Ondokuz Mayıs Üniversitesi

Bilgisayar Mühendisliği

Aleyna KAHRAMAN

20060355

Deney Föyü-1

Gülcan YILDIZ

ÖZET

FÖY raporu kapsamında, SQL Server üzerinde bir dizi önemli işlem gerçekleştirildi. İlk adımda, başarılı bir SQL Server kurulumu tamamlandı. Ardından, veri tabanı oluşturma sürecine geçildi ve Föyde anlatılan kodlar kullanılarak bir veri tabanı ve bu veri tabanına beş tablo eklendi. Her bir tablo içerisinde en az beş farklı veri tipi kullanıldı. Tablolar arasında foreign-key ve primary-key bağlantıları yapılarak veri bütünlüğü sağlandı.

Kullanıcı yönetimi ve yetkilendirme adımlarıyla devam edildi. Veri tabanı için bir kullanıcı tanımlandı ve bu kullanıcı için gerekli yetkilendirmeler hem arayüzden hem de kodlarla yapıldı. Kullanıcıya ekleme, değiştirme ve silme gibi yetkiler verildi. Daha sonra, oluşturulan kullanıcıdan tüm yetkilerin kaldırılması ve kullanıcının sistemden silinmesi işlemi gerçekleştirildi.

Veri tabanının yedeklenmesi ve geri yüklenmesi adımlarıyla veri güvenliği sağlandı. Ayrıca, otomatik yedekleme planları oluşturularak veri kaybı riski minimize edildi. Aylık ve günlük yedekleme planları belirlendi ve düzenlendi. Bu süreçlerin tamamlanmasıyla birlikte, veri tabanı yönetimi ve güvenliği sağlanmış oldu.

BÖLÜM: İ

GİRİŞ

1

AMAÇ

Bu FÖY raporu, SQL Server üzerinde bir veri tabanı yönetimi ve güvenliği sürecini anlatmaktadır. Amacı, veri tabanı oluşturma, tablolar arası ilişkilerin kurulması, kullanıcı yönetimi ve yetkilendirme işlemlerinin gerçekleştirilmesi, veri yedekleme ve geri yükleme süreçlerinin planlanması ve otomatikleştirilmesi olarak özetlenebilir. Bu işlemler, veri tabanlarının doğru bir şekilde oluşturulması, veri bütünlüğünün sağlanması, veri güvenliğinin artırılması ve veri kaybı riskinin azaltılması amacıyla gerçekleştirilmektedir. Sonuç olarak, bu raporun amacı, SQL Server üzerindeki veri tabanı yönetimi süreçlerini adım adım açıklamak ve bu süreçlerin başarılı bir şekilde tamamlanmasıyla veri tabanlarının güvenli ve sağlıklı bir şekilde işlemesini sağlamaktır.

BÖLÜM: İİ

MATERYAL

2

UYGULAMALAR

Microsoft SQL Server 2017 Kurulumu ve SQL Server Management Studio Yüklemesi yapılmıştır.

2.1 *Microsoft SQL Server 2017*

Microsoft SQL Server 2017, işletmelerin veri yönetimi, analizi, raporlama ve iş zekası gibi çeşitli gereksinimlerini karşılayan bir ilişkisel veritabanı yönetim sistemidir. Büyük miktarlarda veri depolama, veri güvenliği sağlama, veri analizi yapma, iş zekası çözümleri oluşturma ve uygulama geliştirme için kullanılır. Ayrıca bulut tabanlı platformlarla entegrasyon sağlayarak ölçeklenebilirlik ve esneklik sunar..

2.2 *SQL Server Management Studio (SSMS)*

SQL Server Management Studio (SSMS), Microsoft SQL Server'ı yönetmek, veritabanı nesnelerini yönetmek, sorguları çalıştırmak, veri işlemek ve performansı izlemek için kullanılan

entegre bir geliştirme ortamıdır. Veritabanı yöneticileri ve geliştiriciler için temel bir araç olan SSMS, SQL Server'a erişimi kolaylaştırır ve yönetimi daha verimli hale getirir.

