

Hava Yolları Yönetim Sistemi

Projeyi Yapan Öğrenciler

Hasan Yılmaz Gürsoy (215260008)

Muhammet Eray Yıldırım (225260080)

Ahmet Taha Ural (225260014)

Bu proje, uçuş, rezervasyon, yolcu ve ödeme işlemlerini yöneten bir Hava Yolları Yönetim Sistemidir. Sistem, müşterilere uçuş arama, rezervasyon, check-in ve ödeme hizmetleri sunarken; yöneticilere uçak, havaalanı ve uçuş bilgilerini yönetme imkanı sağlar. Tüm süreçler, düzenli bir veritabanı tasarımı ve ilişkilerle optimize edilmiştir.

1. Proje Gereksinimleri

Hava Yolları Yönetim Sistemi uçuş, yolcu, rezervasyon ve ödeme işlemlerini yönetmek amacıyla geliştirilmiş bir veritabanı projesidir. Sistemin **temel gereksinimleri** aşağıda listelenmiştir.

1.1. Kullanıcı Türleri Müşteri

Kullanıcılar:

- Uçuş arama ve görüntüleme.
- Uçuş rezervasyonu yapma.
- Rezervasyon bilgilerini görüntüleme.
- Ödeme işlemlerini gerçekleştirme.
- Geri bildirim gönderme.
- Mil puanı kazanma ve kullanma.
- Check-in işlemi yapma.

Yönetici Kullanıcılar:

- Uçuş bilgilerini ekleme, düzenleme ve silme.
- Havaalanı bilgilerini güncelleme.
- Uçak bilgilerini yönetme.
- Kullanıcı geri bildirimlerini görüntüleme.
- Rezervasyon ve ödeme bilgilerini yönetme.

2. Varlıklar ve Nitelikler

2.1. Varlıklar

- **Kullanıcılar (TbKullanıcılar)**

- ✓ **Nitelikler:** KullaniciID, KullaniciAd, KullaniciSoyad, KullaniciEposta, KullaniciSifresi, KullaniciRol, TelefonNo, SonGirisTarihi, HesapDurumu
- ✓ **Kısıtlamalar:** KullaniciEposta benzersiz olmalıdır.

- **Havaalanları (TbHavaalanlari)**

- ✓ **Nitelikler:** HavaalaniNo, HavaalaniAd, HavaalaniUlke, HavaalaniSehir, IATAKodu, ICAOKodu, ZamanDilimi, TerminalSayisi, PistUzunlugu, YillikYolcuSayisi
- ✓ **Kısıtlamalar:** IATAKodu ve ICAOKodu benzersiz olmalıdır.

- **Uçaklar (TbUcaklar)**

- ✓ **Nitelikler:** UcakID, UcakSeriNo, UcakKapasite, UcakModel, UcakTipi, UcakMenzil, UcakUretimYili, UcakMotorSayisi, UcakHizi, KapasiteEkonomi, KapasiteBusiness
- ✓ **Kısıtlamalar:** UcakSeriNo benzersiz olmalıdır.

- **Uçuşlar (TbUcuslar)**

- ✓ **Nitelikler:** UcusKod, VarisHavaalani, KalkisHavaalani, UcakID, InisSaati, KalkisSaati, HavayoluSirketi, UcusSuresi, KapiNumarasi
- ✓ **Kısıtlamalar:** UcusKod benzersiz olmalıdır.

- **Yolcular (TbYolcular)**

- ✓ **Nitelikler:** YolcuID, YolcuAdi, YolcuSoyadi, YolcuTelNo, PasaportNumarasi, YolcuMail, YolcuSicil, YolcuUlke, YolcuEngelDurum, YolcuTcNo, YolcuMilPuan, YolcuDogumTarihi, YolcuCinsiyet, YolcuUyelikDurumu, YolcuAcilKisi, YolcuAcilNumara
- ✓ **Kısıtlamalar:** PasaportNumarasi ve YolcuTcNo benzersiz olmalıdır.

- **Rezervasyon (TbRezervasyon)**

- ✓ **Nitelikler:** RezervasyonNo, UcusID, YolcuID, KullaniciID, RezervasyonTarihi,
- ✓ RezervasyonDurum, KoltukNumarasi, Sinif, Fiyat, CheckinSaati, CheckinDurumu, BagajHakki **Kısıtlamalar:**

KoltukNumarasi benzersiz olmalıdır (aynı uçuş için).

- **Mil Puan (TbMilPuan)**

- ✓ **Nitelikler:** MilID, YolcuID, KazanilanMil, Tarih, MilDurumu, KullanimTarihi, MilKullanimYeri **Kısıtlamalar:** MilID

benzersiz olmalıdır.

- **Ek Hizmetler (TbEkHizmetler)**

- ✓ **Nitelikler:** HizmetID, RezervasyonID, HizmetAdi, HizmetUcreti
- ✓ **Kısıtlamalar:** HizmetID benzersiz olmalıdır.

- **Bagaj (TbBagaj)**

- ✓ **Nitelikler:** BagajID, RezervasyonID, BagajAgirlik, BagajBoyut, BagajTipi, EkBagajUcreti **Kısıtlamalar:** BagajID

benzersiz olmalıdır.

- **Geri Bildirim (TbGeriBildirim)**

- ✓ **Nitelikler:** GeriBildirimID, KullaniciID, UcusKod, Mesaj, Tarih, Durum
- ✓ **Kısıtlamalar:** GeriBildirimID benzersiz olmalıdır.

- **Check-in Bilgileri (TbCheckInBilgileri)**

- ✓ **Nitelikler:** CheckInID, RezervasyonID, CheckInTarihi, KontrolDurumu, KoltukNumarasi **Kısıtlamalar:** CheckInID benzersiz olmalıdır.

- **Ödeme (TbOdeme)**

- ✓ **Nitelikler:** OdemeID, RezervasyonID, OdemeTarihi, OdemeMiktari, OdemeYontemi, Durum **Kısıtlamalar:** OdemeID benzersiz olmalıdır.

3. İlişkiler ve Türleri

1. **TbUcuslar ile TbRezervasyon İlişkisi (1:N - Bireçok)**

- Bir uçuş, birden fazla rezervasyon ile ilişkilendirilebilir.
- Örneğin, aynı uçuş için birden fazla yolcu rezervasyon yapabilir. Bu yüzden, TbUcuslar tablosundaki UcusKod alanı, TbRezervasyon tablosundaki UcusID alanına bağlanmıştır.

2. **TbYolcular ile TbRezervasyon İlişkisi (1:N - Bireçok)**

- Bir yolcu, birden fazla rezervasyon yapabilir. Örneğin, aynı yolcu farklı tarihlerde farklı uçuşlar için rezervasyonlar oluşturabilir. Bu ilişki, TbYolcular tablosundaki YolcuID alanının TbRezervasyon tablosundaki YolcuID alanına bağlanmasıyla sağlanmıştır.

3. **TbKullanıcılar ile TbRezervasyon İlişkisi (1:N - Bireçok)**

- Bir kullanıcı, birden fazla rezervasyon oluşturabilir. Bu durum, kullanıcıların aynı uçuş için veya farklı uçuşlar için farklı yolcular adına rezervasyon yapabilmeleriyle ilgilidir. Bu ilişki, TbKullanıcılar tablosundaki KullaniciID alanının TbRezervasyon tablosundaki KullaniciID alanına bağlanmasıyla sağlanmıştır.

4. **TbRezervasyon ile TbEkHizmetler İlişkisi (1:N - Bireçok)**

- Bir rezervasyon, birden fazla ek hizmetle ilişkilendirilebilir. Örneğin, bir yolcu aynı rezervasyonda yemek, özel koltuk seçimi gibi farklı hizmetler alabilir. Bu ilişki, TbRezervasyon tablosundaki RezervasyonNo alanının TbEkHizmetler tablosundaki RezervasyonID alanına bağlanmasıyla sağlanmıştır.

5. **TbRezervasyon ile TbBagaj İlişkisi (1:N - Bireçok)**

- Bir rezervasyon, birden fazla bagaj kaydıyla ilişkilendirilebilir. Örneğin, bir yolcu aynı rezervasyonda birden fazla bagaj taşıyabilir. Bu ilişki, TbRezervasyon tablosundaki RezervasyonNo alanının TbBagaj tablosundaki RezervasyonID alanına bağlanmasıyla sağlanmıştır.

6. **TbRezervasyon ile TbCheckInBilgileri İlişkisi (1:1 - Birebir)**
 - Her rezervasyon için yalnızca bir check-in bilgisi tutulur. Örneğin, bir rezervasyon check-in yapıldığında, ilgili check-in bilgisi yalnızca bu rezervasyona ait olur. Bu ilişki, TbRezervasyon tablosundaki RezervasyonNo alanının TbCheckInBilgileri tablosundaki RezervasyonID alanına bağlanmasıyla sağlanmıştır.
7. **TbRezervasyon ile TbOdeme İlişkisi (1:1 - Birebir)**
 - Her rezervasyon için yalnızca bir ödeme bilgisi tutulur. Örneğin, bir rezervasyon için yapılan ödeme bilgisi, bu rezervasyonla doğrudan ilişkilidir. Bu ilişki, TbRezervasyon tablosundaki RezervasyonNo alanının TbOdeme tablosundaki RezervasyonID alanına bağlanmasıyla sağlanmıştır.
8. **TbYolcular ile TbMilPuan İlişkisi (1:N - Bireçok)**
 - Bir yolcu, uçuşlarından dolayı birden fazla mil puanı kazanabilir. Bu ilişki, TbYolcular tablosundaki YolcuID alanının TbMilPuan tablosundaki YolcuID alanına bağlanmasıyla sağlanmıştır.
9. **TbUcuslar ile TbGeriBildirim İlişkisi (1:N - Bireçok)**
 - Bir uçuş, birden fazla geri bildirim alabilir. Örneğin, aynı uçuşla ilgili birden fazla yolcu geri bildirim bırakabilir. Bu ilişki, TbUcuslar tablosundaki UcusKod alanının TbGeriBildirim tablosundaki UcusKod alanına bağlanmasıyla sağlanmıştır.
10. **TbKullanicilar ile TbGeriBildirim İlişkisi (1:N - Bireçok)**
 - Bir kullanıcı, birden fazla geri bildirim bırakabilir. Örneğin, farklı uçuşlar hakkında aynı kullanıcı farklı geri bildirimler yapabilir. Bu ilişki, TbKullanicilar tablosundaki KullaniciID alanının TbGeriBildirim tablosundaki KullaniciID alanına bağlanmasıyla sağlanmıştır.
11. **TbHavaalanlari ile TbUcuslar İlişkisi (1:N - Bireçok)**
 - Bir havaalanı, birden fazla uçuşla ilişkilendirilebilir. Örneğin, bir havaalanı hem kalkış hem de varış noktası olarak birçok uçuşa ev sahipliği yapabilir. Bu ilişki, TbHavaalanlari tablosundaki HavaalaniNo alanının TbUcuslar tablosundaki VarisHavaalani ve KalkisHavaalani alanlarına bağlanmasıyla sağlanmıştır.
12. **TbUcaklar ile TbUcuslar İlişkisi (1:N - Bireçok)**
 - Bir uçak, birden fazla uçuş gerçekleştirebilir. Örneğin, aynı uçak farklı tarihlerde farklı uçuşlar için kullanılabilir. Bu ilişki, TbUcaklar tablosundaki UcakID alanının TbUcuslar tablosundaki UcakID alanına bağlanmasıyla sağlanmıştır.

