Keamanan Siber

Kelompok 9

Anggota

Fadil Prasetyo Alfarizzi (123450048)

Reynaldi Rahmad (122450088)

Melinza Nabila (123450122)

Fathya Intami Gusda (123450095)

MISI 1

Business Requirements Analysis

Tujuan : Memahami kebutuhan bisnis dan analitik dari unit kerja keamanan siber untuk pengelolaan insiden keamanan (cyber incident) pada sistem jaringan organisasi.

1. Identifikasi Stakeholders

Penggunaan Utama Data Mart :

- Manajer Keamanan Informasi
- Koordinator Tim Respon Insiden
- Staff Keamanan IT
- Pimpinan Fakultas/Unit TI

Decision Makers

- Kepala Pusat Data & Keamanan TI
- Rektor atau Wakil Rektor bidang TI

2. Analisis Proses Bisnis

Proses bisnis utama yang terjadi dalam penanganan insiden keamanan meliputi:

1. Deteksi Insiden oleh sistem monitoring.
2. Klasifikasi Jenis Serangan (DDoS, Malware, SQL Injection, dll).
3. Pencatatan Lokasi & Server terdampak.
4. Respon oleh Tim Penanganan (Incident Response Team).
5. Pelaporan hasil dan waktu penyelesaian.

Key Performance Indicators (KPI):

- Rata-rata durasi respon insiden.
- Jumlah insiden per bulan.
- Frekuensi serangan per fakultas/unit.
- Jumlah data yang terdampak.
- Persentase insiden yang diselesaikan tepat waktu.

Metrik Bisnis:

- Durasi respon (menit).
- Jumlah data terdampak (file/data).
- Status penyelesaian (selesai/pending).
- Frekuensi serangan (count per periode).

Business Requirements Analysis

3. Kebutuhan Analitik

Pertanyaan Bisnis yang perlu dijawab:

- Berapa banyak insiden keamanan yang terjadi setiap bulan?
- Fakultas atau unit mana yang paling sering terkena serangan?
- Jenis serangan apa yang paling sering terjadi?
- Berapa lama rata-rata waktu yang dibutuhkan untuk menangani setiap insiden?
- Siapa anggota tim yang paling cepat merespon serangan?

Jenis Laporan yang Dibutuhkan:

- Laporan harian: daftar insiden yang baru terjadi dan status penyelesaiannya.
- Laporan mingguan: tren jenis serangan dan aktivitas tim respon.
- Laporan bulanan: evaluasi performa tim, jumlah serangan per lokasi, dan durasi respon rata-rata.

Level Agregasi:

- Harian → Mingguan → Bulanan
- Fakultas → Institusi
- Jenis Serangan → Kategori Serangan

Conceptual Design

Tujuan : Membuat model konseptual yang merepresentasikan entitas bisnis dan hubungan antar entitas pada sistem penanganan insiden keamanan.

1.Identifikasi Entitas

Entitas Utama :

- Incident
- Waktu
- Lokasi
- Jenis_Serangan
- Tim_Respon

Atribut Utama Tiap Entitas

Entitas	Atribut	Primary/Foreign Key
Incident	ID_Incident, WaktuKey, LokasiKey, JenisSeranganKey, TimResponKey, DurasiRespon, StatusPenanganan, JumlahDataTerenam	PK: ID_Incident
Waktu	WaktuKey, Tanggal, Bulan, Tahun, Hari	PK: WaktuKey
Lokasi	LokasiKey, Unit, Fakultas, Server	PK: LokasiKey
Jenis_Serangan	JenisSeranganKey, NamaSerangan, Deskripsi	PK: JenisSeranganKey
Tim_Respon	TimResponKey, Petugas, Jabatan, Unit	PK: TimResponKey

Conceptual Design

2. Definisi Relationship

Relasi	Kardinalitas	Jenis
Waktu ↔ Incident	1 – N	One-to-Many
Lokasi ↔ Incident	1 – N	One-to-Many
Jenis_Serangan ↔ Incident	1 – N	One-to-Many
Tim_Respon ↔ Incident	1 – N	One-to-Many

Logical Design

Identifikasi Fact Table

- Business Process
- Grain
- Measures
- Additive

Identifikasi Dimension Table

Dimensi	Fokus Analisis	Contoh Atribut
Dim_Waktu	Kapan insiden terjadi	Tanggal, Bulan, Tahun, Hari
Dim_Lokasi	Di mana insiden terjadi	Unit, Fakultas, Server
Dim_Jenis_Serangan	Apa jenis serangan	NamaSerangan, Deskripsi
Dim_Tim_Respon	Siapa yang menangani	Petugas, Jabatan, Unit

Desain Star Schema

```
                Dim_Waktu
                    ↑
Dim_Lokasi ← Fact_Incident → Dim_Jenis_Serangan
                    ↓
               Dim_Tim_Respon
```

Foreign Key Relationships:

