', '20', 'TANGSEL', '')
('TUKIYEH', 'P', '2019', '170', '54', 'TANGSEL', '')
('MARBOT', 'L', '2022', '169', '20', 'BEKASI', '')
('JAKA', 'L', '2022', '189', '30', 'BEKASI', '')
('KINIL', 'L', '2021', '170', '33', 'TANGERANG', '')
('MAESAROH', 'P', '2021', '158', '15', 'TANGERANG', '')
('SARI', 'P', '2020', '160', '50', 'JKT BARAT', '')
('PURBA', 'L', '2023', '172', '89', 'JKT BARAT', '')
('JEKI', 'L', '2021', '160', '60', 'BANGKA', '')
('INAN', 'L', '2023', '178', '54', 'CISAIK', '')
('HERI', 'L', '2023', '176', '40', 'TANGSEL', '')
('JAWAN', 'L', '2019', '180', '60', 'TANGSEL', '')
('SRI', 'P', '2018', '161', '30', 'BNDG', '')
('DOENG', 'L', '2020', '190', '22', 'JAKBAR', '')
('MKIYA', 'P', '2018', '166', '50', 'BANGKA', '')
```

4. Lakukan filter data terhadap Kolom Gender, untuk melihat berapa baris data Pria/Wanita (Pilih salah 1).

```
In [37]: import pandas as pd
#mengkonvers hasil kueri ke dataframe pandas
df = pd.DataFrame(result, columns=[desc[0] for desc in cursor.description])

#filter data berdasarkan kolom "brick" yg bernilai 'no'
df_filtered = df[df['COL 2']=='L']

#menampilkan hasil filter
print("\nHasil Fokter:")
print(df_filtered)
```

Hasil Fokter:

	COL 1	COL 2	COL 3	COL 4	COL 5	COL 6	COL 7
1	jett	L	2023	175	50	TANGERANG	
2	felix	L	2023	155	99	JKT	
3	dewa	L	2023	160	88	JKT	
4	UCOK	L	2023	156	54	BNDG	
8	MARBOT	L	2022	169	20	BEKASI	
9	JAKA	L	2022	189	30	BEKASI	
10	KIWIL	L	2021	170	33	TANGERANG	
13	PURBA	L	2023	172	89	JKT BARAT	
14	JEKI	L	2021	160	60	BANGKA	
15	IWAN	L	2023	178	54	CISAIK	
16	HERI	L	2023	176	40	TANGSEL	
17	JAWAN	L	2019	180	60	TANGSEL	
19	DOENG	L	2020	190	22	JAKBAR	

```
In [ ]:
```

5. Lampirkan Screenshot

- ☞ Kode koneksi Jupyter ke Database
- ☞ Kode serta hasil filter di Jupyter

```
In [34]: import mysql.connector

# Membuat koneksi ke MySQL
connection = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    database="ps2[nadhira]"
)

# Membuat objek cursor untuk mengeksekusi kueri
cursor = connection.cursor()

try:
    # Mengeksekusi kueri SQL
    my_query = "SELECT * FROM data_jet;"
    cursor.execute(my_query)

    # Mengambil semua hasil kueri
    result = cursor.fetchall()

    # Menampilkan hasil kueri
    print("\nHasil Kueri:")
    for row in result:
        print(row)

finally:
    # Menutup kursor dan koneksi
    cursor.close()
    connection.close()
```

Hasil Kueri:

