

T.C.
ATATÜRK ÜNİVERSİTESİ
İKTİSADİ VE İDARİ BİLİMLER FAKÜLTESİ
YÖNETİM BİLİŞİM SİSTEMLERİ BÖLÜMÜ

Üniversite Bilgi Sistemi View, Stored Procedure, Table ve Scalar Fonksiyon
Sorguları

Hazırlayan
İbrahim Ethem RONA
200307063

Dersin Öğretim Üyesi
Dr. Öğr. Üyesi Muhammed Fatih ALAEDDİNOĞLU

Erzurum 2022

VIEW

1. Fakülteleri listeleme

```
CREATE VIEW vwFakulteListesi AS
SELECT fakulteID, fakulteAdi FROM fakulte
```

2. Bölüm ders

```
CREATE VIEW vwbolumDersListesi AS
SELECT b.bolumAdi, d.dersAdi FROM bolum_ders bd
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
```

3. Öğrenci ders

```
CREATE VIEW vwogrenciDers AS
SELECT k.ad,k.soyad,b.bolumAdi, d.dersAdi FROM ogrenci_ders od
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID = od.bolum_ogrenciID
INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiID = o.kisiID
```

4. Bölümün hocaları

```
CREATE VIEW vwbolumHocalariListesi
SELECT b.bolumAdi,u.unvan,k.ad,k.soyad FROM bolum_hoca bh
INNER JOIN hoca_unvan hu ON hu.hoca_unvanID = bh.hoca_unvanID
INNER JOIN bolum b ON b.bolumID = bh.bolumID
INNER JOIN hoca h ON h.hocaID = hu.hocaID
INNER JOIN unvan u ON u.unvanID = hu.unvanID
```

INNER JOIN kisi k ON k.kisiID = h.kisiID

5. Bölümün öğrencileri

CREATE VIEW vwBolumOgrencileriListesi

SELECT b.bolumAdi,k.ad,k.soyad FROM bolum_ogrenci bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

6. Öğrenciye ait sınav listesi

CREATE VIEW vwOgrenciSinavListesi

SELECT k.ad,k.soyad,d.dersAdi,d.kredi,st.sinavTuru,st.etkiOrani,os.sinavNotu
FROM ogrenci_sinav os

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = os.ogrenci_dersID

INNER JOIN sinav s ON s.sinavID = os.sinavID

INNER JOIN sinav_tur st ON st.sinav_turID = s.sinav_turID

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID = od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

7. Fakülte öğrencileri

CREATE VIEW vwFakulteOgrenciListesi

SELECT f.fakulteAdi,b.bolumAdi,k.ad,k.soyad FROM bolum_ogrenci bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN fakulte f ON f.fakulteID = b.fakulteID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

8. Bölüm bazlı ders programı listesi

CREATE VIEW vwBolumDersProgramiListesi AS

SELECT

b.bolumAdi,d.dersAdi,dp.bASlangicSaat,bitisSaat,os.ogretimSekli,u.unvan,k.ad,k.soyad,ds.derslik FROM dersprogrami dp

INNER JOIN bolum_ders bd ON bd.bolum_dersID = dp.bolum_dersID

INNER JOIN ogretimsekli os ON os.ogretimSekliID = dp.ogretimSekliID

INNER JOIN hoca_unvan hu ON hu.hoca_unvanID=dp.hoca_unvanID

INNER JOIN derslik ds ON ds.derslikID = dp.derslikID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN hoca h ON h.hocaID = hu.hocaID

INNER JOIN kisi k ON k.kisiID = h.kisiID

INNER JOIN unvan u ON u.unvanID = hu.unvanID

9. Hocalar ve unvanların listesi

CREATE VIEW vwHocaUnvanListesi AS

SELECT u.unvan,k.ad,k.soyad FROM hoca_unvan hu

INNER JOIN hoca h ON h.hocaID = hu.hocaID

INNER JOIN kisi k ON k.kisiID = h.kisiID

INNER JOIN unvan u ON u.unvanID = hu.unvanID

10. Bölüm bazlı öğrenci iletişim listesi

CREATE VIEW vwBolumOgrenciIletisimListesi AS

SELECT b.bolumAdi,o.ogrenciID, k.ad,k.soyad, i.iletisim, ilt.iletisimTur FROM bolum_ogrenci bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

INNER JOIN iletisim i ON i.iletisimID = o.iletisimID

INNER JOIN iletisimturu ilt ON ilt.iletisimturID = i.iletisimturID

11. Bina listesi

CREATE VIEW vwbinaListesi AS

SELECT b.binaAdi,b.binaAdresi FROM bina b

STORED PROCEDURE

1. Öğrenci Id ye göre Öğrencinin aldığı dersler

DELIMITER \$

CREATE PROCEDURE pd_ogrencialdigiders(IN ogrenciID INT)

BEGIN

SELECT o.ogrenciID,k.ad,k.soyad,d.dersAdi,d.kredi FROM ogrenci_ders od

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID =
od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

WHERE o.ogrenciID = ogrenciID ;

END \$

DELIMITER ;

call pd_ogrencialdigiders(2)

3. Öğrenci Id ye göre Öğrencinin iletişim bilgileri listesi

DELIMITER \$

CREATE PROCEDURE pd_ogrenciIletisimBilgileri(IN ogrenciID INT)

BEGIN

SELECT k.ad,k.soyad,i.iletisim,ilt.iletisimTur FROM bolum_ogrenci bo
INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiID = o.kisiID
INNER JOIN iletisim i ON i.iletisimID=o.iletisimID
INNER JOIN iletisimturu ilt ON ilt.iletisimTurID = i.iletisimTurID
WHERE o.ogrenciID=ogrenciID;

END \$

DELIMITER ;

call pd_ogrenciIletisimBilgileri(1)

4. Öğrenci Id ye göre öğrencinin kaldığı dersler

CASE yapısı ile öğrenci harf notu hesaplandı.

