T.C.

[üniversite] ÜNİVERSİTESİ

MESLEK YÜKSEKOKULU

BİLGİSAYAR PROGRAMCILIĞI

SİSTEM ANALİZİ ve TASARIMI DERSİ

DERSHANE OTOMASYONU PROGRAMI PROJESİ

Dersin Öğretim Görevlisi   
[öğretim-görevlisi]

EMRAH ÜNSAL

[öğrenci-no]

[şehir]-2020

**İÇİNDEKİLER**

[1. Giriş 3](#_Toc34667756)

[2. Veritabanı Tasarımı 3](#_Toc34667757)

[2.1. Gereklilik Analizi 3](#_Toc34667758)

[2.2. Kavramsal Tasarım 3](#_Toc34667759)

[2.3. Mantıksal Tasarım 7](#_Toc34667760)

[2.4. Fiziksel Tasarım 11](#_Toc34667761)

[2.4.1. ogrenci Tablosu: 12](#_Toc34667763)

[2.4.2. veli Tablosu: 13](#_Toc34667764)

[2.4.3. personel Tablosu: 14](#_Toc34667765)

[2.4.4. sinif Tablosu: 15](#_Toc34667766)

[2.4.5. program Tablosu: 15](#_Toc34667767)

[2.4.6. ders Tablosu: 16](#_Toc34667768)

[2.4.7. ders Tablosu: 17](#_Toc34667769)

[2.4.8. Veri Tabanı Diyagramı 18](#_Toc34667770)

[2.4.9. Tabloların SQL Kodları 18](#_Toc34667770)

# **1. Giriş**

Dershane otomasyonu programının amacı; dershaneler için bilgileri kayıt altına almak, öğrenci kaydı, veli ilişkilendirmesi, personel kaydı, ödeme kaydı gibi işlemleri yapmak ve tekrar düzenlenmesini sağlamak.

# **2. Veritabanı Tasarımı**

# **2.1. Gereklilik Analizi**

Program gereksinimleri ihtiyaçları yapılan araştırma sonucunda aşağıda belirtildiği şekilde tespit edilmiştir.

* Öğrenci bilgilerinin kayıt altına alınması,
* Veli bilgilerinin kayıt altına alınması,
* Personel bilgilerinin kayıt altına alınması,
* Ödeme bilgilerinin kayıt altına alınması,
* Sınıf bilgilerinin kayıt altına alınması,
* Program bilgilerinin kayıt altına alınması,
* Ders bilgilerinin kayıt altına alınması,

# **2.2. Kavramsal Tasarım**

Kavramsal tasarım aşağıdaki ihtiyaçların giderilmesi maksadıyla tasarlanmıştır.

* Öğrenci bilgilerinin anlaşılır bir şekilde listelenmesi.
* Öğrenci bilgilerinin kolay ve hızlı bir şekilde girilebilmesi.
* Öğrenci bilgilerinin kayıt altına alınabilmesi.
* Öğrenci dönem ücretinin işlenmesi.
* Veri güvenirliğinin sağlanması.
* Her türlü bilgiye ait ekleme, çıkarma, düzeltme ve silme güncelleme işlemlerinin yapılabilmesi.

Bu çerçevede; gereklilik analizinden de yola çıkılarak yapılan çalışma sonucunda aşağıdaki varlıkların planlanması gerekli bulunmuş planlama da buna yönelik olarak yapılmıştır. Buna göre Varlık Kümelerinin aşağıdaki maddelerden oluşması planlanmıştır.

* Öğrenci Varlık Kümesi,
* Veli Varlık Kümesi,
* Personel Varlık Kümesi,
* Sınıf Varlık Kümesi,
* Ders Varlık Kümesi,
* Program Varlık Kümesi,
* Ödeme Varlık Kümesi,

**2.2.1. Öğrenci Varlık Kümesi**

**Öğrenci** Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre **Öğrenci** Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

**Öğrenci** Varlık Kümesi;

* Id
* Sınıf Id
* Tc No
* Ad
* Soyad
* Cinsiyet
* Telefon
* Bölüm
* Dal
* Adres

nitelikleri olarak tasarlanmıştır. Tasarımda **Id** anahtar nitelik olarak tanımlanmıştır.

**2.2.2. Veli Varlık Kümesi**

**Veli** Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre **Veli** Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

**Veli** Varlık Kümesi;

* Id
* Öğrenci Id
* Tc No
* Ad
* Soyad
* Cinsiyet
* Telefon
* Eğitim
* Gelir
* Yakınlık

nitelikleri olarak tasarlanmıştır. Tasarımda **Id** anahtar nitelik olarak tanımlanmıştır.

**2.2.3. Personel Varlık Kümesi**

**Personel** Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre **Personel** Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

**Personel** Varlık Kümesi;

* Id
* Tc No
* Ad
* Soyad
* Cinsiyet
* Telefon
* Dal
* Maaş
* Adres

nitelikleri olarak tasarlanmıştır. Tasarımda **Id** anahtar nitelik olarak tanımlanmıştır.

**2.2.4. Sınıf Varlık Kümesi**

**Sınıf** Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre **Sınıf** Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

**Sınıf** Varlık Kümesi;

* Id
* Ad
* Blok

nitelikleri olarak tasarlanmıştır. Tasarımda **Id** anahtar nitelik olarak tanımlanmıştır.

**2.2.5. Program Varlık Kümesi**

**Program** Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre **Program** Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

**Program** Varlık Kümesi;

* Id
* Ders Id
* Sınıf Id
* Gün
* Başlangıç
* Bitiş

nitelikleri olarak tasarlanmıştır. Tasarımda **Id** anahtar nitelik olarak tanımlanmıştır.

**2.2.6. Ders Varlık Kümesi**

**Ders** Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre **Ders** Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

**Ders** Varlık Kümesi;

* Id
* Personel Id
* Ad

nitelikleri olarak tasarlanmıştır. Tasarımda **Id** anahtar nitelik olarak tanımlanmıştır.

