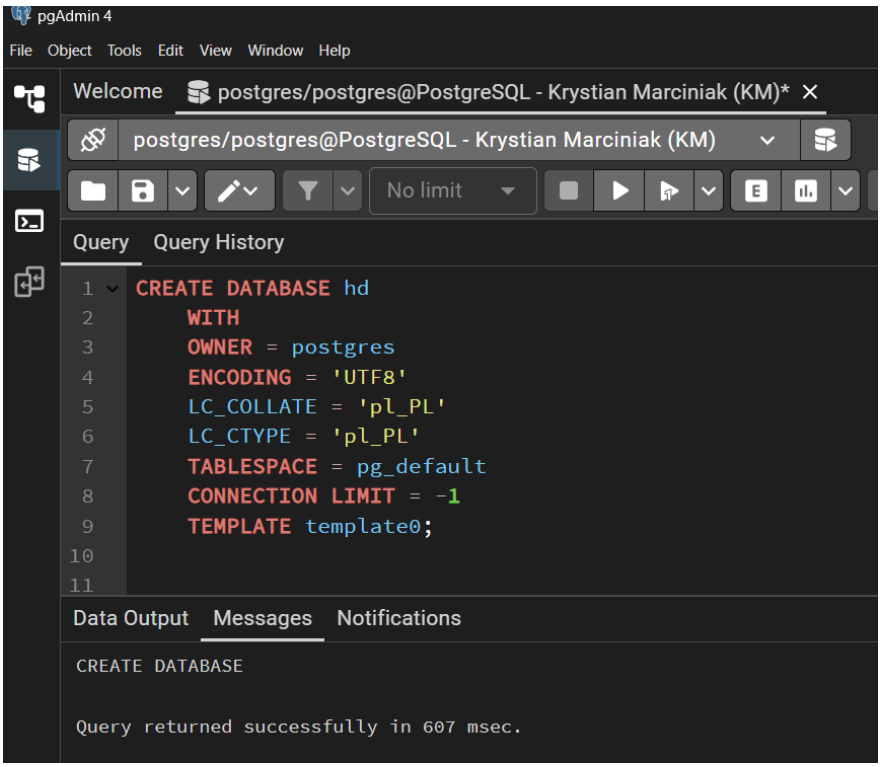
ETAP I – Projektowanie Schematów (Modelowanie)

1. Utwórz pustą bazę danych HD w PostgreSQL.

```
CREATE DATABASE hurtownia
    WITH
    OWNER = postgres
    ENCODING = 'UTF8'
    LC_COLLATE = 'pl_PL'
    LC_CTYPE = 'pl_PL'
    TABLESPACE = pg_default
    CONNECTION LIMIT = -1
    TEMPLATE template0;
```



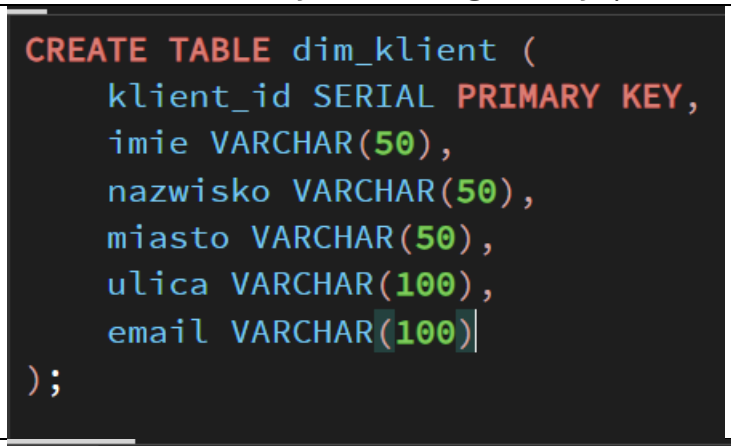
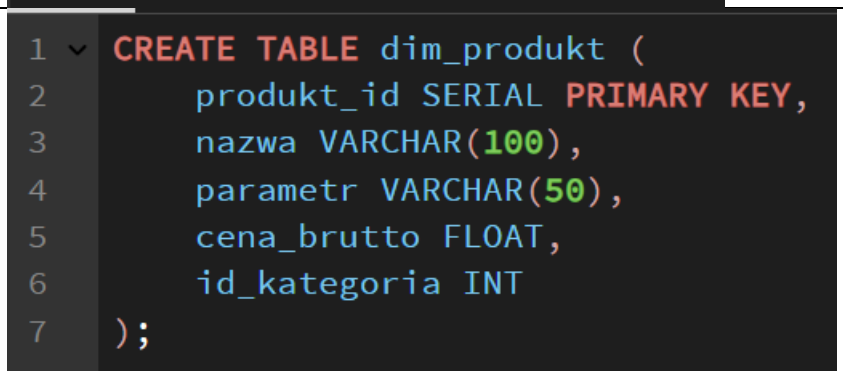
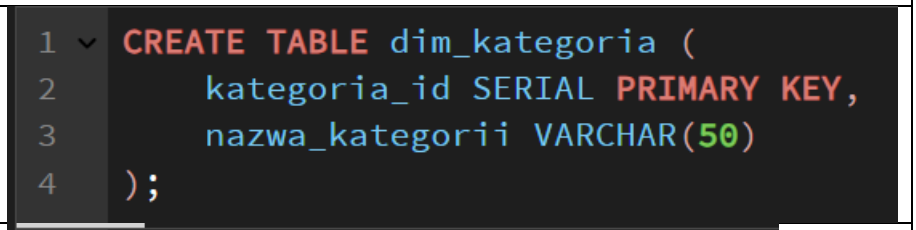
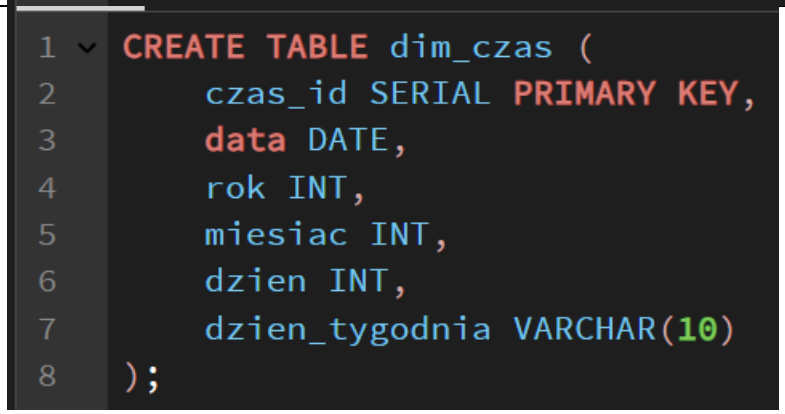
Składniki polecenia i ich znaczenie:	
CREATE DATABASE hurtownia	Tworzy nową bazę danych o nazwie **hurtownia.**
WITH	Wprowadza opcje konfiguracyjne dla tworzonej bazy.
OWNER = postgres	Określa właściciela bazy danych - użytkownika PostgreSQL, który będzie miał pełne uprawnienia (np. **postgres**).
ENCODING = 'UTF8'	Ustawia kodowanie znaków w bazie danych. UTF8 pozwala przechowywać znaki z różnych języków (w tym polskie litery).
LC_COLLATE = 'pl_PL'	Określa reguły sortowania tekstu według języka polskiego (np. **litera "ś" po "s"**).
LC_CTYPE = 'pl_PL'	Określa **klasyfikację znaków** (czy znak to litera, wielka litera, cyfra itd.), też wg języka polskiego.
TABLESPACE = pg_default	Określa domyślne miejsce przechowywania danych na dysku – tu pg_default, czyli standardowy katalog.
CONNECTION LIMIT = -1	Oznacza **brak ograniczenia liczby połączeń** do tej bazy danych (może się łączyć dowolna liczba użytkowników).
TEMPLATE template0	Tworzy bazę **na podstawie czystego szablonu** template0, który umożliwia zmianę LC_COLLATE i LC_CTYPE.

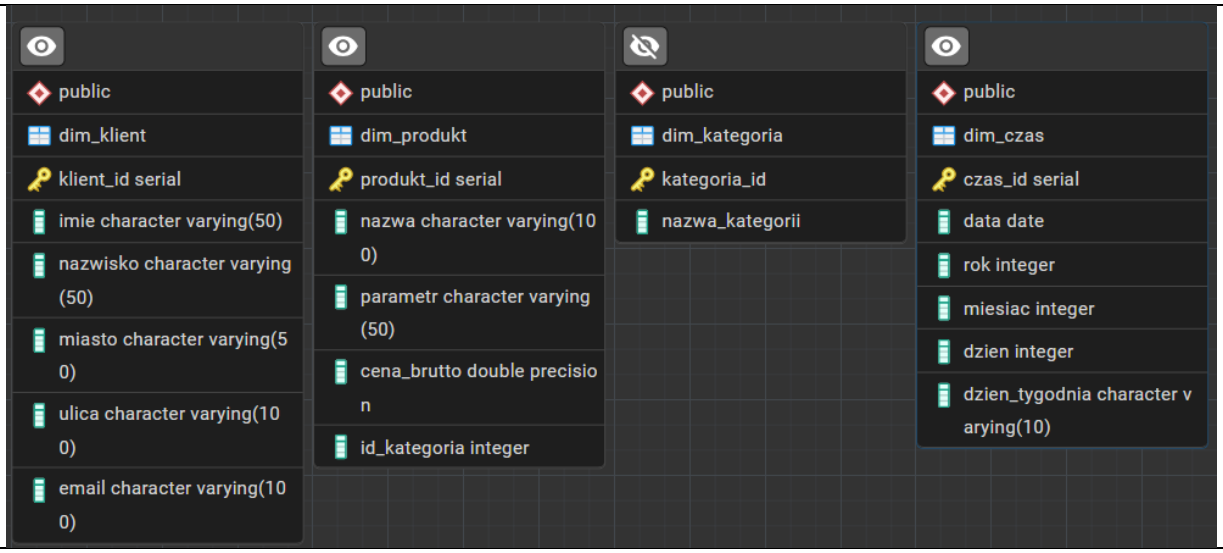
1. Przeanalizuj plik CSV i zaprojektuj architekturę gwiazdy (Star Schema) obejmującą wszystkie dane:

a Wyodrębnij tabelę faktów (np. sprzedaż, zamówienia) oraz tabele wymiarów (np. produkt, klient, czas, region).

