

BUSINESS INTELLIGENCE

LAPORAN PROYEK AKHIR



Oleh :

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

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 pengerjaan 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)

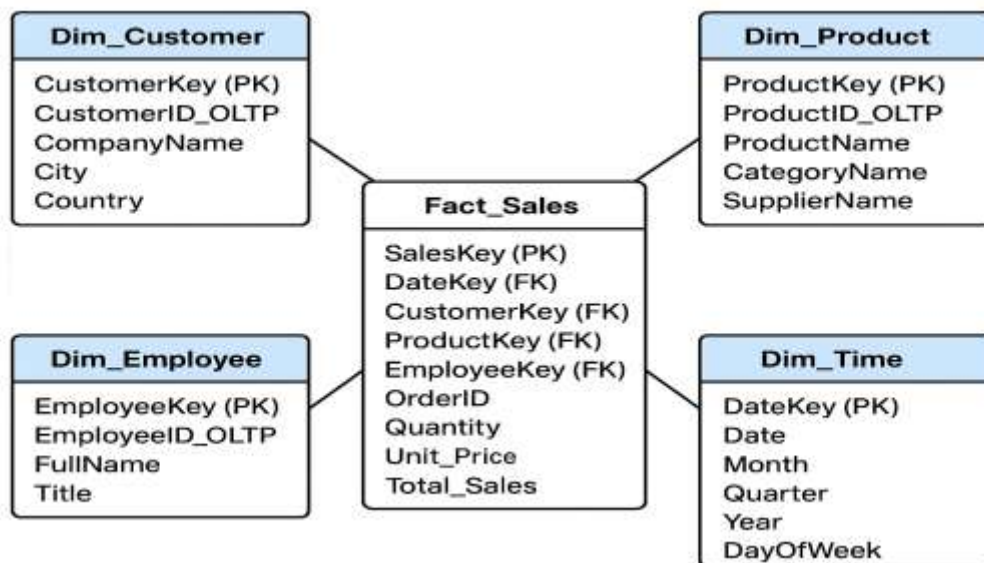
Data sumber berasal dari database operasional (OLTP) Northwind yang berjalan di MySQL. Dataset ini mencakup tabel utama seperti employees, orders, order_details, products, dan customers yang telah divalidasi memiliki komponen 4W (Who, What, Where, When) + 1H (How Much).

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:

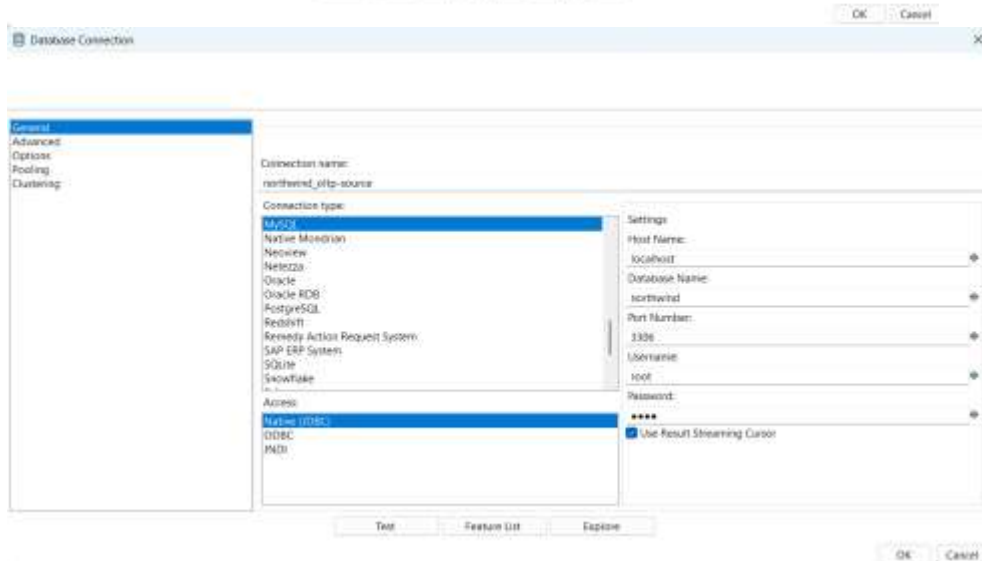
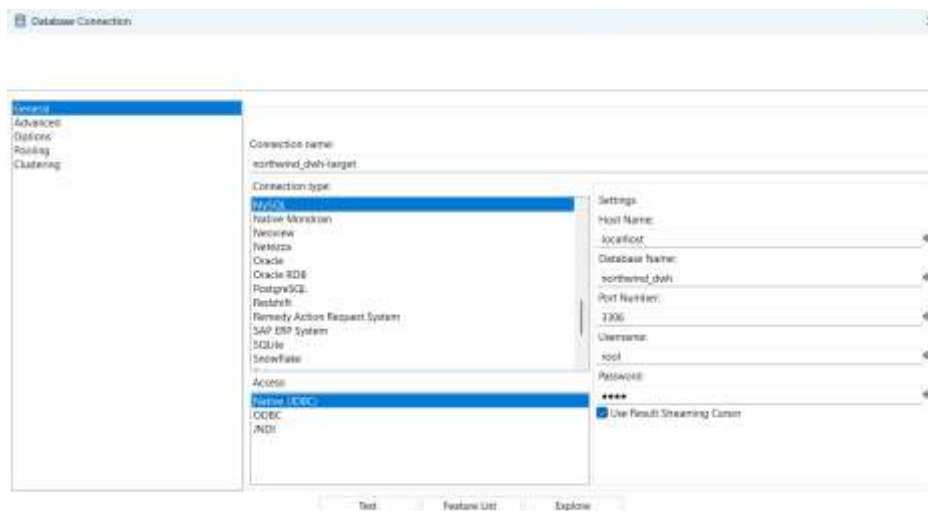
IMPLEMENTASI ETL (EXTRACT, TRANSFORM, LOAD)

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) .