4. ER Diyagramı

1. TbUcaklar Tablosundaki Değişiklikler

- **KapasiteEkonomi ve KapasiteBusiness kolonları eklendi:**
 - Daha önce uçuşa özel bu bilgiler TbUcuslar tablosunda tutuluyordu. Ancak, uçakların kapasite bilgisi genellikle statiktir ve uçuş bazında değişmez. Bu nedenle, uçak detaylarının olduğu TbUcaklar tablosuna taşındı.
 - **Sebep:** Veri tekrarı ve tutarsızlık risklerini azaltmak için normalizasyon.

2. TbUcuslar Tablosundaki Değişiklikler

- **UcakTipi, MaksimumKapasite, KapasiteEkonomi, KapasiteBusiness kolonları kaldırıldı:**
 - Bu bilgiler TbUcaklar tablosuna taşındı, çünkü uçaklara özgüdür ve uçuş bazında değişmez.
 - **Sebep:** Fazla veri tekrarı ve tutarsızlıkları önlemek için normalizasyon.

3. TbRezervasyon Tablosundaki Değişiklikler

- **EkHizmetler kaldırıldı:**
 - Rezervasyon tablosunda çoklu değerlerin depolanması uygun değildir. Bunun yerine TbEkHizmetler adlı yeni bir tablo oluşturuldu.
 - **Sebep:** Çoktan-çoka ilişkilerin daha uygun şekilde yönetilmesi ve 3NF'ye uyum.

4. Yeni Eklenen Tablo: TbEkHizmetler

- Rezervasyonlarla ilişkili ek hizmetler için ayrı bir tablo eklendi.
 - **Sebep:** Çoklu ek hizmetleri depolamak için yeni bir yapı oluşturularak, veri tekrarı önleni ve esneklik sağlandı.

5. Küçük Optimizasyonlar

- **Bazı kolonların kaldırılması ya da taşınması:**
 - Gereksiz veya yanlış tabloda yer alan kolonlar uygun tablolara taşındı (örneğin, uçak tipi ve kapasite bilgileri TbUcaklar tablosuna taşındı).
 - **Sebep:** Normalizasyonun sağlanması ve tasarımın daha anlamlı hale getirilmesi.

Genel Hedefler:

- **BCNF ve 3NF'ye uyum:** Tablolar arasında veri tekrarını önlemek ve veritabanının esnekliğini artırmak.
- **Tasarımın anlamlı hale gelmesi:** Her tablonun yalnızca kendi konusu ile ilgili veri saklaması sağlandı.
- **İlişkilerin optimize edilmesi:** Çoktan-çoka ve bire-bir ilişkiler doğru şekilde tasarlandı.

TABLO OLUŞTURMA

--Kullanıcılar tablosu

```
CREATE TABLE TbKullanicilar (  
    KullaniciID INT PRIMARY KEY IDENTITY(1,1),  
    KullaniciAd VARCHAR(50),  
    KullaniciSoyad VARCHAR(50),  
    KullaniciEposta VARCHAR(50) UNIQUE,  
    KullaniciSifresi VARCHAR(50),  
    KullaniciRol VARCHAR(50),  
    TelefonNo VARCHAR(50),  
    SonGirisTarihi DATETIME,  
    HesapDurumu VARCHAR(50)  
);
```

-- Havaalanları tablosu

```
CREATE TABLE TbHavaalanlari (  
    HavaalaniNo SMALLINT PRIMARY KEY IDENTITY(1,1),  
    HavaalaniAd VARCHAR(50),  
    HavaalaniUlke NCHAR(30),
```



```
HavaalaniSehir NCHAR(30),  
  
IATAKodu CHAR(3),  
  
ICAOKodu CHAR(4),  
  
ZamanDilimi VARCHAR(50),  
  
TerminalSayisi INT,  
  
PistUzunlugu INT,  
  
YillikYolcuSayisi BIGINT  
  
);
```