**2.2.7. Ödeme Varlık Kümesi**

**Ödeme** Varlık Kümesi tasarlanırken bir bu varlık kümesine ait nitelikler diğer varlıklar arasındaki ilişkileri belirleyen nitelikler ile birlikte tasarlanmıştır.

Buna göre **Ödeme** Varlık Kümesi aşağıdaki şekilde oluşturulmuştur.

**Ödeme** Varlık Kümesi;

* Id
* Öğrenci Id
* Yıl
* Ücret

nitelikleri olarak tasarlanmıştır. Tasarımda **Id** anahtar nitelik olarak tanımlanmıştır.

# **2.3. Mantıksal Tasarım**

Dershane Otomasyonu veri tabanı tasarımında varlık kümeleri arasındaki ilişkilerin tasarlanmasından sonra mantıksal tasarım aşamasına geçilmiştir. Bu aşamada

**Öğrenci Tablosu;**

* Id **Birincil Anahtar** olarak tasarlanmıştır**, artan ve benzersiz.**
* Sınıf tablosuna bağlantı için Sınıf Id Yabancı Anahtar,
* Öğrencinin bir sınıfa ait olmasından dolayı her öğrencinin bir sınıfı vardır, Öğrencinin sınıfı Yabancı Anahtar olarak eklenmiştir.

|  |
| --- |
| ÖĞRENCİ TABLOSU |
| ID |
| SINIF ID |
| TC NO |
| AD |
| SOYAD |
| CİNSİYET |
| TELEFON |
| BÖLÜM |
| DAL |
| ADRES |

Bu durumda Öğreci Tablosu yandaki şekilde tasarlanmıştır.

Tablo 1: Öğrenci Tablosu

**Veli Tablosu;**

* Id **Birincil Anahtar** olarak tasarlanmıştır**, artan ve benzersiz.**

Öğrenci tablosuna bağlantı için Öğrenci Id Yabancı Anahtar,

|  |
| --- |
| VELİ TABLOSU |
| ID |
| ÖĞRENCİ ID |
| TC NO |
| AD |
| SOYAD |
| CİNSİYET |
| TELEFON |
| ADRES |
| EĞİTİM |
| GELİR |
| YAKINLIK |

Bu durumda Veli Tablosu yandaki şekilde tasarlanmıştır.

Tablo 2: Veli Tablosu

**Personel Tablosu;**

* Id **Birincil Anahtar** olarak tasarlanmıştır**, artan ve benzersiz.**

|  |
| --- |
| PERSONEL TABLOSU |
| ID |
| TC NO |
| AD |
| SOYAD |
| CİNSİYET |
| DAL |
| TELEFON |
| ADRES |
| MAAŞ |

Bu durumda Personel Tablosu yandaki şekilde tasarlanmıştır.

Tablo 3: Veli Tablosu

**Sınıf Tablosu;**

* Id **Birincil Anahtar** olarak tasarlanmıştır**, artan ve benzersiz.**
* Sınıfların konumu için blok bilgisi alanı eklenmiştir.

|  |
| --- |
| SINIF TABLOSU |
| ID |
| AD |
| BLOK |

Bu durumda Sınıf Tablosu yandaki şekilde tasarlanmıştır.

Tablo 4: Sınıf Tablosu

**Program Tablosu;**

* Id **Birincil Anahtar** olarak tasarlanmıştır**, artan ve benzersiz.**
* Her programın işlediği bir ders olduğundan Ders Id Yabancı Anahtar olarak belirlenmiştir.
* Her Programın gerçekleştirildiği bir sınıf olduğundan Sınıf Id Yabancı Anahtar olarak belirlenmiştir.

|  |
| --- |
| PROGRAM TABLOSU |
| ID |
| DERS ID |
| SINIF ID |
| GÜN |
| BAŞLANGIÇ |
| BİTİŞ |

Bu durumda Veli Tablosu yandaki şekilde tasarlanmıştır.

Tablo 5: Program Tablosu

**Ders Tablosu;**

* Id **Birincil Anahtar** olarak tasarlanmıştır**, artan ve benzersiz.**
* Dersleri veren bir öğretmen/personel oldğundan Personel Id Yabancı Anahtar olarak belirlenmiştir

|  |
| --- |
| DERS TABLOSU |
| ID |
| PERSONEL ID |
| AD |

Bu durumda Sınıf Tablosu yandaki şekilde tasarlanmıştır.

Tablo 6: Ders Tablosu

**Ödeme Tablosu;**

* Id **Birincil Anahtar** olarak tasarlanmıştır**, artan ve benzersiz.**
* Ödemenin bir öğrenci için yapılması durumu olduğu için Öğrenci Id Yabancı Anahtar olarak belirlenmiştir.

|  |
| --- |
| ÖDEME TABLOSU |
| ID |
| ÖĞRENCİ ID |
| YIL |
| ÜCRET |

Bu durumda Sınıf Tablosu yandaki şekilde tasarlanmıştır.

