


Nama: Muhammad Rasya Hadziq NIM: 065001900033	 Praktikum Data Warehouse	MODUL 8 Nama Dosen: Ir. Teddy Siswanto, MMSi
Hari/Tanggal: Sabtu, 09/07/2022		Nama Asisten Labratorium: 1. Azhar Rizki Zulma 065001900001 2. Nadiya Amanda Rizkania 064001900003

Persiapan Proyek Akhir 2

1. Teori Singkat

Data warehouse adalah jenis sistem manajemen data yang dirancang untuk memungkinkan dan mendukung kegiatan business intelligence (BI), terutama analitik. Gudang data semata-mata dimaksudkan untuk melakukan kueri dan analisis dan sering berisi sejumlah besar data historis. Data dalam gudang data biasanya berasal dari berbagai sumber seperti file log aplikasi dan aplikasi transaksi. Gudang data memusatkan dan mengkonsolidasikan sejumlah besar data dari berbagai sumber. Kemampuan analitisnya memungkinkan organisasi untuk memperoleh wawasan bisnis yang berharga dari data mereka untuk meningkatkan pengambilan keputusan. Seiring waktu, ia membangun catatan sejarah yang dapat sangat berharga bagi para ilmuwan data dan analis bisnis. Karena kemampuan ini, gudang data dapat dianggap sebagai "sumber kebenaran tunggal" organisasi.

2. Alat dan Bahan

Hardware : Laptop/PC

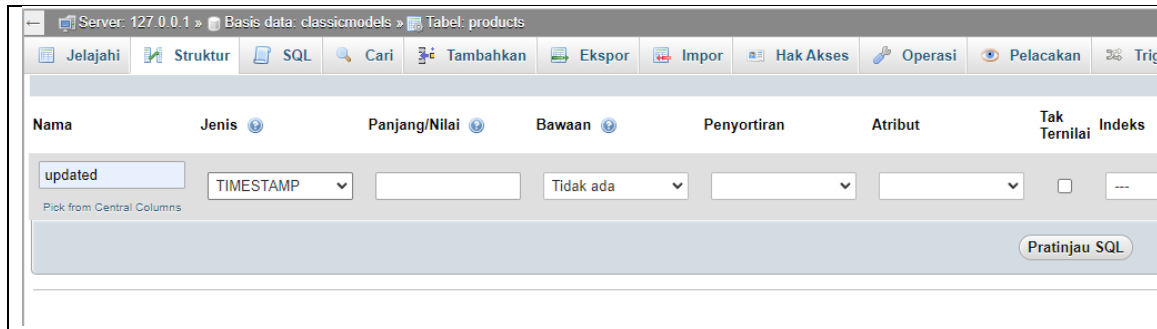
Software : Spoon Pentaho from Hitachi Vantara



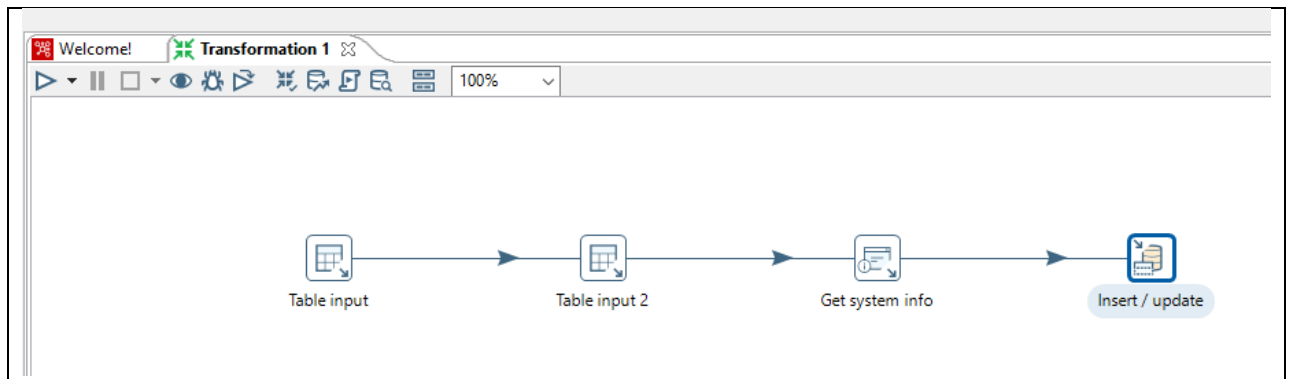
3. Elemen Kompetensi

a. Latihan pertama – Dim Product Transformation

1. Tambahkan kolom baru bernama updated dengan tipe data timestamp pada tabel products dalam database classicmodels seperti pada gambar dibawah ini di MySQL.



2. Struktur Dim Product



3. Max Last Update – Table input



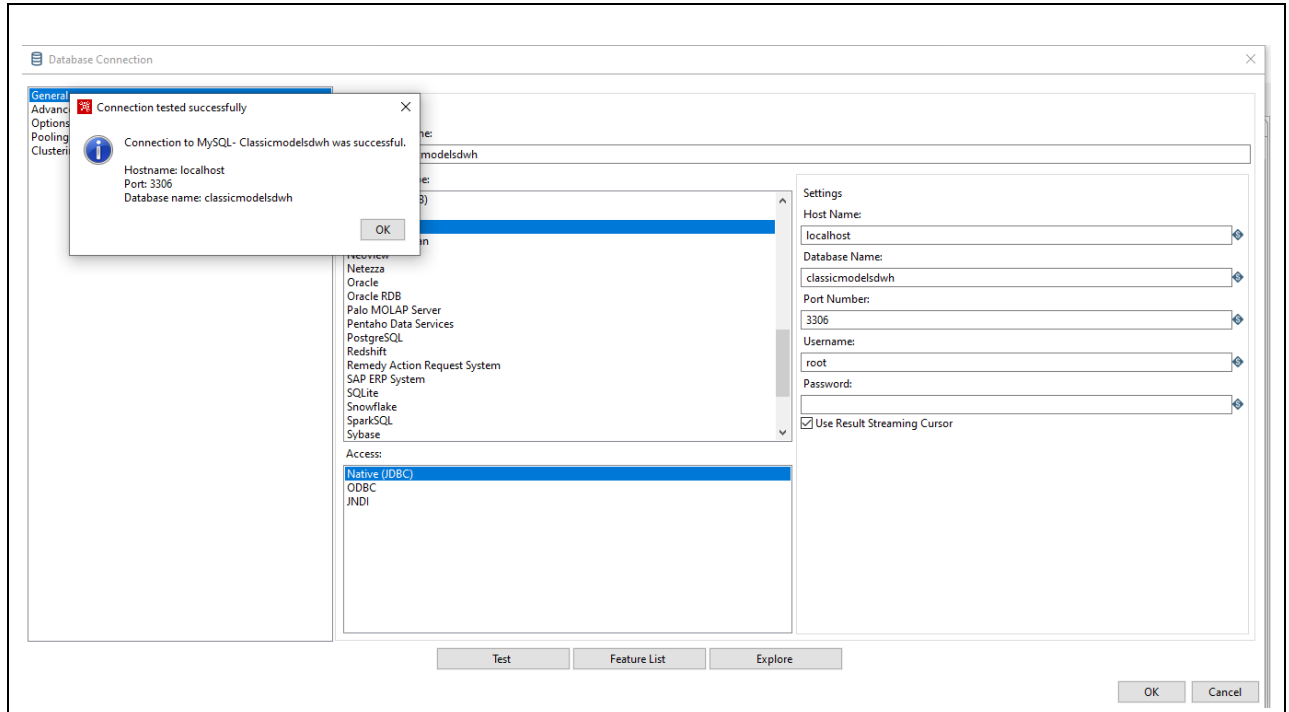


Table input

Step name: Max Last Update

Connection: MySQL- Classicmodelsdwh [Edit... New... Wizard...]

Get SQL select statement...

SQL

```
SELECT  
COALESCE(  
    MAX(last_update),  
    '1970-01-01 00:00:00'  
) max_last_update  
FROM dim_product
```

Line 6 Column 16

Store column info in step meta data ☐

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step [dropdown]

