# ANKARA ÜNİVERSİTESİ

**MÜHENDİSLİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

****

**PROJE RAPORU**

**(BLM4522) AĞ TABANLI PARALEL DAĞITIM SİSTEMLERİ**

**Perizat SAGYNBEKOVA**

**21290895**

**GitHub (https://github.com/Impasbaa/MSSQL.git)**

**25.04.2025**

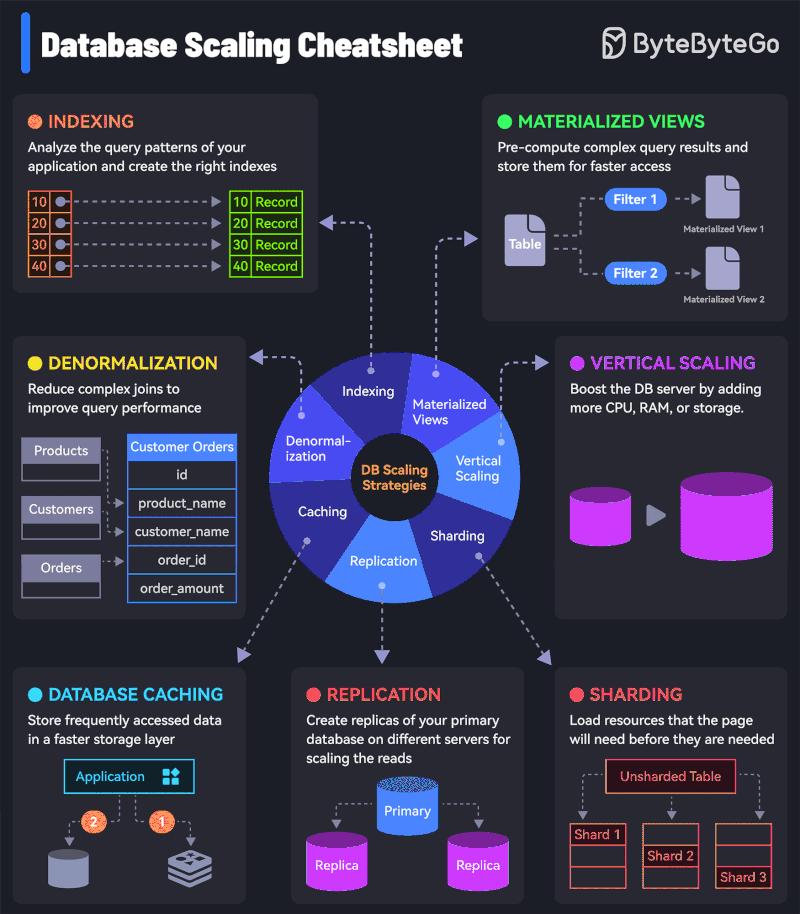
**PROJE 1. VERİTABANI PERFORMANS OPTİMİZASYONU VE İZLEME**

* **PROJE AMACI:** Bu proje kapsamında SQL Server Management Studio (SSMS) üzerinde örnek bir büyük veritabanı (AdventureWorks) kullanılarak performans izleme, indeks yönetimi, disk alanı kullanımı, veri yoğunluğu yönetimi ve optimizasyon konularında işlemler gerçekleştirmektir. Sistemin daha verimli çalışmasını sağlamak ve ileri düzey veritabanı yönetimi becerilerini geliştirmektir. Ayrıca, sistemdeki performansı izlemek için SQL Server araçları kullanılır.
* **KULLANILAN VERİTABANI:** AdventureWorks2022

**Yapılanlar**

1. AdventureWorks2022 dosyası indirildi.
2. SSMS üzerinden Restore Database işlemiyle sistemde kuruldu.

* **VERİTABANI PERFORMANS OPTİMİZASYONU İÇİN ÖLÇEKLENDİRME NEDEN VE NE ZAMAN GEREKLİDİR?**



* **PROJE ADIMLARI:**

**a)** **Performans İzleme**

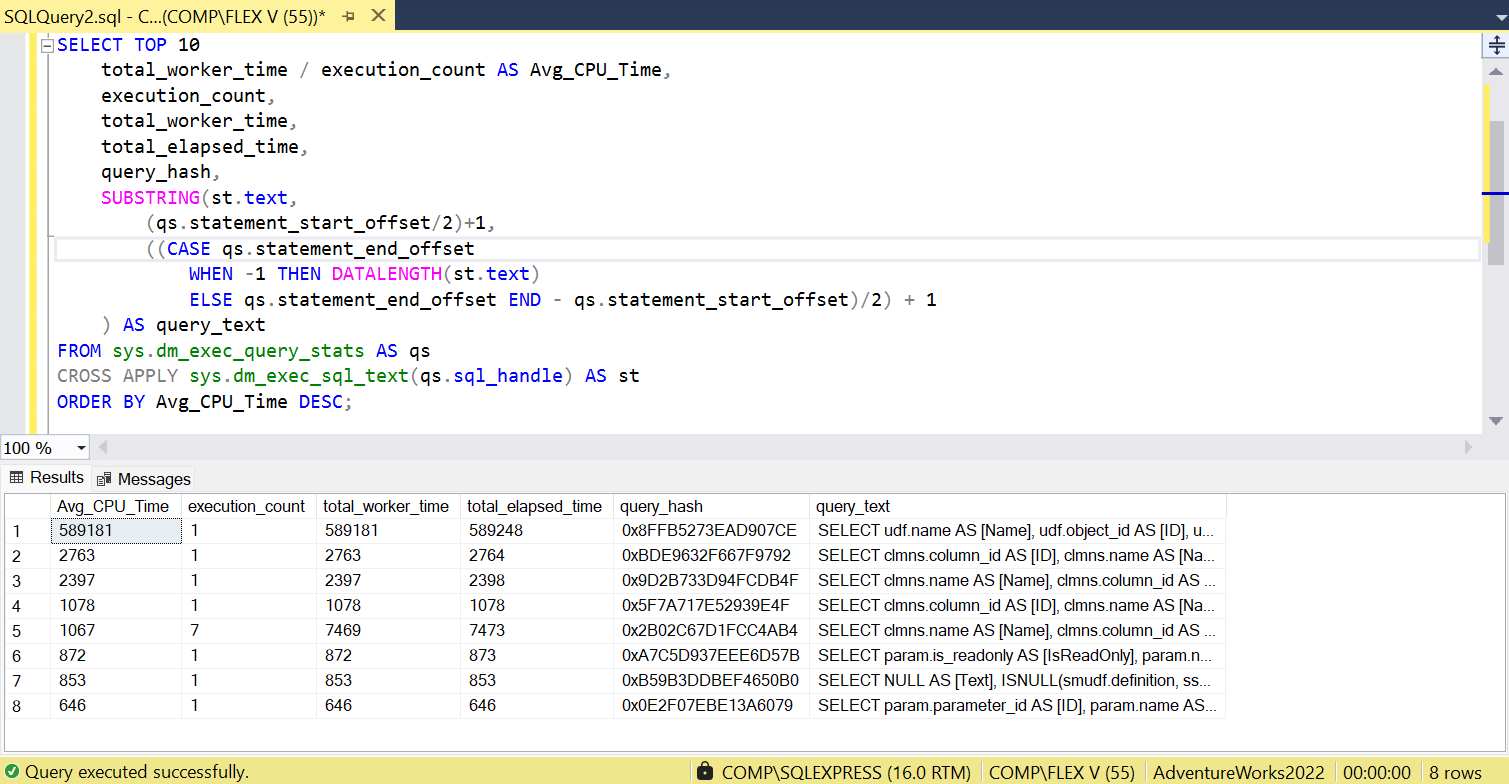
**Araç:** SQL Server Profiler

**Amaç:** Ağır veya yavaş sorguları tespit etmek

**Adımlar:**

* SQL Server Profiler'ı başlatmak
* Yeni bir "Trace" oluşturmak
* Şablonunu seçmek (TSQL\_Duration)
* Hedef veritabanını seçmek (AdventureWorks)
* 10-15 dakika sistemdeki işlemleri izlemek
* En uzun süren sorguları tespit etmek ve kaydetmek

**b)** **Dynamic Management Views (DMV) ile Performans Analizi**

****

*Açıklama: En çok CPU harcayan sorguları listeler. Sorgu metinleri ve çalışma istatistiklerini gösterir.*

