VERİ TABANI YÖNETİM SİSTEMLERİ

ÖDEV1

ENVER USTA
1306170114
MYSQL

Soru-1

1. (10) 'Ali KURT' (name) adlı öğrencinin sid'sini ve notlarını (grade) listeleyiniz.

SELECT take.sid, take.grade

FROM take, student

WHERE student.sid=take.sid AND student.fname='Ali' AND student.lname='KURT';

+-----+-------+

| sid | grade |

+-----+-------+

| 1 | 3 |

| 1 | 2.5 |

| 1 | 3.5 |

| 1 | 3 |

| 1 | 4 |

| 1 | 3 |

+-----+-------+

SORU-2

2. (10) 'Ayşe KURT' (name) adlı öğrencinin aldığı, fakat 'Ali KURT' adlı öğrencinin almadığı derslerin kayıtlarını (yani course tablosunun tüm sütunlarını) listeleyiniz. (EXCEPT kullanınız gerekiyor)

```
SELECT *
FROM course
WHERE cid IN(
        SELECT take.cid
        FROM student, take
        WHERE fname='Ayse' AND lname='KURT' AND take.sid=student.sid
)
EXCEPT
SELECT *
FROM course
WHERE cid IN(
        SELECT take.cid
        FROM student, take
        WHERE fname='Ali' AND lname='KURT' AND take.sid=student.sid
);
```

cid	title	description	credits	did
2	operating system	CENG 341	3	1

SORU-3

3. (10) Öğrencilerin sid'lerini ve aldıkları derslerin sayısını, not ortalamasını, en yüksek ve en düşük notlarını listeleyiniz.

    SELECT take.sid, COUNT(take.cid), AVG(take.grade), MAX(take.grade), MIN(take.grade)

    FROM take, student

    WHERE student.sid=take.sid

    GROUP BY student.sid;

sid	count(take.cid)	avg(take.grade)	max(take.grade)	min(take.grade)
1	6	3.1666666666666665	4	2.5
2	11	3.909090909090909	4	3
3	11	3.8636363636363638	4	3
4	2	2	2.5	1.5
5	3	2.6666666666666665	3.5	1.5
6	1	4	4	4
7	4	2.125	3	1.5
8	2	2.5	3.5	1.5
10	2	3.5	4	3
11	1	1	1	1
12	2	3.5	3.5	3.5

3.soru:

$$\pi_{\text{take.sid, COUNT(take.cid), AVG(take.grade), MAX(take.grade), MIN(take.grade)}} \left(\gamma_{\text{student.sid, COUNT(take.cid), AVG(take.grade), MAX(take.grade), MIN(take.grade)}} \left(\sigma_{\text{student.sid=take.sid}} (\text{take} \times \text{student}) \right) \right)$$
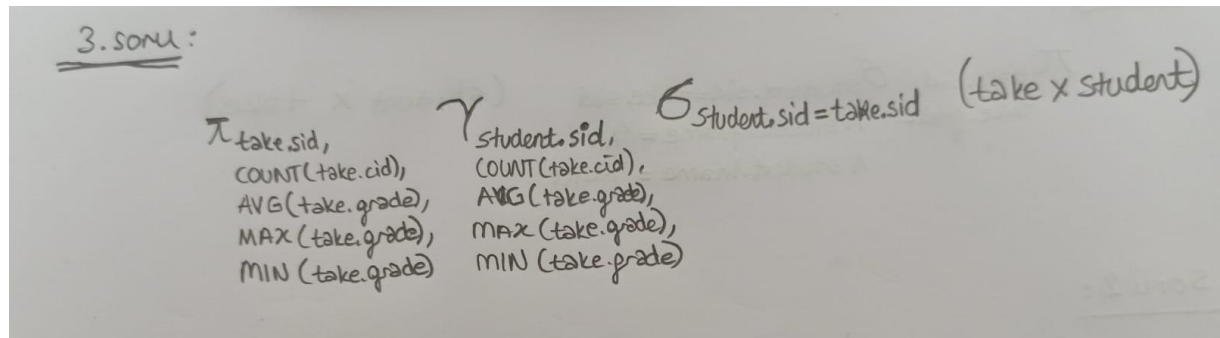
SORU-4

