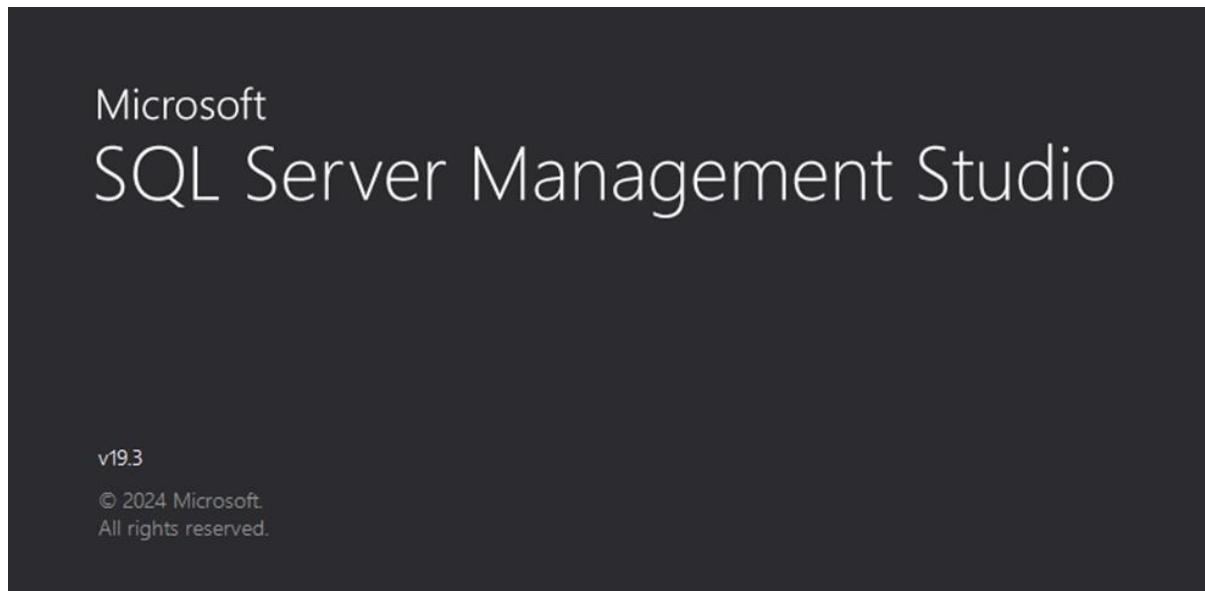
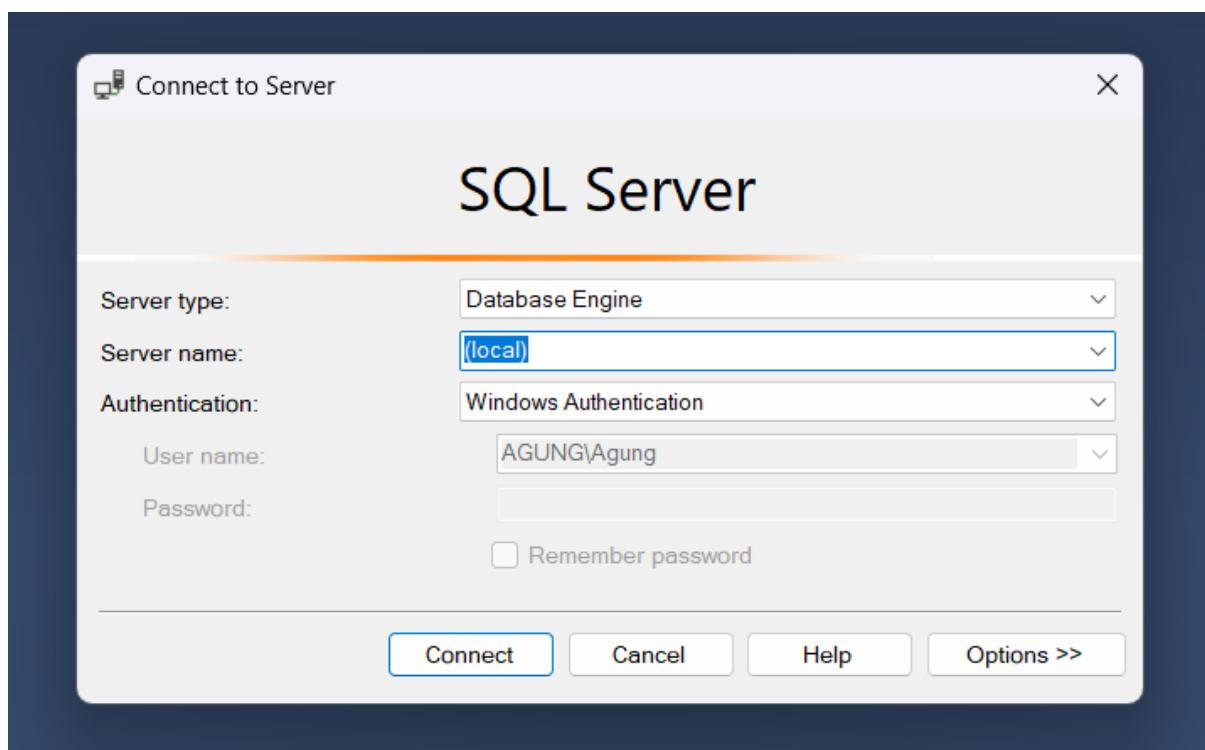


Dokumentasi Tugas Besar Pergudangan Data

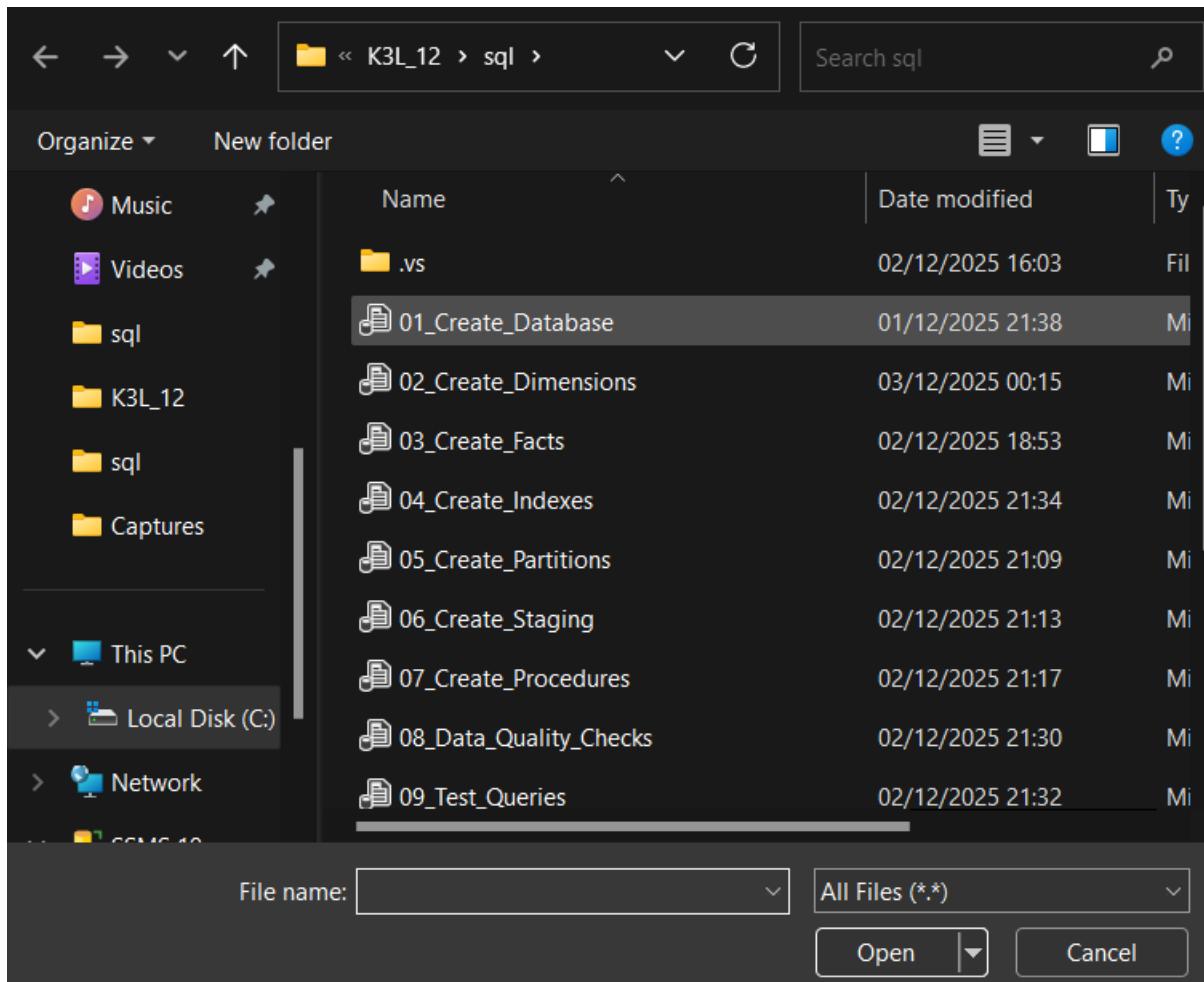
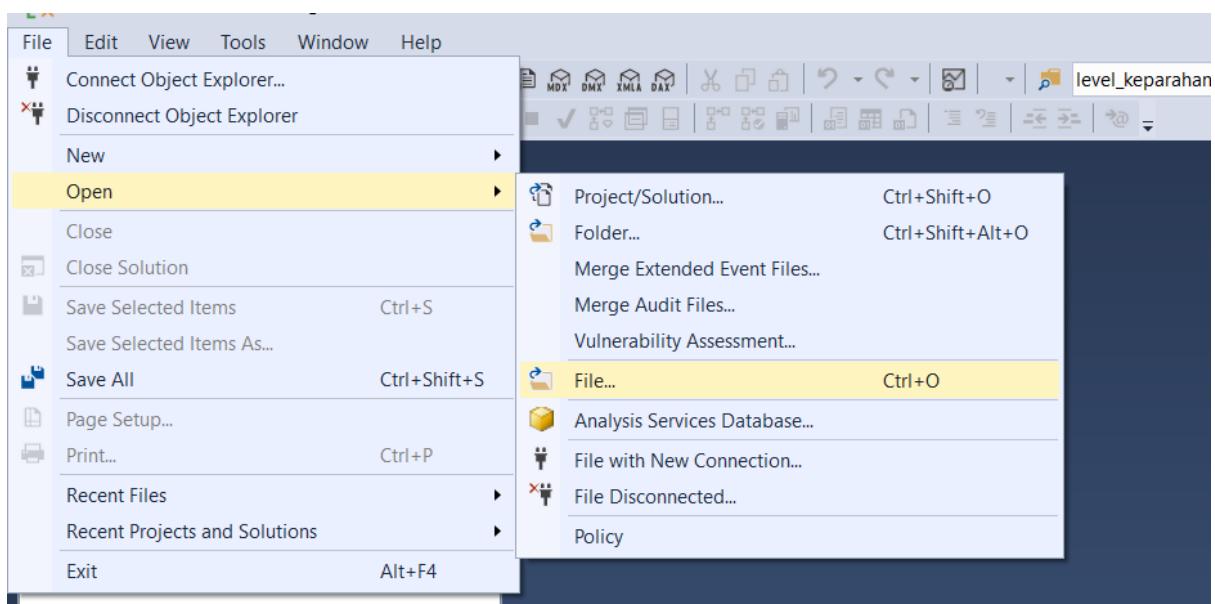
Menggunakan SSMS