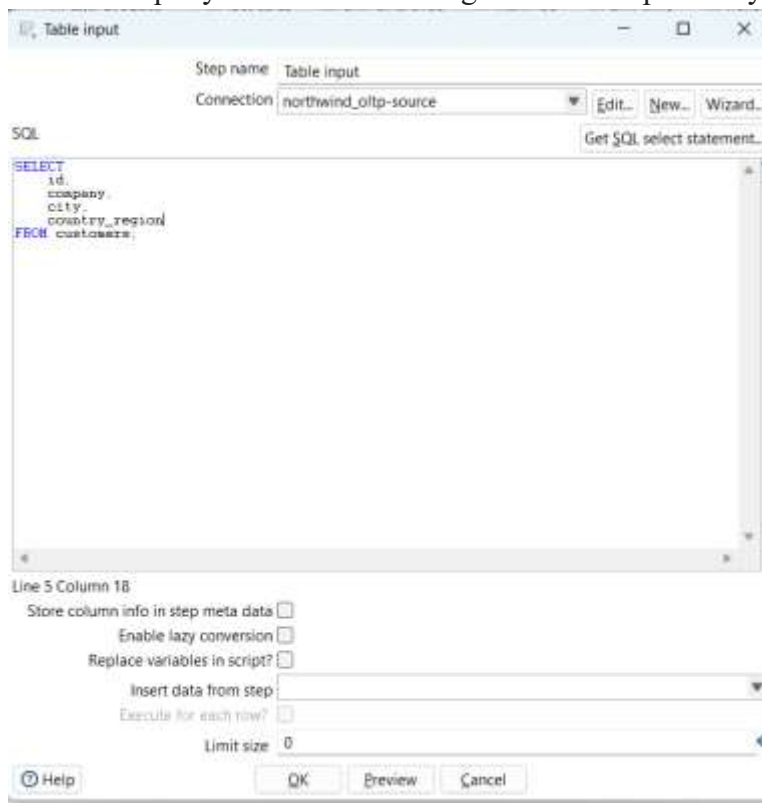
3.2 ETL Dimensi Customer (Dim_Customer)

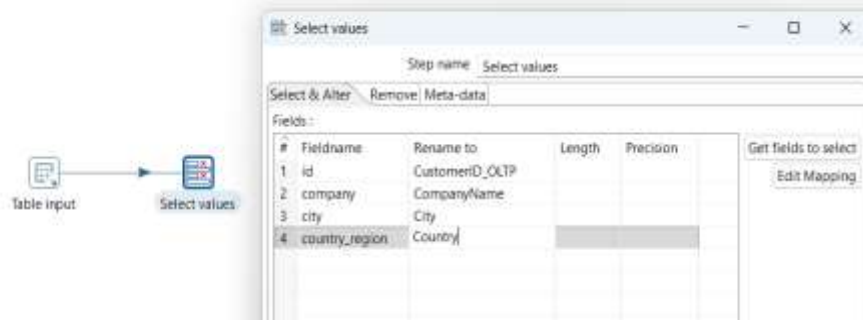
Proses pengisian tabel dimensi pelanggan melibatkan langkah berikut:

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*.