-- Uçaklar tablosu

```
CREATE TABLE TbUcaklar (
```

```
    UcakID INT PRIMARY KEY IDENTITY(1,1),  
  
    UcakSeriNo VARCHAR(50),  
  
    UcakKapasite SMALLINT,  
  
    UcakModel VARCHAR(50),  
  
    UcakTipi NCHAR(20),
```

-- Güncellendi: Uçak tipi bu tabloda tutuluyor

```
    UcakMenzil INT,  
  
    UcakUretimYili DATE,  
  
    UcakMotorSayisi SMALLINT,  
  
    UcakHizi INT,  
  
    KapasiteEkonomi INT,
```

-- Güncellendi: Ekonomi kapasitesi bu tabloda tutuluyor

```
    KapasiteBusiness INT
```

-- Güncellendi: Business kapasitesi bu tabloda tutuluyor

```
);
```

-- Uçuşlar tablosu (Uçaklar ve Havaalanları ile ilişkili)

CREATE TABLE TbUcuslar (

UcusKod INT PRIMARY KEY IDENTITY(1,1),

VarisHavaalani SMALLINT,

KalkisHavaalani SMALLINT,

UcakID INT,

InisSaati DATETIME,

KalkisSaati DATETIME,

HavayoluSirketi NVARCHAR(100),

UcusSuresi VARCHAR(50),

KapiNumarasi VARCHAR(50),

FOREIGN KEY (UcakID) REFERENCES TbUcaklar(UcakID),

FOREIGN KEY (VarisHavaalani) REFERENCES TbHavaalanlari(HavaalaniNo),

FOREIGN KEY (KalkisHavaalani) REFERENCES TbHavaalanlari(HavaalaniNo)

-- Güncellendi: UcakTipi, MaksimumKapasite, KapasiteEkonomi, KapasiteBusiness kaldırıldı

);

-- Yolcular tablosu

CREATE TABLE TbYolcular (

YolcuID INT PRIMARY KEY IDENTITY(1,1),

YolcuAdi VARCHAR(50),

YolcuSoyadi VARCHAR(50),

YolcuTelNo VARCHAR(50),

PasaportNumarasi VARCHAR(50),

YolcuMail NVARCHAR(100),

```
YolcuSicil BIT,  
  
YolcuUlke NVARCHAR(50),  
  
YolcuEngelDurum BIT,  
  
YolcuTcNo VARCHAR(50),  
  
YolcuMilPuan INT,  
  
YolcuDogumTarihi DATE,  
  
YolcuCinsiyet BIT,  
  
YolcuUyelikDurumu VARCHAR(50),  
  
YolcuAcilKisi VARCHAR(50),  
  
YolcuAcilNumara VARCHAR(50)  
  
);
```

-- Mil Puanı tablosu (Yolcular ile ilişkili)

```
CREATE TABLE TbMilPuan (  
  
    MilID INT PRIMARY KEY IDENTITY(1,1),  
  
    YolcuID INT, KazanilanMil INT,  
  
    Tarih DATE,  
  
    MilDurumu VARCHAR(50) DEFAULT 'Kullanilabilir',  
  
    KullanimTarihi DATE,  
  
    MilKullanimYeri VARCHAR(50),  
  
    FOREIGN KEY (YolcuID) REFERENCES TbYolcular(YolcuID)  
  
);
```

-- Rezervasyon tablosu (Yolcular ve Uçuşlar ile ilişkili)

```
CREATE TABLE TbRezervasyon (  
  
    RezervasyonNo INT PRIMARY KEY IDENTITY(1,1),
```

```
UcusID INT,  
  
YolcuID INT,  
  
KullaniciID INT,  
  
-- Kullanıcı ile ilişki  
  
RezervasyonTarihi DATE,  
  
RezervasyonDurum VARCHAR(50),  
  
KoltukNumarasi VARCHAR(50),  
  
Sinif VARCHAR(50),  
  
Fiyat INT,  
  
CheckinSaati DATETIME,  
  
CheckinDurumu BIT,  
  
BagajHakki INT,  
  
FOREIGN KEY (UcusID) REFERENCES TbUcuslar(UcusKod),  
  
FOREIGN KEY (YolcuID) REFERENCES TbYolcular(YolcuID),  
  
FOREIGN KEY (KullaniciID) REFERENCES TbKullanicilar(KullaniciID)  
  
);
```

-- Ek Hizmetler tablosu (Yeni oluşturuldu: Çoklu değerler için)

```
CREATE TABLE TbEkHizmetler (  
  
    HizmetID INT PRIMARY KEY IDENTITY(1,1),  
  
    RezervasyonID INT,  
  
    HizmetAdi VARCHAR(50),  
  
    HizmetUcreti INT,  
  
    FOREIGN KEY (RezervasyonID) REFERENCES TbRezervasyon(RezervasyonNo)  
  
);
```

-- Bagaj tablosu (Rezervasyon ile ilişkili)

```
CREATE TABLE TbBagaj (  
  
    BagajID INT PRIMARY KEY IDENTITY(1,1),  
  
    RezervasyonID INT,  
  
    BagajAgirlik INT,  
  
    BagajBoyut VARCHAR(50),  
  
    BagajTipi VARCHAR(50),  
  
    EkBagajUcreti INT,  
  
    FOREIGN KEY (RezervasyonID) REFERENCES TbRezervasyon(RezervasyonNo)  
  
);
```

