

# **BUSINESS INTELLIGENCE**

## **LAPORAN PROYEK AKHIR**



**Oleh :**

<b>ACHMAD DIAZ HIKMAL BAIHAQI</b>	<b>(2341720228/03/3B)</b>
<b>AHMAD NAUFAL WASKITO AJI</b>	<b>(2341720080/05/3B)</b>
<b>IRSA CAHAYA WIDODO</b>	<b>(2341720193 /13/3B)</b>
<b>RAMADHANI BI HAYYIN</b>	<b>(2341720226/19/3B)</b>
<b>RANGGA PUTRA SYANANDA BUDHI</b>	<b>(2341720079/20/3B)</b>

**PROGRAM STUDI D-IV TEKNIK INFORMATIKA**  
**JURUSAN TEKNOLOGI INFORMASI POLITEKNIK**  
**NEGERI MALANG**

**DESEMBER 2025**

# LAPORAN PROYEK AKHIR: IMPLEMENTASI ETL DAN DATA WAREHOUSE

## Studi Kasus: Northwind Traders

### BAB 1

### PENDAHULUAN

#### 1.1 Latar Belakang

Dalam era digital, pengelolaan data yang efisien sangat krusial bagi keberlangsungan bisnis. Studi kasus ini menggunakan dataset **Northwind Traders**, sebuah perusahaan fiktif yang memiliki data relasional mencakup proses bisnis rantai pasok (Supply Chain), penjualan (Sales), dan sumber daya manusia (SDM).

Pemilihan Northwind didasarkan pada kompleksitas yang pas untuk simulasi Data Warehouse, di mana dataset ini memiliki relasi *many-to-many* namun jumlah tabel masih di bawah 20, sehingga ideal untuk diselesaikan dalam durasi penggerjaan proyek.

#### 1.2 Tujuan

Tujuan dari proyek ini adalah:

1. Mengidentifikasi sumber data publik yang merepresentasikan proses bisnis nyata.
2. Merancang dan mengimplementasikan Data Warehouse menggunakan konsep Dimensional Modeling (Star Schema).
3. Membangun pipeline ETL (Extract, Transform, Load) menggunakan Pentaho Data Integration.
4. Memvisualisasikan hasil data warehouse ke dalam dashboard interaktif untuk analisis KPI.

## BAB 2:

# PERANCANGAN SKEMA (SCHEMA DESIGN)

### 2.1 Sumber Data (OLTP)

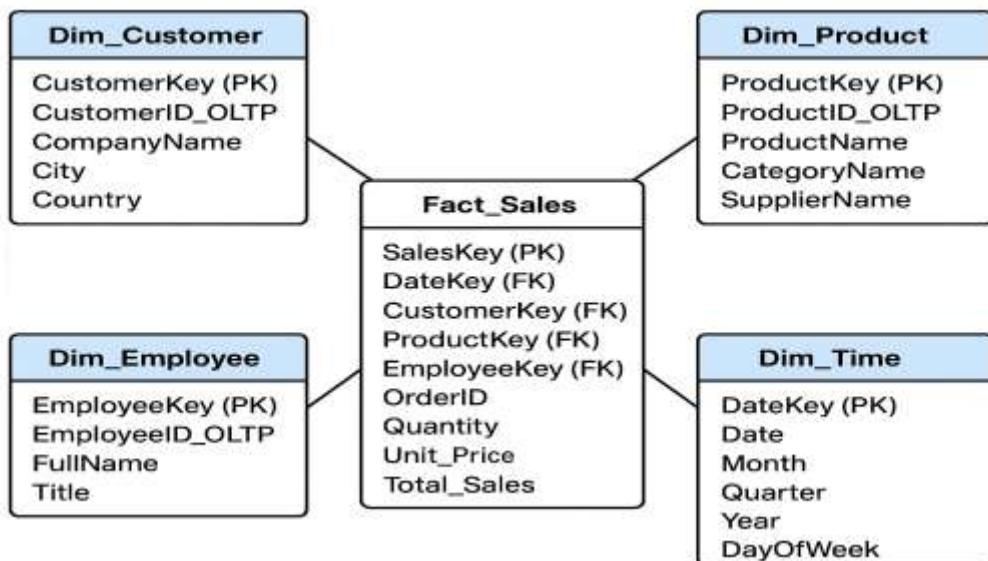
Data sumber berasal dari database operasional (OLTP) Northwind yang berjalan di MySQL. Dataset ini mencakup tabel utama seperti employees, orders, order\_details, products, dan customers yang telah divalidasi memiliki komponen 4W (Who, What, Where, When) + 1H (How Much).

### 2.2 Desain Dimensional Model (OLAP)

Kami mengubah model ERD relasional menjadi **Star Schema** untuk keperluan analisis (OLAP) pada database tujuan northwind\_dwh.

#### Struktur Tabel Fakta dan Dimensi:

- **Fact\_Sales (Tabel Fakta):** Menyimpan metrik kinerja penjualan.
  - Primary Key: SalesKey (Auto Increment).
  - Foreign Keys: DateKey, CustomerKey, ProductKey, EmployeeKey.
  - Measures: Quantity, Unit\_Price, Total\_Sales (Quantity \* Unit\_Price) .
- **Dim\_Customer (Tabel Dimensi):** Menyimpan atribut pelanggan seperti CompanyName, City, dan Country. Kunci utamanya adalah CustomerKey .
- **Dim\_Product (Tabel Dimensi):** Menyimpan detail produk hasil denormalisasi dari tabel products dan categories. Atribut meliputi ProductName, CategoryName, dan SupplierName .
- **Dim\_Employee (Tabel Dimensi):** Menyimpan data karyawan seperti FullName dan Title .
- **Dim\_Time (Tabel Dimensi):** Menyimpan atribut waktu seperti Date, Month, Quarter, dan Year .



# BAB 3:

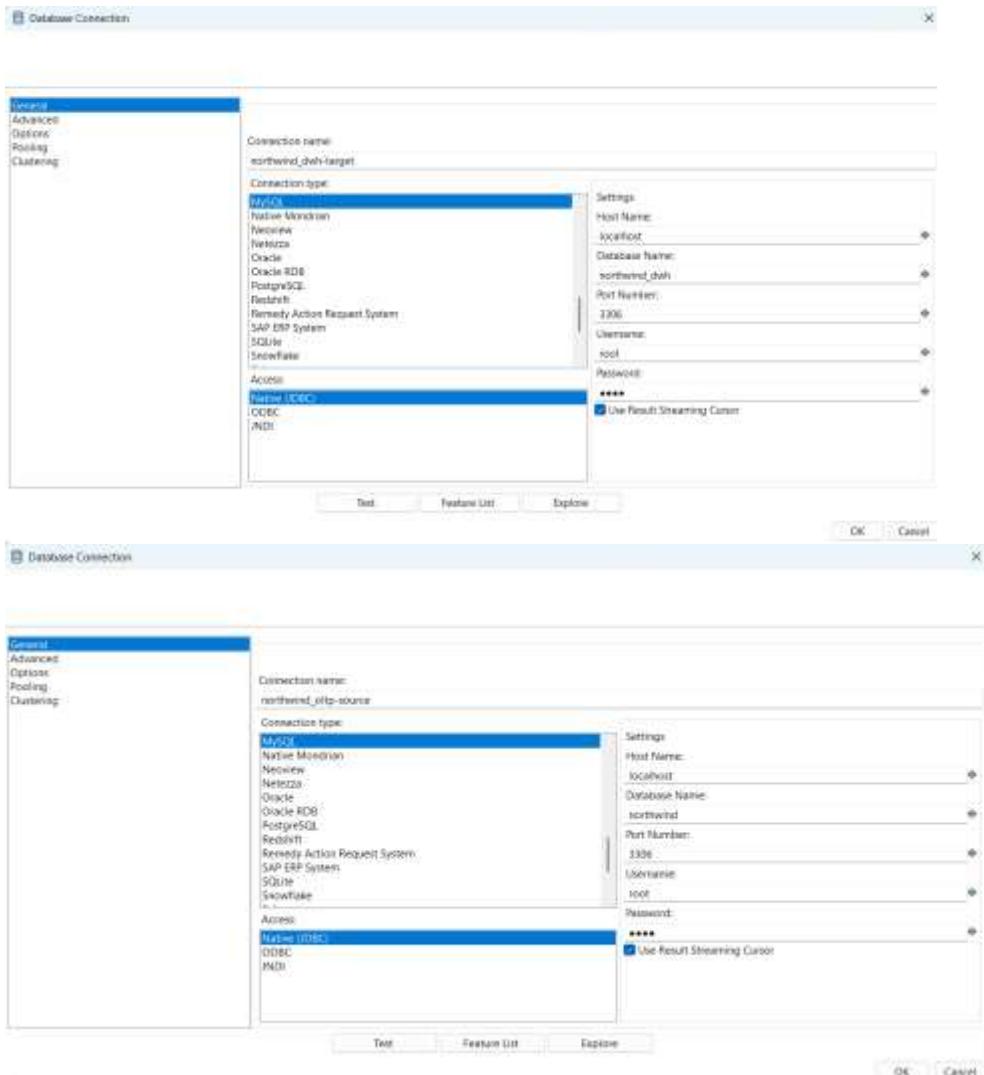
## IMPLEMENTASI ETL (EXTRACT, TRANSFORM, LOAD)

Proses ETL dilakukan menggunakan **Pentaho Data Integration (Spoon)** untuk memindahkan data dari northwind\_oltp ke northwind\_dwh.

