



BLM19303 – Veri Tabanı Yönetim Sistemleri

Proje

İsim Soyisim: Dilara Demirhan

Numara: 1721221002

Konu: Bakım Yönetim Sistemi

Tarih: 03.02.2022

Analiz

a. Varlık-Tablolar

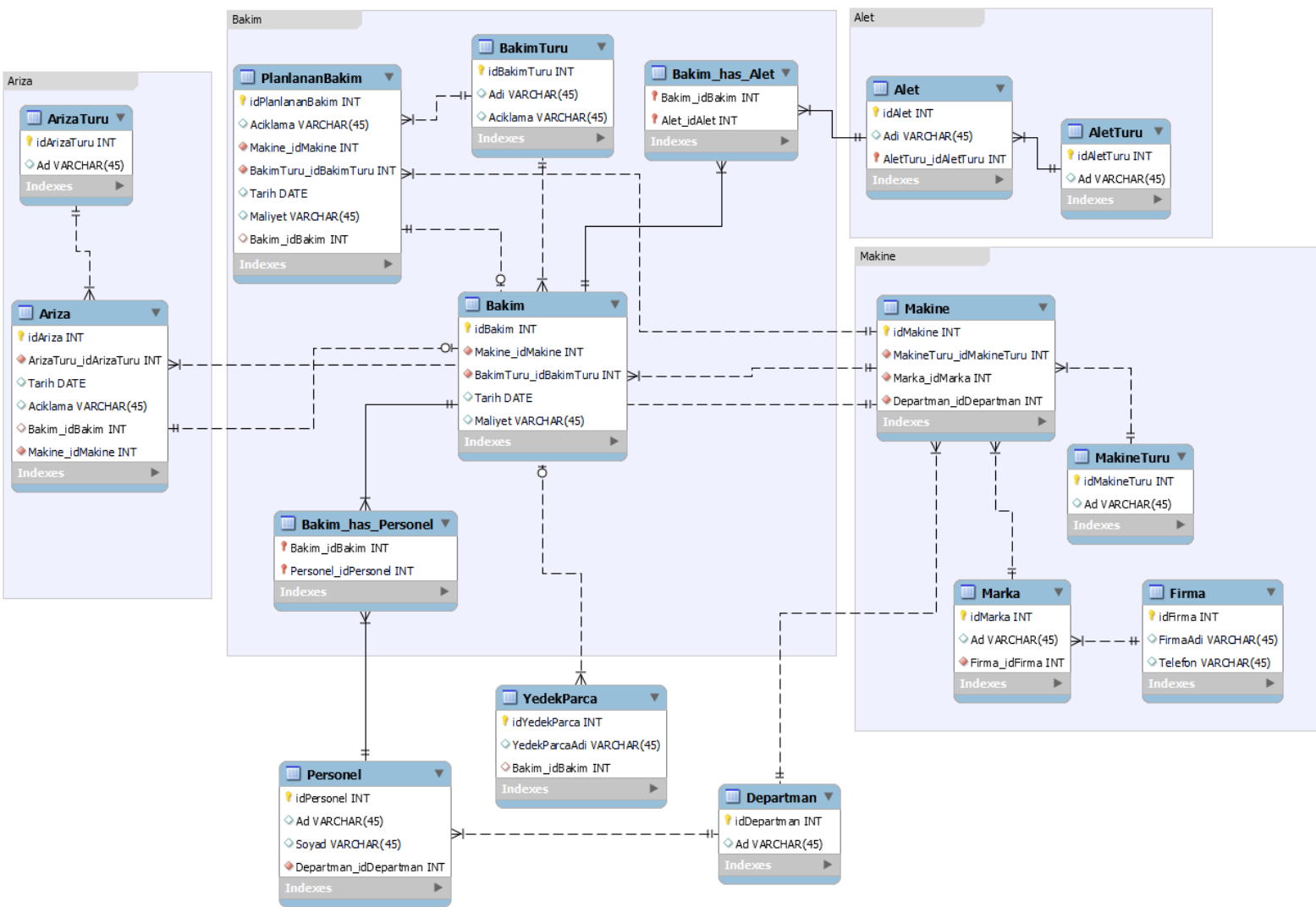
- **Makine Tablosu:** Bakımlar makinelere uygulanır. Makine tablosu, bakım uygulanacak makinelerin bilgilerini içerir.
- **Makine Türü Tablosu:** Programdaki makine türleri ve özelliklerini saklar.
- **Marka Tablosu:** Makinelerin markalarına ait bilgiler saklanır.
- **Firma Tablosu:** Markanın ait olduğu firmanın bilgilerini içerir.
- **Yedek Parça:** Bakımlarda yedek parçalar kullanılabilir. Yedek parçalara ait bilgiler yedek parça tablosunda saklanır.
- **Alet Tablosu:** Sahip olunan aletleri içeren tablodur.
- **Alet Türü Tablosu:** Programdaki alet türleri ve özelliklerini saklar.
- **Arıza Tablosu:** Makinelerde ortaya çıkan arızaları takip etmek için kullanılır.
- **Arıza Türü Tablosu:** Arıza türleri ve özelliklerini saklar.
- **Bakım Tablosu:** Yapılan bakımları takip etmek için kullanılır.
- **Planlanan Bakım Tablosu:** Yapılması planlanan bakımları ve özelliklerini içerir.
- **Bakım Türü Tablosu:** Bakım türleri ve bilgilerini saklar.
- **Bakım has Alet Tablosu:** Bakımlarda kullanılan aletleri takip etmek için kullanılır.
- **Bakım has Personel Tablosu:** Bakım işlemlerinde görev alan personelleri takip etmek için kullanılır.
- **Personel Tablosu:** Personelleri ve bilgilerini içerir.
- **Departman Tablosu:** Departman bilgilerini içerir.