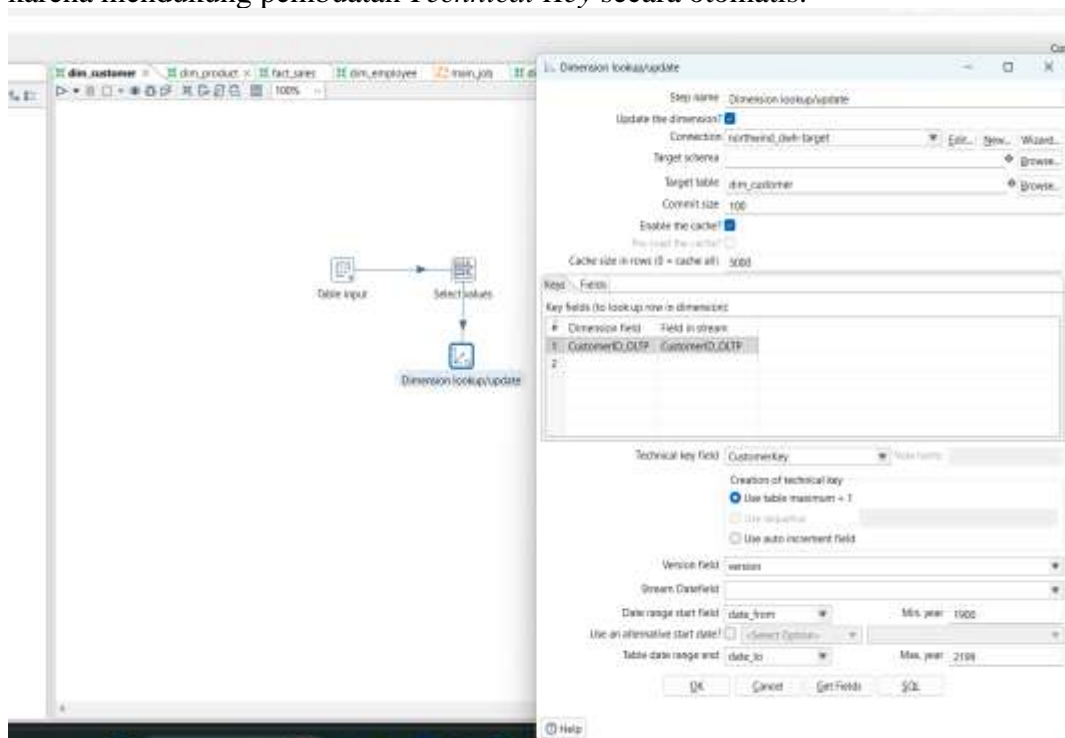
LANGKAH - LANGKAH

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

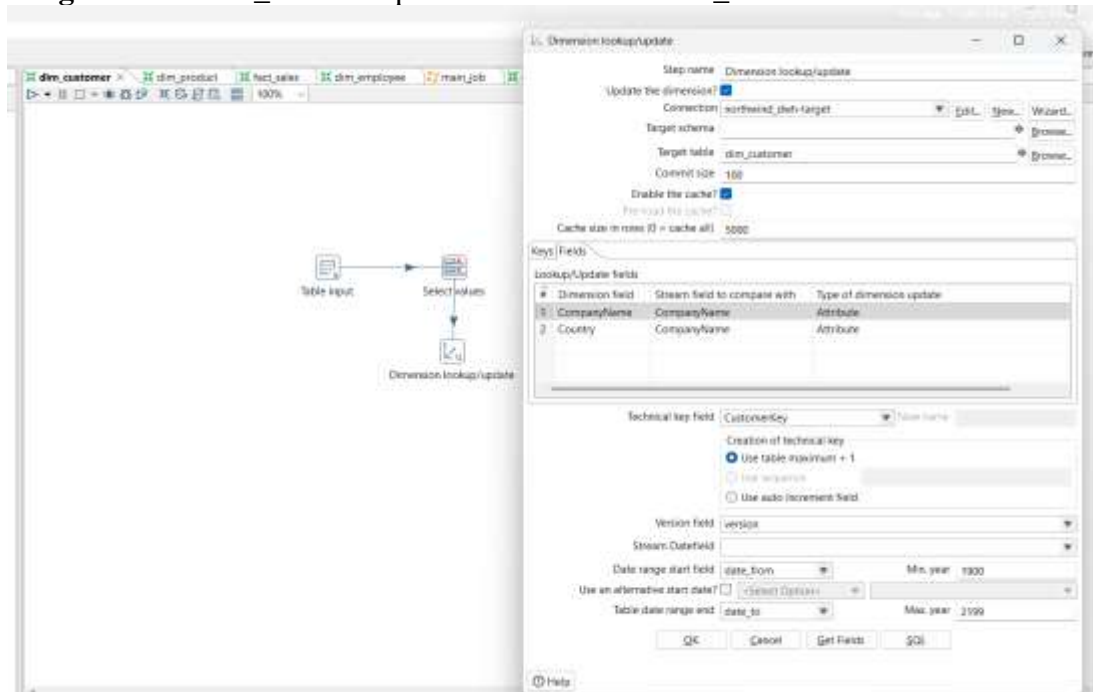




- 3.
4. tambahkan select values dan ubah atau rename menjadi nama sama dengan table target agar lebih mudah mengambilnya nanti
5. **Memuat Data ke Dimensi (Load)** Komponen **Dimension Lookup/Update** digunakan untuk memuat data bersih ke dalam tabel dimensi di Data Warehouse. Metode ini dipilih karena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

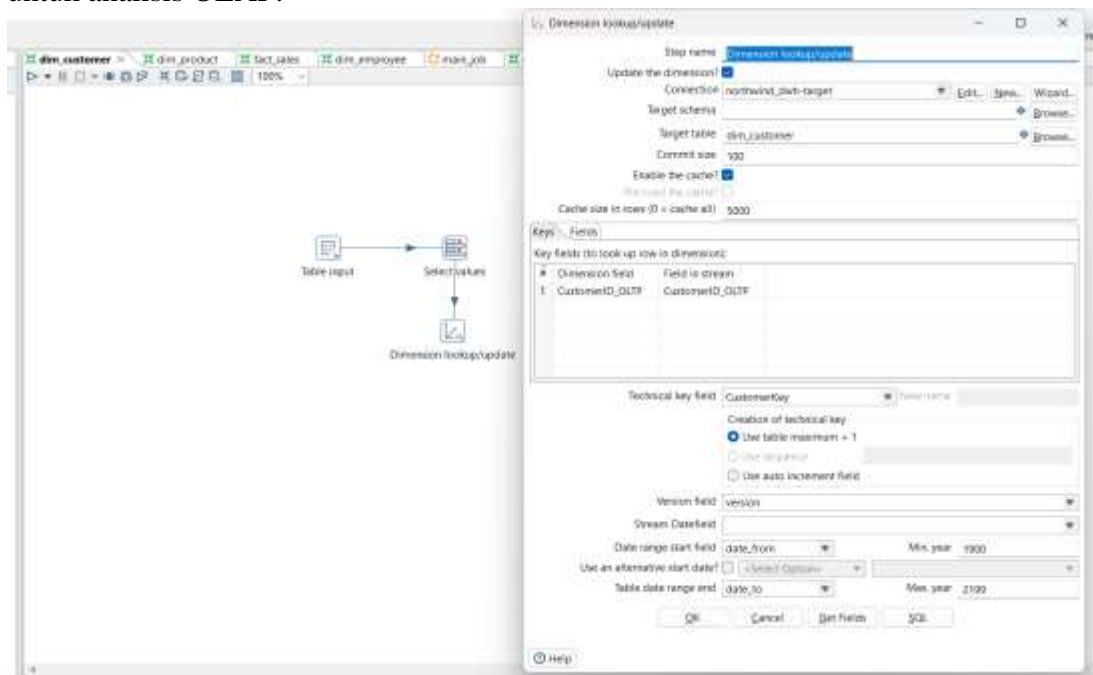


6. **Target Tabel:** dim_customer pada database northwind_dwh.



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 displays the SAP Data Services interface for configuring a 'Table output' step. The main window shows a data flow diagram with 'Table input' and 'Table output' steps. The 'Table output' step is selected, and its configuration is shown in the right-hand pane.

Table output configuration:

- Step name: Table output
- Connection: northwind_dwh-target
- Target schema: dim_customer
- Target table: dim_customer
- Commit size: 1000
- Truncate table: ☒
- Ignore insert errors: ☐
- Specify database fields: ☒

Main options - Database fields:

- Partition data over tables: ☐
- Partitioning field:
- Partition data per month: ☐
- Partition data per day: ☐
- Use batch update for inserts: ☒
- Is the name of the table defined in a field? ☐
- Field that contains name of table:
- Store the table name field: ☐
- Return auto-generated key: ☐
- Name of auto-generated key field:

Fields to insert:

Table field	Stream field
1 CustomerID_OULTP	CustomerID_OULTP
2 CompanyName	CompanyName
3 City	City
4 Country	Country

The 'Execution Results' pane at the bottom shows the log of the transformation execution, including the start and end times of the steps and the status of the data load.

11. HASIL

The screenshot displays a data workflow interface. At the top, there are tabs for various data sources: `dim_customer`, `dim_product`, `fact_sales`, `dim_employee`, `main_job`, and `dim_time`. The main workspace shows a workflow diagram with three nodes: `Table input`, `Select values`, and `Table output`. Arrows indicate the flow from `Table input` to `Select values`, and then to `Table output`. Below the workspace, the `Execution Results` section is active, showing a table of data.

Execution Results

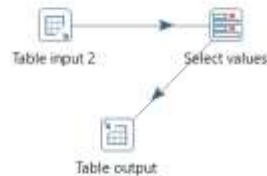
Logging Execution History Step Metrics Performance Graph Metrics Preview data

☒ First rows ☐ Last rows ☐ Off

#	CustomerID_OULTP	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	Boise	USA
8	8	Company H	Portland	USA
9	9	Company I	Salt Lake City	USA