-- Tabele wymiarów (z kluczami sztucznymi - surrogate keys)

3. Dla tabel wymiarów stosuj sztuczne klucze (surrogate keys) typu INT lub SERIAL.

-- Tabele wymiarów (z kluczami sztucznymi - surrogate keys)	
CREATE TABLE dim_klient (klient_id SERIAL PRIMARY KEY, imie VARCHAR(50), nazwisko VARCHAR(50), miasto VARCHAR(50), ulica VARCHAR(100), email VARCHAR(100));	
CREATE TABLE dim_produkt (produkt_id SERIAL PRIMARY KEY, nazwa VARCHAR(100), parametr VARCHAR(50), cena_brutto FLOAT, id_kategoria INT);	
CREATE TABLE dim_kategoria (kategoria_id SERIAL PRIMARY KEY, nazwa_kategorii VARCHAR(50));	
CREATE TABLE dim_czas (czas_id SERIAL PRIMARY KEY, data DATE, rok INT, miesiac INT, dzien INT, dzien_tygodnia VARCHAR(10));	



SERIAL (typ pseudodanych) -> SERIAL to wygodna konstrukcja PostgreSQL, która:

- automatycznie tworzy **sekwencję** (SEQUENCE)

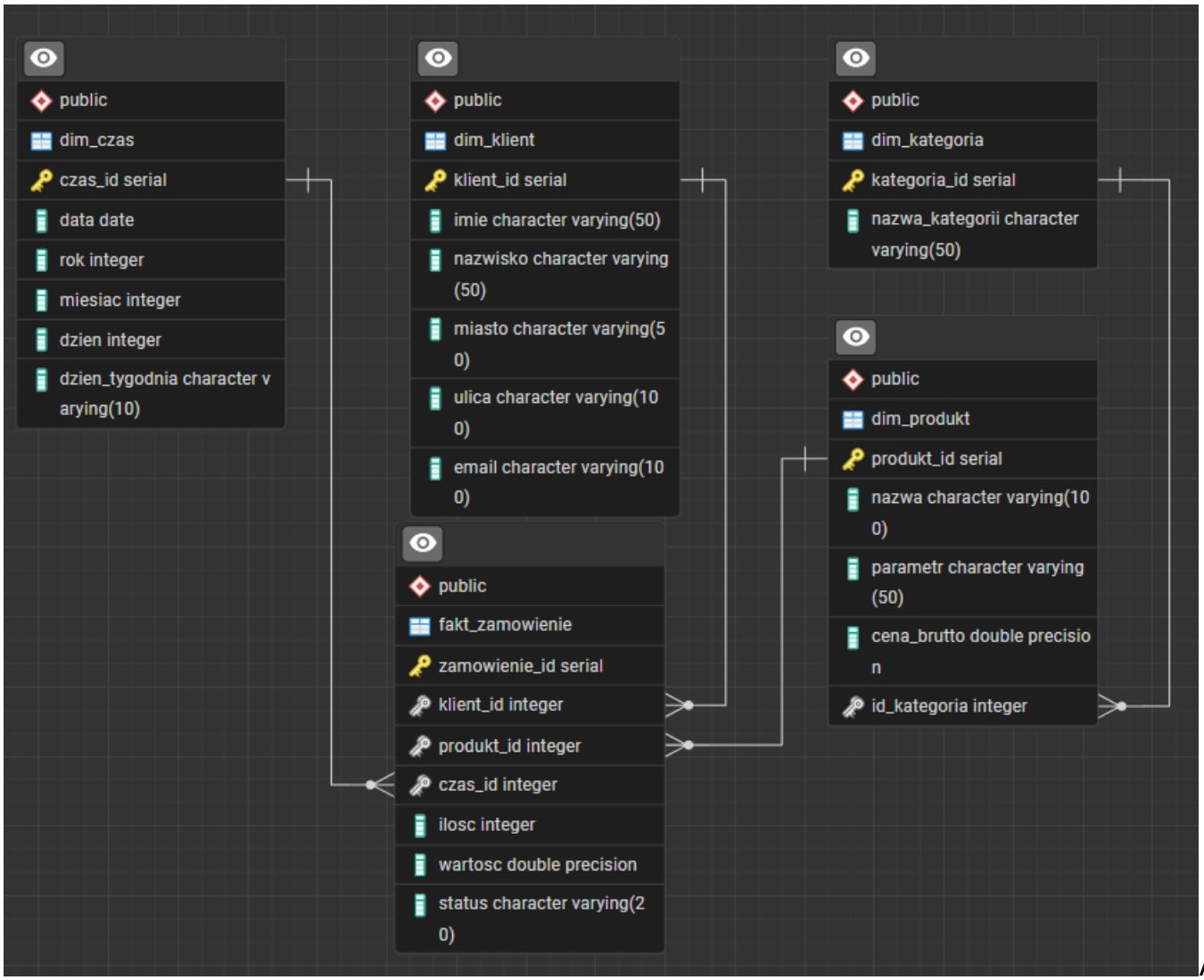
- przypisuje ją do kolumny jako DEFAULT nextval(...)

- działa jak auto_increment w MySQL

-- Tabela faktów (zamówienia)	

CREATE TABLE fakt_zamowienie
(zamowienie_id SERIAL PRIMARY KEY,
klient_id INT REFERENCES dim_klient(klient_id),
produkt_id INT REFERENCES dim_produkt(produkt_id),
czas_id INT REFERENCES dim_czas(czas_id),
ilosc INT NOT NULL,
wartosc FLOAT NOT NULL,
status VARCHAR(20)
);
	`CREATE TABLE fakt_zamowienie (`
	`zamowienie_id SERIAL PRIMARY KEY,`
	`klient_id INT REFERENCES dim_klient(klient_id),`
	`produkt_id INT REFERENCES dim_produkt(produkt_id),`
	`czas_id INT REFERENCES dim_czas(czas_id),`
	`ilosc INT NOT NULL,`
	`wartosc FLOAT NOT NULL,`
	`status VARCHAR(20)`
	`);`

ALTER TABLE dim_produkt
ADD FOREIGN KEY (id_kategoria)
REFERENCES dim_kategoria(kategoria_id);
	`ALTER TABLE dim_produkt`
	`ADD FOREIGN KEY (id_kategoria)`
	`REFERENCES dim_kategoria(kategoria_id);`



A.

Kolejność tworzenia tabel – problem zależności

Jeśli utworzone zostało dim_produkt przed dim_kategoria, wtedy odwołanie do dim_kategoria(kategoria_id) jeszcze nie zadziało - bo ta tabela nie istnieje. Rozdzielenie deklaracji pozwala:

- utworzyć najpierw wszystkie tabele

- potem dodać zależności między nimi

B. Czystość modelowania -> Projektanci czasem oddzielają strukturę tabeli od relacji logicznych, aby zachować przejrzystość:

CREATE TABLE → tylko struktura i kolumny

ALTER TABLE → relacje między tabelami (czyli logika)

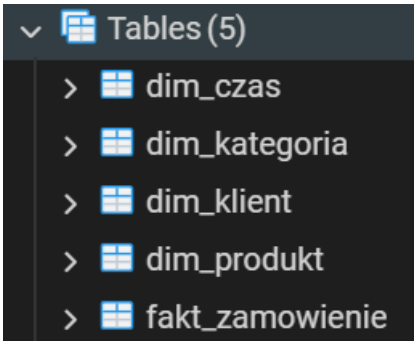
C. Etapowe budowanie schematu

W projektach z kilkoma etapami (np. najpierw modelowanie, potem implementacja relacji) często zostawia się FOREIGN KEY na później

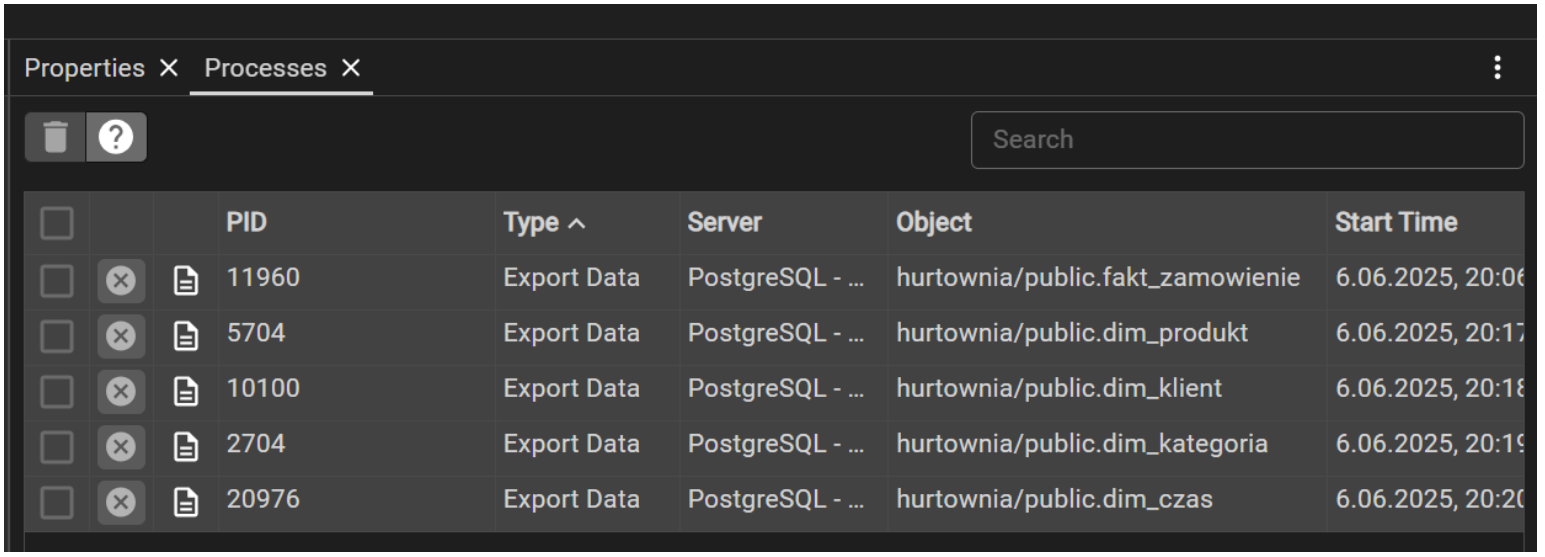
REFERENCES	definiuje klucz obcy (foreign key)
Chroni dane	przed odniesieniem do nieistniejących rekordów