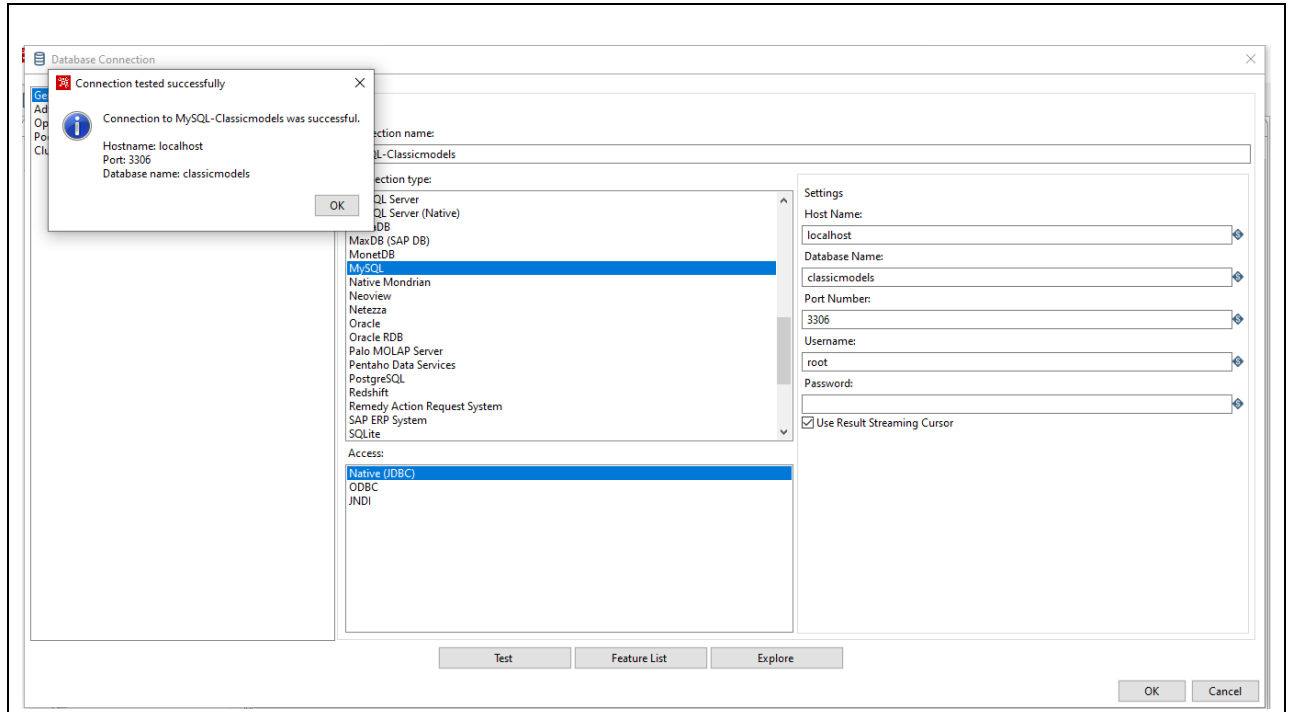
Execute for each row? ☐

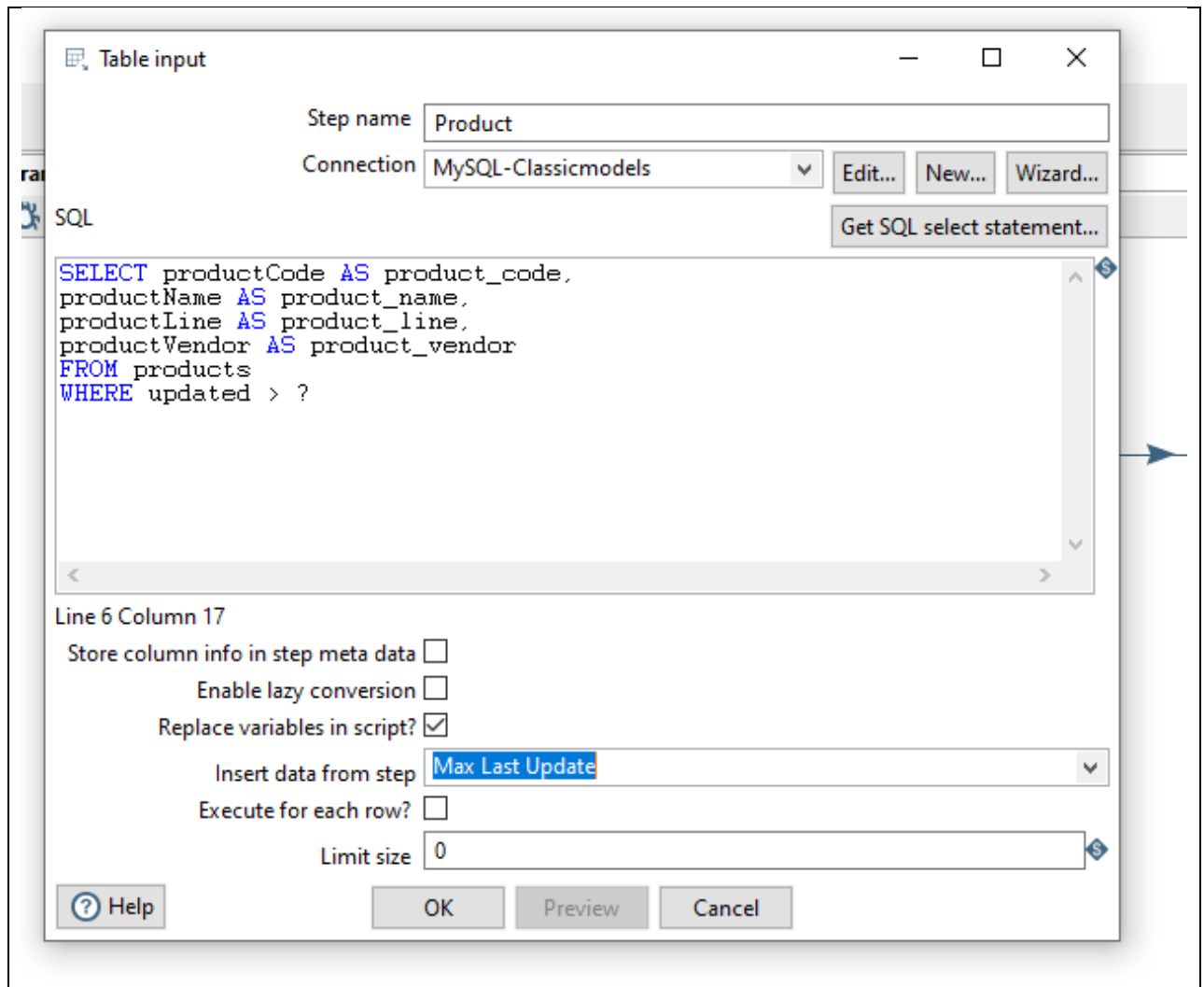
Limit size: 0

[Help] [OK] [Preview] [Cancel]

4. Product – Table Input.

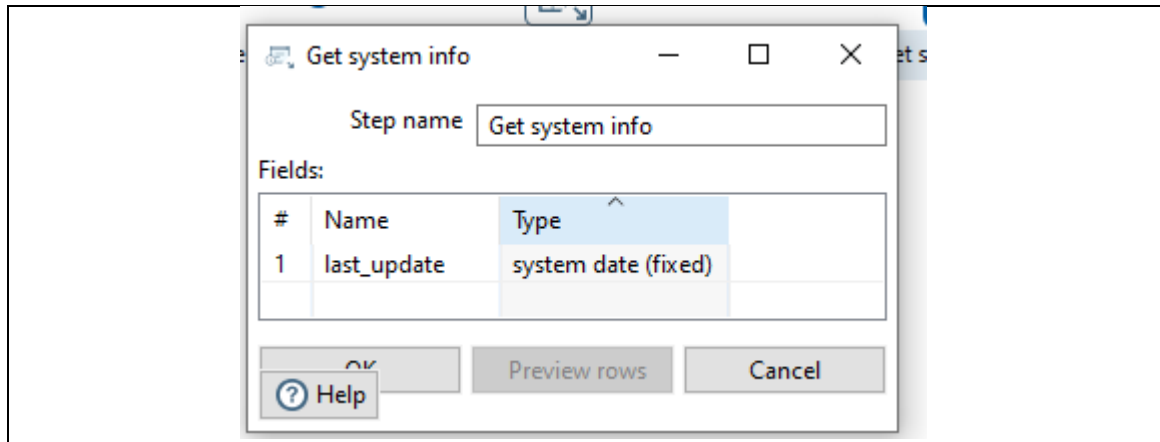






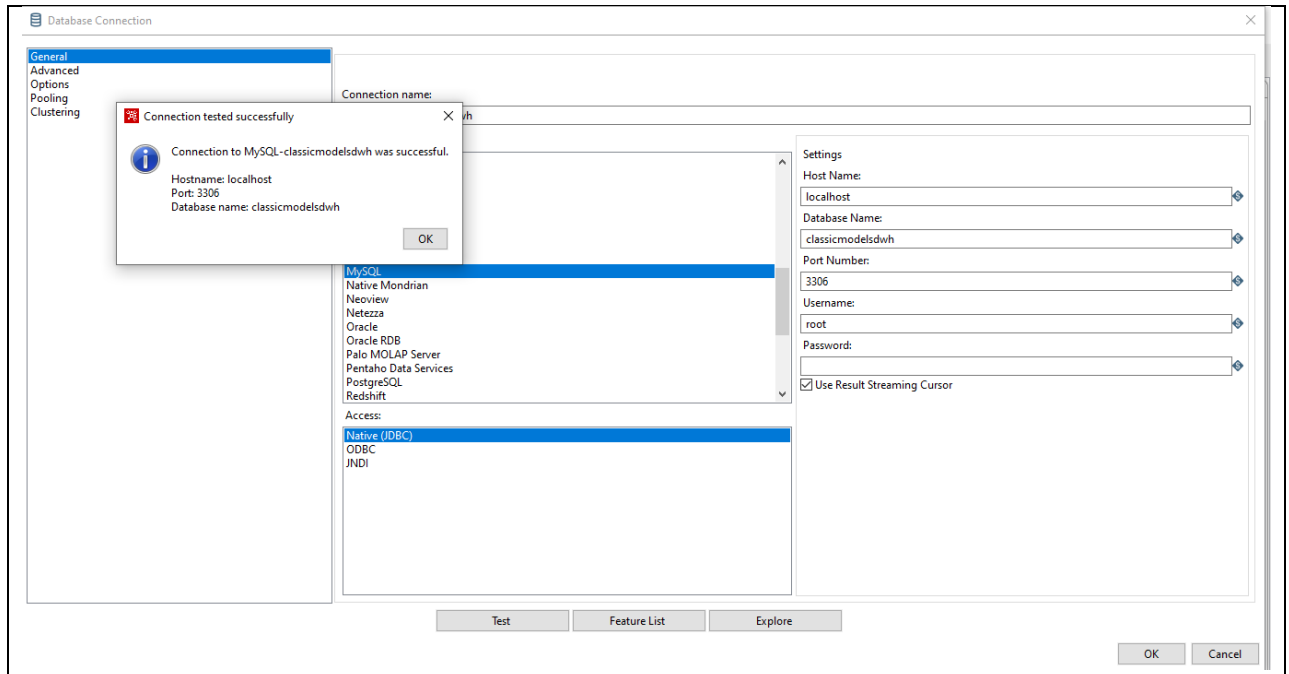
5. Get System Info.





6. Insert / Update.





Insert / update

Step name: Insert / update

Connection: MySQL-classicmodelsdwh [Edit... New... Wizard...]

Target schema: classicmodelsdwh [Browse...]

Target table: dim_product [Browse...]

