**Online Alışveriş Platformu Veri Tabanı**

1. **Projenin Amacı**

Bu projenin amacı, gerçek hayatta kullanılan online alışveriş platformlarının **(Trendyol, Hepsiburada, Amazon vb.)** veri tabanı yapısını modellemek, SQL ile tablolar oluşturmak, veri eklemek, güncellemek, silmek ve çeşitli raporlama sorguları yapmaktır.

**Proje ile birlikte öğrencinin:**

* Veri tabanı tasarımı yapabilmesi,
* SQL komutlarını etkin şekilde kullanabilmesi,
* JOIN, GROUP BY, HAVING gibi sorgularda ustalaşması,
* Gerçekçi senaryolarda rapor hazırlayabilmesi hedeflenmiştir.

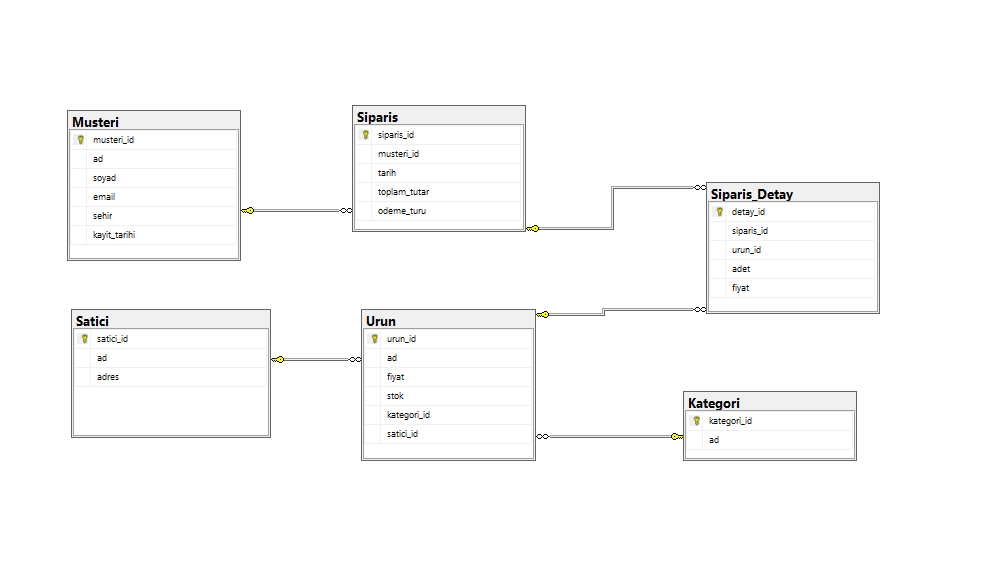
1. **Veri Tabanı Tasarımı**

Veri tabanı tasarımında aşağıdaki tablolar oluşturulmuştur:

* **Musteri:** Müşteri bilgilerini tutar.
* **Urun:** Satışa sunulan ürünleri tutar.
* **Kategori:** Ürünlerin bağlı olduğu kategorileri içerir.
* **Satici:** Ürünleri satan firmaların bilgilerini tutar.
* **Siparis:** Müşteri tarafından verilen siparişleri tutar.
* **Siparis\_Detay:** Siparişlerdeki ürünleri ve adetlerini tutar.

**İlişkiler:**

* Bir müşteri birden fazla sipariş verebilir.
* Bir sipariş birden fazla ürün içerebilir.
* Bir ürün bir kategoride yer alır.
* Bir ürün bir satıcıya aittir.

**ER Diyagramı:**

1. **SQL Scriptleri**
2. **Veri Tabanı ve Tabloların Oluşturulması**

CREATE TABLE Musteri (

musteri\_id INT PRIMARY KEY IDENTITY(1,1),

ad NVARCHAR(50) NOT NULL,

soyad NVARCHAR(50) NOT NULL,

email NVARCHAR(100) UNIQUE,

sehir NVARCHAR(50),

kayit\_tarihi DATE DEFAULT GETDATE()

);

CREATE TABLE Kategori (

kategori\_id INT PRIMARY KEY IDENTITY(1,1),

ad NVARCHAR(100) NOT NULL

);

CREATE TABLE Satici (

satici\_id INT PRIMARY KEY IDENTITY(1,1),

ad NVARCHAR(100) NOT NULL,

adres NVARCHAR(200)

);

CREATE TABLE Urun (

urun\_id INT PRIMARY KEY IDENTITY(1,1),

ad NVARCHAR(100) NOT NULL,

fiyat DECIMAL(10,2) NOT NULL,

stok INT NOT NULL,

kategori\_id INT FOREIGN KEY REFERENCES Kategori(kategori\_id),

satici\_id INT FOREIGN KEY REFERENCES Satici(satici\_id)

);

CREATE TABLE Siparis (

siparis\_id INT PRIMARY KEY IDENTITY(1,1),

musteri\_id INT FOREIGN KEY REFERENCES Musteri(musteri\_id),

tarih DATE DEFAULT GETDATE(),

toplam\_tutar DECIMAL(10,2),

odeme\_turu NVARCHAR(50)

);

CREATE TABLE Siparis\_Detay (

detay\_id INT PRIMARY KEY IDENTITY(1,1),

siparis\_id INT FOREIGN KEY REFERENCES Siparis(siparis\_id),

urun\_id INT FOREIGN KEY REFERENCES Urun(urun\_id),

adet INT NOT NULL,

fiyat DECIMAL(10,2) NOT NULL

);

1. **Veri Ekleme ve Güncelleme**

* Örnek müşteri, ürün, sipariş verileri INSERT ile eklendi.
* UPDATE ile stok güncellemeleri yapıldı.
* DELETE ile örnek silme işlemleri gösterildi.
* TRUNCATE ile tablo sıfırlama örneği verildi.

1. **Veri Sorgulama ve Raporlama**

**Hazırlanan örnek sorgular:**

* En çok sipariş veren 5 müşteri.
* En çok satılan ürünler.
* En yüksek cirosu olan satıcılar.
* Şehirlere göre müşteri sayısı.
* Kategori bazlı toplam satışlar.
* Aylara göre sipariş sayısı.
* Siparişlerde müşteri + ürün + satıcı bilgisi.
* Hiç satılmamış ürünler.
* Hiç sipariş vermemiş müşteriler.

1. **İleri Seviye Sorgular (Opsiyonel)**

* En çok kazanç sağlayan 3 kategori.
* Ortalama sipariş tutarını geçen siparişler.
* En az bir kez elektronik ürün satın alan müşteriler.

1. **Karşılaşılan Sorunlar ve Çözümler**

* **Foreign key hataları:** Tabloların eklenme sırasına dikkat edilerek çözüldü.
* **Stok güncellemeleri:** Sipariş sonrası stok düşmesi için güncelleme sorguları yazıldı.
* **JOIN karmaşıklıkları:** Aliases kullanılarak sorgular okunabilir hale getirildi.

1. **Sonuç**

Bu projede, online alışveriş platformunun veri tabanı tasarımı başarıyla gerçekleştirilmiş, SQL komutları ile veri ekleme, silme, güncelleme işlemleri yapılmış, çeşitli raporlar hazırlanmıştır.

Sonuç olarak, SQL’in gerçek hayat senaryolarındaki gücü ve önemi anlaşılmıştır.