UPDATE ogrenci_snavsonucu oss

INNER JOIN ogrenci_snav os ON os.ogrenci_snavID

INNER JOIN ders_snav ds ON ds.ders_snavID = os.ders_snavID

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID

INNER JOIN snav s ON s.snavID=ds.snavID

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID= od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiId = o.kisiId

```

SET oss.harfNot = CASE WHEN oss.ort>=84 AND oss.ort<=100 THEN "AA"
WHEN oss.ort>=71 AND oss.ort<=83.99 THEN "BA"
WHEN oss.ort>=61 AND oss.ort<=70 THEN "BB" WHEN oss.ort>=54 AND
oss.ort<=60.99 THEN "CB" WHEN oss.ort>=45 AND oss.ort<=53.99 THEN "CC"
WHEN oss.ort>=38 AND oss.ort<=44.99 THEN "DD" WHEN oss.ort>=30 AND
oss.ort<=38.99 THEN "DC" ELSE "FF"
END;

```

```

CREATE PROCEDURE pd_ogrencininKaldigiDers(IN ogrenciID INT)
BEGIN
    SELECT DISTINCT k.ad,k.soyad,d.dersAdi FROM ogrenci_sinavsonucu oss
    INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID
    INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID
    INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
    INNER JOIN sinav s ON s.sinavID=ds.sinavID
    INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
    od.bolum_ogrenciID
    INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
    INNER JOIN bolum b ON b.bolumID = bd.bolumID
    INNER JOIN ders d ON d.dersID = bd.dersID
    INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
    INNER JOIN kisi k ON k.kisiId = o.kisiId
    WHERE o.ogrenciID = ogrenciID AND oss.harfNot= "DC" OR
    oss.harfNot="FF";
END $
DELIMITER ;
call pd_ogrencininKaldigiDers(1)

```

5. Öğrenci Id ye göre öğrencinin geçtiği derslerin kredi toplamları

```

DELIMITER $

```

```
CREATE PROCEDURE pd_ogrencininGectigiDersinToplamKredisi(IN ogrenciID INT)
```

```
BEGIN
```

```
SELECT k.ad,k.soyad, CASE WHEN oss.harfNot= "AA" OR oss.harfNot =  
"BA" OR oss.harfNot = "BB" OR oss.harfNot = "CD" OR oss.harfNot = "CC"  
THEN SUM(DISTINCT d.kredi) END AS toplam_kredi FROM  
ogrenci_sinavsonucu oss
```

```
INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID
```

```
INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID
```

```
INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
```

```
INNER JOIN sinav s ON s.sinavID=ds.sinavID
```

```
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=  
od.bolum_ogrenciID
```

```
INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
```

```
INNER JOIN bolum b ON b.bolumID = bd.bolumID
```

```
INNER JOIN ders d ON d.dersID = bd.dersID
```

```
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
```

```
INNER JOIN kisi k ON k.kisiID = o.kisiID
```

```
WHERE o.ogrenciID = ogrenciID ;
```

```
END $
```

```
DELIMITER ;
```

```
call pd_ogrencininGectigiDersinToplamKredisi(1)
```

6. Öğrenci Id ye göre Öğrencinin sisteme attığı mesajların listesi

```
DELIMITER $
```

```
CREATE PROCEDURE pd_ogrenciSistemMesaji(IN ogrenciID INT)
```

```
BEGIN
```

```
SELECT o.ogrenciID,k.ad, k.soyad, ilt.iletisimTur, m.mesaj FROM  
ogrenci_mesaj om
```

```
INNER JOIN ogrenci o ON o.ogrenciID = om.ogrenciID
```

```
INNER JOIN mesaj m ON m.mesajID = om.mesajID
```



```

INNER JOIN kisi k ON k.kisiID = o.kisiID
INNER JOIN iletisim i ON i.iletisimID = o.iletisimID
INNER JOIN iletisimturu ilt ON ilt.iletisimturuID = i.iletisimturuID
WHERE o.ogrenciID = ogrenciID ;
END $
DELIMITER ;
call pd_ogrenciSistemMesaji(1)

```

8. Öğrenci Id ye göre Öğrencinin nüfus bilgileri

```

DELIMITER $
CREATE PROCEDURE pd_ogrenciNufus(IN ogrenciID INT)
BEGIN
    SELECT
    k.TCKimlikNo,k.ad,k.soyad,k.dogumTarihi,k.anneAdi,k.babaAdi,k.adres,k.me
    deniHali,k.cinsiyet,i.ilAdi,ilc.ilceAdi FROM ogrenci o
    INNER JOIN kisi k ON k.kisiID = o.kisiID
    INNER JOIN ilce ilc ON ilc.ilceID = k.ilceID
    INNER JOIN il i ON i.ilID = ilc.ilID
    WHERE o.ogrenciID = ogrenciID ;
END $
DELIMITER ;
call pd_ogrenciNufus(2)

```

9. Ders Id ye göre her bir dersle alakalı öğrencinin ders devam durumu

```

DELIMITER $
CREATE PROCEDURE pd_ogrenciDevamDurumuListesi(IN ogrenciID INT)
BEGIN
    SELECT k.ad, k.soyad,d.dersAdi,dd.devamDurumu FROM yoklama y
    INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = y.ogrenci_dersID

```

```

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID =
od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN devamdurumu dd ON dd.devamDurumuID = y.devamDurumuID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

WHERE o.ogrenciID = ogrenciID ;

END $

DELIMITER ;

call pd_ogrenciDevamDurumuListesi(2)

```

10. Öğrenci Id ye göre Öğrencinin sınav sonuçları

```

UPDATE ogrenci_sinavsonucu oss

INNER JOIN ders_sinav dsVize ON dsVize.ogrenci_dersID = oss.ogrenci_dersID

INNER JOIN ders_sinav dsFinal ON dsFinal.ogrenci_dersID = oss.ogrenci_dersID

INNER JOIN ogrenci_sinav osVize ON osVize.ders_sinavID= dsVize.ders_sinavID

INNER JOIN ogrenci_sinav osFinal ON osFinal.ders_sinavID = dsFinal.ders_sinavID

INNER JOIN sinav sVize ON sVize.sinavID = dsVize.sinavID AND sVize.sinavID =
1

INNER JOIN sinav sFinal ON sFinal.sinavID = dsFinal.sinavID AND sFinal.sinavID
= 2

SET oss.ort = sVize.etkiOrani*osVize.sinavNotu/100 +
sFinal.etkiOrani*osFinal.sinavNotu/100

DELIMITER $

CREATE PROCEDURE pd_ogrenciIDSinavSonuclari(IN ogrenciID INT)

BEGIN

```

```

SELECT DISTINCT
k.ad,k.soyad,d.dersAdi,d.kredi,st.sinavTuru,s.etkiOrani,os.sinavNotu,oss.ort
FROM ogrenci_sinavsonucu oss

INNER JOIN ogrenci_sinav os ON os.ders_sinavID

INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = oss.ogrenci_dersID

INNER JOIN sinav s ON s.sinavID = ds.sinavID

INNER JOIN sinav_tur st ON st.sinav_turID = s.sinav_turID

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID =
od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN kisi k ON k.kisiID = o.kisiID

WHERE o.ogrenciID = ogrenciID;

END $

DELIMITER ;

call pd_ogrenciIDSinavSonuclari(1)

