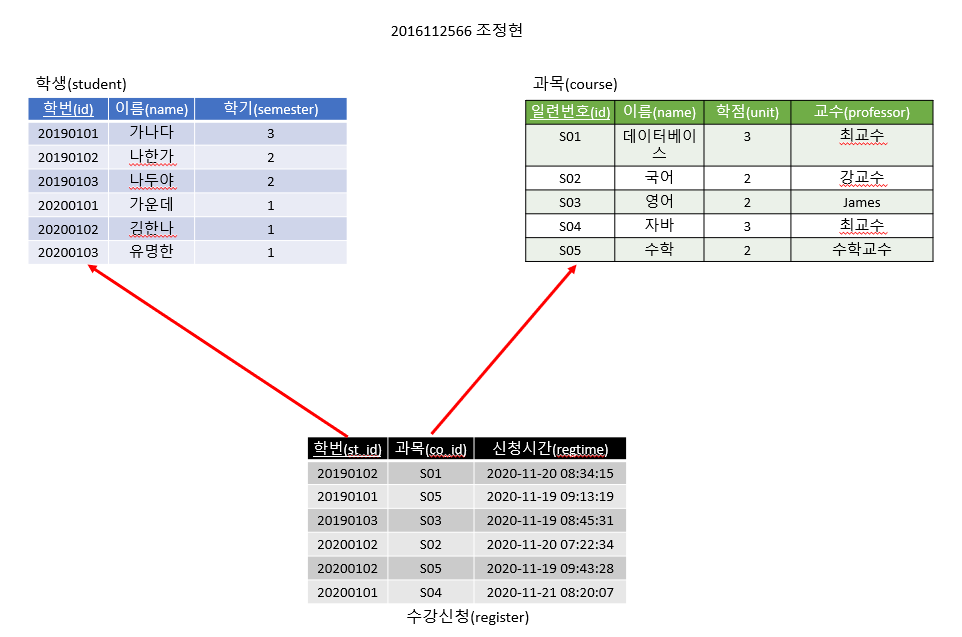
1. E-R 다이어그램 설명을 듣고, 릴레이션을 작성해보세요. 각자의 연습장을 이용해서 만들어봅니다. 

2. CREATE DATABASE 명령을 이용해서 dgu 이름의 데이터베이스를 생성합니다.

MariaDB [dgu]> CREATE DATABASE dgu;

3. CREATE TABLE 명령을 이용해서 해당 릴레이션들을 테이블로 만듭니다.

MariaDB [dgu]> CREATE TABLE student

-> (id CHAR(8), name VARCHAR(20), semester INT, PRIMARY KEY(id));

4. 다음과 같은 학생들을 추가합니다.

20190101 가나다 3

20190102 나한가 2

20190103 나두야 2

20200101 가운데 1

20200102 김한나 1

20200103 유명한 1

그외에 규칙에 맞게 여러명의 학생들을 테이블에 추가합니다.

MariaDB [dgu]> INSERT INTO student VALUES

->(‘20190101’, ‘가나다’, 3)

->(‘20190102’, ‘나한가’, 2)

->(‘20190103’, ‘나두야’, 2)

->(‘20200101’, ‘가운데’, 1)

->(‘20200101’, ‘김한나’, 1);

5. 다음과 같은 과목들을 추가합니다.

S01 데이터베이스 3 최교수

S02 국어 2 강교수

S03 영어 2 James

S04 자바 3 최교수

S05 수학 2 수학교수

그 외에 규칙에 맞게 여러개의 과목들을 테이블에 추가합니다.

MariaDB [dgu]> INSERT INTO course VALUES

->(‘S01’, ‘데이터베이스’, 3, ‘최교수’)

->(‘S02’, ‘국어’, 2, ‘강교수’)

->(‘S03’, ‘영어’, 2, ‘James’)

->(‘S04’, ‘자바’, 3, ‘최교수’)

->(‘S05’, ‘수학’, 2, ‘수학교수’)

6. 김한나 학생은 데이터베이스 과목을 수강하고자 합니다. 김한나 학생이 데이터베이스

과목을 수강 신청하는 것을 SQL로 작성하여 실행해보세요.

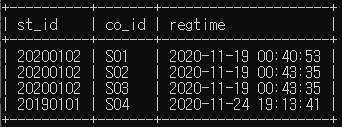
MariaDB [dgu]> INSERT INTO register

-> (SELECT st.id, co.id, now() FROM student st, course co WHERE

-> st.name = '김한나' AND co.name = '데이터베이스');

7. 수강신청 테이블에 올바르게 수강신청이 되었는지 확인해봅니다.

MariaDB [dgu]> SELECT \* FROM register;



8. 김한나 학생이 국어와 영어를 수강신청을 추가했습니다. 이것을 SQL로 작성하여 실행해보세요.

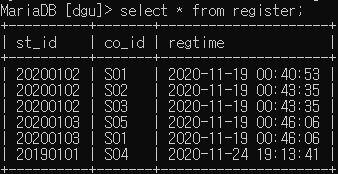
MariaDB [dgu]> INSERT INTO register

-> (SELECT st.id, co.id, now() FROM student st, course co WHERE

-> st.name = '김한나' AND (co.name = '국어' OR co.name = ‘영어’);

9. 수강신청 테이블에 올바르게 수강신청이 되었는지 확인해봅니다.

MariaDB [dgu]> SELECT \* FROM register;



10. 다른 학생들에 대해서도 과목 수강신청을 하도록 합니다.

MariaDB [dgu]> INSERT INTO register VALUES

-> (‘20200103’, ‘S05’, NOW()),

-> (‘20200103’, ‘S01’, NOW()),

-> (‘20190101’, ‘S04’, NOW());

11. 김한나 학생이 수강신청한 과목들 정보를 과목이름, 학점, 교수 순으로 출력하되, 과목이름으로 정렬해서 출력합니다.

MariaDB [dgu]> SELECT co.name, co.unit, co.professor

-> FROM student st, course co, register re

-> WHERE st.name = '김한나' AND st.id = re.st\_id AND re.co\_id= co.id ORDER BY co.name;

12. 수강신청한 학생들의 과목수와 총학점수를 출력하는 SQL문을 작성합니다.

MariaDB [dgu]> SELECT st.name, COUNT(\*) AS 과목수, SUM(co.unit) AS 총학점

-> FROM student st, register re, course co

-> WHERE st.id = re.st\_id AND co.id = re.co\_id

-> GROUP BY st.name;

13. 김한나 학생이 수강신청한 국어를 수강신청에서 제외하려고 합니다. 국어 수강신청에서 제외하는 SQL문을 작성하고 실행합니다.

MariaDB [dgu]> DELETE FROM register

-> WHERE st\_id = (SELECT id FROM student WHERE name = '김한나')

-> AND co\_id = (SELECT id FROM course WHERE name = '국어');

14. 수강을 신청한 학생들은 이번학기에 등록을 한 학생들입니다. 이들 학생들의 학기를 1씩 증가시켜주는 SQL문을 작성해서 실행합니다.

MariaDB [dgu]> UPDATE student

-> SET semester = semester + 1

-> WHERE id in (SELECT DISTINCT st\_id FROM register);

15. 동국대에서는 수강신청 인원이 5명 미만인 수업은 폐강을 합니다. 5명 미만인 수업들에 대해서 삭제하는 SQL문을 작성해보세요.

MariaDB [dgu]> DELETE

-> FROM course

-> WHERE id IN

-> (SELECT re.co\_id FROM register re GROUP BY re.co\_id HAVING COUNT(\*) < 5);