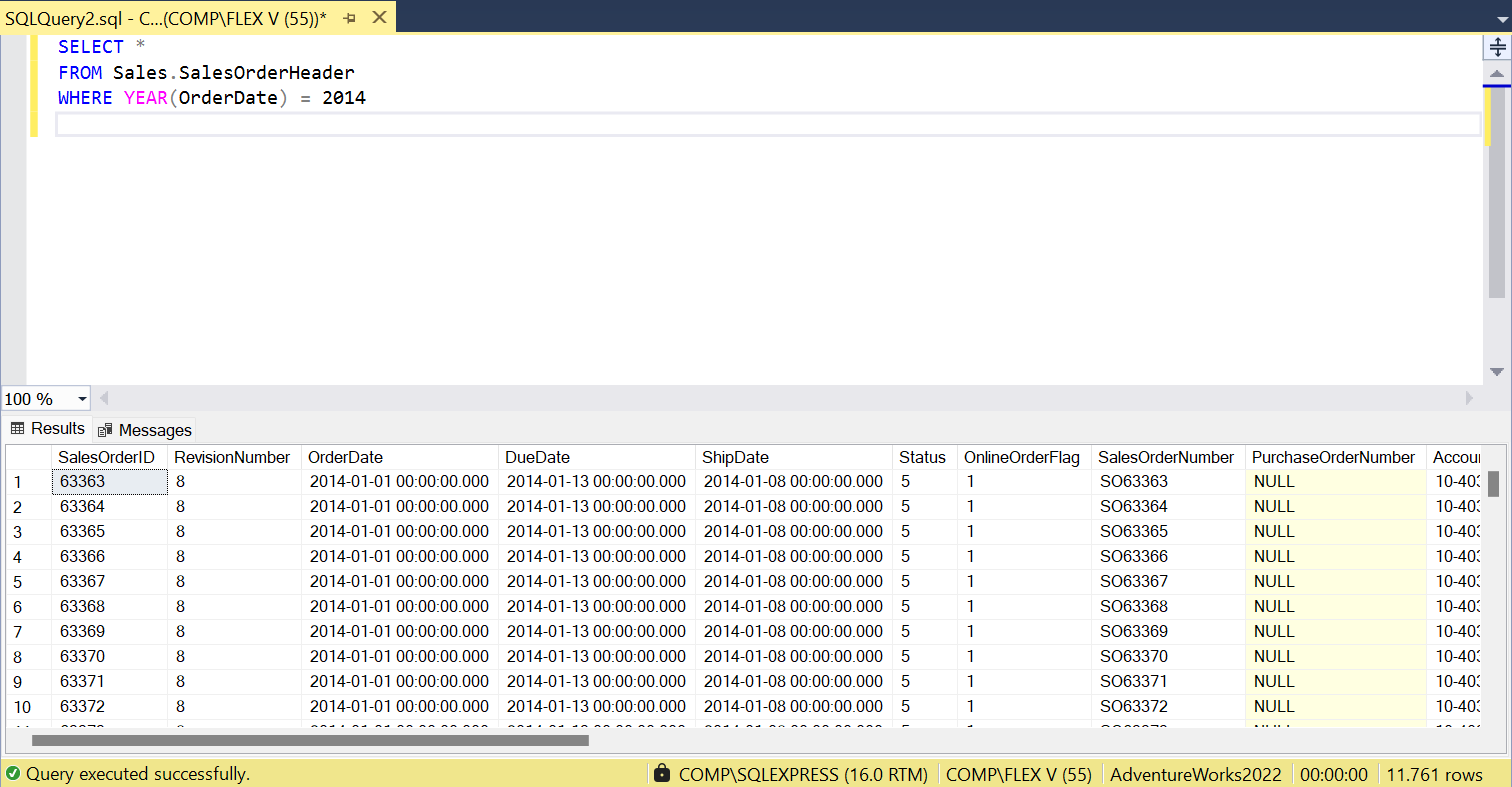
**c) İndeks Analizi ve Yönetimi**

**Amaç:** Gereksiz veya eksik indeksleri bulmak

**Adımlar:**

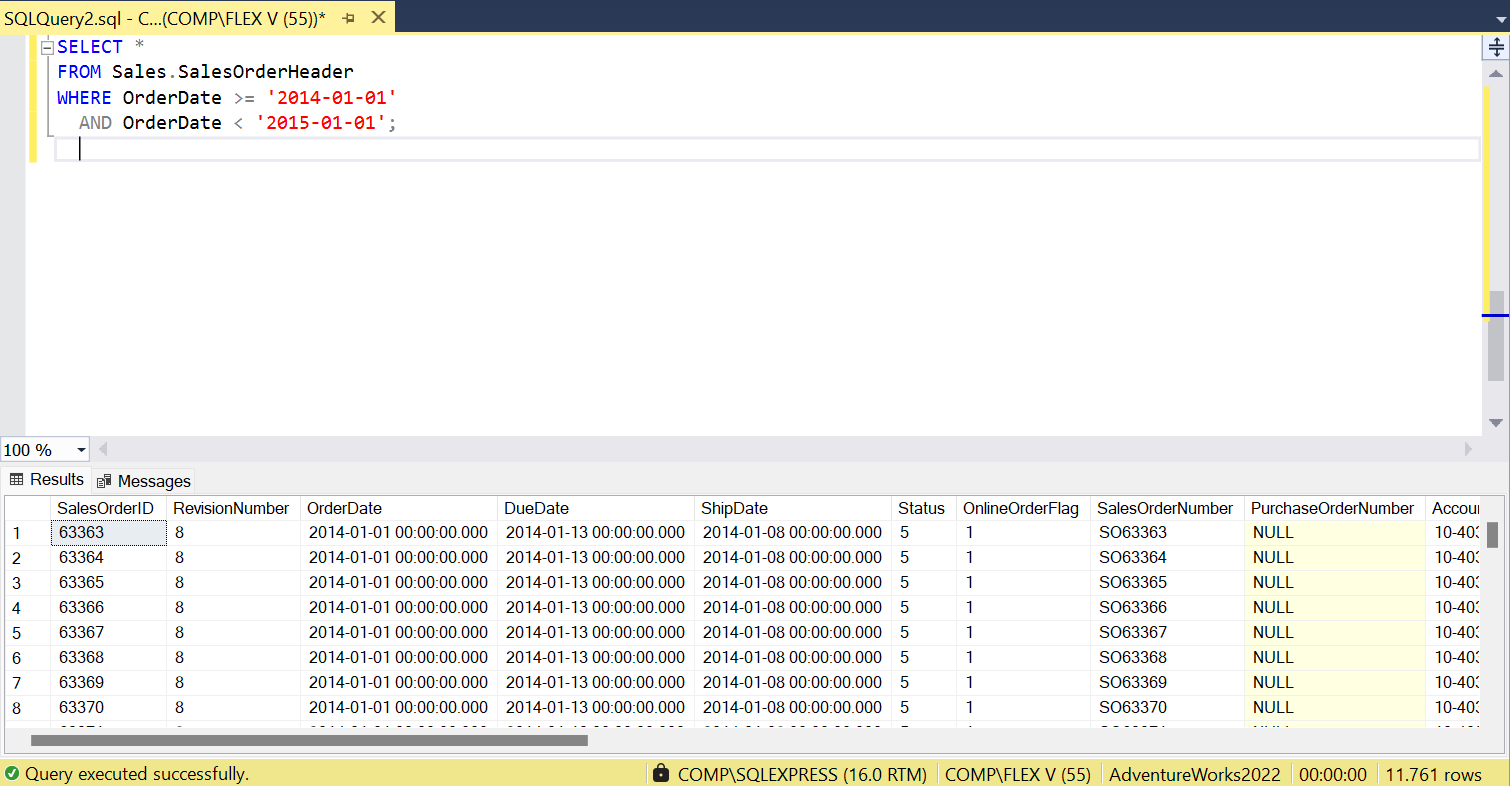
* İndeks kullanımını incelemek

**Optimize Edilmemiş Sorgu:**

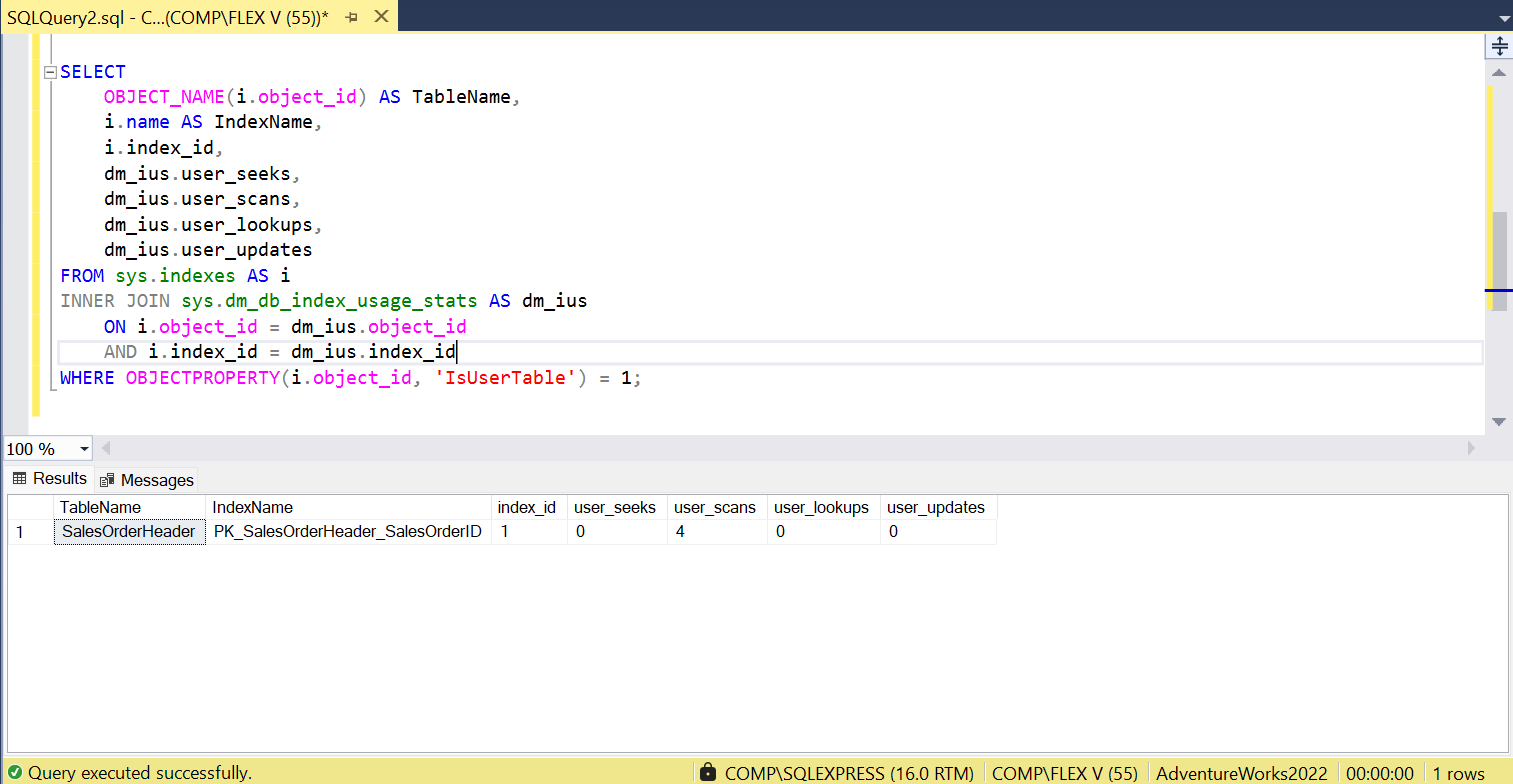
****

*Açıklama: YEAR() fonksiyonu kullanmak, indekslerin kullanılmasını engeller. Bu sorgu optimize edilmelidir.*

**Optimize Edilmiş Sorgu:**



*Açıklama: YEAR() fonksiyonu yerine tarih aralığı kullanmak, indeks kullanımını sağlar ve sorgu hızını artırır.*



*Açıklama: İndekslerin ne kadar kullanıldığını gösterir. Kullanılmayan indeksler tespit edilebilir.*

* Gereksiz veya hiç kullanılmayan indeksleri kaldırmak

DROP INDEX *IndexName* ON *TableName*;

