

Öğrenci Adı Soyadı :

Öğrenci Numarası:

T.C. MARMARA ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ
2024-2025 EĞİTİM ÖĞRETİM YILI
GÜZ DÖNEMİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
VERİTABANI YÖNETİM SİSTEMLERİ DERSİ
PROJE-FİNAL DÖKÜMANI

Tüm sorular soru metnini altındaki alanlara cevaplandırılacaktır. Bu belgenin formatını bozmayınız, sadece gerekli alanları doldurunuz.

1. Gerçekleştirdiğiniz veri tabanı projesi için grup arkadaşlarınızın isimlerini yazınız ve projenize ait veri tabanı/diğer yazılım bileşenleri hakkında bilgi veriniz. (5 p)

Grup Üyeleri: Orçun Samet Tatar Berfin Petek Ali Batuhan Dağcı

Projemiz, akıllı ev cihazlarının merkezi bir sistemden yönetimini sağlamaya odaklanmıştır. MSSQL kullanılarak geliştirilmiş bir veritabanı altyapısı ile cihaz ve oda bilgilerini yönetmeyi hedefledik. Veritabanımız, cihazların durumlarını ve kullanıcı etkileşimlerini kayıt altına alırken, Python ile geliştirilecek bir arayüz üzerinden bu verilere erişim ve kontrol imkânı sunmayı amaçlamaktadır. Projede kullanıcılar, odalar, cihazlar, cihaz özellikleri ve kontrol bilgileri arasındaki ilişkileri tanımlayan bir mimari bulunmaktadır. Ayrıca, güvenlik önlemleri için kullanıcı bazlı oda erişim izinleri de sağlanmıştır.

2. Gerçekleştirdiğiniz veri tabanı projesi için proje dokümanınızı ve dosyalarınızı içeren herkese açık github bağlantılarınızı paylaşınız. (5 p)

<https://github.com/Aidiaru/VYTS>

3. Gerçekleştirdiğiniz projenin amacını detaylı bir şekilde açıklayınız. (10 p)

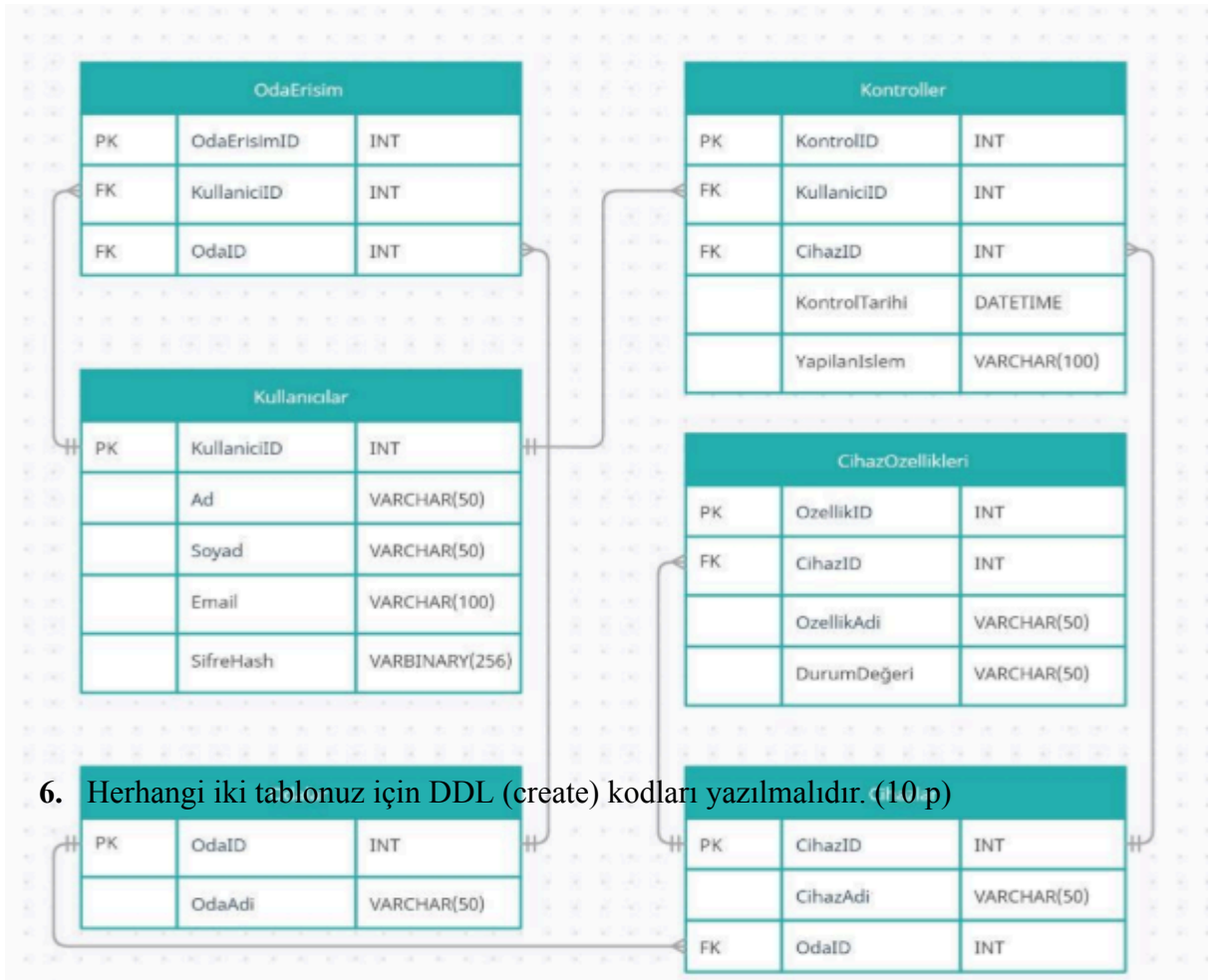
Projenin amacı, akıllı ev cihazlarının yönetimini kolaylaştırmak için bir veritabanı oluşturmak ve bu cihazların durumlarını, kullanıcı kontrollerini, odalara erişim izinlerini merkezi bir sistemde yönetmektir. Bu sistem, kullanıcı deneyimini artırmak ve cihazların durumlarının düzenli bir şekilde izlenmesini sağlamayı hedefler.