Commit size: 100

Don't perform any updates: ☐

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

#	Table field	Comparator	Stream field1	Stream field2
1	product_code	=	product_code	

[Get fields]

Update fields:

#	Table field	Stream field	Update
1	product_code	product_code	Y
2	product_name	product_name	Y
3	product_line	product_line	Y
4	product_vendor	product_vendor	Y
5	last_update	last_update	Y

[Get update fields] [Edit mapping]

[? Help] [OK] [Cancel] [SQL]

7. Output ketika dijalankan/dirunning.

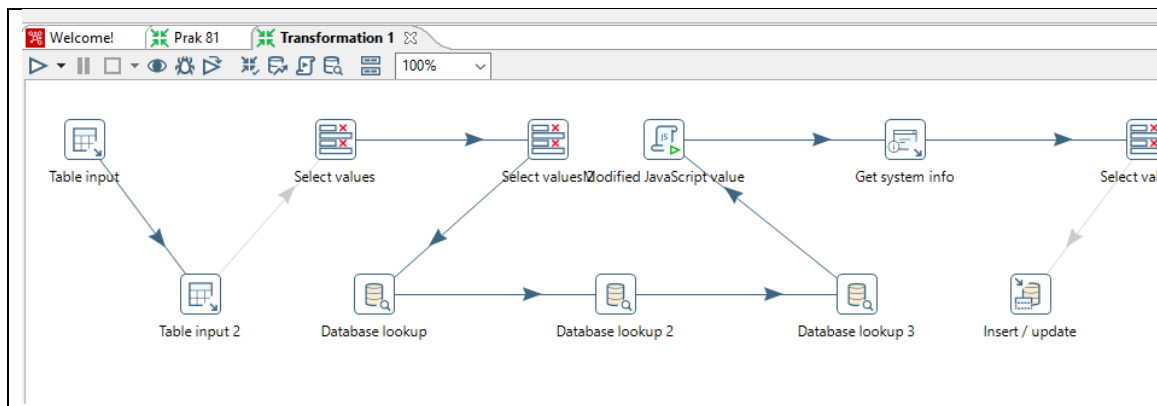


The screenshot shows the SAP Data Services Spoon interface. The left pane displays a catalog of components under 'Output', including 'Insert / update', 'LDAP output', 'SQL file output', 'Salesforce insert', 'Salesforce upsert', 'Synchronize after merge', 'Streaming', 'Transform', 'Utility', 'Flow', 'Scripting', 'Pentaho Server', 'Lookup', 'Joins', 'Data Warehouse', 'Validation', 'Statistics', 'Big Data', 'Agile', 'Cryptography', and 'Job'. The main canvas shows a data flow diagram with four components: 'Max Last Update', 'Product', 'Get system info', and 'Insert / update'. The 'Execution Results' pane at the bottom displays a table of data.

#	product_code	product_name	product_line	product_vendor	last_update
1	S10_1678	1969 Harley Davidson Ultimate Chopper	Motorcycles	Min Lin Diecast	2022/05/30 20:41:56.149
2	S10_1949	1952 Alpine Renault 1300	Classic Cars	Classic Metal Creations	2022/05/30 20:41:56.149
3	S10_2016	1996 Moto Guzzi 1100i	Motorcycles	Highway 66 Mini Classics	2022/05/30 20:41:56.149
4	S10_4698	2003 Harley-Davidson Eagle Drag Bike	Motorcycles	Red Start Diecast	2022/05/30 20:41:56.149
5	S10_4757	1972 Alfa Romeo GTA	Classic Cars	Motor City Art Classics	2022/05/30 20:41:56.149
6	S10_4962	1962 Lancia Delta 16V	Classic Cars	Second Gear Diecast	2022/05/30 20:41:56.149
7	S12_1099	1968 Ford Mustang	Classic Cars	Autoart Studio Design	2022/05/30 20:41:56.149

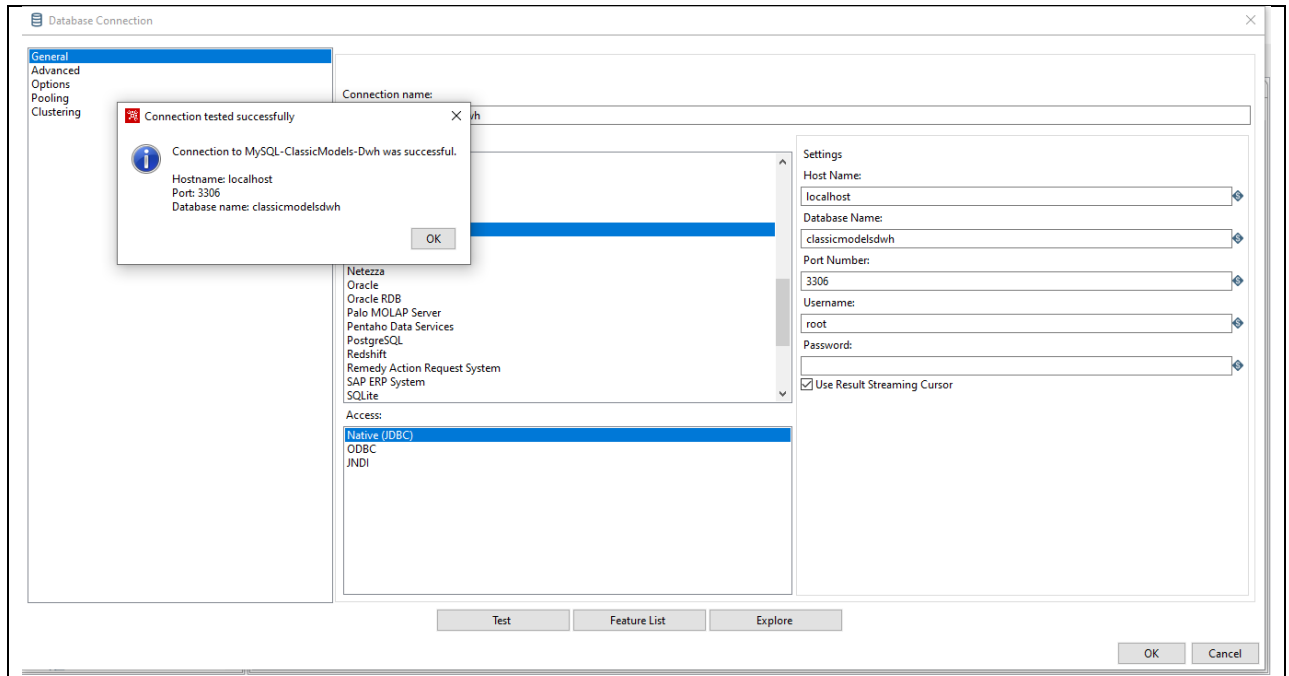
b. Latihan Kedua – Fact Order Transformation

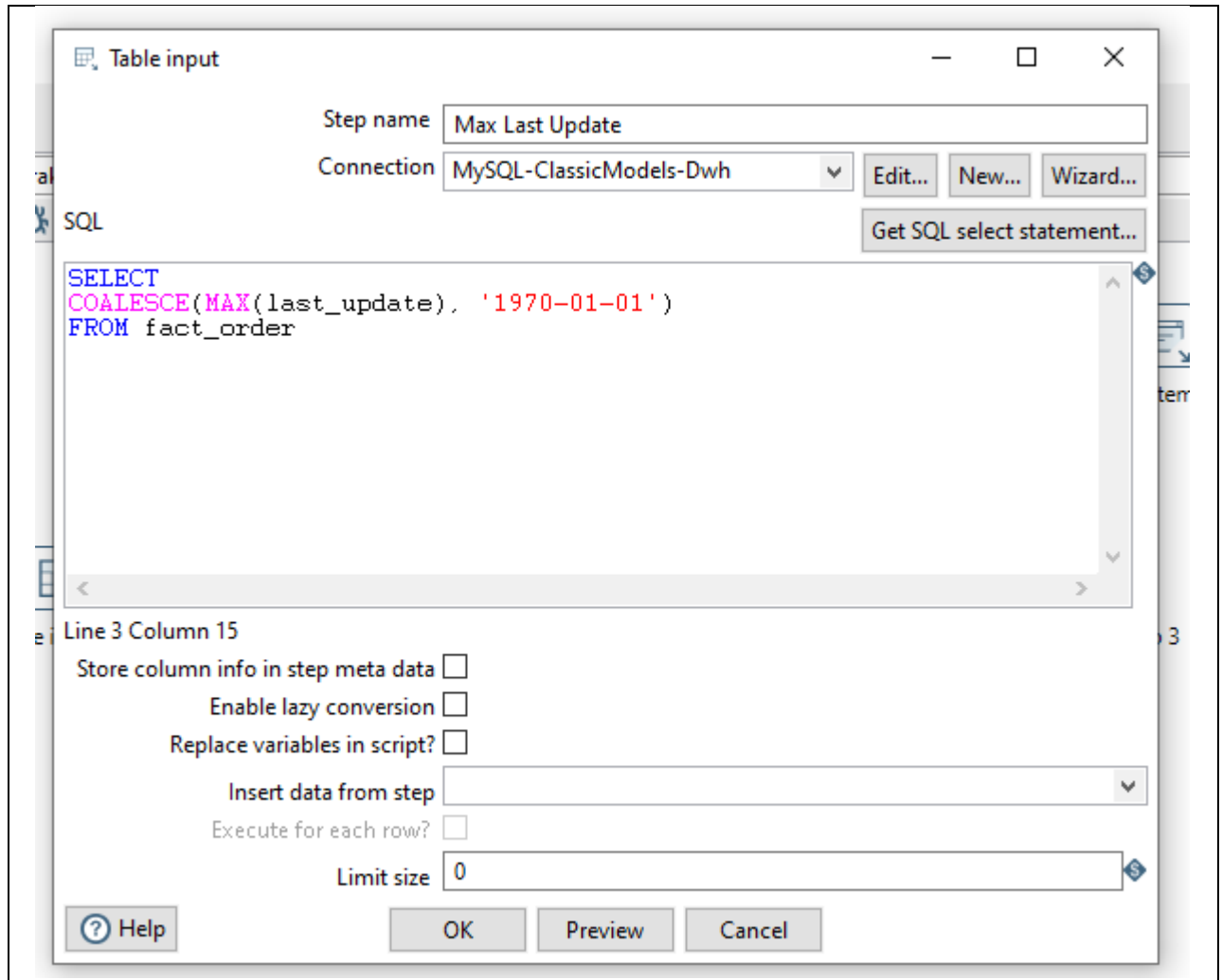
1. Struktur Fact Order.