b. Kurallar

- Bir bakım gerçekleştirmek için çok sayıda alet kullanılabilir. Bir alet farklı bakımlarda bakımı kullanılabilir.
- Bir bakımı gerçekleştirmek için çok sayıda yedek parça kullanılabilir.
- Bir yedek parça sadece bir bakımda kullanılabilir.
- Bir makineye çok kez bakım uygulanabilir.
- Bir bakım işleminde çok sayıda personel çalışabilir. Bir personel farklı bakımlarda çalışabilir.
- Bir arıza için bir bakım uygulanabilir.
- Bir planlanan bakım için bir bakım uygulanabilir.
- Aynı türde birçok makine olabilir. Bir makinenin bir türü olabilir.
- Aynı türde birçok bakım olabilir. Bir bakımın bir türü olabilir.
- Aynı türde birçok arıza olabilir. Bir arızanın bir türü olabilir.
- Aynı türde birçok alet olabilir. Bir aletin bir türü olabilir.
- Makinelerin bir markası vardır, markası aynı olan makineler olabilir.
- Bir firmanın birçok markası olabilir.
- Bir departmanda birçok kişi çalışabilir, bir kişi bir departmanda çalışabilir.

- Bir departmanın birçok makinesi olabilir. Bir makine bir departmana aittir.
- Arıza bakımı ve planlanan bakımlar dışında bakım yapılabilir.

Model



SQL ve İlişkisel Cebir Sorguları

- Üçten fazla bakım uygulanan ve bu bakımlarda hiç yedek parça kullanılmayan makineler:

```
SELECT *
FROM makine
WHERE idmakine IN
    (SELECT idmakine FROM
        (SELECT idmakine, Count(*)
        FROM bakım
        GROUP BY idmakine
        HAVING Count(*) > 3)
    INTERSECT
    (SELECT idmakine
    FROM bakım
    MINUS
    SELECT idmakine
    FROM bakım
    WHERE idbakim IN
        (SELECT idbakim
        FROM yedek_parca)))
```

$\text{makine} \bowtie (\pi \text{idmakine} (\sigma \text{adet} > 3 (\gamma \text{idmakine; count(*)} \rightarrow \text{adet} (\text{bakim}))))$

\cap

$\pi \text{idmakine} (\text{bakim}) - \pi \text{idmakine} (\text{bakim} \bowtie \text{yedek_parca})$