Tablo 7: Ödeme Tablosu

# **2.4. Fiziksel Tasarım**

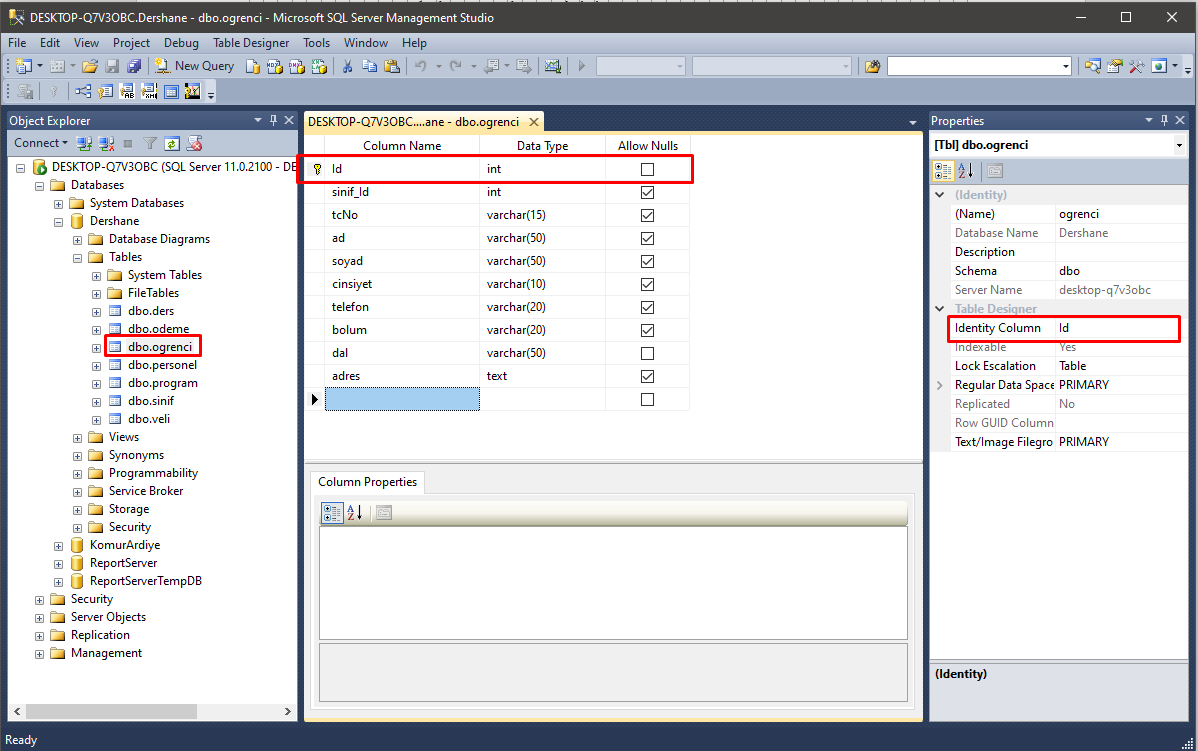
# Proje MSSQL programı ile tasarlanmıştır. Dershane tablosu tüm tabloların bağlı olduğu ana tablodur. Projede toplamda 7 adet tablo bulunmaktadır. Bulunan tablo adları aşağıda listelenmiştir.

* **ogrenci** – Öğrencilerin Bilgilerinin kayıt olduğu tablo adıdır.
* **veli** – Velilerin Bilgilerinin kayıt olduğu tablo adıdır.
* **personel** – Personel Bilgilerinin kayıt olduğu tablo adıdır.
* **sinif** – Sınıf Bilgilerinin kayıt olduğu tablo adıdır.
* **program** – Program Bilgilerinin kayıt olduğu tablo adıdır.
* **ders** – Ders Bilgilerinin kayıt olduğu tablo adıdır.
* **odeme** – Ödeme Bilgilerinin kayıt olduğu tablo adıdır.

### **2.4.1. ogrenci Tablosu:**

Öğrenci Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* Id, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* sinif\_Id, int veri tipi
* tcNo, varchar veri tipi
* ad, varchar veri tipi
* soyad, varchar veri tipi
* cinsiyet, varchar veri tipi
* telefon, varchar veri tipi
* bolum, varchar veri tipi
* dal, varchar veri tipi
* adres, text veri tipi

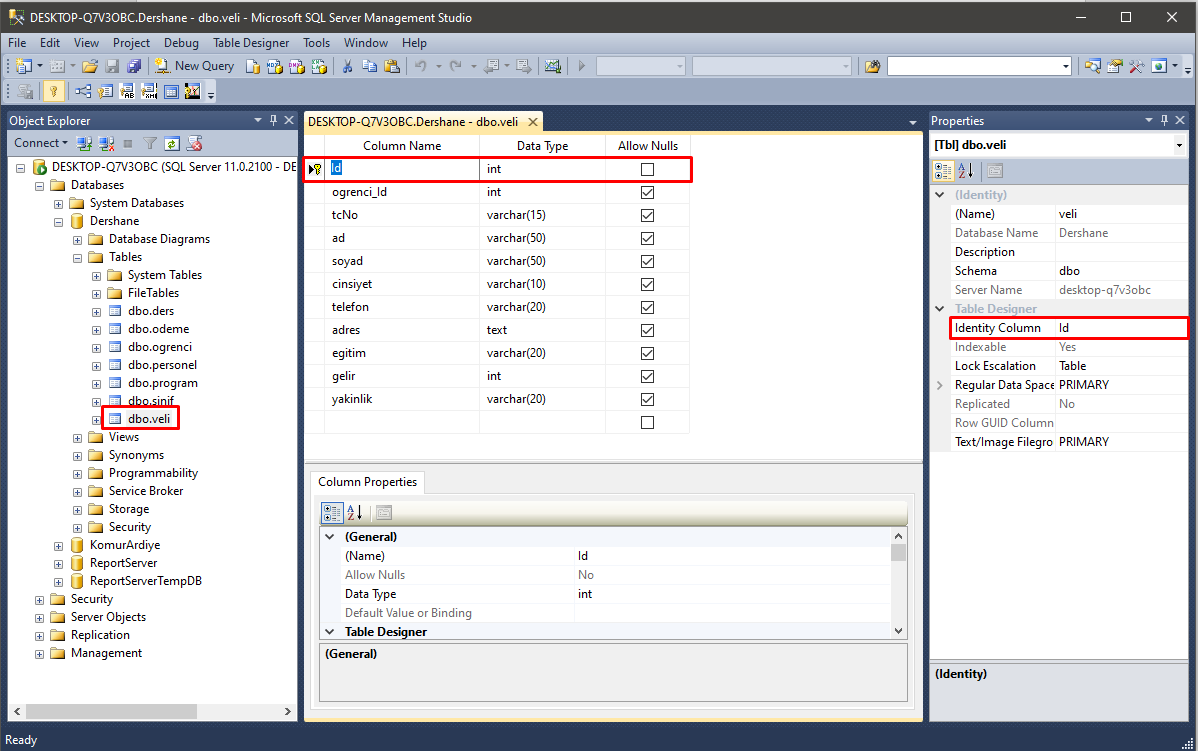


Tablo 1: ogrenci Tablosu

### **2.4.2. veli Tablosu:**

Veli Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* Id, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* ogrenci\_Id, int veri tipi
* tcNo, varchar veri tipi
* ad, varchar veri tipi
* soyad, varchar veri tipi
* cinsiyet, varchar veri tipi
* telefon, varchar veri tipi
* adres, text veri tipi
* egitim, varchar veri tipi
* gelir, varchar veri tipi
* yakinlik, varchar veri tipi