**d) Sorgu Optimizasyonu**

**Amaç:** Yavaş çalışan sorguları daha verimli hale getirmek.

**Yapılacaklar:**

* SQL Profiler veya DMV ile yavaş sorguları belirlemek
* EXPLAIN ya da Execution Plan'ı analiz etmek
* JOIN, WHERE, GROUP BY ifadelerini sadeleştirmek
* INDEX veya TEMP TABLE kullanmak

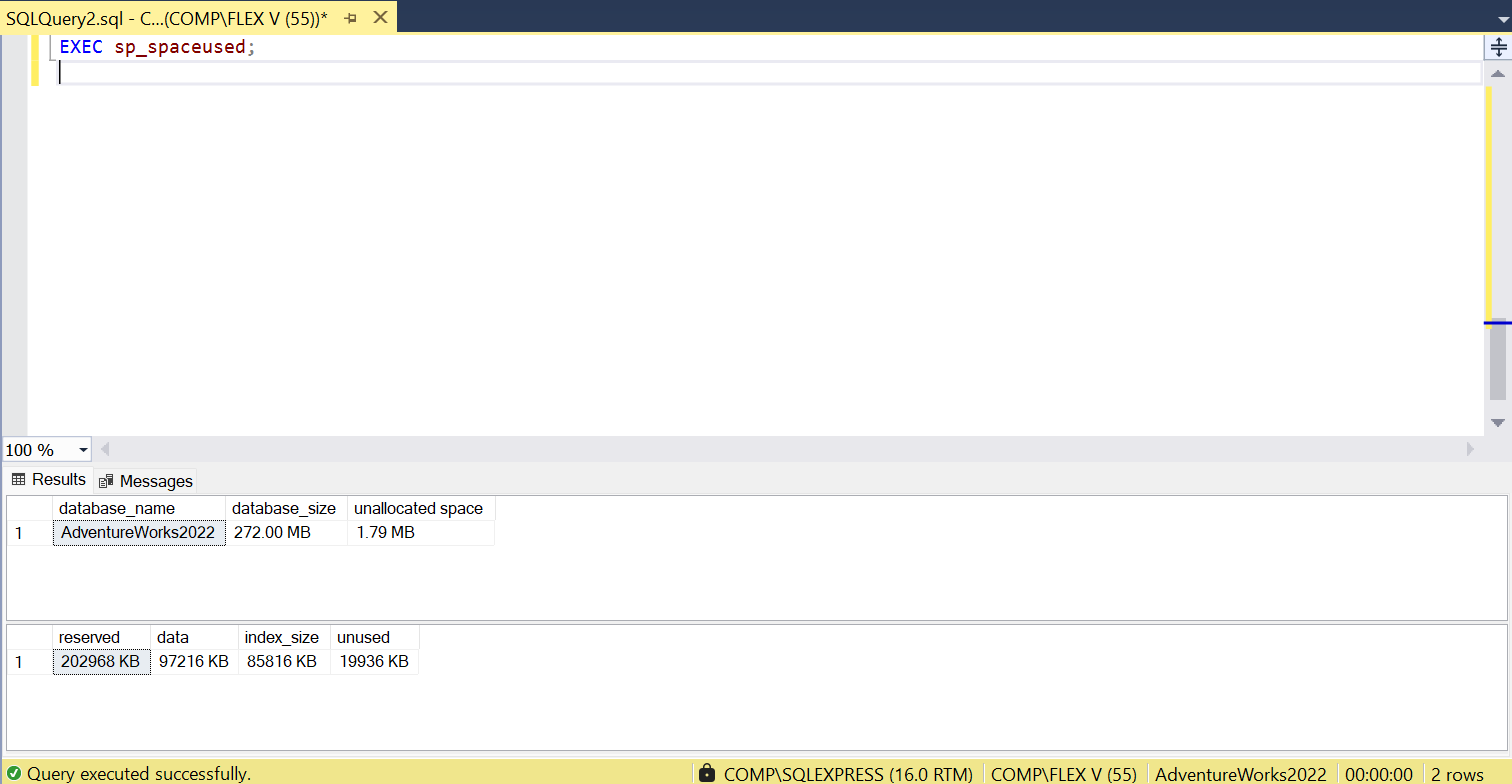
CREATE INDEX IX\_Customer\_LastName

ON Customer (LastName);

*Açıklama: Sorgu performansını artırmak için yeni bir indeks oluşturur.*

**e) Disk Alanı ve Veri Yoğunluğu Yönetimi**

* Kontrol:



*Açıklama: Veritabanının toplam boyutunu ve boş alanını gösterir.*

* Veri aralığını veya parçalanmış indeksleri yeniden düzenleme (fragmentation fix):

ALTER INDEX ALL ON *TableName* REBUILD;

*Açıklama: Tüm indeksleri yeniden oluşturur. Diskteki veri düzenini iyileştirir, performansı artırır.*

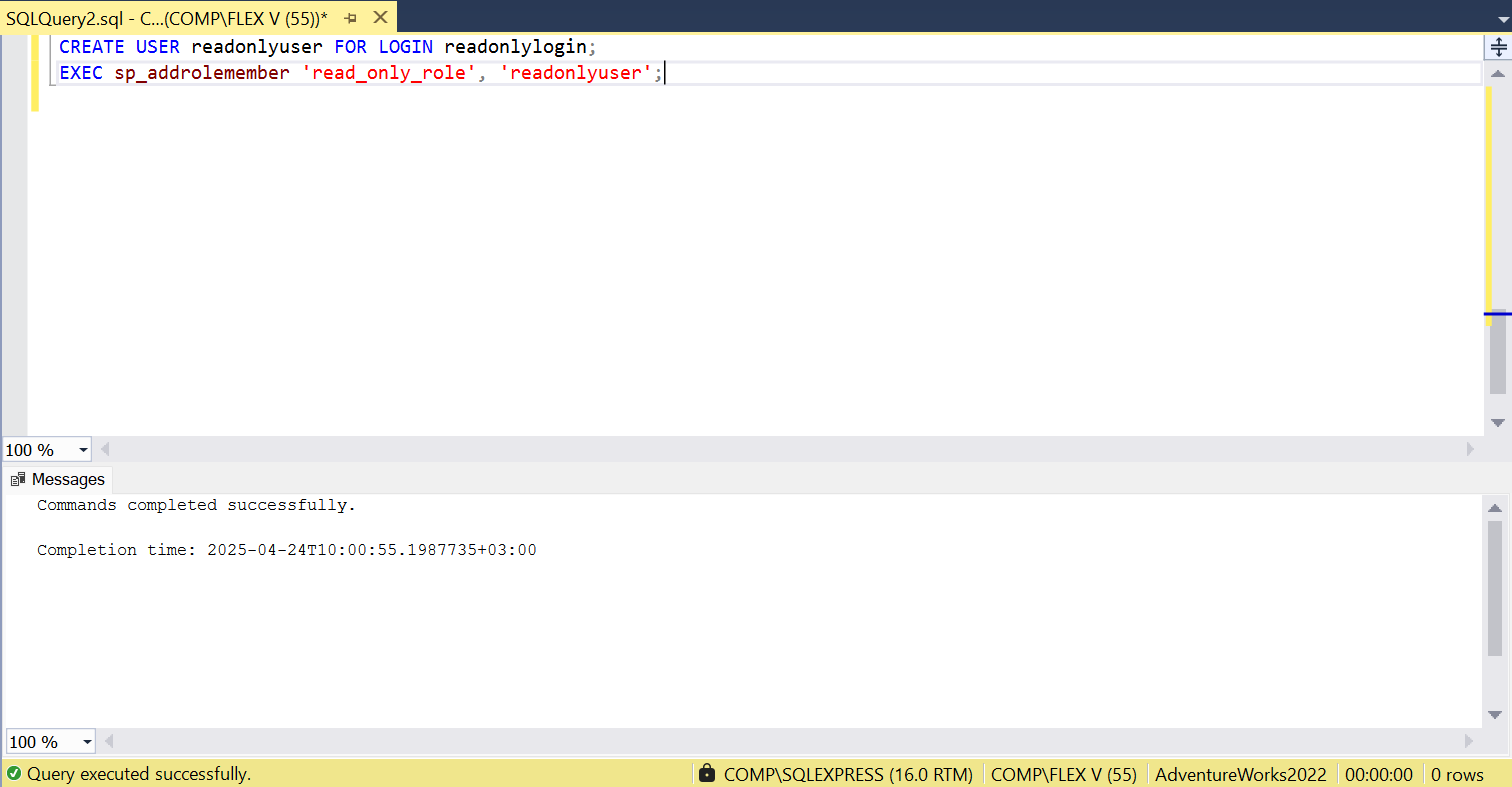
**f) Erişim ve Rol Yönetimi**

**Amaç:** Veritabanı yöneticisi rollerini ayırmak

CREATE ROLE read\_only\_role;

GRANT SELECT ON dbo.TableName TO read\_only\_role;

*Açıklama: Okuma yetkisi olan yeni rol oluşturur.*



*Açıklama: Sadece okuma yetkisine sahip kullanıcı tanımlanır.*

* **KAYNAKLAR**

1. <https://www.youtube.com/watch?v=_1IKwnbscQU&list=TLPQMjQwNDIwMjUq_dK_15bbIg&index=8>
2. Kullanılan Veritabanı: <https://learn.microsoft.com/en-us/sql/samples/adventureworks-install-configure?view=sql-server-ver16&tabs=ssms>
3. Anlatım Videosu: <https://youtu.be/Iw1PrMjMHzE>
4. GitHub Linki: <https://github.com/Impasbaa/MSSQL/blob/main/SQLQuery2.sql>

**PROJE 2. VERİTABANI YÜKSELTME VE SÜRÜM YÖNETİMİ**

* **PROJE AMACI:** Bir veritabanının eski sürümden daha yeni bir sürüme yükseltilmesini gerçekleştirmek, sürüm kontrol mekanizmaları ile değişikliklerin izlenmesini sağlamak ve yükseltme sürecine ait test ve geri dönüş planlarını tanımlamaktır.
* **KULLANILAN VERİTABANI VE ORTAM**

**Veritabanı:** AdventureWorks2022

**Platform:** Microsoft SQL Server 2022

**Araç:** SQL Server Management Studio (SSMS)

* **VERİTABANI YÜKSELTME PLANI**

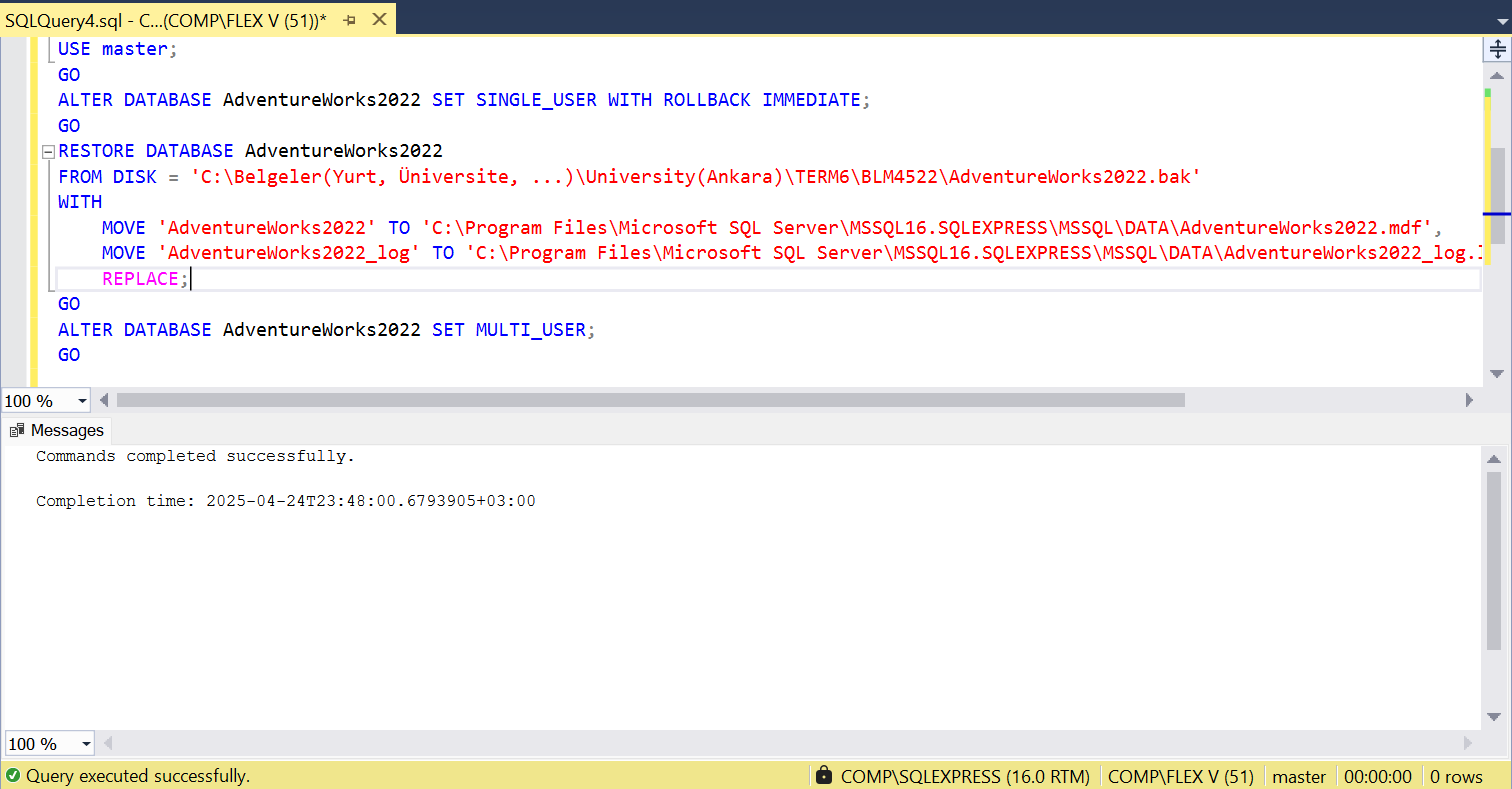
**metin, ekran görüntüsü, yazılım, ekran, görüntüleme içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

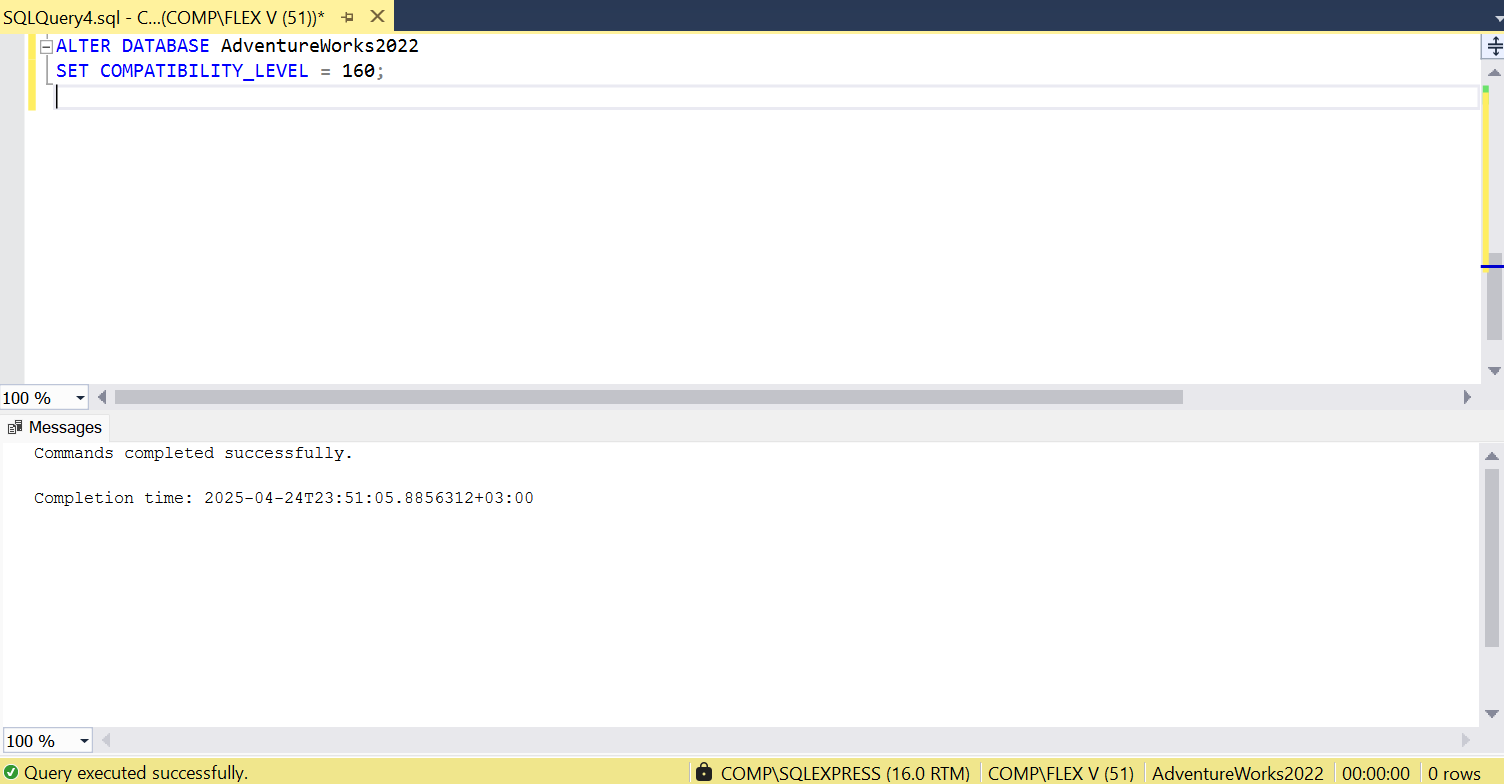
*Açıklama: Mevcut veritabanının incelenmesi.*

