

BTEP104 – Ders Oturumu

Konu: Normalleştirme

Normalleştirme, veritabanındaki tablolarda bulunan **veri tekrarını kaldırmak** ve bu sayede **veri bütünlüğünü sağlamak** için kullanılan bir tekniktir (prosedür).

- 1. Veri tekrar (Data Redundancy):** Bir tablo içerisinde aynı verinin farklı yerlerde tekrar edilmesi ile oluşan problem.

OğrenciDers

Ogrno	İsim	Soyisim	Ortalama	Derskodu	dersAdı	Kredisi	harfnotu
101	Ali	Veli	2.75	BTEP101	Bil. Giris	3	A
101	Ali	Veli	2.75	BTEP102	C	3	B
102	Ayşe	Can	3.25	BTEP101	Bil. Giris	3	C
...							
...							

2. Veri Bütünlüğü (Data Integrity):

- Veritabanındaki her tablodaki satırlar bir anahtar nitelik kullanılarak belirlenmelidir. Dolayısıyla, Her tablonun **Birincil Anahtarı** olmalıdır.
- Veritabanındaki tablolar arasındaki doğru bağlantıların bilinmesi gerekmektedir. Bu bağlantılar, **Yabancı Anahtar** dediğimiz niteliklerle gösterilir.

Örnek: Aşağıdaki, tabloda veri tekrarı bulunmaktadır ve normalleştirilmelidir. Her öğrenci birden fazla ders alabilir ve bir dersi de birden fazla öğrenci alabilir. Öğrenciler, aldıkları her dersten harf notu alırlar.

Ogrno	İsim	Soyisim	Ortalama	Derskodu	dersAdı	Kredisi	Dönem	harfnotu
101	Ali	Veli	2.75	BTEP101	Bil. Giris	3	Bahar2018	A
101	Ali	Veli	2.75	BTEP102	C	3	Bahar2018	F
102	Ayşe	Can	3.25	BTEP101	Bil. Giris	3		C
...								
101	Ali	Veli	2.75	BTEP102	C	3	Güz2018	B
...								

Adımlar:

- Tablonun verileri arasındaki bağlantılar bulunur. Bunlara, **Fonksiyonel Bağımlılıklar (İşlevsel Bağımlılık)** denir.

Fonksiyonel Bağımlılık: Bir tablonun sütunlarının birbirleri ile olan bağlantısına Fonksiyonel Bağımlılık (F.B.) denir.

Örnek: Yukarıda verilen tabloya göre Fonksiyonel Bağımlılıkları bulalım:

1. Ogrno->isim, soyisim, ortalama
2. Derskodu-> dersAdı, kredisi
3. Ogrno,DersKodu->dönem,harfnotu

Belirleyici (Determinant): FB'nin sol tarafındaki elemana/elemanlara verilen isimdir.

OgrenciDers(OgrNo, isim, soyisim, ortalama, dersKodu, dersAdı,kredisi, dönem,harfNotu)

Fonksiyonel Bağımlılıklar:

1. Ogrno->isim, soyisim, ortalama
2. Derskodu-> dersAdı, kredisi
3. Ogrno,DersKodu->dönem,harfnotu

Belirleyici İçeriği Hesaplama (Computing Closure): İçeriğin tablonun tüm sütunlarını içermesi durumunda ilgili Belirleyicinin Süper Anahtar olduğu söylenebilir.

1. {OgrNo}⁺={OgrNo, isim, soyisim, ortalama} X Süper Anahtar
2. {DersKodu}⁺= {Derskodu, dersAdı, kredisi} X Süper Anahtar
3. {OgrNo, DersKodu}⁺={OgrNo, DersKodu, isim, soyisim, ortalama, dersAdı, kredisi, dönem, harfnotu} ✓ Süper Anahtar

OgrenciDers(OgrNo, isim, soyisim, ortalama, dersKodu, dersAdı,kredisi, dönem,harfNotu)

Fonksiyonel Bağımlılıklar:

1. Ogrno->isim, soyisim, ortalama
2. Derskodu-> dersAdı, kredisi
3. Ogrno,DersKodu->dönem,harfnotu

OgrenciDers tablosunu Normalleştirin.

Normalleştirme Adımları (Prosedürü)

- Normal Olmayan Form (**UN**ormalized Form) – Veri tekrarı içeren tablonun olduğu durum (form)
- Birinci Normal Form (1NF)
- İkinci Normal Form (2NF)
- Üçüncü Normal Form (3NF)
- Boyce-Code Normal Form (BCNF)
- Dördüncü Normal Form (4NF)
- Beşinci Normal Form (5NF)

Bir tablonun BCNF Formuna kadar normalleştirilmiş olması yeterlidir.

