

1-) SELECT kredi,ders_ad,bolum_id, MIN(kredi) AS dusuk_kredi FROM dersler WHERE kredi=(SELECT MIN(kredi) FROM dersler) GROUP BY kredi,ders_ad,bolum_id;

2-) SELECT final,ogrenci_id,ders_id,donem, MAX(final) AS en_yuksek_not FROM ders_kayit WHERE final=(SELECT MAX(final) FROM ders_kayit) GROUP BY final,ogrenci_id,ders_id,donem;

3-) SELECT ogrenci_ad,ogrenci_soyad,danisman_id,bolum_id FROM ogrenci WHERE ogrenci_email LIKE "%an%" ORDER BY bolum_id DESC,danisman_id ASC;

4-) SELECT ogretmen_ad,ogretmen_soyad FROM ogretmen WHERE ogretmen_ad NOT LIKE "M%" ORDER BY ogretmen_soyad ASC;

5-) SELECT danisman_id, COUNT() AS ogrenci_sayisi FROM ogrenci GROUP BY danisman_id HAVING COUNT() > 1;

6-) SELECT ders_ad,ROUND(AVG(akts),0) AS yuksek_akts FROM dersler GROUP BY ders_id HAVING ROUND(AVG(akts),0) > 4;

7-) CREATE VIEW ogrenci_not_gorunumu AS SELECT ogrenci_id,ROUND(AVG((vize + final)/2),2) AS ortalama_not FROM ders_kayit GROUP BY ogrenci_id HAVING ROUND(AVG((vize + final)/2),2)>65;

8-) CREATE VIEW ders_zorluk AS SELECT ders_ad,kredi,akts, (kredi0.6 + akts0.4) AS zorluk_derecesi, CASE WHEN (kredi0.6 + akts0.4)>=5 THEN 'Zor' WHEN (kredi0.6 + akts0.4)>=3 THEN 'Orta' ELSE 'Kolay' END AS derece FROM dersler; SELECT*FROM ders_zorluk;

9-) CREATE VIEW ogrenci_bilgi AS SELECT ogrenci_ad,ogrenci_soyad,ogrenci_id,ogrenci_email FROM ogrenci WHERE bolum_id>1 AND ogrenci_ad LIKE "%e%"; SELECT * FROM ogrenci_bilgi;

10-) SELECT ogrenci_ad,ogrenci_soyad FROM ogrenci WHERE ogrenci_id IN (SELECT ogrenci_id FROM ders_kayit WHERE vize>(SELECT AVG(vize) FROM ders_kayit));

11-) SELECT ogretmen_ad,ogretmen_soyad FROM ogretmen WHERE ogretmen_id IN (SELECT ogretmen_id FROM dersler GROUP BY ogretmen_id HAVING COUNT(ders_id)=3);

12-) SELECT ogrenci.ogrenci_ad,ogrenci.ogrenci_soyad,ders_kayit.vize,ders_kayit.final,ders_kayit.donem FROM ogrenci JOIN ders_kayit ON ogrenci.ogrenci_id=ders_kayit.ogrenci_id;

13-) SELECT o.ogrenci_ad,o.ogrenci_soyad,d.ders_ad,d.kredi,d.akts,b.bolum_ad FROM
ogrenci AS o JOIN dersler AS d ON o.bolum_id=d.bolum_id JOIN bolum AS b ON
d.bolum_id=b.bolum_id

14-) DELIMITER //

CREATE PROCEDURE ogretmen_ara(IN harf_arama VARCHAR(50), IN bolum_no INT)
BEGIN SELECT ogretmen_ad, ogretmen_email, bolum_id FROM ogretmen WHERE
ogretmen_email LIKE CONCAT(harf_arama, '%') AND bolum_id >= bolum_no; END //

DELIMITER ; CALL ogretmen_ara('a',1);

15-) DELIMITER // CREATE PROCEDURE dersEkle(IN p_ders_id INT ,IN p_ders_ad
VARCHAR(100),IN p_kredi INT ,IN p_akts INT ,IN p_ogretmen_id INT,IN p_bolum_id INT)