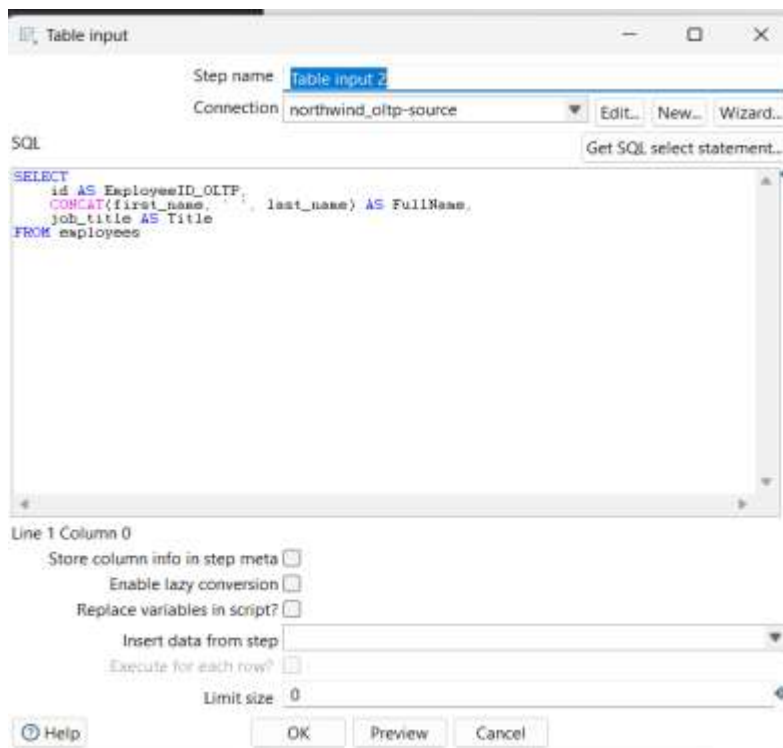
3.3 ETL Dimensi Karyawan (Dim_Employee)

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

#	Fieldname	Rename to	Length	Precision
1	EmployeeID_OLTP			
2	FullName			
3	Title			

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

#	Table field	Stream field
1	EmployeeID_OLTP	EmployeeID_OLTP
2	FullName	FullName
3	Title	Title

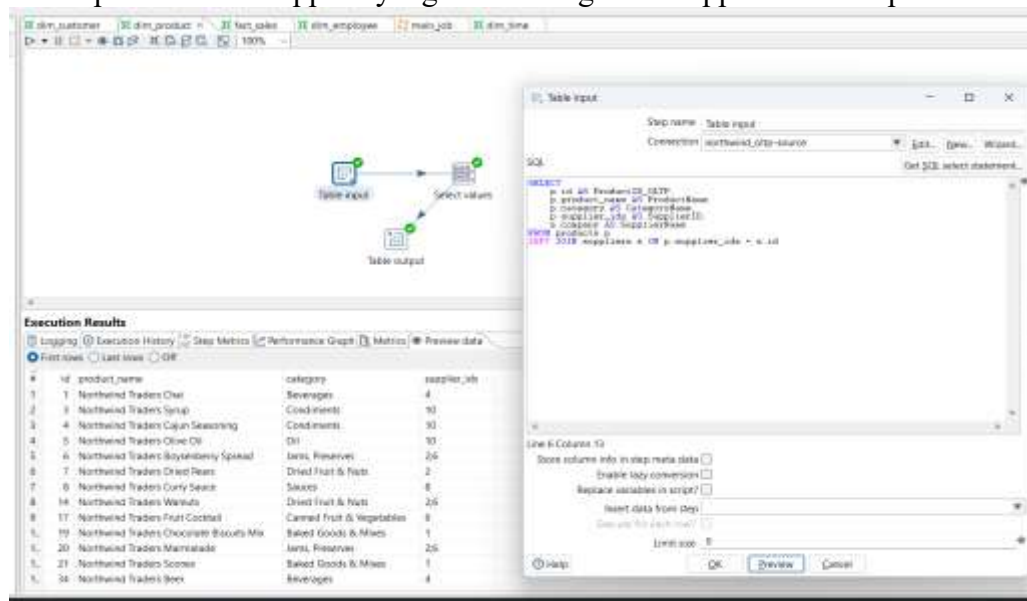
3.4 ETL Dimensi Product (Dim_Product)

Proses ini menggabungkan data produk dan supplier:

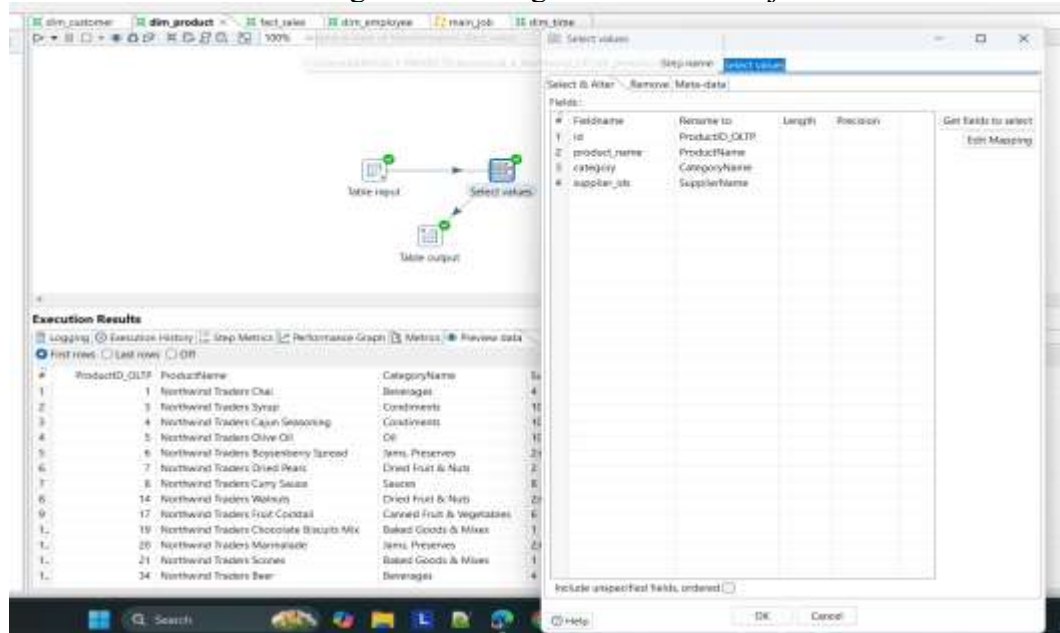
1. **Extract (Table Input):** Melakukan query SQL dengan LEFT JOIN antara tabel products dan suppliers untuk mendapatkan nama supplier dan kategori.
 - o *Query:* SELECT p.id, p.product_name, p.category, s.company AS SupplierName...
2. **Transform (Select Values):** Mapping kolom product_name menjadi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