```

11. Öğrenci Id ye göre öğrencinin son iki dönem agnosu

```

DELIMITER $

CREATE PROCEDURE pd_ogrenciSonIkiDonemAgnosu(IN ogrenciID INT)
BEGIN

SELECT DISTINCT gbd.guz_bahardonem,k.ad,k.soyad,da.agno FROM
donem_agno da

INNER JOIN ogrenci_sinavsonucu oss ON oss.ogrenci_sinavSonucID =
da.ogrenci_sinavSonucID

INNER JOIN guz_bahardonem gbd ON gbd.guzBaharID = da.guzBaharID

INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID

INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID

```

```

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
INNER JOIN sinav s ON s.sinavID=ds.sinavID
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
od.bolum_ogrenciID
INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiId = o.kisiId
WHERE o.ogrenciID = ogrenciID;
END $
DELIMITER ;
call pd_ogrenciSonIlkDonemAgnosu(1)

```

12. Öğrenci Id ye göre öğrencinin son agnosu

Harf not katsayısı belirlendi.

```

UPDATE ogrenci_sinavsonucu oss
INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID
INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID
INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
INNER JOIN sinav s ON s.sinavID=ds.sinavID
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID= od.bolum_ogrenciID
INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiId = o.kisiId

```

```

SET oss.harfNotKatsayisi = CASE WHEN oss.harfNot= "AA" THEN 4 WHEN
oss.harfNot= "BA" THEN 3.5
WHEN oss.harfNot= "BB" THEN 3 WHEN oss.harfNot= "CB" THEN 2.5 WHEN
oss.harfNot= "CC" THEN 2
WHEN oss.harfNot= "DC" THEN 1.5 WHEN oss.harfNot= "DD" THEN 1 ELSE 0
END;
WHERE ogrenci_sinavSonucuID = 1

```

Agno hesabı yapıldı.

```

UPDATE donem_agno da
INNER JOIN ogrenci_sinavsonucu oss ON oss.ogrenci_sinavSonucID =
da.ogrenci_sinavSonucID
INNER JOIN guz_bahardonem gbd ON gbd.guzBaharID = da.guzBaharID
INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID
INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID
INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
INNER JOIN sinav s ON s.sinavID=ds.sinavID
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID= od.bolum_ogrenciID
INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiId = o.kisiId
SET da.agno = (SELECT SUM(DISTINCT d.kredi * oss.harfNotKatsayisi) /
SUM(DISTINCT d.kredi) FROM ogrenci_sinavsonucu )
WHERE donem_agnoID = 1

DELIMITER $

CREATE PROCEDURE pd_ogrenciSonAgnosu(IN ogrenciID INT)

```

BEGIN

```
SELECT DISTINCT k.ad,k.soyad,da.agno FROM donem_agno da
INNER JOIN ogrenci_sinavsonucu oss ON oss.ogrenci_sinavSonucID =
da.ogrenci_sinavSonucID
INNER JOIN guz_bahardonem gbd ON gbd.guzBaharID = da.guzBaharID
INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID
INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID
INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
INNER JOIN sinav s ON s.sinavID=ds.sinavID
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
od.bolum_ogrenciID
INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiID = o.kisiID
```

WHERE o.ogrenciID = ogrenciID;

END \$

DELIMITER ;

call pd_ogrenciSonAgnosu(1)

13. Bölüm Id ye göre Bölümde ders bazlı kalan öğrenci listesi ve sayısı

DELIMITER \$

```
CREATE PROCEDURE pd_bolumBazliKalanOgrenciListesiVeSayisi(IN bolumID
INT)
```

BEGIN

```
SELECT d.dersAdi,k.ad,k.soyad,COUNT(DISTINCT o.ogrenciID) AS
kalanOgrenciSayisi FROM ogrenci_sinavsonucu oss
INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID
INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID
```

```

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
INNER JOIN sinav s ON s.sinavID=ds.sinavID
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
od.bolum_ogrenciID
INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
        INNER JOIN ders d ON d.dersID = bd.dersID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiId = o.kisiId
WHERE b.bolumID=bolumID AND oss.harfNot = "FF";
END $
DELIMITER ;
call pd_bolumBazliKalanOgrenciListesiVeSayisi(1)

```

14. Fakülte Id ye göre Fakültenin ortak ders listesi

```

DELIMITER $
CREATE PROCEDURE pd_fakulteOrtakDersListesi(IN fakulteID INT)
BEGIN
    SELECT d.dersID, IF(d.durumu="FS", d.dersAdi, NULL) AS "Fakülte Dersi"
    FROM fakulte_ders fd
    INNER JOIN bolum_ders bd ON bd.bolum_dersID= fd.bolum_dersID
    INNER JOIN bolum b ON b.bolumID = bd.bolumID
    INNER JOIN fakulte f ON f.fakulteID = b.fakulteID
    INNER JOIN ders d ON d.dersID=bd.dersID
    WHERE f.fakulteID = fakulteID;
END $
DELIMITER ;
call pd_fakulteOrtakDersListesi(1)

```

15. Fakültenin her bir bölümündeki öğrenci sayısı listesi

DELIMITER \$

CREATE PROCEDURE pd_fakulteBolumOgrenciSayisi(IN fakulteID int)

BEGIN

SELECT b.bolumAdi,COUNT(*) AS "öğrenci sayisi" FROM bolum_ogrenci
bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID=bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN fakulte f ON f.fakulteID=b.fakulteID

INNER JOIN kisi k ON k.kisiID = o.kisiID

WHERE f.fakulteID=fakulteID

GROUP BY b.bolumAdi;

END \$

DELIMITER ;

call pd_fakulteBolumOgrenciSayisi(1)

16. Üniversitenin genelindeki bölüm bazlı öğrenci sayısı

DELIMITER \$

CREATE PROCEDURE pd_UniversiteGeneliBolumOgrenciSayisi(IN universiteID
int)

BEGIN

SELECT f.fakulteAdi,b.bolumAdi,COUNT(*) AS "ogrenci sayisi" FROM
bolum_ogrenci bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID=bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN fakulte f ON f.fakulteID=b.fakulteID

INNER JOIN universite u ON u.universiteID = f.universiteID

INNER JOIN kisi k ON k.kisiID = o.kisiID

WHERE u.universiteID=universiteID


```
GROUP BY f.fakulteAdi,b.bolumAdi;

END $

DELIMITER ;

call pd_UniversiteGeneliBolumOgrenciSayisi(1)
```

17. Üniversite genelindeki üstün başarılı öğrenci sayısı

```
DELIMITER $

CREATE PROCEDURE pd_universitedekiUstunBasari(IN universiteID INT)

BEGIN

    SELECT k.ad,k.soyad,da.agno, COUNT(DISTINCT IF(da.agno>=3.5,
    "ÜSTÜN BAŞARILI",NULL)) AS ustunBasariliOgrenciSayisi FROM
    donem_agno da

    INNER JOIN ogrenci_sinavsonucu oss ON oss.ogrenci_sinavSonucID =
    da.ogrenci_sinavSonucID

    INNER JOIN guz_bahardonem gbd ON gbd.guzBaharID = da.guzBaharID

    INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID

    INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID

    INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID

    INNER JOIN sinav s ON s.sinavID=ds.sinavID

    INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
    od.bolum_ogrenciID

    INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID

    INNER JOIN bolum b ON b.bolumID = bd.bolumID

    INNER JOIN ders d ON d.dersID = bd.dersID

    INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

    INNER JOIN fakulte f ON f.fakulteID = b.fakulteID

    INNER JOIN universite u ON u.universiteID = f.universiteID

    INNER JOIN kisi k ON k.kisiId = o.kisiId

    WHERE u.universiteID = universiteID;

END $
```

DELIMITER ;

call pd_universitedekiUstunBasari(1)

18. Üniversitede bölüm bazlı üstün başarılı öğrenci sayısı

DELIMITER \$

CREATE PROCEDURE pd_universitedeBolumBazliBasari(IN bolumID INT)

BEGIN

SELECT k.ad,k.soyad,da.agno, COUNT(DISTINCT IF(da.agno>=3.5,
"BAŞARILI",NULL)) AS ustunBasariliOgrenciSayisi FROM donem_agno da

INNER JOIN ogrenci_sinavsonucu oss ON oss.ogrenci_sinavSonucID =
da.ogrenci_sinavSonucID

INNER JOIN guz_bahardonem gbd ON gbd.guzBaharID = da.guzBaharID

INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID

INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID

INNER JOIN sinav s ON s.sinavID=ds.sinavID

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN fakulte f ON f.fakulteID = b.fakulteID

INNER JOIN universite u ON u.universiteID = f.universiteID

INNER JOIN kisi k ON k.kisiId = o.kisiId

WHERE b.bolumID = bolumID;

END \$

DELIMITER ;

call pd_universitedeBolumBazliBasari(1)

19. Bölüm bazlı dönem müfredatı

DELIMITER \$

CREATE PROCEDURE pd_bolumBazliDonemMufredat(IN bolumID INT)

BEGIN

```
SELECT d.dersAdi, don.donem, dersIcerigi FROM mufredat m
INNER JOIN donem don ON don.donemID = m.donemID
INNER JOIN bolum_ders bd ON bd.bolum_dersID = m.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
WHERE b.bolumID = bolumID;
```

END \$

DELIMITER ;

call pd_bolumBazliDonemMufredat(1)

20. Bölüm Id ye göre Öğrencilerin yaşadıkları memleket listesi

DELIMITER \$

CREATE PROCEDURE pd_BolumOgrenciYASadigiIl(IN bolumID INT)

BEGIN

```
SELECT b.bolumAdi,k.ad,k.soyad,i.ilAdi AS "yaşadığı il" FROM
bolum_ogrenci bo
INNER JOIN bolum_ders bd ON bd.bolum_dersID =bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiID = o.kisiID
INNER JOIN ilce ilc ON ilc.ilceID = k.ilceID
INNER JOIN il i ON i.ilID = ilc.ilID
WHERE b.bolumID=bolumID;
```

END \$

DELIMITER ;

call pd_BolumOgrenciYASadigiIl(1)

21. Öğrencilerin fakülte bazlı doğum yerleri

DELIMITER \$

CREATE PROCEDURE pd_fakulteOgrenciDogumIl(IN fakulteID INT)

BEGIN

SELECT f.fakulteAdi, k.ad,k.soyad,i.ilAdi AS "doğum yerleri" FROM
bolum_ogrenci bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID =bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN fakulte f ON f.fakulteID=b.fakulteID

INNER JOIN kisi k ON k.kisiId = o.kisiId

INNER JOIN ilce ilc ON ilc.ilceID = k.ilceID

INNER JOIN il i ON i.ilID = ilc.ilID

WHERE f.fakulteID=fakulteID;

END \$

DELIMITER ;

call pd_fakulteOgrenciDogumIl(1)

22. Ders Id ye Final sınavına giren öğrencilerin listesi ders bazlı

DELIMITER \$

CREATE PROCEDURE pd_dersBazliFinalSinavinaGirenOgrenciSayisi(IN dersID
INT)

BEGIN

SELECT d.dersAdi,os.ogrenci_sinavID,os.sinavNotu,COUNT(*) AS "öğrenci
sayısı" FROM ogrenci_sinav os

INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID

```

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID =
od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN sinav s ON s.sinavID = ds.sinavID

INNER JOIN sinav_tur st ON st.sinav_turID=s.sinav_turID

INNER JOIN kisi k ON k.kisiID = o.kisiID

INNER JOIN ders d ON d.dersID = bd.dersID

WHERE d.dersID = dersID AND os.sinavNotu IS NOT NULL AND
st.sinavTuru= "Final";

END $

DELIMITER ;

call pd_dersBazliFinalSinavinaGirenOgrenciSayisi(1)

```

23. Ders Id ye göre Final sınavına girmeyen öğrencilerin sayısı

```

DELIMITER $

CREATE PROCEDURE pd_dersBazliFinalSinavinaGirmeyenOgrenciSayisi(IN
dersID int)

BEGIN

SELECT d.dersAdi,os.ogrenci_sinavID, COUNT(*) AS "öğrenci sayısı"
FROM ogrenci_sinav os

INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID =
od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN sinav s ON s.sinavID = ds.sinavID

INNER JOIN sinav_tur st ON st.sinav_turID=s.sinav_turID

INNER JOIN kisi k ON k.kisiID = o.kisiID

INNER JOIN ders d ON d.dersID = bd.dersID

```

```

WHERE d.dersID = dersID AND os.sinavNotu IS NULL AND st.sinavTuru=
"Final" ;

END $

DELIMITER ;

call pd_dersBazliFinalSinavinaGirmeyenOgrenciSayisi(1)