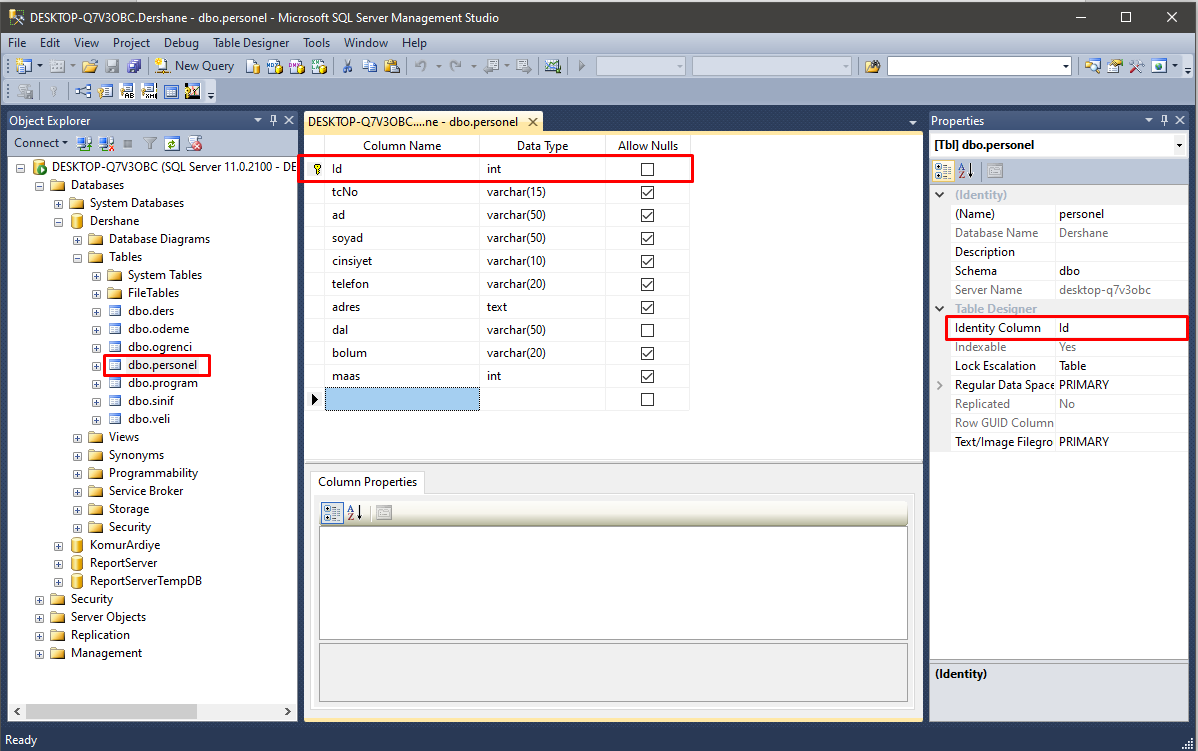


Tablo 2: veli Tablosu

### **2.4.3. personel Tablosu:**

Personel Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* Id, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* tcNo, varchar veri tipi
* ad, varchar veri tipi
* soyad, varchar veri tipi
* cinsiyet, varchar veri tipi
* dal, varchar veri tipi
* telefon, varchar veri tipi
* adres, text veri tipi
* maas, int veri tipi

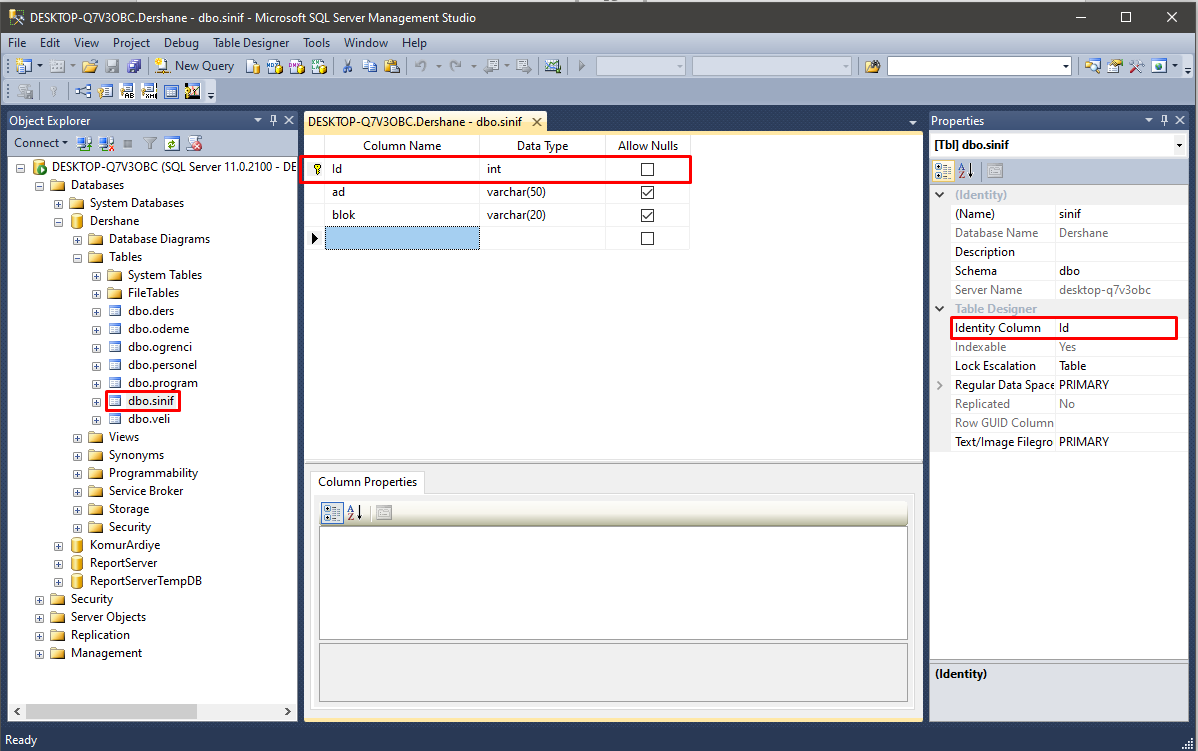


Tablo 3: personel Tablosu

### **2.4.4. sinif Tablosu:**

Sınıf Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* Id, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* ad, varchar veri tipi
* blok, varchar veri tipi