4. Tasarladığınız veri tabanı mimarisinde hangi tablo ve ilişkileri kullanıldığını açıklayınız. (10 p)

Kullanıcılar: Kullanıcı bilgilerini (ad, soyad, e-posta, şifre hash) saklar.
Odalar: Evin odalarına ait bilgileri tutar.
Cihazlar: Her cihazın adı ve bulunduğu odayı içerir.
Kontroller: Kullanıcıların cihazlarla yaptığı işlemleri kaydeder.
Cihaz Özellikleri: Cihazların sahip olduğu özellikler ve durum bilgilerini içerir.
Oda Erişim: Kullanıcıların hangi odalara erişim izni olduğunu belirtir.

Kullanıcılar ↔ Oda Erişim (çoktan çoğa)
Odalar ↔ Cihazlar (bire çok)
Cihazlar ↔ Cihaz Özellikleri (bire çok)
Cihazlar ↔ Kontroller (bire çok)

5. Veri tabanı ER (Entity Relationship) diagramının bilgisayar ortamında çizilmiş halini paylaşınız. (Ara raporda eksik kısımlar bu raporda giderilmelidir ve ER çizme programlarından faydalanılabilir. Elle çizim, çizip fotoğrafını çekme vb. kabul edilmeyecektir.) (10 p)



```
CREATE TABLE Cihazlar (  
  CihazID INT PRIMARY KEY IDENTITY(1,1),  
  CihazAdi VARCHAR(50) NOT NULL,  
  OdaID INT NOT NULL,  
  FOREIGN KEY (OdaID) REFERENCES Odalar(OdaID) ON DELETE  
  CASCADE  
);
```

```
CREATE TABLE CihazOzellikleri (  
  OzellikID INT PRIMARY KEY IDENTITY(1,1),  
  CihazID INT NOT NULL,  
  OzellikAdi VARCHAR(50) NOT NULL,  
  DurumDeğeri VARCHAR(50) NULL,  
  FOREIGN KEY (CihazID) REFERENCES Cihazlar(CihazID) ON DELETE  
  CASCADE  
);
```