**

*Açıklama: Yedek alma.*

**

*Açıklama: Yeni sürümde restore etme.*

**

*Açıklama: Uyumlu sürüm ayarlama.*

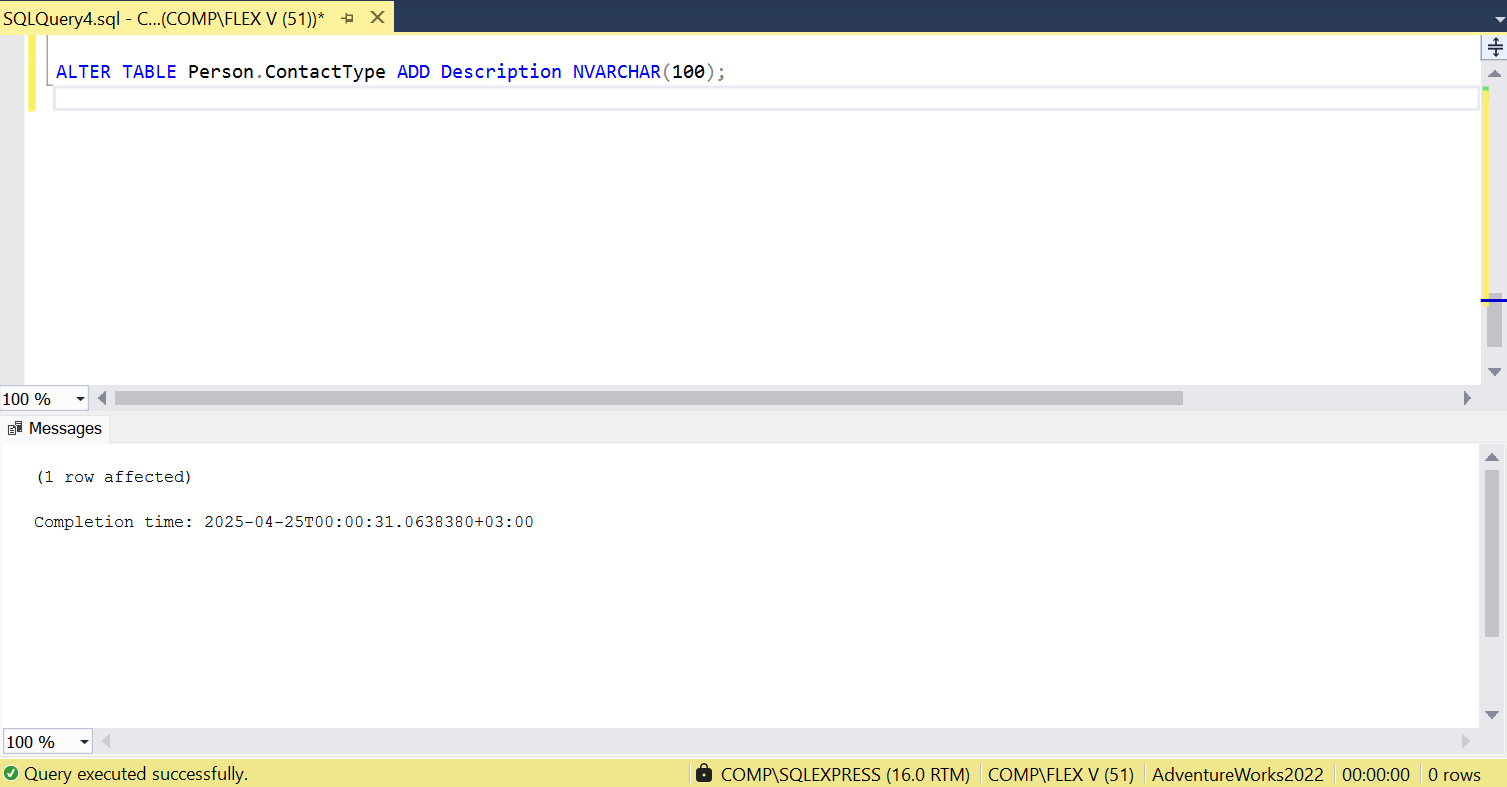
* **SÜRÜM YÖNETİMİ**

**metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Yapay zeka tarafından oluşturulan içerik yanlış olabilir.metin, ekran görüntüsü, yazılım, ekran, görüntüleme içeren bir resim

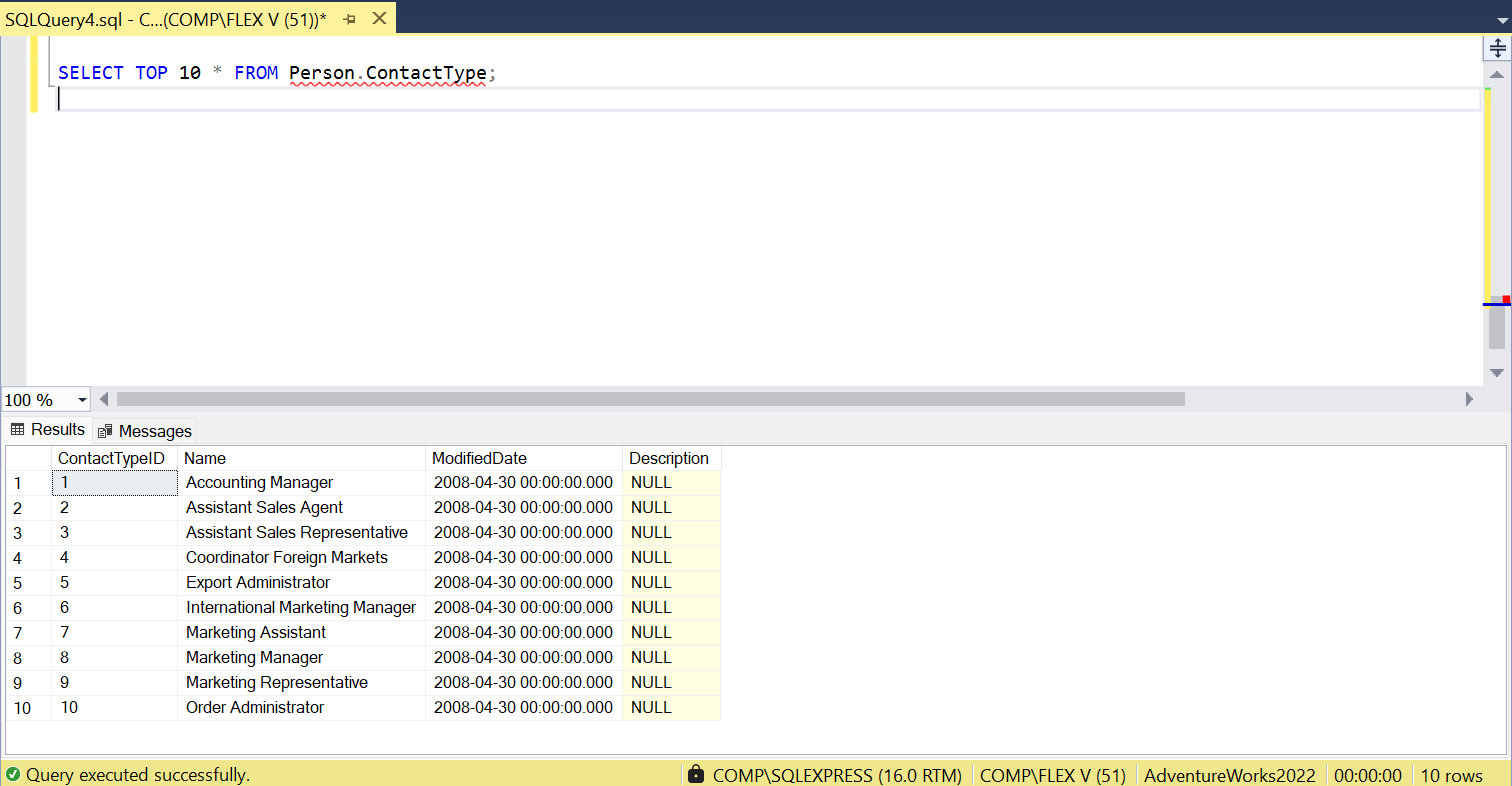
Yapay zeka tarafından oluşturulan içerik yanlış olabilir.**