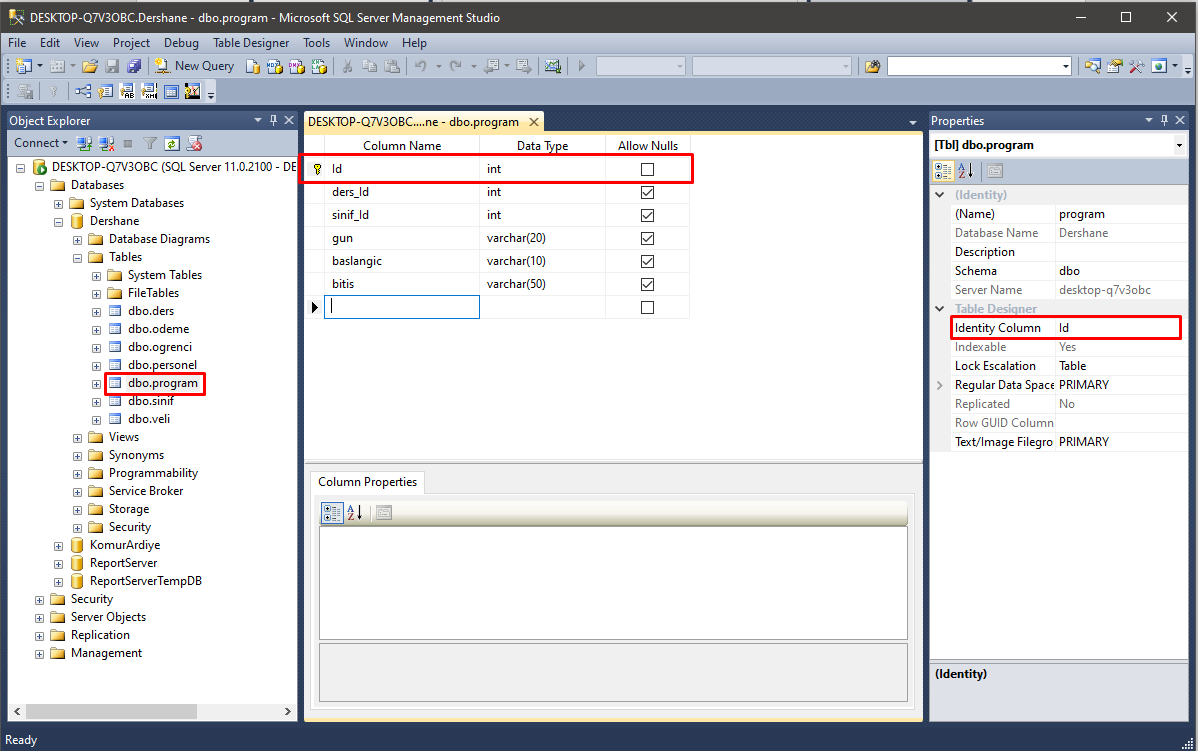


Tablo 4: sinif Tablosu

### **2.4.5. program Tablosu:**

Program Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* Id, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* ders\_Id, int veri tipi
* sinif\_Id, int veri tipi
* gun, varchar veri tipi
* baslangic, varchar veri tipi
* bitis, varchar veri tipi

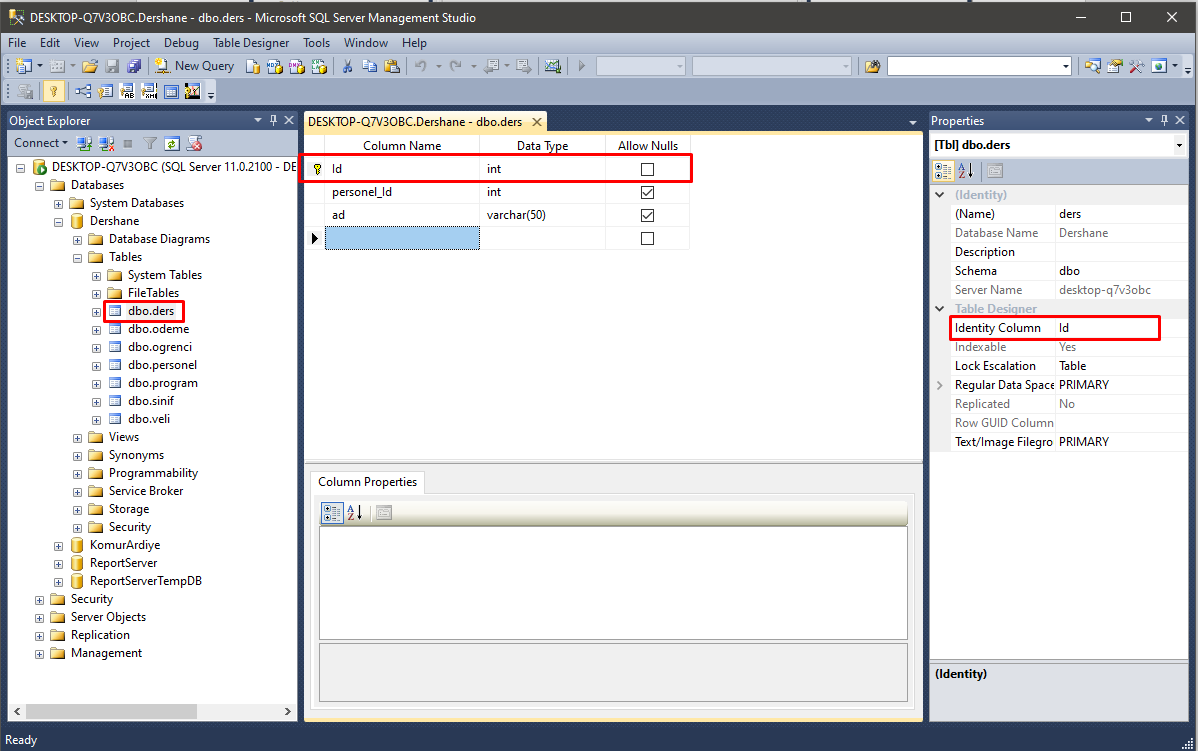


Tablo 5: program Tablosu

### **2.4.6. ders Tablosu:**

Ders Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

* Id, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* personel\_Id, int veri tipi
* ad, varchar veri tipi

