

LAPORAN PERTEMUAN 9
BIG DATA



Oleh :

MOCHAMMAD ZAKARO AL FAJRI 2241720175

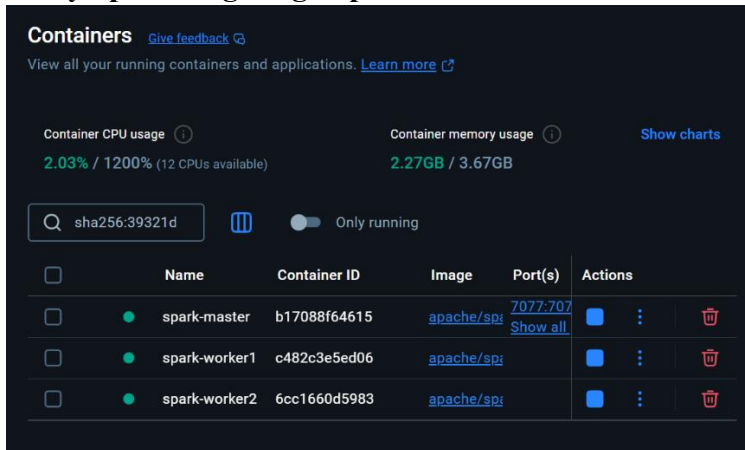
D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2025

Tugas 8 – Spark SQL, DataSources, DataFrame, dan Dataset APIs

Praktikum : Interaksi dengan Spark SQL di Lingkungan Windows Menggunakan Docker

Hasil :

1. Menyiapkan lingkungan praktikum



2. Praktikum: Membangun ETL Pipeline

Keterangan :

- Extract: Baca data dari file CSV (sales_data.csv).
- Transform:
 - Filter transaksi dengan Revenue > \$100.
 - Hitung total penjualan per kategori.
- Load: Simpan hasil ke Parquet.

Pengerjaan :

- Kode :

```
etlpipe.ipynb
[13]: from pyspark.sql import SparkSession
      from pyspark.sql.functions import col, sum

      spark = SparkSession.builder.appName("ETLPipeline").getOrCreate()

      # Extract
      df = spark.read.csv("sales_data.csv", header=True, inferSchema=True)

      # Transform
      df_filtered = df.filter(col("Revenue") > 100)
      df_result = df_filtered.groupBy("Product_Category").agg(sum("Revenue").alias("total_sales"))

      # Load
      df_result.write.mode("overwrite").parquet("output_sales.parquet")

      df_result.show()
      spark.stop()
```

- Hasil :

```
+-----+-----+
|Product_Category|total_sales|
+-----+-----+
|      Clothing|    8198902|
|   Accessories|    13559164|
|         Bikes|    61782134|
+-----+-----+
```

3. Analisis Data Retail

Keterangan :

- Format : CSV (sales_data.csv)
- Tugas :
 - Hitung total pendapatan per bulan.
 - Identifikasi 5 produk terlaris.
 - Simpan hasil dalam format Parquet.
- Solusi : Pendapatan perbulan

Pengerjaan :

- **Pendapatan perbulan :**

Kode :

```
etlpipline.ipynb  analisis_data_retail.ipynb  X  +

[14]: from pyspark.sql import SparkSession
      from pyspark.sql.functions import month, sum, count

      spark = SparkSession.builder.appName("ETLPipeLine").getOrCreate()

      df = spark.read.csv("sales_data.csv", header=True, inferSchema=True)

      df_revenue = df.withColumn("month", month("Date")) \
                      .groupBy("month") \
                      .agg(sum(df["Unit_Price"]*df["Order_Quantity"]).alias("total_revenue"))
      df_revenue.show()
```

Hasil :

month	total_revenue
12	10158080
1	7832338
6	10085537
3	8201790
5	9859851
9	6517880
4	8485163
8	6348349
7	6392045
10	6709394
11	6977157
2	7608734

- **Identifikasi 5 Produk terlaris**

Kode :

```
etlpipline.ipynb  analisis_data_retail.ipynb  X  +

[17]: # Identifikasi 5 Produk terLaris

      # 5 produk terLaris

      df_top_products = df.groupBy("Product") \
                          .agg(count("*").alias("total_orders")) \
                          .orderBy("total_orders", ascending=False) \
                          .limit(5)

      df_top_products.show()
```

Hasil :

Product	total_orders
Water Bottle - 30...	10794
Patch Kit/8 Patches	10416
Mountain Tire Tube	6816
AWC Logo Cap	4358
Sport-100 Helmet,...	4220

- Menyimpan dalam format parquet

Kode dan hasil :

The screenshot shows a Jupyter Notebook interface. On the left is a file explorer with a search bar and a list of files and folders. The 'revenue_by...' folder is selected. On the right is the code editor for 'etlpipeline.ipynb'. The code cell [18] contains the following Python code:

```
[18]: # simpan dalam format parquet

df_revenue.write.parquet("revenue_by_month.parquet")
df_top_products.write.parquet("top_products.parquet")
```

4. Evaluasi

Soal latihan

1) Baca data dari table di database MySQL anda menggunakan Spark.

Jawab :

a. Saya menggunakan database **big_data** sebagai contoh

The screenshot shows a database management tool interface. On the left is a tree view of the database structure, with 'big_data' selected. The main area displays the query result for the query 'SELECT * FROM `mahasiswa`'. The result shows one row with the following data:

nama	absen	nim
Mochammad Zakaro Al Fajri	10	2241720175

b. Melakukan konfigurasi

- Penambahan file mysql-connector-j-8.0.33.jar Apache agar bisa berkomunikasi, membaca atau menulis data dengan MySQL.

```
[11]: !mkdir -p install
      !dpkg-deb -x mysql-connector-8.0.33.deb install/
```

- Melakukan perubahan pada kode yang diberikan agar bisa mengakses database big_data dan tabel mahasiswa yang saya gunakan

```
[21]: from pyspark.sql import SparkSession

# Membuat SparkSession dengan driver MySQL
spark = SparkSession.builder \
    .appName("Read MySQL Table") \
    .config("spark.jars", "install/usr/share/java/mysql-connector-j-8.0.33.jar") \
    .getOrCreate()

# Baca tabel dari database MySQL
df = spark.read.format("jdbc") \
    .option("url", "jdbc:mysql://host.docker.internal:3306/big_data") \
    .option("dbtable", "mahasiswa") \
    .option("user", "root") \
    .option("password", "") \
    .option("driver", "com.mysql.cj.jdbc.Driver") \
    .load()

# Menampilkan 5 baris pertama untuk cek
df.show(5)
```

c. Output :

Menampilkan isi dari tabel mahasiswa

```
+-----+-----+-----+
|          nama|absen|          nim|
+-----+-----+-----+
|Mochammad Zakaro ...| 10|2241720175|
+-----+-----+-----+
```