Dml_query_1.sql

```
select order_id,
        b.customer_id
        customer_city,
from tb_order a
        left join tb_customer b on a.customer_id = b.customer_id
where order_status not like '%canceled%'
```

tahap untuk mengkoneksikan

```

import connection

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

    file_query = 'dml_query_1.sql'

    conn = connection.db_connect()
    with open(path_query + file_query, 'r') as file:
        query = file.read()

    print(query)
```

```
import os
import connection

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

    file_query = 'dml_query_1.sql'

    conn = connection.db_connect()
    cur = conn.cursor()

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

    cur.execute(query)
    data = cur.fetchall()
    print(data)
```

```
import os
import connection

if __name__ == "__main__":

    #list path
    path = os.getcwd()
    path_query = path + '/sql/'

    #list filename
    file_query = 'dml_query_1.sql'

    #connection
    conn = connection.db_connect()
    cur = conn.cursor()


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

    cur.execute(query)
    data = cur.fetchall()
    print(data)
```

proses data injection/proses pengambilan data

```
    cur.execute(query)
    data = cur.fetchall()

    df = pd.DataFrame(data, columns= ['order','customer','city','date'])
    print(df.head(1))
```

proses transformation

```
    #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','customer','city','date'])
    print(df.head(1))
```

```
#transformation
df['date'] = pd.to_datetime(df['date'])
df = df[['date'].dt.year == 2016]
df['data'] = pd.to_datetime(df['date']).dt.strftime('%Y-%m-%d')

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

 df = df\
    .groupby(['city,'date']) \
    .agg({'customer':'nunique'}) \
    .unstack() \
    .to_excel('report_order.xlsx,)

df = df.replace({np.nan:none})
print(df)
```