Veritabanı Sistemleri, Güz 2020, İÜC, Bilgisayar Mühendisliği Bölümü, A. KURT
Ödev 4: Stored Procedures & Triggers.
Teslim: 3 Ocak 20202 Pazar 24:00

VERİTABANI
Kasiyer(tckimlik, isim, tel, eposta)
Tedarikci (vergiNo, isim, telefon, eposta, faturaSayisi, faturaToplamı, faturaOrtalaması)
Urun (barkod, ad, stokMiktari, urunToplamKar, satisFiyatTrendi)
fatura (faturaNo, tarih, toplam, vergiNo) // sipariş veya satınalmalar
faturaUrunleri (faturaNo, barkod, miktar, birim, birimFiyat)
fis (fisNo, tarihSaat, toplam, tckimlik) // satışlar
fisUrunleri (fisNo, barkod, miktar, birim, birimFiyat)

İSTENENLER

- Yukardaki veritabanını PostgreSQL'de CREATE TABLE ile oluşturunuz ve her tablonun içine aşağıdaki soruları yapacak kadar veriyi INSERT ediniz.
- Aşağıdaki trigger ve procedurleri yazınız, test ediniz. Testten önce ve sonra tablo kayıtlarının durumlarını sorgulayarak trigger/procedurlerin doğru çalıştığına emin olunuz.
- Her soru için trigger/procedure kaynak kodunu ve test verisini (yani test için kullanılan insert/update/delete komutunu, komuttan önceki ve sonraki kayıtların verilerini) paylaşınız.

SORULAR

1. *Fis* tablosundaki *toplam* alanını gerektiğinde güncelleyen trigger(lar)ı yazınız. **İpucu**: *fisUrunleri* tablosuna bir kayıt eklendiğinde, silindiğinde veya bir kayıttaki (barkod veya miktar veya birimFiyat alanları) guncellendiğinde çalışacak olan bu trigger fiş toplamı değerini fiş ürünlerindeki "miktar x birimFiyat" değerlerini toplayarak bulur.
2. *Urun* tablosundaki *stokMiktarı* alanını gerektiğinde güncelleyen trigger(lar)ı yazınız. **İpucu**: stokMiktarı bir ürünün tüm alımlardaki(fatura) miktarlar toplamından tüm satişlardaki miktarlar toplamının çıkarılmasıyla hesaplanabilir. Bu hesaplamadan birim alanı kullanılmaz.
3. *Urun* tablosundaki satisFiyatTrendi alanini gerektiğinde güncelleyen trigger(lar)ı yazınız. **İpucu**: satisFiyatTrendi alanı "artan" veya "azalan" olabilir. Bu değer ürünün tarihSaat'e göre son iki satışındaki (fisUrunlari) birimFiyat değerlerinin karşılaştırılmasıyla bulunur: Son satıştaki fiyat bir öncesinden büyükse trend "artan", değilse "azalan" olacaktır.
4. Fatura tablosundaki toplan alanını gerektiğinde güncelleyen trigger(lar)ı yazınız. İpucu: *faturaUrunleri* tablosuna bir kayıt eklendiğinde, silindiğinde veya bir kayıttaki (barkod veya miktar veya birimFiyat alanları) guncellendiğinde çalışacak olan bu trigger toplam değerini fatura ürünlerindeki "miktar x birimFiyat" değerlerini toplayarak bulur.
5. Tedarikçi tablosundaki faturaSayisi, faturaToplamı, faturaOrtalaması alanlarını gerektiğinde güncelleyen trigger(lar)ı yazınız. **İpucu**: fatura tablosuna kayıt eklenmesi, silinmesi veya bu tablodaki toplam değerinin güncellenmesi durumunda çalışacak bu trigger, soru 4'teki trigger'I dolaylı olarak kullanmalıdır. Çünkü faturaUrunleri tablosuna bir kayıt eklenmesi, silinmesi veya güncelleme durumunda tedarikci tablosundaki FaturaToplami ve faturaOrtalaması değerlerinin yeniden hesaplanması gerekecektir.
6. "… satislar(barkod, tarihSaat) returns table…" gibi tanımlanan barkodu verilen bir ürünün verilen bir tarihSaatten sonraki tüm satış kayıtlarını (fisUrunlari tablosundaki kayıtları) döndüren stored function'I yazınız ve bu fonksiyonu çalıştığını gösterebilmek için bir SQL komutunda kullanınız. İpucu: "… satislar(barkod, tarihSaat) returns table…"