-- Geri Bildirim tablosu

```
CREATE TABLE TbGeriBildirim (  
  
    GeriBildirimID INT PRIMARY KEY IDENTITY(1,1),  
  
    KullaniciID INT,  
  
    UcusKod INT,  
  
    Mesaj NVARCHAR(255),  
  
    Tarih DATETIME DEFAULT GETDATE(),  
  
    Durum VARCHAR(50) DEFAULT 'Değerlendirilmiyor',  
  
    FOREIGN KEY (KullaniciID) REFERENCES TbKullanicilar(KullaniciID),  
  
    FOREIGN KEY (UcusKod) REFERENCES TbUcuslar(UcusKod)  
  
);
```

-- Check-in Bilgileri tablosu

```
CREATE TABLE TbCheckInBilgileri (  
  
    CheckInID INT PRIMARY KEY IDENTITY(1,1),  
  
    RezervasyonID INT,
```

```
CheckInTarihi DATETIME DEFAULT GETDATE(),

KontrolDurumu BIT DEFAULT 0,

-- 0: Yapılmadı, 1: Yapıldı

KoltukNumarasi VARCHAR(50),

FOREIGN KEY (RezervasyonID) REFERENCES TbRezervasyon(RezervasyonNo)

);
```

-- Ödeme tablosu (Rezervasyon ile ilişkili)

```
CREATE TABLE TbOdeme (

    OdemeID INT PRIMARY KEY IDENTITY(1,1),

    RezervasyonID INT,

    OdemeTarihi DATETIME DEFAULT GETDATE(),

    OdemeMiktari DECIMAL(10, 2),

    OdemeYontemi VARCHAR(50),

    -- Kredi kartı, nakit, vb.

    Durum VARCHAR(50) DEFAULT 'Başarılı',

    FOREIGN KEY (RezervasyonID) REFERENCES TbRezervasyon(RezervasyonNo)

);
```

ÖRNEK VERİ EKLEME

-- Kullanıcılar tablosuna örnek veriler

```
INSERT INTO TbKullanici (KullaniciAd, KullaniciSoyad, KullaniciEposta, KullaniciSifresi, KullaniciRol,
TelefonNo, SonGirisTarihi, HesapDurumu)
```

```
VALUES
```

('Ahmet', 'Yılmaz', 'ahmet.yilmaz@example.com', 'sifre123', 'Admin', '05551234567', '2024-12-01 10:00:00', 'Aktif'),

('Mehmet', 'Öztürk', 'mehmet.ozturk@example.com', 'sifre456', 'User', '05559876543', '2024-12-10 14:30:00', 'Aktif'),

('Ayşe', 'Çelik', 'ayse.celik@example.com', 'sifre789', 'User', '05551234876', '2024-12-15 11:20:00', 'Aktif'),

('Fatma', 'Şahin', 'fatma.sahin@example.com', 'sifre321', 'Admin', '05551234765', '2024-12-20 16:00:00', 'Pasif');

-- Havaalanları tablosuna örnek veriler

INSERT INTO TbHavaalanlari (HavaalaniAd, HavaalaniUlke, HavaalaniSehir, IATAKodu, ICAOKodu, ZamanDilimi, TerminalSayisi, PistUzunlugu, YillikYolcuSayisi)

VALUES

('İstanbul Havalimanı', 'Türkiye', 'İstanbul', 'IST', 'LTBA', 'GMT+3', 3, 4000, 60000000),

('Los Angeles International', 'Amerika', 'Los Angeles', 'LAX', 'KLAX', 'GMT-8', 4, 5000, 88000000),

('Dubai International', 'Birleşik Arap Emirlikleri', 'Dubai', 'DXB', 'OMDB', 'GMT+4', 3, 5000, 80000000),

('Heathrow Havalimanı', 'Birleşik Krallık', 'Londra', 'LHR', 'EGLL', 'GMT+0', 5, 4500, 75000000);

-- Uçaklar tablosuna örnek veriler

INSERT INTO TbUcaklar (UcakSeriNo, UcakKapasite, UcakModel, UcakTipi, UcakMenzil, UcakUretimYili, UcakMotorSayisi, UcakHizi, KapasiteEkonomi, KapasiteBusiness)

VALUES

('ABC123', 180, 'Boeing 737', 'Jet', 4500, '2018-05-12', 2, 800, 150, 30), ('XYZ456', 250, 'Airbus A320', 'Jet', 5200, '2020-06-15', 2, 850, 200, 50), ('DEF789', 300, 'Boeing 777', 'Jet', 10000, '2019-07-30', 2, 900, 250, 50), ('GHI101', 150, 'Embraer 190', 'Jet', 3000, '2015-03-25', 2, 750, 120, 30);

-- Uçuşlar tablosuna örnek veriler

INSERT INTO TbUcuslar (VarisHavaalani, KalkisHavaalani, UcakID, InisSaati, KalkisSaati, HavayoluSirketi, UcusSuresi, KapiNumarasi)

VALUES

(1, 2, 1, '2024-12-30 15:30:00', '2024-12-30 14:00:00', 'Turkish Airlines', '3 saat 30 dakika', 'A1'),