2. Max Last Update – Table input







3. Order – Table input.



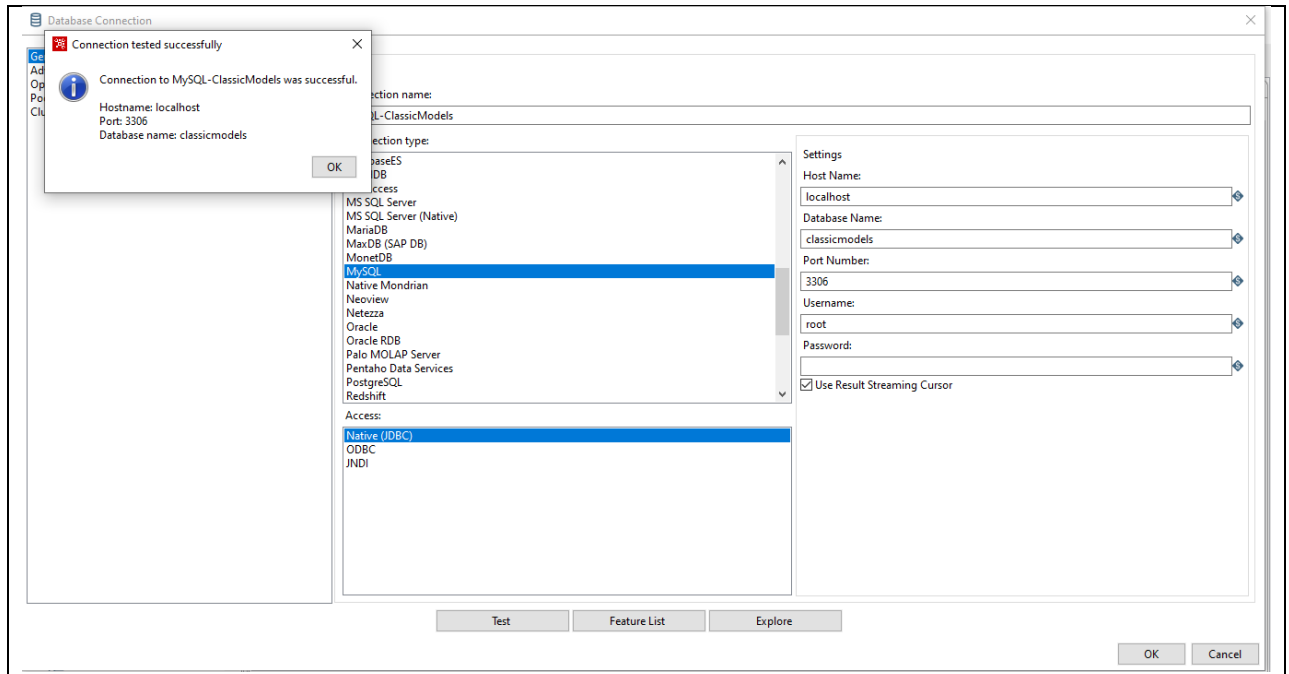


Table input

Step name: Order

Connection: MySQL-ClassicModels

SQL

```
SELECT od.orderLineNumber AS order_line_number,
od.orderNumber AS order_number,
od.productCode AS product_code,
od.quantityOrdered AS quantity_ordered,
od.priceEach AS priceEach,
od.quantityOrdered * od.priceEach AS price_total,
o.orderDate AS order_date,
o.requiredDate AS required_date,
o.shippedDate AS shipped_date,
e.employeeNumber AS employee_number,
c.customerNumber AS customer_number
FROM orderdetails od
LEFT JOIN orders o ON o.orderNumber = od.orderNumber
LEFT JOIN customers c ON c.customerNumber = o.customerNumber
LEFT JOIN employees e ON e.employeeNumber = c.salesRepEmployeeNumber
WHERE o.orderDate > ?
ORDER BY od.orderNumber, od.orderLineNumber
```

Line 17 Column 43

Store column info in step meta data ☐

Enable lazy conversion ☐

Replace variables in script? ☒

Insert data from step: Max Last Update

Execute for each row? ☐

Limit size: 0

Help OK Preview Cancel

4. Get Data Keys String – Select values.



The image displays two screenshots of a 'Select values' window, likely from a data integration tool like Talend. Both windows have a 'Step name' of 'Get Date Keys String' and tabs for 'Select & Alter', 'Remove', and 'Meta-data'.

Top Screenshot: Fields

#	Fieldname	Rename to	Length	Precision
1	order_line_number			
2	order_number			
3	product_code			
4	quantity_ordered			
5	price_each			
6	price_total			
7	order_date			
8	required_date			
9	shipped_date			
10	employee_number			
11	customer_number			

Buttons on the right: 'Get fields to select', 'Edit Mapping'.

Bottom Screenshot: Fields to alter the meta-data for

#	Fieldname	Rename to	Type	Length	Precision	Binary to Normal?	Format	Date Form
1	order_date		String			N		
2	required_date		String			N		
3	shipped_date		String			N		

Button on the right: 'Get fields to change'.

5. Select values



Select values

Step nameSelect values 2

Select & AlterRemoveMeta-data

Fields :

#	Fieldname	Rename to	Length	Precision
1	order_line_number			
2	order_number			
3	product_code			
4	quantity_ordered			
5	price_each			
6	price_total			
7	order_date			
8	required_date			
9	shipped_date			
10	employee_number			
11	customer_number			

Get fields to selectEdit Mapping

Select values

Step nameSelect values 2

Select & AlterRemoveMeta-data

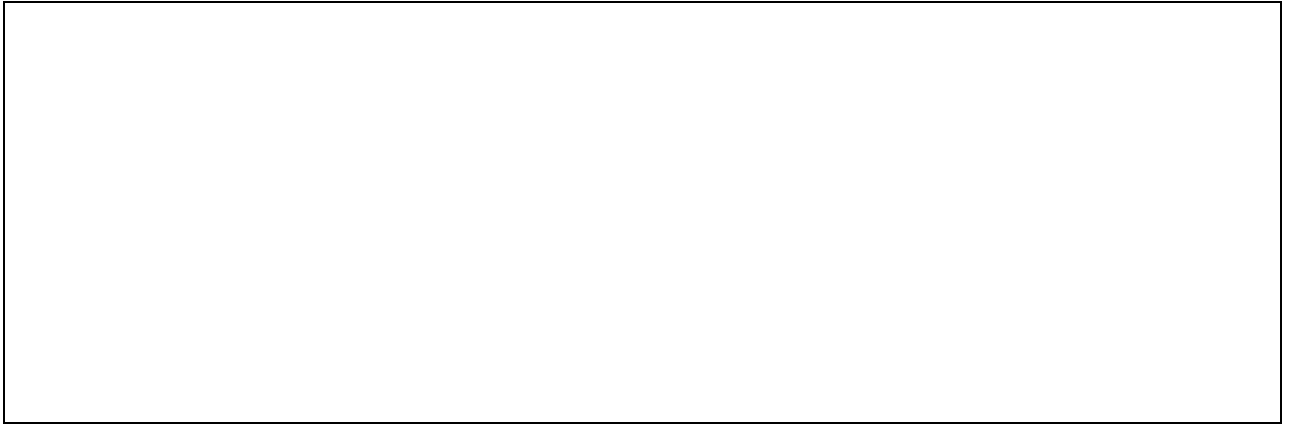
Fields to alter the meta-data for :

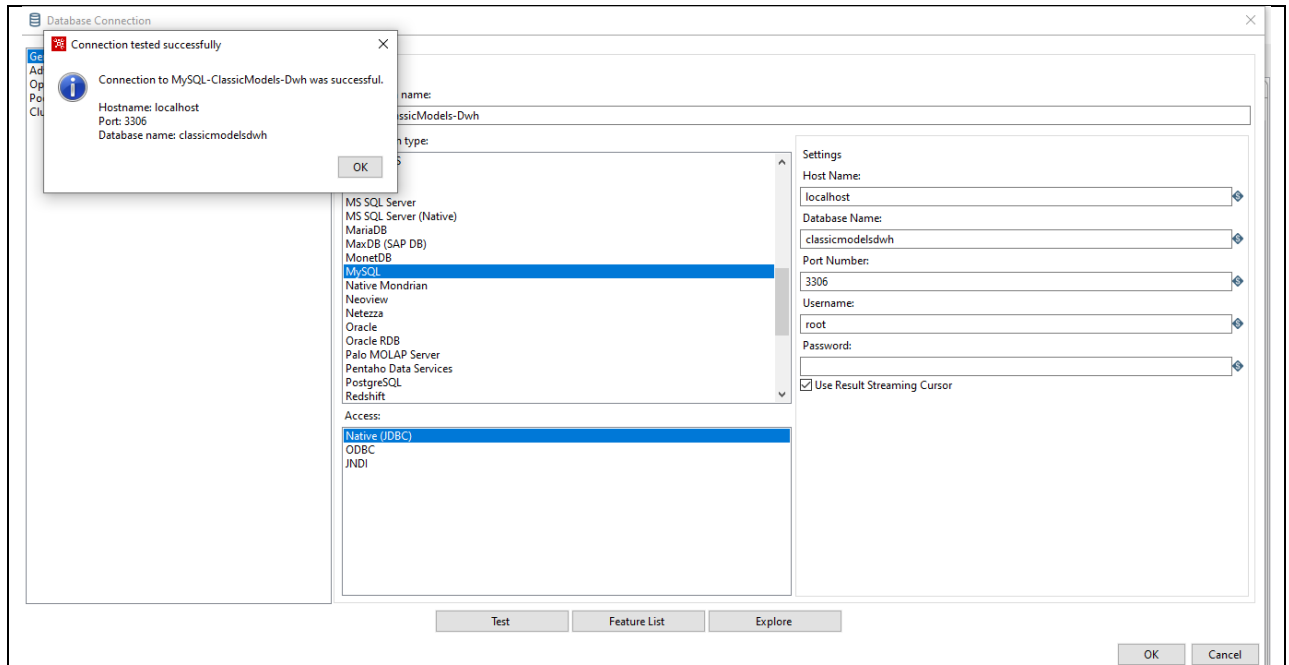
#	Fieldname	Rename to	Type	Length	Precision	Binary to Normal?	Format	Date
1	order_date	order_date_sk	Integer			N		
2	required_date	required_date_sk	Integer			N		
3	shipped_date	shipped_date_sk	Integer			N		