7. 5 adet DML (update, insert, delete) içeren kodları yazılmalıdır. (10 p)

```
INSERT INTO Kullanıcılar (Ad, Soyad, Email, SifreHash)  
VALUES ('Ahmet', 'Kaya', 'ahmet.kaya@example.com',  
HASHBYTES('SHA2_256', 'password123'));  
  
INSERT INTO Odalar (OdaAdi) VALUES ('Oturma Odası');  
  
INSERT INTO Cihazlar (CihazAdi, OdaID) VALUES ('Termostat',  
1);  
  
UPDATE CihazOzellikleri SET DurumDeğeri = '24' WHERE  
OzellikAdi = 'Sıcaklık Derecesi';  
DELETE FROM Kullanıcılar WHERE KullaniciID = 3;
```

8. Projenize ait kendi belirlediğiniz 10 adet SQL sorgusu yazınız, sorguların amacını ve sonuç çıktısını da lütfen ekleyiniz. (Açıklama: Sorgular Select deyimleri ve gruptama fonksiyonlarını HAVING deyimini (min, max, avg, count gibi) ve join deyimlerini (en az iki tablo ile birleştirme sorgusu) içerecek şekilde basitten

karmaşığa doğru gitmelidir. Proje sunum anında veri tabanınıza ait sorular SQL ortamında gösterilecek ve açıklanacaktır. Raporunuzda ise sorgular, sorguların cevap ve sonuçlarının ekran görüntüsü olarak paylaşılması beklenmektedir. (10 p)
Örnek:

Sorgu 1: Tüm odaları listeleme

Amaç: Veritabanındaki tüm oda kayıtlarını listelemek.

```
SELECT * FROM Odalar;
```

OdaID	OdaAdi
1	Oturma Odası
2	Mutfak

Sorgu 2: Kullanıcı başına düşen cihaz sayısı

Amaç: Her kullanıcının kaç farklı cihazı kontrol ettiğini bulmak.

```
SELECT Kullanıcılar.Ad, Kullanıcılar.Soyad,  
COUNT(Kontroller.CihazID) AS CihazSayisi
```

```
FROM Kullanıcılar
JOIN Kontroller ON Kullanıcılar.KullaniciID =
Kontroller.KullaniciID
GROUP BY Kullanıcılar.Ad, Kullanıcılar.Soyad;
```

AD	SOYAD	CİHAZ SAYISI
Ahmet	Kaya	3
Ayşe	Demir	2

Sorgu 3: En çok kontrol edilen cihaz

Amaç: Hangi cihazın en fazla kontrol edildiğini tespit etmek.

```
SELECT TOP 1 CihazAdi, COUNT(*) AS KontrolSayisi
FROM Kontroller
JOIN Cihazlar ON Kontroller.CihazID = Cihazlar.CihazID
GROUP BY CihazAdi
ORDER BY KontrolSayisi DESC;
```

CİHAZ ADI	KONTROL SAYISI
Termostat	15

Sorgu 4: Bir kullanıcının erişimi olan odalar

Amaç: Belirli bir kullanıcının hangi odalara erişim yetkisi olduğunu görmek.

```
SELECT Kullanıcılar.Ad, Kullanıcılar.Soyad, Odalar.OdaAdi
FROM Kullanıcılar
JOIN OdaErisim ON Kullanıcılar.KullaniciID =
OdaErisim.KullaniciID
JOIN Odalar ON OdaErisim.OdaID = Odalar.OdaID
WHERE Kullanıcılar.Ad = 'Ahmet';
```

AD	SOYAD	ODA ADI
Ahmet	Kaya	Oturma Odası
Ahmet	Kaya	Mutfak

Sorgu 5: Her cihazın ortalama kontrol sayısı

Amaç: Her cihazın kontrol edilme sayısının ortalamasını bulmak

```
SELECT CihazAdi, AVG(KontrolSayisi) AS OrtalamaKontrol
FROM (
    SELECT CihazID, COUNT(*) AS KontrolSayisi
    FROM Kontroller
    GROUP BY CihazID
) AS AltSorgu
JOIN Cihazlar ON AltSorgu.CihazID = Cihazlar.CihazID
GROUP BY CihazAdi;
```

Cihaz Adı	Ortalama Kontrol
Termostat	5.0
Işık	3.0

Sorgu 6: Belirli bir cihazı kontrol eden kullanıcılar

Amaç: Termostat cihazını kimlerin kontrol ettiğini listelemek.

```
SELECT Kullanıcılar.Ad, Kullanıcılar.Soyad
FROM Kullanıcılar
JOIN Kontroller ON Kullanıcılar.KullaniciID =
Kontroller.KullaniciID
JOIN Cihazlar ON Kontroller.CihazID = Cihazlar.CihazID
WHERE CihazAdi = 'Termostat';
```