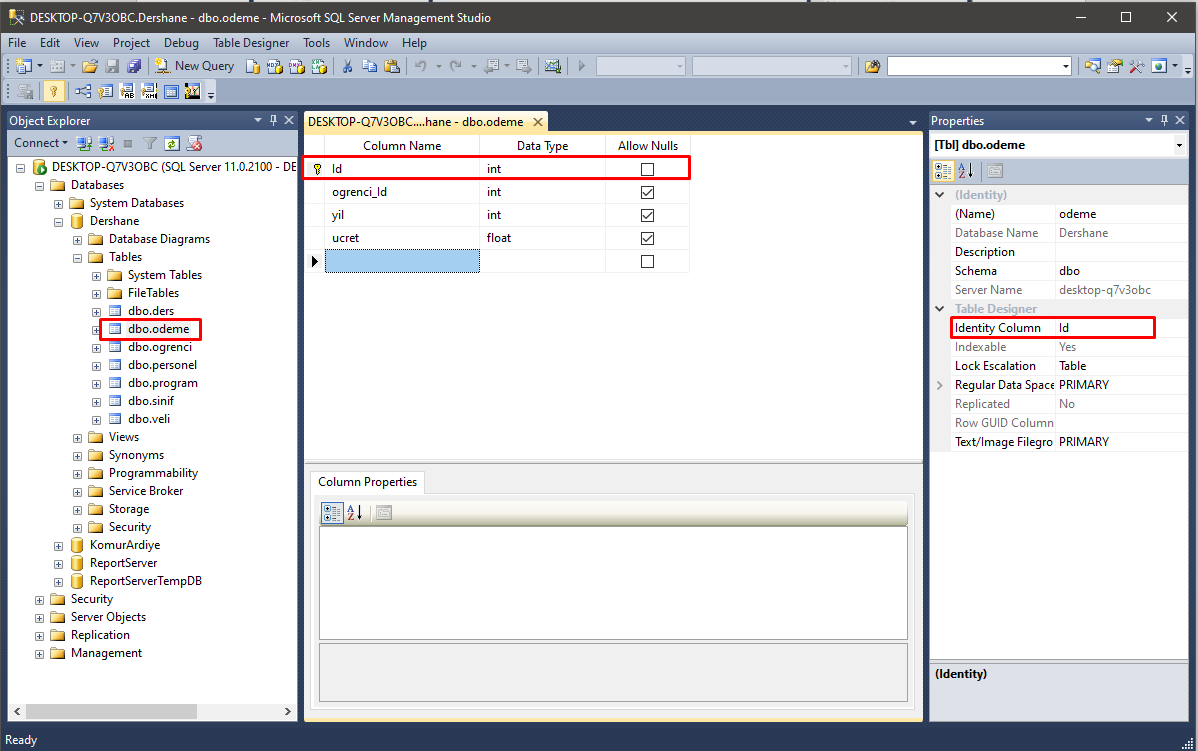


Tablo 6: ders Tablosu

### **2.4.7. ders Tablosu:**

Ders Tablosu Alanları değişkenleri aşağıda belirtildiği şekilde tespit edilmiştir.

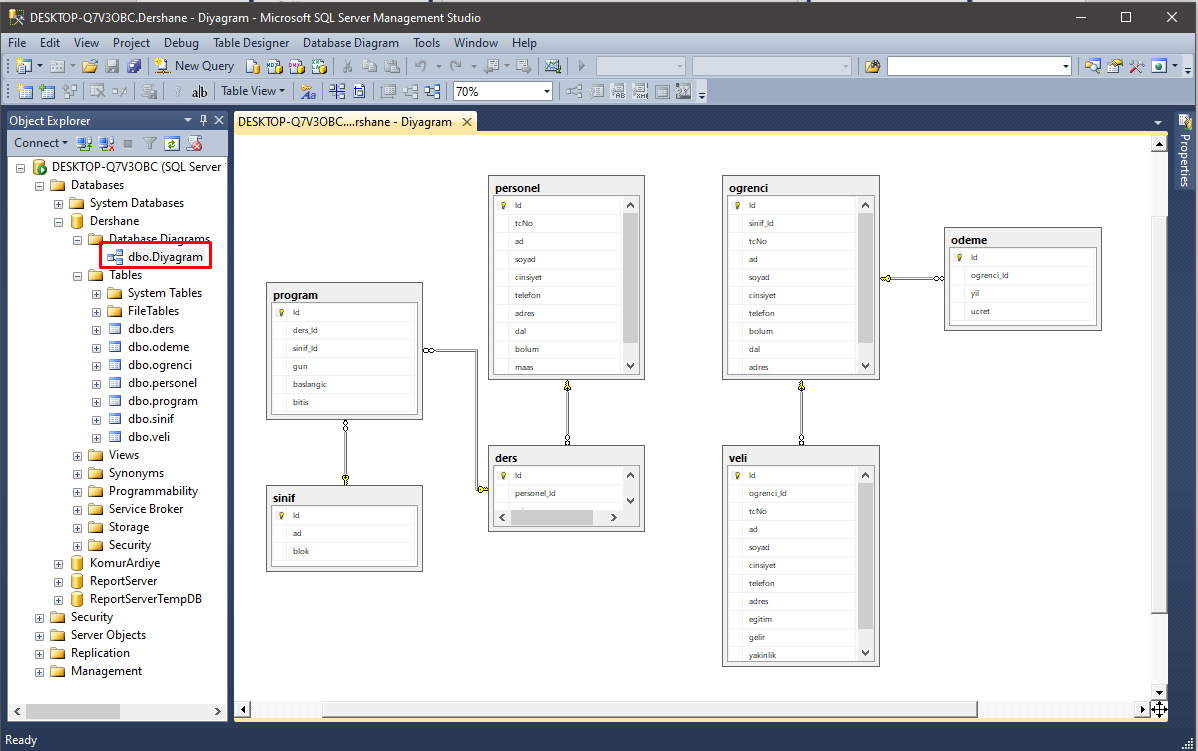
* Id, int veri tipi, Birincil Anahtar, Otomatik Artan ve Benzersiz
* ogrenci\_Id, int veri tipi
* yil, int veri tipi
* ucret, int veri tipi