```
('NAMA', 'GENDER', 'ANGKATAN', 'TINGGI BADAN', 'WAKTU PERJALANAN', 'TEMPAT TINGGAL', '')
('jett', 'L', '2023', '175', '50', 'TANGERANG', '')
('felix', 'L', '2023', '155', '99', 'JKT', '')
('dewa', 'L', '2023', '160', '88', 'JKT', '')
('UCOK', 'L', '2023', '156', '54', 'BNDG', '')
('PUPUT', 'P', '2020', '167', '70', 'TANGSEL', '')
('MIYASA', 'P', '2021', '166', '20', 'TANGSEL', '')
('TUKIYEM', 'P', '2019', '170', '54', 'TANGSEL', '')
('MARBOT', 'L', '2022', '169', '20', 'BEKASI', '')
('JAKA', 'L', '2022', '189', '30', 'BEKASI', '')
('KIWIL', 'L', '2021', '170', '33', 'TANGERANG', '')
('MAESAROH', 'P', '2021', '158', '15', 'TANGERANG', '')
('SARI', 'P', '2020', '160', '50', 'JKT BARAT', '')
('PURBA', 'L', '2023', '172', '89', 'JKT BARAT', '')
('JEKI', 'L', '2021', '160', '60', 'BANGKA', '')
('IWAN', 'L', '2023', '178', '54', 'CISAIK', '')
('HERI', 'L', '2023', '176', '40', 'TANGSEL', '')
('JAWAN', 'L', '2019', '180', '60', 'TANGSEL', '')
('SRI', 'P', '2018', '161', '30', 'BNDG', '')
('DOENG', 'L', '2020', '190', '22', 'JAKBAR', '')
('MKIYA', 'P', '2018', '166', '50', 'BANGKA', '')
```

```
In [37]: import pandas as pd
#mengkonvers hasil kueri ke dataframe pandas
df = pd.DataFrame(result, columns=[desc[0] for desc in cursor.description])

#filter data berdasarkan kolom "brick" yg bernilai 'no'
df_filtered = df[df['COL 2']=='L']

#menampilkan hasil filter
print("\nHasil Fokter:")
print(df_filtered)
```

Hasil Fokter:

	COL 1	COL 2	COL 3	COL 4	COL 5	COL 6	COL 7
1	jett	L	2023	175	50	TANGERANG	
2	felix	L	2023	155	99	JKT	
3	dewa	L	2023	160	88	JKT	
4	UCOK	L	2023	156	54	BNDG	
8	MARBOT	L	2022	169	20	BEKASI	
9	JAKA	L	2022	189	30	BEKASI	
10	KIWIL	L	2021	170	33	TANGERANG	
13	PURBA	L	2023	172	89	JKT BARAT	
14	JEKI	L	2021	160	60	BANGKA	
15	IWAN	L	2023	178	54	CISAIK	
16	HERI	L	2023	176	40	TANGSEL	
17	JAWAN	L	2019	180	60	TANGSEL	
19	DOENG	L	2020	190	22	JAKBAR	

In []:

4. File Praktikum

Github Repository:

https://github.com/NadhiraAninditaRalena/prak2_probstat.git

5. Kesimpulan

- a. Dalam pengerjaan praktikum Statistika, ...penyaringan data sangat penting dalam analisis statistika, Dengan menggunakan filter data yang tepat, kita dapat menghasilkan informasi yang relevan dan meningkatkan kualitas analisis statistika yang dilakukan.

- b. Kita juga dapat mengetahui...Koneksi ke database memungkinkan pengambilan data langsung dari sumbernya, secara cepat dan akurat. Pemahaman tentang cara terhubung ke database, serta kemampuan untuk mengelola data dari sumber yang besar dan beragam, menjadi keterampilan kunci dalam analisis statistika modern.

6. Cek List (✓)

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	✓	
2.	Latihan Kedua	✓	
3.	Latihan Ketiga	✓	
4.	Latihan Keempat	✓	
5.	Latihan Kelima	✓	
6.	Latihan Keenam	✓	

7. Formulir Umpan Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	...10 Menit	menarik
2.	Latihan Kedua	...10 Menit	menarik
3.	Latihan Ketiga	... 10 Menit	menarik

4.	Latihan Keempat	... 15 Menit	menarik
5.	Latihan Kelima	...15 Menit	menarik
6.	Latihan Keenam	...15 Menit	menarik

Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang