

PROGRAMLAMA LABORATUVARI 2

3. PROJE

Bilgisayar Mühendisliği Bölümü Kocaeli Üniversitesi

Sinem ERCÜMERT 200202016@kocaeli.edu.tr
İlkay Mehmet BORA 200202030@kocaeli.edu.tr

I. ÖZET

Bu doküman Programlama Laboratuvarı 2 dersi 3. projesi için çözümümüzü açıklamaya yönelik oluşturulmuştur. Dokümanda projenin özet, giriş, yöntem, deneysel sonuçlar, sonuç ve ER diyagramı kısımlarına yer verilmiştir. Doküman sonunda projemizi hazırlarken kullandığımız kaynaklar bulunmaktadır.

II. GİRİŞ

Bu projenin amacı, bir bankanın yönetim sistemi için bir veritabanı tasarlanması ve bu veritabanının üzerinde gerekli işlemleri gerçekleyen bir uygulama geliştirilmesidir. Tasarlanan veritabanı, bankanın müşterilere, çalışanlara, hesaplara ve işlemlere ilişkin bilgileri organize bir şekilde işlemesine yardımcı olacaktır. Bu şekilde bankanın ve müşterilerin ihtiyacı olan bilgilere daha kolay ulaşabilmesi sağlanmış olacaktır. Ayrıca, veritabanı kullanarak müşteri istekleri ve bankanın ihtiyaçları doğrultusunda raporlar hazırlanabilecektir. Banka içerisinde müşteri, temsilci ve banka müdürü olmak üzere 3 adet rol bulunmaktadır. Müşteriler ve çalışanlar için gerekli tanımlayıcı bilgiler (Ad Soyad, Telefon, TC No, Adres, E-posta) veri tabanında saklanmaktadır. Bir müşterinin birden fazla hesabı bulunabilir. Hesaplar sistem içerisinde kayıtlı bulunan herhangi bir para birimi cinsinden açılabilir (TL varsayılan olarak gelmektedir). Hesaplar arası para transferinde gerekli durumlarda kur dönüşümü otomatik olarak yapılmaktadır. Tüm bu eylemlerin tasarlanan bir arayüz üzerinden görsel bir şekilde gösterilmesi gerekmektedir.

III. YÖNTEM

A. Geliştirme Ortamı

Projenin gelişimini takip edebilmek için Windows ve MacOS sisteminde MySQL workbench ve Visual Studio kullanılmıştır. Proje C # ve SQL programlama dili ile yazılmıştır.

B. Yazılım Tasarımı

Class'lar:

musteri → Müşterinin yapabileceği roller gerçekleştirilir.
musteri-temsilcisi → Müşteri temsilcisinin yapabileceği roller gerçekleştirilir.
mudur → Müdürün yapabileceği roller gerçekleştirilir.
kayıt-formu → Kayıtlar yapılırken kullanılan ekranın form kısmı gerçekleştirilir.
giris → Giriş ekranını burada düzenlenir.
para-islem → Kredi, havale, para çekme, para yatırma gibi işlemler burada gerçekleştirilir.

C. Yalancı Kod

musteri

Kullanılacak kütüphaneler tanımlanır.
C # ve SQL bağlantısı için gerekli kütüphane kurulur.

public partial class musteriler : Form
1-MySQL için gerekli fonksiyonlar yazılır.
2-Form açmak için gerekli fonksiyon yazılır.

void MusteriGetir()

1-Mysql veri tabanında oluşturulan musteriler veri tabanı çekilir ve okunur.

2-while (oku.Read())

1- Giriş ekranına müşterinin girdiği gerekli bilgiler string olarak çekilir.

void HesapGetir()

1-Müşterilerin açtığı hesaplar ekrana getirilir.

void ToplamPara()

1-Müşterinin hesaplarındaki toplam para ekrana yazdırılır.

private void Para-Cek-Click(object sender, EventArgs e)

1-Müşteri istediği hesabından istediği miktarda para çeker.

private void Bilgi-Goruntule-Click(object sender, EventArgs e)

1-Müşteri kendi girdiği bilgileri güncelleyebilir.

private void musteriguncelle-Click(object sender, EventArgs e)

1-Müşteri kendi girdiği bilgileri güncelleyebilir.

