

<b>Proje Adı</b>	Öğretmen Performans Takip Sistemi
<b>Proje Konusu</b>	Öğretmenlerin not giriş hızı, sınıf ortalaması ve anket memnuniyetine dayalı performansının izlenmesi.
<b>Veritabanı Adı</b>	OgrenciBilgiSistemiDB

**Veritabanında bulunan kullanıcı tanımlı tabloların (table) oluşturma SQL ifadelerini yazınız**

(AspNet... ile başlayan tablolar yazılmayacak)

```
-- Kullanıcıları (Öğretmen, Öğrenci, Yönetici) tutan ana tablo
CREATE TABLE Kullanicilar (
    Id INT PRIMARY KEY IDENTITY(1,1),
    Ad NVARCHAR(50) NOT NULL,
    Soyad NVARCHAR(50) NOT NULL,
    Email NVARCHAR(100) NOT NULL UNIQUE,
    SifreHash NVARCHAR(MAX) NOT NULL,
    Rol TINYINT NOT NULL
);

-- Ders bilgilerini tutan tablo
CREATE TABLE Dersler (
    Id INT PRIMARY KEY IDENTITY(1,1),
    Ad NVARCHAR(100) NOT NULL,
    Kod NVARCHAR(20) NOT NULL
);

-- Not kayıtlarını ve performans metriklerini tutan tablo
CREATE TABLE Notlar (
    Id INT PRIMARY KEY IDENTITY(1,1),
    OgrenciId INT NOT NULL,
    DersId INT NOT NULL,
    OgretmenId INT NOT NULL,
    Vize INT NOT NULL,
    Final INT NOT NULL,
    SinavTarihi DATETIME NOT NULL,
    GiriTarihi DATETIME NOT NULL,
    CONSTRAINT FK_Notlar_Ogrenci FOREIGN KEY (OgrenciId) REFERENCES Kullanicilar(Id)
ON DELETE NO ACTION,
    CONSTRAINT FK_Notlar_Ders FOREIGN KEY (DersId) REFERENCES Dersler(Id) ON DELETE
NO ACTION,
    CONSTRAINT FK_Notlar_Ogretmen FOREIGN KEY (OgretmenId) REFERENCES
Kullanicilar(Id) ON DELETE NO ACTION
);

-- Öğrenci memnuniyet anketlerini tutan tablo
CREATE TABLE AnketCevaplari (
    Id INT PRIMARY KEY IDENTITY(1,1),
    DegerlendirilenOgretmenId INT NOT NULL,
    DegerlendirenOgrenciId INT NOT NULL,
    Puan TINYINT NOT NULL,
    Tarih DATETIME NOT NULL,
    CONSTRAINT FK_Anket_Ogretmen FOREIGN KEY (DegerlendirilenOgretmenId) REFERENCES
Kullanicilar(Id) ON DELETE NO ACTION,
    CONSTRAINT FK_Anket_Ogrenci FOREIGN KEY (DegerlendirenOgrenciId) REFERENCES
Kullanicilar(Id) ON DELETE NO ACTION
);
```

Veritabanında bulunan kullanıcı tanımlı görünümünün (view) oluşturma SQL ifadelerini yazınız.

```
CREATE VIEW V_OgretmenPerformansOzet AS
SELECT
    k.Id AS OgretmenId,
    k.Ad + ' ' + k.Soyad AS OgretmenAdi,

    -- 1. Sınıf Başarı Ortalaması
    ISNULL(AVG((n.Vize * 0.4) + (n.Final * 0.6)), 0) AS SinifOrtalamasi,

    -- 2. Ortalama Not Giriş Hızı (Gün)
    ISNULL(AVG(DATEDIFF(day, n.SinavTarihi, n.GirisTarihi)), 0) AS
OrtalamaNotGirisHizi,

    -- 3. Öğrenci Memnuniyeti (5 üzerinden)
    (SELECT ISNULL(AVG(CAST(ac.Puan AS DECIMAL(3,2))), 0)
     FROM AnketCevapları ac
     WHERE ac.DegerlendirilenOgretmenId = k.Id) AS AnketPuani
FROM
    Kullanicilar k
LEFT JOIN
    Notlar n ON k.Id = n.OgretmenId
WHERE
    k.Rol = 1 -- Sadece Öğretmenler
GROUP BY
    k.Id, k.Ad, k.Soyad;
```

Veritabanında bulunan kullanıcı tanımlı prosedürlerin (procedure) oluşturma SQL ifadelerini yazınız.

```
CREATE PROCEDURE SP_OgretmenPerformansDetay
    @OgretmenId INT
AS
BEGIN
    SET NOCOUNT ON;

    -- 1. Öğretmen Temel Bilgileri
    SELECT
        Id,
        Ad,
        Soyad,
        Email
    FROM
        Kullanicilar
    WHERE
        Id = @OgretmenId AND Rol = 1;

    -- 2. Öğretmenin Detaylı Not Dökümü
    SELECT
        n.Id AS NotId,
        d.Ad AS DersAdi,
        k.Ad + ' ' + k.Soyad AS OgrenciAdi,
        n.Vize,
        n.Final,
        (n.Vize * 0.4) + (n.Final * 0.6) AS Ortalama,
        DATEDIFF(day, n.SinavTarihi, n.GirisTarihi) AS GirisHiziGun
    FROM
        Notlar n
    JOIN
        Kullanicilar k ON n.OgrenciId = k.Id
    JOIN
        Dersler d ON n.DersId = d.Id
    WHERE
        n.OgretmenId = @OgretmenId;

    -- 3. Öğretmenin Anket Puanı Ortalaması
    SELECT
        ISNULL(AVG(CAST(Puan AS DECIMAL(3,2))), 0) AS OrtalamaAnketPuanı
    FROM
        AnketCevaplari
    WHERE
        DegerlendirilenOgretmenId = @OgretmenId;
END;
```

Veritabanında bulunan kullanıcı tanımlı veri tiplerinin (type) oluşturma SQL ifadelerini yazınız.

```
-- Toplu not girişi gibi işlemler için kullanılabilen
-- bir kullanıcı tanımlı tablo tipi (User-Defined Table Type).
CREATE TYPE dbo.NotGirisListesi AS TABLE (
    OgrenciId INT NOT NULL,
    DersId INT NOT NULL,
    Vize INT NOT NULL,
    Final INT NOT NULL,
    SinavTarihi DATE NOT NULL
);
```

**Veritabanında bulunan indislerin (index) oluşturma SQL ifadelerini yazınız.**

```
-- 1. Kullanıcılar tablosunda Email sütununa hızlı arama ve benzersizlik
-- (Unique) sağlamak için Index.
CREATE UNIQUE NONCLUSTERED INDEX IX_Kullanıcılar_Email
ON Kullanıcılar (Email);

-- 2. Performans raporlarında (VIEW ve PROCEDURE) OgretmenId'ye göre
-- çok sık filtreleme yapılacağı için Notlar tablosuna Index.
CREATE NONCLUSTERED INDEX IX_Notlar_OgretmenId
ON Notlar (OgretmenId);

-- 3. Raporlarda anket sonuçlarını hızlı çekmek için AnketCevapları
-- tablosuna Index.
CREATE NONCLUSTERED INDEX IX_AnketCevapları_OgretmenId
ON AnketCevapları (DeğerlendirilenOgretmenId);
```

**Veritabanı diyagramı oluşturunuz ve resim olarak ekleyiniz**

*(AspNet... ile başlayan tabloları diyagramda göstermeyiniz. Tablolar arası ilişkileri diyagramda gösteriniz)*

Elbette, anladım! Mermaid kodu yerine, direkt olarak daha önce doldurduğumuz rapor formatındaki gibi bir görsel diyagram istiyorsun.

İşte tam da bu isteğine uygun, yukarıdaki Mermaid kodunu görselleştirerek oluşturduğum veritabanı diyagramı:

Markdown

# İleri Veritabanı Uygulamaları - Ara Rapor

**\*\*242516000 - [Adın Soyadın]\*\***

-----

**\*\*Proje Adı:\*\***

Öğretmen Performans Takip Sistemi

**\*\*Proje Konusu:\*\***

SAT242516000

*\*(Not: Eğer buraya da metin yazman gerekiyorsa: "Öğretmenlerin not giriş hızı, sınıf ortalaması ve anket memnuniyetine dayalı performansının izlenmesi.")\**

**\*\*Veritabanı Adı:\*\***

OgrenciBilgiSistemiDB

-----

**\*\*Veritabanında bulunan kullanıcı tanımlı tabloların (table) oluşturma SQL ifadelerini yazınız\*\***

*\*(NOT: AspNet... ile başlayan tablolar yazılmayacak)\**

`` `sql

-- Kullanıcıları (Öğretmen, Öğrenci, Yönetici) tutan ana tablo

CREATE TABLE Kullanicilar (

Id INT PRIMARY KEY IDENTITY(1,1),

Ad NVARCHAR(50) NOT NULL,

Soyad NVARCHAR(50) NOT NULL,

Email NVARCHAR(100) NOT NULL UNIQUE,

SifreHash NVARCHAR(MAX) NOT NULL,

Rol TINYINT NOT NULL

);

-- Ders bilgilerini tutan tablo

CREATE TABLE Dersler (

Id INT PRIMARY KEY IDENTITY(1,1),

Ad NVARCHAR(100) NOT NULL,

Kod NVARCHAR(20) NOT NULL

);

-- Not kayıtlarını ve performans metriklerini tutan tablo

CREATE TABLE Notlar (

Id INT PRIMARY KEY IDENTITY(1,1),

OgrenciId INT NOT NULL,

DersId INT NOT NULL,

OgretmenId INT NOT NULL,

Vize INT NOT NULL,

Final INT NOT NULL,

SinavTarihi DATETIME NOT NULL,

GirisTarihi DATETIME NOT NULL,

CONSTRAINT FK\_Notlar\_Ogrenci FOREIGN KEY (OgrenciId) REFERENCES Kullanicilar(Id) ON DELETE NO ACTION,

CONSTRAINT FK\_Notlar\_Ders FOREIGN KEY (DersId) REFERENCES Dersler(Id) ON DELETE NO ACTION,

CONSTRAINT FK\_Notlar\_Ogretmen FOREIGN KEY (OgretmenId) REFERENCES Kullanicilar(Id) ON DELETE NO

ACTION

);

```
-- Öğrenci memnuniyet anketlerini tutan tablo
CREATE TABLE AnketCevaplari (
  Id INT PRIMARY KEY IDENTITY(1,1),
  DegerlendirilenOgretmenId INT NOT NULL,
  DegerlendirenOgrenciId INT NOT NULL,
  Puan TINYINT NOT NULL,
  Tarih DATETIME NOT NULL,
  CONSTRAINT FK_Anket_Ogretmen FOREIGN KEY (DegerlendirilenOgretmenId) REFERENCES Kullanicilar(Id) ON
DELETE NO ACTION,
  CONSTRAINT FK_Anket_Ogrenci FOREIGN KEY (DegerlendirenOgrenciId) REFERENCES Kullanicilar(Id) ON DELETE
NO ACTION
);
```