### 3.1 Koneksi Database

Langkah pertama adalah membuat koneksi database di Pentaho:

- **Source:** northwind\_oltp-source (MySQL localhost).
- **Target:** northwind\_dwh-target (MySQL localhost) .



### 3.2 ETL Dimensi Customer (Dim\_Customer)

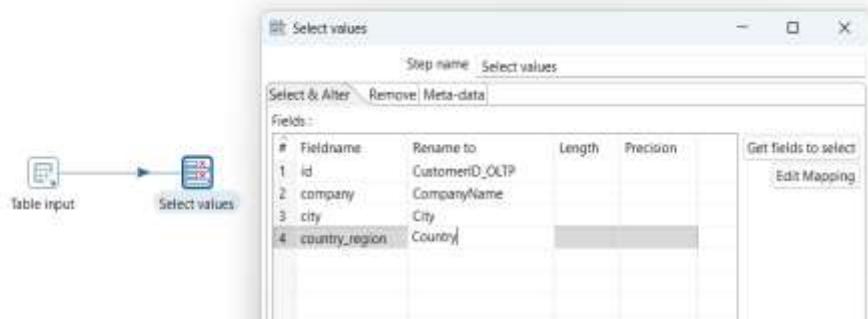
Proses pengisian tabel dimensi pelanggan melibatkan langkah berikut:

1. **Extract (Table Input):** Mengambil data id, company, city, country\_region dari tabel customers.
2. **Transform (Select Values):** Mengubah nama kolom agar sesuai dengan target, misal id menjadi CustomerID\_OLTP dan country\_region menjadi Country.
3. **Load (Dimension Lookup/Table Output):** Memasukkan data ke tabel dim\_customer. CustomerKey dibuat secara otomatis sebagai *Surrogate Key*.

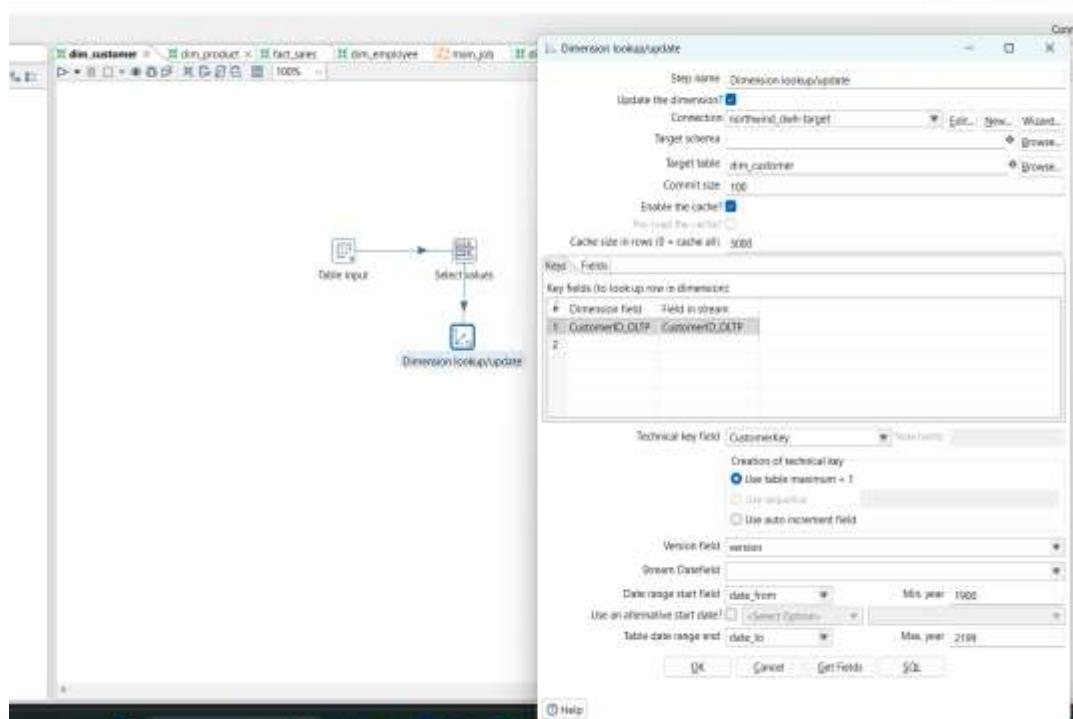
#### LANGKAH - LANGKAH

1. add table input
2. masukkan query berikut untuk mengambil beberapa field yang dibutuhkan

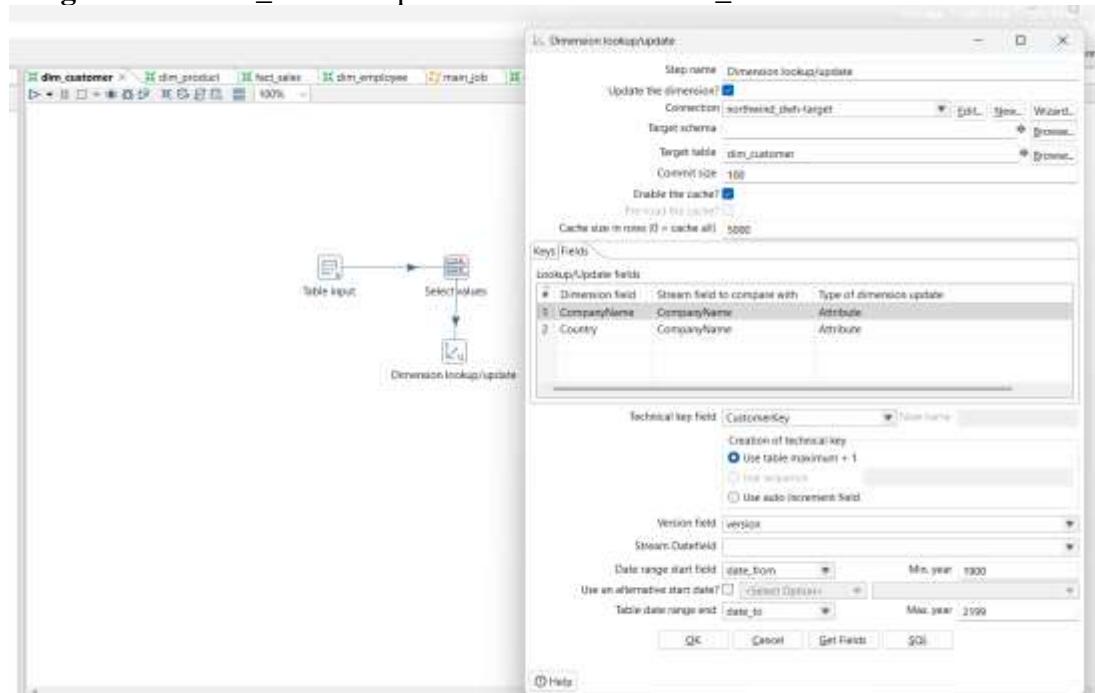




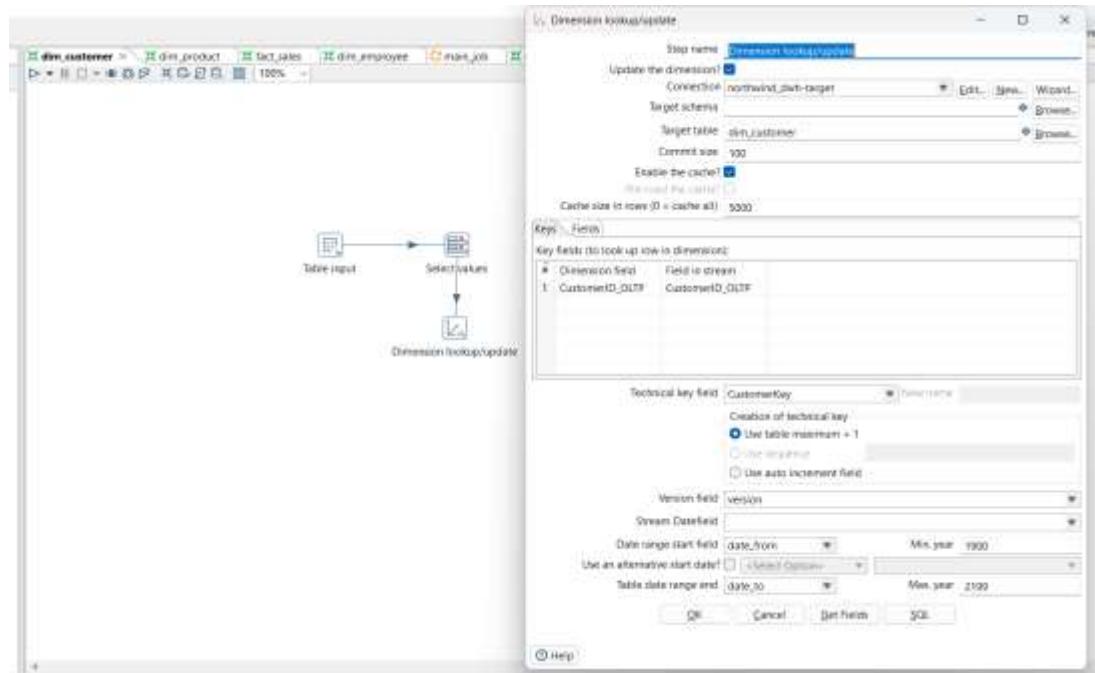
- 3.
4. tambahkan select values dan ubah atau rename menjadi nama sama dengan table target agar lebih mudah mengambilnya nanti
5. **Memuat Data ke Dimensi (Load)** Komponen **Dimension Lookup/Update** digunakan untuk memuat data bersih ke dalam tabel dimensi di Data Warehouse. Metode ini dipilih karena mendukung pembuatan *Technical Key* secara otomatis.