Tworzy powiązania między tabelą faktów i tabelami wymiarów

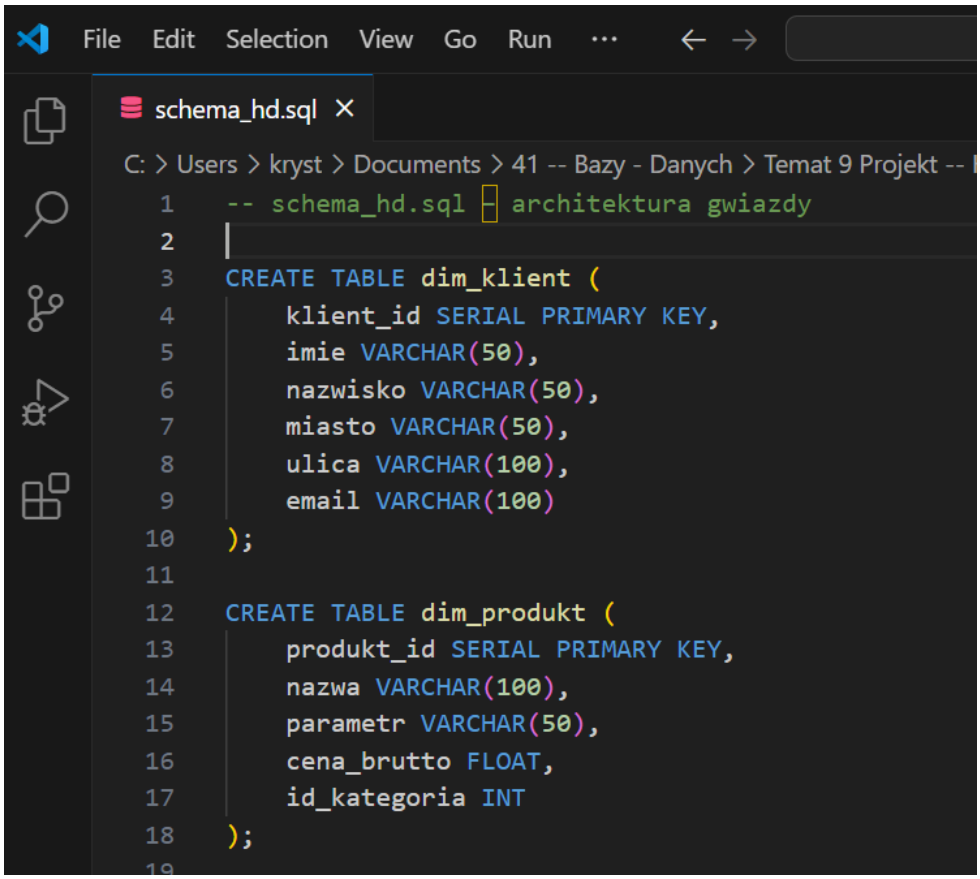
Ułatwia zapytania np. JOIN, GROUP BY

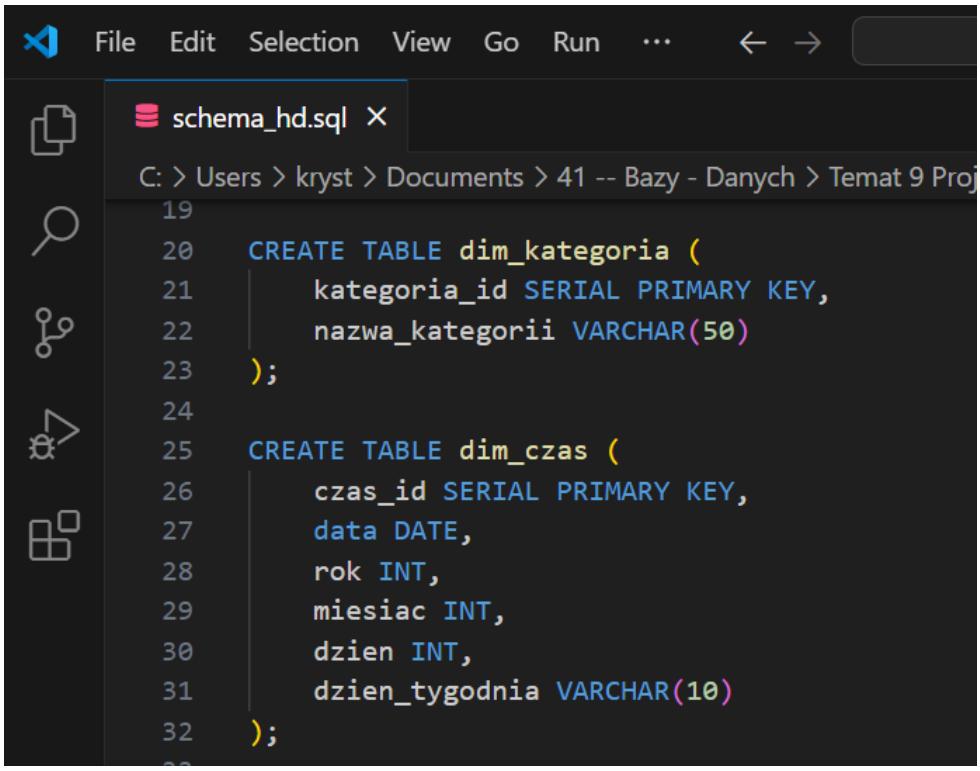


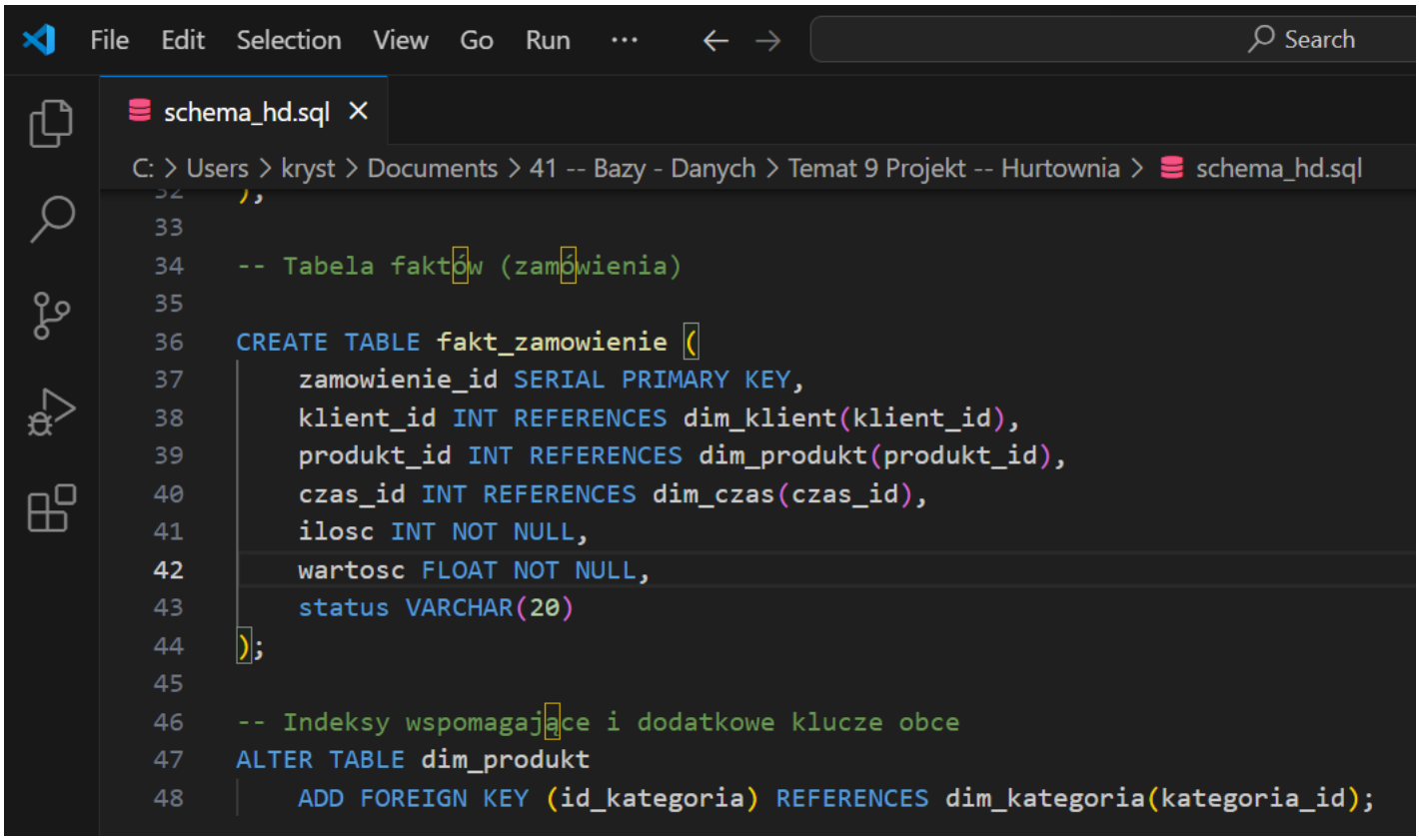
Eksport danych



5. Utwórz skrypt SQL schema_hd.sql zawierający wszystkie polecenia DDL.







-- schema_hd.sql –

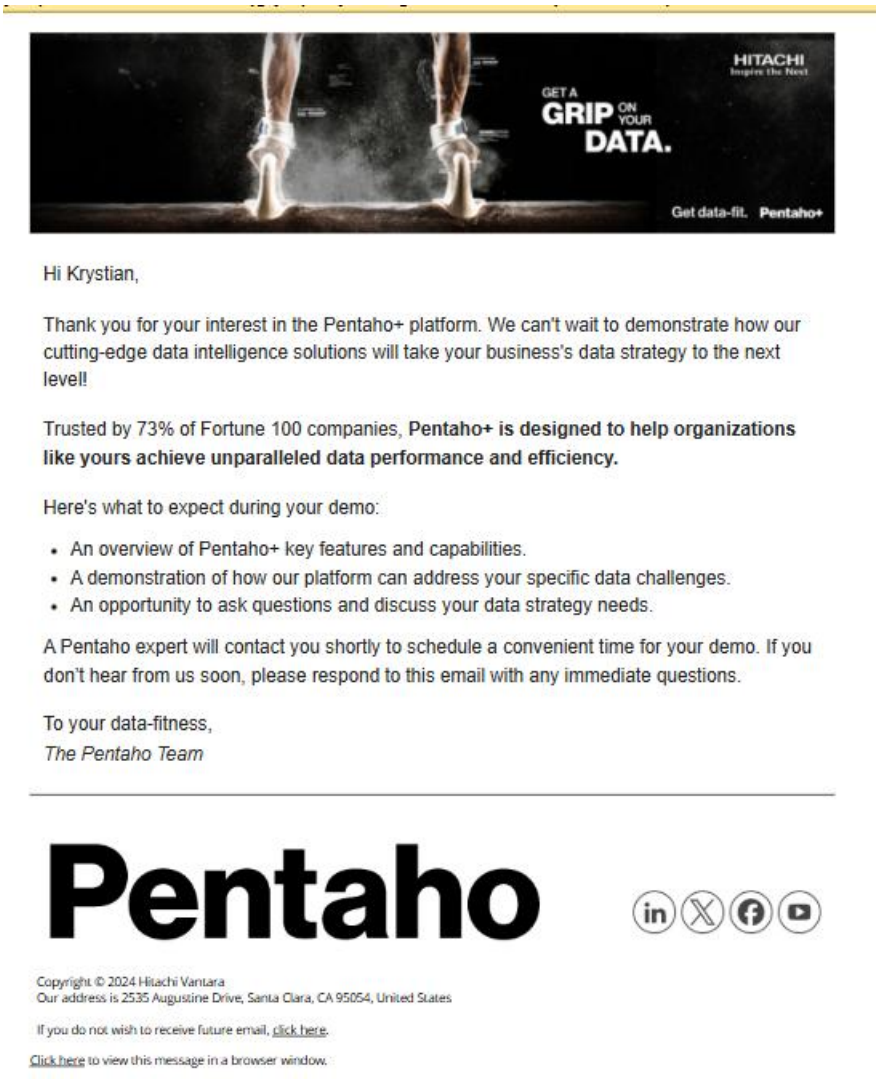
```
CREATE TABLE dim_klient (
    klient_id SERIAL PRIMARY KEY,
    imie VARCHAR(50),
    nazwisko VARCHAR(50),
    miasto VARCHAR(50),
```

```
    ulica VARCHAR(100),
    email VARCHAR(100)
);

CREATE TABLE dim_produkt (
    produkt_id SERIAL PRIMARY KEY,
    nazwa VARCHAR(100),
    parametr VARCHAR(50),
    cena_brutto FLOAT,
    id_kategoria INT
);

CREATE TABLE dim_kategoria (
    kategoria_id SERIAL PRIMARY KEY,
    nazwa_kategorii VARCHAR(50)
);

CREATE TABLE dim_czas (
    czas_id SERIAL PRIMARY KEY,
    data DATE,
    rok INT,
    miesiac INT,
    dzien INT,
    dzien_tygodnia VARCHAR(10)
);

-- Tabela faktów (zamówienia)

CREATE TABLE fakt_zamowienie (
    zamowienie_id SERIAL PRIMARY KEY,
    klient_id INT REFERENCES dim_klient(klient_id),
```