4. Bölümlerin did'leri, öğrenci sayılarını öğrenci syılarına göre azalan sırada listeleyiniz (İlişkisel cebirle yazmayınız)

SELECT department.did, COUNT(student.sid)

FROM department, student

WHERE department.did=student.did

GROUP BY department.did

ORDER BY COUNT(student.sid) DESC;

```
+-----+-------------------+
| did | count(student.sid) |
+-----+-------------------+
|  1 |         4 |
|  2 |         3 |
|  4 |         3 |
|  3 |         2 |
+-----+-------------------+
```

SORU-5

5. (10) 2'den fazla ders veren hocaların sid'leri, verdikleri ders sayısı ve derslerini alan öğrencilerin sayılarını listeleyiniz.

```
CREATE VIEW std_table AS

SELECT teacher.tid, COUNT(student.sid) AS countOfStd

FROM teacher, teach, student, take

WHERE teacher.tid = teach.tid AND student.sid = take.sid AND take.cid = teach.cid

GROUP BY teach.tid;


CREATE VIEW teach_table AS

SELECT teacher.tid, COUNT(teach.tid) AS countOfTeach

FROM teacher, teach

WHERE teacher.tid = teach.tid

GROUP BY teach.tid

HAVING countOfTeach > 2;


SELECT std_table.tid, std_table.countOfStd, teach_table.countOfTeach

FROM std_table

INNER JOIN teach_table ON std_table.tid = teach_table.tid;
```

SORU-6

6. (10) 'Bilgisayar Müh' (department.name) adlı bölümdeki öğrencilerden 'Elektrik Müh' (department.name) adlı bölümdeki derslerden alanlarının (take tablosunu kullan) kayıtlarını (student tablosundaki tüm alanları listele) listeleyiniz.

```
SELECT *
FROM student
WHERE did=(
        SELECT did
        FROM department
        WHERE dname='Comp. Eng.'
) AND sid in(
        SELECT sid FROM take WHERE cid in(
```

```
SELECT cid FROM course where did=(

        SELECT did

        FROM department

        WHERE dname='Elec. Eng.'

    )

  )

);
```

```
+-----+-------+-------+------------+------------+------+
| sid | fname | lname | birthdate  | birthplace | did  |
+-----+-------+-------+------------+------------+------+
|  1  | Ali   | KURT  | 1980-01-23 | istanbul   |   1  |
|  2  | Ahmet | buyuk | 1979-07-02 | ankara     |   1  |
|  3  | Leyla | Sahin | 1982-05-18 | izmir      |   1  |
+-----+-------+-------+------------+------------+------+
```


6.soru:

1st step: Write the subqueries.

$course \bowtie \pi_{did, course.*} \sigma_{dname='Elec. Eng.' \wedge course.did=department.did}$ (department x course)

→ we've also simplified course tables and did, they're not necessary

$(student \times take) \bowtie \pi_{cid} \sigma [\pi_{course.*, take.*, student.*} \sigma_{dname='Elec. Eng.' \wedge course.did=department.did \wedge take.cid=course.cid \wedge student.sid=take.sid}$ (department x take x student)]

2nd step: Find the other subquery.

$(student) \bowtie \pi_{did, student.*} \sigma_{dname='Comp. Eng.' \wedge student.did=department.did}$ (student x department)

Result = 3th step: Combine all of them.

$\pi_{student.*} \sigma [\pi_{student.*} \sigma_{dname='Comp. Eng.' \wedge student.did=department.did}$ (department)]

$\wedge [(student \times take) \bowtie \pi_{student.*, take.*, student.*} \sigma [\pi_{course.*} \sigma_{dname='Elec. Eng.' \wedge course.did=department.did \wedge take.cid=course.cid \wedge student.sid=take.sid}$ (department x take x student)]