BÖLÜM: İİİ

YÖNTEM

3

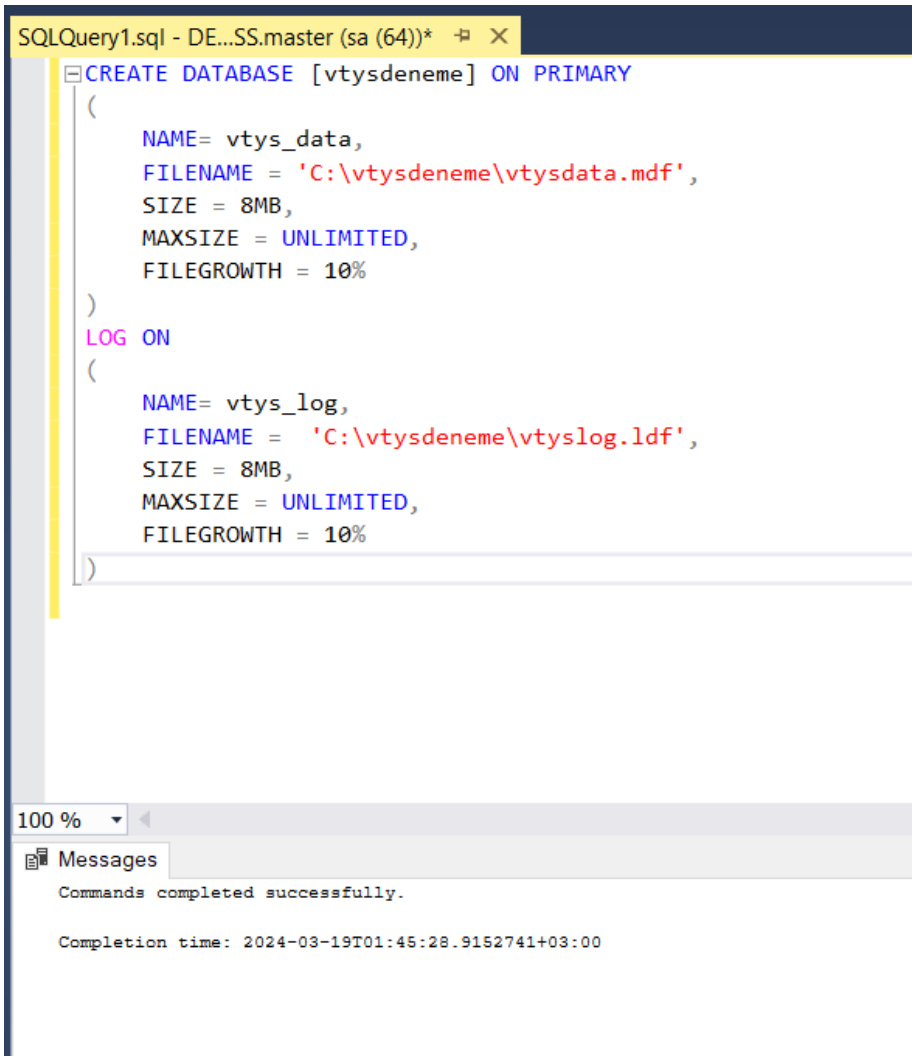
SQL SERVER

Bu rapor, SQL Server üzerinde bir veritabanı oluşturma, yönetme ve yedekleme işlemlerini kapsamaktadır.

3.1 *SQL Server Kurulumu*

Microsoft SQL Server 2017 Kurulumu ve SQL Server Management Studio Yüklemesi yapılmıştır. Microsoft SQL Server 2017 kurulumu ve SQL Server Management Studio (SSMS) yüklemesi, modern veri yönetimi gereksinimlerini karşılamak için temel adımlardır. SQL Server, büyük ölçekli veri depolama, işleme ve analiz işlevlerini sağlayarak kurumsal düzeyde veri yönetimi çözümleri sunar. Kurulum süreci, SQL Server'ın bilgisayar ortamına başarıyla entegre edilmesini sağlar, böylece veri tabanları oluşturulabilir, yönetilebilir ve güvenli bir şekilde saklanabilir. Ayrıca, SQL Server Management Studio'nun yüklenmesi, grafik arayüzü aracılığıyla veri tabanlarının yönetimini kolaylaştırır. SSMS, veri tabanı nesnelerinin görsel olarak tanımlanmasını ve yönetilmesini sağlar, aynı zamanda sorgu çalıştırma, yedekleme ve geri yükleme gibi önemli görevleri gerçekleştirmek için kullanılır.