*Açıklama: DDL Trigger ile şema değişikliği takibi.*

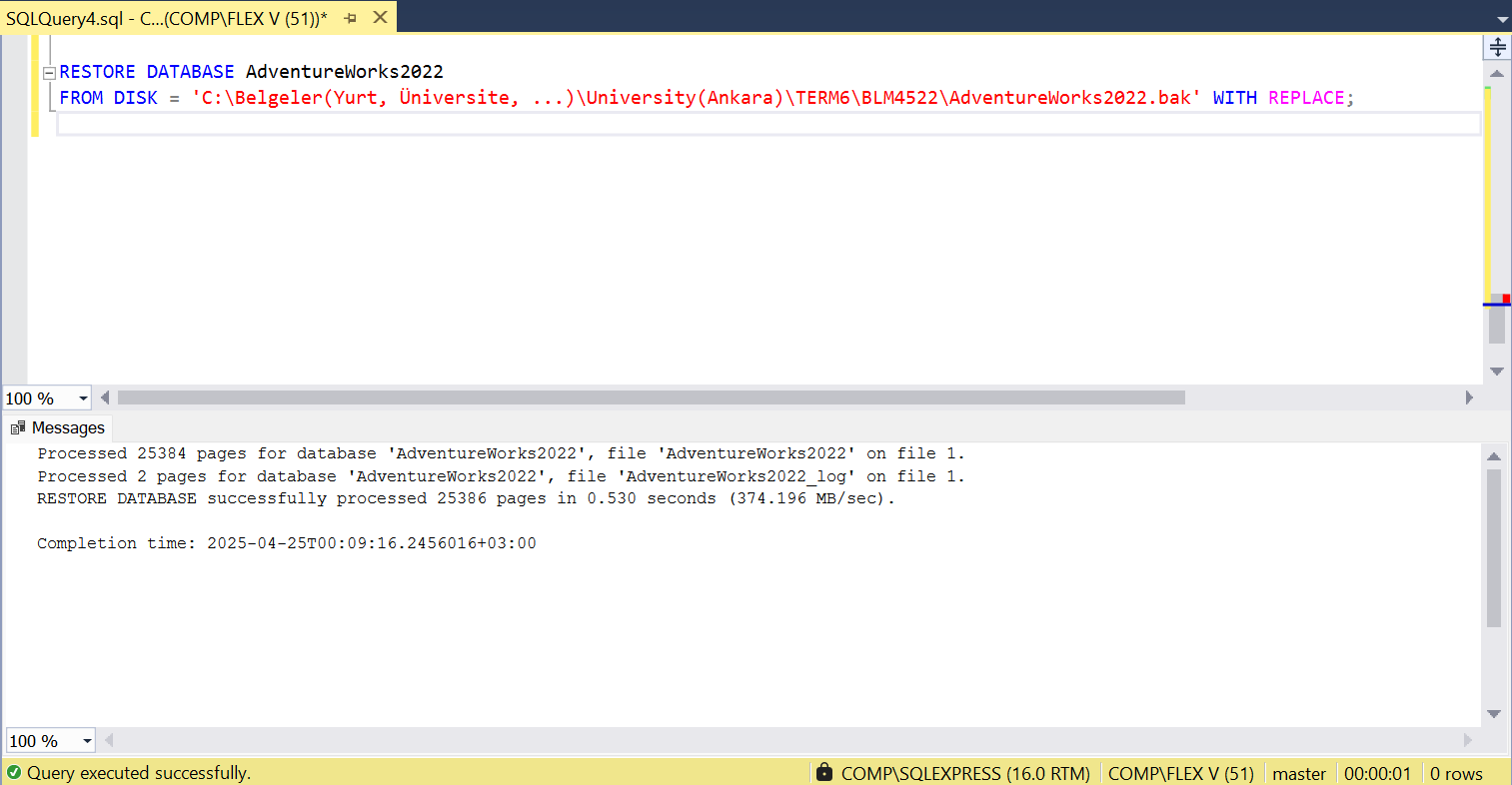
**

*Açıklama: Test amaçlı şema değişikliği.*

* **TEST VE GERİ DÖNÜŞ PLANI**
* Geri dönüş için alınan tam yedek kullanılabilir
* Yükseltme sonrası test sorguları:



* İşlevsel test, CRUD işlemlerinin denenmesi
* Sorun durumunda:



* **SONUÇ**

Veritabanı başarıyla yeni sürüme geçirildi ve şema değişiklikleri izlenebilir hale getirildi. Ayrıca da geri dönüş planı test edildi ve çalıştığı doğrulandı.

* **KAYNAKLAR**

1. Kullanılan Veritabanı: <https://learn.microsoft.com/en-us/sql/samples/adventureworks-install-configure?view=sql-server-ver16&tabs=ssms>
2. Anlatım Videosu: https://youtu.be/BBNB-ZdF2vU
3. GitHub Linki: <https://github.com/Impasbaa/MSSQL/blob/main/SQLQuery4.sql>

**PROJE 3. VERİTABANI GÜVENLİĞİ VE ERİŞİM KONTROLÜ**

* **PROJE AMACI:** Microsoft SQL Server kullanılarak veritabanı güvenliği sağlanması hedeflenmektedir. Kullanıcı erişim yönetimi, veri şifreleme, SQL Injection testleri ve audit log (kayıt) sistemlerinin uygulanması gerçekleştirilecektir.
* **KULLANILAN VERİTABANI VE ORTAM**

**Veritabanı:** AdventureWorks2022

**Platform:** Microsoft SQL Server 2022

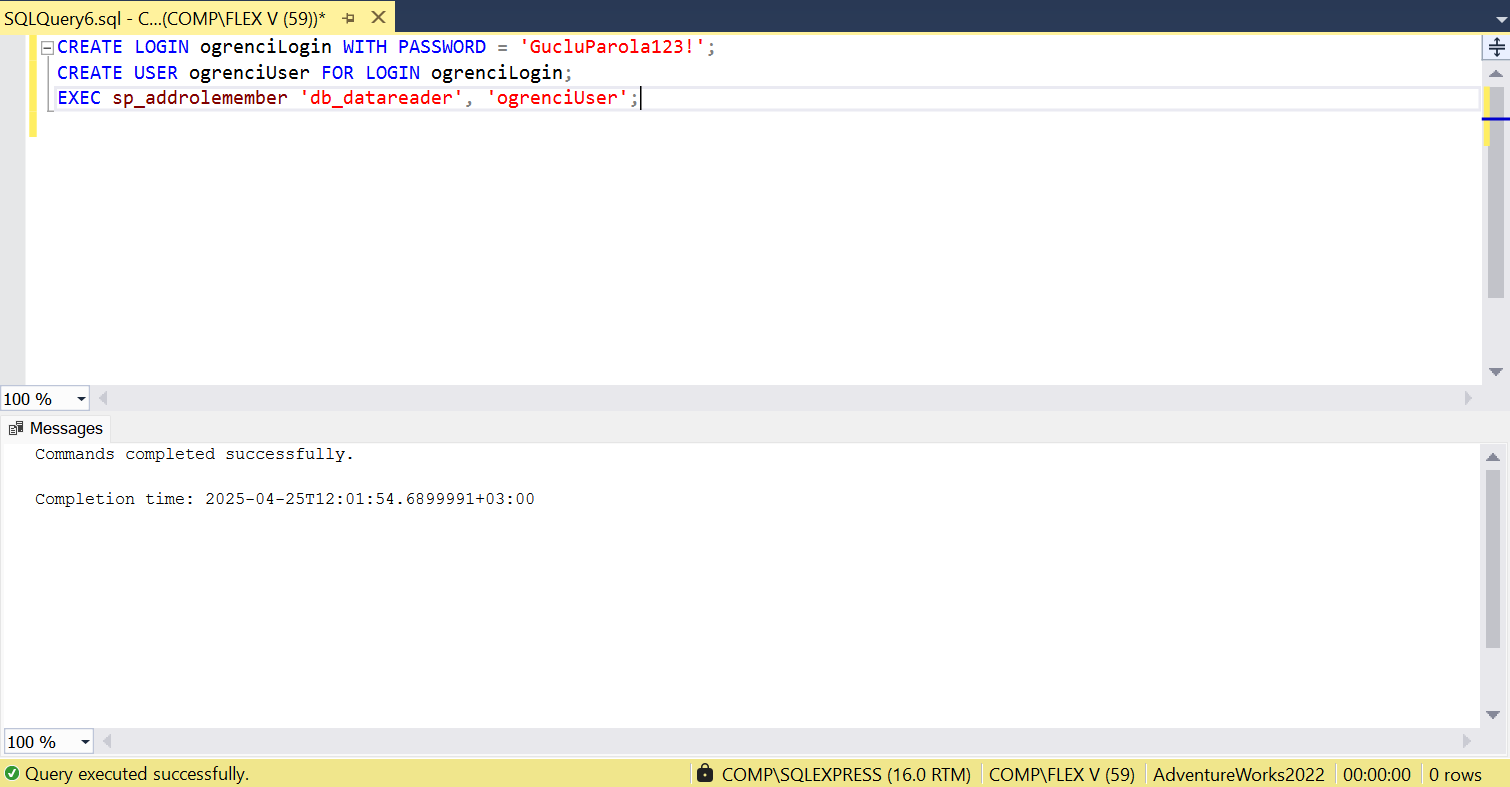
**Araç:** SQL Server Management Studio (SSMS)