Connect to SQL server



01_Create_Database



Execute code

The screenshot shows the SSMS interface. The Object Explorer on the left lists the server (local) and its databases, including master, model, msdb, tempdb, and K3L_DataMart. The query window on the right contains T-SQL code for creating a database named K3L_DataMart. The code includes specifying file and log properties, setting recovery mode to SIMPLE, and printing a success message.

```
-- Create Database K3L_DataMart
ON PRIMARY
(
    NAME = N'K3L_DataMart_Data',
    FILENAME = N'C:\K3L\K3L_DataMart_Data.mdf',
    SIZE = 16GB,
    MAXSIZE = UNLIMITED,
    FILEGROWTH = 256MB
)
LOG ON
(
    NAME = N'K3L_DataMart_Log',
    FILENAME = N'C:\K3L\K3L_DataMart_Log.ldf',
    SIZE = 256MB,
    MAXSIZE = 2GB,
    FILEGROWTH = 64MB
);
GO

PRINT 'Database K3L_DataMart created successfully.';
GO

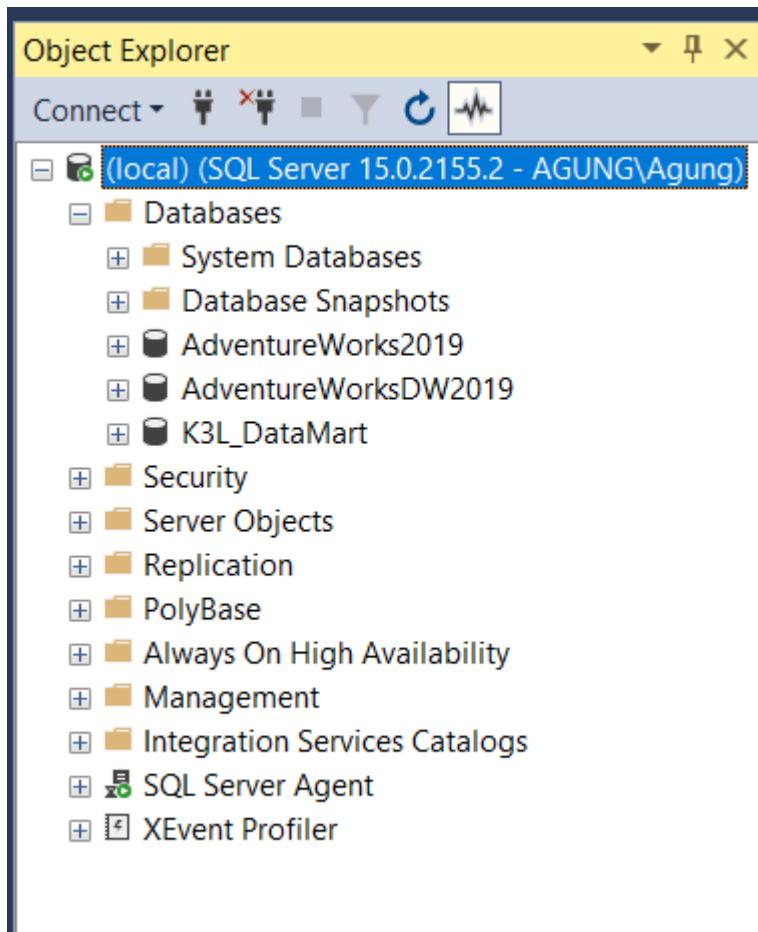
-- Set database options
ALTER DATABASE K3L_DataMart
SET RECOVERY SIMPLE;
GO
```

Output : Database K3L_DataMart berhasil dibuat

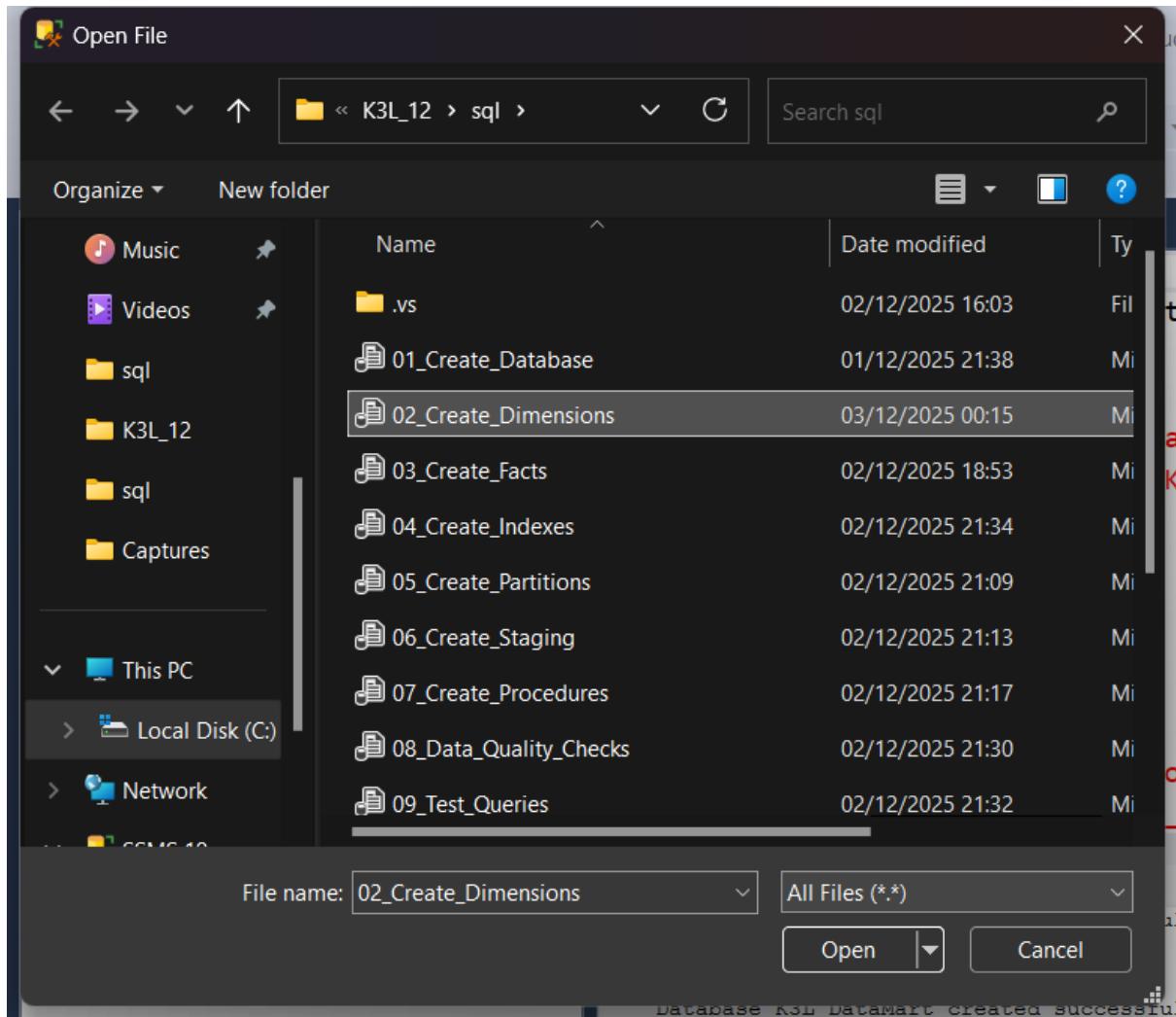
```
Database K3L_DataMart created successfully.
Database configuration completed.

Database K3L_DataMart created successfully!

Completion time: 2025-12-08T15:53:42.7783443+07:00
```



02_Create_Dimension



Execute code

The screenshot shows the SSMS interface. The Object Explorer on the left shows the database structure, including the K3L_DataMart database. The central Query Editor window displays the SQL script "02_Create_Dimensions.sql" which contains the following code:

```
-- Use the database
USE K3L_DataMart;
GO

-- CREATE ALL DIMENSION TABLES FOR DATA MART K3L (ITERA) WITH IDENTITY
-----

-- 1. Dim_Date
CREATE TABLE Dim_Date (
    DateID INT IDENTITY(1,1) PRIMARY KEY,
    Tanggal DATE NOT NULL,
    Tahun INT NOT NULL,
    Bulan VARCHAR(20),
    Kuarter VARCHAR(10)
);
GO

-- 2. Dim_Lokasi
CREATE TABLE Dim_Lokasi (
    LokasiID INT IDENTITY(1,1) PRIMARY KEY,
    NamaLokasi VARCHAR(100),
    Gedung VARCHAR(100),
    Lantai VARCHAR(25)
);
GO

-- 3. Dim_UnitKerja
CREATE TABLE Dim_UnitKerja (
    
```

The status bar at the bottom right indicates "Connected. (1/1)".