6. **Target Table:** dim\_customer pada database northwind\_dwh.



7. **Konfigurasi Keys (Kunci Pencarian):** Field CustomerID\_OLTP pada stream dibandingkan dengan field CustomerID\_OLTP pada tabel dimensi untuk memastikan data pelanggan dicocokkan berdasarkan ID unik aslinya.
8. **Konfigurasi Technical Key:** Field CustomerKey diatur sebagai *primary key* baru dengan metode *Table Maximum + 1* (Auto Increment). Ini berfungsi sebagai *Surrogate Key* untuk analisis OLAP.



9. **Update Fields:** Kolom CompanyName dan Country diatur tipe pembaruannya (*Type of dimension update*) sebagai "Attribute", yang berarti data akan diperbarui jika terjadi perubahan pada sumber.

The screenshot shows the Apache NiFi Spoon interface with two separate transformations for the dim\_customer dimension.

**Transformation 1 (Top):**

- Table Input:** A standard table input node.
- Select Values:** A select values node connected to the Table Input.
- Table Output:** A table output node connected to the Select Values node.
- Table Output Properties (Step name: Table output):**
  - Connection: northwind\_dwh-target
  - Target schema: dim\_customer
  - Commit size: 1000
  - Truncate table: checked
  - Main options: Database fields
  - Partition data over tables: unchecked
  - Partitions per partition: 1
  - Partitions per day: 1
  - Use batch update for inserts: checked
  - Is the same as the table defined in a field?: unchecked
  - Find the column name of table: dropdown menu
  - Store the table name field: dropdown menu
  - Return auto-generated key: checked
  - Name of auto-generated key field: dropdown menu

**Execution Results:**

```

2025-12-10 14:39:02,259 - Spoon - Transformation opened.
2025-12-10 14:39:02,259 - Spoon - Launching transformation [dim_customer]...
2025-12-10 14:39:02,259 - Spoon - Started the transformation execution.
2025-12-10 14:39:02,314 - dim_customer - Dispatching started for transformation [dim_customer].
2025-12-10 14:39:02,360 - Table input[0] - Finished reading query, closing connection
2025-12-10 14:39:02,367 - Table input[0] - Finished processing (0.29, O=0, R=0, W=29, U=0, E=0)
2025-12-10 14:39:02,367 - Dimension lookup/update[0] - ER904 (version 10.2.0.0-222, build 0.0 from
2025-12-10 14:39:02,368 - Dimension lookup/update[0] - ER904 (version 10.2.0.0-222, build 0.0 from
2025-12-10 14:39:02,368 - Dimension lookup/update[0] - Error inserting unknown row in dimension
2025-12-10 14:39:02,368 - Dimension lookup/update[0] - Dimension lookup/update[0] - Couldn't execute SQL: insert into dim_cus...
2025-12-10 14:39:02,368 - Dimension lookup/update[0] - Dimension lookup/update[0] - Couldn't execute SQL: insert into dim_cus...

```

**Transformation 2 (Bottom):**

- Table Input:** A standard table input node.
- Select Values:** A select values node connected to the Table Input.
- Table Output:** A table output node connected to the Select Values node.
- Table Output Properties (Step name: Table output):**
  - Connection: northwind\_dwh-target
  - Target schema: dim\_customer
  - Commit size: 1000
  - Truncate table: checked
  - Main options: Database fields
  - Fields to insert:
    - Table field: CustomerID\_OLEP Stream field: CustomerID\_OLEP
    - Table field: CompanyName Stream field: CompanyName
    - Table field: City Stream field: City
    - Table field: Country Stream field: Country

**Execution Results:**

```

2025-12-10 14:39:02,259 - Spoon - Transformation opened.
2025-12-10 14:39:02,259 - Spoon - Launching transformation [dim_customer]...
2025-12-10 14:39:02,259 - Spoon - Started the transformation execution.
2025-12-10 14:39:02,314 - dim_customer - Dispatching started for transformation [dim_customer].
2025-12-10 14:39:02,360 - Table input[0] - Finished reading query, closing connection
2025-12-10 14:39:02,367 - Table input[0] - Finished processing (0.29, O=0, R=0, W=29, U=0, E=0)
2025-12-10 14:39:02,367 - Dimension lookup/update[0] - ER904 (version 10.2.0.0-222, build 0.0 from
2025-12-10 14:39:02,368 - Dimension lookup/update[0] - ER904 (version 10.2.0.0-222, build 0.0 from
2025-12-10 14:39:02,368 - Dimension lookup/update[0] - Error inserting unknown row in dimension
2025-12-10 14:39:02,368 - Dimension lookup/update[0] - Dimension lookup/update[0] - Couldn't execute SQL: insert into dim_cus...
2025-12-10 14:39:02,368 - Dimension lookup/update[0] - Dimension lookup/update[0] - Couldn't execute SQL: insert into dim_cus...

```

## 11. HASIL

The screenshot shows a data processing interface with the following components:

- Table input:** Represented by a green icon with a table symbol.
- Select values:** Represented by a green icon with a list symbol.
- Table output:** Represented by a blue icon with a table symbol.

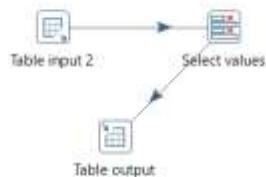
The flow starts with "Table input" pointing to "Select values", which then points to "Table output".

**Execution Results:**

#	CustomerID_OLTP	CompanyName	City	Country
1	1	Company A	Seattle	USA
2	2	Company B	Boston	USA
3	3	Company C	Los Angeles	USA
4	4	Company D	New York	USA
5	5	Company E	Minneapolis	USA
6	6	Company F	Milwaukee	USA
7	7	Company G	Baltimore	USA
8	8	Company H	Portland	USA
9	9	Company I	Salt Lake City	USA

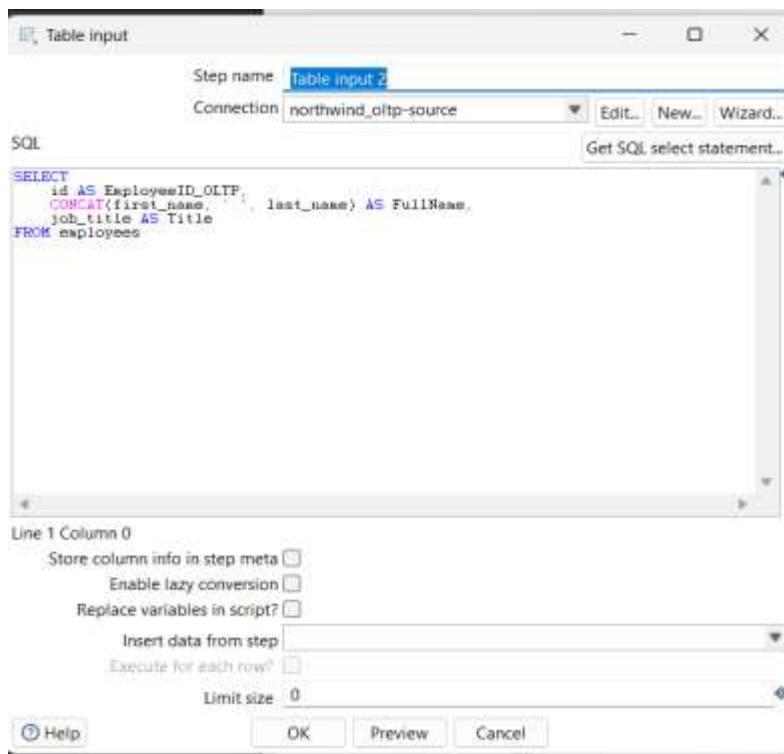
### 3.3 ETL Dimensi Karyawan (Dim\_Employee)

1. Proses ETL ini bertujuan untuk memindahkan data sumber daya manusia (SDM) dari tabel employees ke tabel dimensi dim\_employee. Fokus utama pada tahap ini adalah menggabungkan nama depan dan belakang karyawan menjadi satu kolom nama lengkap (*Full Name*) untuk memudahkan pelaporan.
2. **Alur Transformasi:** Proses ini menggunakan alur sederhana yang terdiri dari ekstraksi data dengan kustomisasi SQL, penyesuaian metadata, dan pemuatan ke tabel target.