```

24. Fakültelerdeki agnosu 3 ün üstünde olan öğrencilerin listesi

```

DELIMITER $

CREATE PROCEDURE pd_fakultedeAgnosuUcUzeriOlanOgrenci(IN fakulteID
INT)
BEGIN

    SELECT DISTINCT IF(da.agno>3,k.ad,NULL) AS
    agnosuUcUzeriOlanOgrenci FROM donem_agno da

    INNER JOIN ogrenci_sinavsonucu oss ON oss.ogrenci_sinavSonucID =
    da.ogrenci_sinavSonucID

    INNER JOIN guz_bahardonem gbd ON gbd.guzBaharID = da.guzBaharID

    INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID

    INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID

    INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID

    INNER JOIN sinav s ON s.sinavID=ds.sinavID

    INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
    od.bolum_ogrenciID

    INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID

    INNER JOIN bolum b ON b.bolumID = bd.bolumID

    INNER JOIN ders d ON d.dersID = bd.dersID

    INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

    INNER JOIN fakulte f ON f.fakulteID = b.fakulteID

    INNER JOIN universite u ON u.universiteID = f.universiteID

    INNER JOIN kisi k ON k.kisiId = o.kisiId

    WHERE f.fakulteID = fakulteID;

END $

```

DELIMITER ;

call pd_fakultedeAgnosuUcUzeriOlanOgrenci(1)

25. ÖSYM ile sınavı kazanmış kaydı tamamlanmamış öğrencilerin listesi

DELIMITER \$

CREATE PROCEDURE pd_osymKazanmisKaydiTamamlanmamisOgrenciListesi(IN ogrenciID INT)

BEGIN

SELECT o.ogrenciID,k.ad,k.soyad, s.statu, gs.gelisSekli FROM ogrenci o

INNER JOIN statu s ON s.statuID = o.statuID

INNER JOIN gelissekli gs ON gs.gelisSekliID = o.gelisSekliID

INNER JOIN kisi k ON k.kisiID = o.kisiID

WHERE o.ogrenciID = ogrenciID AND s.statu = "Kaydı Tamamlanmamış"
AND gs.gelisSekli = "ÖSYM" ;

END \$

DELIMITER ;

call pd_osymKazanmisKaydiTamamlanmamisOgrenciListesi(1)

26. Bölüm bazlı telefon numarası eksik olan öğrencilerin listesi

DELIMITER \$

CREATE PROCEDURE pd_bolumBazliIletisimiOlmayanOgrenciListesi(IN bolumID INT)

BEGIN

SELECT k.ad, k.soyad, i.iletisim, ilt.iletisimTur FROM bolum_ogrenci bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

INNER JOIN iletisim i ON i.iletisimID=o.iletisimID

INNER JOIN iletisimturu ilt ON ilt.iletisimTurID = i.iletisimTurID

```

WHERE b.bolumID=bolumID AND i.iletisim IS NULL AND ilt.iletisimTurID
= 1;

END $

DELIMITER ;

call pd_bolumBazliIletisimiOlmayanOgrenciListesi(1)

```

27. Bölüme ders veren hocaların listesi

```

DELIMITER $

CREATE PROCEDURE pd_bolumedersverenhocalar(IN bolumID INT)
BEGIN
    SELECT b.bolumAdi, d.dersAdi, u.unvan, k.ad, k.soyad FROM hoca_ders hd
    INNER JOIN hoca_unvan hu ON hu.hoca_unvanID = hd.hoca_unvanID
    INNER JOIN bolum_ders bd ON bd.bolum_dersID = hd.bolum_dersID
    INNER JOIN bolum b ON b.bolumID = bd.bolumID
    INNER JOIN ders d ON d.dersID = bd.dersID
    INNER JOIN hoca h ON h.hocaID = hu.hocaID
    INNER JOIN unvan u ON u.unvanID = hu.unvanID
    INNER JOIN kisi k ON k.kisiID = h.kisiID
    WHERE b.bolumID = bolumID ;

END $

DELIMITER ;

call pd_bolumedersverenhocalar(1)

```

28. Dekanlık barajına takılan öğrencilerin listesi ders bazlı

```

DELIMITER $

CREATE PROCEDURE pd_dersBazliDekanlikBarajiTakilanOgrenciListesi(IN
dersID INT)
BEGIN

```



```
SELECT k.ad, k.soyad, d.dersAdi, oss.ort, IF(oss.ort<39.5 , "Dekanlık Barajına Takıldı" , "Dekanlık Barajından Geçti") AS "Dekanlık Barajına Takılan Öğrenci" FROM ogrenci_snavsonucu oss
```

```
INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = oss.ogrenci_dersID
```

```
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID =  
od.bolum_ogrenciID
```

```
INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID
```

```
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
```

```
INNER JOIN kisi k ON k.kisiID = o.kisiID
```

```
INNER JOIN ders d ON d.dersID = bd.dersID
```

```
WHERE d.dersID = dersID;
```

```
END $
```

```
DELIMITER ;
```

```
call pd_dersBazliDekanlikBarajiTakilanOgrenciListesi(1)
```

29. Sistemde maili olmayan öğrencilerin listesi bölüm bazlı

```
DELIMITER $
```

```
CREATE PROCEDURE pd_bolumBazliMailiOlmayanOgrenciListesi(IN bolumID  
int)
```

```
BEGIN
```

```
SELECT k.ad,k.soyad,i.iletisim,ilt.iletisimTur FROM bolum_ogrenci bo
```

```
INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID
```

```
INNER JOIN bolum b ON b.bolumID = bd.bolumID
```

```
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
```

```
INNER JOIN kisi k ON k.kisiID = o.kisiID
```

```
INNER JOIN iletisim i ON i.iletisimID=o.iletisimID
```

```
INNER JOIN iletisimturu ilt ON ilt.iletisimTurID = i.iletisimTurID
```

```
WHERE b.bolumID=bolumID AND i.iletisim IS NULL AND ilt.iletisimTurID  
= 2;
```

```
END $
```

```
DELIMITER ;
```

call pd_bolumBazliMailiOlmayanOgrenciListesi(1)

30. Hiçbir sınava girmeyen öğrencilerin listesi ders bazlı

DELIMITER \$

CREATE PROCEDURE pd_dersBazliHicbirSinavaGirmeyenOgrenciListesi(IN
dersID INT)

BEGIN

```
SELECT k.ad,k.soyad,d.dersAdi,st.sinavTuru,os.sinavNotu FROM  
ogrenci_sinav os  
INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID  
INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID  
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID =  
od.bolum_ogrenciID  
INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID  
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID  
INNER JOIN sinav s ON s.sinavID = ds.sinavID  
INNER JOIN sinav_tur st ON st.sinav_turID=s.sinav_turID  
INNER JOIN kisi k ON k.kisiID = o.kisiID  
INNER JOIN ders d ON d.dersID = bd.dersID  
WHERE d.dersID = dersID AND os.sinavNotu IS NULL;
```

END \$

DELIMITER ;

call pd_dersBazliHicbirSinavaGirmeyenOgrenciListesi(1)

