DB\_20200708\_Alpha

Lab. Database

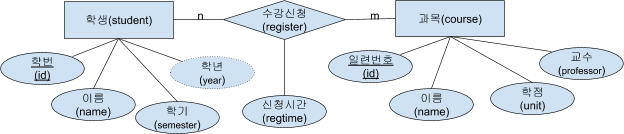
By Changrak Choi

Dongguk University

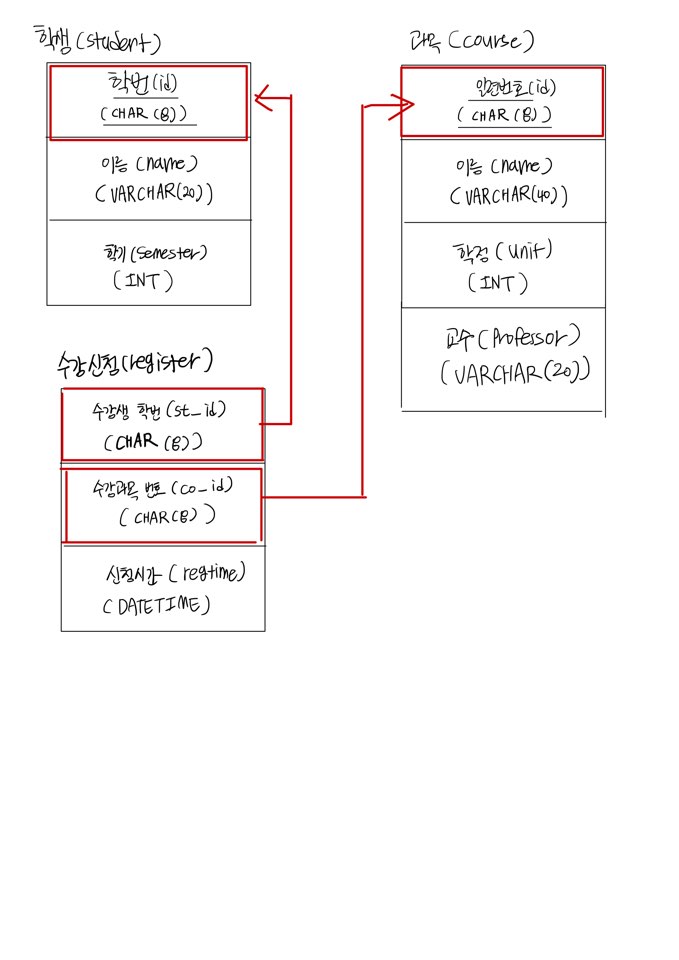
Lab 1. SQL 실습

동국대에서 수강신청을 받는 시스템을 만들고자 합니다. 워낙 다급하게 이루어진 일이라서, 학생 데이터와 수업 데이터, 그리고 수강 시스템을 여러분이 작성해야합니다.

E-R 다이어그램은 다음과 같습니다.



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



1. CREATE DATABASE 명령을 이용해서 dgu 이름의 데이터베이스를 생성합니다.

답: CREATE DATABASE dgu;

1. CREATE TABLE 명령을 이용해서 해당 릴레이션들을 테이블로 만듭니다.

답: CREATE TABLE student (id CHAR(8), name VARCHAR(20), semester INT, PRIMARY KEY(id));

CREATE TABLE course (id CHAR(8), name VARCHAR(40), unit INT, professor VARCHAR(20), PRIMARY KEY(id));

CREATE TABLE register (st\_id CHAR(8), co\_id CHAR(8), regtime DATETIME);

1. 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.

답: INSERT INTO student VALUES

-> ('20190101','가나다',3),

-> ('20190102','나한가',2),

-> ('20190103','나두야',2),

-> ('20200101','가운데',1),

-> ('20200102','김한나',1),

-> ('20200103','유명한',1),

-> ('20200104','학교가',1),

-> ('20200105','이명성',1),

-> ('20200106','한박자',1);

1. 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.

답: INSERT INTO course VALUES

-> ('S01', '데이터베이스', 3, '최교수'),

-> ('S02', '국어', 2, '강교수'),

-> ('S03', '영어', 2, 'James'),

-> ('S04', '자바', 3, '최교수'),

-> ('S05','수학', 2, '수학교수'),

-> ('S06','영양학',3,'조교수'),

-> ('S07','생물',2,'이교수'),

-> ('S08','천문학',3,'오교수');

1. 김한나 학생은 데이터베이스 과목을 수강하고자 합니다. 김한나 학생이 데이터베이스 과목을 수강 신청하는 것을 SQL로 작성하여 실행해보세요.

답: INSERT INTO register

-> (SELECT st.id, co.id, now()

-> FROM student st, course co

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

1. 수강신청 테이블에 올바르게 수강신청이 되었는지 확인해봅니다.
2. 김한나 학생이 국어와 영어를 수강신청을 추가했습니다. 이것을 SQL로 작성하여 실행해보세요.

답: 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'));

1. 수강신청 테이블에 올바르게 수강신청이 되었는지 확인해봅니다.
2. 다른 학생들에 대해서도 과목 수강신청을 하도록 합니다.

답: INSERT INTO register

-> (SELECT st.id, co.id, now()

-> FROM student st, course co

-> WHERE st.name = '가나다' AND (co.name = '국어' OR co.name = '영어'));

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

답: SELECT co.name as 과목이름, co.unit as 학점, co.professor as 교수

-> FROM student st, register re, course co

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

-> ORDER BY co.name;

1. 수강신청한 학생들의 과목수와 총학점수를 출력하는 SQL문을 작성합니다.

답: SELECT st.name AS 이름, COUNT(\*) AS 과목수, SUM(co.unit) AS 학점

-> FROM student st, register re, course co

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

-> GROUP BY st.name ;

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

답: DELETE FROM register

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

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

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

답: UPDATE student

SET semester=semester+1

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

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

답: Delete from register

-> WHERE register.co\_id

-> in(SELECT register.co\_id

-> FROM register

-> GROUP BY register.co\_id

-> HAVING count(\*)<5);