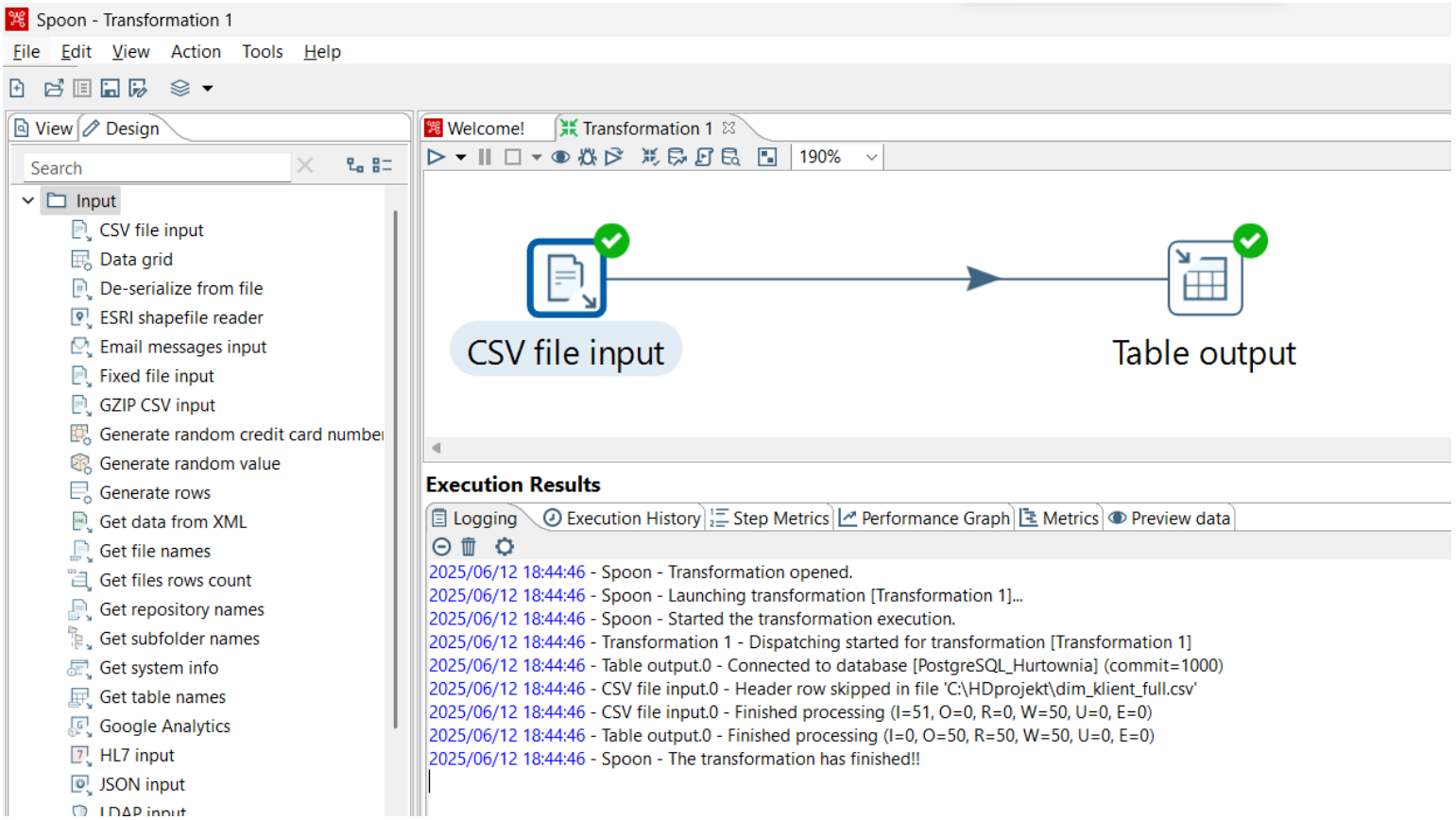
```
    produkt_id INT REFERENCES dim_produkt(produkt_id),
    czas_id INT REFERENCES dim_czas(czas_id),
    ilosc INT NOT NULL,
    wartosc FLOAT NOT NULL,
    status VARCHAR(20)
);

-- Indeksy wspomagające i dodatkowe klucze obce
ALTER TABLE dim_produkt
    ADD FOREIGN KEY (id_kategoria) REFERENCES dim_kategoria(kategoria_id);
```

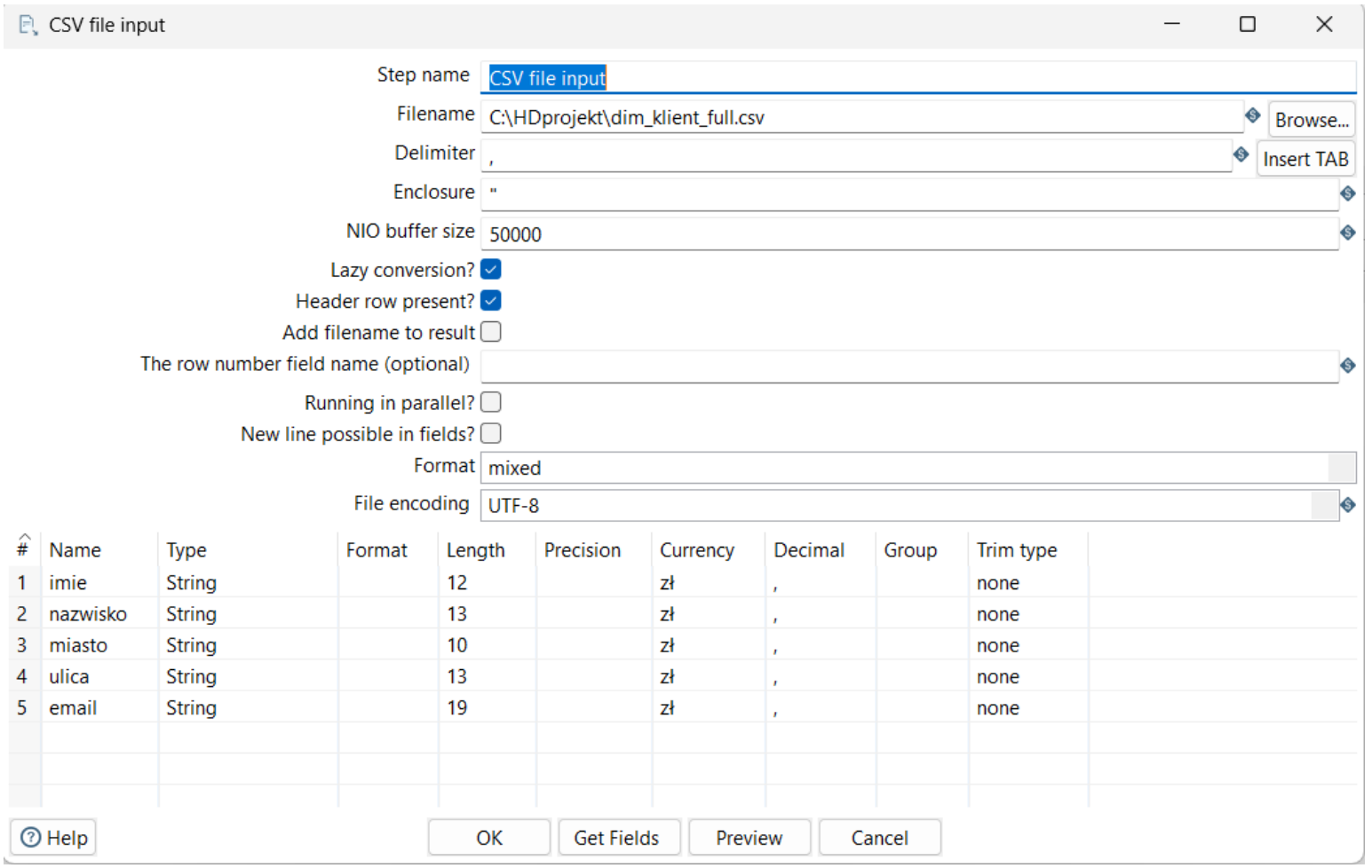
ETAP II – Implementacja Procesu ETL i Zasilenie Hurtowni

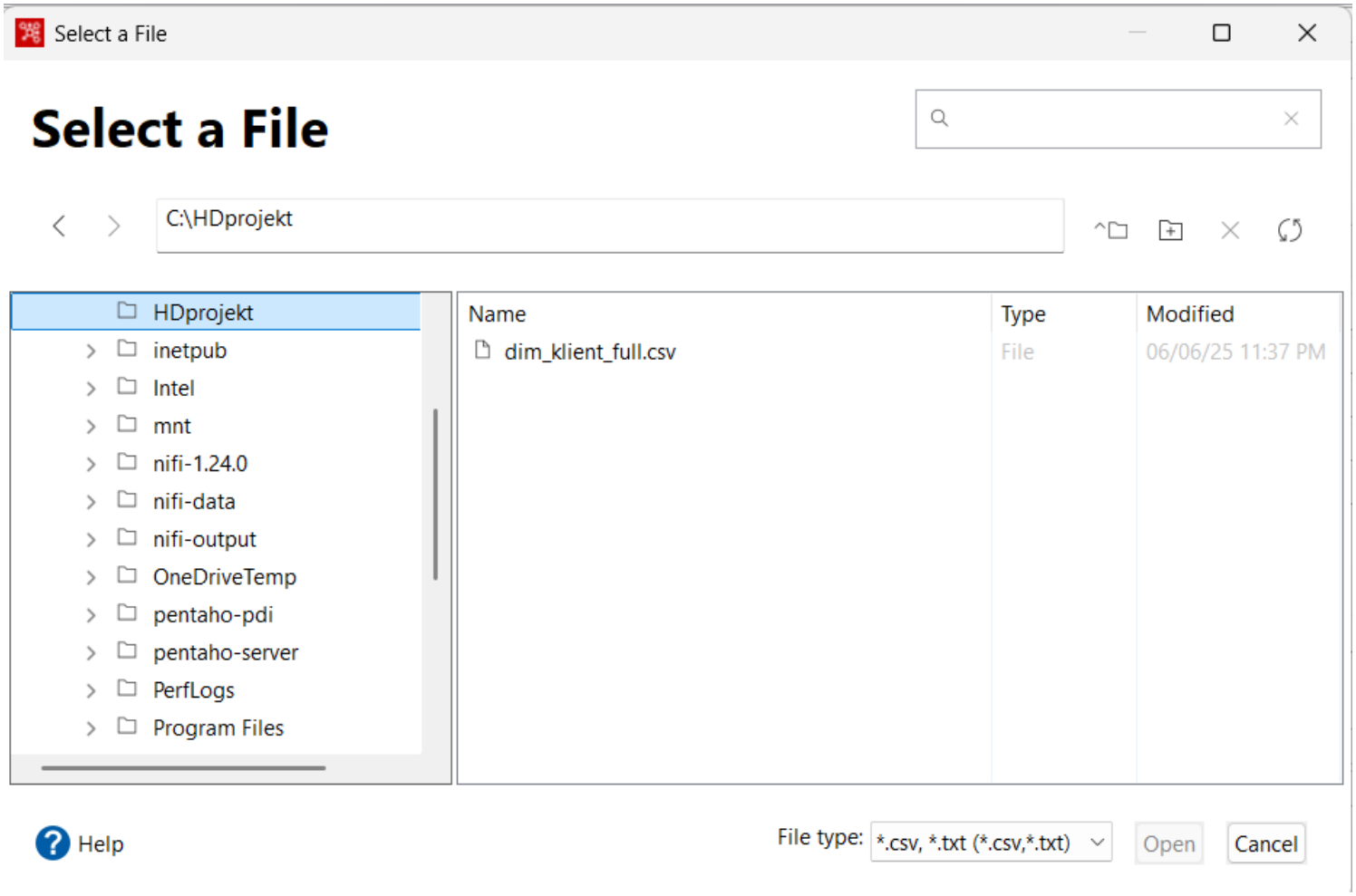
1. Przygotuj transformacje zasilające tabele wymiarów i tabelę faktów.