- Fact_Incident.WaktuKey → Dim_Waktu.WaktuKey
- Fact_Incident.LokasiKey → Dim_Lokasi.LokasiKey
- Fact_Incident.JenisSeranganKey → Dim_Jenis_Serangan.JenisSeranganKey
- Fact_Incident.TimResponKey → Dim_Tim_Respon.TimResponKey

Logical Design

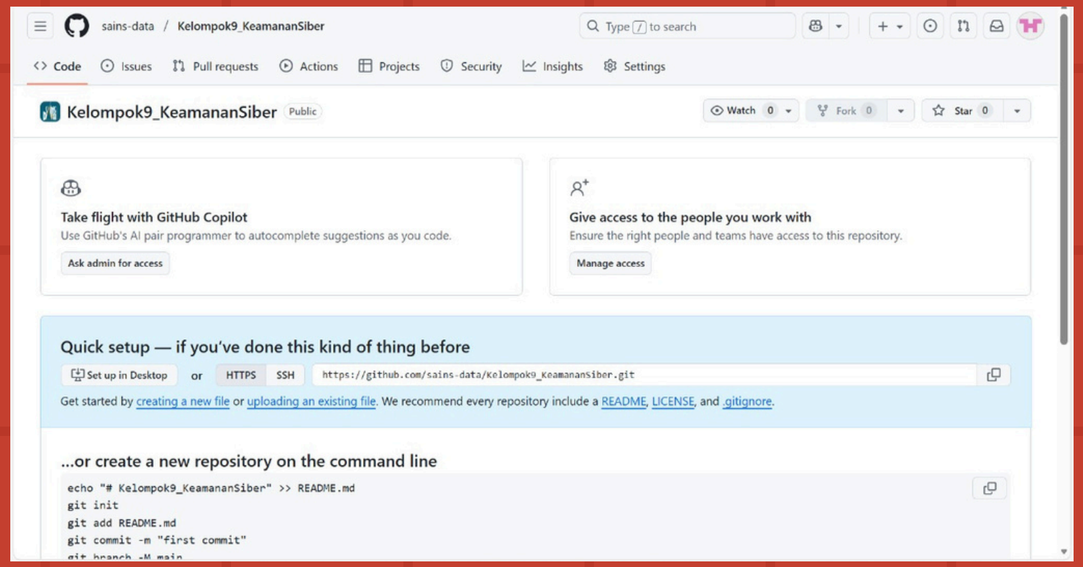
Penentuan Surrogate Keys

- Semua dimensi menggunakan integer surrogate key.
- Naming Convention : Dim_Waktu_SK, Dim_Lokasi_SK, Dim_Jenis_Serangan_SK, Dim_Tim_Respon_SK
- Strategi Slowly Changing Dimensions (SCD): Tipe 2 (History Tracking): bila data tim respon atau lokasi berubah, versi baru dibuat dengan SK baru agar histori tetap terjaga.

DATA DICTIONARY

Tabel	Kolom	Tipe Data	Deskripsi
Fact_Incident	ID_Incident	int (PK)	Kunci utama insiden
	WaktuKey	int (FK)	Referensi ke Dim_Waktu
	LokasiKey	int (FK)	Referensi ke Dim_Lokasi
	JenisSeranganKey	int (FK)	Referensi ke Dim_Jenis_Serangan
	TimResponKey	int (FK)	Referensi ke Dim_Tim_Respon
	DurasiRespon	int	Lama penanganan insiden
	JumlahDataTerenam	int	Banyaknya data terdampak
	StatusPenanganan	string	Status: selesai/pending
Dim_Waktu	WaktuKey	int (PK)	Surrogate key waktu
	Tanggal	date	Tanggal kejadian
	Bulan	int	Bulan ke-n
	Tahun	int	Tahun kejadian
	Hari	string	Nama hari
Dim_Lokasi	LokasiKey	int (PK)	Surrogate key lokasi
	Unit	string	Nama unit kerja
	Fakultas	string	Fakultas terkait
	Server	string	Nama server terdampak
Dim_Jenis_Serangan	JenisSeranganKey	int (PK)	Surrogate key jenis serangan
	NamaSerangan	string	Nama jenis serangan
	Deskripsi	string	Penjelasan singkat
Dim_Tim_Respon	TimResponKey	int (PK)	Surrogate key tim respon
	Petugas	string	Nama petugas
	Jabatan	string	Jabatan petugas
	Unit	string	Unit kerja tim

