



The ER diagram illustrates the following tables and their attributes:

- Avukatlar**: AvukatID (PK), Ad, Soyad, UzmanlikAlani, BaroNumarasi.
- Muvekkiller**: MuvekkilID (PK), Ad, Soyad, Telefon, Eposta, Adres.
- Mahkemeler**: MahkemeID (PK), MahkemeAdi, Sehir, Adres.
- Davalar**: DavaID (PK), DavaTuru, BaslangicTarihi, Durum, MahkemeID (FK).
- Avukat\_Dava**: AvukatID (FK), DavaID (FK).
- Odemeler**: OdemeID (PK), MuvekkilID (FK), Tutar, Tarih, OdemeTuru.
- Dava\_Taraflari**: DavaID (FK), MuvekkilID (FK), AvukatID (FK), TarafRol.
- Sikayetler**: SikayetID (PK), DavaID (FK), SikayetMetni, SikayetTarihi.
- Muvekkil\_Avukat**: MuvekkilID (FK), AvukatID (FK), IliskiBaslangicTarihi.
- Belgeler**: BelgeID (PK), DavaID (FK), BelgeTuru, YuklenmeTarihi.

Relationships are indicated by lines connecting primary keys to foreign keys with crow's foot notation:

- Avukatlar** to **Muvekkiller**: One-to-many relationship.
- Muvekkiller** to **Mahkemeler**: One-to-many relationship.
- Mahkemeler** to **Davalar**: One-to-many relationship.
- Avukatlar** to **Avukat\_Dava**: One-to-many relationship.
- Davalar** to **Avukat\_Dava**: One-to-many relationship.
- Muvekkiller** to **Odemeler**: One-to-many relationship.
- Davalar** to **Dava\_Taraflari**: One-to-many relationship.
- Muvekkiller** to **Dava\_Taraflari**: One-to-many relationship.
- Avukatlar** to **Dava\_Taraflari**: One-to-many relationship.
- Dava\_Taraflari** to **Sikayetler**: One-to-many relationship.
- Muvekkil\_Avukat** to **Sikayetler**: One-to-many relationship.
- Davalar** to **Belgeler**: One-to-many relationship.

## 1. Avukatlar Tablosu Nitelikleri

- **AvukatID:** INT (Primary Key, Identity)
  - **Ad:** NVARCHAR(50) (NOT NULL)
  - **Soyad:** NVARCHAR(50) (NOT NULL)
  - **UzmanlikAlani:** NVARCHAR(100)
  - **BaroNumarasi:** NVARCHAR(20)
- 

## 2. Müvekkiller Tablosu Nitelikleri

- **MuvekkilID:** INT (Primary Key, Identity)
  - **Ad:** NVARCHAR(50) (NOT NULL)
  - **Soyad:** NVARCHAR(50) (NOT NULL)
  - **Telefon:** NVARCHAR(15)
  - **Eposta:** NVARCHAR(100)
  - **Adres:** NVARCHAR(200)
- 

## 3. Mahkemeler Tablosu Nitelikleri

- **MahkemeID:** INT (Primary Key, Identity)
  - **MahkemeAdi:** NVARCHAR(100) (NOT NULL)
  - **Sehir:** NVARCHAR(50)
  - **Adres:** NVARCHAR(200)
- 

## 4. Davalar Tablosu Nitelikleri

- **DavaID:** INT (Primary Key, Identity)
  - **DavaTuru:** NVARCHAR(100)
  - **BaslangicTarihi:** DATE (NOT NULL)
  - **Durum:** NVARCHAR(50)
  - **MahkemeID:** INT (Foreign Key, REFERENCES Mahkemeler(MahkemeID), ON DELETE CASCADE)
- 

## 5. Avukat\_Dava Tablosu Nitelikleri (Ara Tablo)

- **AvukatID:** INT (Primary Key, Foreign Key, REFERENCES Avukatlar(AvukatID), ON DELETE CASCADE)
  - **DavaID:** INT (Primary Key, Foreign Key, REFERENCES Davalar(DavaID), ON DELETE CASCADE)
-

## 6. Dava\_Taraflari Tablosu Nitelikleri (Ara Tablo)

- **DavaID:** INT (Primary Key, Foreign Key, REFERENCES Davalar(DavaID), ON DELETE CASCADE)
  - **MuvekkilID:** INT (Primary Key, Foreign Key, REFERENCES Muvekkiler(MuvekkilID), ON DELETE CASCADE)
  - **AvukatID:** INT (Primary Key, Foreign Key, REFERENCES Avukatlar(AvukatID), ON DELETE CASCADE)
  - **TarafRol:** NVARCHAR(50)
- 

## 7. Müvekkil\_Avukat Tablosu Nitelikleri (Ara Tablo)

- **MuvekkilID:** INT (Primary Key, Foreign Key, REFERENCES Muvekkiler(MuvekkilID), ON DELETE CASCADE)
  - **AvukatID:** INT (Primary Key, Foreign Key, REFERENCES Avukatlar(AvukatID), ON DELETE CASCADE)
  - **IliskiBaslangicTarihi:** DATE
- 

## 8. Belgeler Tablosu Nitelikleri

- **BelgeID:** INT (Primary Key, Identity)
  - **DavaID:** INT (Foreign Key, REFERENCES Davalar(DavaID), ON DELETE CASCADE)
  - **BelgeTuru:** NVARCHAR(50)
  - **YuklenmeTarihi:** DATE (NOT NULL)
- 

## 9. Ödemeler Tablosu Nitelikleri

- **OdemeID:** INT (Primary Key, Identity)
  - **MuvekkilID:** INT (Foreign Key, REFERENCES Muvekkiler(MuvekkilID), ON DELETE CASCADE)
  - **Tutar:** DECIMAL(10, 2) (NOT NULL)
  - **Tarih:** DATE (NOT NULL)
  - **OdemeTuru:** NVARCHAR(50)
-

## 10. Şikayetler Tablosu Nitelikleri

- **SikayetID:** INT (Primary Key, Identity)
  - **DavaID:** INT (Foreign Key, REFERENCES Davalar(DavaID), ON DELETE CASCADE)
  - **SikayetMetni:** NVARCHAR(MAX) (NOT NULL)
  - **SikayetTarihi:** DATE (NOT NULL)
- 

## Tablolar Arasındaki İlişkiler

1. **Avukatlar ile Müvekkiller**
  - **N-N İlişkisi:** Müvekkil\_Avukat ara tablosu üzerinden, bir avukat birden fazla müvekkile ve bir müvekkil birden fazla avukata sahip olabilir.
2. **Avukatlar ile Davalar**
  - **N-N İlişkisi:** Avukat\_Dava ara tablosu üzerinden, bir avukat birden fazla davaya ve bir dava birden fazla avukata sahip olabilir.
3. **Davalar ile Mahkemeler**
  - **1-N İlişkisi:** Bir mahkeme birçok davaya ev sahipliği yapabilir, ancak her dava sadece bir mahkemeye atanabilir.
4. **Davalar ile Müvekkiller**
  - **N-N İlişkisi:** Dava\_Taraflari ara tablosu üzerinden, bir dava birden fazla müvekkili ve bir müvekkil birden fazla davayı içerebilir.
5. **Davalar ile Belgeler**
  - **1-N İlişkisi:** Bir dava birden fazla belgeye sahip olabilir, ancak her belge sadece bir davaya ait olabilir.
6. **Davalar ile Şikayetler**
  - **1-N İlişkisi:** Bir dava birden fazla şikayete sahip olabilir, ancak her şikayet sadece bir davaya ait olabilir.
7. **Müvekkiller ile Ödemeler**
  - **1-N İlişkisi:** Bir müvekkil birden fazla ödeme yapabilir, ancak her ödeme bir müvekkile ait olabilir.

### **AVUKATLAR TABLOSU**

```
CREATE TABLE Avukatlar (  
    AvukatID INT PRIMARY KEY IDENTITY(1,1),  
    Ad NVARCHAR(50) NOT NULL,  
    Soyad NVARCHAR(50) NOT NULL,  
    UzmanlikAlanı NVARCHAR(100),  
    BaroNumarasi NVARCHAR(20)  
);
```

### **MÜVEKKİLLER TABLOSU**

```
CREATE TABLE Muvekkiller (  
    MuvekkilID INT PRIMARY KEY IDENTITY(1,1),  
    Ad NVARCHAR(50) NOT NULL,  
    Soyad NVARCHAR(50) NOT NULL,  
    Telefon NVARCHAR(15),  
    Eposta NVARCHAR(100),  
    Adres NVARCHAR(200)  
);
```

### **MAHKEMELER TABLOSU**

```
CREATE TABLE Mahkemeler (  
    MahkemeID INT PRIMARY KEY IDENTITY(1,1),  
    MahkemeAdi NVARCHAR(100) NOT NULL,  
    Sehir NVARCHAR(50),  
    Adres NVARCHAR(200)  
);
```

## **DAVALAR TABLOSU**

CREATE TABLE Davalar (

DavaID INT PRIMARY KEY IDENTITY(1,1),

DavaTuru NVARCHAR(100),

BaslangicTarihi DATE NOT NULL,

Durum NVARCHAR(50),

MahkemeID INT,

FOREIGN KEY (MahkemeID) REFERENCES Mahkemeler(MahkemeID) ON  
DELETE CASCADE

);

## **AVUKAT-DAVA İLİŞKİ TABLOSU**

CREATE TABLE Avukat\_Dava (

AvukatID INT,

DavaID INT,

PRIMARY KEY (AvukatID, DavaID),

FOREIGN KEY (AvukatID) REFERENCES Avukatlar(AvukatID) ON  
DELETE CASCADE,

FOREIGN KEY (DavaID) REFERENCES Davalar(DavaID) ON DELETE  
CASCADE

);

## **DAVA TARAFLARI TABLOSU**

CREATE TABLE Dava\_Taraflari (

DavaID INT,

MuvekkilID INT NOT NULL,

AvukatID INT NOT NULL,

TarafRol NVARCHAR(50),

PRIMARY KEY (DavaID, MuvekkilID, AvukatID),

FOREIGN KEY (DavaID) REFERENCES Davalar(DavaID) ON DELETE CASCADE,

FOREIGN KEY (MuvekkilID) REFERENCES Muvekkiller(MuvekkilID) ON DELETE CASCADE,

FOREIGN KEY (AvukatID) REFERENCES Avukatlar(AvukatID) ON DELETE CASCADE

);

## **MÜVEKKİL-AVUKAT İLİŞKİ TABLOSU**

CREATE TABLE Muvekkil\_Avukat (

MuvekkilID INT,

AvukatID INT,

IliskiBaslangicTarihi DATE,

PRIMARY KEY (MuvekkilID, AvukatID),

FOREIGN KEY (MuvekkilID) REFERENCES Muvekkiller(MuvekkilID) ON DELETE CASCADE,

FOREIGN KEY (AvukatID) REFERENCES Avukatlar(AvukatID) ON DELETE CASCADE

);

## **BELGELER TABLOSU**

```
CREATE TABLE Belgeler (  
    BelgeID INT PRIMARY KEY IDENTITY(1,1),  
    DavaID INT,  
    BelgeTuru NVARCHAR(50),  
    YuklenmeTarihi DATE NOT NULL,  
    FOREIGN KEY (DavaID) REFERENCES Davalar(DavaID) ON DELETE  
    CASCADE  
);
```