Output

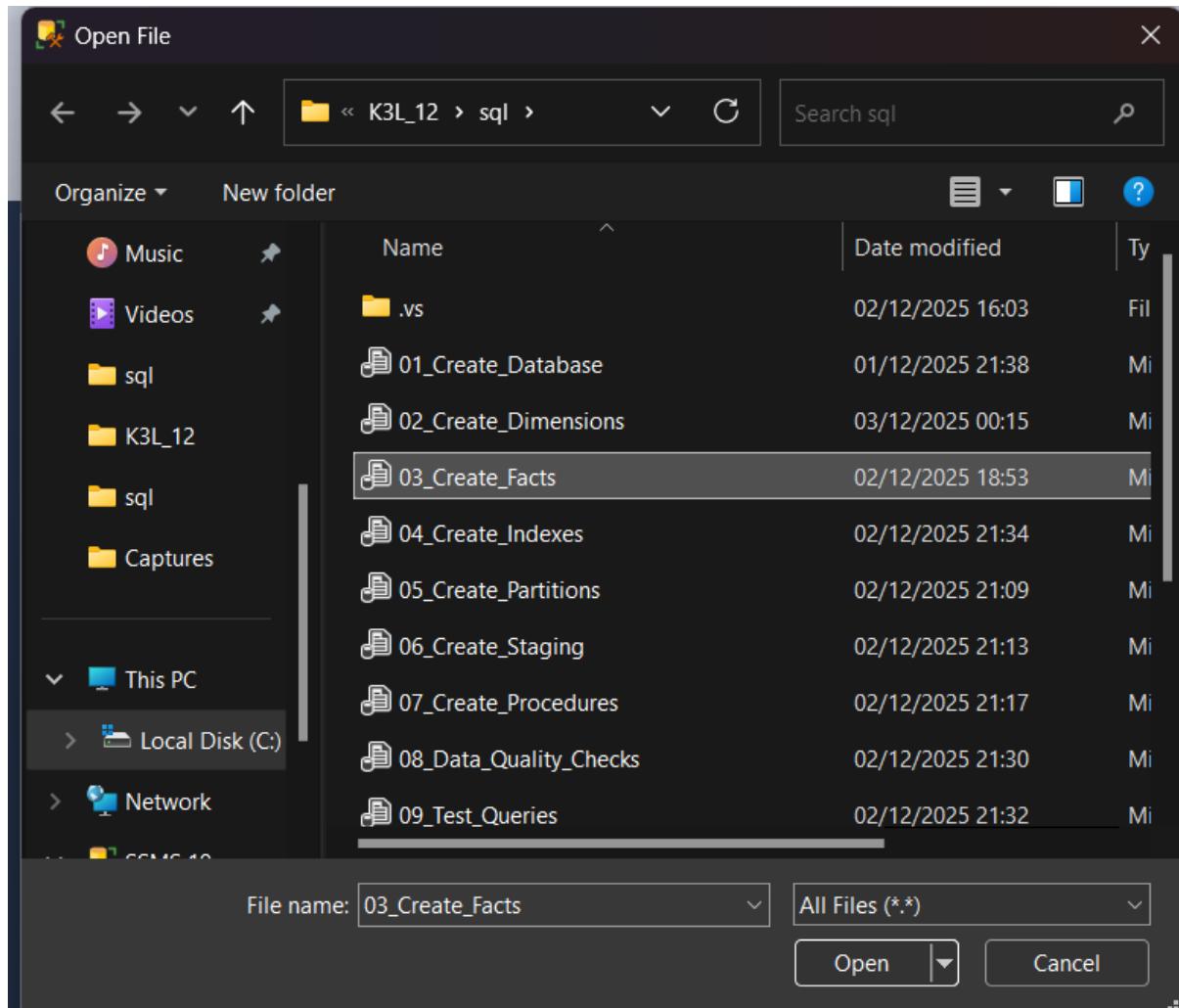
Messages

Commands completed successfully.

Completion time: 2025-12-08T15:57:32.7429602+07:00

- [-] K3L_DataMart
 - [+] Database Diagrams
 - [-] Tables
 - [+] System Tables
 - [+] FileTables
 - [+] External Tables
 - [+] Graph Tables
 - [+] dbo.Dim_Date
 - [+] dbo.Dim_JenisInsiden
 - [+] dbo.Dim_JenisLimbah
 - [+] dbo.Dim_JenisPeralatan
 - [+] dbo.Dim_Lokasi
 - [+] dbo.Dim_Petugas
 - [+] dbo.Dim_Severity
 - [+] dbo.Dim_UnitKerja
 - [-]

03_Create_Facts



Execute code

The screenshot shows the SQL Server Management Studio (SSMS) interface. The Object Explorer on the left shows the database structure, including the "K3L_DataMart" database and its tables. The central pane displays the T-SQL script for creating fact tables:

```
-- Use the database
USE K3L_DataMart;
GO

-- CREATE FACT TABLES - DATA MART K3L ITERA WITH IDENTITY AND INT PK
-----  
-----  
-- 1. Fact_Insiden
CREATE TABLE Fact_Insiden (
    InsidenID INT IDENTITY(1,1) PRIMARY KEY,
    DateID INT,
    LokasiID INT,
    UnitKerjaID INT,
    JenisInsidenID INT,
    SeverityID INT,
    PetugasID INT,
    JumlahKorban INT,
    HariKerjaHilang INT,
    FOREIGN KEY (DateID) REFERENCES Dim_Date(DateID),
    FOREIGN KEY (LokasiID) REFERENCES Dim_Lokasi(LokasiID),
    FOREIGN KEY (UnitKerjaID) REFERENCES Dim_UnitKerja(UnitKerjaID),
    FOREIGN KEY (JenisInsidenID) REFERENCES Dim_JenisInsiden(JenisInsidenID),
    FOREIGN KEY (SeverityID) REFERENCES Dim_Severity(SeverityID),
    FOREIGN KEY (PetugasID) REFERENCES Dim_Petugas(PetugasID)
);
GO
```

The status bar at the bottom indicates "Connected. (1/1)" and "(local) (15.0 RTM) AGUNG\Agung (66) K3L_DataMart 00:00:00 0 rows".

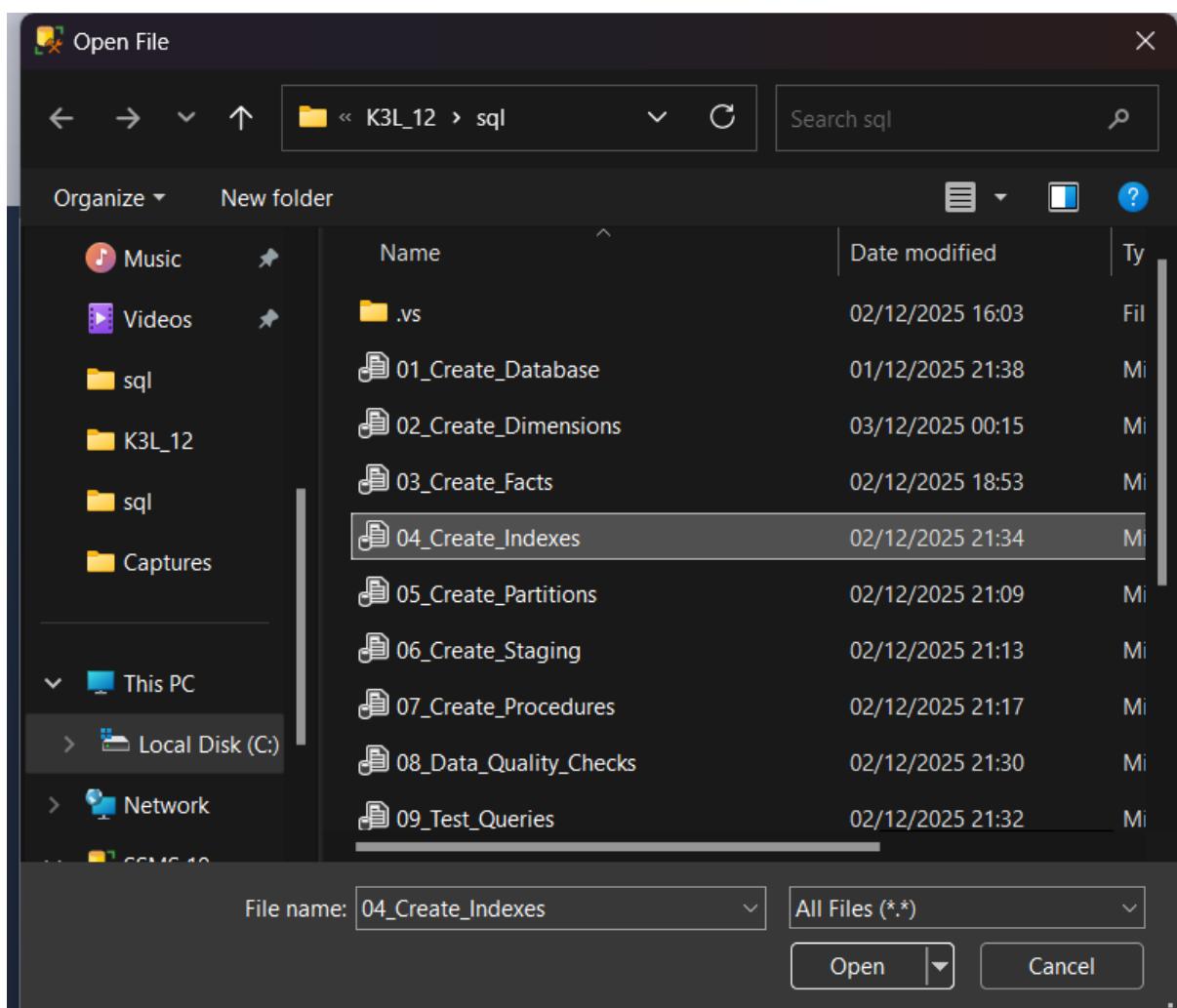
Output

```
Messages
Commands completed successfully.

Completion time: 2025-12-08T16:01:49.2030438+07:00

    + [+] dbo.Fact_Insiden
    + [+] dbo.Fact_Inspeksi
    + [+] dbo.Fact_Limbah
    + [+] Views
    + [+] External Resources
    + [+] Synonyms
    + [+] Programmability
    + [+] Service Broker
    + [+] Storage
    + [+] Security
```

04_Create_Indexes



Excute Code

```
-- FILE: 04_Create_Indexes.sql
-- FIXED VERSION: Without clustered index creation

USE K3L_DataMart;
GO

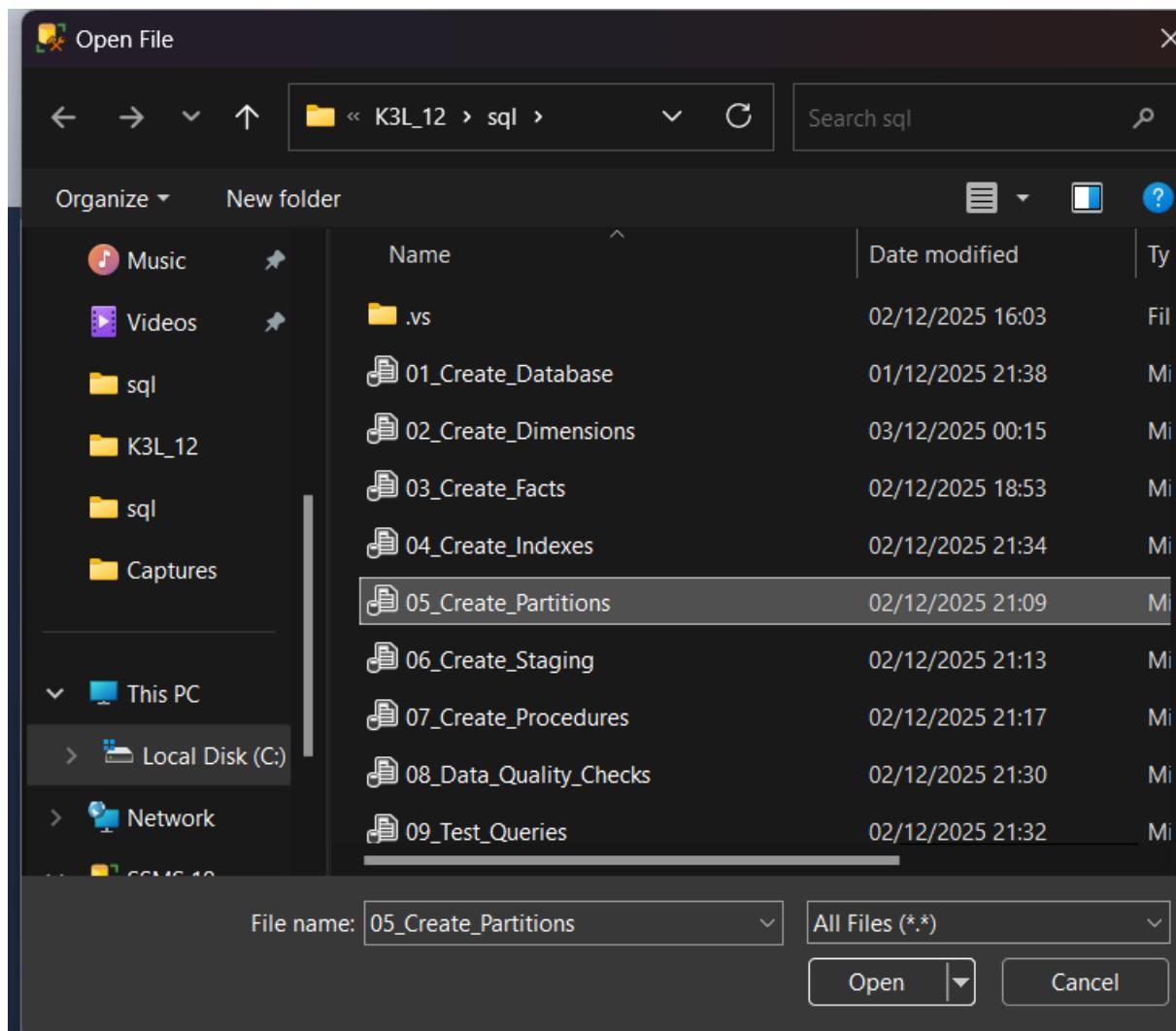
/* =====
A. INDEX FOR FACT_INSIDEN
===== */

-- Foreign Key Indexes
CREATE NONCLUSTERED INDEX IX_Fact_Insiden_Date
```