REPOSITORI

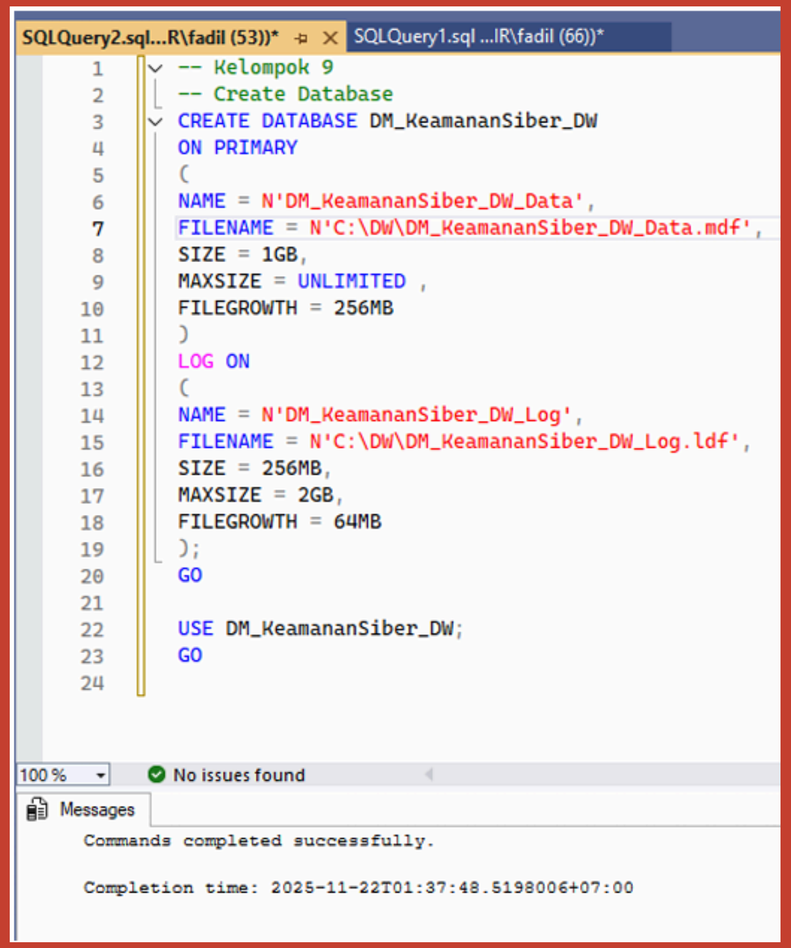


Link

https://github.com/sains-data/Kelompok9_KeamananSiber.git

Misi 2

Physical Database Design



Physical Database Design

Create Dimension Tabel

```sql
-- Kelompok 9
-- Dimension Tables for DM_KeamananSiber_DW


-- Dimension Table: Dim_Waktu
CREATE TABLE dbo.Dim_Waktu (
    WaktuKey INT PRIMARY KEY NOT NULL ,
    Tanggal DATE NOT NULL ,
    Bulan INT NOT NULL ,
    Tahun INT NOT NULL ,
    Hari VARCHAR(20) NOT NULL ,

    -- SCD Type 2 attributes
    EffectiveDate DATE NOT NULL DEFAULT GETDATE(),
    ExpiryDate DATE NULL,
    IsCurrent BIT NOT NULL DEFAULT 1,

    -- Metadata
    CreatedDate DATETIME DEFAULT GETDATE(),
    ModifiedDate DATETIME DEFAULT GETDATE(),

    CONSTRAINT CK_DimWaktu_Bulan CHECK (Bulan BETWEEN 1 AND 12),
    CONSTRAINT CK_DimWaktu_Tahun CHECK (Tahun BETWEEN 2023 AND 2025),
    CONSTRAINT CK_DimWaktu_Hari CHECK (Hari IN (
        'Monday','Tuesday','Wednesday','Thursday','Friday','Saturday','Sunday'
    ))
);
GO

-- Dimension Table: Dim_Lokasi
CREATE TABLE dbo.Dim_Lokasi (
    LokasiKey INT IDENTITY(1,1) PRIMARY KEY NOT NULL ,
    Fakultas VARCHAR(50) NOT NULL ,
    Server VARCHAR(50) NOT NULL ,

    -- SCD Type 2
    EffectiveDate DATE NOT NULL DEFAULT GETDATE(),
    ExpiryDate DATE NULL,
    IsCurrent BIT NOT NULL DEFAULT 1,

    -- Metadata
    CreatedDate DATETIME DEFAULT GETDATE(),
    ModifiedDate DATETIME DEFAULT GETDATE(),
```

```
Commands completed successfully.

Completion time: 2025-11-22T01:40:00.7757834+07:00
```