2-“Başarıyla Güncellendi” mesajı ekrana yazdırılır.

private void hesap-sil-Click(object sender, EventArgs e)

1-Müşteri silmek istediği hesabı silmek için istek gönderir.

2-“Hesabınız Başarıyla silindi” ve “Bizi tercih ettiğiniz için teşekkür ederiz.” mesajları ekrana yazdırılır.

private void dataGridView1-CellEnter(object sender, DataGridViewCellEventArgs e)

1-Müşterinin ekranda text butonlarına yazdığı datalar ekrandaki veri tabanı kısmına girilir

private void hesap-olustur-button-Click(object sender, EventArgs e)

1-Müşteri hesap açma talebini müşteri temsilcisine gönderir.

2- Ekranda "Hesap Açma Talebiniz Gönderildi." mesajı yazdırılır.

private void kredi-basvuru-button-Click(object sender, EventArgs e)

1-Müşteri krediye başvurmak istediği hesabı ve miktarı girer.

2-Talep müşteri temsilcisine gönderilir.

3-"Kredi Talebiniz Gönderildi." mesajı ekrana yazdırılır.

private void havale-button-Click(object sender, EventArgs e)

1-Müşteri havale yap butonuna bastığında karşısına çıkan ekrandan havale yapmak istediği hesabı girer.

private void para-yukle-button-Click(object sender, EventArgs e)

1-Müşteri para yükle butonuna bastığında karşısına çıkan ekrandan yükleme yapmak istediği hesabı girer.

private void para-cek-button-Click(object sender, EventArgs e)

1-Müşteri para çek butonuna bastığında karşısına çıkan ekrandan çekmek istediği hesabı girer.

private void kredi-odeme-button-Click(object sender, EventArgs e)

1-Müşteri kredi öde butonuna bastığında karşısına çıkan ekrandan ödemek istediği hesabı girer.

para-islemleri

Kullanılacak kütüphaneler tanımlanır.

C # ve SQL bağlantısı için gerekli kütüphane kurulur.

public para-islem(int value,int hesap-no)

1-Müşteri para yüklemek istediğinde ekrana gelen miktar kutusuna giriş yapar.

2-Müşteri para çekmek istediğinde ekrana gelen kutuya giriş yapar.

3-Müşteri kredi ödemek istediğinde ekrana gelen miktar kutusuna giriş yapar.

4-Müşteri havale istediğinde ekrana gelen miktar ve hesap no kutusuna giriş yapar.

5-Müşteri kredi çekmek istediğinde ekrana gelen miktar kutusuna giriş yapar.

void bakiye()

1-Müşteri ekranında görmek istediği hesap no'yu girdiği zaman veri tabanından çekilen hesabın bakiyesini görür.

void kredi()

1-Müşteri ekranında kredisini ödemek istediği hesap no'yu girdiği zaman veri tabanından çekilen kredinin miktarını görür.

private void onay-button-Click(object sender, EventArgs e)

1-Hesaba para yüklenmek istendiğinde ve yeterli para hesapta var ise: 1-Müşteri para yüklemek istediğinde bakiyeye girilen miktar eklenir ve yeni miktar veri tabanına yansıtılır.

2-“Başarıyla Onaylandı.” mesajı ekrana yansıtılır.

2-Hesaptan para çekilmek istendiğinde ve yeterli para hesapta var ise:

1-Müşterinin çekmek istediği para miktarı

bakiyeden azaltılır ve yeni miktar veri tabanına yansıtılır.

2-"Başarıyla Onaylandı." mesajı ekrana yansıtılır.

3-Kredi borcu ödenmek istendiğinde ve girilen miktar bakiyeden büyük ise:

1-"Bakiyenizde Yeterli Para Yoktur!" mesajı ekrana yansıtılır.

4-Kredi borcu ödenmek istendiğinde ve girilen miktar krediden büyük ise:

1-"Girdiğiniz Miktar Kredi Borcunuzdan Fazla!" mesajı ekrana yansıtılır.