3.2 *Veritabanı Oluşturma ve Tabloların Eklenmesi*



Şekil 1: Veritabanı Oluşturma

Bu sorgu [1], "vtysdeneme" adında bir SQL Server veritabanı oluşturur. Veritabanı, `vtys_data` adında bir veri dosyası ve `vtys_log` adında bir işlem günlüğü dosyası içerir. Her ikisi de C'de bulunan vtysdeneme klasöründe depolanır. Veritabanı ve işlem günlüğü dosyalarının başlangıç boyutu 8 MB olarak belirlenir ve her biri maksimum boyutları sınırsız olacak şekilde yapılandırılır. Dosyalar her büyüdüğünde %10 oranında büyür.

SQL kodu [2], "vtysdeneme" adında bir veritabanının içine beş farklı tablo yerleştirir. Bu

SQLQuery4.sql - DE...SS.master (sa (53))*

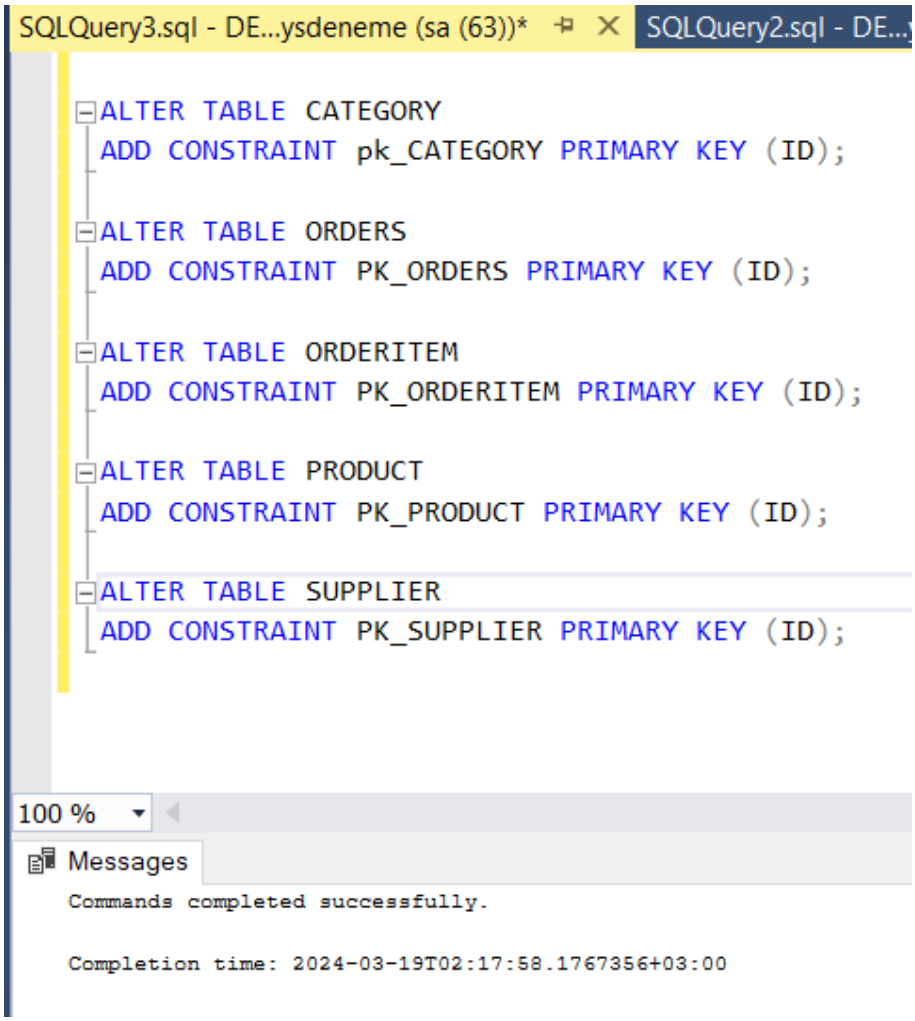
```
CREATE TABLE ORDERS (
    ID INT NOT NULL,
    ORDERDATE DATETIME NOT NULL,
    ORDERNUMBER NVARCHAR(10) NULL,
    CUSTOMERID INT NOT NULL,
    TOTALAMOUNT DECIMAL(12,2) NULL,
    ISCOMPLETE BIT
)

CREATE TABLE ORDERITEM (
    ID INT NOT NULL,
    ORDERID INT NOT NULL,
    PRODUCTID INT NOT NULL,
    UNITPRICE DECIMAL(12,2) NOT NULL,
    QUANTITY INT NOT NULL,
    ISAVAILABLE BIT
)

CREATE TABLE PRODUCT (
    ID INT NOT NULL,
    PRODUCTNAME NVARCHAR(50) NOT NULL,
    SUPPLIERID INT NOT NULL,
    UNITPRICE DECIMAL(12,2) NOT NULL,
    PACKAGE NVARCHAR(30) NULL,
    ISDISCONTINUED BIT NULL
)

CREATE TABLE SUPPLIER (
    ID INT NOT NULL,
    COMPANYNAME NVARCHAR(40) NOT NULL,
    CONTACTNAME NVARCHAR(50) NULL,
    CONTACTTITLE NVARCHAR(40) NULL,
    CITY NVARCHAR(40) NULL,
    COUNTRY NVARCHAR(40) NULL,
    PHONE NVARCHAR(20) NULL,
    FAX NVARCHAR(30) NULL
)
CREATE TABLE CATEGORY (
    ID INT NOT NULL,
    CATEGORYNAME NVARCHAR(50) NOT NULL,
    DESCRIPTION NVARCHAR(MAX) NULL,
    CREATEDDATE DATETIME NOT NULL,
    ISACTIVE BIT
)
```