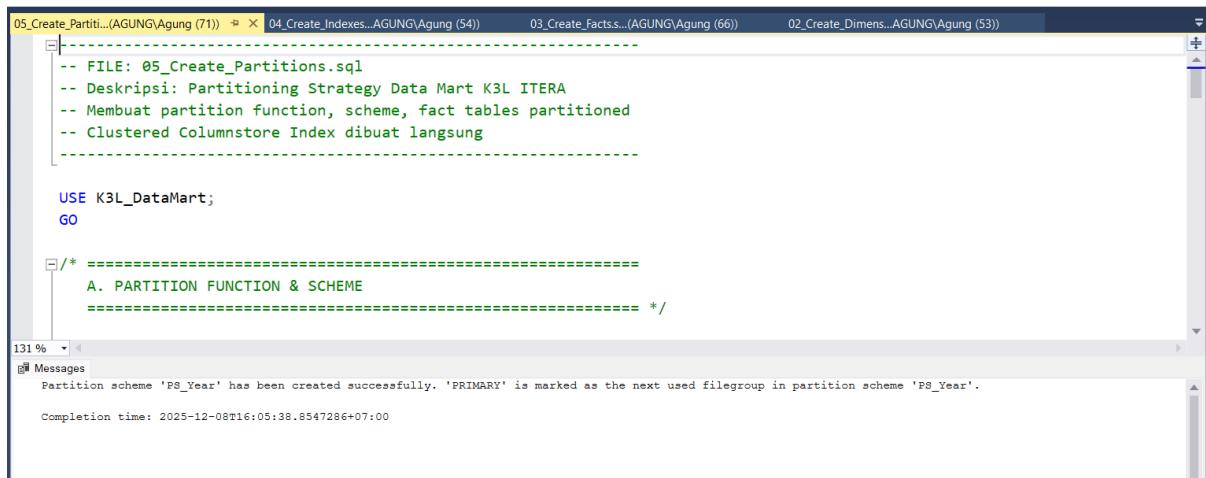
131 % Messages Commands completed successfully.

Completion time: 2025-12-08T16:03:26.4876770+07:00

05_Create_Partitions



Excute code & menampilkan output



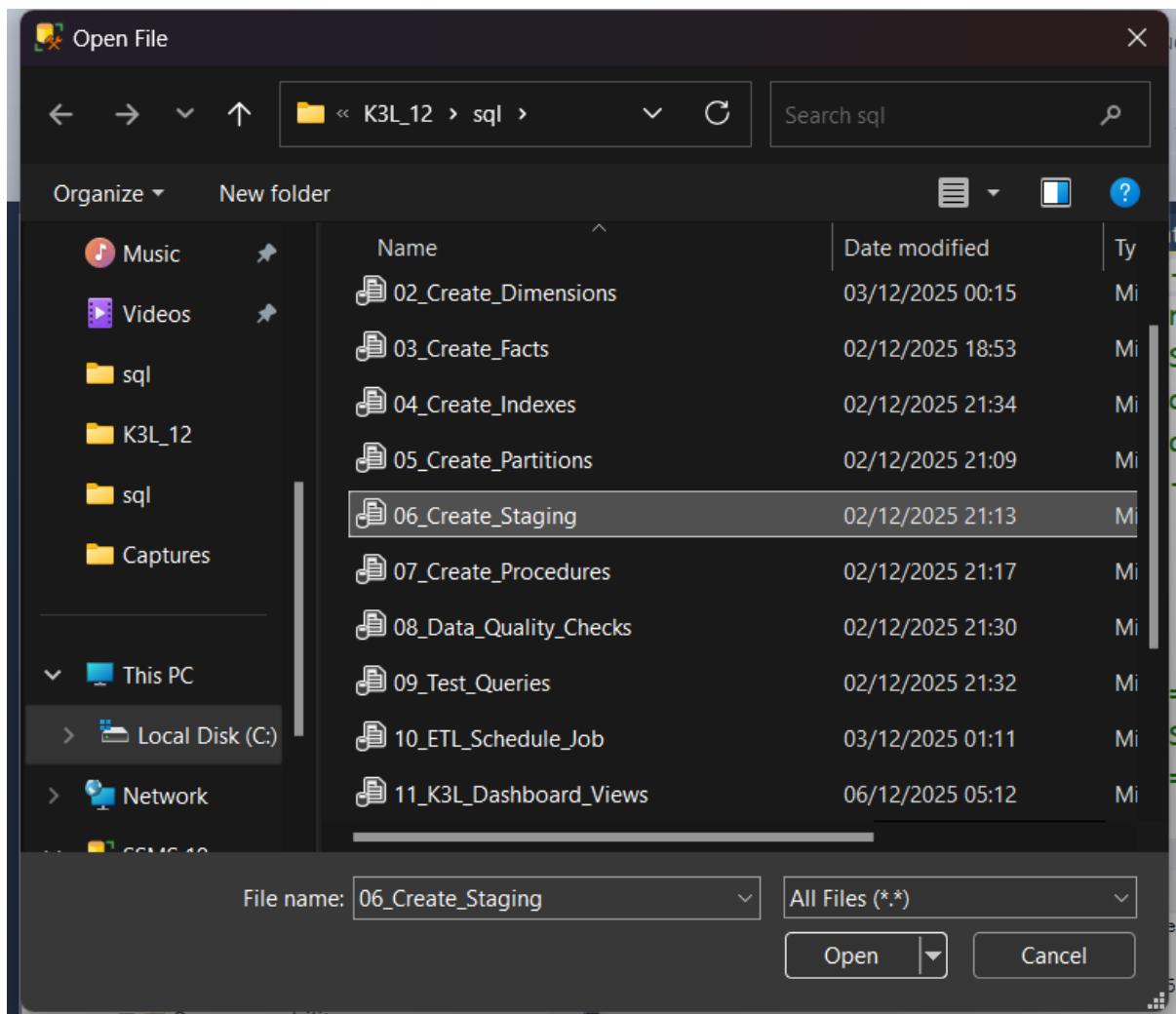
```
-- FILE: 05_Create_Partitions.sql
-- Deskripsi: Partitioning Strategy Data Mart K3L ITERA
-- Membuat partition function, scheme, fact tables partitioned
-- Clustered Columnstore Index dibuat langsung

USE K3L_DataMart;
GO

/* =====
 A. PARTITION FUNCTION & SCHEME
===== */

131 % 4
Messages
Partition scheme 'PS_Year' has been created successfully. 'PRIMARY' is marked as the next used filegroup in partition scheme 'PS_Year'.
Completion time: 2025-12-08T16:05:38.8547286+07:00
```

Open file 06_Create_Staging



Excute code & menampilkan output

```
06_Create_Staging(72)  x 05_Create_Partitions(AGUNG\Agung (71))  04_Create_Indexes..._AGUNO\Agung (54)  03_Create_Facts...(AGUNG\Agung (66))

-- FILE: 06_Create_Staging.sql
-- Deskripsi: Membuat Staging Schema dan Staging Tables untuk Data Mart K3L
-- ETL Step: Extract -> Staging

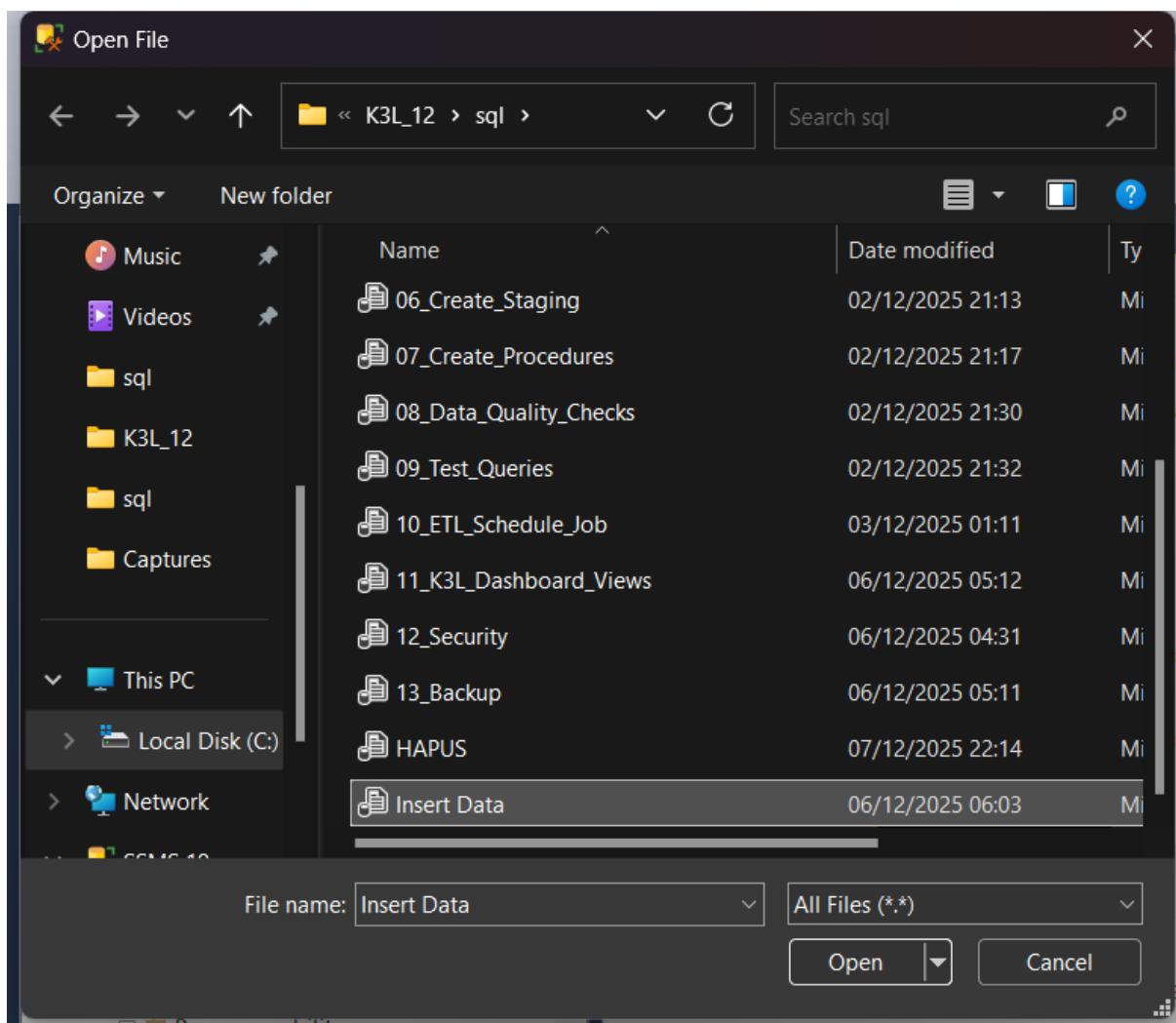
USE K3L_DataMart;
GO

/*
=====
A. CREATE STAGING SCHEMA
=====
*/
CREATE SCHEMA stg;
GO

131 %  |  Messages
Commands completed successfully.

Completion time: 2025-12-08T16:06:37.4385406+07:00
```