5-Kredi borcu ödenmek istendiğinde ve girilen miktar kredi miktarına eşit ise:

1-"Kredi Ödeme İşleminiz başarıyla onaylandı" mesajı ekrana yansıtılır.

6-Müşteri havale yapmak istediğinde ekrana gelen text butonlarına gerekli bilgileri girer ve sql tabanından gerekli bilgiler çekilerek kontrol edilir.

7-Alınan bilgiler müşteri temsilcisinin onayına gönderilir.

8-"Havale Talebiniz Gönderildi." mesajı ekrana yansıtılır.

9-Müşteri kredi çekmek istediğinde ekrana gelen text butonlarına gerekli bilgileri girer ve sql tabanından gerekli bilgiler çekilerek kontrol edilir.

10-Alınan bilgiler müşteri temsilcisinin onayına gönderilir.

11-"Kredi Talebiniz Gönderildi." mesajı ekrana yansıtılır.

giris

Kullanılacak kütüphaneler tanımlanır.

C # ve SQL bağlantısı için gerekli kütüphane kurulur.

```
private void textBox1_TextChanged(object sender, EventArgs e)
```

1-Giriş ekranında ki butonlar da sadece rakam kullanılması için kullanılır.

```
bool giris-dogrulama-musteri(string id, string sifre)
```

1-Müşteriden id ve şifre istenir.

2-Alınan veriler veri tabanında ki değerlerle eşit mi diye kontrol edilir.

3-Okunacak satır varsa true yoksa false döndürülür.

```
bool giris-dogrulama-temsilci(string id, string sifre)
```

1-Müşteri temsilcisinden id ve şifre istenir.

2-Alınan veriler veri tabanında ki değerlerle eşit mi diye kontrol edilir.

3-Okunacak satır varsa true yoksa false döndürülür.

```
bool giris-dogrulama-mudur(string id, string sifre)
```

1-Veri tabanı metodu bağlantısı çağırılır.

2-Veri tabanına gönderilecek sorguyu tutabilmesi için nesne oluşturulur.

3-Nesnenin içine sorgu yazılır.

4-textbox'lara girilen değerler parametrelere aktarılır.

5-Parametre kullanarak injection'a karşı önlem alınır.

6-Komut veri tabanına gönderilir.

7-Yollanan komutlar dönen değerleri satır satır okur.

```
8-Okunacak satır varsa true yoksa false döndürülür.
private void giris-buton-Click(object sender, EventArgs e)
```

1-Textbox a girilen değerler string değişkenlere atanır.

2-Eğer müşteri textbox'larından biri boşsa "Lütfen tüm alanları doldurun" mesajı yayınlanır.

3-Eğer müşteri textbox'larından biri yanlış girilmişse "Hatalı Kullanıcı Adı veya Şifre!" mesajı yayınlanır.

4-Eğer müşteri temsilcisi textbox'larından biri boşsa "Lütfen tüm alanları doldurun" mesajı yayınlanır.

5-Eğer müşteri temsilcisi textbox'larından biri yanlış girilmişse "Hatalı Kullanıcı Adı veya Şifre!" mesajı yayınlanır.

6-Eğer müdür textbox'larından biri boşsa "Lütfen tüm alanları doldurun" mesajı yayınlanır.

7-Eğer müdür textbox'larından biri yanlış girilmişse "Hatalı Kullanıcı Adı veya Şifre!" mesajı yayınlanır.

```
private void sifre-goster-CheckedChanged(object sender, EventArgs e)
```

1-Şifre göster butonuna basıldığında şifre gösterilir.

```
private void Kayit-buton-Click(object sender, EventArgs e)
```

1-Kayıt butonuna basıldığında kayıt yapılır.

```
private void Giris-FormClosed(object sender, FormClosedEventArgs e)
```

1-Kapatma tuşuna basıldığında ekran kapatılır.

kayıt-formu

Kullanılacak kütüphaneler tanımlanır.

C # ve SQL bağlantısı için gerekli kütüphane kurulur.

```
private void back-button-Click(object sender, EventArgs e)
```

1-Geri git butonuna tıklanınca girişi gösterip kalanı görünmez yapar.

```
private void onay-buton-Click(object sender, EventArgs e)
```

1-textbox'tan alınan değerleri string değişkenlerine atanır.