Windows Authentication / SQL Server Authentication

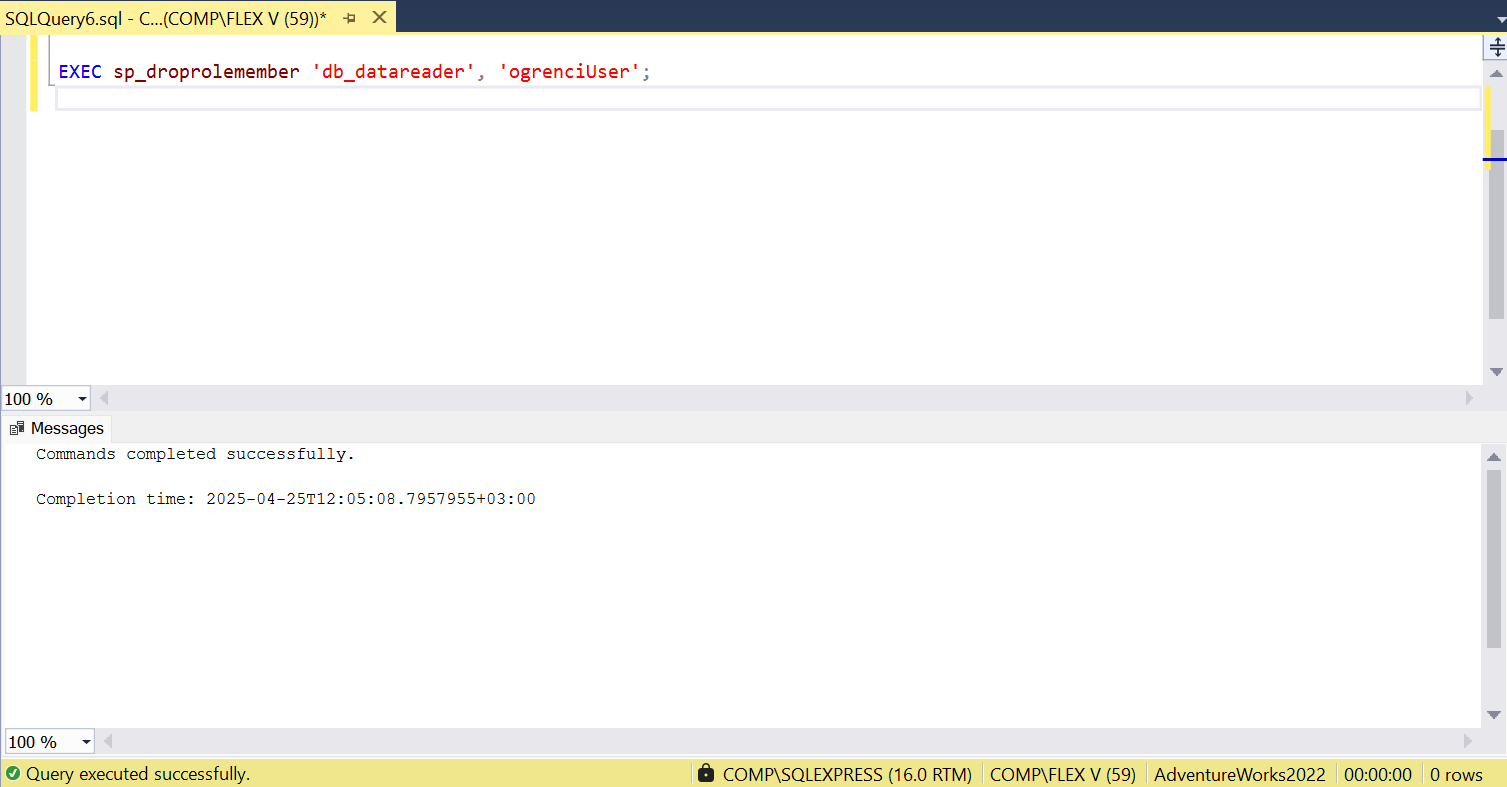
* **PROJE ADIMLARI:**

1. **Erişim Yönetimi**

*Kullanıcı Oluşturma (SQL Authentication):*

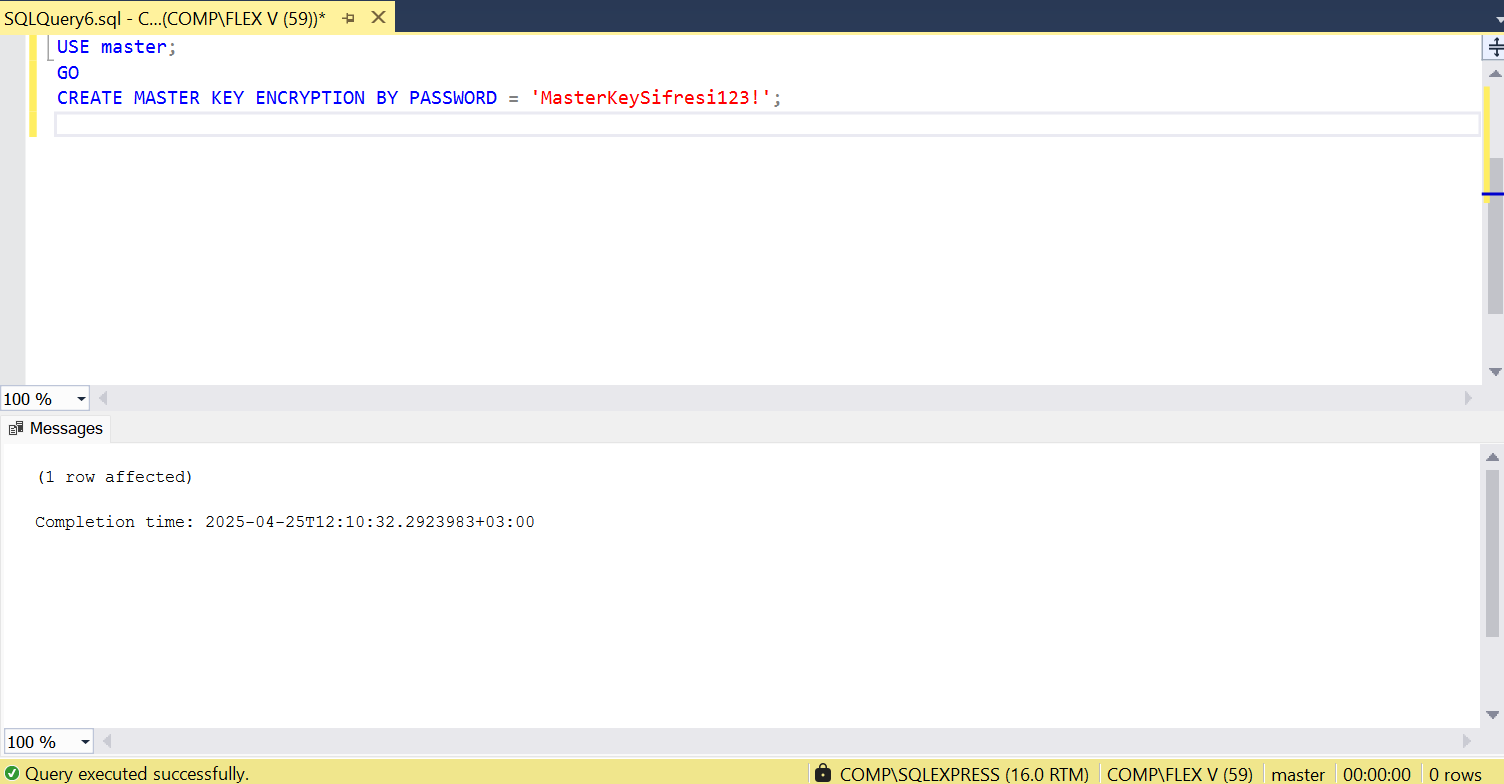


*Yetkiyi Geri Alma:*

****

1. **Veri Şifreleme (TDE - Transparent Data Encryption)**

*Ana Anahtar Oluşturma:*



*Sertifika Oluşturma:*

CREATE CERTIFICATE TDESertifika

WITH SUBJECT = 'Veritabanı Şifreleme Sertifikası';

*Veritabanı Şifreleme Anahtarı (DEK) oluşturma:*

USE AdventureWorks2022;

GO

CREATE DATABASE ENCRYPTION KEY

WITH ALGORITHM = AES\_256

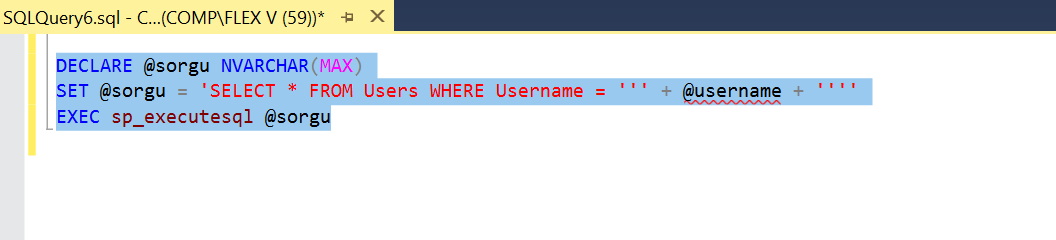
ENCRYPTION BY SERVER CERTIFICATE TDESertifika;