Insert data



Excute dan menampilkan output

The screenshot shows the SSMS interface with the following details:

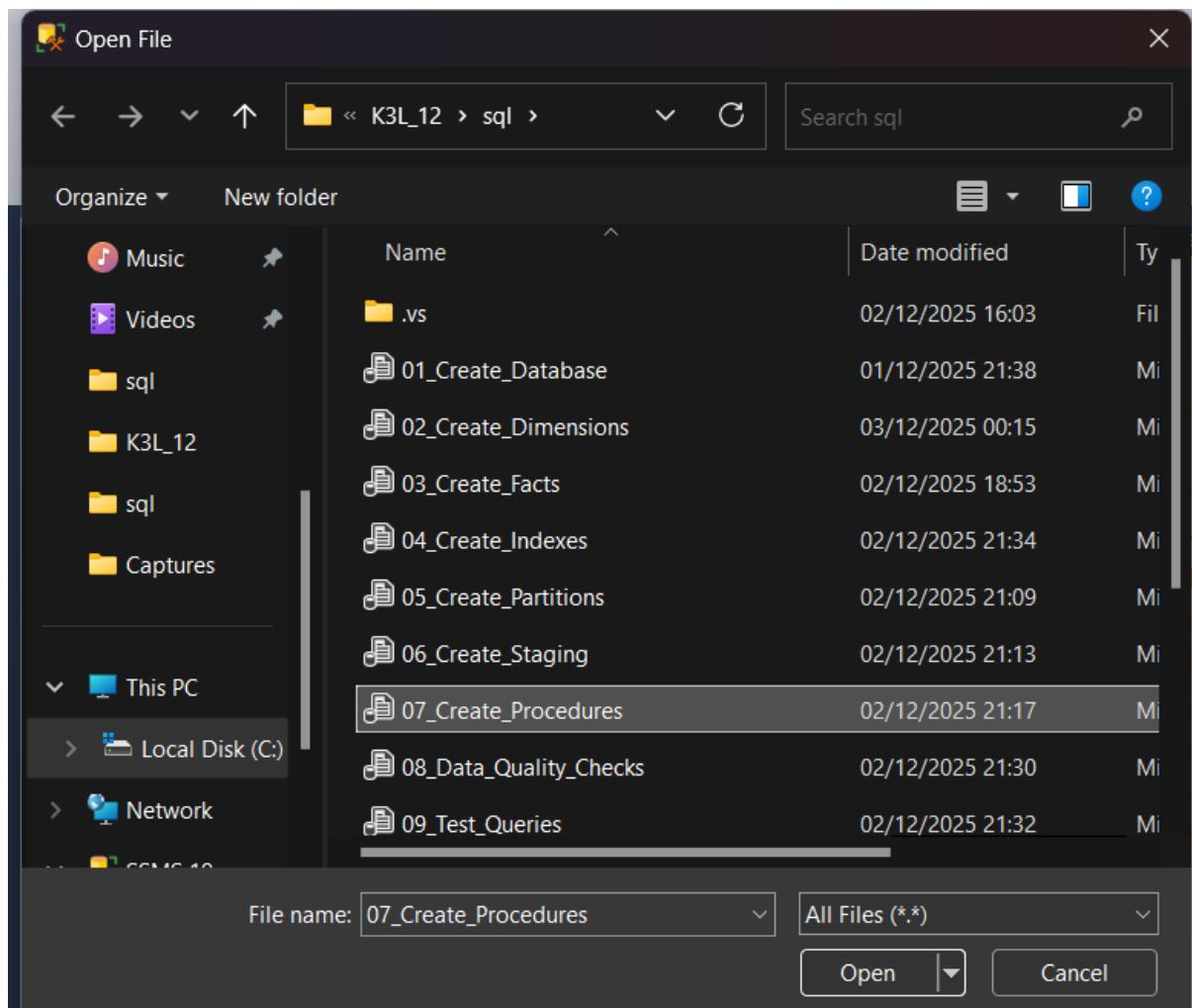
- Object Explorer:** Shows the database structure with objects like '06_Create_Staging', '05_Create_Partitions', and '04_Create_Indexes'.
- Script Editor:** Contains the following T-SQL code:

```
USE K3L_DataMart;
GO

-- =====
-- 1. DIM_DATE (50 data: 2021-2025)
-- =====

SET IDENTITY_INSERT Dim_Date ON;
INSERT INTO Dim_Date (DateID, Tanggal, Tahun, Bulan, Kuartal) VALUES
(1, '2021-01-05', 2021, 'Januari', 'Q1'),
(2, '2021-02-15', 2021, 'Februari', 'Q1'),
(3, '2021-03-10', 2021, 'Maret', 'Q1'),
(4, '2021-04-20', 2021, 'April', 'Q2'),
(5, '2021-05-05', 2021, 'Mei', 'Q2'),
(6, '2021-06-15', 2021, 'Juni', 'Q2'),
```
- Results Tab:** Displays the results of the executed query, showing the count of rows inserted into the 'Dim_Date' table.
- Messages Tab:** Shows two status messages: 'TIDAK ADA DATA KEMBAR' and 'DATA UNIK SEMUA'.

Create_Procedures



Excute dan menampilkan output

The screenshot shows a SQL Server Management Studio (SSMS) window with multiple tabs at the top: '07_Create_Procedures...AGUNG\Agung (55)', 'Insert Data.sql - ..(AGUNG\Agung (74))', '06_Create_Staging...AGUNG\Agung (72)', and '05_Create_Partitions...AGUNG\Agung (71)'. The main pane displays a T-SQL script for creating a stored procedure named 'usp_Load_FactInsiden'. The script includes comments indicating it's for ETL implementation and specifies the use of the 'K3L_DataMart' database. The execution completed successfully with a completion time of 2025-12-08T16:09:11.042Z.

```
-- FILE: 07_ETL_Load_Facts.sql
-- Deskripsi: ETL Implementation - Load Fact Tables (K3L)

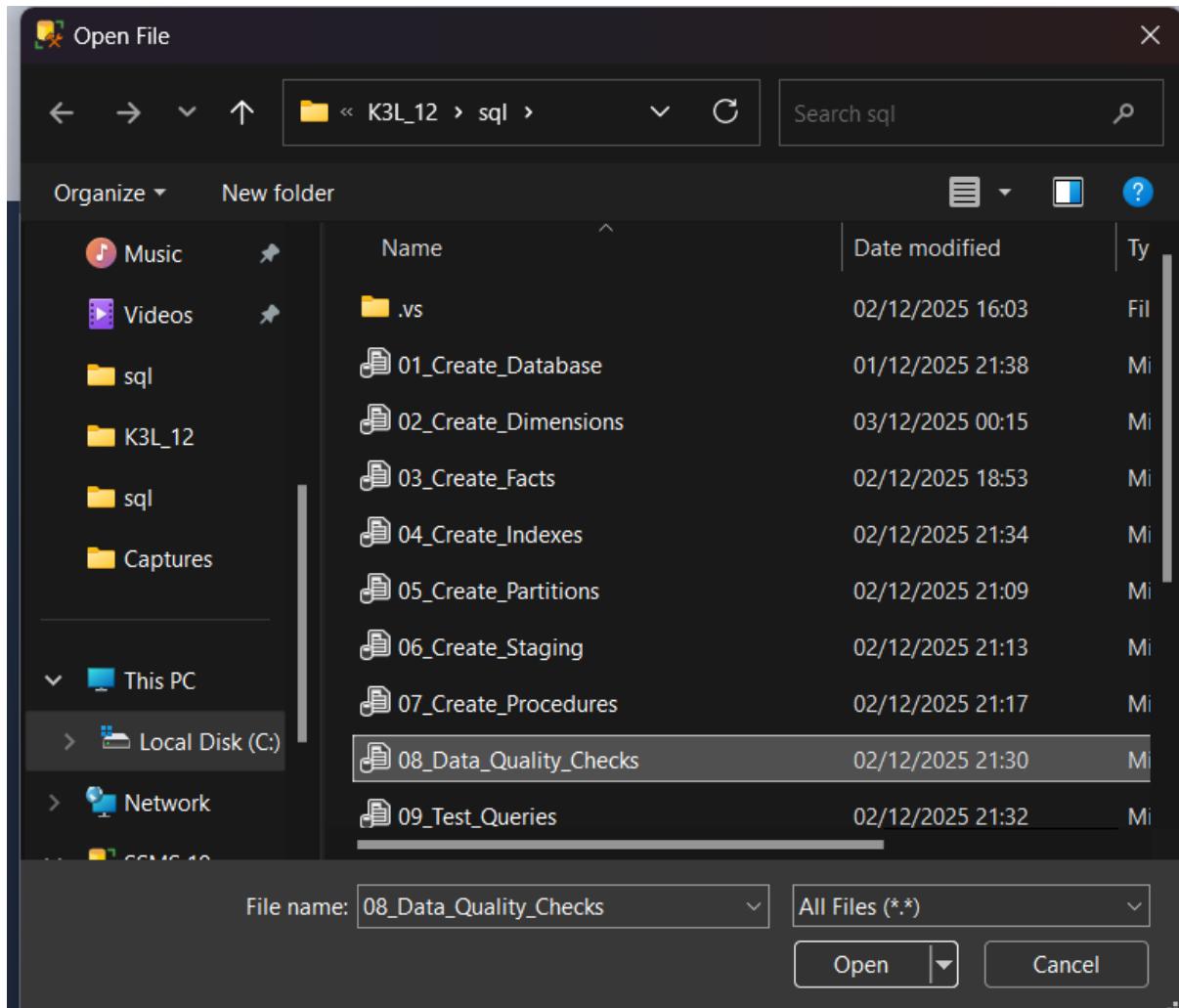
USE K3L_DataMart;
GO

/*
=====
1. Load Fact_Insiden_Partitioned
=====
*/
CREATE OR ALTER PROCEDURE usp_Load_FactInsiden
AS
BEGIN
131 % Messages
Commands completed successfully.

Completion time: 2025-12-08T16:09:11.042Z
07 ET

```

Data_Quality_Check



Excute dan menampilkan output

The screenshot shows the execution of a SQL script named 08_Data_Quality_Checks.sql in SQL Server Management Studio. The script performs data quality checks on three fact tables: Fact_Insiden, Fact_Inspeksi, and Fact_Limbah. The results are displayed in three separate result sets.