2-Eğer textbox'lardan biri boşsa uyarı mesajı yayınlar.

3-Şifre doğru değilse "Girdiğiniz şifre eşleşmiyor!" mesajı ekrana yayınlanır.

4-Şifre ve id doğruysa başarılı mesajı yazdırılır.

5-Alınan bilgiler sql'e gönderilir ve kaydedilir.

musteri-temsilcisi

Kullanılacak kütüphaneler tanımlanır.

C # ve SQL bağlantısı için gerekli kütüphane kurulur.

```
private void musteri-bilgi-Click(object sender, EventArgs e)
```

1-Kullanıcı müşteri bilgisi kutusuna tıkladığında kenarına kahverengi bir box getirir.

2-Diğer kutuları görünmez kılar.

```
private void taleple-Click(object sender, EventArgs e)
```

1-Kullanıcı talepler kutusuna tıkladığında kenarına kahverengi bir box getirir.

2-Diğer kutuları görünmez kılar.

void MusteriGetir()

1-Müşteri görüntüle ekranında müşteri temsilcisine atanan müşterilerinin bilgileri veri tabanından çekilerek ekranda görüntülenir.

void TalepGetir()

1-Talepleri görüntüle ekranında müşteri temsilcisine atanan müşterilerinin talepleri veri tabanından çekilerek ekranda görüntülenir.

void HesapGetir()

1-Hesap görüntüle ekranında müşteri temsilcisine atanan müşterilerinin hesapları veri tabanından çekilerek ekranda görüntülenir.

```
private void dataGridView1-CellEnter(object sender, DataGridViewCellEventArgs e)
```

1-Müşteri görüntüle ekranındaki textbox'lara girilen değerleri string değerine dönüştürür.

```
private void musteri-ekle-button-Click(object sender, EventArgs e)
```

1-Müşteri ekle butonuna dokununca butonu görünür kılar.

```
private void musteri-guncelle-button-Click(object sender, EventArgs e)
```

1-Güncelle butonuna basıldığında yeni değerler eski değerlerine geçer.

```
void bakiye(int hesap-no)
```

1-Veri tabanından müşterilerin bakiyeleri okunur.

2-Bakiyeler ekrana yazdırılır.

```
void kredi(int hesap-no)
```

1-Veri tabanından müşterilerin bakiyeleri okunur.

2-Bakiyeler ekrana yazdırılır.

```
void talep-bilgi()
```

1-Müşterilerden gelen talep bilgileri ekrana yazdırılır.

```
private void talep-onay-Click(object sender, EventArgs e)
```

1-talep-bilgi ekranından gelen talepler onaylanırsa eğer eski değerlerin yerine yeni değerler yazdırılır.

2-Kredi başvurusu onaylanırsa eğer eski değerlerin yerine yeni değerler yazdırılır.

3-Havale başvurusu onaylanırsa eğer eski değerlerin yerine yeni değerler yazdırılır.

4-Hesap açma başvurusu onaylanırsa eğer eski değerlerin yerine yeni değerler yazdırılır.

5-Değiştirilen bilgiler sql veri tabanında güncellenir.

```
private void talep-ret-Click(object sender, EventArgs e)
```

1-talep-bilgi ekranından gelen talepler reddedilirse eğerdeğişiklik yapılmaz.

```
private void hesap-button-Click-1(object sender, EventArgs e)
```

1-Hesap getir butonuna kutusuna tıkladığında kenarına kahverengi bir box getirir.

```
private void hesap-sil-button-Click(object sender, EventArgs e)
```

1-Hesap sil butonuna kutusuna tıkladığında kenarına kahverengi bir box getirir.

IV. DENEYSEL SONUÇLAR

Yazdığımız tüm işlemler sonucunda müşteriler:

-Kayıt yapabilir,

-Bilgilerini güncelleyebilir,

-Para çekip yatırabilir,

-Kredi çekip kredi ödemesi yapabilir.

Müşteri temsilcileri:

-Müşteri ekleyip bilgilerini güncelleyebilir ve silebilir,

-İlgilendikleri müşterilerin genel durumlarını ve işlemlerini (gelir, gider ve toplam bakiye) görüntüleyebilir,

-Müşterilerden gelen hesap açma, silme ve kredi

taleplerini onaylayıp reddedebilir.