Create Fact Tabel

```sql
-- Kelompok 9
-- Fact Tables for DM_KeamananSiber_DW

-- Fact Table: Fact_Incident
CREATE TABLE dbo.Fact_Incident (
    ID_Incident INT PRIMARY KEY NOT NULL ,

    -- Foreign Keys
    WaktuKey INT NOT NULL ,
    LokasiKey INT NOT NULL ,
    JenisSeranganKey INT NOT NULL ,
    TimResponKey INT NOT NULL ,

    -- Measures
    DurasiRespon INT NOT NULL ,
    JumlahDataTerenam INT NOT NULL ,

    StatusPenanganan VARCHAR(20) NOT NULL ,

    -- Metadata
    CreatedDate DATETIME DEFAULT GETDATE(),
    ModifiedDate DATETIME DEFAULT GETDATE(),

    CONSTRAINT CK_FactIncident_Status CHECK (StatusPenanganan IN ('selesai','pending')) ,

    -- FK relationships
    CONSTRAINT FK_Fact_Waktu FOREIGN KEY (WaktuKey)
        REFERENCES dbo.Dim_Waktu(WaktuKey),

    CONSTRAINT FK_Fact_Lokasi FOREIGN KEY (LokasiKey)
        REFERENCES dbo.Dim_Lokasi(LokasiKey),

    CONSTRAINT FK_Fact_Serangan FOREIGN KEY (JenisSeranganKey)
        REFERENCES dbo.Dim_JenisSerangan(JenisSeranganKey),

    CONSTRAINT FK_Fact_Tim FOREIGN KEY (TimResponKey)
        REFERENCES dbo.Dim_TimRespon(TimResponKey)
);
GO
```

```
Commands completed successfully.

Completion time: 2025-11-22T01:40:40.2623460+07:00
```

Partitioning Strategy

SQLQuery2.sql....R\fadil (53))* | SQLQuery1.sql ...IR\fadil (66))*

```sql
-- Kelompok 9
-- Partitioning Setup for Fact Tables in DM_KeamananSiber_DW

IF EXISTS (SELECT * FROM sys.partition_schemes WHERE name = 'PS_Waktu')
    DROP PARTITION SCHEME PS_Waktu;
GO

IF EXISTS (SELECT * FROM sys.partition_functions WHERE name = 'PF_Waktu')
    DROP PARTITION FUNCTION PF_Waktu;
GO

-- Create Partition Function (by Calendar Year)
CREATE PARTITION FUNCTION PF_Waktu (INT)
AS RANGE RIGHT FOR VALUES
(
    20230101,  -- 2023
    20240101,  -- 2024
    20250101   -- 2025
);
GO

-- Create Partition Scheme
CREATE PARTITION SCHEME PS_Waktu
AS PARTITION PF_Waktu
ALL TO ([PRIMARY]);
GO


-- Create Partitioned Fact Table (tanpa PRIMARY KEY inline)
CREATE TABLE dbo.Fact_Incident_Partitioned (
    -- Hapus PRIMARY KEY di sini, akan dibuat terpisah sebagai Non-Clustered
    ID_Incident INT NOT NULL,

    -- Foreign keys
    WaktuKey INT NOT NULL, -- Kolom Partisi
    LokasiKey INT NOT NULL,
    JenisSeranganKey INT NOT NULL,
    TimResponKey INT NOT NULL,

    -- Measures
    DurasiRespon INT NOT NULL,
    JumlahDataTerenam INT NOT NULL,
    StatusPenanganan VARCHAR(20) NOT NULL,
```

100 % | 4 | 0

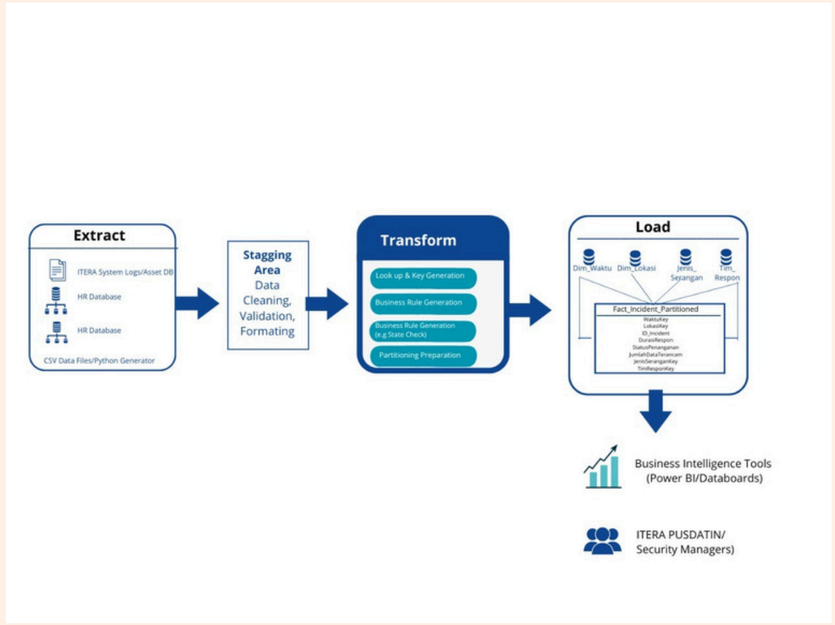
Messages

```
Partition scheme 'PS_Waktu' has been created successfully. 'PRIMARY' is marked as the next used filegroup in partition scheme 'PS_Waktu'.

Completion time: 2025-11-22T01:55:42.5122674+07:00
```

ETL DESIGN

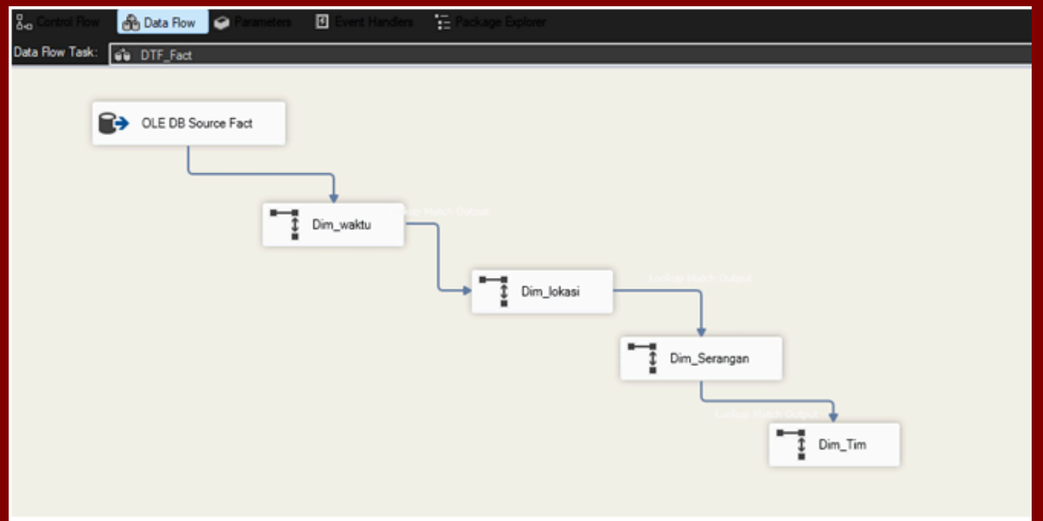
ETL Architecture Design

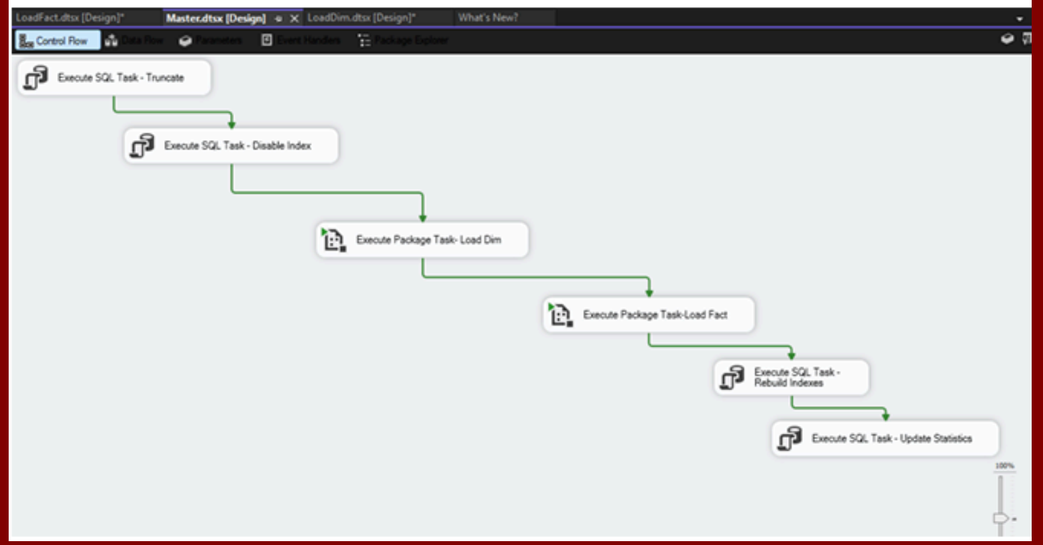


ETL Mapping

Source	Source Column	Target	Target Column
dbo.Dim_Waktu	WaktuKey	Dim_Waktu	WaktuKey
dbo.Dim_Waktu	Tanggal	Dim_Waktu	Tanggal
dbo.Dim_Waktu	Bulan	Dim_Waktu	Bulan
dbo.Dim_Waktu	Tahun	Dim_Waktu	Tahun
dbo.Dim_Waktu	Hari	Dim_Waktu	Hari
dbo.Dim_Lokasi	LokasiKey	Dim_Lokasi	LokasiKey
dbo.Dim_Lokasi	Fakultas	Dim_Lokasi	Fakultas
dbo.Dim_Lokasi	Server	Dim_Lokasi	Server
dbo.Dim_JenisSerangan	JenisSeranganKey	Dim_JenisSerangan	JenisSeranganKey
dbo.Dim_JenisSerangan	NamaSerangan	Dim_JenisSerangan	NamaSerangan
dbo.Dim_JenisSerangan	Deskripsi	Dim_JenisSerangan	Deskripsi
dbo.Dim_TimRespon	TimResponKey	Dim_TimRespon	TimResponKey
dbo.Dim_TimRespon	Petugas	Dim_TimRespon	Petugas
dbo.Dim_TimRespon	Jabatan	Dim_TimRespon	Jabatan
dbo.Dim_TimRespon	Unit	Dim_TimRespon	Unit
dbo.Fact_Incident	ID_Incident	Fact_Incident	ID_Incident
dbo.Fact_Incident	WaktuKey	Fact_Incident	WaktuKey
dbo.Fact_Incident	LokasiKey	Fact_Incident	LokasiKey
dbo.Fact_Incident	JenisSeranganKey	Fact_Incident	JenisSeranganKey
dbo.Fact_Incident	TimResponKey	Fact_Incident	TimResponKey
dbo.Fact_Incident	DurasiRespon	Fact_Incident	DurasiRespon
dbo.Fact_Incident	JumlahDataTerenam	Fact_Incident	JumlahDataTerenam
dbo.Fact_Incident	StatusPenanganan	Fact_Incident	StatusPenanganan