(2, 1, 2, '2024-12-30 18:00:00', '2024-12-30 16:30:00', 'American Airlines', '11 saat', 'B3'),

(3, 4, 3, '2024-12-31 09:00:00', '2024-12-31 07:30:00', 'Emirates', '6 saat', 'C1'),

(4, 1, 4, '2024-12-31 13:30:00', '2024-12-31 12:00:00', 'British Airways', '2 saat 30 dakika', 'D5');

-- Yolcular tablosuna örnek veriler

```
INSERT INTO TbYolcular (YolcuAdi, YolcuSoyadi, YolcuTelNo, PasaportNumarasi, YolcuMail, YolcuSicil,
YolcuUlke, YolcuEngelDurum, YolcuTcNo, YolcuMilPuan, YolcuDogumTarihi, YolcuCinsiyet,
YolcuUyelikDurumu, YolcuAcilKisi, YolcuAcilNumara)
```

VALUES

```
('Ali', 'Kara', '05551012345', 'P1234567', 'ali.kara@example.com', 0, 'Türkiye', 0, '12345678901', 1000,
'1990-05-21', 1, 'Aktif', 'Ayşe Kara', '05551098765'),
```

```
('Zeynep', 'Demir', '05552345678', 'P7654321', 'zeynep.demir@example.com', 0, 'Türkiye', 0,
'98765432101', 1500, '1985-08-15', 0, 'Aktif', 'Fatma Demir', '05552341234'),
```

```
('Veli', 'Yıldırım', '05553456789', 'P8765432', 'veli.yildirim@example.com', 1, 'Türkiye', 0, '11223344556',
2000, '1992-10-10', 1, 'Aktif', 'Mehmet Yıldırım', '05553456701'),
```

```
('Selin', 'Aksoy', '05554567890', 'P1357924', 'selin.aksoy@example.com', 0, 'Türkiye', 0, '22334455667',
2500, '1995-12-25', 1, 'Aktif', 'Ayşe Aksoy', '05554567891');
```

-- Mil Puanı tablosuna örnek veriler

```
INSERT INTO TbMilPuan (YolcuID, KazanilanMil, Tarih, MilDurumu, KullanimTarihi, MilKullanimYeri)
```

VALUES

```
(1, 500, '2024-12-01', 'Kullanilabilir', NULL, NULL),
```

```
(2, 1000, '2024-12-15', 'Kullanilabilir', NULL, NULL),
```

```
(3, 1500, '2024-12-18', 'Kullanilabilir', NULL, NULL),
```

```
(4, 2000, '2024-12-20', 'Kullanilabilir', NULL, NULL);
```

-- Rezervasyon tablosuna örnek veriler

```
INSERT INTO TbRezervasyon (UcusID, YolcuID, KullaniciID, RezervasyonTarihi, RezervasyonDurum,
KoltukNumarasi, Sinif, Fiyat, CheckinSaati, CheckinDurumu, BagajHakki)
```

VALUES

```
(1, 1, 1, '2024-12-01', 'Onaylı', '12A', 'Business', 3000, '2024-12-29 09:00:00', 1, 30),
```

```
(2, 2, 2, '2024-12-10', 'Beklemede', '22B', 'Ekonomi', 1500, '2024-12-30 08:00:00', 0, 20),
```

```
(3, 3, 3, '2024-12-12', 'Onaylı', '15A', 'Business', 3500, '2024-12-29 10:00:00', 1, 30),
```

```
(4, 4, 4, '2024-12-20', 'Beklemede', '10B', 'Ekonomi', 1800, '2024-12-30 07:00:00', 0, 20);
```

-- Ek Hizmetler tablosuna örnek veriler

INSERT INTO TbEkHizmetler (RezervasyonID, HizmetAdi, HizmetUcreti)

VALUES

(1, 'Açık büfe yemek', 200),

(2, 'Bagaj artışı', 100),

(3, 'Özel kabin', 500),

(4, 'Yolcu karşılama hizmeti', 300);

-- Bagaj tablosuna örnek veriler

INSERT INTO TbBagaj (RezervasyonID, BagajAgirlik, BagajBoyut, BagajTipi, EkBagajUcreti)

VALUES

(1, 25, '75x50x30', 'Kabin dışı', 100),

(2, 15, '55x40x20', 'Kabin içi', 0),

(3, 20, '65x45x25', 'Kabin dışı', 150),

(4, 18, '60x40x20', 'Kabin içi', 0);

-- Geri Bildirim tablosuna örnek veriler

INSERT INTO TbGeriBildirim (KullaniciID, UcusKod, Mesaj, Durum)

VALUES

(1, 1, 'Uçuş mükemmeldi, her şey yolundaydı.', 'Değerlendirilmiyor'),

(2, 2, 'Uçuş biraz uzun ama konforlu bir yolculuktu.', 'Değerlendirilmiyor'), (3, 3, 'Uçuş harika, personel çok yardımcıydı.', 'Değerlendirilmiyor'),

(4, 4, 'Uçak biraz geç kalktı ama genel olarak iyi bir deneyimdi.', 'Değerlendirilmiyor');

-- Check-in Bilgileri tablosuna örnek veriler

INSERT INTO TbCheckInBilgileri (RezervasyonID, KoltukNumarasi)