LANGKAH - LANGKAH

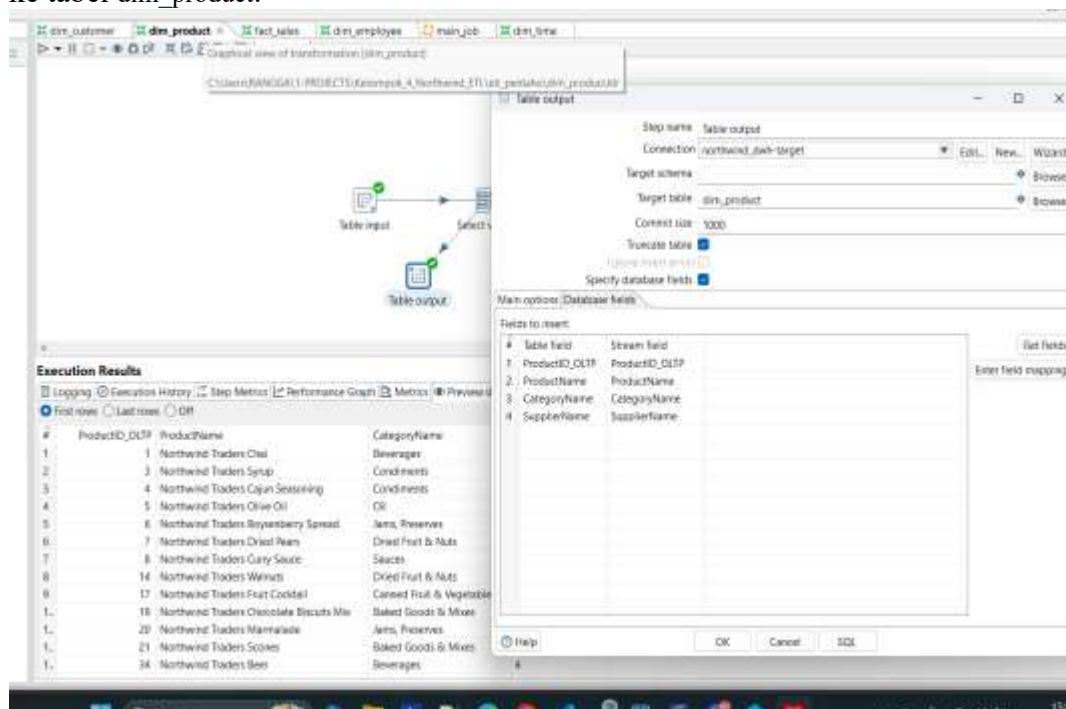
1. **Query SQL:** Melakukan LEFT JOIN antara tabel products (p) dan suppliers (s) untuk mendapatkan nama supplier yang sesuai dengan ID supplier di tabel produk.



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

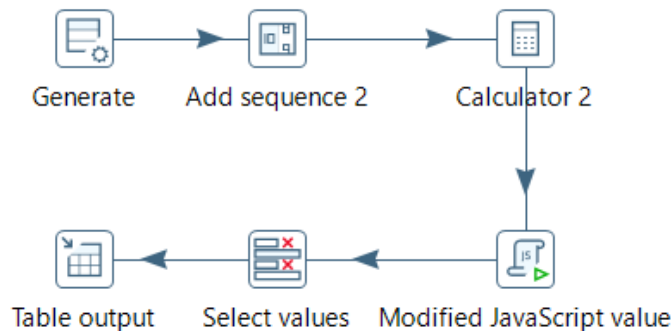


3. **Memuat Data (Load)** Menggunakan komponen **Table Output** untuk menyimpan data ke tabel `dim_product`.



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, mengkalkulasi 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).

The screenshot shows the 'Generate rows' dialog box in Pentaho. The 'Step name' is 'Generate'. The 'Limit' is set to 15000. The 'Never stop generating rows' checkbox is unchecked. The 'Interval in ms (delay)' is set to 5000. The 'Current row time field name' is 'now'. The 'Previous row time field name' is 'FiveSecondsAgo'.

Fields:

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Value	Set empty
1	start_date	Date	yyyy-MM-dd						1995-01-01	N

Buttons: Help, OK, Preview, Cancel

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

Step name: Add sequence 2

Name of value: day_increment

Use a database to generate the sequence

Use DB to get sequence? ☐

Connection: northwind_dwh-target

Schema name:

Sequence name: SEQ_

Use a transformation counter to generate the sequence

Use counter to calculate sequence? ☒

Counter name (optional):

Start at value: 0

Increment by: 1

Maximum value: 99999999

Help OK Cancel

- 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 Kuartal (Quarter) dari date_actual menggunakan fungsi internal Calculator.

Step name: Calculator

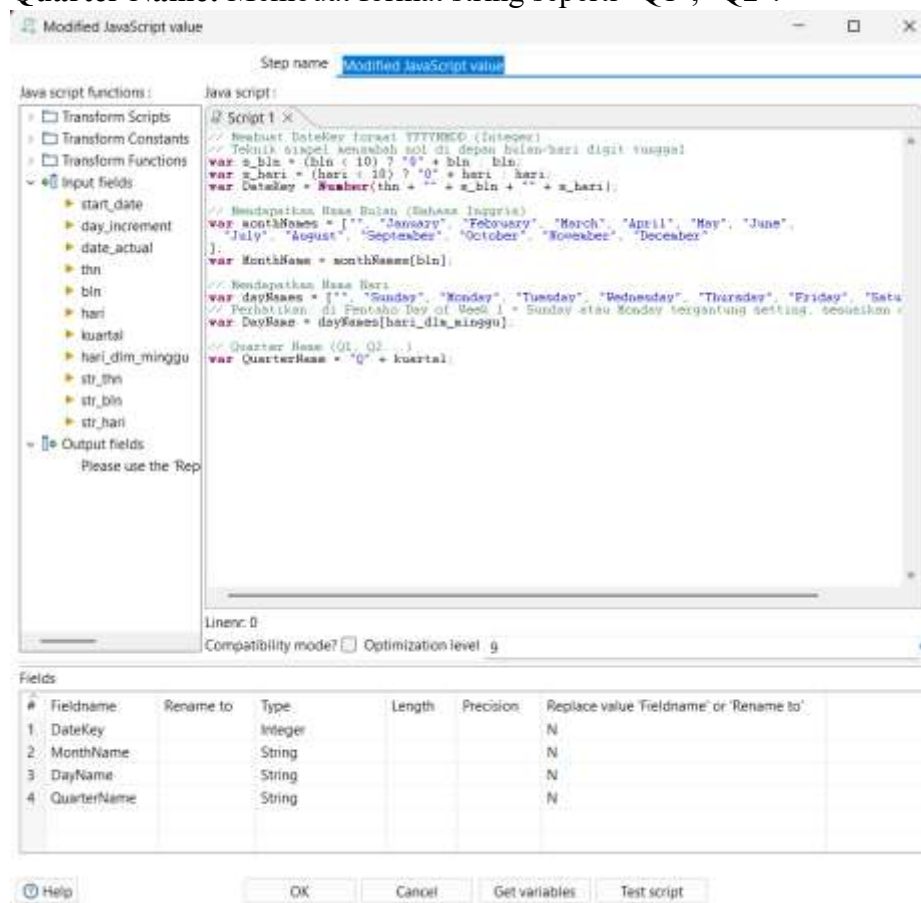
Throw an error on non existing files

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

Help OK Cancel

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: $(\text{Year} * 10000) + (\text{Month} * 100) + \text{Day}$.
- **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 & AlterRemoveMeta-data