Müdür:

- Bankanın genel durumunu (gelir, gider, kar ve toplam bakiye) görüntüleyebilir,
- Çalışanların maaş ücretlerini belirleyebilir,
- Müşteri ekleyebilir,
- Bankada gerçekleşen tüm işlemleri (para çekme, yatırma ve transfer) görüntüleyebilir,
- Kredi ve gecikme faiz oranını belirleyebilir.

V. SONUÇ

Yaptığımız projenin bize kazandırdıkları;

- 1- SQL ve C # dilini efektif kullanmayı,
- 2- MySQL ve Visual Studio ide'sini kullanmayı,
- 3- Veri tabanı oluşturmayı ve ilişkilendirmeyi,
- 4- Arayüz tasarlamayı,
- 5- Normalizasyon yöntemlerini ve uygulanmasını,
- 6- ER diyagramı oluşturmayı,
- 4- Bir problemle karşılaştığımızda çözüm için nasıl yollar arayacağımızı,
- 5- Algoritma mantığını kurma ve uygulamada kendimizi geliştirdik ve öğrendik.

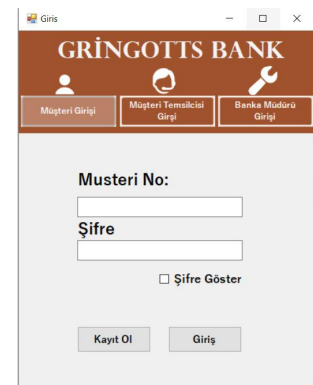
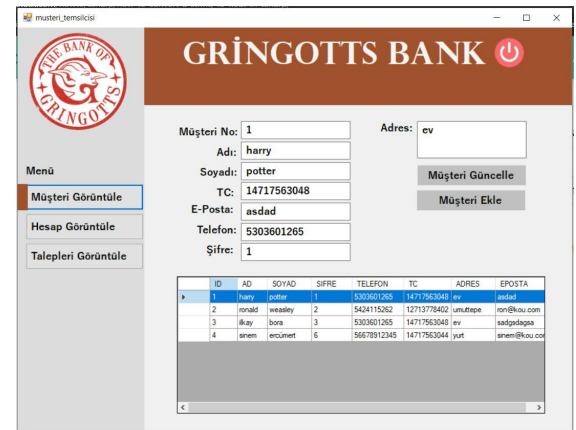
VI. KAYNAKÇA

1- Araştırma için:

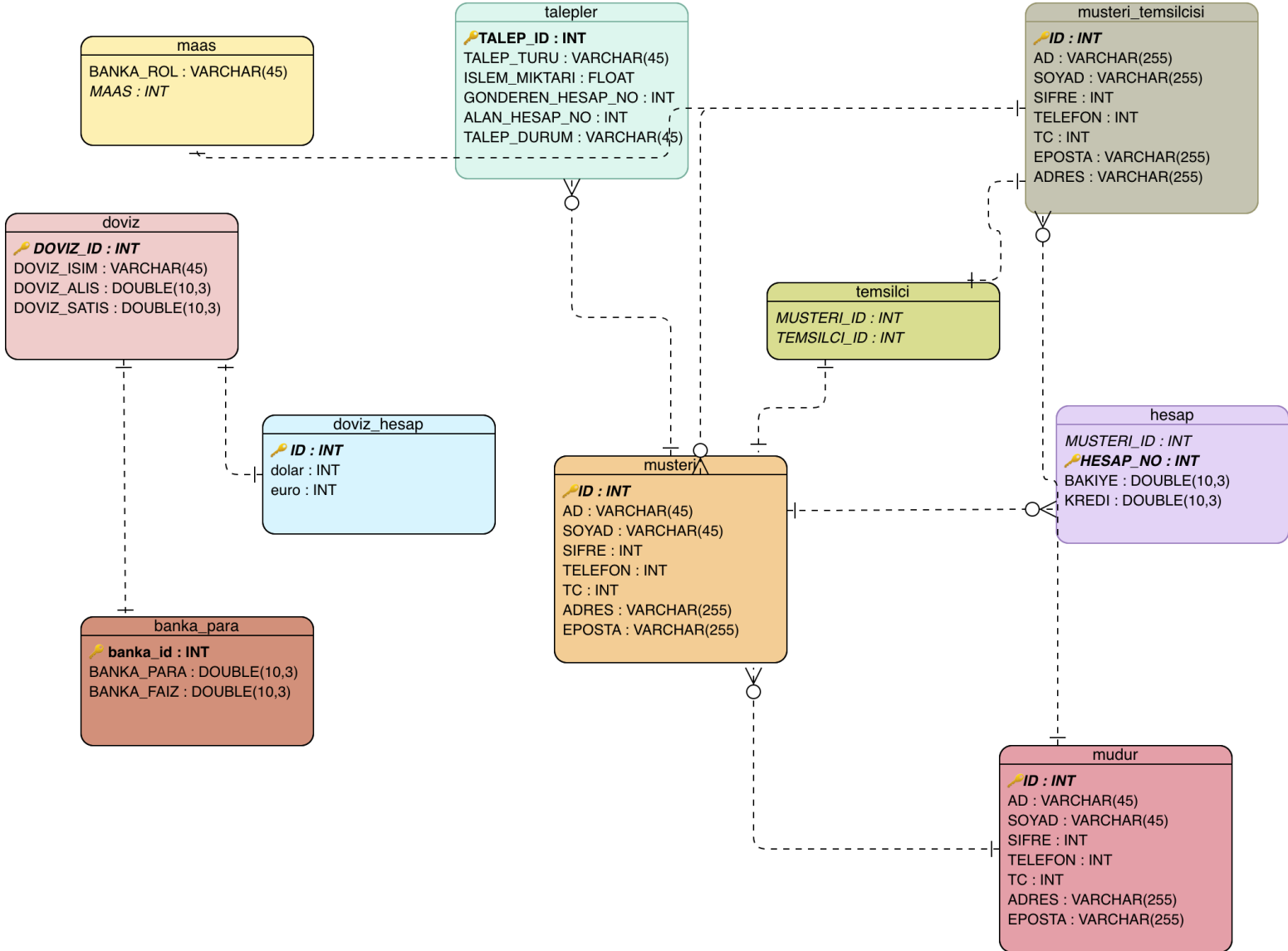
- <https://www.essentialsql.com/database-normalization/>
- <https://blog.ineuron.ai/Build-a-Bank-Database-using-SQL-Uc6hHUFt8N>
- <https://www.yazilimkodlama.com/programlama/c-veritabani-musteri-bakiye-ve-tahsilat-islemleri/>
- <https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/>
- <https://www.guru99.com/database-normalization.html>
- <https://bilgisayarkavramlari.com/2009/03/05/ucuncu-normal-sekil-third-normal-form-3nf/>
- <https://www.tasarimkodlama.com/veritabani/mysql/mysql-foreign-key-kullanimi/>

3- Karşılaştığımız çeşitli sorunlar için:

- <https://www.geeksforgeeks.org/>
- <https://stackoverflow.com/>
- <https://www.quora.com/>



ER DİYAGRAMI



NORMALİZASYON ADIMLARI

| Column | Datatype | PK | NN | UQ | B... | UN | ZF | AI | G | Default / Expression |
|---------------|--------------|----|-------------------------------------|-------------------------------------|------|----|----|----|---|----------------------|
| AD | VARCHAR(255) | | | | | | | | | |
| SOYAD | VARCHAR(255) | | | | | | | | | |
| KIMLIK_NO | BIGINT(11) | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | | | | | | |
| TELEFON_NO | BIGINT(11) | | | | | | | | | |
| ADRES | TEXT | | | | | | | | | NULL |
| E_POSTA | VARCHAR(255) | | | | | | | | | NULL |
| MAAS | BIGINT(20) | | | | | | | | | |
| MUSTERI_AD... | VARCHAR(255) | | | | | | | | | |

[<click to edit>](#)

[illegible][illegible]