SORU-7

7. Her dersteki öğrenci sayılarının ortalamalarını (take tablosundan her dersi kaç öğrencinin aldığı bulunacak, sonra da bu sayıların ortalamaları bulunacak) bulup, bu ortalamadan daha fazla öğrencisi olan derslerin kayıtlarını listeleyiniz. (önce ortalamadan daha fazla öğrencisi olan derslerin cid'leri bulunacak, sonra bu cid'lerden yola çıkarak course tablosundaki ders kayıtları bulunacak)

SELECT *

FROM course

WHERE cid in(

       SELECT cid

       FROM take

       GROUP BY cid

       HAVING COUNT(cid) > (

              SELECT COUNT(sid)

              FROM take)

              /

              (SELECT COUNT(DISTINCT cid)

              FROM take

)

);

cid	title	description	credits	did
1	database	CENG 351	3	1
2	operating system	CENG 341	3	1
5	statistic	IE 301	4	4
8	operation research	IE 208	3	4

7.soru.

1st step: We are going to find subqueries and then combine them.

→ $\pi_{COUNT(DISTINCT\ cid)}$ (take) combine

→ $\pi_{COUNT(sid)}$ (take) $\Rightarrow$ $\pi_{COUNT(DISTINCT\ cid)} \div \pi_{COUNT(sid)}$ (take)

2nd step. Find the other subquery. subquery will come here.

$\pi_{cid,}\ \sigma_{COUNT(cid) > \dots}\ \gamma_{cid,\ COUNT(cid),}\ \sigma_{take.cid = course.cid}$ (take × course)
course.*

• Also add FROM clause and simplify.

(course) $\bowtie$ $\pi_{cid,}\ \sigma_{count(cid) > \dots}\ \gamma_{cid,}\ \sigma_{take.cid = course.cid}$
course.* COUNT(cid)

• Add "select" and "where"

$\pi_{course.*}\ \sigma\left[\pi_{course.*}\ \sigma_{count(cid) > \dots}\ \gamma_{cid,}\ \sigma_{take.cid = course.cid}\ (take\ ×\ course)\right]$
COUNT(cid)

Result = **3th step.** Add the subquery that we found in 1st step to '...'. field.

$= \pi_{course.*}\ \sigma\left[\pi_{course.*}\ \sigma_{count(cid) > \pi_{count(DISTINCT\ cid)} \div \pi_{COUNT(sid)}\ (take)}\ \gamma_{cid,}\ \sigma_{take.cid = course.cid}\ (take)\right]$
COUNT(cid)

SORU-8

8. GROUP BY kullanmadan iki farklı ders alan öğrencilerin kayıtlarını listeleyiniz. (2 farklı ders alan dendiği için take tablosunun 2 defa kullanılması gerekiyor! Sınıfta örnek yapmıştık. Slidelarda da örnek var. NOT: önce bu öğrencilerin sid'leri bir (alt) sorgu ile bulunacak sonra bu sid'ler üzerinden student tablosundaki kayıtlara yani tüm sütunlara ulaşılacak)

SELECT *

FROM student

WHERE sid in(

    SELECT sid

    FROM take as t1

    WHERE sid in(

        SELECT sid

        FROM take AS t2

        WHERE t1.cid!=t2.cid

```
        )
);

+-----+---------+---------+-----------+-----------+------+
| sid | fname   | lname   | birthdate | birthplace| did  |
+-----+---------+---------+-----------+-----------+------+
|   1 | Ali     | KURT    | 1980-01-23| istanbul  |    1 |
|   2 | Ahmet   | buyuk   | 1979-07-02| ankara    |    1 |
|   3 | Leyla   | Sahin   | 1982-05-18| izmir     |    1 |
|   4 | Can     | Turkoglu| 1985-11-16| manisa    |    2 |
|   5 | Aziz    | Keskin  | 1978-12-11| istanbul  |    2 |
|   7 | Kamuran | Kece    | 1980-08-26| adana     |    3 |
|   8 | Turgut  | Cemal   | 1977-09-30| bursa     |    4 |
|  10 | Pelin   | Tugay   | 1980-03-09| izmir     |    4 |
|  12 | Ayse    | KURT    | 1980-12-12| izmir     |    1 |
+-----+---------+---------+-----------+-----------+------+
```


8. soru.

