

Tugas Project BI 2

- Install Library

Lakukan install library yang dibutuhkan, berikut command install library pada python 3 :

pip install psycpg2

pip install pandas

pip install numpy

pip install openpyxl

- Connection Database

➤ Create Folder config untuk meletakkan file db.conf dengan isi sebagai berikut :

host = localhost

user = postgres

password = Admin1234

database = db_transaction

port = 5432

Data diatas akan di panggil oleh script berikut :

```
def db_connect():  
    config = db_config()  
  
    host = config['host'].strip()  
    db = config['database'].strip()  
    port = config['port'].strip()  
    user = config['user'].strip()  
    pwd = config['password'].strip()
```

➤ Create file connection.py, pada bagian atas terdapat command sebagai berikut :
Import os #untuk call library yang berfungsi untuk berinteraksi dengan operational system
Import psycpg2 #untuk call library yang berfungsi sebagai adaptor postgresql

```
def db_file_config():  
    config_file = "\\config\\db.conf"  
    return config_file  
#fungsi untuk memanggil file konfigurasi database yang di definisikan sebelumnya
```

```
def db_config():  
    config = {}  
    fileconf = db_file_config()  
    rfile = open(os.getcwd() + fileconf, "r")
```

#open() untuk membaca file, **os.getcwd()** untuk memanggil path root this project ,
variable rfile untuk membaca file **db.conf** .

```

for line in rfile:
    line = line.replace("\n", "")
    length = len(line)

    if (length != 0):
        line = line.replace(" ", "")
        (name, value) = line.split("=")
        config[name] = value
return config

```

#fungsi ini untuk mengembalikan value variable **db.conf** tadi ke dalam variable array config

```

def db_connect():
    config = db_config()

    host = config['host'].strip()
    db = config['database'].strip()
    port = config['port'].strip()
    user = config['user'].strip()
    pwd = config['password'].strip()

    while True:
        try:
            conn = psycopg2.connect(host=host, database=db, user=user, password=pwd,
port=port)
        except :
            print("Oops! Cannot Connect To PostgreSQL, Try again...")
        else:
            break
    return conn

```

#**psycopg2.connect** berfungsi untuk menghubungkan database yang sudah di define pada file **db.conf**, Jika fungsi ini tidak menemukan file connection yang benar maka akan me looping sampai koneksi di temukan, dan akan menampilkan **“Oops! Cannot Connect To PostgreSQL, Try again...”**

- Main Transformation Data

➤ Buat File app.py file ini yang akan digunakan sebagai main script, kemudian isi file tersebut dengan script sebagai berikut :

```

Import os #untuk call library yang berfungsi untuk berinteraksi dengan operational system
import connection #untuk memanggil file connection.py

```

```

import pandas as pd #untuk memanggil library pandas yang digunakan untuk data frame
import numpy as np #berfungsi untuk proses komputasi numerik

```

```

if __name__ == "__main__":
    path = os.getcwd()      #define path root project
    path_query = path + '/sql/'

    #file_query
    file_query = 'dump_query.sql'

    #connection
    conn = connection.db_connect() #menghubungkan ke database
    cur = conn.cursor()
    #define cursor yang digunakan untuk mengeksekusi query

#read_data
    with open(path_query + file_query,'r') as file:
        query = file.read()

    cur.execute(query)
    data = cur.fetchall()
    df = pd.DataFrame(data, columns=['order','costumer','city','date'])

#script diatas berfungsi untuk membaca file dump_query.sql , file tersebut berisi syntax
sql, untuk cur.execute(query) untuk mengeksekusi perintah sql tersebut, kemudian result
nya disimpan ke variable data.
df = pd.DataFrame(data, columns=['order','costumer','city','date'])
#untuk menampilkan data query kedalam bentuk data frame

#transformation
    df['date'] = pd.to_datetime(df['date']) #convert to datetime
    df = df[df['date'].dt.year == 2016] #filter data hanya tahun 2016
    df['date'] = df['date'].dt.strftime('%Y-%m-%d') #reformat kolom date menjadi
yyymmdd

    df \
        .groupby(['city','date']) \
        .agg({'order':'count'}) \
        .unstack() \
        .to_excel('report_order.xlsx')
#transformasi groupby dengan fungsi agregat count pada kolom order, fungsi unstack()
digunakan untuk conver row to column, to_excel('report_order.xlsx') untuk menyimpan
hasil transformasi ke dalam bentuk xlsx.

```

```
df \
    .groupby(['city','date']) \
    .agg({'costumer':'count'}) \
    .unstack() \
    .to_excel('report_order.xlsx')
```

#transformasi **groupby** dengan fungsi agregat count pada kolom costumer, fungsi **unstack()** digunakan untuk conver row to column, **to_excel('report_order.xlsx')** untuk menyimpan hasil transformasi ke dalam bentuk xlsx.

- Running Command

Berikut command yang digunakan untuk menjalankan aplikasi :
Python3 app.py