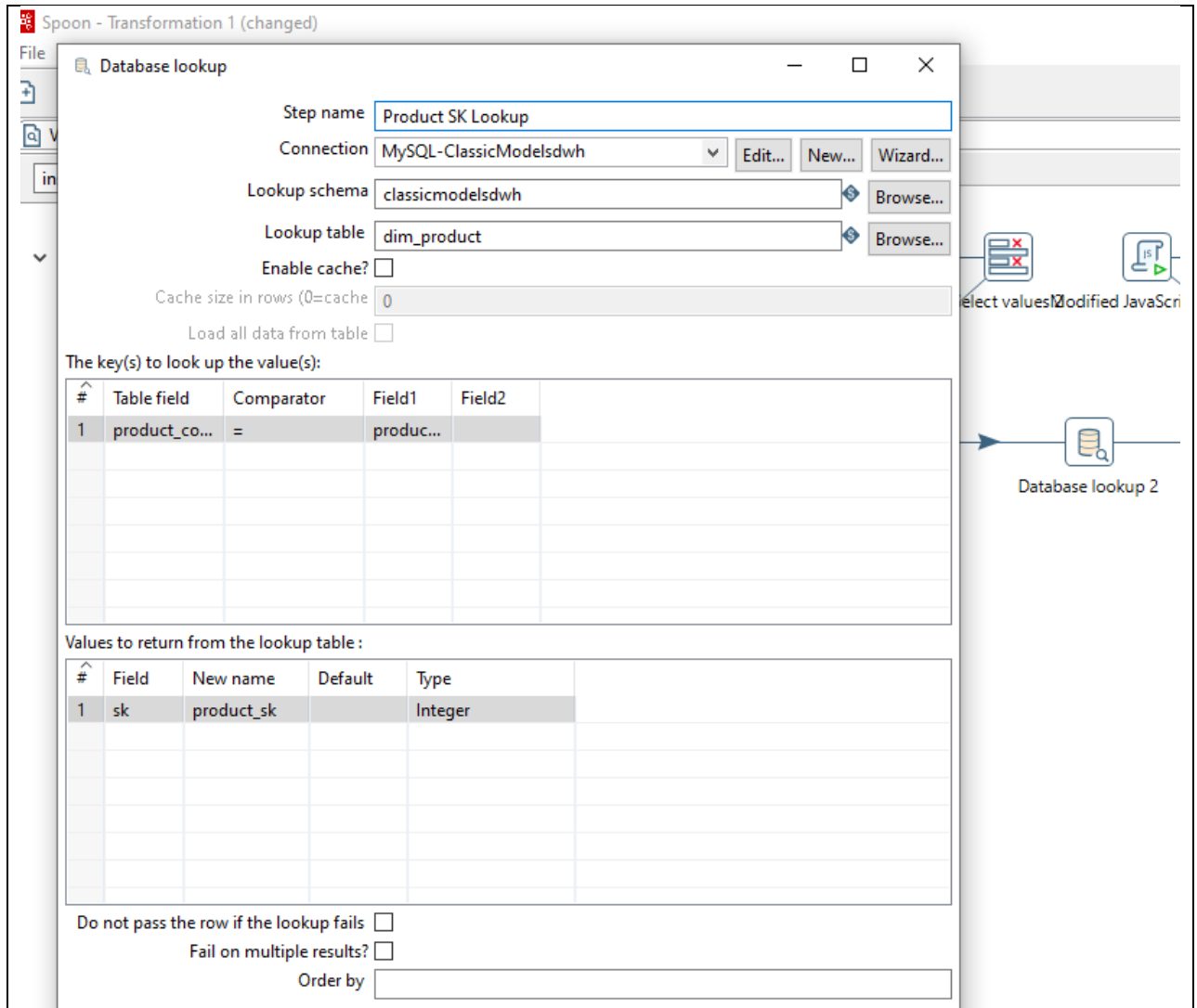
Get fields to change

6. Product SK Lookup – Database Lookup.



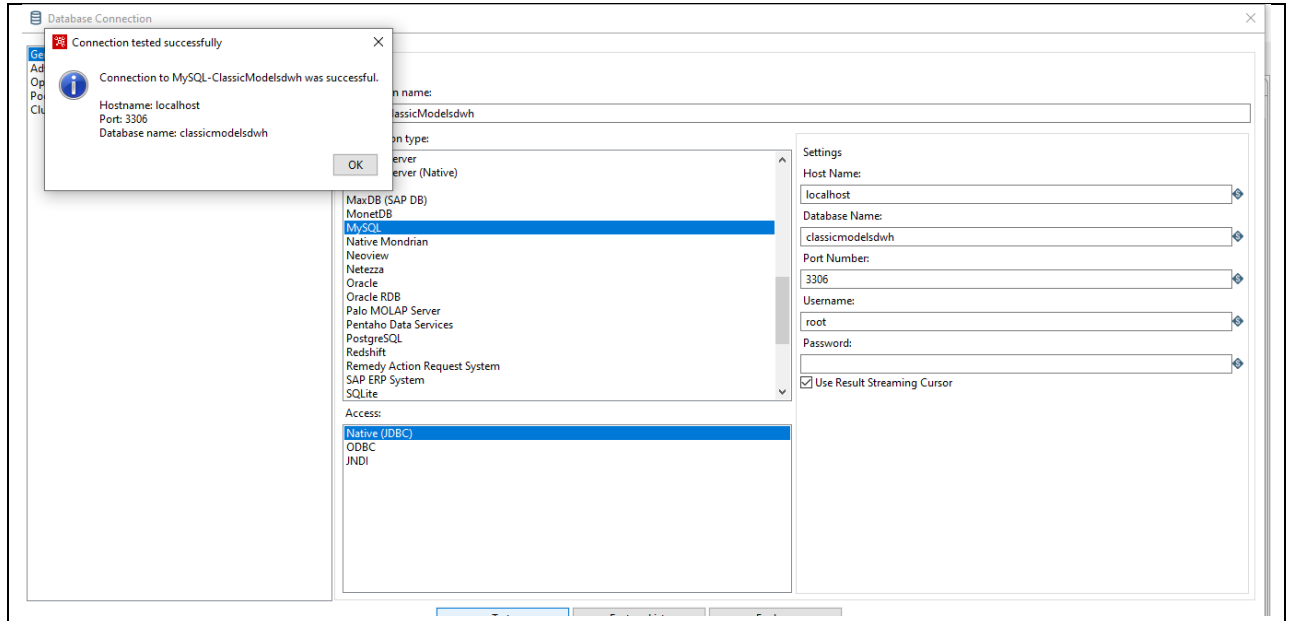






7. Costumer SK Lookup – Database Lookup.





Database lookup

Step name: Customer SK Lookup 2

Connection: MySQL-Classicmodels_dwh [Edit... New... Wizard...]

Lookup schema: classicmodelsdwh [Browse...]

Lookup table: dim_customer [Browse...]

Enable cache? ☐

Cache size in rows (0=cache): 0

Load all data from table ☐

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

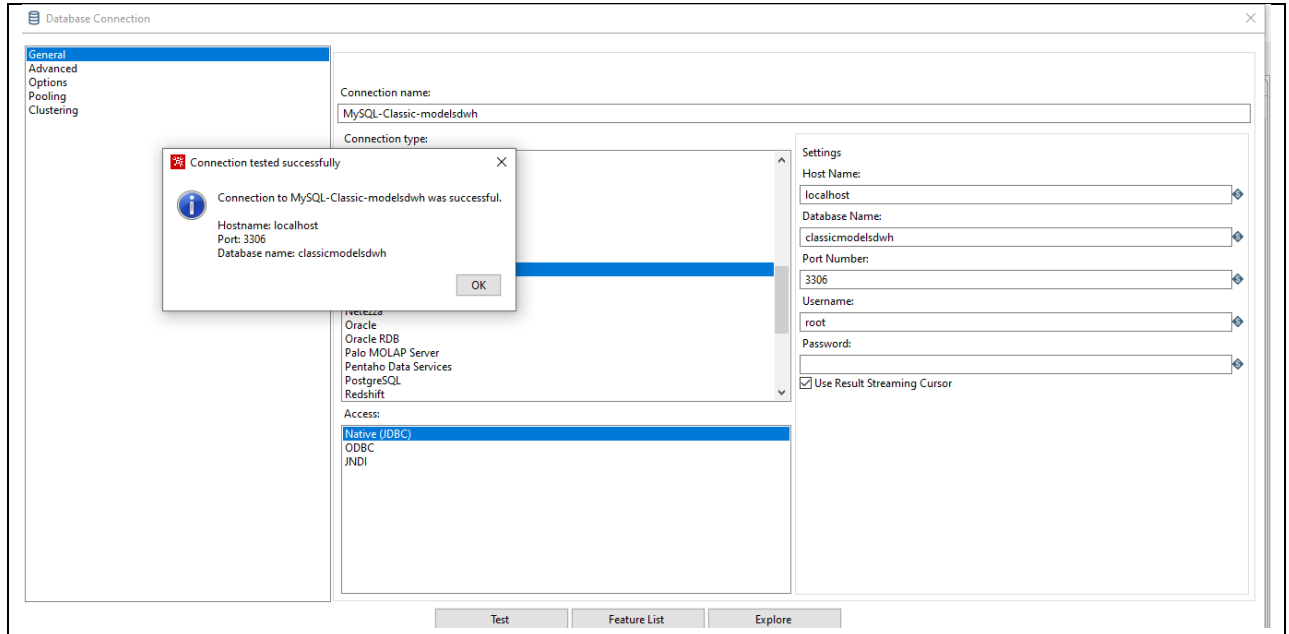
#	Table field	Comparator	Field1	Field2
1	customer_...	=	customer_number	

Values to return from the lookup table:

#	Field	New name	Default	Type
1	sk	customer_sk		Integer

8. Employee SK Lookup – Database Lookup.





Database lookup

Step name: Employee SK Lookup 3

Connection: MySQL-Classic-modelsdwh [Edit... New... Wizard...]

Lookup schema: classicmodelsdwh [Browse...]

Lookup table: dim_employee [Browse...]

