Dokumentasi Simple ETL Dengan API ke PostgreSQL

1. Ambil data dengan API

etl-api.py - C:\Users\diana\OneDrive\Documents\File Script Postgres\DE\etl-api.py (3.9.7)

File Edit Format Run Options Window Help

```
import requests
import pandas as pd
import psycopg2

# 1. Extract: Ambil data dari API
response = requests.get("https://jsonplaceholder.typicode.com/users")
data = response.json()

# Convert data dari JSON ke pandas DataFrame
df = pd.DataFrame(data)

# Simpan data mentah ke file CSV sebagai "backup" atau bagian dari data lake
df.to_csv('data_lake/users_raw.csv', index=False)
print("Data mentah berhasil diambil dari API dan disimpan ke data_lake/users_raw

# 2. Transform: Membersihkan dan memilih data
# Misalnya, kita hanya mengambil kolom name, email, dan city dari alamat penggun
df_transformed = df[['name', 'email']]
df_transformed['city'] = df['address'].apply(lambda x: x['city'])

# Simpan data yang sudah ditransformasi
df_transformed.to_csv('data_lake/users_clean.csv', index=False)
print("Data berhasil ditransformasi dan disimpan di data_lake/users_clean.csv")

# 3. Load: Memuat data yang sudah bersih ke PostgreSQL
try:
    conn = psycopg2.connect(
        database="guest",
        user="postgres",
        password="postgres"
    )

    cur = conn.cursor()

    # Buat tabel jika belum ada
    cur.execute('''
    CREATE TABLE IF NOT EXISTS users (
        name VARCHAR(100),
        email VARCHAR(100),
        city VARCHAR(100)
```

Ln: 21 Col: 0

IDLE Shell 3.9.7

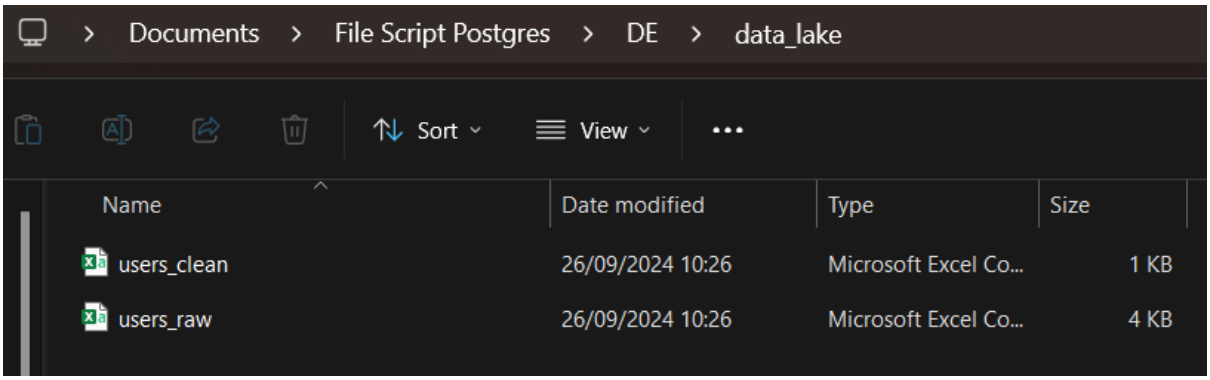
File Edit Shell Debug Options Window Help

```
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)] on win
32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\diana\OneDrive\Documents\File Script Postgres\DE\etl-api.py
Data mentah berhasil diambil dari API dan disimpan ke data_lake/users_raw.csv

Warning (from warnings module):
  File "C:\Users\diana\OneDrive\Documents\File Script Postgres\DE\etl-api.py", l
ine 19
    df_transformed['city'] = df['address'].apply(lambda x: x['city'])
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stab
le/user_guide/indexing.html#returning-a-view-versus-a-copy
Data berhasil ditransformasi dan disimpan di data_lake/users_clean.csv
Data berhasil dimuat ke PostgreSQL
>>>
```

2. Backup data mentah



3. Transform data dan Load data ke PostgreSQL