	IDMAKINE	MAKINEADI	IDMAKINETURU	IDMARKA	IDDEPARTMAN
1	5	Makine5	1	132	423

- Planlanan bakımlardan zamanında yapılmayanların toplam planlanan maliyeti:

```
SELECT SUM(maliyet)
FROM planlananbakim
WHERE idplanlananbakim IN (SELECT idplanlananbakim
                           FROM bakım b
                           JOIN planlananbakim p
                           ON p.idbakim = b.idbakim
                           WHERE b.tarih > p.tarih
                           UNION
                           SELECT idplanlananbakim
                           FROM planlananbakim
                           WHERE tarih < sysdate AND idbakim IS NULL)
```

y; sum(maliyet)→toplam

(planlananbakim ⋈ (π idplanlananbakim (σ bakım.tarih > planlananbakim.tarih (σ planlananbakim.idbakim = bakım.idbakim (planlananbakim ⋈ bakım))))

∪

π idplanlananbakim (σ tarih < date('2022-02-03') ∧ idbakim is null (idplanlananbakim))))

	SUM(MALİYET)
1	166200

- Maliyeti kendinden önceki bakımların maliyet ortalamasından yüksek olan bakımlar:

```
SELECT *
FROM bakım
WHERE idbakim IN (SELECT b.idbakim
                  FROM bakım b, bakım b2
                  WHERE b.tarih > b2.tarih
                  GROUP BY b.idbakim, b.maliyet
                  HAVING (AVG(b2.maliyet)) < b.maliyet)
```

bakim ✖

```
(π idbakim (σ b1.maliyet > ortalama (γ b1.idbakim, b1.maliyet;  
avg(b2.maliyet)→ortalama (σ b1.tarih > b2.tarih (ρ b1 (bakim) × ρ  
b2 (bakim))))))
```

	IDBAKIM	IDMAKINE	IDBAKIMTURU	TARİH	MALİYET
1	122	387	21	06/01/2022	79000
2	176	12	242	10/01/2022	86000
3	200	387	242	10/01/2022	44000
4	185	12	431	25/11/2015	90000
5	21	12	21	30/01/2022	70000

PLSQL Paket ve Fonksiyonlar

- Paket

```
CREATE OR REPLACE PACKAGE BAKIM_PACK AS
```

```
--Bir bakım türünde yapılan bakım sayısı
```

```
FUNCTION get_bakim_adet(p_bakimturu VARCHAR2) RETURN PLS_INTEGER;
```

```
--Yıllara göre bakım sayılarını yazdıran procedure
```

```
PROCEDURE yil_bakim;
```

```
--Planlanan bakımın gerçek maliyeti ile planlanan maliyeti arasındaki fark
```

```
FUNCTION get_fark(p_idplanlanan PLS_INTEGER) RETURN PLS_INTEGER;
```

```
END BAKIM_PACK;
```

- Bir bakım türünde yapılan bakım sayısı

```
FUNCTION get_bakim_adet(p_bakimturu VARCHAR2) RETURN PLS_INTEGER AS  
CURSOR c_bakim IS  
    SELECT * FROM bakım WHERE idbakimturu IN  
        (SELECT idbakimturu FROM bakimturu WHERE adi = p_bakimturu);  
p_adet PLS_INTEGER := 0;  
p_bakim bakım%ROWTYPE;  
BEGIN  
    FOR p_bakim IN c_bakim LOOP  
        p_adet := p_adet + 1;  
    END LOOP;  
RETURN p_adet;  
END get_bakim_adet;
```

	BAKIM.BAKIM_PACK.GET_BAKIM_ADET('PERİYODIKBAKIM')
1	3

- Yıllara göre bakım sayılarını yazdıran procedure

```
PROCEDURE yil_bakim AS
    adet pls_integer;
    min_yil pls_integer;
    max_yil pls_integer;
    BEGIN
        SELECT MIN(EXTRACT(YEAR FROM tarih)) INTO min_yil FROM bakim;
        SELECT MAX(EXTRACT(YEAR FROM tarih)) INTO max_yil FROM bakim;
        DBMS_OUTPUT.PUT_LINE('Yil          || 'Bakim Sayisi');
        DBMS_OUTPUT.PUT_LINE('-----');

        FOR yil IN min_yil .. max_yil LOOP
            SELECT COUNT(*)
            INTO   adet
            FROM   bakim
            WHERE  EXTRACT( YEAR FROM tarih ) = yil;
            IF adet > 0 THEN
                DBMS_OUTPUT.PUT_LINE(yil || '          '||adet);
            END IF;
        END LOOP;

    END yil_bakim;
```

Yil	Bakim Sayisi

2008	2
2015	2
2020	3
2021	5
2022	6

- Planlanan bakımın gerçek maliyeti ile planlanan maliyeti arasındaki fark

```

FUNCTION get_fark(p_idplanlanan PLS_INTEGER) RETURN PLS_INTEGER AS
    CURSOR c_planlanan IS
        SELECT * FROM planlananbakim WHERE idplanlananbakim = p_idplanlanan;
    CURSOR c_bakim IS
        SELECT * FROM bakim;
    p_planlanan planlananbakim%ROWTYPE;
    p_bakim bakim%ROWTYPE;
    p_fark PLS_INTEGER;
BEGIN
    OPEN c_planlanan;
    FETCH c_planlanan INTO p_planlanan;
    OPEN c_bakim;
    LOOP
        FETCH c_bakim INTO p_bakim;
        EXIT WHEN c_bakim%NOTFOUND;
        IF p_bakim.idbakim = p_planlanan.idbakim THEN
            p_fark := p_bakim.maliyet - p_planlanan.maliyet;
            p_fark := ABS(p_fark);
        END IF;
    END LOOP;
    CLOSE c_bakim;
    CLOSE c_planlanan;
    RETURN p_fark;
END get_fark;

```

	BAKIM.BAKIM_PACK.GET_FARK(4)
1	2000

PLSQL Tablo Trigger

Arıza için yapılan bakımın tarihi arıza tarihinden önceki bir tarihsen hata vardır ve update yapılamaz.

```

create or replace TRIGGER trg_ariza_tarih_kontrol
BEFORE UPDATE OF idbakim ON ariza
FOR EACH ROW
DECLARE
    p_tarih_ariza DATE;
    p_tarih_bakim DATE;
BEGIN
    p_tarih_ariza := :old.tarih;
    SELECT tarih INTO p_tarih_bakim FROM bakim WHERE idbakim = :new.idbakim;

    IF p_tarih_ariza > p_tarih_bakim THEN
        RAISE_APPLICATION_ERROR(-20555, 'Bakimin tarihi ariza tarihinden once olamaz');
    END IF;
END;

```



```
Error starting at line : 67 in command -
UPDATE ariza
SET idbakim = 20
WHERE idariza = 3
Error report -
ORA-20555: Bakimin tarihi ariza tarihinden once olamaz
ORA-06512: konum "BAKIM.TRG_ARIZA_TARİH_KONTROL",  satır 9
ORA-04088: 'BAKIM.TRG_ARIZA_TARİH_KONTROL' tetikleyicisinin yürütülmesi sırasında hata
```

Normalleştirme Analizi

Bakim	
idBakim	INT
BakimTuru	VARCHAR(45)
idMakine	INT
makineAdi	VARCHAR(45)
idPersonel	INT
personelAdi	VARCHAR(45)

Birinci Normal Form

Bir tablonun birinci normal formda olması için bir kayıttaki tüm alanlar bir tek manaya sahip veri içermelidir. Bu tabloyu birinci normal forma uydurmak için personel adı ve soyadını aynı alanda tutmak yerine personel soyadı için yeni kolon oluşturabilir.

Bakim	
idBakim	INT
BakimTuru	VARCHAR(45)
idMakine	INT
makineAdi	VARCHAR(45)
idPersonel	INT
personelAdi	VARCHAR(45)
personelSoyadi	VARCHAR(45)