VALUES

(1, '12A'),

(2, '22B'),

(3, '15A'),

```
(4, '10B');
```

-- **Ödeme tablosuna örnek veriler**

```
INSERT INTO TbOdeme (RezervasyonID, OdemeMiktari, OdemeYontemi, Durum)
```

```
VALUES
```

```
(1, 3000, 'Kredi kartı', 'Başarılı'),
```

```
(2, 1500, 'Nakit', 'Başarılı'),
```

```
(3, 3500, 'Kredi kartı', 'Başarılı'),
```

```
(4, 1800, 'Nakit', 'Başarılı');
```

Stored Prodecure Kodları

-- **Kullanıcı Ekleme**

```
CREATE PROCEDURE spKullaniciEkle
```

```
@KullaniciAd VARCHAR(50),
```

```
@KullaniciSoyad VARCHAR(50),
```

```
@KullaniciEposta VARCHAR(50),
```

```
@KullaniciSifresi VARCHAR(50),
```

```
@KullaniciRol VARCHAR(50),
```

```
@TelefonNo VARCHAR(50),
```

```
@SonGirisTarihi DATETIME,
```

```
@HesapDurumu VARCHAR(50)
```

```
AS
```

BEGIN

BEGIN TRY

-- Yeni kullanıcıyı ekle

INSERT INTO TbKullanıcılar (KullanıcıAd, KullanıcıSoyad, KullanıcıEposta,
KullanıcıSifresi, KullanıcıRol, TelefonNo, SonGirisTarihi, HesapDurumu)

VALUES (@KullanıcıAd, @KullanıcıSoyad, @KullanıcıEposta,
@KullanıcıSifresi, @KullanıcıRol, @TelefonNo, @SonGirisTarihi, @HesapDurumu);

PRINT 'Kullanıcı başarıyla eklendi.';

END TRY

BEGIN CATCH

PRINT 'Hata oluştu: ' + ERROR_MESSAGE();

END CATCH

END;

-- Uçuş Güncelleme

CREATE PROCEDURE spUcusGuncelle

@UcusKod INT,

@VarisHavaalani SMALLINT,

@KalkisHavaalani SMALLINT,

@UcakID INT,

@InisSaati DATETIME,

@KalkisSaati DATETIME,

@HavayoluSirketi NVARCHAR(100),

@UcusSuresi VARCHAR(50),

@KapiNumarasi VARCHAR(50)

AS

BEGIN

BEGIN TRY

-- Uçuşu güncelle

UPDATE TbUcuslar

SET

VarisHavaalani = @VarisHavaalani,

KalkisHavaalani = @KalkisHavaalani,

UcakID = @UcakID,

InisSaati = @InisSaati,

KalkisSaati = @KalkisSaati,

HavayoluSirketi = @HavayoluSirketi,

UcusSuresi = @UcusSuresi,

KapiNumarasi = @KapiNumarasi

WHERE UcusKod = @UcusKod;

PRINT 'Uçuş başarıyla güncellendi.';

END TRY

BEGIN CATCH

PRINT 'Hata oluştu: ' + ERROR_MESSAGE();

END CATCH

END;

--Mill Puanı Güncelleme

CREATE PROCEDURE spMilPuanGuncelle

@YolcuID INT,

@KazanilanMil INT,

@MilDurumu VARCHAR(50)

AS

BEGIN

```

BEGIN TRY

    -- Mil puanlarını güncelle

    UPDATE TbMilPuan

    SET

        KazanilanMil = KazanilanMil + @KazanilanMil,

        MilDurumu = @MilDurumu,

        KullanimTarihi = GETDATE()

    WHERE YolcuID = @YolcuID;

    PRINT 'Mil puanı başarıyla güncellendi.';
END TRY
BEGIN CATCH
    PRINT 'Hata oluştu: ' + ERROR_MESSAGE();
END CATCH

END;

```

Stored Procedure ve Transaction Ortak Kullanım Senaryosu

--Rezervasyon Oluşturma

```

CREATE PROCEDURE spRezervasyonOlustur

    @UcusID INT,

    @YolcuID INT,

    @KullaniciID INT,

    @Sinif VARCHAR(50),

    @Fiyat INT,

    @KoltukNumarasi VARCHAR(50),

    @BagajHakki INT

```

AS

BEGIN

BEGIN TRANSACTION;

BEGIN TRY

-- Uçuş kapasitesini kontrol et

DECLARE @KalanKoltuk INT;

SELECT

@KalanKoltuk =

CASE

WHEN @Sinif = 'Ekonomi' THEN KapasiteEkonomi

WHEN @Sinif = 'Business' THEN KapasiteBusiness

END

FROM TbUcaklar

INNER JOIN TbUcuslar ON TbUcuslar.UcakID = TbUcaklar.UcakID

WHERE TbUcuslar.UcusKod = @UcusID;

-- Eğer kapasite yetersizse hata fırlat

IF (@KalanKoltuk <= 0)

BEGIN

THROW 50000, 'Yeterli koltuk bulunmamaktadır.', 1;

END

-- Rezervasyonu oluştur

INSERT INTO TbRezervasyon (UcusID, YolcuID, KullaniciID, RezervasyonTarihi,
RezervasyonDurum, KoltukNumarasi, Sinif, Fiyat, BagajHakki)