Result Set 1: Fact_Insiden

```
USE K3L_DataMart;
GO

/* =====
1. Fact_Insiden Summary
===== */

SELECT
    'Fact_Insiden' AS TableName,
```

TableName	TotalRecord	NullDateID	NullLokasiID	NullUnitKerjaID	NullJenisInsidenID	NullSeverityID	NullPetugasID
Fact_Insiden	50	0	0	0	0	0	0

CheckName	OrphanCount
Fact_Insiden_DateID	0
Fact_Insiden_LokasiID	0
Fact_Insiden_UnitKerjaID	0
Fact_Insiden_JenisInsidenID	0
Fact_Insiden_SeverityID	0
Fact_Insiden_PetugasID	0

Result Set 2: Fact_Inspeksi

TableName	TotalRecord	NullDateID	NullLokasiID	NullUnitKerjaID	NullPeralatanID	NullPetugasID
Fact_Inspeksi	50	0	0	0	0	0

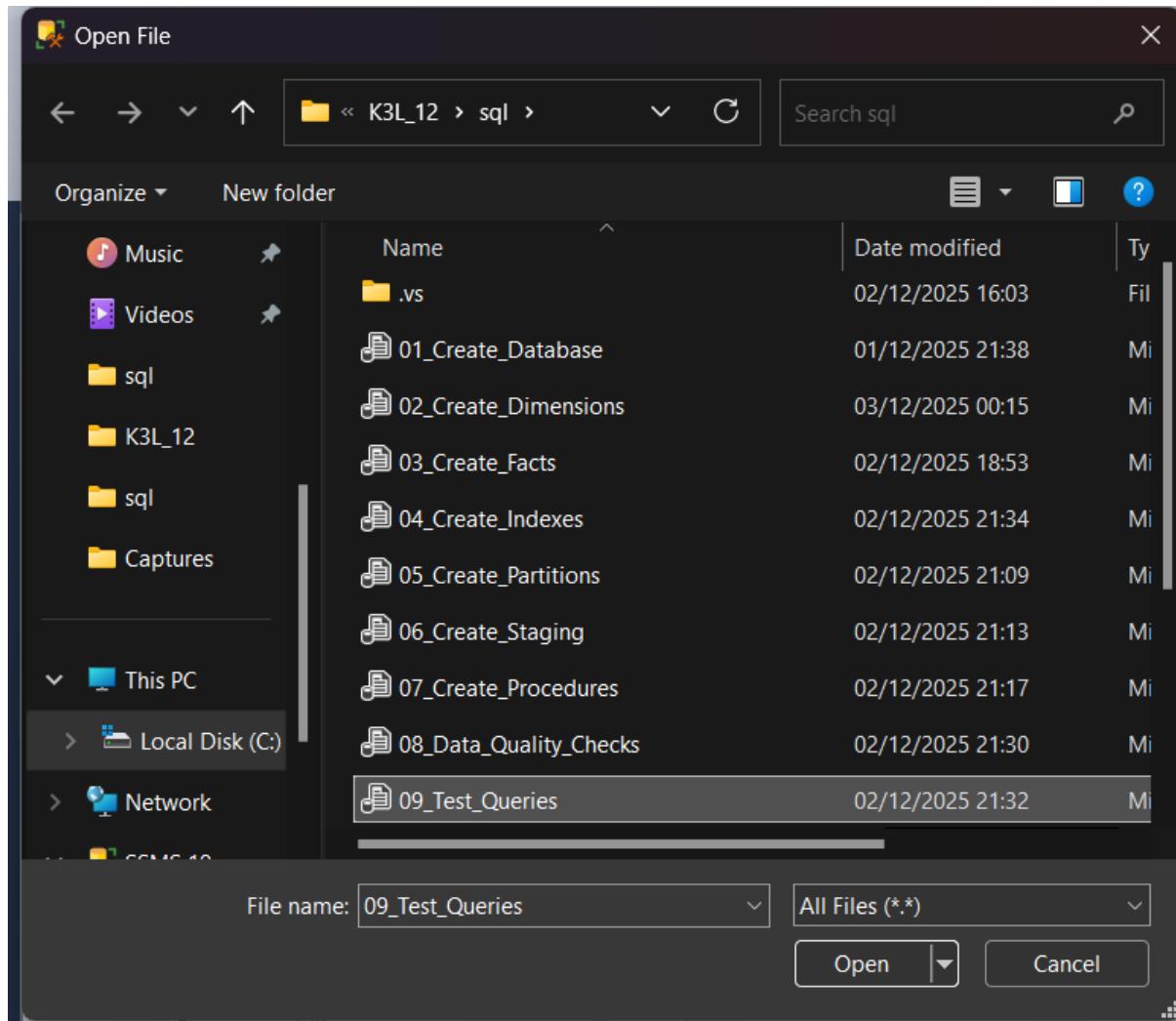
CheckName	OrphanCount
Fact_Inspeksi_DateID	0
Fact_Inspeksi_LokasiID	0
Fact_Inspeksi_UnitKerjaID	0
Fact_Inspeksi_PeralatanID	0
Fact_Inspeksi_PetugasID	0

Result Set 3: Fact_Limbah

TableName	TotalRecord	NullDateID	NullLokasiID	NullUnitKerjaID	NullLimbahID	NullPetugasID
Fact_Limbah	50	0	0	0	0	0

CheckName	OrphanCount
Fact_Limbah_DateID	0
Fact_Limbah_LokasiID	0
Fact_Limbah_UnitKerjaID	0
Fact_Limbah_LimbahID	0
Fact_Limbah_PetugasID	0

Test_Qualities



Execute dan menampilkan output

The screenshot shows a SQL Server Management Studio (SSMS) window with the following details:

- Object Explorer:** Shows multiple open queries and a file named "Insert Data.sql".
- Script pane:**

```
-- FILE: 09_Performance_Testing.sql
-- Deskripsi: Performance Testing untuk K3L Data Mart
-- Tujuan: Menguji dan mengoptimalkan performa query

USE K3L_DataMart;
GO

-- Aktifkan statistik waktu dan IO untuk analisis performa
SET STATISTICS TIME ON;
SET STATISTICS IO ON;
GO
```
- Results pane:**

131 %

Lokasi	TotalInsiden	TotalKorban	RataHariHilang
Labkom Prodi 2	1	1	0
Ruang Laboran	1	1	3
Area Parkir Selatan	1	2	5
Ruang Seminar B	1	3	15
Ruang Seminar A	1	1	7
Ruang TUFITIK	1	2	10
Lab Biologi	1	3	15
Lab Kimia Dasar	1	1	1

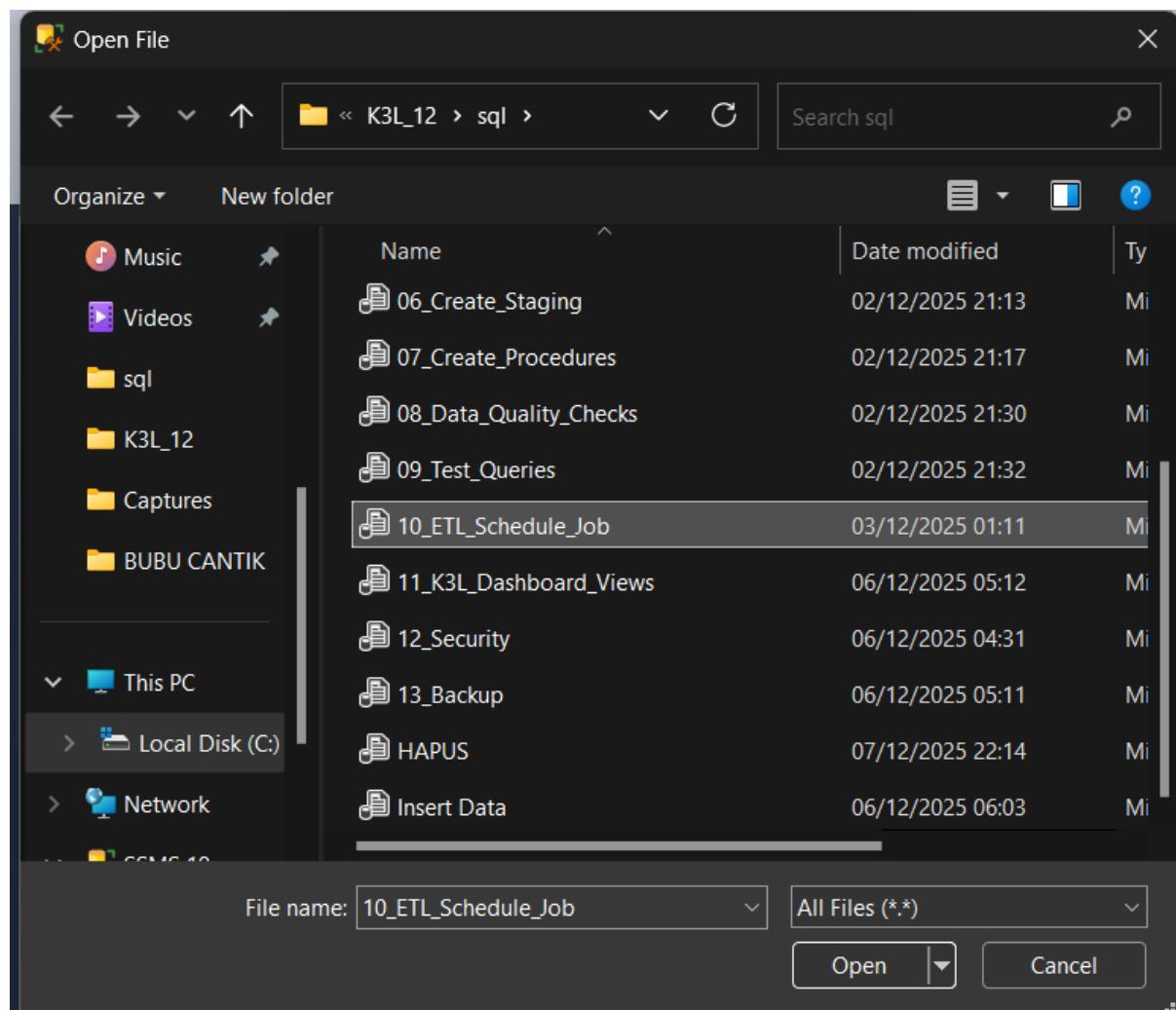
UnitKerja	TotalInspeksi	TotalTidakSesuai	RataDilinspeksi
Hukum	4	15	
Audit Internal	4	15	
Dekanat FTIK	4	15	
Alumni	4	14	
Pemeliharaan	4	14	
Kerjasama	4	15	
Matematika	4	14	