AD	SOYAD
Ahmet	Kaya
Ayşe	Demir

Sorgu 7: Her odadaki cihaz sayısı

Amaç: Her odada bulunan cihaz sayısını tespit etmek.

```
SELECT Odalar.OdaAdi, COUNT(Cihazlar.CihazID) AS  
CihazSayisi  
FROM Odalar  
LEFT JOIN Cihazlar ON Odalar.OdaID = Cihazlar.OdaID  
GROUP BY Odalar.OdaAdi;
```

ODA ADI	CİHAZ SAYISI
Oturma Odası	2
Mutfak	1
Yatak Odası	0

Sorgu 8: Kontrol edilen cihazların toplam sayısı

Amaç: Kontrol edilen benzersiz cihazların sayısını bulmak.

```
SELECT COUNT(DISTINCT CihazID) AS ToplamKontrolEdilenCihaz  
FROM Kontroller;
```

Toplam Kontrol Eilen Cihaz Sayısı : 3

Sorgu 9: Son bir hafta içinde yapılan kontroller

Amaç: Son bir hafta içinde hangi cihazların kontrol edildiğini listelemek.

```
SELECT CihazAdi, Kontroller.KontrolTarihi
FROM Kontroller
JOIN Cihazlar ON Kontroller.CihazID = Cihazlar.CihazID
WHERE KontrolTarihi >= DATEADD(DAY, -7, GETDATE());
```

CİHAZ ADI	KONTROL TARİHİ
Termostat	2024-12-18 14:30:00
Işık	2024-12-18 14:45:00

Sorgu 10: Her kullanıcının son kontrol tarihi

Amaç: Her kullanıcının en son ne zaman cihaz kontrol ettiğini bulmak.

```
SELECT Kullanıcılar.Ad, Kullanıcılar.Soyad,
MAX(Kontroller.KontrolTarihi) AS SonKontrolTarihi
FROM Kullanıcılar
JOIN Kontroller ON Kullanıcılar.KullaniciID =
Kontroller.KullaniciID
GROUP BY Kullanıcılar.Ad, Kullanıcılar.Soyad;;
```

AD	SOYAD	SON KONTROL TARİHİ
Ahmet	Kaya	2024-12-18 14:30:00
Ayşe	Demir	2024-12-18 14:38:00

9. Eğer gerçekleştirmiş iseniz, veri tabanı bağlama ve uygulama geliştirme aşamalarınızı kısaca açıklayarak, kullanıcı ara yüz ekranından bir örnek veriniz. Ve geliştirdiğiniz ara yüzü anlatınız. (10 p)

Projemizde, kullanıcı arayüzü Python ile geliştirilmiş, MSSQL ile veritabanı tasarımı gerçekleştirilmiştir. Veritabanı bağlama işlemi tamamlanmış olup aşağıdaki aşamalar izlenmiştir:

1. **Veritabanı Tasarımı:**

- MSSQL kullanılarak "Kullanıcılar", "Odalar", "Cihazlar" ve "Cihaz Özellikleri" tabloları oluşturulmuştur.
- Tablolar arası ilişkiler belirlenmiş ve gerekli birincil ve yabancı anahtar tanımlamaları yapılmıştır.

2. **Uygulama Geliştirme:**

- **Arayüz Tasarımı:** Python'da Tkinter kütüphanesi ile kullanıcı dostu bir arayüz oluşturulmuştur.
- **Özellikler:**
 - Kullanıcı ekleme, silme ve düzenleme.
 - Odaların ve cihazların yönetimi.
 - Cihazlara özel kontrol özellikleri (örn: ışık seviyesi ayarı).
- **Veritabanı Bağlantısı:** pyodbc kütüphanesi kullanılarak MSSQL ile bağlantı sağlanmıştır.

3. **Kullanıcı Arayüz Örneği:**

- "Kullanıcılar", "Odalar", "Kontroller" ve "Cihazlar" sekmeleri olan bir ana ekran tasarlanmıştır.
- Kullanıcıların odalara atanması ve cihaz özelliklerinin düzenlenmesi işlevleri arayüzden sağlanmaktadır.