Tablo 7: odeme Tablosu

### **2.4.8. Veri Tabanı Diyagramı**

Tablolar yukarıda belirtilen analizler ışığında aşağıda belirtildiği şekilde birbirleri ile ilişkilendirilmiş ve diyagram oluşturulmuştur.



Veritabanı Diyagramı

**2.4.9. Tabloların SQL Kodları**

Oluşturulan Tablolara ait SQL Kodları aşağıda sunulmuştur.

USE [Dershane]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[ders\_Ekle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[ders\_Ekle]

@personel\_Id int,

@ad varchar(50)

As

Begin

Insert Into ders (personel\_Id,ad) Values(@personel\_Id,@ad)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[ders\_Guncelle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[ders\_Guncelle]

@Id int,

@personel\_Id int,

@ad varchar(50)

As

Begin

Update ders set personel\_Id=@personel\_Id,ad=@ad

Where Id=@Id

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Getir] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[Getir]

@Tablo varchar(50),

@Id int

As

Begin

EXEC('Select \*From '+ @Tablo+' Where Id='+@Id)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Kayitlar] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[Kayitlar]

@Tablo varchar(50)

As

Begin

EXEC('Select \* From ' + @Tablo)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[odeme\_Ekle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[odeme\_Ekle]

@ogrenci\_Id int,

@yil int,

@ucret int

As

Begin

Insert Into odeme (ogrenci\_Id,yil,ucret) Values(@ogrenci\_Id,@yil,@ucret)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[odeme\_Guncelle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[odeme\_Guncelle]

@Id int,

@ogrenci\_Id int,

@yil int,

@ucret int

As

Begin

Update odeme set ogrenci\_Id=@ogrenci\_Id,yil=@yil,ucret=@ucret

Where Id=@Id

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[ogrenci\_Ekle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[ogrenci\_Ekle]

@sinif\_Id int,

@tcNo varchar(15),

@ad varchar(50),

@soyad varchar(50),

@cinsiyet varchar(10),

@telefon varchar(20),

@bolum varchar(20),

@dal varchar(50),

@adres text

As

Begin

Insert Into ogrenci(sinif\_Id,tcNo,ad,soyad,cinsiyet,telefon,bolum,dal,adres) Values(@sinif\_Id,@tcNo,@ad,@soyad,@cinsiyet,@telefon,@bolum,@dal,@adres)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[ogrenci\_Guncelle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[ogrenci\_Guncelle]

@Id int,

@sinif\_Id int,

@tcNo varchar(15),

@ad varchar(50),

@soyad varchar(50),

@cinsiyet varchar(10),

@telefon varchar(20),

@bolum varchar(20),

@dal varchar(50),

@adres text

As

Begin

Update ogrenci set sinif\_Id=@sinif\_Id,tcNo=@tcNo,ad=@ad,soyad=@soyad,cinsiyet=@cinsiyet,telefon=@telefon,bolum=@bolum,dal=@dal,adres=@adres

Where Id=@Id

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[personel\_Ekle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[personel\_Ekle]

@tcNo varchar(15),

@ad varchar(50),

@soyad varchar(50),

@cinsiyet varchar(10),

@telefon varchar(20),

@adres text,

@dal varchar(50),

@bolum varchar(50),

@maas int

As

Begin

Insert Into personel(tcNo,ad,soyad,cinsiyet,telefon,adres,dal,bolum,maas) Values(@tcNo,@ad,@soyad,@cinsiyet,@telefon,@adres,@dal,@bolum,@maas)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[personel\_Guncelle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[personel\_Guncelle]

@Id int,

@tcNo varchar(15),

@ad varchar(50),

@soyad varchar(50),

@cinsiyet varchar(10),

@telefon varchar(20),

@adres text,

@dal varchar(50),

@maas int,

@bolum varchar(20)

As

Begin

Update personel set tcNo=@tcNo,ad=@ad,soyad=@soyad,cinsiyet=@cinsiyet,telefon=@telefon,adres=@adres,dal=@dal,bolum=@bolum,maas=@maas

Where Id=@Id

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[program\_Ekle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[program\_Ekle]

@ders\_Id int,

@sinif\_Id int,

@gun varchar(20),

@baslangic varchar(10),

@bitis varchar(10)

As

Begin

Insert Into program (ders\_Id,sinif\_Id,gun,baslangic,bitis)

Values (@ders\_Id,@sinif\_Id,@gun,@baslangic,@bitis)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[program\_Guncelle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[program\_Guncelle]

@Id int,

@ders\_Id int,

@sinif\_Id int,

@gun varchar(20),

@baslangic varchar(10),

@bitis varchar(10)

As

Begin

Update program set ders\_Id=@ders\_Id,sinif\_Id=@sinif\_Id,gun=@gun,baslangic=@baslangic,bitis=@bitis

Where Id=@Id

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[Sil] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[Sil]

@Tablo varchar(50),

@Id int

As

Begin

EXEC('Delete From '+ @Tablo+' Where Id='+@Id)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sinif\_Ekle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[sinif\_Ekle]

@ad varchar(50),

@blok varchar(20)

