

DDL (Veri Tanımlama Dili/Data Definition Language) Örnekler:

- 1) “*bolum*” adında bir tablo için; tablo sütunları sırasıyla, “*bolum_id*”, “*bolum_adi*”, “*aciklama*”, “*eposta*” ’dır. İlgili niteliklere(sütunlara) ait veri türleri aşağıdaki tabloda verilmiştir. Ayrıca *bolum_id* ve *bolum_adi* sütunlarıdaki etki alanı boş olamaz.(NOT NULL Constraint) kısıtlamasına sahiptir ve *bolum_id* sütunu **Primary Key**’dir. İlgili DDL cümlelerini yazınız.

bolum_id → int

bolum_adi → varchar(30)

aciklama → varchar(50)

eposta → varchar(30)

bolum

Column	Type	Default Value	Nullable
◇ <i>bolum_id</i>	int(5)		NO
◇ <i>bolum_adi</i>	varchar(30)		NO
◇ <i>aciklama</i>	varchar(50)		YES
◇ <i>eposta</i>	varchar(30)		YES

```
CREATE TABLE `bolum` (
  `bolum_id` int(5) NOT NULL,
  `bolum_adi` varchar(30) NOT NULL,
  `aciklama` varchar(50) DEFAULT NULL,
  `eposta` varchar(30) DEFAULT NULL,
  PRIMARY KEY (`bolum_id`)
)
```

- 2) “bolum” Tablosuna, aşağıda verilen görseldeki kayıtları ekleyen SQL cümlesini/cümlelerini yazınız.

bolum_id	bolum_adi	aciklama	eposta
1	Bil. Müh.	Bilgisayar Mühendisliği Bölümü	cem@marmara.edu.tr
2	Elo. Müh.	Elektronik Mühendisliği Bölümü	muhammed@marmara.edu.tr
3	Bil. Pro.	Bilgisayar Programcılığı Bölümü	akif@marmara.edu.tr
4	Elek. Tek.	Elektronik ve Haberleşme Teknolojisi Bölümü	selim@marmara.edu.tr

```
INSERT INTO `bolum` (`bolum_id`, `bolum_adi`, `aciklama`, `eposta`) VALUES  
(1, 'Bil. Müh.', 'Bilgisayar Mühendisliği Bölümü', 'cem@marmara.edu.tr'),  
(2, 'Elo. Müh.', 'Elektronik Mühendisliği Bölümü', 'muhammed@marmara.edu.tr'),  
(3, 'Bil. Pro.', 'Bilgisayar Programcılığı Bölümü', 'akif@marmara.edu.tr'),  
(4, 'Elek. Tek.', 'Elektronik ve Haberleşme Teknolojisi Bölümü', 'selim@marmara.edu.tr');
```

- 3) bolum tablosuna varchar(25) veri tipinde “bina” adlı sütunu ekleyiniz.

```
ALTER TABLE bolum ADD bina varchar(25);
```

- 4) bolum tablosundaki “bina” sütununu siliniz.

```
ALTER TABLE bolum DROP bina
```