Akıllı Ev Yönetimi

Kullanıcılar Odalar Cihazlar Kontroller

Cihaz

Özellik

Termostat

Isık

Hoparlör

22

Aık

100

Cihaz Adı:

Isık2

Cihaz Ekle

Cihaz Sil

Kontrol:

Aık-Kapalı

Derece

Parlaklık

Kontrol Ata

Cihazlar:

Isık2

Özellikler:

Değer:

Set

Akıllı Ev Yönetimi

Kullanıcılar Odalar Cihazlar Kontroller

Oda

Oturma Odası

Mutfak

Oda Adı:

Oda Ekle

Oda Sil

Akıllı Ev Yönetimi

Kullanıcılar Odalar Cihazlar Kontroller

Ad:

Soyad:

Email:

Şifre:

Kullanıcı Ekle

Kullanıcı

Oda

Azıye Deme

Yok

Kullanıcı Sil

Odalar:

Oturma Odası

Mutfak

Oda Ata

Akıllı Ev Yönetimi

Kullanıcılar Odalar Cihazlar Kontroller

Kontrol ID	Kullanıcı ID	Cihaz ID	Kontrol Tarihi	Yapılan İşlem
1	101	201	2024-12-25 12:00:00	Açıldı
2	102	202	2024-12-24 15:30:00	Kapandı

Geçmiş Sil

Cihaz Özellikleri:

WiFi

Bluetooth

GPS

Yeni Özellik:

Cihaz Seç:

Kontrol Ekle

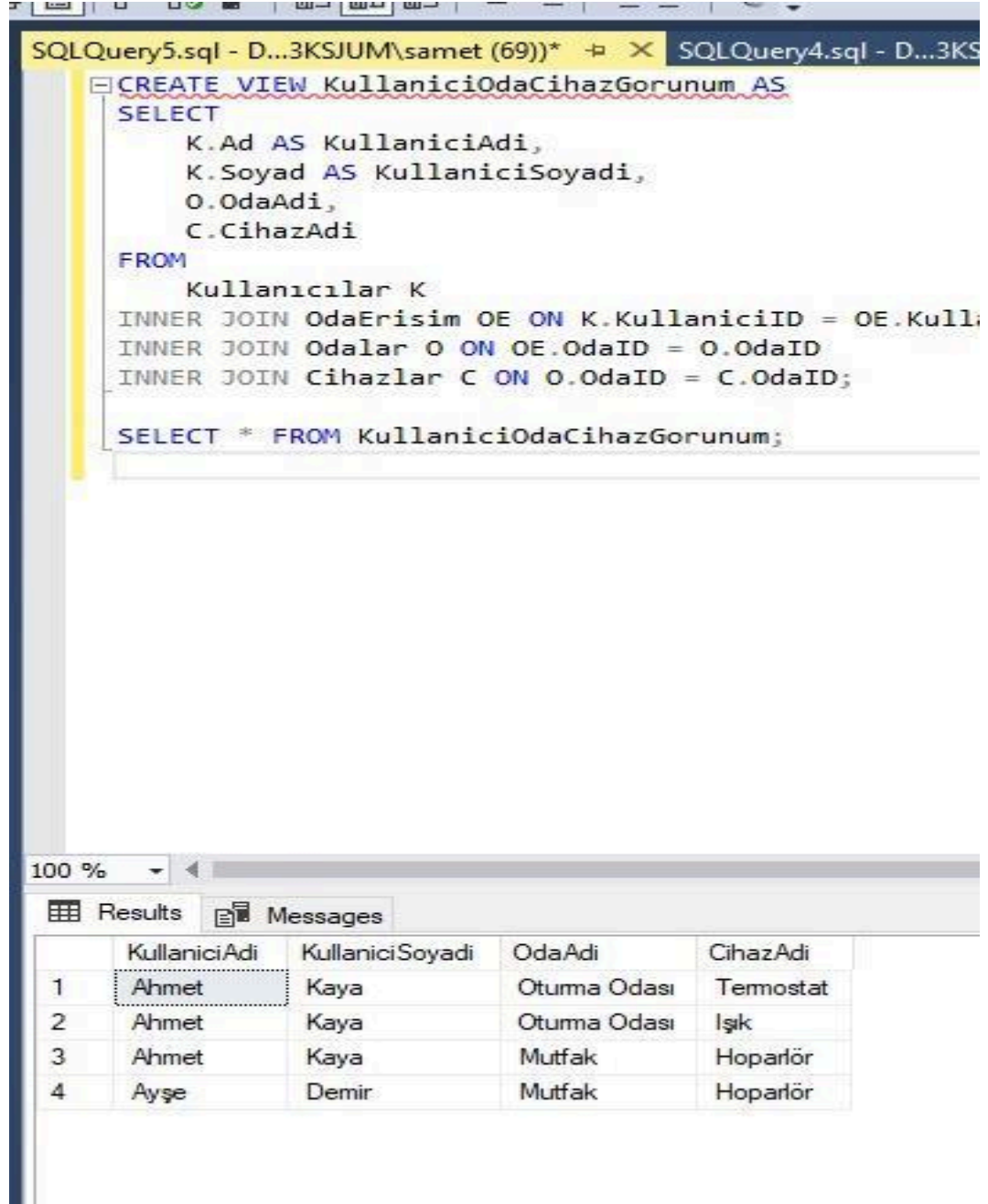
10. Eğer veri tabanı bağlama işlemini gerçekleştirmemiş iseniz VTYS sistemlerinde Transaction nedir açıklayınız ve çalışmanızdan bir Transaction örneği veriniz. (10 p)

