

Veritabanı Tasarım Dokümantasyonu

Bu projede **basit bir e-ticaret veritabanı** tasarladım. Tabloları ihtiyaçlara göre belirledim:

- **Kategori:** Ürünlerin ait olduğu kategorileri tutuyor.
- **Müşteri:** Müşteri bilgilerini içeriyor (ad, soyad, email, şehir, kayıt tarihi).
- **Satıcı:** Ürünleri sağlayan satıcı bilgilerini tutuyor.
- **Ürün:** Her ürünün adı, fiyatı, stok miktarı, kategori ve satıcı ilişkisi yer alıyor.
- **Sipariş:** Müşterilerin verdiği siparişleri kaydediyor.
- **Sipariş Detay:** Siparişlerdeki ürünleri, adetlerini ve fiyatlarını gösteriyor.

Tasarım Süreci

- Öncelikle tabloların birbirleriyle ilişkilerini **ER diyagramına** göre belirledim.
- Her tabloda birincil anahtar (ID) kullanarak verilerin eşsiz olmasını sağladım.
- İlişkileri `FOREIGN KEY` ile bağladım; örneğin `Urun.kategori_id → Kategori.id`.

Karşılaşılan Sorunlar

- **ID Çakışması / Tekrar Eden Ürünler:** Bazı ürünleri yanlışlıkla birden fazla kez eklemiştim. Bu durum, Sipariş Detay eklerken `FOREIGN KEY` hatasına yol açtı. Çözüm olarak tekrar eden ürünleri tespit edip, **Sipariş Detay'daki referansları güncelledim** ve fazlalıkları silerek tabloyu temizledim.
- **Sipariş Ekleme Hataları:** Başlangıçta Sipariş tablosuna ekleme yaparken bazı `musteri_id` değerleri mevcut olmadığından hata alıyordum. Bunu, müşteri tablosuna uygun ID'leri ekleyerek ve tarih değerlerini kontrol ederek çözdüm.
- **Stok Güncellemeleri:** Ürün stoklarını güncellerken, stok miktarının sıfırın altına düşmemesi için `CASE` yapısını kullandım.

Sonuç

- Tablolar artık birbirleriyle doğru şekilde ilişkilenmiş durumda.
- Veritabanı, sipariş ve ürün işlemlerini güvenli bir şekilde takip edebilecek hâle geldi.
- Raporlama ve analiz işlemleri için sorgular yazılabilir; örneğin müşteri bazlı toplam sipariş tutarı, en çok satılan ürünler gibi.