BEGIN INSERT INTO dersler(ders_id, ders_ad, kredi, akts, ogretmen_id, bolum_id)
VALUES(p_ders_id,p_ders_ad,p_kredi,p_akts,p_ogretmen_id,p_bolum_id); END//

DELIMITER ;

CALL dersEkle(309,'veri yapilari',3,5,1001,1);

SELECT*FROM dersler;

16-) DELIMITER // CREATE PROCEDURE ogrenciDersNot()

BEGIN SELECT o.ogrenci_ad,o.ogrenci_soyad,d.ders_ad,dk.vize,dk.final FROM ogrenci
AS o JOIN ders_kayit AS dk ON o.ogrenci_id=dk.ogrenci_id JOIN dersler AS d ON
d.ders_id=dk.ders_id;

END // DELIMITER ; CALL ogrenciDersNot();

17-) START TRANSACTION; DELIMITER // CREATE PROCEDURE dersEkle(IN p_ders_id
INT, IN p_ders_ad VARCHAR(50), IN p_kredi INT, IN p_akts INT, IN p_ogretmen_id INT,
IN p_bolum_id INT) BEGIN

IF p_kredi>4 THEN ROLLBACK; SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Kredi
4 uzeri olamaz';

ELSEIF p_akts<3 THEN ROLLBACK; SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT =
'Akts en az 3 olmalidir';

ELSE INSERT INTO dersler(ders_id, ders_ad, kredi, akts, ogretmen_id, bolum_id)
VALUES (p_ders_id,p_ders_ad,p_kredi,p_akts,p_ogretmen_id,p_bolum_id);

END IF;

END // DELIMITER ;

CALL dersEkle(300,'Ayrik Matematik',4,2,1002,1);

SELECT*FROM dersler ;

18-) CREATE TABLE dersharfnotu(ogrenci_id INT, vize INT, final INT, ortalama
DECIMAL(5,2) , harf_notu CHAR(2));

DELIMITER //

CREATE PROCEDURE harfnotuhesapla() BEGIN START TRANSACTION; IF EXISTS(
SELECT 1 FROM ders_kayit WHERE vize<0 OR vize>100 OR final<0 OR final>100) THEN
ROLLBACK; SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Geçersiz vize veya final
girilemez';

ELSE INSERT INTO dersharfnotu(ogrenci_id,vize,final,ortalama,harf_notu) SELECT
ogrenci_id,vize,final, (vize0.4 + final0.6) AS ortalama, CASE WHEN (vize0.4 + final0.6)
>=90 THEN 'AA' WHEN (vize0.4 + final0.6) >=80 THEN 'BB' WHEN (vize0.4 + final0.6)
>=70 THEN 'CC' WHEN (vize0.4 + final0.6) >=60 THEN 'DD' ELSE 'FF' END AS harf_notu
FROM ders_kayit; COMMIT; END IF; END // DELIMITER ;

CALL harfnotuhesapla(); SELECT*FROM dersharfnotu;

19-) DELIMITER //

CREATE PROCEDURE ogrenciEkle(IN p_ogrenci_id INT, IN p_ogrenci_ad VARCHAR(50),
IN p_ogrenci_soyad VARCHAR(50), IN p_ogrenci_mail VARCHAR(50), IN p_bolum_id
INT, IN p_vize INT, IN p_final INT, IN p_danisman_id INT) BEGIN START TRANSACTION;

IF (p_vize + p_final)/2 < 60 THEN

ROLLBACK;

SELECT '60 ortalamanin altinda ogrenci alinamaz' AS mesaj;

ELSE

INSERT INTO ogrenci(
ogrenci_id, ogrenci_ad, ogrenci_soyad, ogrenci_email,
bolum_id, danisman_id
) VALUES (
p_ogrenci_id, p_ogrenci_ad, p_ogrenci_soyad, p_ogrenci_mail,
p_bolum_id, p_danisman_id
);
COMMIT;

```
        SELECT 'Ogrenci basariyla eklendi' AS mesaj;  
END IF;
```

```
END // DELIMITER;
```

```
CALL ogrenciEkle(2010, 'Ahmet', 'Yilmaz', 'ahmet@example.com', 3, 70, 80, 1004);
```