SCALAR FONKSİYON

1. Öğrenci ID' ye göre T.C. getiren fonksiyon

DELIMITER \$\$

```

CREATE FUNCTION sf_ogrenciTCKimlik(ogrenciID INT)
RETURNS VARCHAR(45) DETERMINISTIC
BEGIN
    SET @tc=(SELECT k.TCKimlikNo FROM ogrenci o
    INNER JOIN kisi k ON k.kisiID = o.kisiID
    WHERE o.ogrenciID = ogrenciID) ;
RETURN @tc;
END;
SELECT sf_ogrenciTCKimlik(1);

```

2. Öğrenci ID' ye göre agnoyu getiren fonksiyon

```

DELIMITER $$
CREATE FUNCTION sf_ogrenciAgno(ogrenciID INT)
RETURNS FLOAT DETERMINISTIC
BEGIN
    SET @agnoGetir=(SELECT (SUM(DISTINCT d.kredi * oss.harfNotKatsayisi)
    / SUM(DISTINCT d.kredi)) AS son_agno FROM ogrenci_snavsonucu oss
    INNER JOIN ogrenci_snav os ON os.ogrenci_snavID
    INNER JOIN ders_snav ds ON ds.ders_snavID = os.ders_snavID
    INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
    INNER JOIN sinav s ON s.sinavID=ds.sinavID
    INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
    od.bolum_ogrenciID
    INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
    INNER JOIN bolum b ON b.bolumID = bd.bolumID
    INNER JOIN ders d ON d.dersID = bd.dersID
    INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
    INNER JOIN kisi k ON k.kisiId = o.kisiId
    WHERE o.ogrenciID = ogrenciID) ;
RETURN @agnoGetir;

```

END;

SELECT sf_ogrenciAgno(1);

3. Öğrenci ID' ye göre toplam krediyi getiren fonksiyon

DELIMITER \$\$

CREATE FUNCTION sf_ogrenciToplamKredi(ogrenciID INT)

RETURNS INT DETERMINISTIC

BEGIN

SET @krediGetir=(SELECT CASE WHEN oss.harfNot= "AA" OR oss.harfNot = "BA" OR oss.harfNot = "BB" OR oss.harfNot = "CD" OR oss.harfNot = "CC"

THEN SUM(DISTINCT d.kredi) END AS toplam_kredi FROM
ogrenci_sinavsonucu oss

INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID

INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID

INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID

INNER JOIN sinav s ON s.sinavID=ds.sinavID

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiId = o.kisiId

WHERE o.ogrenciID = ogrenciID) ;

RETURN @krediGetir;

END;

SELECT sf_ogrenciToplamKredi(1);

4. Bölüm ID' ye göre toplam öğrenci sayısını getiren fonksiyon

DELIMITER \$\$

CREATE FUNCTION sf_ogrenciSayisi(bolumID INT)

RETURNS INT DETERMINISTIC

BEGIN

SET @ogrenciSayisiGetir=(SELECT COUNT(*) AS "ogrenci sayisi" FROM
bolum_ogrenci bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID=bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

WHERE b.bolumID = bolumID) ;

RETURN @ogrenciSayisiGetir;

END;

SELECT sf_ogrenciSayisi(1);

5. Bölüm ID' ye göre kaç sınıf var

DELIMITER \$\$

CREATE FUNCTION sf_bolumSinifSayisi(bolumID INT)

RETURNS INT DETERMINISTIC

BEGIN

SET @sinifSayisiGetir=(SELECT COUNT(*) AS "sinif" FROM dersprogrami
dp

INNER JOIN bolum_ders bd ON bd.bolum_dersID = dp.bolum_dersID

INNER JOIN ogretimsekli os ON os.ogretimSekliID = dp.ogretimSekliID

INNER JOIN derslik ds ON ds.derslikID = dp.derslikID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

WHERE b.bolumID = bolumID) ;

RETURN @sinifSayisiGetir;

END;

SELECT sf_ogrenciSayisi(1);

6. Fakülteye göre bölüm sayısını getiren

DELIMITER \$\$

CREATE FUNCTION sf_fakulteBolumSayisi(fakulteID INT)

RETURNS INT DETERMINISTIC

BEGIN

SET @bolumSayisiGetir=(SELECT COUNT(*) AS "bölüm sayısı" FROM
bolum b

INNER JOIN fakulte f ON f.fakulteID=b.fakulteID

WHERE f.fakulteID = fakulteID) ;

RETURN @bolumSayisiGetir;

END;

SELECT sf_fakulteBolumSayisi(1);

7. Uni ID' ye göre fakülte bölüm sayısını getiren

DELIMITER \$\$

CREATE FUNCTION sf_universiteFakulteSayisi(universiteID INT)

RETURNS INT DETERMINISTIC

BEGIN

SET @universiteFakulteSayisiGetir=(SELECT COUNT(*) AS "bölüm sayısı"
FROM bolum b

INNER JOIN fakulte f ON f.fakulteID=b.fakulteID

INNER JOIN universite u ON u.universiteID = f.universiteID

WHERE u.universiteID = universiteID) ;

RETURN @universiteFakulteSayisiGetir;

END;

SELECT sf_universiteFakulteSayisi(1);

8. Derse göre dersi alan öğrenci sayısını getiren

DELIMITER \$\$

CREATE FUNCTION sf_dersOgrenciSayisi(dersID INT)

RETURNS INT DETERMINISTIC

BEGIN

SET @dersOgrenciSayisiGetir=(SELECT COUNT(*) AS ogrenci_sayisi
FROM ogrenci_ders od

INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID =
od.bolum_ogrenciID

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID = bd.bolumID

INNER JOIN ders d ON d.dersID = bd.dersID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

INNER JOIN kisi k ON k.kisiID = o.kisiID

WHERE d.dersID = dersID) ;

RETURN @dersOgrenciSayisiGetir;

END;

SELECT sf_dersOgrenciSayisi(1);

9. Fakülte ID' ye göre öğrenci sayısını getiren

DELIMITER \$\$

CREATE FUNCTION sf_fakulteOgrenciSayisi(fakulteID INT)

RETURNS INT DETERMINISTIC

BEGIN

SET @fakulteOgrenciSayisiGetir=(SELECT COUNT(*) AS ogrenci_sayisi
FROM bolum_ogrenci bo

INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID

INNER JOIN bolum b ON b.bolumID=bd.bolumID

INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID

```

INNER JOIN fakulte f ON f.fakulteID=b.fakulteID
INNER JOIN kisi k ON k.kisiID = o.kisiID
WHERE f.fakulteID = fakulteID) ;
RETURN @fakulteOgrenciSayisiGetir;
END;
SELECT sf_fakulteOgrenciSayisi(1);