70 %

Şekil 2: Tablolar

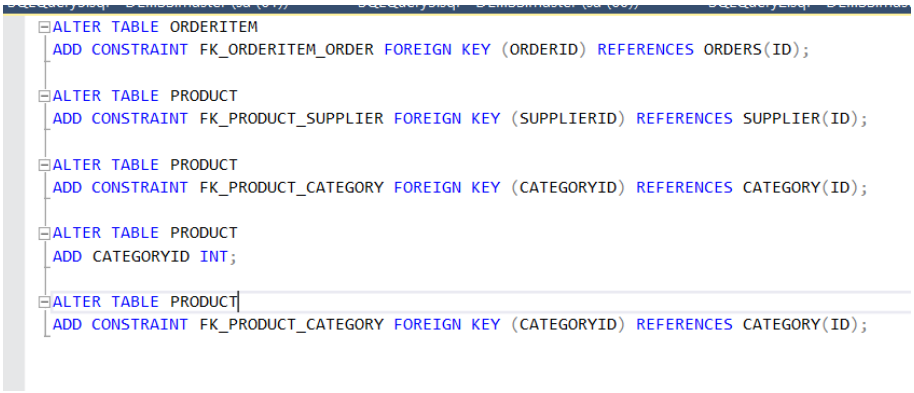
tablolar, bir ticari işletmenin işleyişini takip etmek için temel veri yapılarını oluşturur. Her bir tablonun belirli bir amacı ve içeriği vardır. Tablolar 5 farklı veri tipinden oluşmuştur. Tabloların veri tipleri, her sütunun hangi tür veriyi sakladığını belirtir ve bu veri tipleri tabloların veri bütünlüğünü sağlamak için önemlidir. Veri tipleri, saklanacak verinin doğası ve boyutu gibi faktörlere bağlı olarak seçilir.

3.3 *Foreign-Key ve Primary-Key Bağlantısı*



Şekil 3: Primary Key

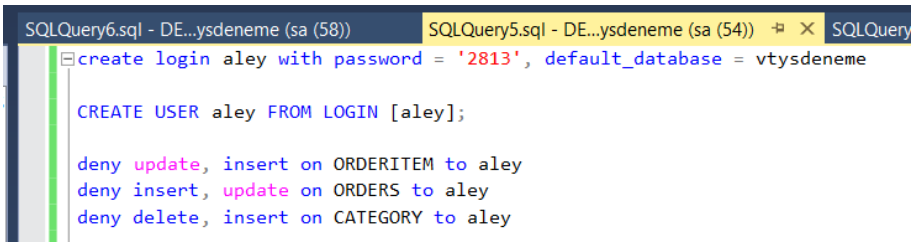
Her bir ALTER TABLE ifadesi, ilgili tabloya birincil anahtar kısıtlaması ekler. Birincil anahtarlar, tablodaki her satırı benzersiz olarak tanımlamak için kullanılan sütun veya sütunlar kümesidir. Bu sorgular [3], veritabanı modelinin veri bütünlüğünü sağlamak için kullanılır. Veritabanı tasarımında, foreign keyler tablolar arasında ilişkileri tanımlamak için kullanılır. Bu ilişkiler [4], bir tablodaki bir alanın diğer bir tablodaki bir alanı referans almasıyla sağlanır. Örneğin, "ORDERITEM" tablosundaki "ORDERID" alanı, "ORDERS" tablosundaki "ID" alanını referans alarak her bir siparişi belirtir. Benzer şekilde, "PRODUCT" tablosundaki