	Lokasi	JenisLimbah	TotalLimbah
1	Ruang TU FTSL	Limbah Radioaktif	30.00
2	Ruang UKM Olahraga	Baterai Bekas	30.00
3	Ruang Dosen Rektor...	Limbah Farmasi	25.00
4	Kantin Utara	Limbah Elektronik	25.00
5	Lab. Biologi	Jarum Suntik	20.00
6	Area Parkir Timur	Koran Bekas	20.00
7	Gudang Bahan Kimia	Besi Tua	20.00
8	Ruang Mushola	Debu Industri	20.00

	Severity	Bulan	TotalInsiden
1	1 - Rendah	Agustus	2
2	1 - Rendah	Februari	1
3	1 - Rendah	Januari	2
4	1 - Rendah	Juli	1
5	1 - Rendah	Juni	2
6	1 - Rendah	Novem...	1
7	2 - Ringan	Dese...	1
8	2 - Ringan	Februari	2

	Petugas	TotalInsiden	TotalInspeksi	TotalLimbah
1	Zainal Abidin	1	1	1
2	Umar Said	1	1	1
3	Gory Sandi...	1	1	1
4	Johan Seti...	1	1	1
5	Mega Chan	1	1	1
6	Eka Chandra	1	1	1
7	Bima Sakti	1	1	1
8	Budi Santoso	1	1	1

ETL_Schedule_Job



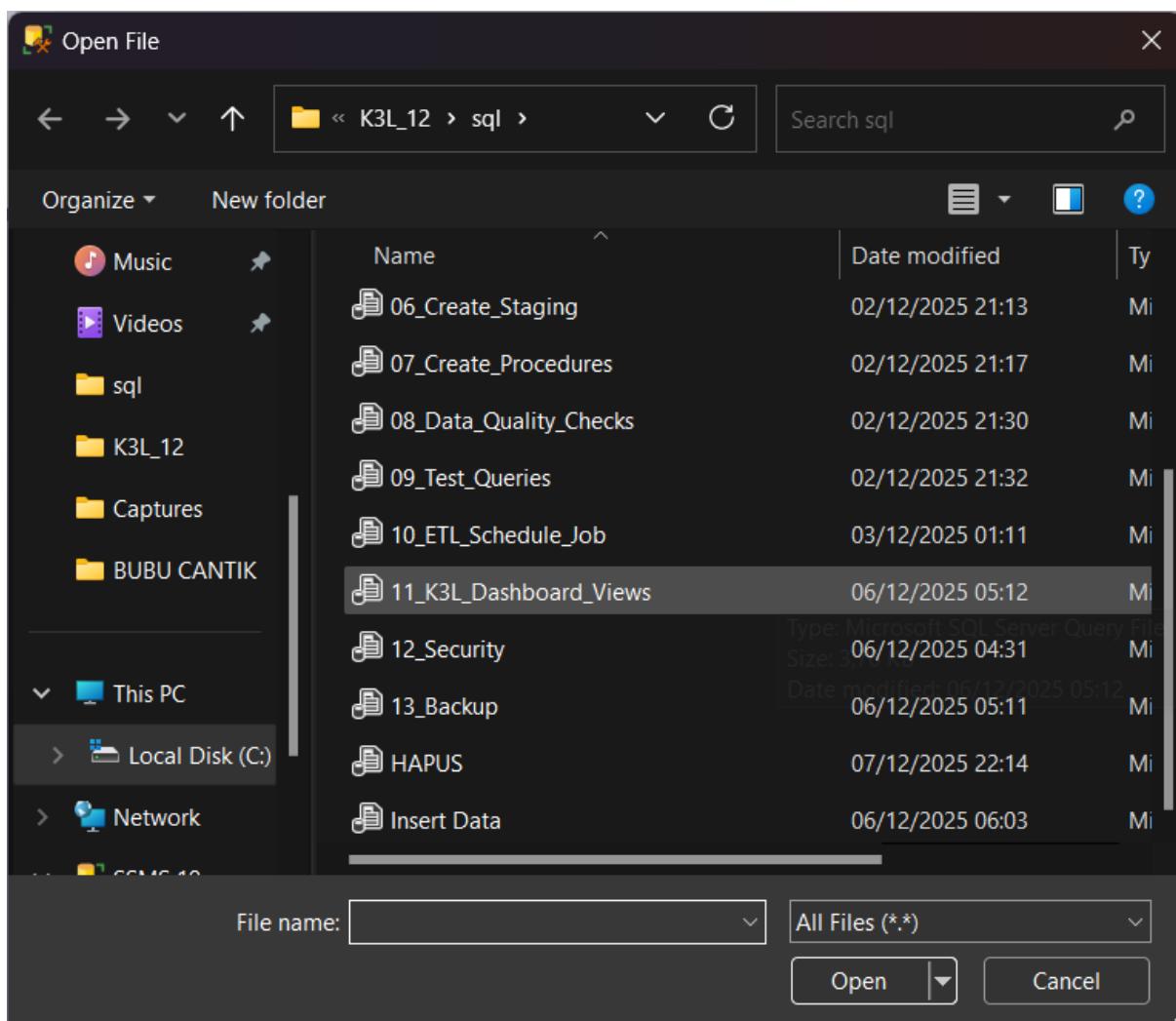
Excute dan menampilkan output

```
Messages
Commands completed successfully.

Completion time: 2025-12-08T21:23:42.0611042+07:00

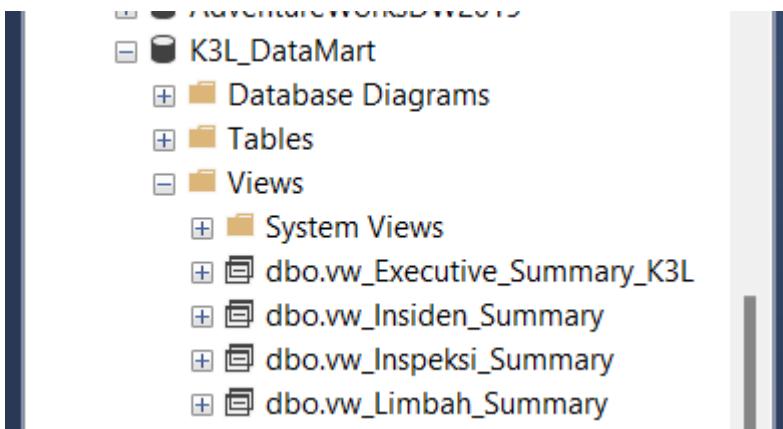
[+] dbo.sysjobactivity
[+] dbo.sysjobhistory
[+] dbo.sysjobs
[+] dbo.sysjobschedules
[+] dbo.sysjobservers
[+] dbo.sysjobsteps
[+] dbo.sysjobstepslogs
```

K3L_Dashboard_View



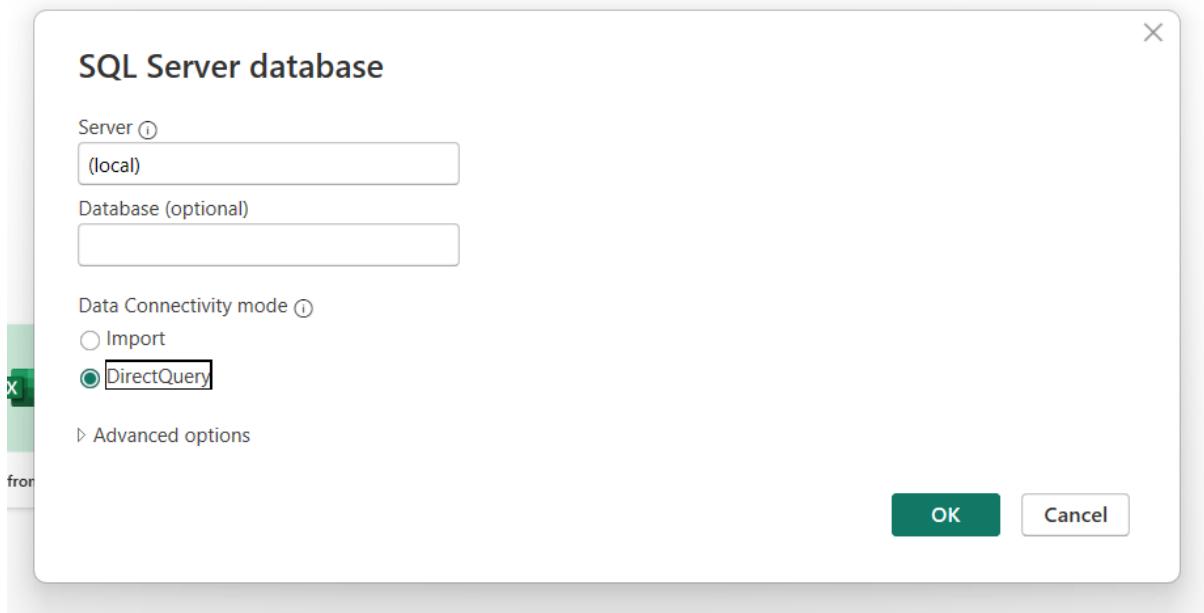
Commands completed successfully.

Completion time: 2025-12-08T21:28:00.8594667+07:00



Connect ke Power BI

The screenshot shows the Microsoft Power BI desktop application. The ribbon at the top includes tabs for File, Home, Insert, Modeling, View, Optimize, and Help. The Home tab is selected. The main workspace displays a 'Getting started' diagram with various icons representing data sources and visualization types. Below it, a callout box says 'Intro—What is Power BI?'. The 'Common data sources' pane on the left lists options like Excel workbook, Power BI semantic models, Dataflows, Dataverse, SQL Server, Analysis Services, Text/CSV, Web, OData feed, Blank query, and Power BI Template Apps. A tooltip for 'Import data from a SQL Server database.' points to the 'SQL Server' option. At the bottom, there's a 'Get data from another source →' button with options for 'Get data from Excel', 'Import data from SQL Server', 'Paste data into a blank table', and 'Use sample data'.