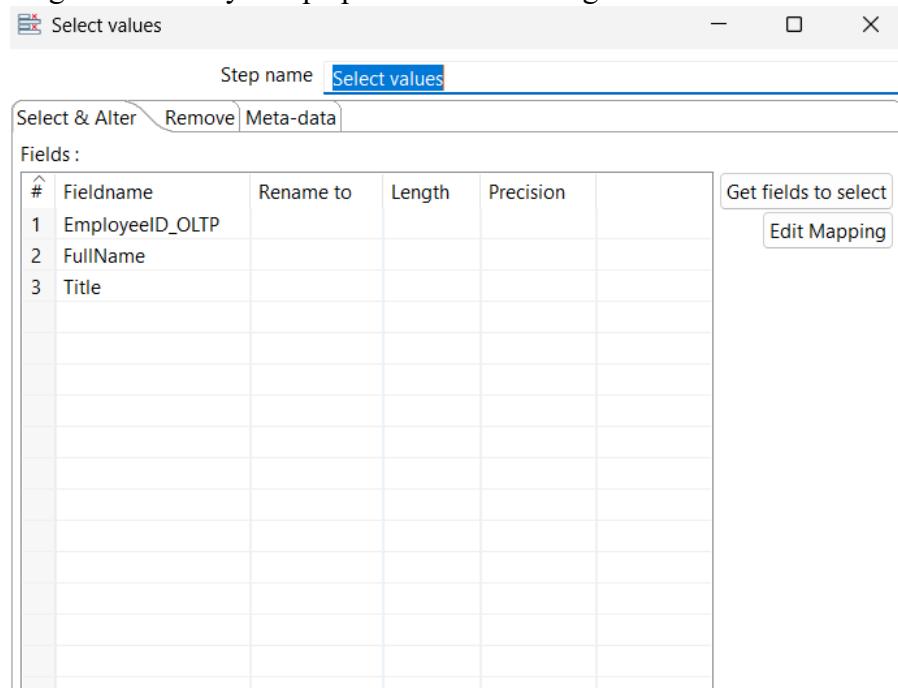


## LANGKAH – LANGKAH

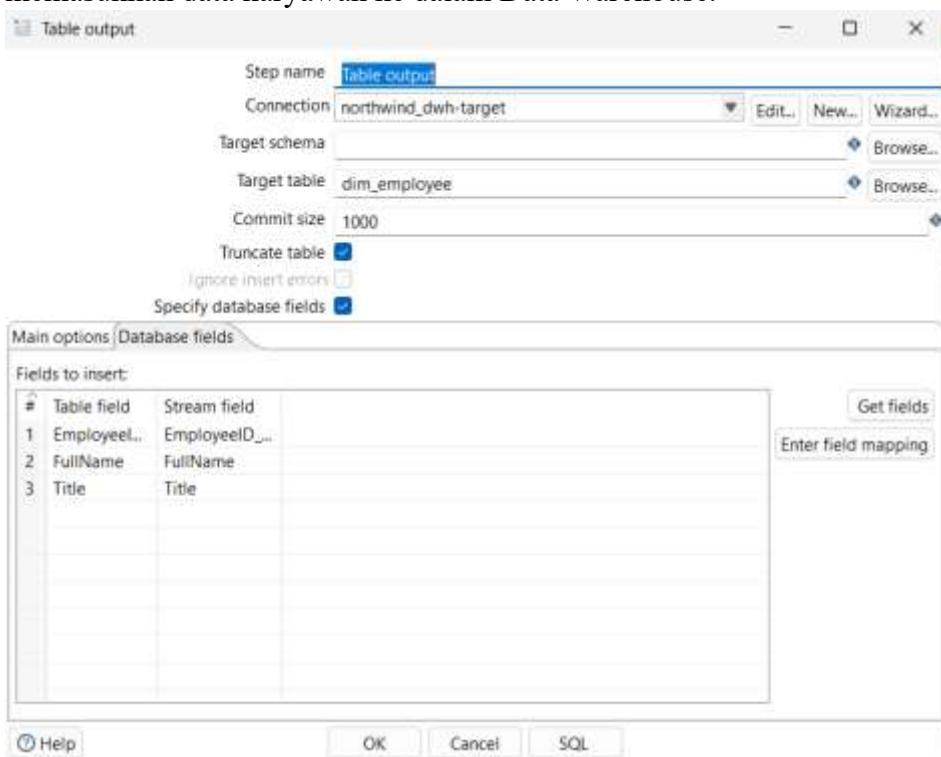
1. **Ekstraksi dan Manipulasi Data (Extract)** Berbeda dengan dimensi lain yang mengambil data mentah secara langsung,
  - id diambil sebagai EmployeeID\_OLTP untuk referensi kunci asli.
  - Fungsi CONCAT(first\_name, ' ', last\_name) digunakan untuk menggabungkan first\_name dan last\_name dengan pemisah spasi menjadi satu kolom baru bernama FullName.



2. **Penyesuaian Metadata (Transform)** Komponen **Select Values** digunakan untuk memastikan tipe data dan nama kolom yang mengalir di dalam *stream* sudah sesuai dengan spesifikasi tabel tujuan.
- **Aktivitas:** Memastikan field EmployeeID\_OLTP, FullName, dan Title diteruskan ke langkah berikutnya tanpa perubahan nama lagi karena sudah di-handle di query SQL



3. **Memuat Data ke Dimensi (Load)** Komponen **Table Output** digunakan untuk memasukkan data karyawan ke dalam Data Warehouse.



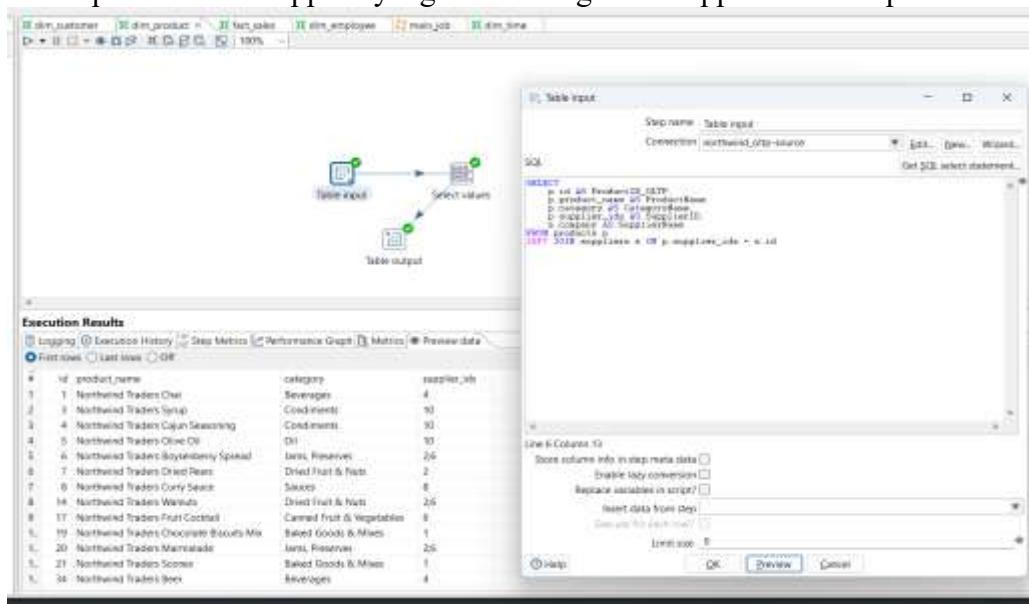
### 3.4 ETL Dimensi Product (Dim\_Product)

Proses ini menggabungkan data produk dan supplier:

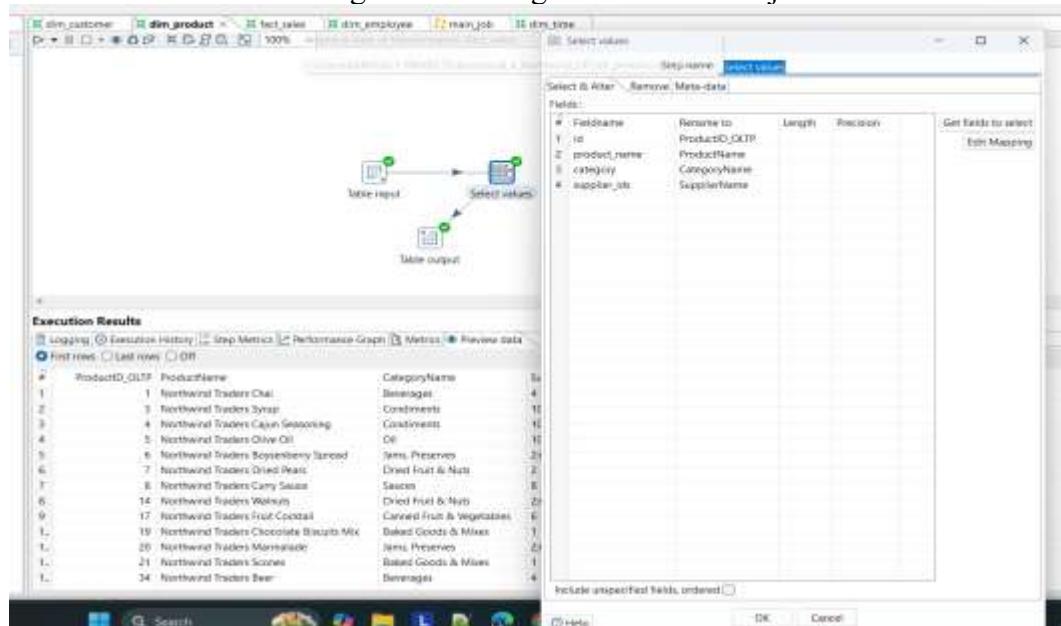
1. **Extract (Table Input):** Melakukan query SQL dengan LEFT JOIN antara tabel products dan suppliers untuk mendapatkan nama supplier dan kategori.
  - o *Query:* SELECT p.id, p.product\_name, p.category, s.company AS SupplierName...
2. **Transform (Select Values):** Mapping kolom product\_name menjadi ProductName, category menjadi CategoryName.
3. **Load (Table Output):** Data bersih dimuat ke tabel dim\_product di data warehouse.