```

10. Öğrencinin kaldığı derslerin toplam kredisi

```
DELIMITER $$
```

```
CREATE FUNCTION sf_kaldigitoplamOgrenciKredi(ogrenciID INT)
```

```
RETURNS INT DETERMINISTIC
```

```
BEGIN
```

```
SET @toplamKrediSayisiGetir=(SELECT CASE WHEN oss.harfNot= "FF"
```

```
THEN SUM(DISTINCT d.kredi) END AS toplam_kredi FROM
ogrenci_snavsonucu oss
```

```
INNER JOIN ogrenci_snav os ON os.ogrenci_snavID
```

```
INNER JOIN ders_snav ds ON ds.ders_snavID = os.ders_snavID
```

```
INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID
```

```
INNER JOIN snav s ON s.snavID=ds.snavID
```

```
INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID=
od.bolum_ogrenciID
```

```
INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID
```

```
INNER JOIN bolum b ON b.bolumID = bd.bolumID
```

```
INNER JOIN ders d ON d.dersID = bd.dersID
```

```
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
```

```
INNER JOIN kisi k ON k.kisiId = o.kisiId
```

```
WHERE o.ogrenciID = ogrenciID ) ;
```

```
RETURN @toplamKrediSayisiGetir;
```

```
END;
```

```
SELECT sf_kaldigitoplamOgrenciKredi(1);
```


TRIGGER

1.Öğrenci tablosunda güncelleme ve silme ile tetiklenip LOG tablosuna kaydeden trigger ı yazın.

```
DELIMITER $$
```

```
CREATE TRIGGER ogrenciGuncelleme
```

```
AFTER UPDATE ON ogrenci
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET ogrenciID = NEW.ogrenciID;
```

```
END;
```

```
DELIMITER $$
```

```
CREATE TRIGGER ogrenciSilme
```

```
AFTER DELETE ON ogrenci
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET ogrenciID = OLD.ogrenciID;
```

```
END;
```

2. Ders tablosunda ekleme, güncelleme ve silme ile tetiklenip LOG tablosuna kaydeden trigger ı yazın.

```
DELIMITER $$
```

```
CREATE TRIGGER dersEkle
```

```
AFTER INSERT ON ders
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET dersID = NEW.dersID;
```

```
END;
```

```
DELIMITER $$  
  
CREATE TRIGGER dersGuncelle  
AFTER UPDATE ON ders  
FOR EACH ROW  
BEGIN  
INSERT INTO log SET dersID = NEW.dersID;  
END;
```

```
DELIMITER $$  
  
CREATE TRIGGER dersSil  
AFTER DELETE ON ders  
FOR EACH ROW  
BEGIN  
INSERT INTO log SET dersID = OLD.dersID;  
END;
```

3. DersOgrenci tablosunda ekleme, güncelleme ve silme ile tetiklenip LOG tablosuna kaydeden trigger 1 yazın.

```
DELIMITER $$  
  
CREATE TRIGGER ders_ogrenciEkle  
AFTER INSERT ON ogrenci_ders  
FOR EACH ROW  
BEGIN  
INSERT INTO log SET ogrenci_dersID = NEW.ogrenci_dersID;  
END;
```

```
DELIMITER $$
```

```
CREATE TRIGGER ders_ogrenciGuncelle
AFTER UPDATE ON ogrenci_ders
FOR EACH ROW
BEGIN
INSERT INTO log SET ogrenci_dersID = NEW.ogrenci_dersID;
END;
```

```
DELIMITER $$
CREATE TRIGGER ders_ogrenciSil
AFTER DELETE ON ogrenci_ders
FOR EACH ROW
BEGIN
INSERT INTO log SET ogrenci_dersID = OLD.ogrenci_dersID;
END;
```

4.Program tablosunda ekleme ve güncelleme ile tetiklenip LOG tablosuna kaydeden trigger ı yazın.

```
DELIMITER $$
CREATE TRIGGER programEkle
AFTER INSERT ON dersprogrami
FOR EACH ROW
BEGIN
INSERT INTO log SET dersProgramiID = NEW.dersProgramiID;
END;
```

```
DELIMITER $$
CREATE TRIGGER programGuncelle
AFTER UPDATE ON dersprogrami
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET dersProgramiID = NEW. dersProgramiID;
```

```
END;
```

5.İletisim tablosunda ekleme, güncelleme ve silme ile tetiklenip LOG tablosuna kaydeden trigger 1 yazın.

```
DELIMITER $$
```

```
CREATE TRIGGER iletisimEkle
```

```
AFTER INSERT ON iletisim
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET iletisimID = NEW. iletisimID;
```

```
END;
```

```
DELIMITER $$
```

```
CREATE TRIGGER iletisimSil
```

```
AFTER DELETE ON iletisim
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET iletisimID = OLD. iletisimID;
```

```
END;
```

```
DELIMITER $$
```

```
CREATE TRIGGER iletisimGuncelle
```

```
AFTER UPDATE ON iletisim
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET iletisimID = NEW. iletisimID;
```

```
END;
```

6. Müfredat tablosunda güncelleme ve silme ile tetiklenip LOG tablosuna kaydeden trigger ı yazın.

```
DELIMITER $$
```

```
CREATE TRIGGER mufredatGuncelle
```

```
AFTER UPDATE ON mufredat
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET mufredatDersID = NEW.mufredatDersID;
```

```
END;
```

```
DELIMITER $$
```

```
CREATE TRIGGER mufredatSil
```

```
AFTER DELETE ON mufredat
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET mufredatDersID = OLD.mufredatDersID;
```

```
END;
```

7. Ders tablosunda güncelleme ve silme ile tetiklenip LOG tablosuna kaydeden trigger ı yazın.

```
DELIMITER $$
```

```
CREATE TRIGGER ders
```

```
AFTER UPDATE ON ders
```

```
FOR EACH ROW
```

```
BEGIN
```

```
INSERT INTO log SET dersID = NEW.dersID;
```

```
END;
```

```
DELIMITER $$
```

```
CREATE TRIGGER ders
AFTER DELETE ON ders
FOR EACH ROW
BEGIN
INSERT INTO log SET dersID = OLD.dersID;
END;
```

8. Ders tablosunda kredi ve ACTS sütunlarında güncelleme ve silme ile tetiklenip LOG tablosuna kaydeden trigger 1 yazın.

```
DELIMITER $$
CREATE TRIGGER dersKrediGuncelle
AFTER UPDATE ON ders
FOR EACH ROW
BEGIN
INSERT INTO log SET kredi = NEW.kredi;
END;
```

```
DELIMITER $$
CREATE TRIGGER dersKrediSilme
AFTER DELETE ON ders
FOR EACH ROW
BEGIN
INSERT INTO log SET kredi = OLD.kredi;
END;
```