ETL Implementation





Data Quality Assurance

MSDAXQuery1....ot connected | SQLQuery1.sql...R\fadil (70))*

```sql
-- Pastikan Anda menjalankan script ini dengan koneksi ke database DM_KeamananSiber_DW

-- ============================================================
-- Check 1: Completeness - NULL values (Cek Dimensi & Fact)
-- ============================================================
SELECT
    'Dim_Lokasi' AS TableName,
    COUNT(*) AS TotalRows,
    SUM(CASE WHEN LokasiKey IS NULL THEN 1 ELSE 0 END) AS NullLokasiKey,
    SUM(CASE WHEN Fakultas IS NULL THEN 1 ELSE 0 END) AS NullFakultas,
    SUM(CASE WHEN Server IS NULL THEN 1 ELSE 0 END) AS NullServer
FROM dbo.Dim_Lokasi
UNION ALL
SELECT
    'Fact_Incident' AS TableName,
    COUNT(*) AS TotalRows,
    SUM(CASE WHEN ID_Incident IS NULL THEN 1 ELSE 0 END) AS NullIDIncident,
    SUM(CASE WHEN DurasiRespon IS NULL THEN 1 ELSE 0 END) AS NullDurasi,
    SUM(CASE WHEN StatusPenanganan IS NULL THEN 1 ELSE 0 END) AS NullStatus
FROM dbo.Fact_Incident;

---

-- ============================================================
-- Check 2: Consistency - Referential Integrity (Orphan Records)
-- ============================================================
```

100 % | 1 | 0

Results | Messages

	TableName	TotalRows	NullLokasiKey	NullFakultas	NullServer
1	Dim_Lokasi	0	NULL	NULL	NULL
2	Fact_Incident	0	NULL	NULL	NULL

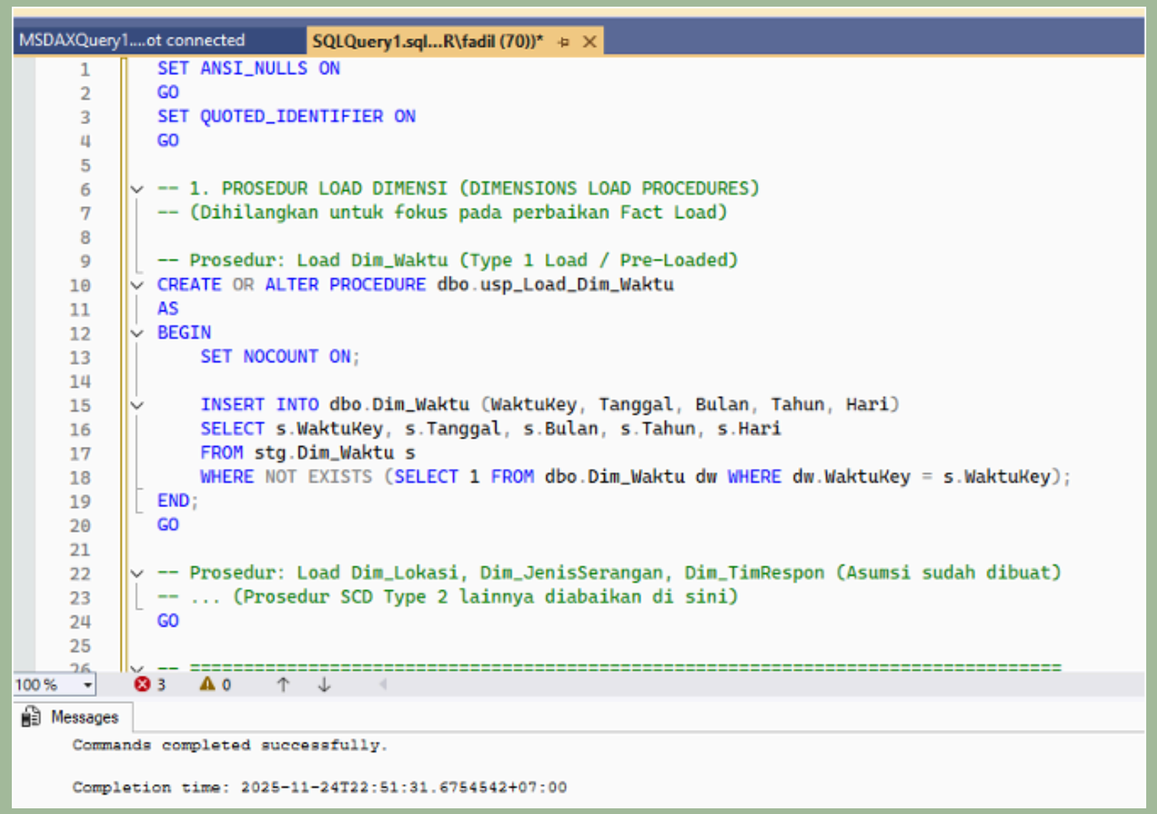
	OrphanKey	OrphanRecords
1	TimResponKey	0
2	LokasiKey	0