Enable cache? ☐

Cache size in rows (0=cache): 0

Load all data from table ☐

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

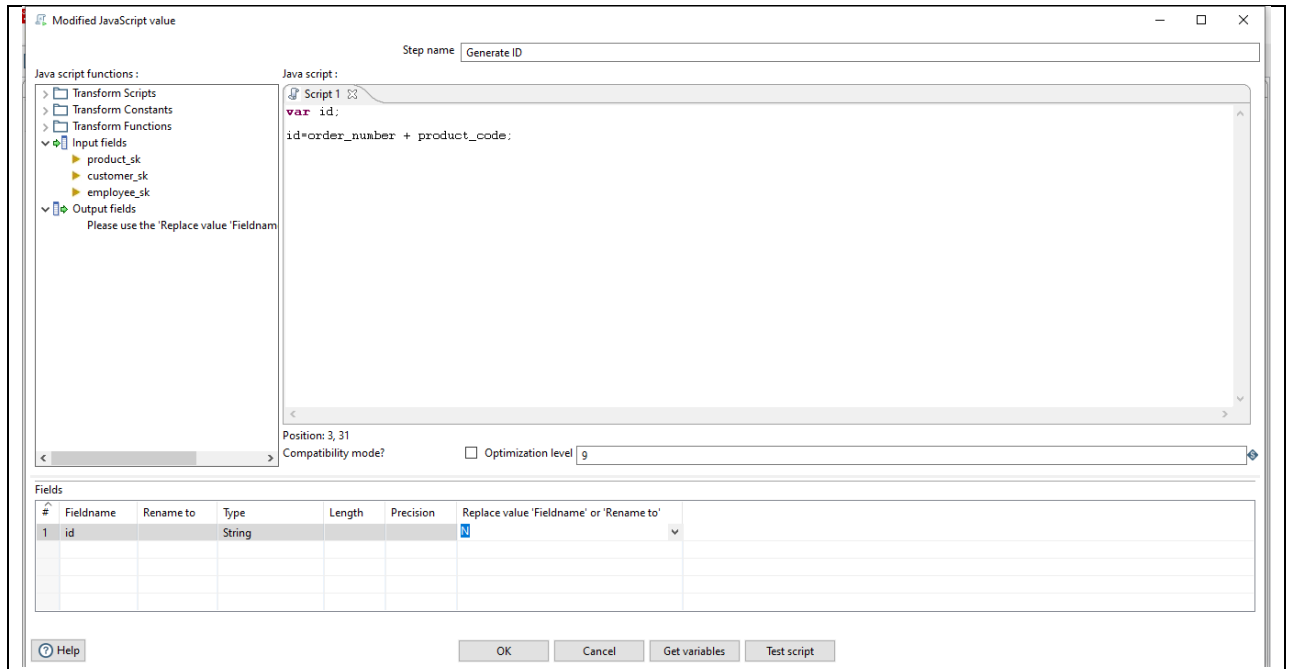
#	Table field	Comparator	Field1	Field2
1	employee_...	=	emplo...	

Values to return from the lookup table:

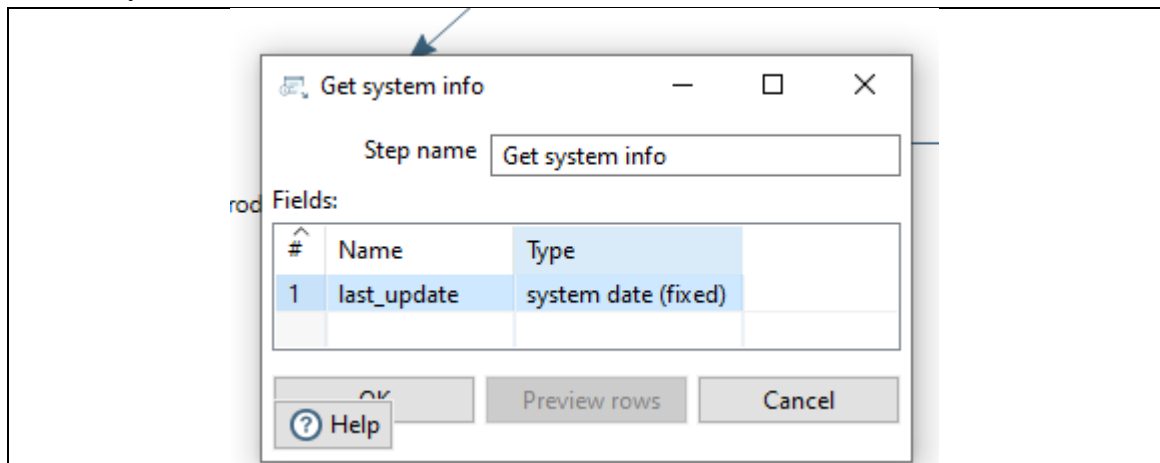
#	Field	New name	Default	Type
1	sk	employee_sk		Integer

9. Generate ID – Modified JavaScript value.



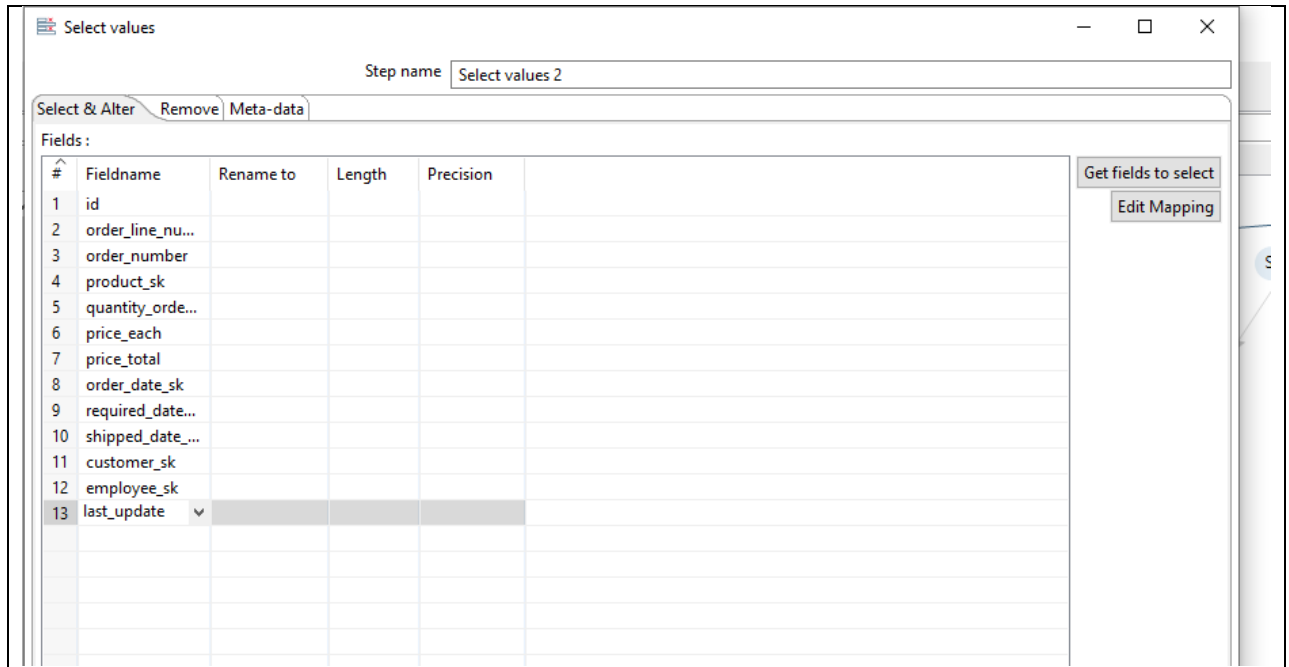


10. Get System Info.



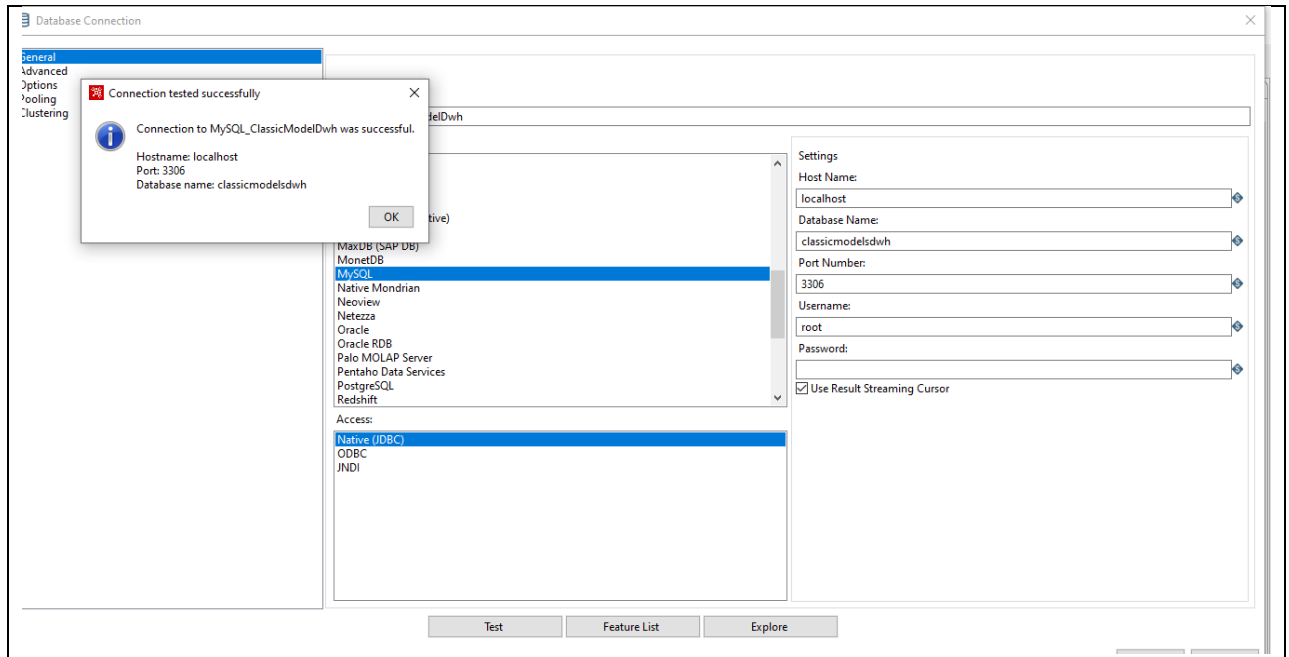
11. Select values 2.





12. Insert / Update.





Insert / update

Step name: Insert / update

Connection: MySQL_ClassicModelDwh [Edit... New... Wizard...]

Target schema: classicmodelsdwh [Browse...]