*Şifrelemeyi Etkinleştirme:*

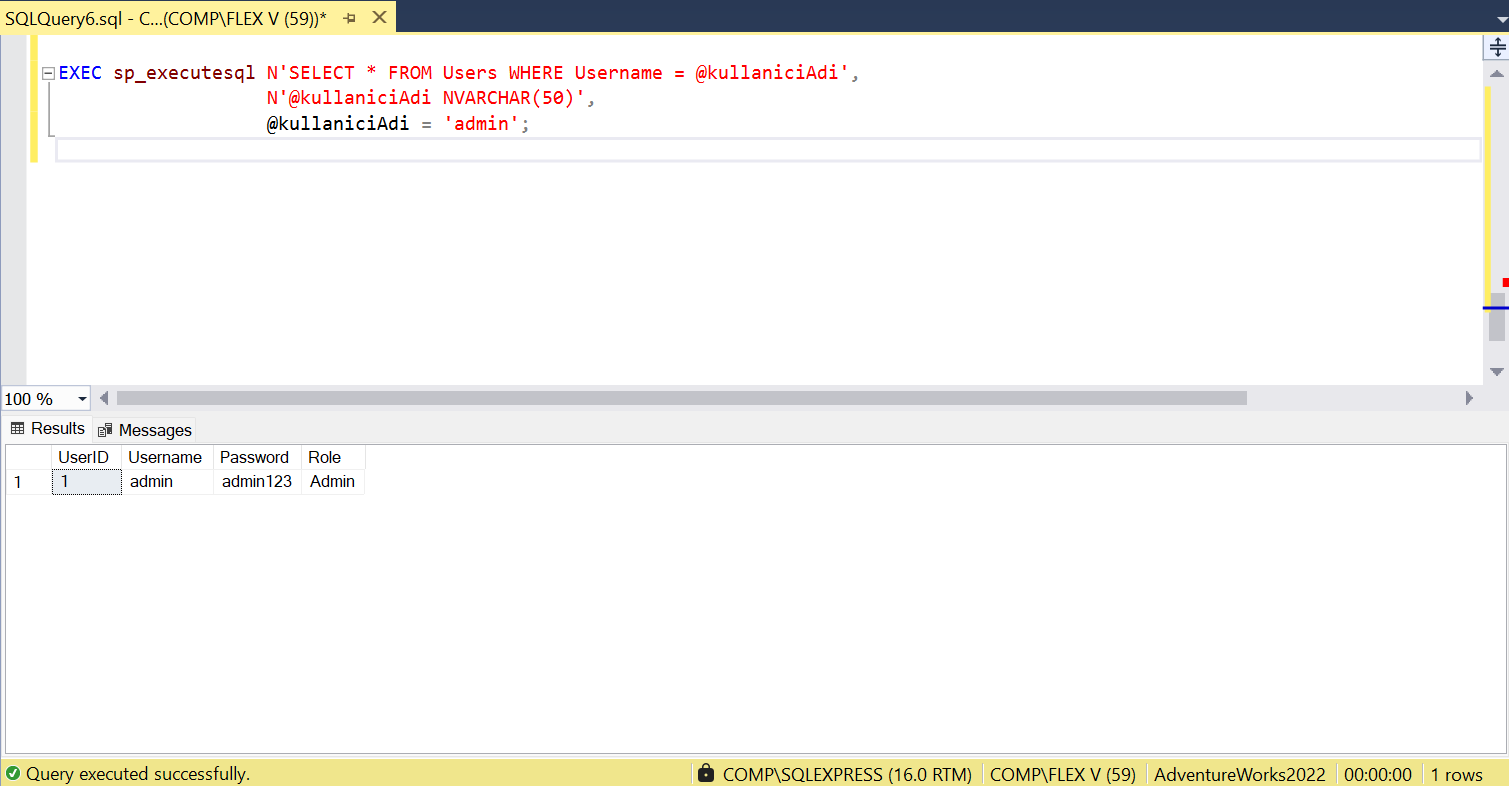
ALTER DATABASE AdventureWorks2022 SET ENCRYPTION ON;

1. **SQL Injection Testi ve Önleme**

*Kötü Yazılmış Dinamik SQL (Savunmasız):*



*Doğru Kullanım (Parametreli Sorgu ile):*

**

1. **Audit Logları (SQL Server Audit)**

*Server Audit Oluşturma:*

USE master;

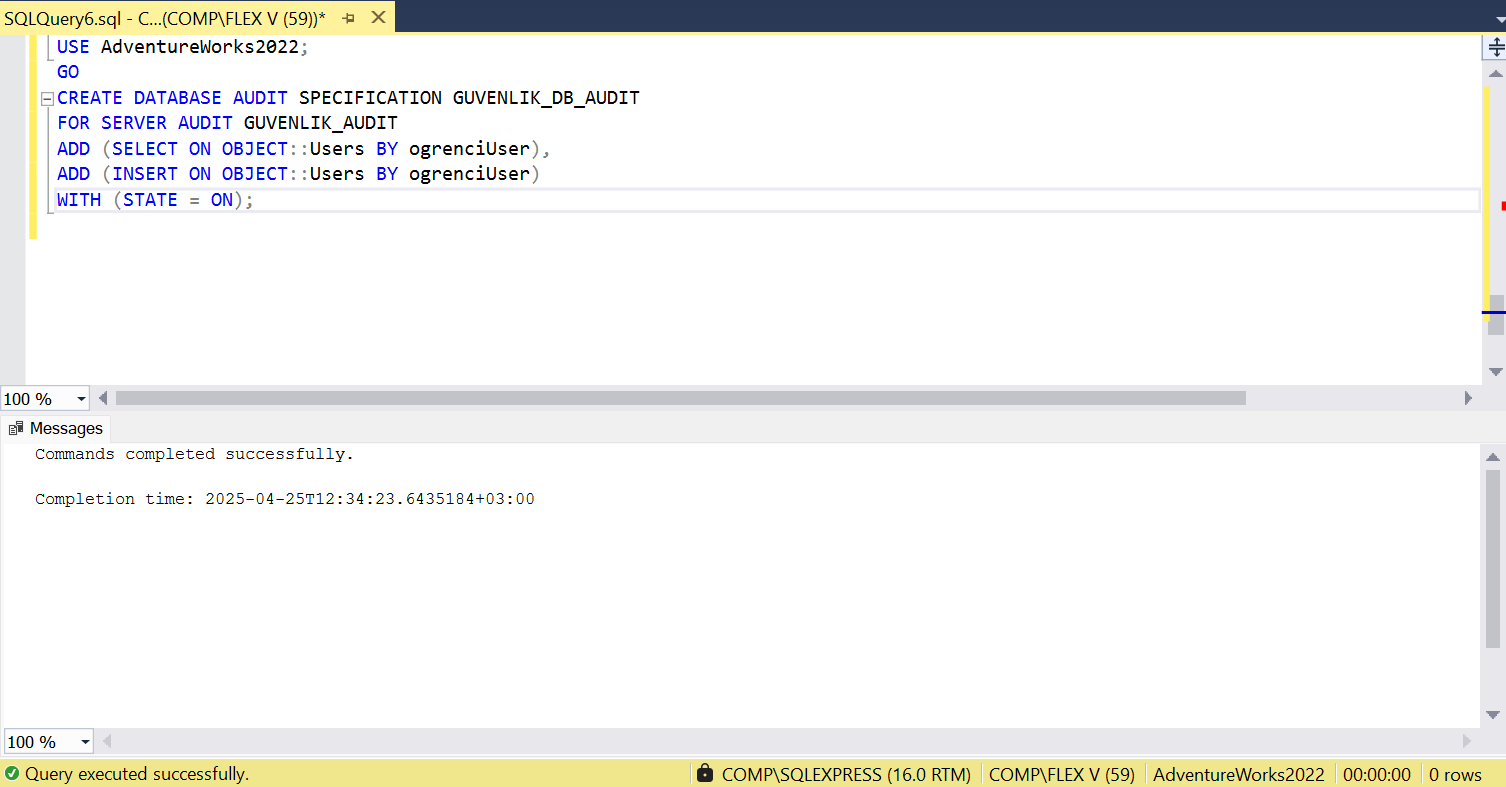
GO

CREATE SERVER AUDIT GUVENLIK\_AUDIT

TO FILE (FILEPATH = 'C:\AuditLogs\');

ALTER SERVER AUDIT GUVENLIK\_AUDIT WITH (STATE = ON);

*Database-Level Audit Oluşturma:*



* **SONUÇ**

Bu projede SQL Server üzerinde temel güvenlik önlemleri başarıyla uygulanmıştır. Kullanıcı erişimi sınırlandırılmış ve hassas veriler şifrelenmiş, injection’a karşı önlem alınmış ve kullanıcı aktiviteleri kayıt altına alınmıştır*.*

* **KAYNAKLAR**

1. Kullanılan Veritabanı: <https://learn.microsoft.com/en-us/sql/samples/adventureworks-install-configure?view=sql-server-ver16&tabs=ssms>
2. GitHub Linki: <https://github.com/Impasbaa/MSSQL/blob/main/SQLQuery6.sql>