11. View nedir açıklayınız ve bir adet view, bir adet saklı yordam (Stored Procedure) ifadesine ait SQL deyimlerinin sorgusunu ve cevabını yazınız. (10 p)

View Nedir?

View, birden fazla tablodan veri almak veya karmaşık sorguları sadeleştirmek için kullanılan sanal bir tablodur. View'lar, fiziksel olarak veri tutmaz ve yalnızca sorgu sonucunu döner. Projenizde akıllı ev cihazlarının durumlarını ve odalara göre dağılımını kolayca görüntülemek için View kullanılabilir.

View Örneği:



The screenshot shows a SQL Server Enterprise Manager window with a SQL query editor. The query is as follows:

```
CREATE VIEW KullaniciOdaCihazGorunum AS
SELECT
    K.Ad AS KullaniciAdi,
    K.Soyad AS KullaniciSoyadi,
    O.OdaAdi,
    C.CihazAdi
FROM
    Kullanicilar K
INNER JOIN OdaErisim OE ON K.KullaniciID = OE.Kull
INNER JOIN Odalar O ON OE.OdaID = O.OdaID
INNER JOIN Cihazlar C ON O.OdaID = C.OdaID;

SELECT * FROM KullaniciOdaCihazGorunum;
```

Below the query editor, the 'Results' tab is active, displaying the following data:

	KullaniciAdi	KullaniciSoyadi	OdaAdi	CihazAdi
1	Ahmet	Kaya	Oturma Odası	Termostat
2	Ahmet	Kaya	Oturma Odası	Işık
3	Ahmet	Kaya	Mutfak	Hoparlör
4	Ayşe	Demir	Mutfak	Hoparlör

Stored Procedure Nedir?

Stored Procedure, birden fazla SQL sorgusunu bir araya getirerek sık kullanılan işlemleri otomatikleştirir. Parametre alabilir ve veritabanında değişiklik yapabilir.

```
SQLQuery5.sql - D...3KSJUM\samet (69))* X SQLQuery4.sql - D...3KSJUM\samet (61))* SQLQuery3.sql - D...
CREATE PROCEDURE KontrolEkle
    @KullaniciID INT,
    @CihazID INT,
    @YapilanIslem VARCHAR(100)
AS
BEGIN
    -- Kontrol tablosuna işlem ekle
    INSERT INTO Kontroller (KullaniciID, CihazID, YapilanIslem)
    VALUES (@KullaniciID, @CihazID, @YapilanIslem);

    -- İşlemin başarıyla tamamlandığını bildirin
    PRINT 'Kontrol kaydı başarıyla eklendi.';
END;

EXEC KontrolEkle @KullaniciID = 1, @CihazID = 2, @YapilanIslem = 'Işığı Kapattı';

SELECT * FROM Kontroller
```

100 %

Results Messages

	KontrollID	KullaniciID	CihazID	KontrolTarihi	YapilanIslem
1	1	1	2	2024-12-25 22:24:13.220	Işığı Kapattı