Şekil 4: Foreign Key

"SUPPLIERID" alanı, "SUPPLIER" tablosundaki "ID" alanını referans alarak her bir ürünün tedarikçisini belirtir. Bu foreign key bağlantıları, veri bütünlüğünü sağlar ve veritabanındaki ilişkisel yapının düzenli ve doğru olmasını sağlar. Bu sayede, veritabanında tutulan verilerin tutarlılığı ve doğruluğu artar, veri analizi ve raporlama gibi işlemler daha güvenilir hale gelir.

3.4 *Kullanıcı Tanımlama ve Yetkilendirme*



Şekil 5: Kullanıcı Tanımlama ve Yetkilendirme

Tablolara erişim yetkilerini kontrol etmek için SQL komutlarını [5] kullanarak 'aley' kullanıcısına belirli izinleri ve reddedilen yetkileri tanımlıyoruz. Bu komutlar sayesinde 'aley' kullanıcısının sadece belirli tablolarda belirli işlemleri yapmasına izin veriyoruz ve diğer işlemleri reddediyoruz. Öncelikle, CREATE LOGIN aley WITH PASSWORD = '2813',

`DEFAULT_DATABASE = vtysdeneme` komutuyla 'aley' adında bir giriş oluşturuyoruz. Bu giriş '2813' şifresiyle korunacak ve varsayılan olarak 'vtysdeneme' veritabanına yönlendirilecek.

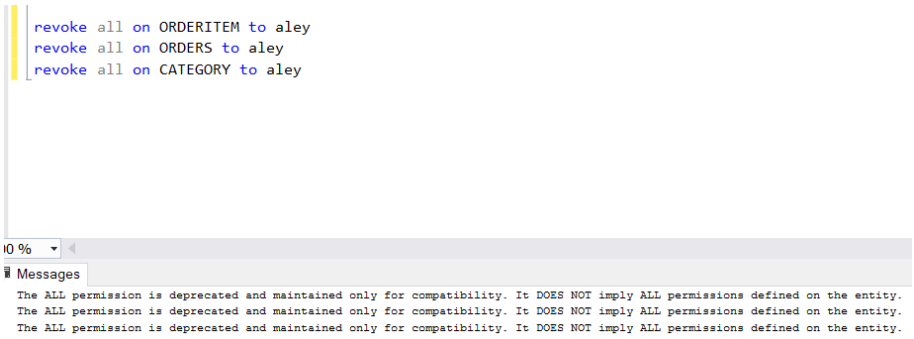
Daha sonra, CREATE USER aley FROM LOGIN [aley] komutuyla 'aley' kullanıcısını 'aley' girişiyle ilişkilendiriyoruz. Bu, 'aley' kullanıcısının 'aley' girişiyle oturum açabileceği anlamına gelir.

Sonrasında, DENY UPDATE, INSERT ON ORDERITEM TO aley komutuyla 'aley' kullanıcısına ORDERITEM tablosunda güncelleme ve ekleme işlemlerini reddediyoruz. Yani, 'aley' bu tablodaki verilere güncelleme veya ekleme yapamaz.