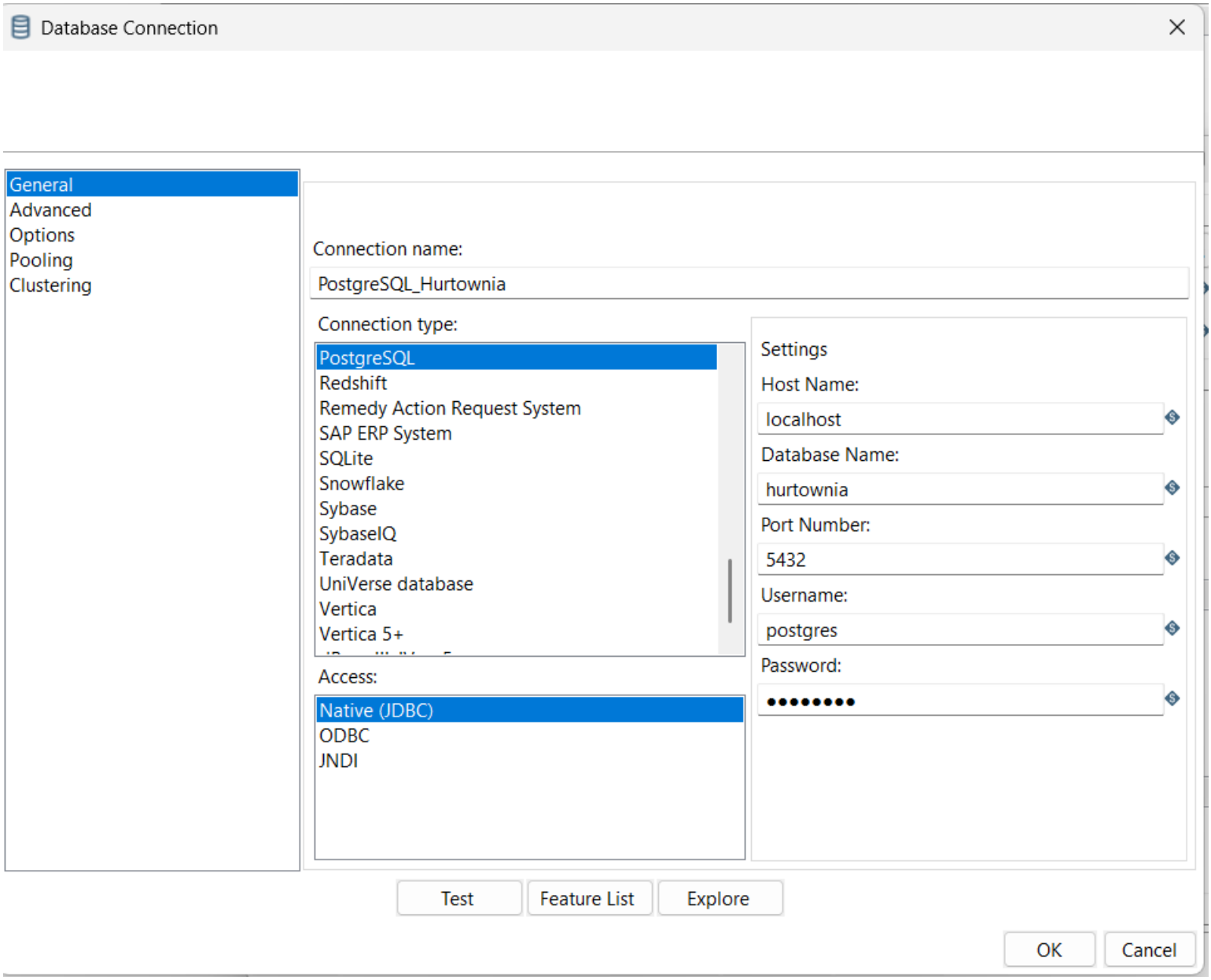


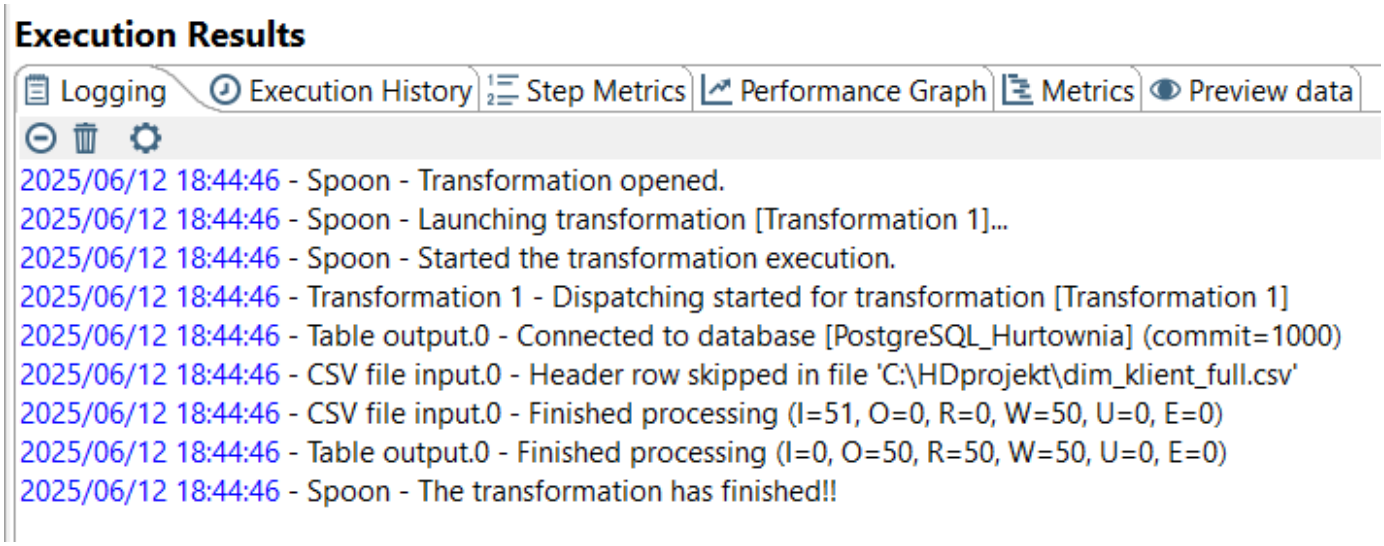
Pentaho


Spoon - Transformation 1
File Edit View Action Tools Help
View Design
Search
Input
CSV file input
Data grid
De-serialize from file
ESRI shapefile reader
Email messages input
Fixed file input
GZIP CSV input
Generate random credit card numbe
Generate random value
Generate rows
Get data from XML
Get file names
Get files rows count
Get repository names
Get subfolder names
Get system info
Get table names
Google Analytics
HL7 input
JSON input
Welcome!
Transformation 1
190%
CSV file input
Table output
Execution Results
Logging
Execution History
Step Metrics
Performance Graph
Metrics
Preview data
2025/06/12 18:44:46 - Spoon - Transformation opened.
2025/06/12 18:44:46 - Spoon - Launching transformation [Transformation 1]...
2025/06/12 18:44:46 - Spoon - Started the transformation execution.
2025/06/12 18:44:46 - Transformation 1 - Dispatching started for transformation [Transformation 1]
2025/06/12 18:44:46 - Table output.0 - Connected to database [PostgreSQL_Hurtownia] (commit=1000)
2025/06/12 18:44:46 - CSV file input.0 - Header row skipped in file 'C:\HDprojekt\dim_klient_full.csv'
2025/06/12 18:44:46 - CSV file input.0 - Finished processing (I=51, O=0, R=0, W=50, U=0, E=0)
2025/06/12 18:44:46 - Table output.0 - Finished processing (I=0, O=50, R=50, W=50, U=0, E=0)
2025/06/12 18:44:46 - Spoon - The transformation has finished!!


CSV file input
Step name
CSV file input
Filename
C:\HDprojekt\dim_klient_full.csv
Browse...
Delimiter
,
Insert TAB
Enclosure
"
NIO buffer size
50000
Lazy conversion?
Header row present?
Add filename to result
The row number field name (optional)
Running in parallel?
New line possible in fields?
Format
mixed
File encoding
UTF-8
Name Type Format Length Precision Currency Decimal Group Trim type
1 imie String 12 zł , none
2 nazwisko String 13 zł , none
3 miasto String 10 zł , none
4 ulica String 13 zł , none
5 email String 19 zł , none
Help
OK
Get Fields
Preview
Cancel

