Stok Takip Veritabanı Notları

Veritabanı Tabloları

# 1.Personel:

1.PersonelId

2.Ad

3.Soyad

4.Görev

5.Tel No

6.E-posta

7.Şifre

8.Kayıt tarih(bu sisteme üye olduğu zaman otomatik olarak kaydeder)

**1. Personel**

* Giriş ve yetkilendirme için gereken tüm bilgiler var.
* ✅ KayıtTarihi olması çok iyi.
* 🔹 Şifreyi kriptolu saklayacak şekilde tasarlarsan mükemmel olur.

# 2.Gruplar:

1.GrupId:

2.GrupAdı:

\*\*Grupları ben otomatik belirlicem

**2. Gruplar**

* Otomatik tanımlanacağı için sade kalması doğru.
* ✅ Çok mantıklı.

# 3.StokKartı(Urun detayları):

1.StokKartıId

2.ÜrünAdı

3.StokKodu

4.GrupId

5.StokBirimi

6.MinStok

7.MaxStok

8.DepoAdresi

9.ResimYolu

10.Açıklama

11.KayıtTarihi

12.FirmaKodu(sence gerekli mi)

14.FirmaAdı

15.PersonelId

**3. StokKartı**

* Ürüne ait neredeyse tüm bilgiler mevcut.
* ✅ GrupId ile grup ilişkisi kurulmuş.
* ✅ MinStok, MaxStok — kritik kontrol için çok yerinde.
* ❓ FirmaKodu: Eğer firmaları ayrı tabloda tutmuyorsan, sadece FirmaAdı yeterli olabilir. Ama ileride firma bazlı raporlama planlıyorsan tut.
* ✅ ResimYolu ile görsel eklenebilir.

# 4.Stok Durumu:

1.DurumId

2.UrunId

3.DepoAdı

4.SerbesetMiktar

5.KaliteMiktar

6.BlokeMiktar

**4. StokDurumu**

* Bir ürünün hangi durumda ne kadar stokta olduğunu detaylı şekilde ayırmışsın (Serbest, Kalite, Bloke).
* ✅ Çok yerinde. Özellikle kalite ve bloke takibi ciddi sistemlerde olur.

# 5.Stok Hareketi(bunun neden olduğunu anlamadım)

1.HareketId

2.UrunId

3.PersonelId

4.ProjeId

5.Tip(Girdi)

6.Miktar

7.Tarih

8.Acıklama

**5. StokHareketi**

“Bunun neden olduğunu anlamadım” demişsin. Açıklayayım:

Bu tablo, **geçmişe yönelik tüm giriş-çıkış işlemlerini** tutar. Yani:

* Hangi üründen ne zaman ne kadar girmiş veya çıkmış?
* Kim tarafından? Hangi projeye ait?

Bu **log (geçmiş kayıt)** sistemidir.  
✅ Zorunlu ve çok kritik bir tablo.  
Ayrıca stok miktarını bu tabloya göre hesaplamak en doğrusudur (örneğin: toplam giriş - toplam çıkış = mevcut stok).

# 6.Satınalma

1.SiparışId

2.ÜrünId

3.SiparışTarihi

4.Miktar(Siparış edilen miktar mı)

5.CariAdı

6.GelenMiktar

7.BirimFiyat(1ürün döviz cinsinden fiyatı)

8.Kur(o günkü fiyatı)

9.Para birimi

10.Açıklama

**6. Satınalma**

* ✅ Çok detaylı ve yerinde.
* 🔹 Miktar: evet, sipariş edilen miktar.
* 🔹 GelenMiktar: eksik/eksik gelen siparişler kontrolü için süper.
* ✅ Kur, para birimi gibi alanlar profesyonel düşünülmüş.

# 7.Proje:

1.ProjeId

2.ProjeAdı

3.Başlangıç T

4.Bitiş T

5.PersonelId

6.Projeyi başlatan eleman(BUNU SİLDİK CÜNKÜ PROJEDE PERSONELID KULLANIM AMACI O ZATEN)

7.Açıklama

8.Durum

**7. Proje**

* Tüm temel bilgiler var.
* ✅ Projeyi başlatan, PersonelId, Durum gibi alanlar sayesinde kontrol sağlanabilir.
* 🔹 Durum: Aktif, Tamamlandı, İptal gibi enum/lookup tablosuyla eşleştirilebilir.

# 8.Proje Kullanılan ürünler

1.KullanılanurunId

2.Projeıd

3.ÜrünId

4.Miktar(kullanılan ürün adeti mi bence öyle)

5.Maliyet(ürün tutarı)

**8. ProjeKullanilanUrunler**

“Ayrı tablo olmasının nedeni ne?” demişsin. Şu nedenlerle ayrı olmalı:

* Bir projede birden fazla ürün olabilir.
* Aynı ürün birden fazla projede kullanılabilir.

Yani **çoktan çoğa (many-to-many)** ilişki. Bu ilişki **ayrı tabloda tutulur.**  
✅ Doğru karar.  
✅ Miktar + Maliyet alanları sayesinde raporlama kolaylaşır.

# 9.Rapor

1.RaporId

2.ProjeId

3.Başlangıc T

4.Bitiş T

5.PersonelId

6.Harcanan Tutar

7.Gelen Tutar

8.Kullanılan Maliyet

9.Proje Açıklama

**9. Rapor**

* Projeye ait özet veriler çok güzel düşünülmüş.
* ✅ Kullanılan Maliyet, Harcanan Tutar, Gelen Tutar gibi alanlar profesyonel düzeyde.
* 🔹 RaporId olabilir (şu an sadece "Rapor" yazmışsın).
* Bu tablo, “bitmiş projelerin özet raporu” gibi çalışır. Çok doğru.

# 🔗 Tablolar Arası İlişki Önerisi (Temel Bazda)

Personel (1) ─────┬─< StokHareket

├─< Proje

└─< Satınalma

Grup (1) ──< StokKartı

StokKartı (1) ──< StokDurumu

└─< StokHareket

└─< Satınalma

Proje (1) ──< ProjeKullanilanUrunler >── (1) StokKartı

Proje (1) ──< Rapor