#### LANGKAH - LANGKAH

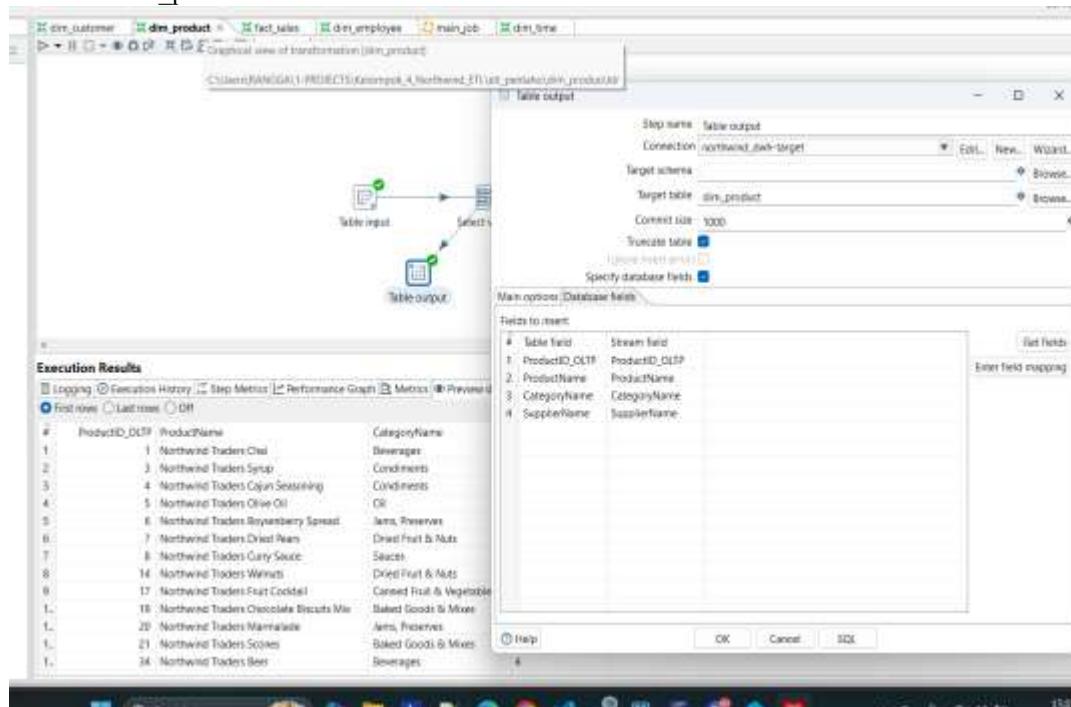
1. **Query SQL:** Melakukan LEFT JOIN antara tabel products (p) dan suppliers (s) untuk mendapatkan nama supplier yang sesuai dengan ID supplier di tabel produk.



2. **Transformasi (Transform)** Menggunakan komponen **Select Values** untuk memetakan nama kolom dari sumber agar sesuai dengan skema tabel tujuan.

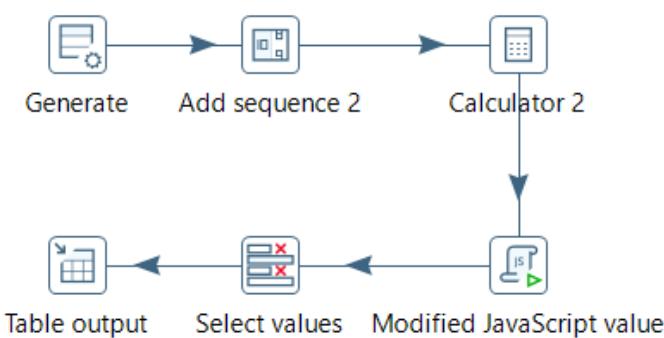


3. **Memuat Data (Load)** Menggunakan komponen **Table Output** untuk menyimpan data ke tabel dim\_product.



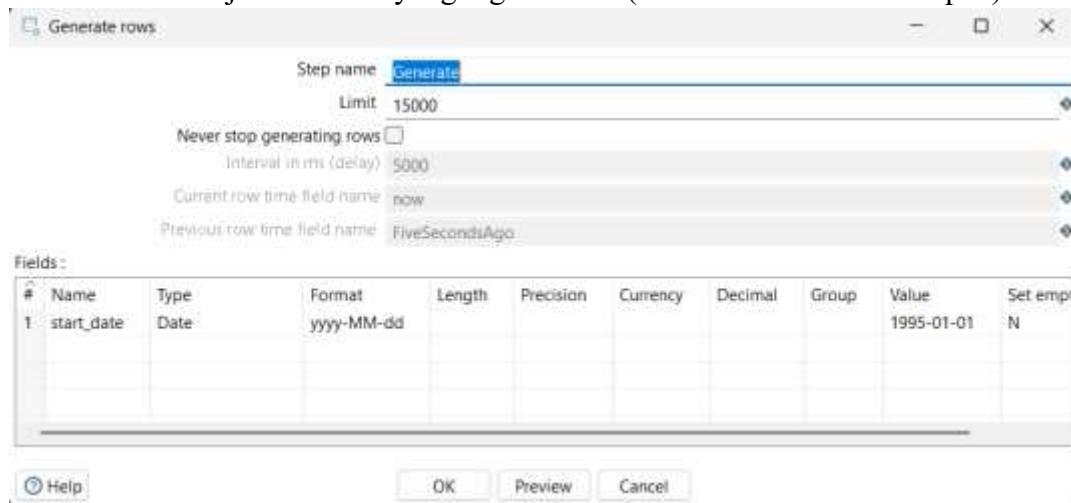
### 3.5 ETL Dimensi Waktu (Dim\_Time)

- 1 Dimensi waktu adalah komponen krusial dalam Data Warehouse yang memungkinkan analisis berdasarkan periode (Tahun, Kuartal, Bulan, Hari). Berbeda dengan dimensi lain yang diekstrak dari database sumber, data dimensi waktu dibangkitkan (*generated*) secara prosedural menggunakan transformasi Pentaho untuk mencakup rentang waktu transaksi perusahaan.
- 2 **Alur Transformasi:** Proses dimulai dengan membangkitkan baris kosong, membuat urutan angka, mengalkulasi tanggal, memperkaya data dengan JavaScript (nama bulan/hari), menyesuaikan metadata, dan memuat ke database.

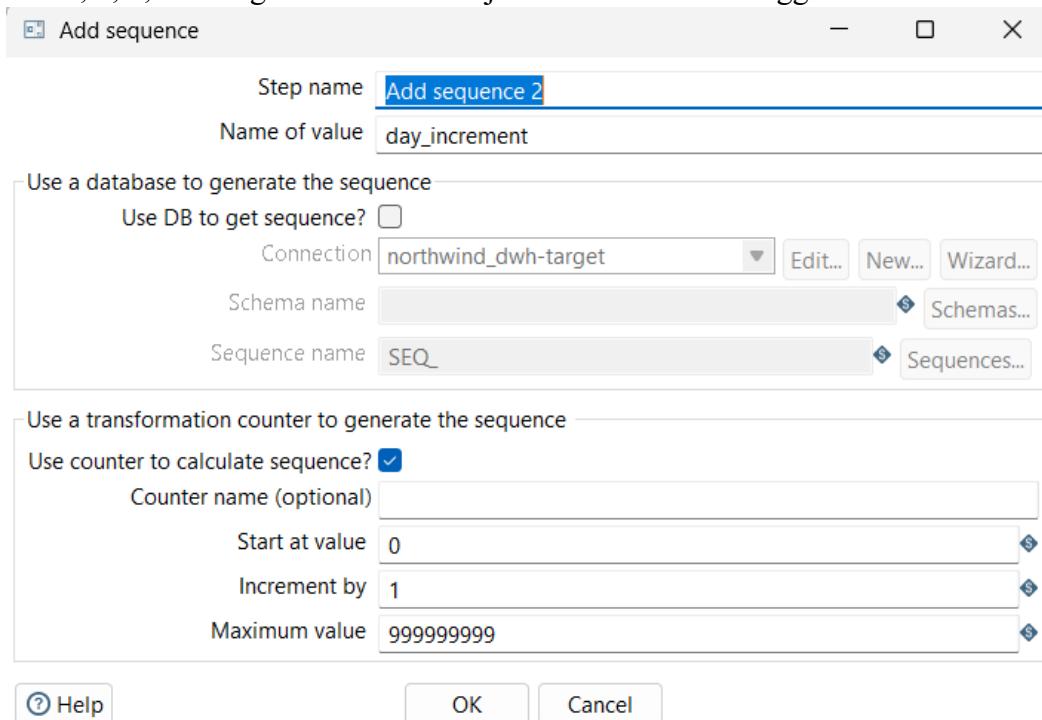


#### LANGKAH – LANGKAH

- 1 **Membangkitkan Data Deret Waktu (Generate & Sequence)** Karena tabel waktu tidak tersedia di database sumber,
  - **Generate Rows:** Langkah ini menginisialisasi titik awal tanggal (misal: 1995-01-01) dan menentukan jumlah baris yang ingin dibuat (misal: 15.000 hari ke depan).