Target table: fact_order [Browse...]

Commit size: 100

Don't perform any updates: ☐

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

#	Table field	Comparator	Stream field1	Stream field2
1	id	=	id	

[Get fields]

Update fields:

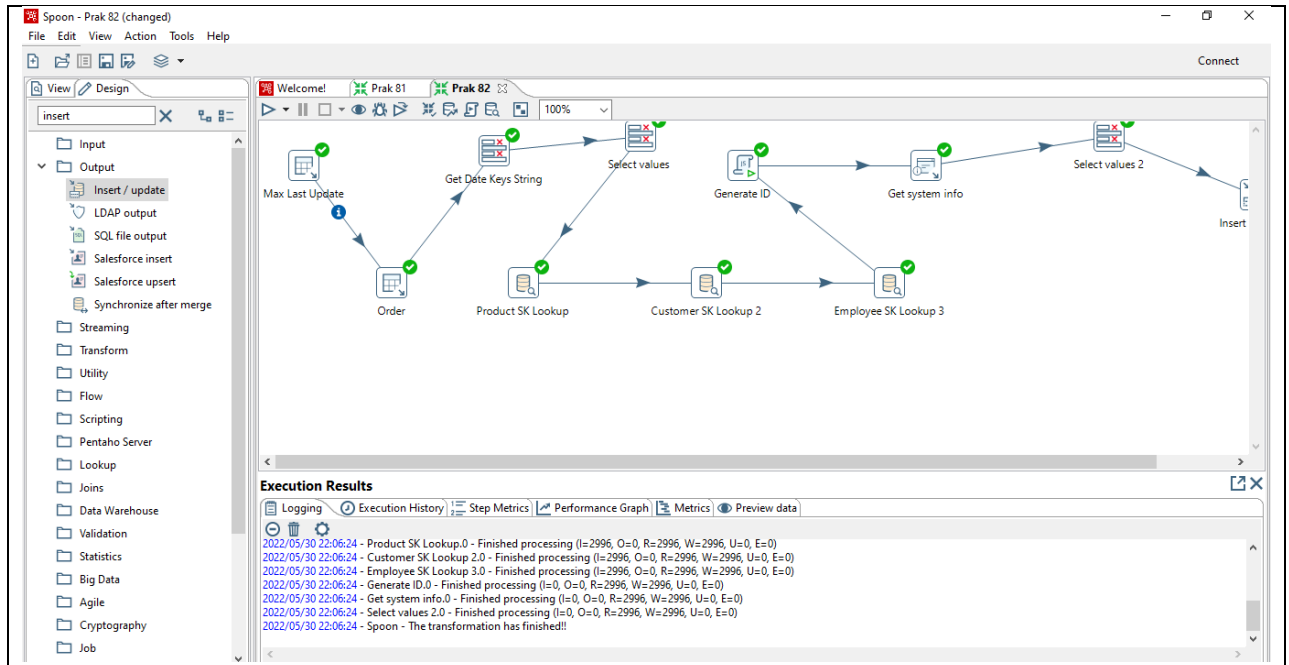
#	Table field	Stream field	Update
1	id	id	Y
2	order_line_number	order_line_number	Y
3	order_number	order_number	Y
4	product_sk	product_sk	Y
5	quantity_ordered	quantity_ordered	Y
6	price_each	price_each	Y
7	price_total	price_total	Y
8	order_date_sk	order_date_sk	Y
9	required_date_sk	required_date_sk	Y
10	shipped_date_sk	shipped_date_sk	Y
11	customer_sk	customer_sk	Y
12	employee_sk	employee_sk	Y
13	last_update	last_update	Y

[Get update fields] [Edit mapping]

[?] Help [OK] [Cancel] [SQL]

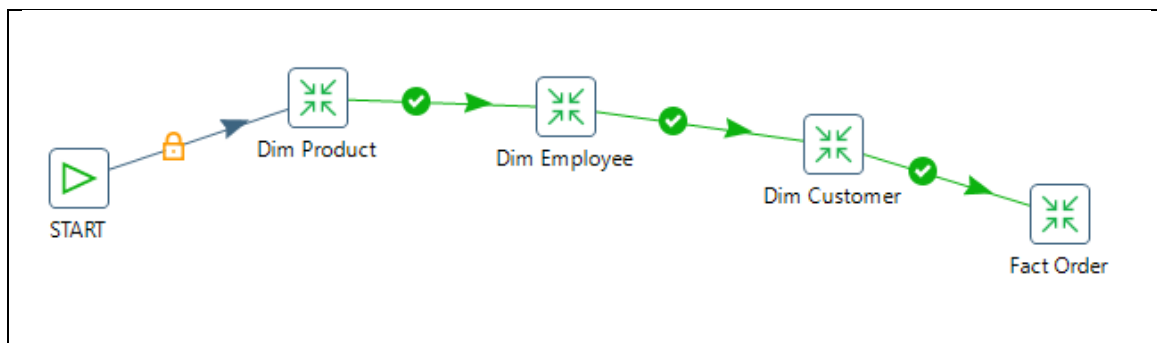
12. Output.



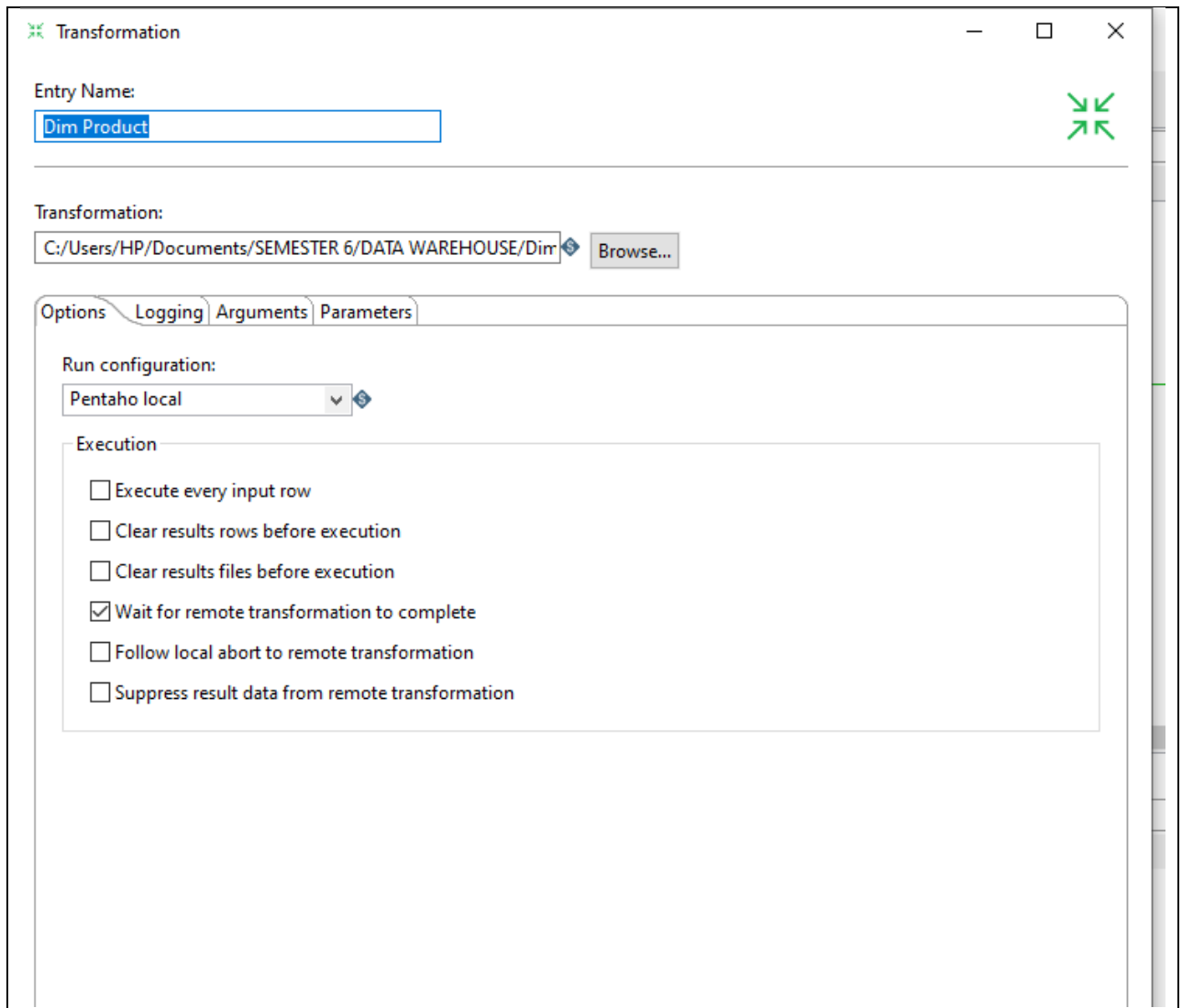


c. Latihan Ketiga – Order Job

1. Buatlah Job baru dan simpan dengan nama Order. Lalu buatlah struktur seperti gambar dibawah ini.



2. Dim Product – Transformation.



The screenshot shows the 'Transformation' configuration window in Pentaho. The 'Entry Name' is 'Dim Product'. The 'Transformation' path is 'C:/Users/HP/Documents/SEMESTER 6/DATA WAREHOUSE/Dim'. The 'Options' tab is selected, showing the 'Run configuration' dropdown set to 'Pentaho local'. Under the 'Execution' section, the following options are listed:

- ☐ Execute every input row
- ☐ Clear results rows before execution
- ☐ Clear results files before execution
- ☒ Wait for remote transformation to complete
- ☐ Follow local abort to remote transformation
- ☐ Suppress result data from remote transformation

3. Dim Employee – Transformation.



The screenshot shows the 'Transformation' configuration window in Pentaho. The 'Entry Name' is 'Dim Employee'. The 'Transformation' path is 'C:/Users/HP/Documents/SEMESTER 6/DATA WAREHOUSE/Dimr'. The 'Options' tab is selected, showing 'Run configuration' set to 'Pentaho local'. Under the 'Execution' section, the following options are listed:

- ☐ Execute every input row
- ☐ Clear results rows before execution
- ☐ Clear results files before execution
- ☒ Wait for remote transformation to complete
- ☐ Follow local abort to remote transformation
- ☐ Suppress result data from remote transformation