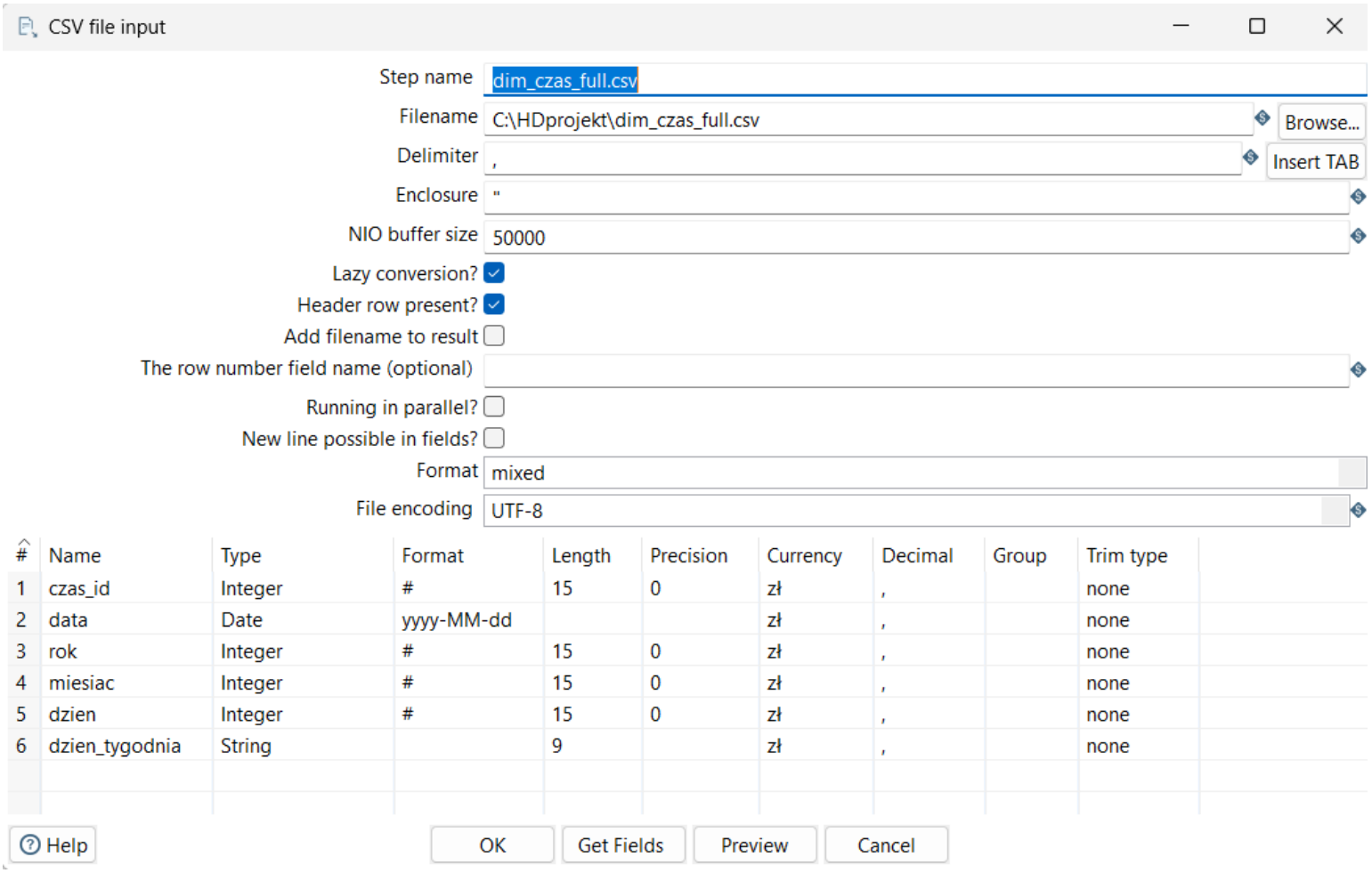

Select a File
Select a File
C:\HDprojekt
HDprojekt
inetpub
Intel
mnt
nifi-1.24.0
nifi-data
nifi-output
OneDriveTemp
pentaho-pdi
pentaho-server
PerfLogs
Program Files
Name
Type
Modified
dim_klient_full.csv
File
06/06/25 11:37 PM
Help
File type:
*.csv, *.txt (*.csv,*.txt)
Open
Cancel



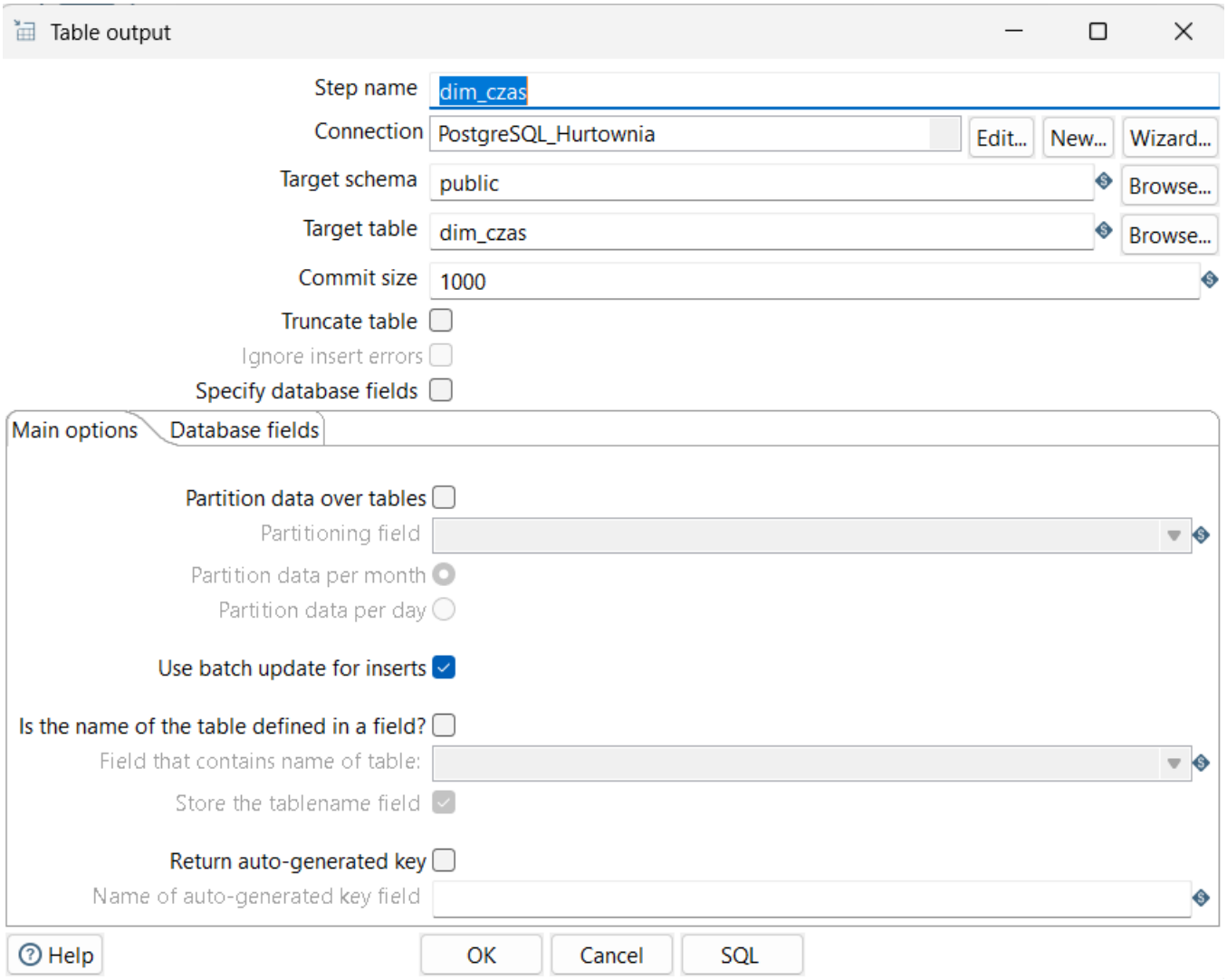


TRUNCATE TABLE fakt_zamowienie, dim_czas RESTART IDENTITY CASCADE;

dim_czas



#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	czas_id	Integer	#	15	0	zł	,		none
2	data	Date	yyyy-MM-dd			zł	,		none
3	rok	Integer	#	15	0	zł	,		none
4	miesiac	Integer	#	15	0	zł	,		none
5	dzien	Integer	#	15	0	zł	,		none
6	dzien_tygodnia	String		9		zł	,		none

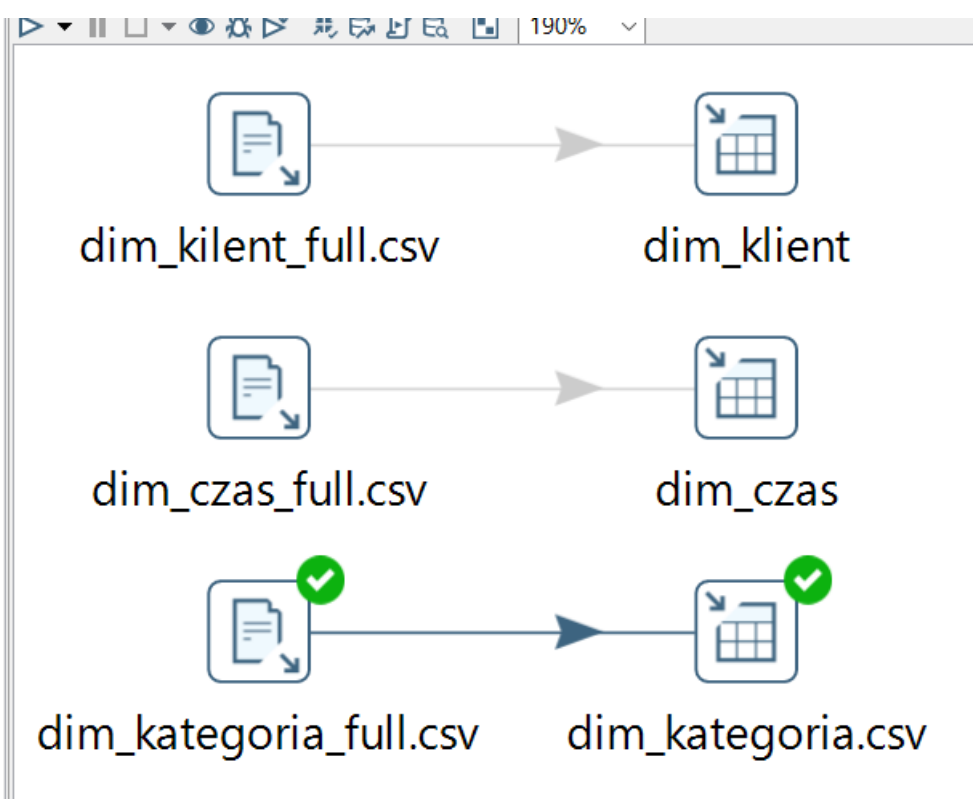


Execution Results

Logging | Execution History | Step Metrics | Performance Graph | Metrics | Preview data

```
2025/06/12 20:27:56 - Spoon - Transformation opened.
2025/06/12 20:27:56 - Spoon - Launching transformation [Transformation 1]...
2025/06/12 20:27:56 - Spoon - Started the transformation execution.
2025/06/12 20:27:56 - Transformation 1 - Dispatching started for transformation [Transformation 1]
2025/06/12 20:27:56 - dim_klient.0 - Connected to database [PostgreSQL_Hurtownia] (commit=1000)
2025/06/12 20:27:56 - dim_czas.0 - Connected to database [PostgreSQL_Hurtownia] (commit=1000)
2025/06/12 20:27:56 - dim_kilent_full.csv.0 - Header row skipped in file 'C:\HDprojekt\dim_klient_full.csv'
2025/06/12 20:27:56 - dim_czas_full.csv.0 - Header row skipped in file 'C:\HDprojekt\dim_czas_full.csv'
2025/06/12 20:27:56 - dim_kilent_full.csv.0 - Finished processing (I=51, O=0, R=0, W=50, U=0, E=0)
2025/06/12 20:27:56 - dim_czas_full.csv.0 - Finished processing (I=366, O=0, R=0, W=365, U=0, E=0)
2025/06/12 20:27:56 - dim_klient.0 - Finished processing (I=0, O=50, R=50, W=50, U=0, E=0)
2025/06/12 20:27:56 - dim_czas.0 - Finished processing (I=0, O=365, R=365, W=365, U=0, E=0)
2025/06/12 20:27:56 - Spoon - The transformation has finished!!
```

```
--select * from dim_kategoria ;

--TRUNCATE TABLE fakt_zamowienie, dim_kategoria RESTART IDENTITY CASCADE;
```



CSV file input

Step name: dim_kategoria_full.csv

Filename: C:\HDprojekt\dim_kategoria_full.csv [Browse...]