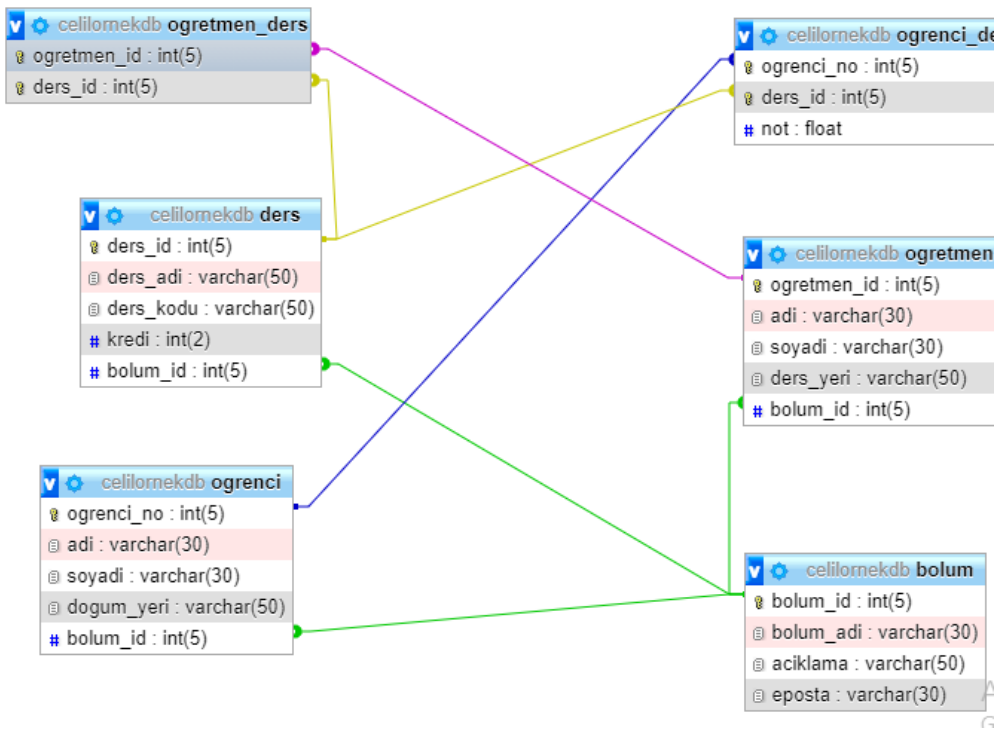
- 5) bolum tablosundaki bolum_adi sütununun veri türünü varchar(40) olarak güncelleyiniz.

```
ALTER TABLE bolum  
MODIFY bolum_adi varchar(40);
```

- 6) bolum tablosunu siliniz.

```
DROP TABLE bolum **!!**
```

Ornek Veritabanı Tabloları ve İlişkiler



Yukarıda gösterilen tabloların anlık kayıtlarının olduğu görüntüler aşağıdadır.

bolum Tablosu

bolum_id	bolum_adi	aciklama	eposta
1	Bil. Müh.	Bilgisayar Mühendisliği Bölümü	cem@marmara.edu.tr
2	Elo. Müh.	Elektronik Mühendisliği Bölümü	muhammed@marmara.edu.tr
3	Bil. Pro.	Bilgisayar Programcılığı Bölümü	akif@marmara.edu.tr
4	Elek. Tek.	Elektronik ve Haberleşme Teknolojisi Bölümü	selim@marmara.edu.tr
NULL	NULL	NULL	NULL

ders Tablosu

ders_id	ders_adi	ders_kodu	kredi	bolum_id
1	Veritabanı YS	CENG 351	3	1
2	İşletim Sistemleri	CENG 341	3	1
3	Nesne Tabanlı Programlama	CENG 101	5	1
4	Elektronığe Giriş	EE 101	2	2
5	İstatistik	IE 301	2	4
6	Bilgisayar Ağları	EE 202	4	2
7	Yazılım Mimari ve Tasarımı	CENG 102	3	3
8	Mobil Programlama	IE 208	3	4
9	Algoritma ve Programlamaya Giriş	CP 299	2	4
10	Ayrık Matematik	ENV 299	3	3
11	İnternet Programlama	CENG 299	5	1
12	Yaz Stajı	EE 299	1	2
13	Gömülü Sistemler	EE	3	1

ogrenci Tablosu

	ogrenci_no	adi	soyadi	dogum_yeri	bolum_id
▶	1	Ali	Arı	Eskişehir	1
	2	Ahmet	Sezgin	Ankara	1
	3	Leyla	Sezer	Kocaeli	1
	4	Cengiz	Türkoğlu	Manisa	2
	5	Aziz	Süleymanoğlu	İstanbul	2
	6	Taylan	Şanlı	Kars	3
	7	Kemal	İnce	Adana	3
	8	Özcan	Cemal	İzmir	4
	9	Öznur	Kurtoğlu	Bolu	2
	10	Pelin	Taş	Balıkesir	4
	11	Sevgi	Tan	İzmir	4