## **ÖDEMELER TABLOSU**

```
CREATE TABLE Odemeler (  
    OdemeID INT PRIMARY KEY IDENTITY(1,1),  
    MuvekkilID INT,  
    Tutar DECIMAL(10, 2) NOT NULL,  
    Tarih DATE NOT NULL,  
    OdemeTuru NVARCHAR(50),  
    FOREIGN KEY (MuvekkilID) REFERENCES Muvekkiller(MuvekkilID) ON  
    DELETE CASCADE  
);
```

## **ŞİKAYETLER TABLOSU**

```
CREATE TABLE Sikayetler (  
    SikayetID INT PRIMARY KEY IDENTITY(1,1),  
    DavaID INT,  
    SikayetMetni NVARCHAR(MAX) NOT NULL,  
    SikayetTarihi DATE NOT NULL,  
    FOREIGN KEY (DavaID) REFERENCES Davalar(DavaID) ON DELETE  
    CASCADE);
```



## Avukatlar Tablosuna Ekleme

```
INSERT INTO Avukatlar (Ad, Soyad, UzmanlikAlanı, BaroNumarasi)
VALUES
('Ahmet', 'Yılmaz', 'Ceza Hukuku', 'BR12345'),
('Elif', 'Demir', 'Aile Hukuku', 'BR23456'),
('Kemal', 'Kaya', 'Ticaret Hukuku', 'BR34567');
```

## Muvekkiller Tablosuna Ekleme

```
INSERT INTO Muvekkiller (Ad, Soyad, Telefon, Eposta, Adres)
VALUES
('Mehmet', 'Can', '5551234567', 'mehmetcan@example.com', 'İstanbul, Kadıköy'),
('Ayşe', 'Öztürk', '5559876543', 'ayseozturk@example.com', 'Ankara, Çankaya'),
('Ali', 'Kurt', '5556543210', 'alikurt@example.com', 'İzmir, Karşıyaka');
```

## Mahkemeler Tablosuna Ekleme

```
INSERT INTO Mahkemeler (MahkemeAdi, Sehir, Adres)
VALUES
('İstanbul 5. Ağır Ceza Mahkemesi', 'İstanbul', 'Adliye Sarayı, Çağlayan'),
('Ankara 2. Aile Mahkemesi', 'Ankara', 'Adliye Sarayı, Sıhhiye'),
('İzmir 3. Ticaret Mahkemesi', 'İzmir', 'Adliye Sarayı, Bayraklı');
```

## Davalar Tablosuna Ekleme

```
INSERT INTO Davalar (DavaTuru, BaslangicTarihi, Durum, MahkemeID)
VALUES
('Dolandırıcılık', '2024-01-10', 'Devam Ediyor', 1),
('Boşanma', '2024-02-15', 'Beklemede', 2),
('Haksız Rekabet', '2024-03-01', 'Tamamlandı', 3);
```

## Avukat\_Dava Tablosuna Ekleme

```
INSERT INTO Avukat_Dava (AvukatID, DavaID)
VALUES
(1, 1),
(2, 2),
(3, 3);
```

## Dava\_Tarafları Tablosuna Ekleme

```
INSERT INTO Dava_Taraflari (DavaID, MuvekkilID, AvukatID, TarafRol)
VALUES
(1, 1, 1, 'Davacı'),
(2, 2, 2, 'Davalı'),
(3, 3, 3, 'Davacı');
```

## Muvekkil\_Avukat Tablosuna Ekleme

```
INSERT INTO Muvekkil_Avukat (MuevekkilID, AvukatID, IliskiBaslangicTarihi)
VALUES
(1, 1, '2023-01-01'),
(2, 2, '2023-02-01'),
(3, 3, '2023-03-01');
```

## Belgeler Tablosuna Ekleme

```
INSERT INTO Belgeler (DavaID, BelgeTuru, YuklenmeTarihi)
VALUES
(1, 'İddianame', '2024-01-15'),
(2, 'Dilekçe', '2024-02-20'),
(3, 'Karar', '2024-03-10');
```