4. Dim Customer – Transformation.



Transformation

Entry Name:

Transformation:

Options | Logging | Arguments | Parameters

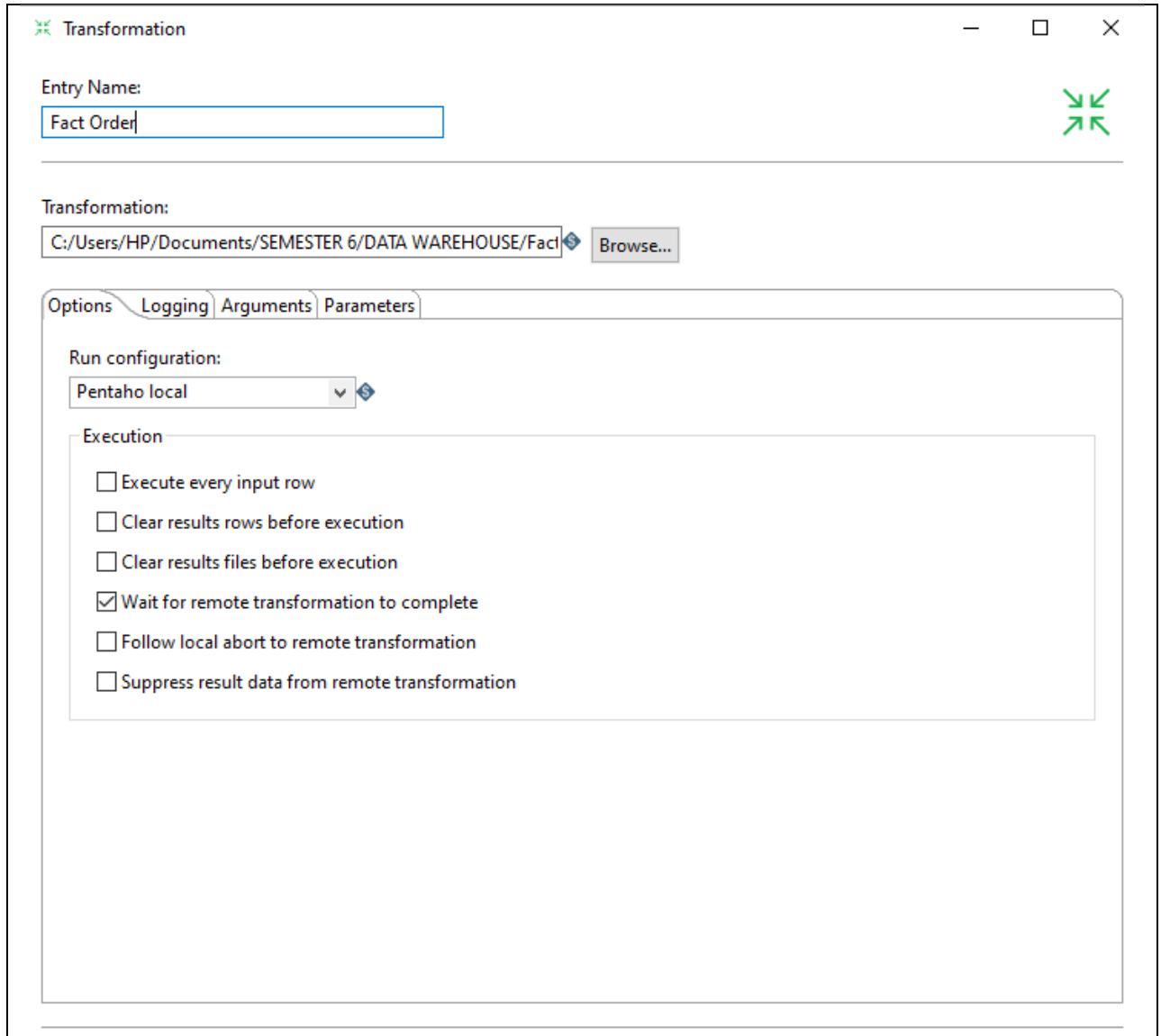
Run configuration:

Execution

- ☐ Execute every input row
- ☐ Clear results rows before execution
- ☐ Clear results files before execution
- ☒ Wait for remote transformation to complete
- ☐ Follow local abort to remote transformation
- ☐ Suppress result data from remote transformation

5. Fact Order – Transformation.





The screenshot shows the 'Transformation' configuration window in Pentaho. The 'Entry Name' field is set to 'Fact Order'. The 'Transformation' field shows a file path 'C:/Users/HP/Documents/SEMESTER 6/DATA WAREHOUSE/Fact' with a 'Browse...' button. Below this is a tabbed interface with 'Options', 'Logging', 'Arguments', and 'Parameters' tabs. The 'Options' tab is active, showing 'Run configuration' set to 'Pentaho local'. Under the 'Execution' section, there are several checkboxes: 'Execute every input row', 'Clear results rows before execution', 'Clear results files before execution', 'Wait for remote transformation to complete' (which is checked), 'Follow local abort to remote transformation', and 'Suppress result data from remote transformation'.

Transformation

Entry Name:
Fact Order

Transformation:
C:/Users/HP/Documents/SEMESTER 6/DATA WAREHOUSE/Fact Browse...

Options Logging Arguments Parameters

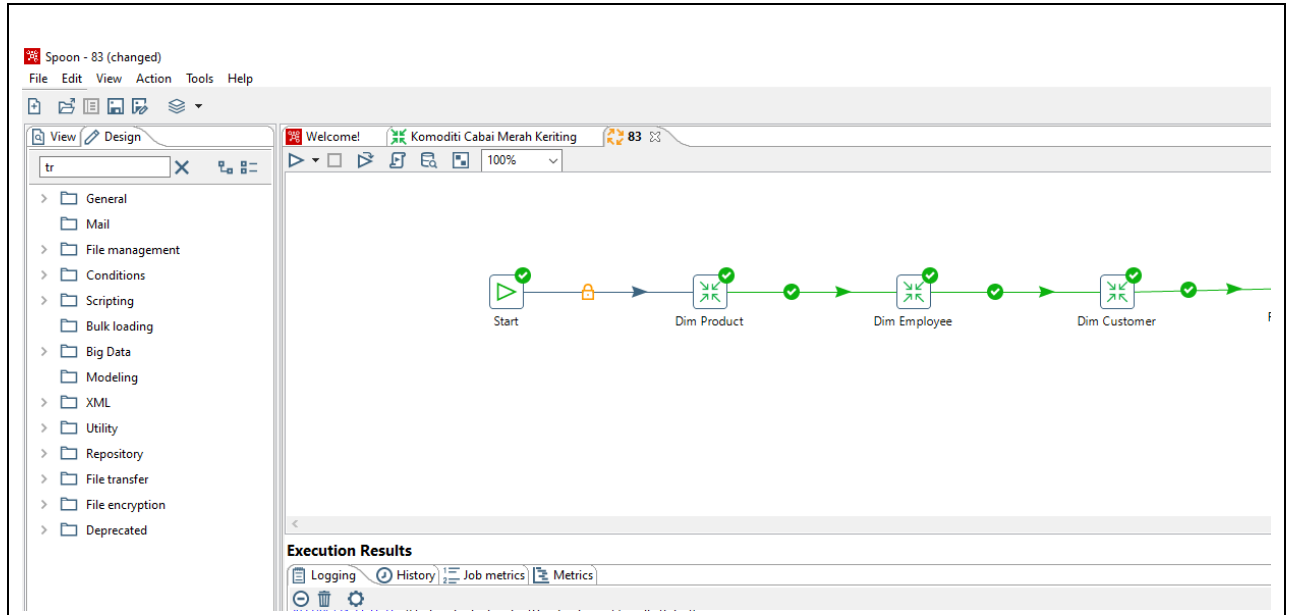
Run configuration:
Pentaho local

Execution

- ☐ Execute every input row
- ☐ Clear results rows before execution
- ☐ Clear results files before execution
- ☒ Wait for remote transformation to complete
- ☐ Follow local abort to remote transformation
- ☐ Suppress result data from remote transformation

6. Output





7. Output Pada Database

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
<input type="checkbox"/> dim_customer	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	122	InnoDB	utf8mb4_general_ci	112 KB	-
<input type="checkbox"/> dim_date	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	10,000	InnoDB	utf8mb4_general_ci	2.4 MB	-
<input type="checkbox"/> dim_employee	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	23	InnoDB	utf8mb4_general_ci	80 KB	-
<input type="checkbox"/> dim_product	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	110	InnoDB	utf8mb4_general_ci	80 KB	-
<input type="checkbox"/> fact_order	Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	112 KB	-
5 tabel	Jumlah	10,255	InnoDB	utf8mb4_general_ci	2.8 MB	0 B

4. File Praktikum

Github Repository:

<https://github.com/rasyahdz/PraktikumDWH.git>

5. Soal Latihan



Jurusan Teknik Informatika & Sistem Informasi
Fakultas Teknologi Industri – Universitas Trisakti

Soal:

1. Apa perbedaan OLAP dengan OLTP?
2. Apa yang dimaksud dengan ETL?

Jawaban:

1. **OLTP** adalah pemrosesan transaksional sedangkan **OLAP** adalah sistem pemrosesan analitis.
2. **ETL** adalah singkatan dari extract, transform, dan load. Melansir IBM, ia merupakan proses integrasi data. Di sana, data akan dikombinasikan dari berbagai sumber. Setelah itu, mereka disimpan di tempat bernama data warehouse.

6. Kesimpulan

OLTP adalah sistem yang mengelola aplikasi berorientasi transaksi di internet misalnya ATM. **OLAP** adalah sistem online yang melaporkan ke kueri analitik multidimensi seperti pelaporan keuangan, peramalan, dll.

7. Cek List (✓)

No	Elemen Kompetensi	Penyelesaian	
		Selesai	Tidak Selesai
1.	Latihan Pertama	...	
2.	Latihan Kedua	...	
3.	Latihan Ketiga	...	

8. Formulir Umpan Balik

No	Elemen Kompetensi	Waktu Pengerjaan	Kriteria
1.	Latihan Pertama	... Menit	...
2.	Latihan Kedua	... Menit	...
3.	Latihan Ketiga	... Menit	...

Keterangan:

1. Menarik
2. Baik
3. Cukup
4. Kurang