1NF

| Column | Datatype | PK | NN | UQ | B... | UN | ZF | AI | G | Default / Expression |
|----------------|--------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|----------------------|
| ID | BIGINT(11) | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| AD | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| SOYAD | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| TELEFON | BIGINT(11) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| TC | BIGINT(11) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| ADRES | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| E-POSTA | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| TEMSILCI_ID | BIGINT(11) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| HESAPI_NO | BIGINT(11) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| HESAPI_BAK... | BIGINT(11) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| HESAPI2_NO | BIGINT(11) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| HESAPI2_BAK... | BIGINT(11) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |

[click to edit>](#)

| Column | Datatype | PK | NN | UQ | B... | UN | ZF | AI | G | Default / Expression |
|--------------|--------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|----------------------|
| ID | BIGINT | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| AD | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| SOYAD | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| TC | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| TELEFON | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| ADRES | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| E-POSTA | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| MAAS | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| MUSTERI_ID_1 | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| MUSTERI_ID_2 | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |

<click to edit>

| ID | AD | SOYAD | TELEFON | TC | ADRES | E-POSTA | TEMSİLÇİ_ID | HESAP1_NO | HESAP1_BAKIYE | HESAP2_NO | HESAP2_BAKIYE |
|----|-------|-------|------------|-------------|-----------------------------|---------------|-------------|-----------|---------------|-----------|---------------|
| 1 | ILKAY | BORA | 5555555555 | 11111111111 | wefrefg | ilkay@kou.com | 1 | 1 | 1 | 2 | 2 |
| 2 | SİNEM | ERC | 5555555555 | 22222222222 | ASDFDS RTHRTEH ERTHREH RHRH | SINEM@KOU.COM | 2 | 2 | 2 | 3 | 3 |

| ID | AD | SOYAD | TC | TELEFON | ADRES | E-POSTA | MAAS | MUSTERI_ID_1 | MUSTERI_ID_2 |
|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|--------------|--------------|
| 1 | A | A | 111111111 | 111111111 | ASDDSDC | ASDCADSC | 4000 | 1 | 2 |
| 2 | B | B | 2222222 | 22222222 | ASDCDC | DVDSV | 5000 | 3 | 4 |
| 3 | C | C | 3333333 | 33333333 | SDFEF | EFEFWE | 6000 | 5 | 6 |
| MUSTERI_1 | MUSTERI_2 | MUSTERI_3 | MUSTERI_4 | MUSTERI_5 | MUSTERI_6 | MUSTERI_7 | MUSTERI_8 | MUSTERI_9 | MUSTERI_10 |

2NF

Name:

Schema: bank

| Column | Datatype | PK | NN | UQ | B... | UN | ZF | AI | G | Default / Expression |
|-------------|--------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|----------------------|
| ID | BIGINT | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| AD | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| SOYAD | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| TEMSİLCİ_ID | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| TELEFON | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| TC | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| ADRES | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| E-POSTA | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |

Name:

Schema: bank

| Column | Datatype | PK | NN | UQ | B... | UN | ZF | AI | G | Default / Expression |
|---------|--------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|----------------------|
| ID | BIGINT | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| AD | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| SOYAD | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| TC | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| TELEFON | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| ADRES | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| E-POSTA | VARCHAR(255) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| MAAS | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |

Name:

Schema: bank

| Column | Datatype | PK | NN | UQ | B... | UN | ZF | AI | G | Default / Expression |
|-----------------|----------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|----------------------|
| MUSTERİ_ID | BIGINT | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| HESAP_NO | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| BAKIYE | BIGINT | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | NULL |
| <click to edit> | | | | | | | | | | |

| | MUSTERİ_ID | HESAP_NO | BAKIYE |
|---|------------|----------|----------|
| ▶ | 1 | 11 | 1000 |
| ▶ | 2 | 21 | 345456 |
| ▶ | 3 | 31 | 0 |
| ▶ | 3 | 32 | -4543.53 |
| ▶ | 4 | 41 | 5 |
| ▶ | NULL | NULL | NULL |

3NF

[illegible][illegible][illegible]

| | MUSTERI_ID | TEMSILCI_ID |
|---|------------|-------------|
| ▶ | 1 | 1 |
| | 2 | 2 |
| * | NULL | NULL |

[illegible][illegible]