- **Add Sequence:** Membuat counter atau penomoran otomatis (day\_increment) mulai dari 0, 1, 2, dst. Angka ini mewakili "jumlah hari setelah tanggal awal".



- 2 Kalkulasi Atribut Tanggal (Calculator) Komponen Calculator digunakan untuk menghitung tanggal aktual dan mengekstrak komponen waktu dasarnya.

- **Date Calculation:** Rumus Date A + B Days digunakan untuk menambahkan start\_date dengan day\_increment. Hasilnya adalah kolom date\_actual yang berisi tanggal berurutan.
- **Extraction:** Mengekstrak Tahun (Year), Bulan (Month), Hari (Day), dan Kuarter (Quarter) dari date\_actual menggunakan fungsi internal Calculator.

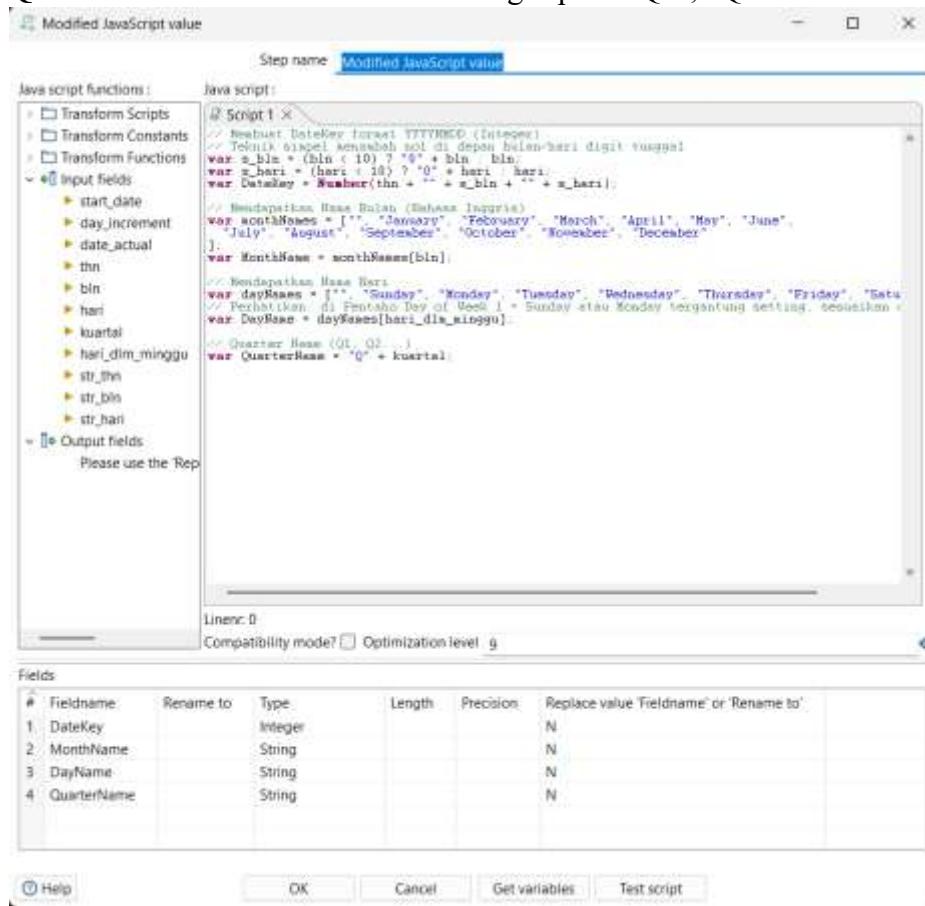
Step name **Calculator**

Fields:

#	New Field	Calculation	Field A	Field B	Field C	Value type	Length	Precision	Remove	Conversion mask	Decimal symbol	Grouping separator
1	date_actual	Date A + B Days	start_date	day_increment		Date	N	N	N			
2	thn	Year of date A	date_actual			Integer	N	N	N			
3	bln	Month of date A	date_actual			Integer	N	N	N			
4	hari	Day of month of date A	date_actual			Integer	N	N	N			
5	kuarter	Quarter of date A	date_actual			Integer	N	N	N			
6	hari_dlm_minggu	Day of week of date A	date_actual			String	N	N	N			
7	str_thn	Create a copy of field A	thn			String	N	N	N			
8	str_bln	Create a copy of field A	bln			String	N	N	N			
9	str_hari	Create a copy of field A	hari			String	N	N	N			

3 Pengayaan Data dengan Scripting (Modified JavaScript Value) Komponen ini digunakan untuk logika yang lebih kompleks yang tidak tersedia di Calculator standar.

- **DateKey (Integer):** Membuat Primary Key format integer (YYYYMMDD) untuk performa indexing yang lebih cepat.
  - Rumus: (Year \* 10000) + (Month \* 100) + Day.
- **Nama Bulan & Hari:** Mengubah angka bulan (1-12) menjadi nama (January-December) dan angka hari menjadi nama hari (Sunday-Saturday) menggunakan array JavaScript.
- **Quarter Name:** Membuat format string seperti "Q1", "Q2".



- 4 Standarisasi Metadata (Select Values) Komponen Select Values digunakan untuk mengganti nama variabel teknis menjadi nama kolom bisnis yang sesuai dengan skema database.

Select values

Step name |

Select & Alter Remove Meta-data

Fields :

#	Fieldname	Rename to	Length	Precision	
1	DateKey				
2	date_actual	Date			
3	bln	Month			
4	kuartal	Quarter			
5	thn	Year			
6	hari_dlm_minggu	DayOfWeek			
7	MonthName				
8	QuarterName				
9	DayName	Dayname			

Get fields to select

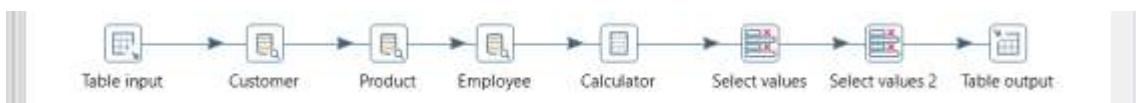
Edit Mapping

The screenshot shows a software interface titled 'Select values'. At the top, there's a 'Step name' input field. Below it, a navigation bar has tabs: 'Select & Alter' (which is active), 'Remove', and 'Meta-data'. Underneath, a section labeled 'Fields:' contains a table. The table has columns: '#', 'Fieldname', 'Rename to', 'Length', 'Precision', and an empty column. There are 9 rows of data. Row 5 (thn) has its 'Rename to' field ('Year') highlighted with a light blue background. On the right side of the table, there are two buttons: 'Get fields to select' and 'Edit Mapping'.

### 3.6 ETL Tabel Fakta (Fact\_Sales)

Tabel fakta adalah pusat dari data warehouse yang menyimpan data transaksi:

1. **Extract:** Melakukan query JOIN pada sumber (orders dan order\_details) untuk mengambil data transaksi.
2. **Lookup Keys:**
  - o Mengambil CustomerKey dari dim\_customer berdasarkan CustomerID.
  - o Mengambil ProductKey dari dim\_product berdasarkan ProductID .
3. **Transform (Calculator):** Membuat kolom baru Total\_Sales dengan rumus perkalian Quantity \* UnitPrice.
4. **Load:** Menyimpan hasil akhir ke tabel fact\_sales.



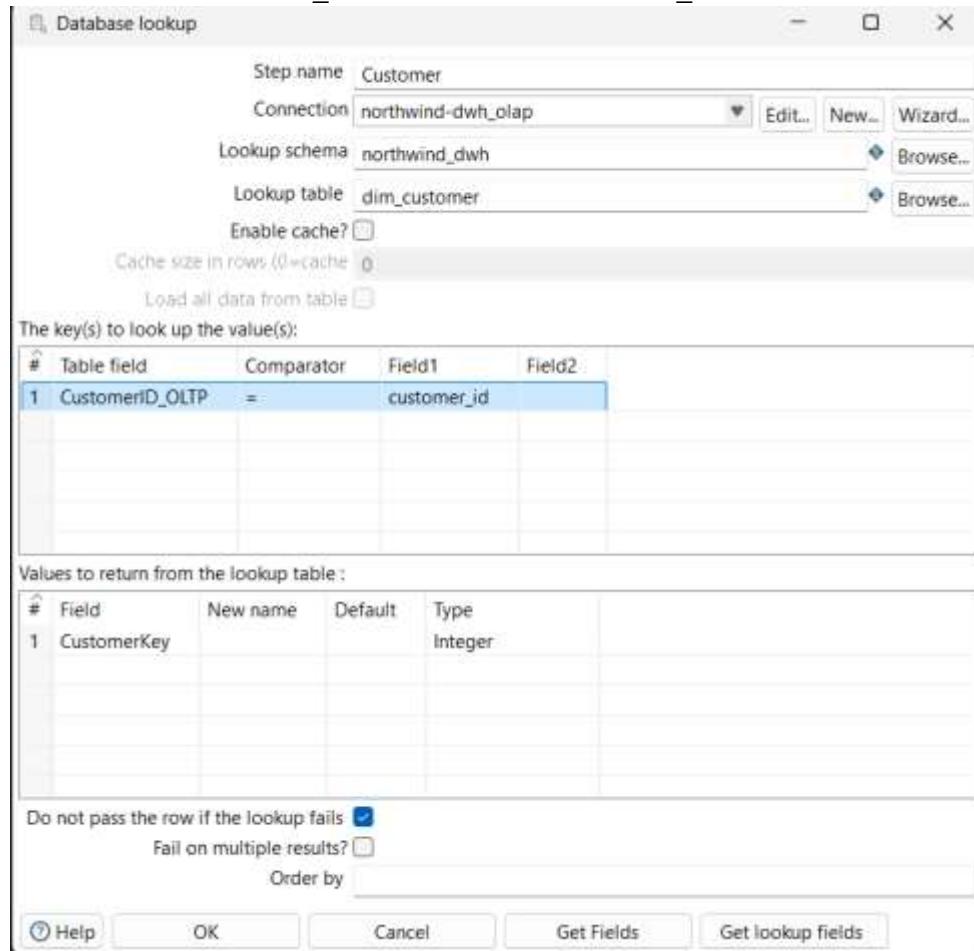
#### LANGKAH LANGKAH

1. **Ekstraksi Data Transaksi (Extract)** Menggunakan komponen **Table Input** untuk mengambil data gabungan antara tabel orders (header transaksi) dan order\_details (detail barang).

