

# ONLINE ALIŞVERİŞ PLATFORMU OLUŞTURMA PROJESİ

## 1.Projenin Amacı

Bu proje bir e-ticaret platformunda gerçekleşen, müşteri, kategori, satıcı, ürün, sipariş ve sipariş detayları verileri ile çeşitli sorgular yaparak bir veri tabanı sistemi oluşturmak amacıyla hazırlanmıştır.

## 1.2.Projenin Veri Yapısı

Veri tabanı Alisveris\_Platformu olarak adlandırılmıştır ve 6 tane tablodan oluşmaktadır. Bu tablolar; dbo.Musteri, dbo.Kategori, dbo.Satici, dbo.Urun, dbo.Siparis ve dbo.Siparis\_Detay şeklindedir.

dbo.Musteri: Müşterilerin id, ad, soy ad, email, şehir ve kayıt tarihi bilgilerini içermektedir.

dbo.Kategori: Kategorilerin id ve ad bilgini içermektedir.

dbo.Satici: Satıcıların id, ad ve adres bilgisini içermektedir.

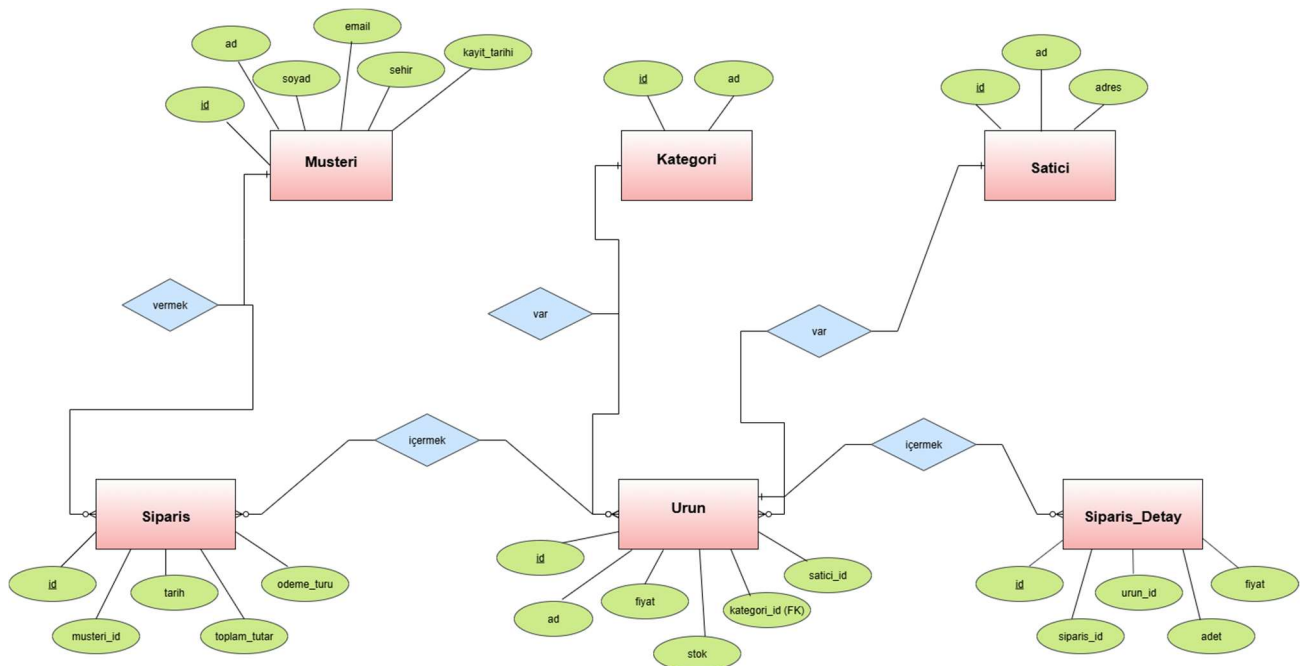
dbo.Urun: Ürünlerin id, ad, fiyat, stok, kategori id ve satıcı id bilgilerini içermektedir.

dbo.Siparis: Sipariş id, müşteri id, tarih, toplam tutar ve ödeme türü bilgilerini içermektedir.

dbo.Siparis\_Detay: Siparişlerin id, sipariş id, ürün id, adet ve fiyat bilgilerini içermektedir.

## 1.3.Projenin ER Diyagramı

Projenin ER Diyagramı aşağıda olduğu gibidir.



## 2.Projenin Hazırlanması

Projenin hazırlanması kısmında ilk başta CREATE komutu ile bir database oluşturdum ve sonra tabloları yine CREATE komutu ile oluşturdum. Primary key ile id'lerin birbirinden benzersiz ve boşluk olmadan oluşturulmasını sağladım. Tablolara gerekli verileri ise INSERT INTO komutu ile ekledim.