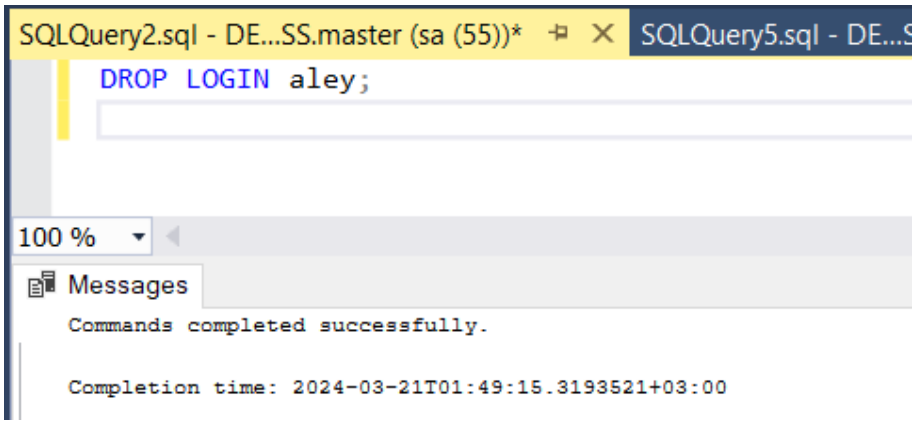
Benzer şekilde, DENY INSERT, UPDATE ON ORDERS TO aley ve DENY DELETE, INSERT ON CATEGORY TO aley komutlarıyla 'aley' kullanıcısına sırasıyla ORDERS ve CATEGORY tablolarında belirli işlemleri reddediyoruz. Bu sayede, 'aley' sadece belirli işlemleri yapmasına izin verilen belirli tablolara sınırlı erişime sahip olur.

3.5 *Yetkilerin Kaldırılması ve Kullanıcının Silinmesi*

SQL kodu [6], 'aley' kullanıcısının ORDERITEM, ORDERS ve CATEGORY tablolarındaki tüm yetkilerinin geri alınmasını sağlar. Yani, bu tablolarda 'aley' kullanıcısına daha önce verilen tüm erişim ve işlem yetkileri iptal edilir.



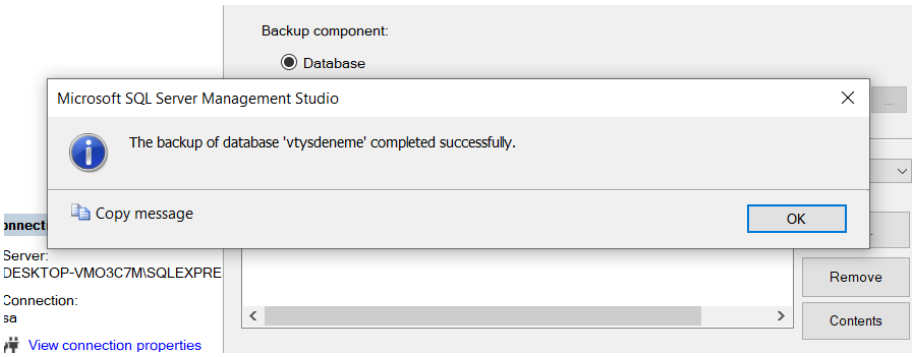
Şekil 6: Yetkilerin Kaldırılması



Şekil 7: Kullanıcı Silme

SQL komutu [7], "aley" adında belirtilen bir kullanıcıyı Microsoft SQL Server'dan siler.

3.6 *Veritabanının Yedeklenmesi ve Geri Yüklenmesi*



Şekil 8: Veritabanı Yedeklemesi

Yedekleme işlemi başarıyla tamamlanmıştır.

3.7 *Otomatik Yedekleme Planı*

BÖLÜM: İV

SONUÇ

4

SONUÇ

Föy raporu kapsamında SQL Server kurulumu tamamlandı ve istenilen adımlar takip edilerek bir veritabanı oluşturuldu. Bu veritabanına beş tablo eklendi ve her bir tablo içinde en az beş farklı veri tipi kullanıldı. Ayrıca tablolar arasında foreign-key ve primary-key bağlantıları yapıldı.

Kullanıcı tanımlama ve yetkilendirme adımları da başarıyla gerçekleştirildi. Hem arayüzden hem de kodlarla kullanıcıya eklemeyi, değişiklik yapmayı ve silmeyi sağlayan yetkiler verildi. Sonrasında oluşturulan kullanıcıdan bütün yetkiler alındı ve kullanıcı silindi.

Yedekleme işlemi yapılmıştır. Fakat SQL Server 2017'nin express sürümü yedekleme işlemlerini otomatik olarak gerçekleştirmek için gerekli özellikleri sağlamamaktadır. Bu nedenle otomatik yedekleme işlemleri yapılamamıştır.