ogrenci_ders Tablosu

	ogrenci_no	ders_id	not	ogrenci_no	ders_id	not	ogrenci_no	ders_id	not
▶	1	1	3	2	11	4	5	3	NULL
	1	3	2.5	3	1	4	5	5	1.5
	1	4	3.5	3	2	4	5	11	3.5
	1	6	3	3	3	4	5	13	NULL
	1	9	4	3	4	4	6	2	4
	1	10	3	3	5	4	7	1	2.5
	2	1	4	3	6	4	7	2	3
	2	2	4	3	7	4	7	5	1.5
	2	3	4	3	8	4	7	8	1.5
	2	4	4	3	9	4	8	2	3.5
	2	5	4	3	10	3	8	7	1.5
	2	6	4	3	11	3.5	9	1	NULL
	2	7	4	4	1	2.5	10	2	4
	2	8	4	4	5	1.5	10	8	3
	2	9	4	5	1	3	11	8	1
	2	10	3	5	2	NULL	NULL	NULL	NULL

ogretmen Tablosu

ogretmen_id	adi	soyadi	dogum_yeri	bolum_id
1	Sezer	Can	Amasya	1
2	Cem	Taha	İstanbul	1
3	Deniz	Çimen	Mersin	1
4	Tuyan	Çağlar	İstanbul	2
5	Furkan	Çelen	İstanbul	3
6	Tahsin	Uğur	İzmir	4
7	Merve	Çağlar	Amasya	4
NULL	NULL	NULL	NULL	NULL

ogretmen_ders Tablosu

ogretmen_id	ders_id
1	1
3	2
2	3
4	4
7	5
4	6
5	7
6	8
7	9
5	10
1	11
4	12
NULL	NULL

Aşağıda verilen sorgular ve sonuçlar için yukarıda kayıtlar ve görselleri bulunan tabloları kullanınız. Tablo ve kayıtlar için [tıklayınız](#)

DML(Data Manipulation Language/Veri Manipulasyon Dili) Örnekler:

Aşağıdaki SQL ifadelerini yazınız.

- 1) “ders” Tablosunda bulunan tüm kayıtları listeleyin.

```
SELECT * FROM ders
```

- 2) “ogretmen” Tablosunda bulunan kayıtlar içinden, doğum yeri “İstanbul” olan kayıtları listeleyin.

```
SELECT * FROM ogretmen WHERE dogum_yeri="İstanbul"
```

- 3) “ogrenci” Tablosunda bulunan kayıtlar içinden, doğum yeri “İzmir” olan kayıtların sadece ad ve soyad bilgisini listeleyin.

```
SELECT adi,soyadi FROM ogrenci WHERE dogum_yeri="İzmir"
```

- 4) “ogretmen” Tablosunda bulunan kayıtlar içinden, adında “er” ifadesi geçen kayıtların sadece ad bilgisini listeleyin.

```
SELECT adi FROM ogretmen where adi LIKE '%er%'
```

- 5) “ogrenci” Tablosunda bulunan kayıtlar içinden soyadı T ile başlayan kayıtların sadece doğum yeri bilgisini listeleyiniz.

```
SELECT dogum_yeri FROM ogrenci WHERE soyadi LIKE 'T%'
```

- 6) “ogrenci” Tablosunda soyadı 3 karakter olan kayıtları listeleyiniz.

```
SELECT * FROM ogrenci WHERE soyadi LIKE '___'
```

- 7) “ogrenci_ders” Tablosunda kredisi 3 ve 5 arasında olan kayıtları listeleyiniz.

```
SELECT * FROM ders WHERE kredi BETWEEN 3 and 5
```

- 8) “ogrenci” Tablosunda kişilerin adlarına göre alfabetik olarak sıralanmış şekilde kayıtları listeleyiniz.

```
SELECT * FROM ogrenci ORDER BY adi
```