As

Begin

Insert Into sinif (ad,blok) Values(@ad,@blok)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sinif\_Guncelle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[sinif\_Guncelle]

@Id int,

@ad varchar(50),

@blok varchar(20)

As

Begin

Update sinif set ad=@ad,blok=@blok

Where Id=@Id

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[veli\_Ekle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[veli\_Ekle]

@ogrenci\_Id int,

@tcNo varchar(15),

@ad varchar(50),

@soyad varchar(50),

@cinsiyet varchar(10),

@telefon varchar(20),

@adres text,

@egitim varchar(20),

@gelir int,

@yakinlik varchar(20)

As

Begin

Insert Into veli (ogrenci\_Id,tcNo,ad,soyad,cinsiyet,telefon,adres,egitim,gelir,yakinlik) Values(@ogrenci\_Id,@tcNo,@ad,@soyad,@cinsiyet,@telefon,@adres,@egitim,@gelir,@yakinlik)

End

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[veli\_Guncelle] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

Create Proc [dbo].[veli\_Guncelle]

@Id int,

@ogrenci\_Id int,

@tcNo varchar(15),

@ad varchar(50),

@soyad varchar(50),

@cinsiyet varchar(10),

@telefon varchar(20),

@adres text,

@egitim varchar(20),

@gelir int,

@yakinlik varchar(20)

As

Begin

Update veli set ogrenci\_Id=@ogrenci\_Id,tcNo=@tcNo,ad=@ad,soyad=@soyad,cinsiyet=@cinsiyet,telefon=@telefon,adres=@adres,egitim=@egitim,gelir=@gelir,yakinlik=@yakinlik

Where Id=@Id

End

GO

/\*\*\*\*\*\* Object: Table [dbo].[ders] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[ders](

[Id] [int] IDENTITY(1,1) NOT NULL,

[personel\_Id] [int] NULL,

[ad] [varchar](50) NULL,

CONSTRAINT [PK\_ders] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[odeme] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[odeme](

[Id] [int] IDENTITY(1,1) NOT NULL,

[ogrenci\_Id] [int] NULL,

[yil] [int] NULL,

[ucret] [float] NULL,

CONSTRAINT [PK\_odeme] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

/\*\*\*\*\*\* Object: Table [dbo].[ogrenci] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[ogrenci](

[Id] [int] IDENTITY(1,1) NOT NULL,

[sinif\_Id] [int] NULL,

[tcNo] [varchar](15) NULL,

[ad] [varchar](50) NULL,

[soyad] [varchar](50) NULL,

[cinsiyet] [varchar](10) NULL,

[telefon] [varchar](20) NULL,

[bolum] [varchar](20) NULL,

[dal] [varchar](50) NOT NULL,

[adres] [text] NULL,

CONSTRAINT [PK\_ogrenci\_1] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[personel] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[personel](

[Id] [int] IDENTITY(1,1) NOT NULL,

[tcNo] [varchar](15) NULL,

[ad] [varchar](50) NULL,

[soyad] [varchar](50) NULL,

[cinsiyet] [varchar](10) NULL,

[telefon] [varchar](20) NULL,

[adres] [text] NULL,

[dal] [varchar](50) NOT NULL,

[bolum] [varchar](20) NULL,

[maas] [int] NULL,

CONSTRAINT [PK\_personel] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[program] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[program](

[Id] [int] IDENTITY(1,1) NOT NULL,

[ders\_Id] [int] NULL,

[sinif\_Id] [int] NULL,

[gun] [varchar](20) NULL,

[baslangic] [varchar](10) NULL,

[bitis] [varchar](50) NULL,

CONSTRAINT [PK\_program] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[sinif] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[sinif](

[Id] [int] IDENTITY(1,1) NOT NULL,

[ad] [varchar](50) NULL,

[blok] [varchar](20) NULL,

CONSTRAINT [PK\_sinif] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

/\*\*\*\*\*\* Object: Table [dbo].[veli] Script Date: 9.03.2020 15:36:20 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[veli](

[Id] [int] IDENTITY(1,1) NOT NULL,

[ogrenci\_Id] [int] NULL,

[tcNo] [varchar](15) NULL,

[ad] [varchar](50) NULL,

[soyad] [varchar](50) NULL,

[cinsiyet] [varchar](10) NULL,

[telefon] [varchar](20) NULL,

[adres] [text] NULL,

[egitim] [varchar](20) NULL,

[gelir] [int] NULL,

[yakinlik] [varchar](20) NULL,

CONSTRAINT [PK\_ogrenci] PRIMARY KEY CLUSTERED

(

[Id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

ALTER TABLE [dbo].[ders] WITH CHECK ADD CONSTRAINT [FK\_ders\_personel] FOREIGN KEY([personel\_Id])

REFERENCES [dbo].[personel] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[ders] CHECK CONSTRAINT [FK\_ders\_personel]

GO

ALTER TABLE [dbo].[odeme] WITH CHECK ADD CONSTRAINT [FK\_odeme\_ogrenci] FOREIGN KEY([ogrenci\_Id])

REFERENCES [dbo].[ogrenci] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[odeme] CHECK CONSTRAINT [FK\_odeme\_ogrenci]

GO

ALTER TABLE [dbo].[program] WITH CHECK ADD CONSTRAINT [FK\_program\_ders] FOREIGN KEY([ders\_Id])

REFERENCES [dbo].[ders] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[program] CHECK CONSTRAINT [FK\_program\_ders]

GO

ALTER TABLE [dbo].[program] WITH CHECK ADD CONSTRAINT [FK\_program\_sinif] FOREIGN KEY([sinif\_Id])

REFERENCES [dbo].[sinif] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[program] CHECK CONSTRAINT [FK\_program\_sinif]

GO

ALTER TABLE [dbo].[veli] WITH CHECK ADD CONSTRAINT [FK\_veli\_ogrenci] FOREIGN KEY([ogrenci\_Id])

REFERENCES [dbo].[ogrenci] ([Id])

ON DELETE CASCADE

GO

ALTER TABLE [dbo].[veli] CHECK CONSTRAINT [FK\_veli\_ogrenci]

GO