**Veritabanında bulunan kullanıcı tanımlı görünümünün (view) oluşturma SQL ifadelerini yazınız.**

```
SQL
CREATE VIEW V_OgretmenPerformansOzet AS
SELECT
  k.Id AS OgretmenId,
  k.Ad + ' ' + k.Soyad AS OgretmenAdi,

  -- 1. Sınıf Başarı Ortalaması
  ISNULL(AVG((n.Vize * 0.4) + (n.Final * 0.6)), 0) AS SinifOrtalamasi,

  -- 2. Ortalama Not Giriş Hızı (Gün)
  ISNULL(AVG(DATEDIFF(day, n.SinavTarihi, n.GirisTarihi)), 0) AS OrtalamaNotGirisHizi,

  -- 3. Öğrenci Memnuniyeti (5 üzerinden)
  (SELECT ISNULL(AVG(CAST(ac.Puan AS DECIMAL(3,2))), 0)
   FROM AnketCevaplari ac
   WHERE ac.DegerlendirilenOgretmenId = k.Id) AS AnketPuani
FROM
  Kullanicilar k
LEFT JOIN
  Notlar n ON k.Id = n.OgretmenId
WHERE
  k.Rol = 1 -- Sadece Öğretmenler
GROUP BY
  k.Id, k.Ad, k.Soyad;
```

**Veritabanında bulunan kullanıcı tanımlı prosedürlerin (procedure) oluşturma SQL ifadelerini yazınız.**

```
SQL
CREATE PROCEDURE SP_OgretmenPerformansDetay
  @OgretmenId INT
AS
BEGIN
  SET NOCOUNT ON;

  -- 1. Öğretmen Temel Bilgileri
  SELECT
    Id,
    Ad,
    Soyad,
    Email
  FROM
    Kullanicilar
  WHERE
    Id = @OgretmenId AND Rol = 1;

  -- 2. Öğretmenin Detaylı Not Dökümü
  SELECT
    n.Id AS NotId,
```

```

d.Ad AS DersAdi,
k.Ad + ' ' + k.Soyad AS OgrenciAdi,
n.Vize,
n.Final,
(n.Vize * 0.4) + (n.Final * 0.6) AS Ortalama,
DATEDIFF(day, n.SinavTarihi, n.GirisTarihi) AS GirisHiziGun
FROM
    Notlar n
JOIN
    Kullanicilar k ON n.OgrenciId = k.Id
JOIN
    Dersler d ON n.DersId = d.Id
WHERE
    n.OgretmenId = @OgretmenId;

-- 3. Öğretmenin Anket Puanı Ortalaması
SELECT
    ISNULL(AVG(CAST(Puan AS DECIMAL(3,2))), 0) AS OrtalamaAnketPuanı
FROM
    AnketCevaplari
WHERE
    DegerlendirilenOgretmenId = @OgretmenId;
END;

```

**Veritabanında bulunan kullanıcı tanımlı veri tiplerinin (type) oluşturma SQL ifadelerini yazınız.**

```

SQL
-- Toplu not girişi gibi işlemler için kullanılabilen
-- bir kullanıcı tanımlı tablo tipi (User-Defined Table Type).
CREATE TYPE dbo.NotGirisListesi AS TABLE (
    OgrenciId INT NOT NULL,
    DersId INT NOT NULL,
    Vize INT NOT NULL,
    Final INT NOT NULL,
    SinavTarihi DATE NOT NULL
);

```

**Veritabanında bulunan indislerin (index) oluşturma SQL ifadelerini yazınız.**

```

SQL
-- 1. Kullanicilar tablosunda Email sütununa hızlı arama ve benzersizlik
-- (Unique) sağlamak için Index.
CREATE UNIQUE NONCLUSTERED INDEX IX_Kullanicilar_Email
ON Kullanicilar (Email);

-- 2. Performans raporlarında (VIEW ve PROCEDURE) OgretmenId'ye göre
-- çok sık filtreleme yapılacağı için Notlar tablosuna Index.
CREATE NONCLUSTERED INDEX IX_Notlar_OgretmenId
ON Notlar (OgretmenId);

-- 3. Raporlarda anket sonuçlarını hızlı çekmek için AnketCevaplari
-- tablosuna Index.
CREATE NONCLUSTERED INDEX IX_AnketCevaplari_OgretmenId
ON AnketCevaplari (DegerlendirilenOgretmenId);

```

Veritabanı diyagramı oluşturunuz ve resim olarak ekleyiniz  
(Aşağıdaki diyagram, oluşturduğumuz 4 tablo arasındaki ilişkileri göstermektedir.)