Örnek bir INSERT INTO komutu:

```
INSERT INTO dbo.Satici (ad, adres) VALUES  
    (N'GlobalTech Ticaret', N'Fatih-İstanbul'),  
    (N'Beyaz Moda Evi', N'Mamak-Ankara'),  
    (N'Renkli Bakım', N'Konak-İzmir'),  
    (N'Kitap Kurdu', N'Gebze-Kocaeli'),  
    (N'Aslan Süpermarket', N'Sarıyer-İstanbul');
```

DELETE komutu ile hatalı olan bir satırı ve satıcısı olmayan bir kategoriye sildim.

UPDATE komutu ile de stok verilerini güncelledim.

**Örnek:**

- **DELETE FROM** dbo.Siparis\_Detay  
**WHERE** id = 74;
- **UPDATE** U  
**SET** stok = U.stok - SD.SatılanAdet **FROM** dbo.Urun U  
**JOIN** (  
 **SELECT** urun\_id, **SUM**(adet) SatılanAdet **FROM** dbo.Siparis\_Detay  
 **GROUP BY** urun\_id  
 ) SD **ON** U.id = SD.urun\_id;

### 2.1.Projede Karşılaşılan Bazı Hatalar

Projede veri girişi yaparken ve sorgulamaları çalıştırırken bazı hatalarla karşılaşıldı.

Bu hatalardan bazıları şu şekilde:

- The INSERT statement conflicted with the CHECK constraint "CK\_\_Siparis\_\_odeme\_t\_\_4BAC3F29". The conflict occurred in database "Alisveris\_Platformu", table "dbo.Siparis", column 'odeme\_turu'.
- The INSERT statement conflicted with the FOREIGN KEY constraint "FK\_SDetay\_Siparis". The conflict occurred in database "Alisveris\_Platformu", table "dbo.Siparis", column 'id'.

- Column 'dbo.Musteri.id' is invalid in the select list because it is not contained in either an aggregate function or the GROUP BY clause.
- An aggregate may not appear in the WHERE clause unless it is in a subquery contained in a HAVING clause or a select list, and the column being aggregated is an outer reference.

Aldığım hatalardan biri veri girişi sırasında yanlış id girmemden kaynaklanmıştı.

Aldığım başka bir hata Aggregate fonksiyonu kullanıp GROUP BY kullanmamamdan dolayı kaynaklanmıştı.

Bunların haricinde karşılaştığım diğer sorunları çözemediğim zaman TRUNCATE TABLE ile tablodaki verileri temizleyip çözmeye çalıştım. Ancak her zaman işe yaramadı. Bu yüzden bazı durumlarda da DROP TABLE ile tabloyu sildim ve CREATE ile tekrar oluşturup INSERT INTO ile verileri tekrar ekledim.

### 3.Proje Veri Sorgulama

Oluşturulan projede istenilen sorguları SQL komutları ile analiz edildi.

Örnek bazı sorgular ve çıktıları:

- En çok sipariş veren 5 müşteri.

- `SELECT TOP 5 M.id, M.ad, M.soyad, COUNT(S.id) Siparis_Sayisi FROM dbo.Musteri M JOIN dbo.Siparis S ON M.id = S.musteri_id GROUP BY M.id, M.ad, M.soyad ORDER BY Siparis_Sayisi DESC;`

	id	ad	soyad	Siparis_Sayisi
1	8	Melike	Çınar	6
2	1	Sevgi	Asil	5
3	3	Ayşe	Yıldız	5
4	5	Coşkun	Atmaca	4
5	7	Mahmut	Çelik	4

- Şehirlere göre müşteri sayısı.

- `SELECT M.sehir, COUNT(M.id) Musteri_Sayisi FROM dbo.Musteri M GROUP BY M.sehir ORDER BY Musteri_Sayisi DESC;`

	sehir	Musteri_Sayisi
1	İstanbul	4
2	Kocaeli	3
3	Ankara	2
4	İzmir	2

- En çok kazanç sağlayan ilk 3 kategori.

- `SELECT TOP 3 K.id KategoriID, K.ad Kategori_Adi, SUM(SD.fiyat * SD.adet)`  
Toplam\_Kazanc `FROM` dbo.Kategori K  
`JOIN` dbo. Urun U `ON` K.id = U.kategori\_id  
`JOIN` dbo.Siparis\_Detay SD `ON` U.id = SD.urun\_id  
`GROUP BY` K.id, K.ad  
`ORDER BY` Toplam\_Kazanc `DESC`;

	KategoriID	Kategori_Adi	Toplam_Kazanc
1	1	Elektronik	261100.00
2	2	Giyim	7500.00
3	3	Kozmetik	4625.00

#### 4.SONUÇ

Bu çalışma kapsamında, online alışveriş platformunun gereksinimlerine uygun bir veri tabanı tasarımı oluşturulmuştur. İlişkisel veri tabanı modeli, ER Diyagramı ile görselleştirilmiş, veri bütünlüğü kısıtlamaları uygulanmış, örnek veri eklemeleri gerçekleştirilmiş ve bu veriler üzerinden istenen tüm analiz sorguları etkin bir şekilde tamamlanmıştır.

Saygılarımla,

Kübra Çam

05.10.2025