## Odemeler Tablosuna Ekleme

```
INSERT INTO Odemeler (MuvekkilID, Tutar, Tarih, OdemeTuru)
VALUES
(1, 5000.00, '2024-01-12', 'Nakit'),
(2, 3000.00, '2024-02-18', 'Havale'),
(3, 7000.00, '2024-03-05', 'Kredi Kartı');
```

## Sikayetler Tablosuna Ekleme

```
INSERT INTO Sikayetler (DavaID, SikayetMetni, SikayetTarihi)
VALUES
(1, 'Sanık suçlamaları reddediyor.', '2024-01-20'),
(2, 'Mahkeme tarihi sürekli erteleniyor.', '2024-02-25'),
(3, 'Avukatın iletişimi yetersiz.', '2024-03-15');
```

## 1. Saklı Yordam (Stored Procedure)

```
CREATE PROCEDURE YeniOdemeEkle
    @MuevkkilID INT,
    @Tutar DECIMAL(10, 2),
    @Tarih DATE,
    @OdemeTuru NVARCHAR(50)
AS
BEGIN
    INSERT INTO Odemeler (MuevkkilID, Tutar, Tarih, OdemeTuru)
    VALUES (@MuevkkilID, @Tutar, @Tarih, @OdemeTuru);

    PRINT 'Yeni ödeme başarıyla eklendi.';
END;
GO
```

## 2. OdemeGecmisi Tablosunun Oluşturulması

```
CREATE TABLE OdemeGecmisi (
    MuevkkilID INT PRIMARY KEY,
    ToplamTutar DECIMAL(15, 2),
    SonOdemeTarihi DATE,
    FOREIGN KEY (MuevkkilID) REFERENCES Muevkkiller(MuevkkilID)
);
```

### 3. Trigger (Tetikleyici)

```
CREATE TRIGGER OdemeGuncelleTrigger
ON Odemeler
AFTER INSERT
AS
BEGIN

    UPDATE OdemeGecmisi
    SET
        ToplamTutar = ToplamTutar + INSERTED.Tutar,
        SonOdemeTarihi = INSERTED.Tarih
    FROM OdemeGecmisi
    INNER JOIN INSERTED ON OdemeGecmisi.MuvekkilID = INSERTED.MuvekkilID;

    INSERT INTO OdemeGecmisi (MuvekkilID, ToplamTutar, SonOdemeTarihi)
    SELECT MuvekkilID, Tutar, Tarih
    FROM INSERTED
    WHERE NOT EXISTS (
        SELECT 1
        FROM OdemeGecmisi
        WHERE OdemeGecmisi.MuvekkilID = INSERTED.MuvekkilID
    );
END;
```

### 4. Test - Saklı Yordamı ve Trigger'ı Test Etme

```
EXEC YeniOdemeEkle
    @MuvekkilID = 3,
    @Tutar = 1500.00,
    @Tarih = '2024-12-30',
    @OdemeTuru = 'Havale';
SELECT * FROM OdemeGecmisi;
```

### 5. Transaction Yöntemi

```
BEGIN TRY
    BEGIN TRANSACTION;

    INSERT INTO Odemeler (MuvekkilID, Tutar, Tarih, OdemeTuru)
    VALUES (1, 4000.00, '2024-12-29', 'Kredi Kartı');

    UPDATE Davalar
    SET Durum = 'Ödeme Yapıldı'
    WHERE DavaID = 1;

    INSERT INTO Belgeler (DavaID, BelgeTuru, YuklenmeTarihi)
    VALUES (1, 'Ödeme Makbuzu', '2024-12-29');

    COMMIT;
    PRINT 'Tüm işlemler başarıyla tamamlandı.';

END TRY
BEGIN CATCH

    ROLLBACK;
    PRINT 'Bir hata oluştu. İşlem geri alındı. Hata: ' + ERROR_MESSAGE();
END CATCH;
```