Fields :

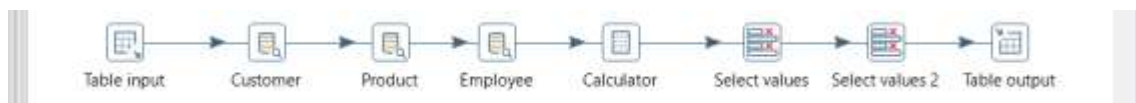
#	Fieldname	Rename to	Length	Precision	
1	DateKey				
2	date_actual	Date			
3	bln	Month			
4	kuartal	Quarter			
5	thn	Year			
6	hari_dlm_minggu	DayOfWeek			
7	MonthName				
8	QuarterName				
9	DayName	Dayname			

Get fields to selectEdit Mapping

3.6 ETL Tabel Fakta (Fact_Sales)

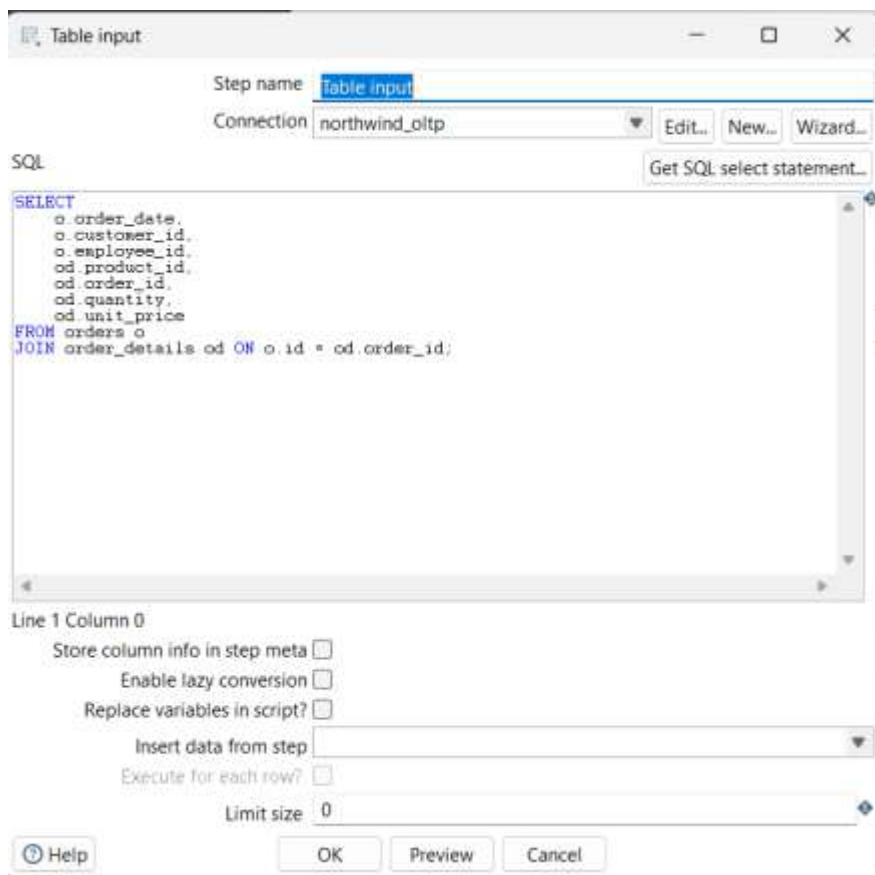
Tabel fakta adalah pusat dari data warehouse yang menyimpan data transaksi:

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.

Database lookup

Step name: Customer

Connection: northwind-dwh_olap

Lookup schema: northwind_dwh

Lookup table: dim_customer

Enable cache? ☐

Cache size in rows (0=cache): 0

Load all data from table ☐

The key(s) to look up the value(s):

#	Table field	Comparator	Field1	Field2
1	CustomerID_OLTP	=	customer_id	

Values to return from the lookup table:

#	Field	New name	Default	Type
1	CustomerKey			Integer

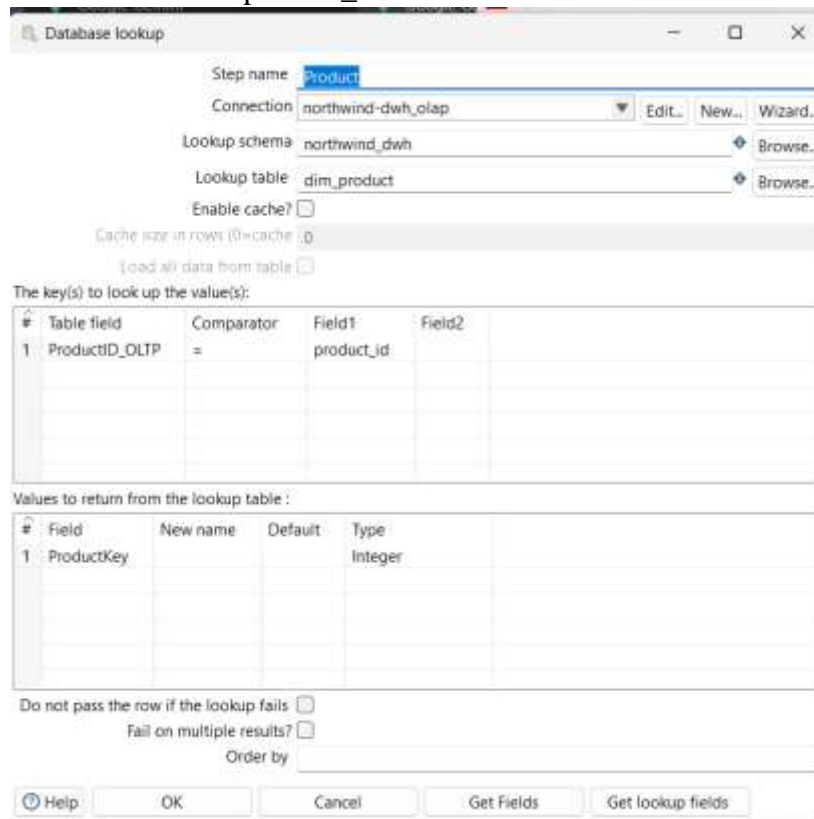
Do not pass the row if the lookup fails ☒

Fail on multiple results? ☐

Order by:

Buttons: Help, OK, Cancel, Get Fields, Get lookup fields

- **Product Lookup:** Mengambil ProductKey dari tabel dim_product dengan mencocokkan product_id.