Delimiter: , [Insert TAB]

Enclosure: "

NIO buffer size: 50000

Lazy conversion? ☑

Header row present? ☑

Add filename to result ☐

The row number field name (optional):

Running in parallel? ☐

New line possible in fields? ☐

Format: mixed

File encoding: UTF-8

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	kategoria_id	Integer	#	15	0	zł	,		none
2	nazwa_kategorii	String		39		zł	,		none

Help | OK | Get Fields | Preview | Cancel

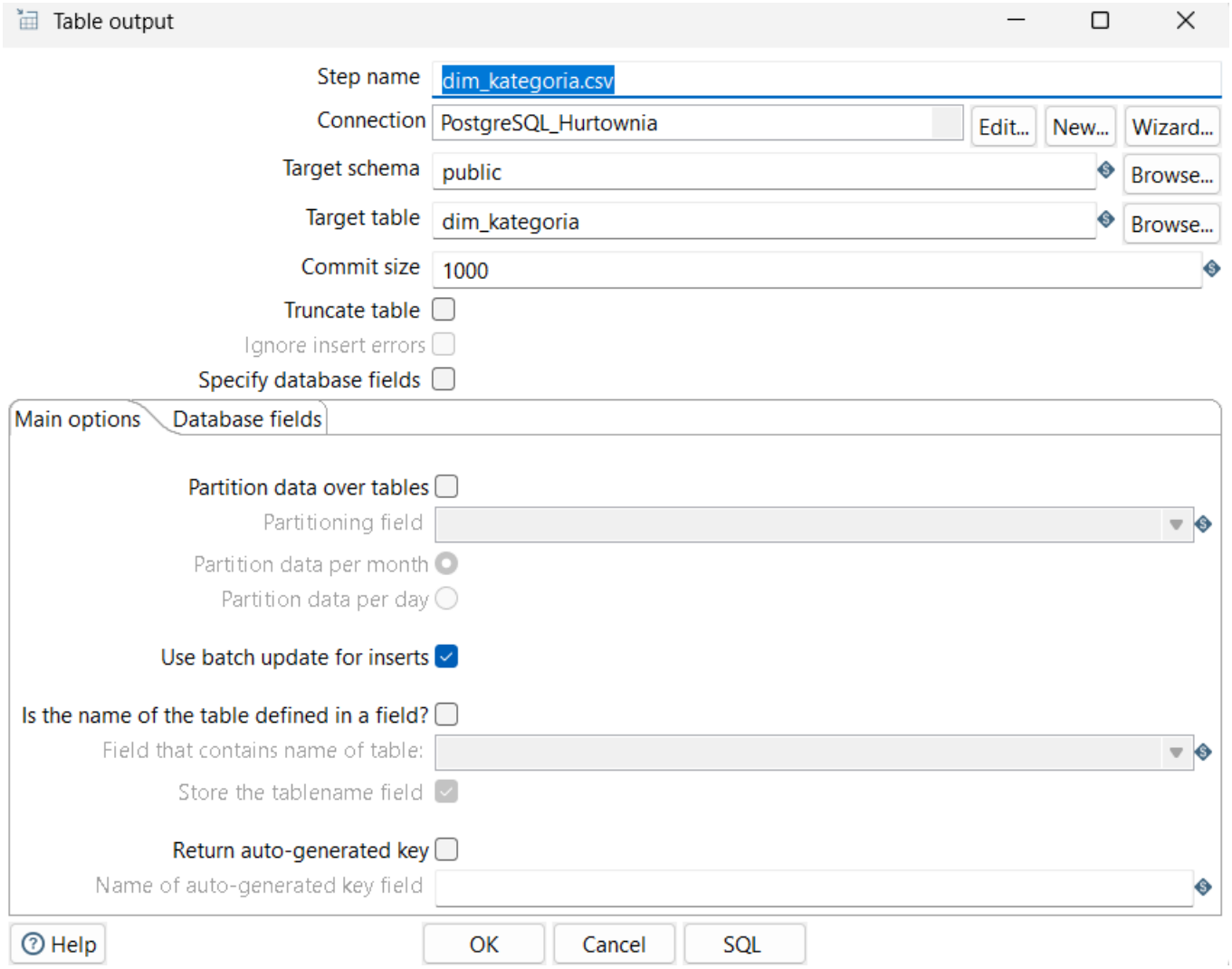
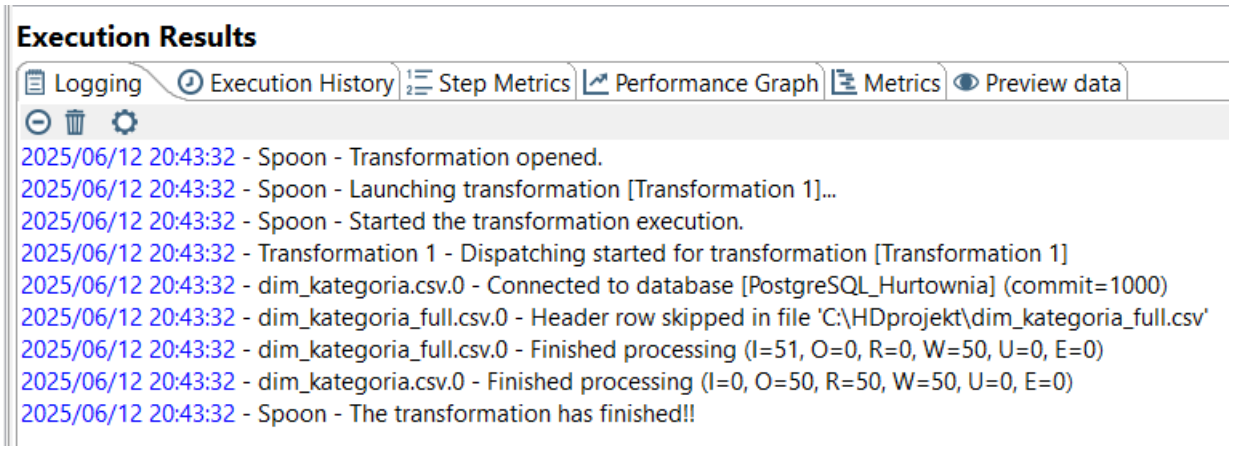
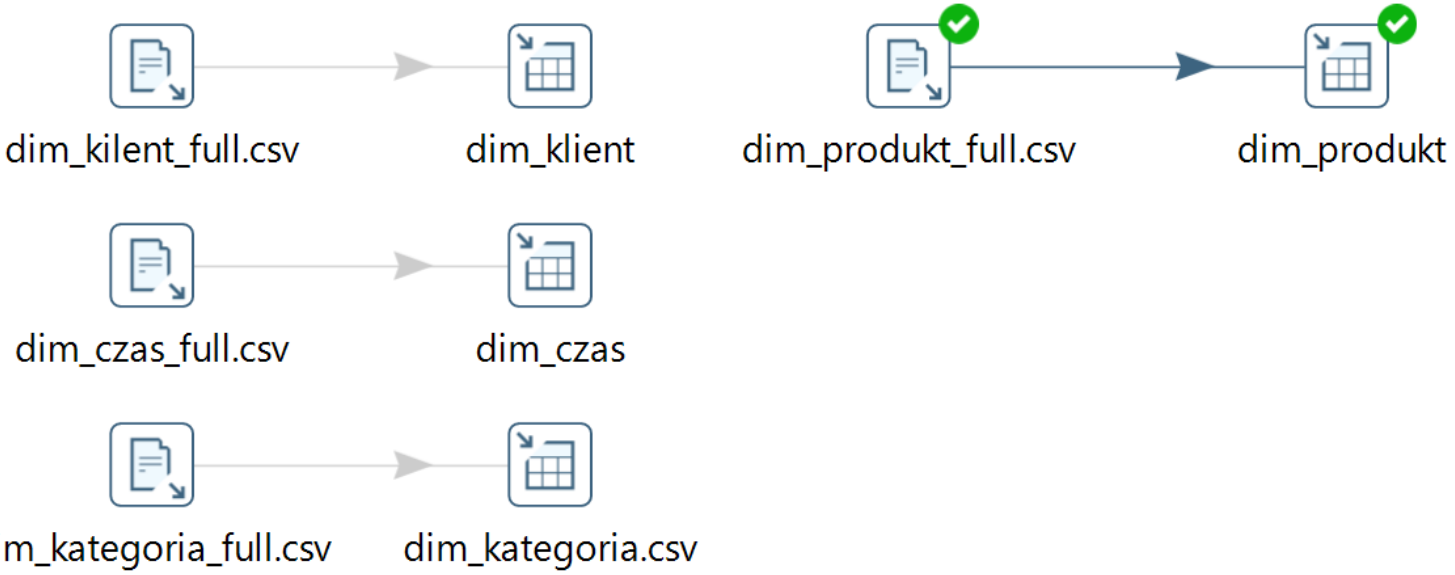


Table output
Step name
dim_kategoria.csv
Connection
PostgreSQL_Hurtownia
Edit...
New...
Wizard...
Target schema
public
Browse...
Target table
dim_kategoria
Browse...
Commit size
1000
Truncate table
Ignore insert errors
Specify database fields
Main options
Database fields
Partition data over tables
Partitioning 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 tablename field
Return auto-generated key
Name of auto-generated key field
Help
OK
Cancel
SQL


Execution Results
Logging
Execution History
Step Metrics
Performance Graph
Metrics
Preview data
2025/06/12 20:43:32 - Spoon - Transformation opened.
2025/06/12 20:43:32 - Spoon - Launching transformation [Transformation 1]...
2025/06/12 20:43:32 - Spoon - Started the transformation execution.
2025/06/12 20:43:32 - Transformation 1 - Dispatching started for transformation [Transformation 1]
2025/06/12 20:43:32 - dim_kategoria.csv.0 - Connected to database [PostgreSQL_Hurtownia] (commit=1000)
2025/06/12 20:43:32 - dim_kategoria_full.csv.0 - Header row skipped in file 'C:\HDprojekt\dim_kategoria_full.csv'
2025/06/12 20:43:32 - dim_kategoria_full.csv.0 - Finished processing (I=51, O=0, R=0, W=50, U=0, E=0)
2025/06/12 20:43:32 - dim_kategoria.csv.0 - Finished processing (I=0, O=50, R=50, W=50, U=0, E=0)
2025/06/12 20:43:32 - Spoon - The transformation has finished!!

dim_produkt



CSV file input

Step name: dim_produkt_full.csv

Filename: C:\HDprojekt\dim_produkt_full.csv — Browse...