Navigator

Display Options ▾

- ◀ (local) [3]
 - ▶ AdventureWorks2019
 - ▶ AdventureWorksDW2019
- ◀ K3L_DataMart [21]
 - vw_Executive_Summary_K3L
 - vw_Insiden_Summary
 - vw_Inspeksi_Summary
 - vw_Limbah_Summary
 - Dim_Date
 - Dim_JenisInsiden
 - Dim_JenisLimbah
 - Dim_JenisPeralatan
 - Dim_Lokasi
 - Dim_Petugas
 - Dim_Severity
 - Dim_UnitKerja
 - Fact_Insiden
 - Fact_Insiden_Partitioned
 - Fact_Inspeksi
 - Fact_Inspeksi_Partitioned

vw_Limbah_Summary

Preview downloaded on Wednesday

Tahun	NamaUnit	NamaLokasi	JenisLimbah
2021	Teknik Kimia	Ruang Kelas A202	Asam Kuat
2021	Teknik Informatika	Labkom Prodi 1	Limbah B3 Padat
2023	Ekonomi	Ruang TU FTSL	Limbah Radioaktif
2025	Alumni	Ruang Laboran	Kayu Bekas
2022	Farmasi	Ruang Dosen FTIK	Spesimen Biologi
2023	Perpustakaan	Ruang Rapat 2	Kantong Plastik
2024	SDM	Gudang Bahan Kimia	Besi Tua
2024	IT Support	Ruang Server	Botol Kaca
2021	Teknik Mesin	Labkom Prodi 3	Limbah B3 Gas
2025	Penerimaan	Ruang Teknisi	Limbah Tekstil
2025	Internasional	Ruang Konseling	Batu Bata Patah
2022	Teknik Geologi	Lab. Kimia Dasar	Sianida
2022	Kimia	Lab. Fisika Dasar	Arsen
2022	Matematika	Lab. Biologi	Jarum Suntik
2022	Statistika	Lab. Mikrobiologi	Perban Bekas
2024	Kerjasama	Area Parkir Barat	Sampah Logam
2025	Rektorat	Ruang Security	Serbuk Kayu
2025	Dekanat FTIK	Ruang Mushola	Debu Industri
2023	Akuntansi	Ruang TU FTIK	Vaksin Bekas
2023	Sosiologi	Perpustakaan FTIK	Sampah Organik
2024	Kemahasiswaan	Area Parkir Timur	Koran Bekas
2024	Logistik	Gudang Umum	Sampah Kaca

Select Related Tables Load Transform Data Cancel

Tampilan Dashboard

K3L DASHBOARD — EXECUTIVE SUMMARY

Monitoring Insiden, Korban, dan Lost Workdays

50
Sum of TotalInsiden

227
Sum of TotalHariKerja...

60
Sum of TotalKorbanIn...

Total Insiden Selama 5 Tahun

Tahun	Sum of TotalInsiden	Sum of TotalKorban	Sum of TotalHariKerjaHilang	Severity Rate
2021	7	6	24	3.43
2024	9	13	53	5.89
2022	10	13	47	4.70
2023	10	12	45	4.50
2025	14	16	58	4.14
Total	50	60	227	4.54

Total Insiden Berdasarkan Nama

Total Korban Berdasarkan Unit

Unit	Sum of TotalKorban
Matematika	3
Pemeliharaan	3
Public Relations	3
Rektorat	3
Administrasi Umum	2

EXECUTIVE SUMMARY ACADEMIC PERFORMANCE FINANCIAL ANALYSIS

K3L DASHBOARD — ACADEMIC PERFORMANCE

Analisis Kinerja K3L Per Unit Kerja

Tingkat Kepatuhan Per Unit Kerja

NamaUnit	Sum of TotalDiinspeksi	Sum of TotalTidakSesuai
Audit Internal	1	18
Dekanat FTIK	1	18
Hukum	1	15
Kerjasama	1	18
Teknik Industri	1	15
Alumni	1	18

Sebaran Tingkat Keparahan Per Unit Kerja

NamaLokasi	1 - Rendah	2 - Ringan	3 - Sedang	4 - Signifikan
Ruang UKM Seni	1	1	1	1
Ruang UKM Sains	1	1	1	1
Ruang UKM Olahraga	1	1	1	1
Ruang TU FTSL	1	1	1	1
Ruang TU FTIK	1	1	1	1
Ruang Teknisi	1	1	1	1
Ruang Server	1	1	1	1
Ruang Seminar B	1	1	1	1
Total	9	9	8	8

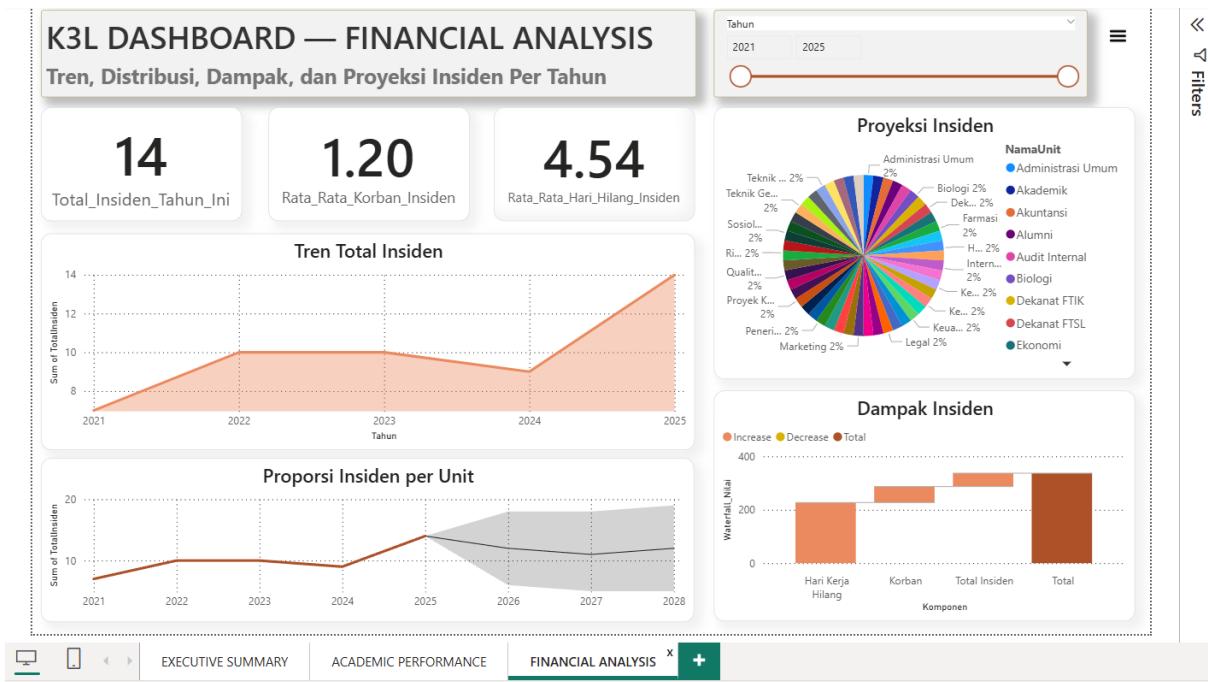
Tabel Unit Kerja dengan Kinerja Terbaik

NamaUnit	Sum of TotalInsiden	Sum of TotalTidakSesuai	Sum of TotalLimbah	Compliance Rate (%)
Administrasi Umum	1	90	453.70	0.83
Akademik	1	90	453.70	0.83
Akuntansi	1	90	453.70	0.83
Alumni	1	90	453.70	0.83
Audit Internal	1	90	453.70	0.83
Biologi	1	90	453.70	0.83
Dekanat FTIK	1	90	453.70	0.83
Dekanat FTSL	1	90	453.70	0.83
Ekonomi	1	90	453.70	0.83
Total	50	90	453.70	0.83

Capaian Target Kinerja K3L Tahunan

0.83 / 0.95

EXECUTIVE SUMMARY ACADEMIC PERFORMANCE FINANCIAL ANALYSIS



Security

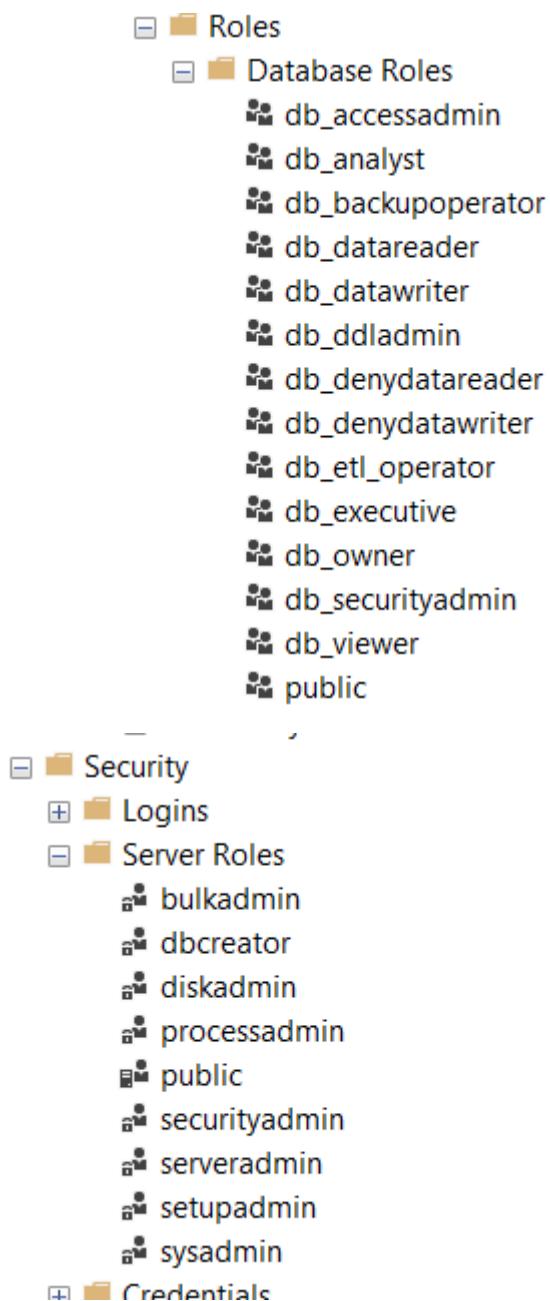
```

12_Security.sql - ... (AGUNG\Agung (67))  × 11_K3L_Dashboard...GUNG\Agung (62)  SQLQuery2.sql - ... (AGUNG\Agung (64))*  HAPUS.sql - (local... (AGUNG\Agung (51))
/*****************************************************************************************************
SECURITY IMPLEMENTATION - K3L_DataMart
******/
USE K3L_DataMart;
GO

/*=====
1.1. CREATE USER ROLES
=====*/
-- Create Roles
CREATE ROLE db_executive;
CREATE ROLE db_analyst;
CREATE ROLE db_viewer;

```

131 %



Backup

```
13_Backup.sql - (...(AGUNG\Agung (68)) 12_Security.sql - (..(AGUNG\Agung (67)) 11_K3L_Dashboard...GUNG\Agung (62)) SQLQuery2.sql - (...(AGUNG\Agung (64))*
BACKUP DATABASE K3L_DataMart
TO DISK = N'C:\Backup\K3L_DataMart_Full.bak'
WITH
    COMPRESSION,
    INIT,
    NAME = N'Full Backup K3L_DataMart',
    STATS = 10;
GO
BACKUP DATABASE K3L_DataMart
TO DISK = N'C:\Backup\K3L_DataMart_Diff.bak'
WITH
    DIFFERENTIAL,
    COMPRESSION,
    INIT,
131 %
```

Messages

```
11 percent processed.
22 percent processed.
34 percent processed.
40 percent processed.
50 percent processed.
60 percent processed.
70 percent processed.
80 percent processed.
90 percent processed.
100 percent processed.
Processed 1128 pages for database 'K3L_DataMart', file 'K3L_DataMart_Data' on file 1.
Processed 2 pages for database 'K3L_DataMart', file 'K3L_DataMart_Log' on file 1.
BACKUP DATABASE successfully processed 1130 pages in 0.076 seconds (116.108 MB/sec).
32 percent processed.
43 percent processed.
54 percent processed.
64 percent processed.
```