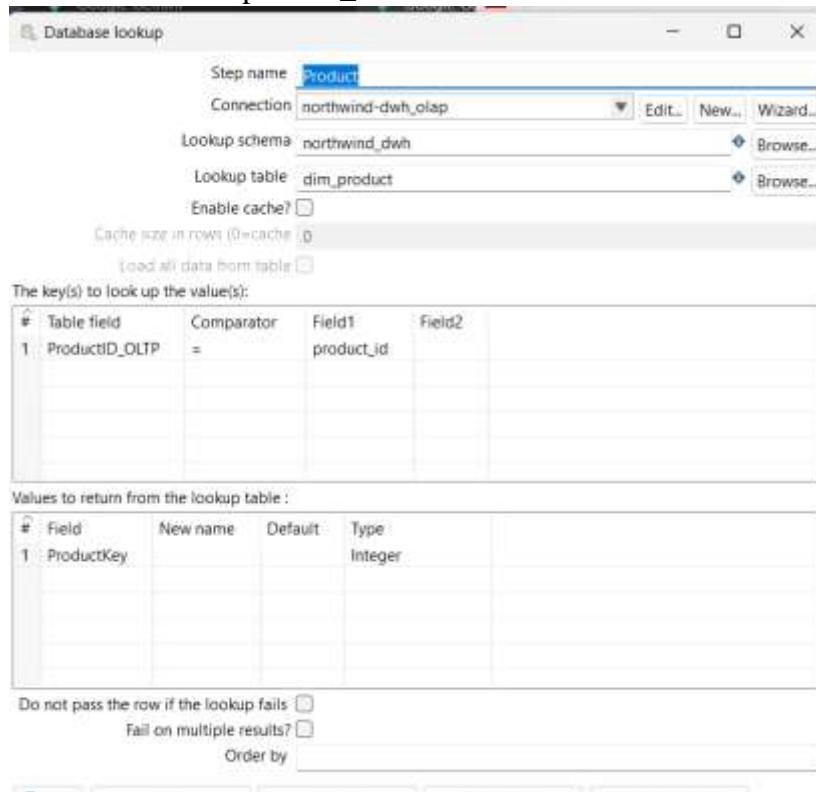


2. **Penggantian Kunci Dimensi (Lookup Keys)** Data dari sumber operasional (OLTP) masih menggunakan ID asli (misal: 'ALFKI'). Di Data Warehouse, ID ini harus diganti dengan *Surrogate Key* (Integer) yang telah dibuat pada proses ETL dimensi sebelumnya.

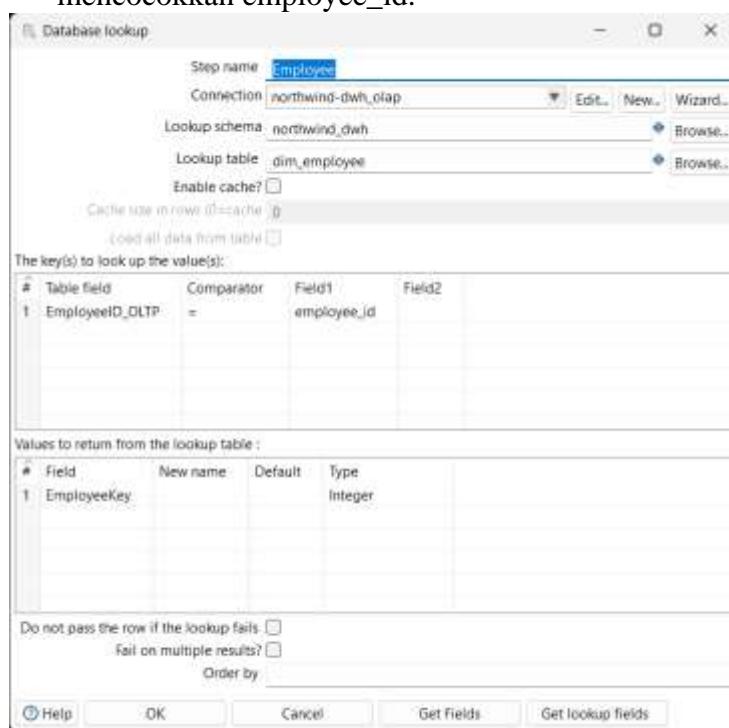
- **Customer Lookup:** Mengambil CustomerKey dari tabel dim\_customer dengan mencocokkan customer\_id sumber dan CustomerID\_OLTP dimensi.



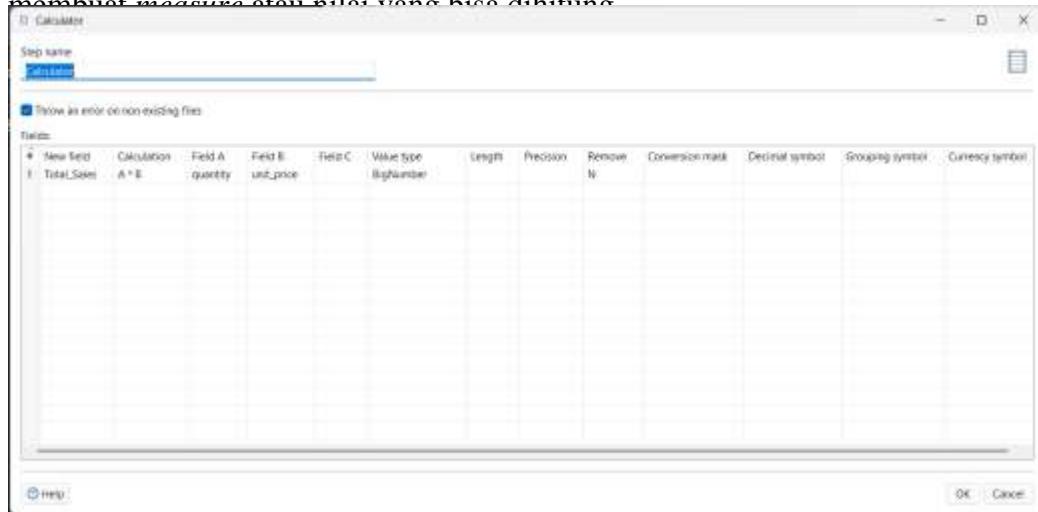
- **Product Lookup:** Mengambil ProductKey dari tabel dim\_product dengan mencocokkan product\_id.



- **Employee Lookup:** Mengambil EmployeeKey dari tabel dim\_employee dengan mencocokkan employee\_id.

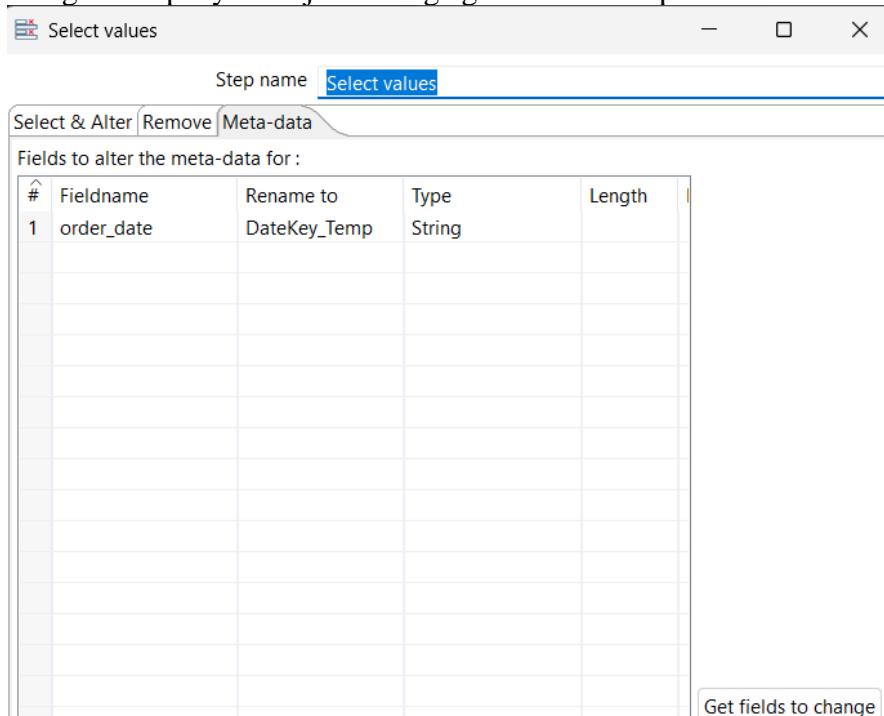


3. **Perhitungan Metrik (Calculation)** Menggunakan komponen **Calculator** untuk membuat measure atau nilai yang bisa dihitung.