Birinci Normal Form: Bir tablonun 1NF’da olabilmesi için veri tekrarı içermemesi / Birincil Anahtarının biliniyor olması gerekmektedir.

Önemli Not: Her tablonun (Birincil Anahtarının biliniyor olması koşulu ile) 1NF’da olduğunu varsayacağız.

İkinci Normal Form: Bir tablonun 2NF’da olabilmesi için;

1. Tablonun 1NF’da olması,
2. Kısmi Bağımlılık içermemesi

germektedir.

Kısmi Bağımlılık: Birincil anahtarın (BA) **bir parçasının**, anahtar olmayan bir niteliği belirlemesi durumuna Kısmi Bağımlılık (KB) deriz.

Aşağıdaki tabloda Kısmi Bağımlılık var mı bir göz atalım!!!

OgrenciDers(OgrNo, isim, soyisim, ortalama, dersKodu, dersAdı,kredisi, dönem,harfNotu)

Fonksiyonel Bağımlılıklar:

1. Ogrno->isim, soyisim, ortalama
2. Derskodu-> dersAdı, kredisi
3. Ogrno,DersKodu->dönem,harfnotu

ogrNo ve dersKodu (BA'nın parçaları) ayrı ayrı belirleyici konumunda olduklarından, 2NF'un kuralı olan Kısmi Bağımlılığa yol açıyorlar. Dolayısıyla, 2NF'u bozuyorlar.

OgrenciDers tablosunu 2NF'a çevirelim:

Adım 1: Kısmi Bağımlılığa yol açtıkları tespit edilen FB/FB'lar, ayrı ayrı tabloya dönüştürülür.

Ogrenci(Ogrno, isim, soyisim, ortalama)

1. OgrNo->isim, soyisim, ortalama

Ders(Derskodu, dersAdı, kredisi)

2. Derskodu-> dersAdı, kredisi

OgrenciDers(OgrNo, isim, soyisim, ortalama, dersKodu, dersAdı, kredisi, dönem, harfNotu)

Adım 2: Yukarıdaki adımda oluşturduğumun tablo/tablolara anahtar olmayan niteliklerini ANA tablodan (OgrenciDers) silip, kalanlardan yeni bir tablo oluşturuyoruz.

OgrenciDers(OgrNo, derskodu, dönem, harfnotu)

Nihayetinde, tekrar içeren Ogrenciders tablosunu normalleştirirken, uyguladığımız adımlar sayesinde, 3 tane daha küçük, ancak tekrar içermeyen, ayrıca Veri Bütünlüğünün Sağlandığı tablolar oluşturduk.

Ogrenci(Ogrno, isim, soyisim, ortalama)

Ders(Derskodu, dersAdı, kredisi)

OgrenciDers(OgrNo, derskodu, dönem, harfnotu)

OgrenciDers Tablosu 2NF'a kadar normalleştirilmiş oldu.

Üçüncü Normal Form: Bir tablonun 3NF'da olabilmesi için;

1. Tablonun 2NF'da olması,
2. Geçişli Bağımlılık (GB) içermemesi gerekmektedir.

Geçişli Bağımlılık (GB): Anahtar olmayan niteliğin, anahtar olmayan başka bir niteliği belirlemesi duruma denir.

Örnek:

R(a, b, c, d, e)

FB:

1. a,b-> c,d,e
2. c,d->e

R tablosunu 3NF'a kadar normalleştirin.

R tablosu 1NF'da mı? Evet. BA biliniyor.

R tablosu 2NF'da mı? Evet. Kısmi Bağımlılık yok.

R tablosu 3NF'da mı? Fonksiyonel Bağımlılık 2 – Geçişli Bağımlılığa yol açtığından, tablo 3NF'da değildir.

ÇÖZÜM:

ADIM 1:

R1(c, d, e)

ADIM 2:

R(a, b, c, d, e)

Yukarıda (adım1) oluşturduğumuz tablonun anahtar olmayan niteliğini ana tablomuzdan silip, kalanlardan yeni bir tablo oluşturuyoruz.

R2(a, b, c, d)

Sonuç:

R1(c, d, e)

R2(a, b, (c, d))

R1 ve R2 tabloları nihayetinde, 3NF'a kadar normalleştirilmiş oldu.