Database lookup

Step name: **Product**

Connection: northwind-dwh_olap

Lookup schema: northwind_dwh

Lookup table: dim_product

Enable cache? ☐

Cache size in rows (0=cache): 0

Load all data from table ☐

The key(s) to look up the value(s):

#	Table field	Comparator	Field1	Field2
1	ProductID_OLTP	=	product_id	

Values to return from the lookup table:

#	Field	New name	Default	Type
1	ProductKey			Integer

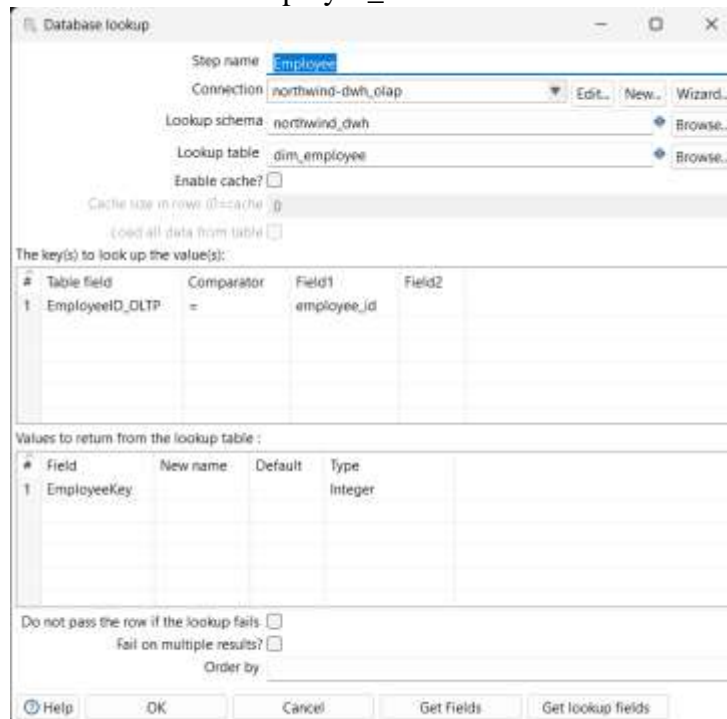
Do not pass the row if the lookup fails ☐

Fail on multiple results? ☐

Order by:

Buttons: Help, OK, Cancel, Get Fields, Get lookup fields

- **Employee Lookup:** Mengambil EmployeeKey dari tabel dim_employee dengan mencocokkan employee_id.



Database lookup

Step name: **Employee**

Connection: northwind-dwh_olap

Lookup schema: northwind_dwh

Lookup table: dim_employee

Enable cache? ☐

Cache size in rows (0=cache): 0

Load all data from table ☐

The key(s) to look up the value(s):

#	Table field	Comparator	Field1	Field2
1	EmployeeID_OLTP	=	employee_id	

Values to return from the lookup table:

#	Field	New name	Default	Type
1	EmployeeKey			Integer

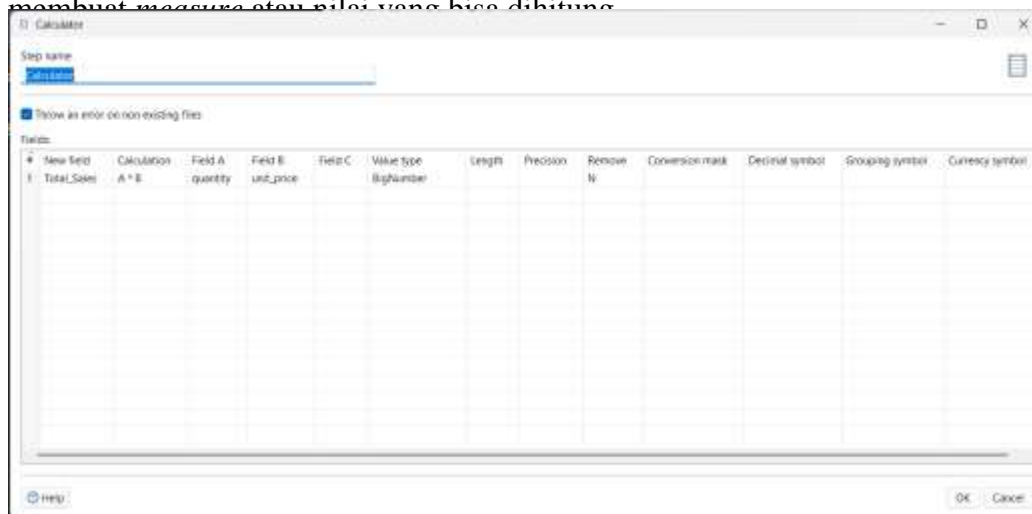
Do not pass the row if the lookup fails ☐

Fail on multiple results? ☐

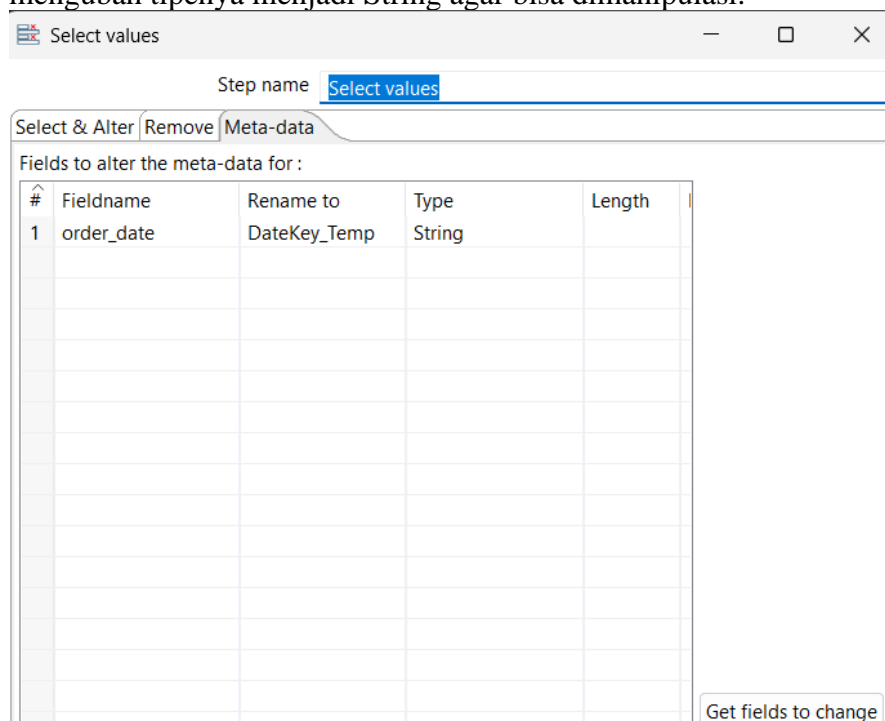
Order by:

Buttons: Help, OK, Cancel, Get Fields, Get lookup fields

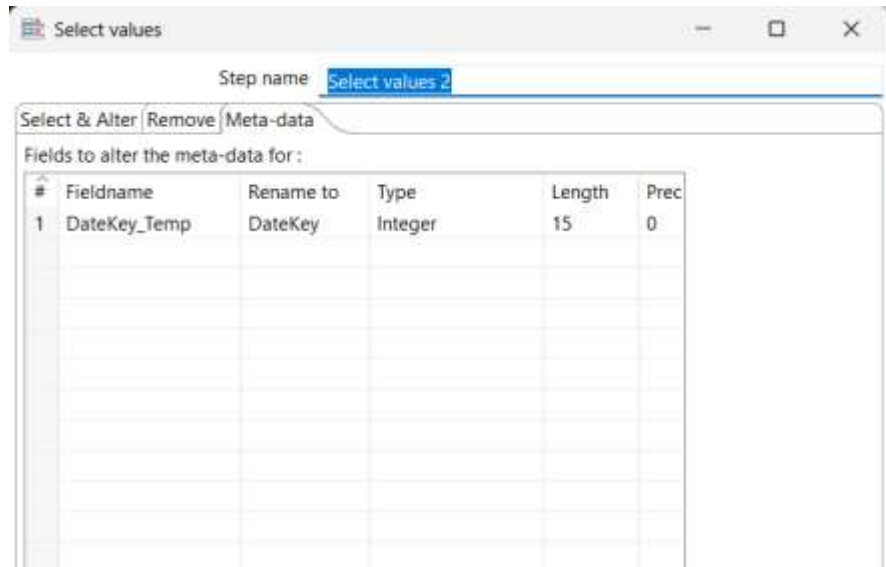
3. **Perhitungan Metrik (Calculation)** Menggunakan komponen **Calculator** untuk membuat pengukuran atau nilai yang bisa dihitung



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.

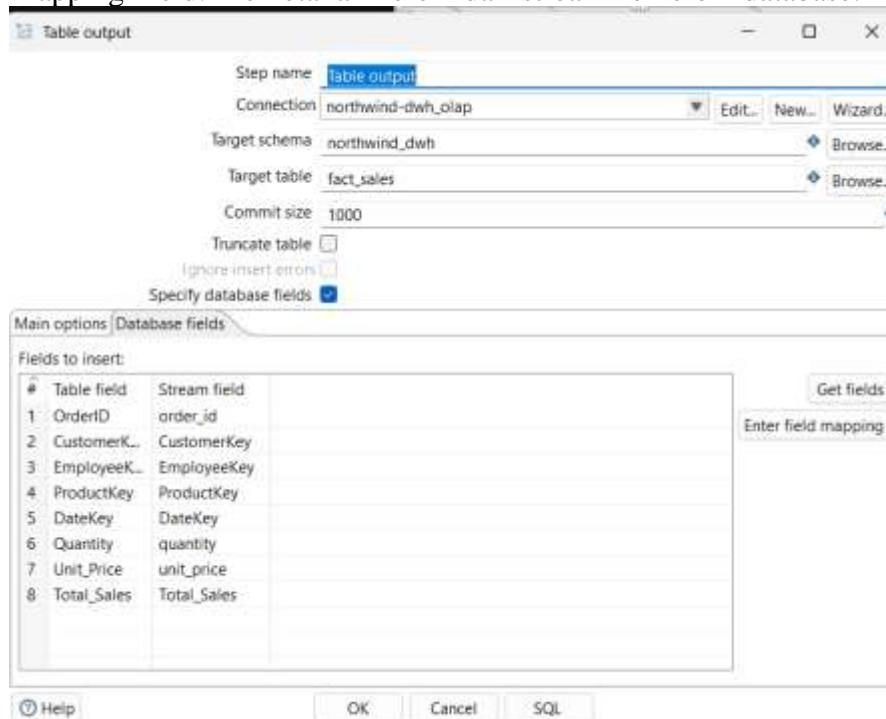


- Select Values 2 (Final): Mengubah format string tersebut menjadi Integer pada kolom DateKey final.



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:



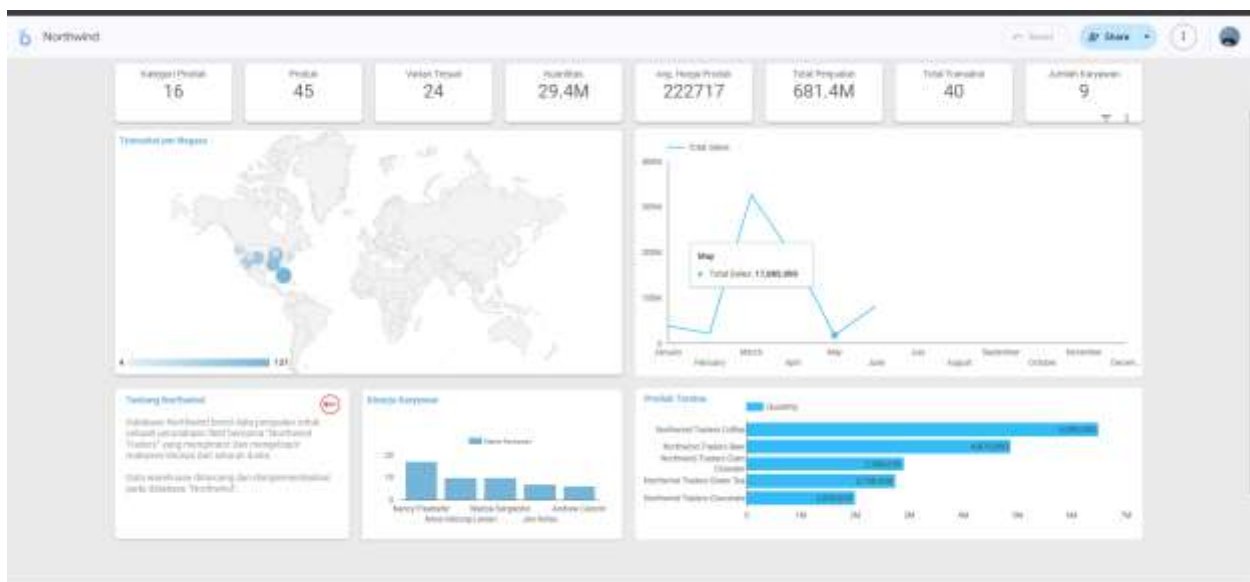
BAB 4: HASIL DAN ANALISIS

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 .