9. İletişim tablosunda iletişimBilgi (telefon, mail, fax gibi bilgileri içeren sütun) sütununda silme ile tetiklenip LOG tablosuna kaydeden trigger 1 yazın.

```
DELIMITER $$
CREATE TRIGGER iletisimTurSilme
```

```
AFTER DELETE ON iletisim
FOR EACH ROW
BEGIN
INSERT INTO log SET iletisimTurID = OLD.iletisimTurID;
END;
```

10. OgrenciSinav tablosunda sinavPuan sütununda ekleme, güncelleme ve silme ile tetiklenip LOG tablosuna kaydeden trigger ı yazın.

```
DELIMITER $$
CREATE TRIGGER ogrenci_sinavEkle
AFTER INSERT ON ogrenci_sinav
FOR EACH ROW
BEGIN
INSERT INTO log SET sinavNotu = NEW.sinavNotu;
END;
```

```
DELIMITER $$
CREATE TRIGGER ogrenci_sinavGuncelle
AFTER UPDATE ON ogrenci_sinav
FOR EACH ROW
BEGIN
INSERT INTO log SET sinavNotu = NEW.sinavNotu;
END;
```

```
DELIMITER $$
CREATE TRIGGER ogrenci_sinavSil
AFTER INSERT ON ogrenci_sinav
FOR EACH ROW
BEGIN
```

```
INSERT INTO log SET sinavNotu = OLD.sinavNotu;  
END;
```

TABLO FONKSİYONLARI

1. Öğrenci Id ye göre Öğrenci bazlı kalıp yeniden alması gereken zorunlu derslerin listesi

```
CREATE FUNCTION f_kaldigiAlmasiGerekenZorunluDersListesi(@ogrenciID INT)  
RETURNS TABLE  
AS  
RETURN (SELECT DISTINCT d.dersAdi FROM ogrenci_sinavsonucu oss  
        INNER JOIN ogrenci_sinav os ON os.ogrenci_sinavID  
        INNER JOIN ders_sinav ds ON ds.ders_sinavID = os.ders_sinavID  
        INNER JOIN ogrenci_ders od ON od.ogrenci_dersID = ds.ogrenci_dersID  
        INNER JOIN sinav s ON s.sinavID=ds.sinavID  
        INNER JOIN bolum_ogrenci bo ON bo.bolum_ogrenciID= od.bolum_ogrenciID  
        INNER JOIN bolum_ders bd ON bd.bolum_dersID=bo.bolum_dersID  
        INNER JOIN bolum b ON b.bolumID = bd.bolumID  
        INNER JOIN ders d ON d.dersID = bd.dersID  
        INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID  
        INNER JOIN kisi k ON k.kisiId = o.kisiId  
        WHERE o.ogrenciID = @ogrenciID AND oss.harfNot= "FF" AND d.durumu =  
        "Z" )
```

4. Bölüm Id ye göre Ders bazlı canlı derse katılan öğrencilerin listesi

```
CREATE FUNCTION f_ogrenciCanliDersListesi(@bolumID INT)  
RETURNS TABLE  
AS  
RETURN (SELECT d.dersAdi, k.ad,k.soyad FROM ogrenci_dersprogrami odp  
        INNER JOIN dersprogrami dp ON dp.dersProgramiID = odp.dersProgramiID
```



```

INNER JOIN bolum_ogrenci bo ON dp.bolum_ogrenciID
INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID
INNER JOIN ogretimsekli os ON os.ogretimSekliID = dp.ogretimSekliID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiID = o.kisiID
WHERE b.bolumID = @bolumID AND os.ogretimSekliID = 2)

```

5. Bölüm Id ye göre Ders bazlı her bir dersin canlı ders sayısı

```

CREATE FUNCTION f_canliDersSayisi(@bolumID INT)
RETURNS TABLE
AS
RETURN (SELECT COUNT(DISTINCT dersAdi) AS
canli_dersSayisi FROM ogrenci_dersprogrami odp
INNER JOIN dersprogrami dp ON dp.dersProgramiID = odp.dersProgramiID
INNER JOIN bolum_ogrenci bo ON dp.bolum_ogrenciID
INNER JOIN bolum_ders bd ON bd.bolum_dersID = bo.bolum_dersID
INNER JOIN ogretimsekli os ON os.ogretimSekliID = dp.ogretimSekliID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ders d ON d.dersID = bd.dersID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN kisi k ON k.kisiID = o.kisiID
WHERE b.bolumID = @bolumID AND os.ogretimSekliID = 2)

```

6. Fakülte Id ye göre Öğrenci kayıt türleri listesi

```

CREATE FUNCTION f_ogrenciKayitTurListesi(@fakulteID INT)
RETURNS TABLE
AS

```

```

RETURN (SELECT kyt.kayitTur FROM bolum_ogrenci bo
INNER JOIN bolum_Ders bd ON bd.bolum_dersID = bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN fakulte f ON f.fakulteID = b.fakulteID
INNER JOIN kisi k ON k.kisiID = o.kisiID
INNER JOIN kayittur kyt ON kyt.kayitID = o.kayitID)
WHERE f.fakulteID = @fakulteID )

```

7. Bölüm Id ye göre Öğrencinin öğrenim türü

```

CREATE FUNCTION f_ogrenciOgretimTurListesi(@bolumID INT)
RETURNS TABLE
AS
RETURN (SELECT DISTINCT b.ogrenimTur FROM bolum_ogrenci bo
INNER JOIN bolum_Ders bd ON bd.bolum_dersID = bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN fakulte f ON f.fakulteID = b.fakulteID
INNER JOIN kisi k ON k.kisiID = o.kisiID)
WHERE b.bolumID = @bolumID )

```

8. Bölüm Id ye göre Bölüm bazlı öğrencilerin mezun oldukları liselerin listesi

```

CREATE FUNCTION f_ogrencininGeldigiLise(@bolumID INT)
RETURNS TABLE
AS
RETURN (SELECT o.geldigiOkul FROM bolum_ogrenci bo
INNER JOIN bolum_Ders bd ON bd.bolum_dersID = bo.bolum_dersID
INNER JOIN bolum b ON b.bolumID = bd.bolumID
INNER JOIN ogrenci o ON o.ogrenciID = bo.ogrenciID
INNER JOIN fakulte f ON f.fakulteID = b.fakulteID

```

```
INNER JOIN kisi k ON k.kisiID = o.kisiID  
WHERE b.bolumID = @bolumID)
```