

Big Data

Spark SQL (Jobsheet 9)

Dosen :

M. Hasyim Ratsanjani, S.Kom., M.Kom.



Dibuat Oleh :

Nurhaliza Anindya Putri

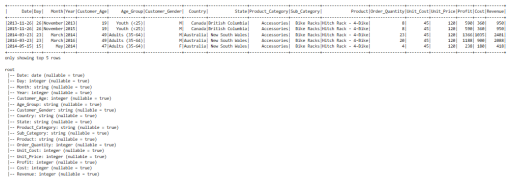
2241720016

Kelas : TI – 3D

POLITEKNIK NEGERI MALANG

2025

## 1. Implementasi Python Spark (di Dockers)

No	Gambar	Deskripsi
	<pre> from pyspark.sql import SparkSession  # Buat SparkSession spark = SparkSession.builder.appName("Ex tractCSV").getOrCreate()  # Extract - Membaca file CSV df = spark.read.csv("sales_data.csv", header=True, inferSchema=True)  # Tampilkan beberapa baris pertama df.show(5)  # Cek struktur data df.printSchema() </pre>	<p>Extract: Baca data dari file CSV (sales_data.csv).</p>
		<p>Hasil ss dari ekstrak sales_data.csv</p>
	<pre> from pyspark.sql.functions import col  # Filter transaksi dengan Revenue &gt; 100 df_filtered = df.filter(col("Revenue") &gt; 100)  # Tampilkan hasil filter df_filtered.show(5) </pre>	<p>Filter transaksi dengan Revenue &gt; \$100.</p>



	<pre> month(col("Date"))  # Hitung total pendapatan per bulan df_monthly_revenue = df_with_month.groupBy("Month") \  .agg(sum("Revenue").alias("total_r evenue")) \ .orderBy("Month")  # Tampilkan hasil df_monthly_revenue.show() </pre>	
	<pre> +-----+  Month total_revenue  +-----+   1      7005895     2      6834583     3      7347164     4      7602750     5      8836763     6      9043008     7      5721459     8      5711193     9      5841885    10      5995079    11      6244298    12      9086931   +-----+ </pre>	Hasil ss Hitung total pendapatan per bulan.
	<pre> from pyspark.sql.functions import desc, sum  # Kelompokkan berdasarkan kolom Product dan hitung total Revenue df_top_products = df.groupBy("Product") \  .agg(sum("Revenue").alias("total_r evenue")) \ .orderBy(desc("total_revenue")) \ .limit(5)  # Tampilkan hasil df_top_products.show() </pre>	Identifikasi 5 produk terlaris.

	<pre> +-----+-----+        Product       total_revenue  +-----+-----+   Road-150 Red, 62      3829416     Mountain-200 Blac...      3366248     Road-150 Red, 52      3188848     Road-150 Red, 56      3158885     Mountain-200 Silv...      3081878   +-----+-----+ </pre>	Hasil ss Identifikasi 5 produk terlaris.
	<pre> # Simpan hasil ke Parquet df_top_products.write.mode("over write").parquet("top5_products.par quet") </pre>	Simpan hasil dalam format Parquet.
	<pre> from pyspark.sql.functions import month, sum  # Ambil bulan dari kolom Date df_with_month = df.withColumn("Month", month("Date"))  # Kelompokkan berdasarkan bulan dan hitung total pendapatan df_monthly_revenue = df_with_month.groupBy("Month") \  .agg(sum("Revenue").alias("total_r evenue"))  # Tampilkan hasil df_monthly_revenue.show()  # Simpan hasil ke Parquet df_monthly_revenue.write.mode("o verwrite").parquet("monthly_reven ue.parquet") </pre>	Pendapatan perbulan

	<pre> +-----+  Month total_revenue  +-----+   12     9086931     1     7085895     6     9043008     3     7347164     5     8836763     9     5841885     4     7602750     8     5711193     7     5721459    10     5995079    11     6244298     2     6834583  +-----+ </pre>	Hasil ss Pendapatan perbulan
	<pre> from pyspark.sql.functions import sum, desc  # Kelompokkan berdasarkan produk dan hitung total pendapatan df_top_products = df.groupBy("Product") \  .agg(sum("Revenue").alias("total_r evenue")) \ .orderBy(desc("total_revenue")) \ .limit(5)  # Tampilkan hasil df_top_products.show() </pre>	Identifikasi 5 Produk terlaris
	<pre> +-----+  Product total_revenue  +-----+  Road-150 Red, 62     3829416   Mountain-200 Blac...     3366248   Road-150 Red, 52     3180040   Road-150 Red, 56     3158805   Mountain-200 Silv...     3081078  +-----+ </pre>	Hasil ss Identifikasi 5 Produk terlaris
	<pre> # Simpan hasil ke Parquet df_top_products.write.mode("over write").parquet("top5_products.par quet") </pre>	simpan dalam format parquet

[illegible]