Delimiter: , — Insert TAB

Enclosure: "

NIO buffer size: 50000

Lazy conversion? ☑

Header row present? ☑

Add filename to result ☐

The row number field name (optional):

Running in parallel? ☐

New line possible in fields? ☐

Format: mixed

File encoding: UTF-8

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	nazwa	String		28		zł	,		none
2	parametr	String		44		zł	,		none
3	cena_brutto	Integer	#	15	0	zł	,		none
4	id_kategoria	Integer	#	15	0	zł	,		none

Help | OK | Get Fields | Preview | Cancel

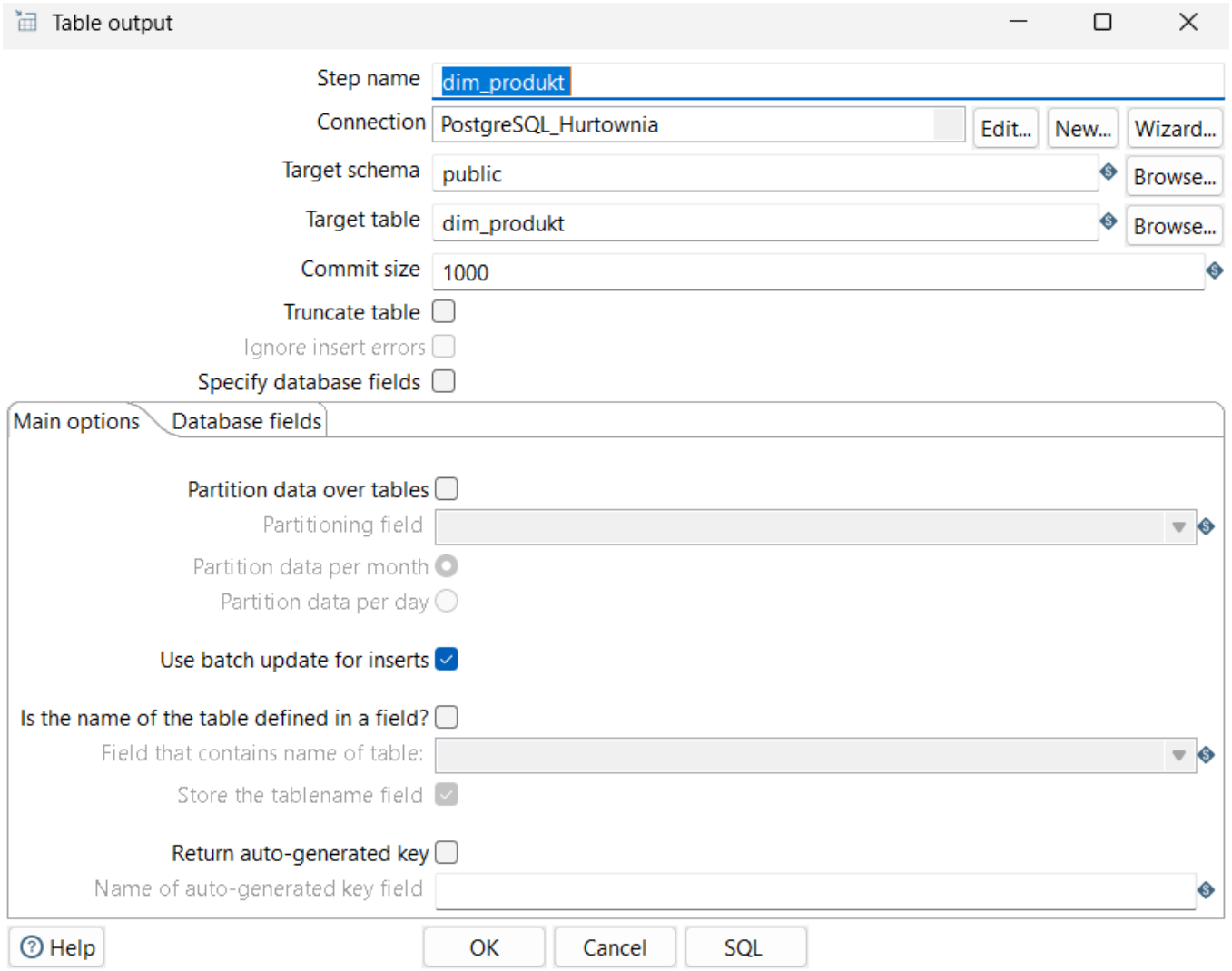
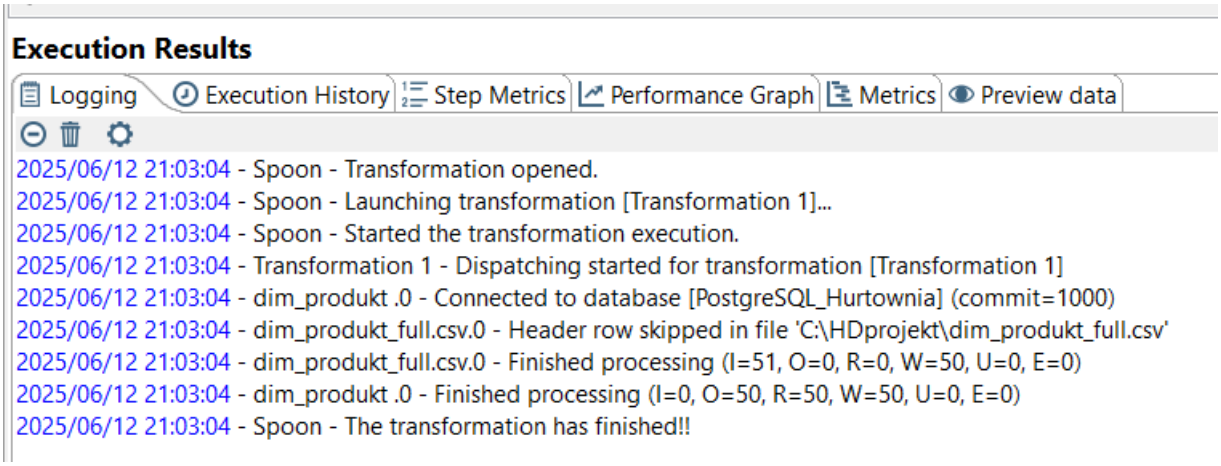
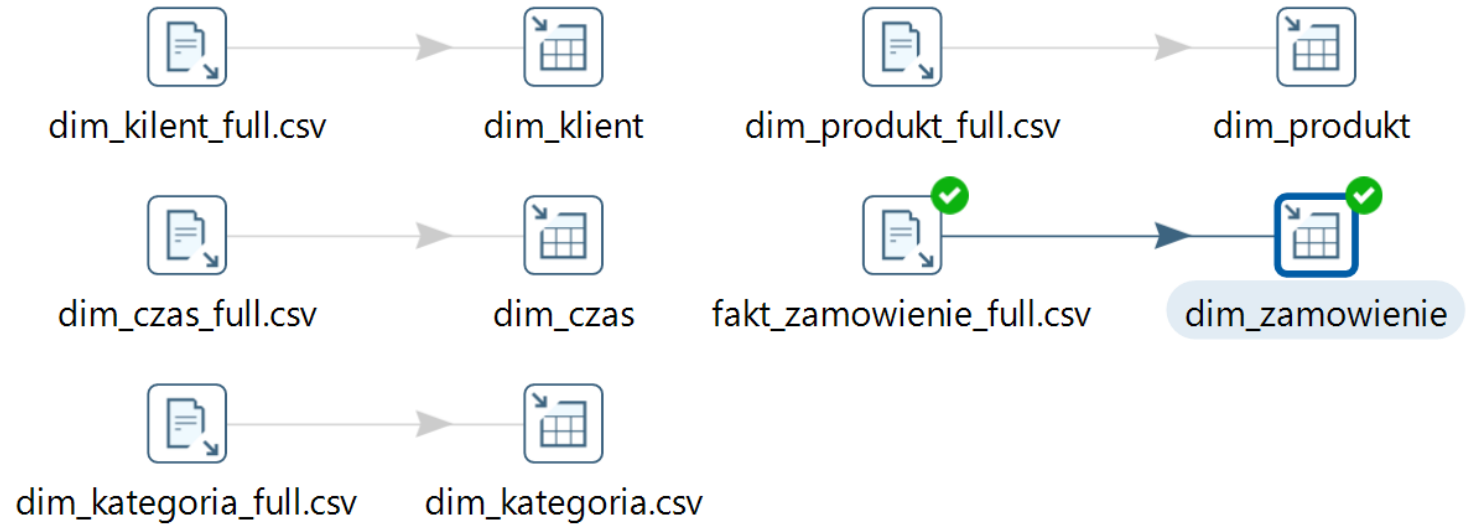


Table output
Step name
dim_produkt
Connection
PostgreSQL_Hurtownia
Edit...
New...
Wizard...
Target schema
public
Browse...
Target table
dim_produkt
Browse...
Commit size
1000
Truncate table
Ignore insert errors
Specify database fields
Main options
Database fields
Partition data over tables
Partitioning 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 tablename field
Return auto-generated key
Name of auto-generated key field
Help
OK
Cancel
SQL


Execution Results
Logging
Execution History
Step Metrics
Performance Graph
Metrics
Preview data
2025/06/12 21:03:04 - Spoon - Transformation opened.
2025/06/12 21:03:04 - Spoon - Launching transformation [Transformation 1]...
2025/06/12 21:03:04 - Spoon - Started the transformation execution.
2025/06/12 21:03:04 - Transformation 1 - Dispatching started for transformation [Transformation 1]
2025/06/12 21:03:04 - dim_produkt .0 - Connected to database [PostgreSQL_Hurtownia] (commit=1000)
2025/06/12 21:03:04 - dim_produkt_full.csv.0 - Header row skipped in file 'C:\HDprojekt\dim_produkt_full.csv'
2025/06/12 21:03:04 - dim_produkt_full.csv.0 - Finished processing (I=51, O=0, R=0, W=50, U=0, E=0)
2025/06/12 21:03:04 - dim_produkt .0 - Finished processing (I=0, O=50, R=50, W=50, U=0, E=0)
2025/06/12 21:03:04 - Spoon - The transformation has finished!!


dim_kilent_full.csv
dim_klient
dim_produkt_full.csv
dim_produkt
dim_czas_full.csv
dim_czas
fakt_zamowienie_full.csv
dim_zamowienie
dim_kategoria_full.csv
dim_kategoria.csv

CSV file input

Step name: fakt_zamowienie_full.csv
Filename: C:\HDprojekt\fakt_zamowienie_full.csv [Browse...]
Delimiter: , [Insert TAB]
Enclosure: "
NIO buffer size: 50000
Lazy conversion? ☑
Header row present? ☑
Add filename to result ☐
The row number field name (optional):
Running in parallel? ☐
New line possible in fields? ☐
Format: mixed
File encoding: UTF-8

#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	klient_id	Integer	#	15	0	zł	,		none
2	produkt_id	Integer	#	15	0	zł	,		none
3	czas_id	Integer	#	15	0	zł	,		none
4	ilosc	Integer	#	15	0	zł	,		none
5	wartosc	Number	#.#	15	0	zł	.	,	none
6	status	String		12		zł	,		none

Help | OK | Get Fields | Preview | Cancel

Table output

Step name: dim_zamowienie
Connection: PostgreSQL_Hurtownia [Edit...] [New...] [Wizard...]
Target schema: public [Browse...]
Target table: fakt_zamowienie [Browse...]
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 tablename field ☑

Return auto-generated key ☐
Name of auto-generated key field:

Help | OK | Cancel | SQL

Execution Results

Logging | Execution History | Step Metrics | Performance Graph | Metrics | Preview data

2025/06/12 21:08:20 - Spoon - Transformation opened.
2025/06/12 21:08:20 - Spoon - Launching transformation [Transformation 1]...
2025/06/12 21:08:20 - Spoon - Started the transformation execution.
2025/06/12 21:08:20 - Transformation 1 - Dispatching started for transformation [Transformation 1]
2025/06/12 21:08:20 - dim_zamowienie.0 - Connected to database [PostgreSQL_Hurtownia] (commit=1000)
2025/06/12 21:08:20 - fakt_zamowienie_full.csv.0 - Header row skipped in file 'C:\HDprojekt\fakt_zamowienie_full.csv'
2025/06/12 21:08:20 - fakt_zamowienie_full.csv.0 - Finished processing (I=51, O=0, R=0, W=50, U=0, E=0)
2025/06/12 21:08:20 - dim_zamowienie.0 - Finished processing (I=0, O=50, R=50, W=50, U=0, E=0)
2025/06/12 21:08:20 - Spoon - The transformation has finished!!