- 9) “ders” Tablosunda bulunan dersleri yüksek kredili dersten, düşük kredili derse göre sıralayın.

```
SELECT * FROM ders ORDER BY kredi DESC
```

- 10) Bilgisayar mühendisliği bölümü altında verilen dersleri listeleyiniz.(“ders” ve “bolum” tablolarını kullanınız.)

```
SELECT * FROM ders natural join bolum where bolum_adi="Bil. Müh."
```

```
SELECT * FROM ders join bolum using(bolum_id) where bolum_adi="Bil. Müh."
```

- 11) Bilgisayar Mühendisliği bölümünde ders veren öğretmenleri listeleyiniz.(“ogretmen” ve “bolum” tablolarını kullanınız.)

```
SELECT * FROM ogretmen join bolum on ogretmen.bolum_id = bolum.bolum_id where bolum_adi = "Bil. Müh."
```

- 12) Öğretmenlerin görevlendirildikleri dersleri listeleyiniz.(ogretmen,ogretmen_ders ve ders tablolarını kullanınız.)

```
SELECT adi,soyadi,ders_adi FROM ogretmen join ogretmen_ders using(ogretmen_id) join ders using(ders_id)
```

- 13) “dersler” Tablosunda bulunan derslerin kredi ortalamasını bulunuz.

```
SELECT AVG(kredi) AS Ortalama FROM ders
```

- 14) Veritabanı YS dersini alan öğrenci sayısını bulunuz.(ders,öğrenci_ders,ogrenci)

```
SELECT COUNT(adi) FROM ders join ogrenci_ders using(ders_id) join ogrenci using(ogrenci_no) where ders_adi="Veritabanı YS"
```

- 15) Öğrencin aldığı dersleri ve ilgili derslerin notlarını listeleyin.

```
SELECT * FROM ogrenci_ders join ogrenci using(ogrenci_no) join ders using(ders_id)
```

- 16) Öğrenci adi, soyadi, ders adi, ders kodu ve ders kredi bilgilerinin olduğu, Öğrenci ve ders bilgilerinin listelendiği “**ders_ogrenci_listesi**” adında bir VIEW oluşturunuz.

```
CREATE VIEW ogrenci_ders_listesi AS SELECT adi,soyadi,ders_adi,ders_kodu,kredi FROM ogrenci_ders join ogrenci using(ogrenci_no) join ders using(ders_id)
```

- 17) Oluşturmuş olduğunuz “**ders_ogrenci_listesi**” adlı VIEW’i görüntüleyiniz.

```
SELECT * FROM ogrenci_ders_listesi
```

- 18) “ders” Tablosunda bulunan dersleri yüksek kredili dersten, düşük kredili derse göre sıralayan sorgu için “**sirali_ders_kredi_listesi**” adında bir View tanımlayınız.

```
CREATE VIEW sirali_ders_kredi_listesi AS SELECT * FROM ders ORDER BY kredi DESC
```

- 19) “**sirali_ders_kredi_listesi**” adlı View’i siliniz.

```
DROP VIEW sirali_ders_kredi_listesi
```

- 20) Öğrenci numarası 15 olan kaydı siliniz.

```
SET SQL_SAFE_UPDATES=0; /***(Eğer silme izin hatası alırsanız bu satırı ekleyin.)***
```

```
SET FOREIGN_KEY_CHECKS=0; /*Error code 1451 hatası dersteki hata çözümü*/
```

```
DELETE FROM ogrenci WHERE ogrenci_no =11
```

- 21) Her öğrencinin aldığı derslerin kredi toplamalarını bulunuz.(JOIN,SUM,GROUP BY)

```
SELECT adi,soyadi,SUM(kredi) AS Toplam_Kredi FROM (SELECT ogrenci_no,adi,soyadi,kredi FROM ogrenci JOIN ogrenci_ders USING(ogrenci_no) JOIN ders USING(ders_id)) AS Liste GROUP BY Liste.ogrenci_no
```