	TableName	InvalidDurasiRespon	InvalidStatusDomain
1	Fact_Incident	NULL	NULL

	ID_Incident	DuplicateCount

	DataSource	RecordCount
1	Staging	0
2	Warehouse	0

Performance Testing



Misi 3

Production Database Credentials

SQLQuery8.sql...IR\fadil (53))* | SQLQuery7.sql ...IR\fadil (81)) | SQLQuery6.sql ...IR\fadil (60)) | SQLQuery5.sql ...IR\fadil (95))

```
USE msdb;
GO

-- 1. Menambahkan Job Utama
EXEC sp_add_job
    @job_name = N'ETL_KeamananSiber_Daily_Load', -- Nama Job disesuaikan
    @enabled = 1,
    @description = N'Daily ETL load for DM_KeamananSiber_DW, executing the Master SP.';
GO

-- 2. Menambahkan Job Step (Langkah Eksekusi)
EXEC sp_add_jobstep
    @job_name = N'ETL_KeamananSiber_Daily_Load',
    @step_name = N'Execute Master ETL',
    @subsystem = N'TSQL',
    -- Perintah memanggil Master Stored Procedure Anda
    @command = N'EXEC dbo.usp_Master_ETL_KeamananSiber;',
    -- Ganti dengan nama database Data Warehouse Anda
    @database_name = N'DM_KeamananSiber_DW',
    @retry_attempts = 3,
    @retry_interval = 5;
GO

-- 3. Menambahkan Schedule (Jadwal)
EXEC sp_add_schedule
    @schedule_name = N'Daily at 02:00 AM',
```

100 % | No issues found

Messages

```
SQLServerAgent is not currently running so it cannot be notified of this action.

Completion time: 2025-12-01T21:48:06.2726713+07:00
```

Dashboard Development

SQLQuery1.sql...R\fadil (58))*

```sql
CREATE VIEW dbo.vw_Response_Team_Analytics AS
SELECT
    dtr.Unit,
    dw.Tahun,
    dw.Bulan,
    COUNT(f.ID_Incident) AS TotalIncidentsHandled,
    AVG(f.DurasiRespon) AS AvgDuration,
    SUM(f.JumlahDataTerenam) AS TotalDataImpacted,

    CAST(SUM(CASE WHEN f.DurasiRespon < 60 THEN 1 ELSE 0 END) * 100.0
    / NULLIF(COUNT(f.ID_Incident), 0) AS DECIMAL(5,2)) AS QuickResponseRate
FROM dbo.Fact_Incident f
INNER JOIN dbo.Dim_TimRespon dtr ON f.TimResponKey = dtr.TimResponKey
INNER JOIN dbo.Dim_Waktu dw ON f.WaktuKey = dw.WaktuKey
GROUP BY dtr.Unit, dw.Tahun, dw.Bulan
-- Hapus baris ORDER BY yang lama: ORDER BY dw.Tahun, dw.Bulan, TotalIncidentsHandled DESC;
GO
```

100 % No issues found

Messages

```
Commands completed successfully.

Completion time: 2025-12-01T21:56:36.5813119+07:00
```

Security Implementation

Create User Roles

```sql
USE [DM_KeamananSiber_DW];
GO

-- CATATAN: Perintah CREATE ROLE dilewati karena Roles sudah ada di database.

-- ============================================================
-- Grant Permissions for ETL Operator (Izin untuk Proses ETL)
-- ============================================================
-- Memberikan hak EXECUTE untuk menjalankan Master Stored Procedure
GRANT EXECUTE ON dbo.usp_Master_ETL_KeamananSiber TO db_etl_operator;
-- Memberikan hak penuh pada Staging Area (untuk memuat dan membersihkan data)
GRANT SELECT, INSERT, UPDATE, DELETE ON SCHEMA::stg TO db_etl_operator;
-- Memberikan hak INSERT pada skema dbo (untuk memuat data ke Fact/Dimensi)
GRANT INSERT ON SCHEMA::dbo TO db_etl_operator;
GO

-- ============================================================
-- Grant Permissions for Executive (Izin untuk Pengambilan Keputusan Tingkat Tinggi)
-- ============================================================
-- Memberikan hak SELECT pada skema dbo (semua data final di DW)
GRANT SELECT ON SCHEMA::dbo TO db_executive;
-- Memberikan hak untuk menjalankan Master SP (untuk pengawasan atau eksekusi darurat)
GRANT EXECUTE ON dbo.usp_Master_ETL_KeamananSiber TO db_executive;
GO

-- ============================================================
```