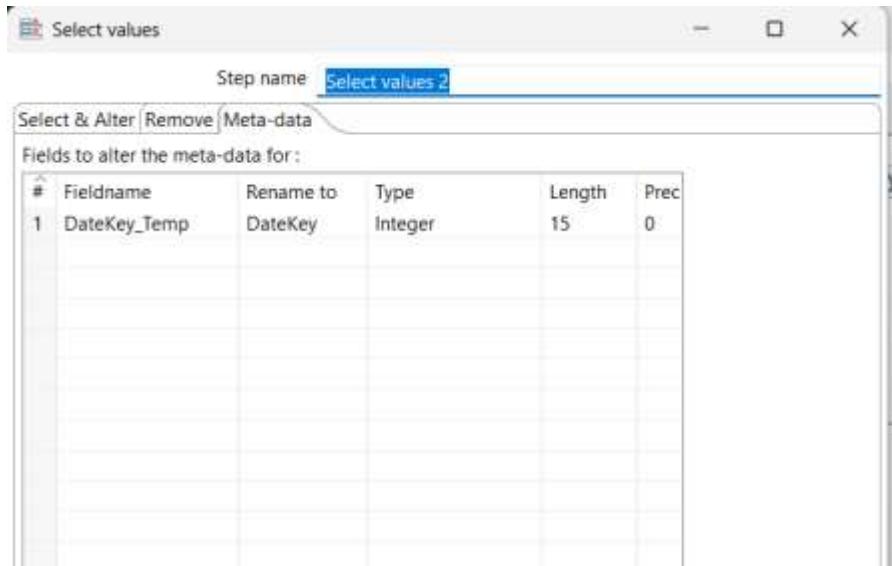


4. Transformasi Format Tanggal (DateKey) Dalam Data Warehouse, tanggal sering disimpan sebagai Integer (misal: 20231025) agar performa join ke Dimensi Waktu lebih cepat.

- **Select Values 1 (Metadata):** Mengubah nama `order_date` menjadi `DateKey_Temp` dan mengubah tipenya menjadi String agar bisa dimanipulasi.

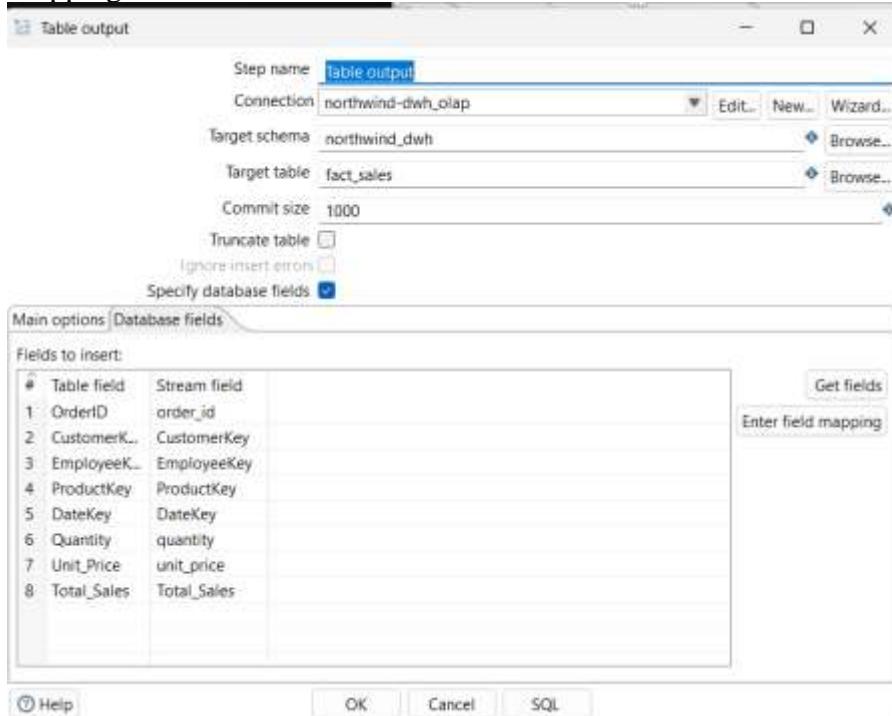


- Select Values 2 (Final): Mengubah format string tersebut menjadi Integer pada kolom DateKey final.



5. Memuat Data ke Tabel Fakta (Load) Langkah terakhir menggunakan komponen Table Output untuk menyimpan data yang sudah matang ke tabel fact\_sales.

- Target Table: fact\_sales di database northwind\_dwh.
- Mapping Field: Memetakan kolom dari stream ke kolom database:



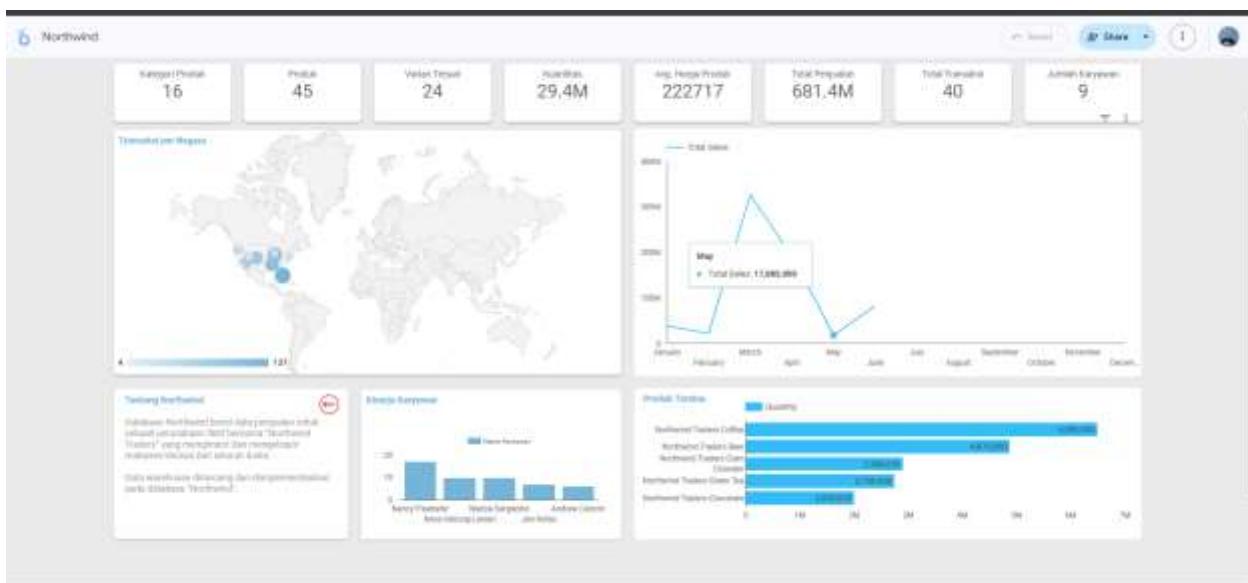
# BAB 4: HASIL DAN ANALISIS

## 4.1 Hasil Data Warehouse

Setelah proses ETL dijalankan (Execute), data warehouse northwind\_dwh telah terisi dengan data yang terstruktur dan bersih. Data ini siap digunakan untuk analisis bisnis menggunakan tools visualisasi seperti Power BI/Tableau/Looker.

## 4.2 Visualisasi Dashboard & Analisis KPI

Berikut adalah visualisasi dashboard yang menjawab Key Performance Indicators (KPI) bisnis:



### Analisis 1: Total Penjualan per Periode

- Visualisasi:** Line Chart.
- Insight:** Grafik ini menunjukkan tren pendapatan perusahaan dari waktu ke waktu. (Tambahkan analisis Anda: misal, "Penjualan meningkat pada kuartal ke-X").

### Analisis 2: Top 5 Produk Terlaris

- Visualisasi:** Bar Chart.
- Insight:** Mengidentifikasi produk unggulan yang paling banyak berkontribusi pada pendapatan. Produk-produk ini harus diprioritaskan dalam manajemen stok.

### Analisis 3: Persebaran Penjualan per Negara

- Visualisasi:** Map/Geo Chart.
- Insight:** Menunjukkan negara mana yang menjadi pasar terbesar bagi Northwind Traders, yang berguna untuk strategi pemasaran regional.

## **BAB 5: KESIMPULAN**

Proyek ini berhasil mentransformasi data operasional Northwind menjadi Data Warehouse dengan skema bintang. Pipeline ETL yang dibangun menggunakan Pentaho Data Integration berhasil melakukan ekstraksi, transformasi, dan loading data dengan akurat. Dashboard yang dihasilkan memberikan wawasan bisnis yang cepat mengenai performa penjualan, produk terlaris, dan distribusi pelanggan .