İkinci Normal Form

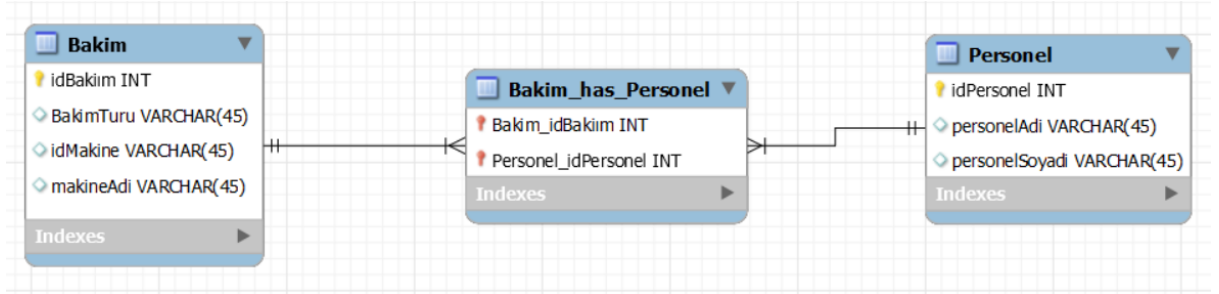
Tablonun ikinci normal forma uygun olması için aday anahtar harici tüm alanlar tüm aday anahtarla tam fonksiyonel bağımlı olmalıdır. Bu tablonun aday anahtarı idBakim, idPersonel olmasına rağmen personelAdi, personelSoyadi alanlarını idPersonel'in belirleyebilmesi ikinci normal formu bozar.

R1 { **idPersonel**, personelAdi, personelSoyadi }

R2 { **idBakim, idPersonel**, BakimTuru, idMakine, makineAdi, personelAdi, personelSoyadi }

R2.1 { **idBakim**, BakimTuru, idMakine, makineAdi }

R2.2 { **idPersonel, idBakim** }



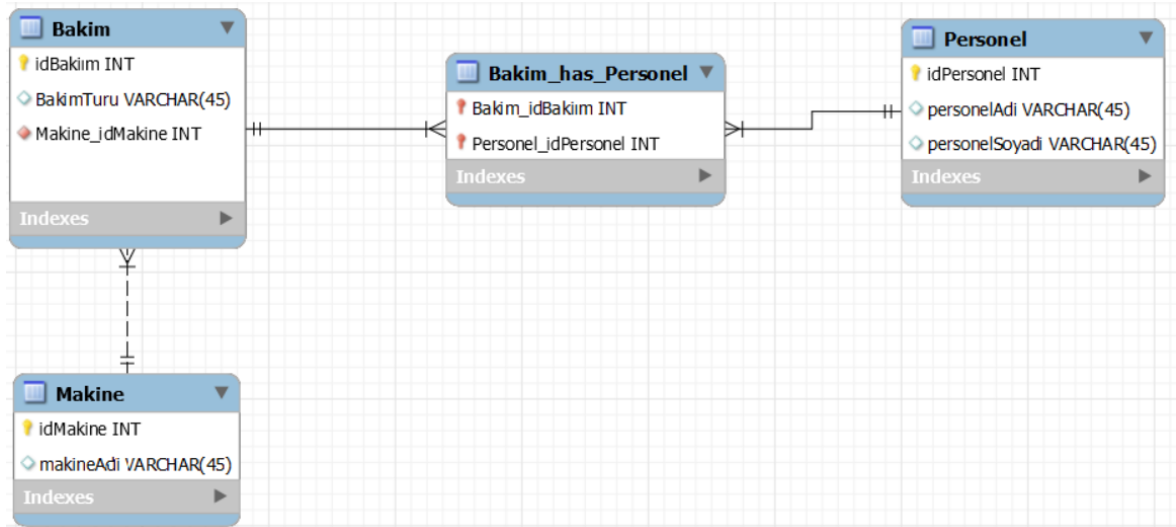
Üçüncü Normal Form

Tablonun üçüncü normal forma uygun olması için birincil anahtar haricindeki alanlar arasında hiç bir fonksiyonel bağımlılık olmamalıdır. Burada idMakine'nin makineAdi'ni belirleyebilmesi üçüncü normal formu bozar.

R2.1 { **idBakim**, BakimTuru, idMakine, makineAdi }

R2.1.1 { **idMakine**, makineAdi }

R2.1.2 { **idBakim**, BakimTuru, idMakine }



Boyce Codd Normal Form

Tablolar üçüncü normal formda ve tüm belirleyiciler aday anahtar olduğu için Boyce-Codd normal formu bozan bir durum yoktur.

Denormalizasyon

Arıza bakımının maliyeti sık sorgulanan bir bilgi olduğu için arıza tablosuna maliyet kolonu eklenebilir böylece çok fazla join işlemi yapılmamış olur.

```
ALTER TABLE ariza
ADD maliyet NUMBER(10);

UPDATE ariza a
SET maliyet = (SELECT maliyet FROM bakim WHERE idbakim = a.idbakim)
```