VALUES (@UcusID, @YolcuID, @KullaniciID, GETDATE(), 'Aktif', @KoltukNumarasi,
@Sinif, @Fiyat, @BagajHakki);

-- Uçak kapasitesini güncelle

UPDATE TbUcaklar

SET KapasiteEkonomi = CASE WHEN @Sinif = 'Ekonomi' THEN KapasiteEkonomi - 1
ELSE KapasiteEkonomi END,

KapasiteBusiness = CASE WHEN @Sinif = 'Business' THEN KapasiteBusiness - 1 ELSE
KapasiteBusiness END

WHERE TbUcaklar.UcakID = (SELECT UcakID FROM TbUcuslar WHERE UcusKod =
@UcusID);

-- Değişiklikleri kaydet

COMMIT TRANSACTION;

PRINT 'Rezervasyon başarıyla oluşturuldu.';

END TRY

BEGIN CATCH

-- Hata durumunda değişiklikleri geri al

ROLLBACK TRANSACTION;

```
        PRINT 'Hata oluştu: ' + ERROR_MESSAGE();  
    END CATCH
```

```
END;
```

Trigger Kodları

--Rezervasyon Silme

```
CREATE TRIGGER trgRezervasyonSil
```

```
ON TbRezervasyon
```

```
AFTER DELETE
```

```
AS
```

```
BEGIN
```

```
DECLARE @UcusID INT;
```

```
DECLARE @Sinif VARCHAR(50);
```

```
SELECT
```

```
    @UcusID = d.UcusID,
```

```
    @Sinif = d.Sinif
```

```
FROM DELETED d;
```

```
-- Kapasiteyi artır
```

```
UPDATE TbUcaklar
```

```
SET KapasiteEkonomi = CASE WHEN @Sinif = 'Ekonomi' THEN KapasiteEkonomi + 1 ELSE KapasiteEkonomi  
END,
```

```
    KapasiteBusiness = CASE WHEN @Sinif = 'Business' THEN KapasiteBusiness + 1 ELSE KapasiteBusiness  
END
```

```
WHERE TbUcaklar.UcakID = (SELECT UcakID FROM TbUcuslar WHERE UcusKod = @UcusID);
```

```
END;
```

--Geri Bildirim Güncelleme

```
CREATE TRIGGER trgGeriBildirimDurumGuncelle
```

```
ON TbGeriBildirim
```

```
AFTER INSERT
```

AS

BEGIN

DECLARE @GeriBildirimID INT;

DECLARE @Durum VARCHAR(50);

SELECT

@GeriBildirimID = gb.GeriBildirimID,

@Durum = gb.Durum

FROM INSERTED gb;

-- Geri bildirimde durum "Değerlendiriliyor" olarak ayarlandı

IF @Durum = 'Değerlendirilmiyor'

BEGIN

UPDATE TbGeriBildirim

SET Durum = 'Değerlendiriliyor'

WHERE GeriBildirimID = @GeriBildirimID;

END

END;

--Check-in Bilgisi Güncelleme

CREATE TRIGGER trgCheckInGuncelle

ON TbCheckInBilgileri

AFTER INSERT

AS

BEGIN

DECLARE @RezervasyonID INT;


```
DECLARE @CheckInDurumu BIT;
```

```
DECLARE @BagajHakki INT;
```

```
SELECT
```

```
    @RezervasyonID = ci.RezervasyonID,
```

```
    @CheckInDurumu = ci.KontrolDurumu
```

```
FROM INSERTED ci;
```

```
-- Check-in yapılmışsa rezervasyon durumunu güncelle
```

```
IF @CheckInDurumu = 1
```

```
BEGIN
```

```
    UPDATE TbRezervasyon
```

```
    SET RezervasyonDurum = 'Check-in Yapıldı'
```

```
    WHERE RezervasyonNo = @RezervasyonID;
```

```
-- Bagaj hakkı güncelleniyor
```

```
SELECT @BagajHakki = BagajHakki FROM TbRezervasyon WHERE RezervasyonNo = @RezervasyonID;
```

```
UPDATE TbRezervasyon
```

```
SET BagajHakki = @BagajHakki + 1
```

```
WHERE RezervasyonNo = @RezervasyonID;
```

```
END
```

```
END;
```

```
-- Uçuş Tarihi Güncelleme
```

```
CREATE TRIGGER trgUcusTarihiGuncelle
```

```
ON TbUcuslar
```

AFTER UPDATE

AS

BEGIN

DECLARE @UcusKod INT;

DECLARE @InisSaati DATETIME;

DECLARE @KalkisSaati DATETIME;

SELECT

@UcusKod = u.UcusKod,

@InisSaati = u.InisSaati,

@KalkisSaati = u.KalkisSaati

FROM INSERTED u;

-- Eğer uçuş saati değiştiyse, uçak kapasitesini güncelle

IF @InisSaati <> @KalkisSaati

BEGIN

UPDATE TbUcaklar

SET KapasiteEkonomi = KapasiteEkonomi - 1

WHERE UcakID = (SELECT UcakID FROM TbUcuslar WHERE UcusKod = @UcusKod);

END

END;