100 % No issues found

Messages

```
Commands completed successfully.

Completion time: 2025-12-01T22:04:49.0721597+07:00
```

users and assign roles

```sql
CREATE LOGIN viewer_user WITH PASSWORD = 'StrongP@ssw0rd!', CHECK_POLICY = ON;
CREATE LOGIN etl_service WITH PASSWORD = 'StrongP@ssw0rd!', CHECK_POLICY = ON;
GO

-- ============================================================
-- 2. Create Database Users (Tingkat Database)
-- ============================================================
USE DM_KeamananSiber_DW;
GO

-- Membuat user di database DW, memetakan ke Login yang baru dibuat
CREATE USER executive_user FOR LOGIN executive_user;
CREATE USER analyst_user FOR LOGIN analyst_user;
CREATE USER viewer_user FOR LOGIN viewer_user;
CREATE USER etl_service FOR LOGIN etl_service;
GO

-- ============================================================
-- 3. Assign Users to Roles (Menetapkan User ke Role)
-- ============================================================
-- Menetapkan setiap user ke Database Role yang sesuai
ALTER ROLE db_executive ADD MEMBER executive_user;
ALTER ROLE db_analyst ADD MEMBER analyst_user;
ALTER ROLE db_viewer ADD MEMBER viewer_user;
ALTER ROLE db_etl_operator ADD MEMBER etl_service;
GO
```

100 % No issues found

Messages

```
Commands completed successfully.

Completion time: 2025-12-01T22:06:33.5524772+07:00
```

Security Implementation

Data Masking

```sql
USE DM_KeamananSiber_DW;
GO

-- ============================================================
-- 1. Terapkan Masking pada Kolom Sensitif (Petugas di Dim_TimRespon)
-- ============================================================

-- Target PII: Nama Petugas (Staf) di tim respon
-- Menggunakan fungsi partial() untuk menampilkan hanya huruf pertama nama
-- Contoh: 'Fadlan Nugraha' akan ditampilkan sebagai 'FXXXXX'

ALTER TABLE dbo.Dim_TimRespon
ALTER COLUMN Petugas ADD MASKED WITH (FUNCTION = 'partial(1,"XXXXX",0)');
GO

-- ============================================================
-- 2. Grant UNMASK Permission (Memberikan Izin Melihat Data Asli)
-- ============================================================

-- Memberikan izin UNMASK kepada Executive dan Analyst agar mereka dapat melihat nama asli petugas
GRANT UNMASK TO db_executive;
GRANT UNMASK TO db_analyst;
GO
```

```
Commands completed successfully.

Completion time: 2025-12-01T22:16:01.5690821+07:00
```

Audit Trail

```sql
USE master;
GO

-- 1. DROP EXISTING SERVER AUDIT (Mencegah konflik jika sudah ada)
IF EXISTS (SELECT * FROM sys.server_audits WHERE name = 'DataWarehouse_Audit')
BEGIN
    ALTER SERVER AUDIT DataWarehouse_Audit WITH (STATE = OFF);
    DROP SERVER AUDIT DataWarehouse_Audit;
END
GO

-- 2. CREATE SERVER AUDIT
-- PERHATIAN: Ganti N'C:\SQLAuditLogs\' dengan jalur folder yang valid di Server Anda.
-- Folder ini harus sudah ada dan dapat ditulisi oleh SQL Server Service Account.
CREATE SERVER AUDIT DataWarehouse_Audit
TO FILE
( FILEPATH = N'C:\SQLAuditLogs\',
```

```
Commands completed successfully.

Completion time: 2025-12-01T23:37:50.4012356+07:00
```

```sql
USE DM_KeamananSiber_DW;
GO

-- 1. DDL FIX: Tambah Kolom PII yang hilang (Agar DDL tidak gagal di langkah lain)
IF NOT EXISTS (SELECT * FROM sys.columns WHERE Name = N'Email' AND Object_ID = Object_ID(N'dbo.Dim_TimRespon'))
BEGIN
    ALTER TABLE dbo.Dim_TimRespon
    ADD Email VARCHAR(100) NULL,
        Phone VARCHAR(20) NULL;
END
GO

-- 2. Create Audit Log Table (Drop jika sudah ada untuk menghindari Error 2714)
IF OBJECT_ID('dbo.AuditLog', 'U') IS NOT NULL
    DROP TABLE dbo.AuditLog;
GO
CREATE TABLE dbo.AuditLog (
```

```
Commands completed successfully.

Completion time: 2025-12-01T23:38:17.5469475+07:00
```