1st step. We have 2 subqueries. Find them.

$$\pi_{sid, \; t1.*} \; \sigma_{t1.sid = t2.sid \; \wedge \; t1.cid \, != \, t2.cid} \left(\rho_{t1}(take) \times \rho_{t2}(take) \right)$$

$\rightarrow$ Make simplifications

$$= \rho_{t1}(take) \bowtie \pi_{sid, \; t1.*} \; \sigma_{t1.sid = t2.sid \; \wedge \; t1.cid \, != \, t2.cid} \left(\rho_{t1}(take) \times \rho_{t2}(take) \right)$$

$$= \pi_{t1.*} \; \sigma_{t1.sid = t2.sid \; \wedge \; t1.cid \, != \, t2.cid} \left(\rho_{t2}(take) \right)$$

2nd step. Find the result via using the subquery that we found at 1st step.

simplify.

$$(student) \bowtie \pi_{t1.*} \; \sigma_{t1.sid = t2.sid \; \wedge \; t1.cid \, != \, t2.cid \; \wedge \; student.sid = t1.sid} \left(\rho_{t2}(take) \times student \right)$$

RESULT:

$$\pi_{student.*} \; \sigma \left[\pi_{student.*} \; \sigma_{t1.sid = t2.sid \; \wedge \; t1.cid \, != \, t2.cid \; \wedge \; student.sid = t1.sid} \; \rho_{t2}(take) \right]$$

SORU-9

9. Hiç ders vermeyen (take tablosunda bu hocaya ait kayıt yok demektir) hocaları listeleyiniz.

        SELECT *

        FROM teacher

        WHERE tid NOT IN(

                SELECT tid

                FROM teach

        );

```
+-----+-------+-------+------------+------------+------+
| tid | fname | lname | birthdate  | birthplace | did  |
+-----+-------+-------+------------+------------+------+
|  8  | Enver | Usta  | 1999-07-10 | istanbul   |  1   |
|  9  | Ahmet | Demir | 0000-00-00 | sakarya    |  1   |
+-----+-------+-------+------------+------------+------+
```


SORU-10

10. Ders veren hocaların kayıtlarını (teacher tablosundaki tüm sütunları) listeleyiniz. (Yani take tablosunda tid geçen tüm hocalar)

```sql
SELECT *
FROM teacher
WHERE tid IN(
        SELECT tid FROM teach
);
```

```
+-----+--------+----------+------------+------------+------+
| tid | fname  | lname    | birthdate  | birthplace | did  |
+-----+--------+----------+------------+------------+------+
|  1  | Selami | Durgun   | 1970-11-10 | amasya     |  1   |
|  2  | Cengiz | Tahir    | 1960-07-21 | istanbul   |  1   |
|  3  | Derya  | Seckin   | 1962-05-15 | mersin     |  1   |
|  4  | Dogan  | Gedikli  | 1965-11-16 | istanbul   |  2   |
|  5  | Ayten  | Kahraman | 1968-12-11 | istanbul   |  3   |
|  6  | Tahsin | Ugur     | 1963-12-12 | izmir      |  4   |
|  7  | Selcuk | Ozan     | 1960-10-30 | amasya     |  4   |
+-----+--------+----------+------------+------------+------+
```

10. soru.

1st step : write the subquery

$$\pi_{teach.tid} \; \sigma_{teach.tid = teacher.tid} \; (teach)$$

2nd step: Add context relations and context parameters.

$$\pi_{teacher.*} \; \sigma_{teach.tid = teacher.tid} \; (\, teach \times teacher \,)$$
teach.tid

3th step: Add 'from clause'.

$$(teacher) \bowtie \pi_{teacher.*} \; \sigma_{teach.tid = teacher.tid} \; (teach \times teacher)$$
teach.tid

4th step: Simplification is possible.

$$\pi_{teacher.*} \; \sigma_{teach.tid = teacher.tid} \; (teach) \times teacher)$$

Result = 5th step : Add select and where.

$$\pi_{teacher.*} \; \sigma \left[\pi_{teacher.*} \; \sigma_{teach.tid = teacher.tid} \; (teach) \right]$$