

데이터 베이스 기초 과제#2

2020.03.30. 월 B반송명훈

- 1. [복습]입시학원 DB 모델링 해보기.
- 2. [예습]
 - ① 오전 실습과제로 조사한 DML, DDL 명령어 8건의 예제를 실습 후 화면 캡쳐 하기
 - ② 명령문을 입력하고 결과화면 캡쳐
- 3. [예습] 2번 문제의 예문들이 어떤 명령에 대한 결과인지 조사
- 4. [예습] Data type 조사



과제 #1. 입시학원 DB모델링

과제 #1. 입시학원 DB 모델링

실습 #1. 문제

- 입시학원 DB 모델링 해보기
 - 1. 필수 테이블
 - 1. 학원생 명부: No, 이름, 전화번호, 주소, 학교, 학년 등이 포함되어야함
 - 2. 선생님 명부: No, 이름, 과목, 전화번호, 주소 등이 포함되어야함
 - 3. 개설반 목록: No, 반이름, 시간, 수강료, 강의실 등이 포함되어야함
 - 4. 그외

2. 조건

- 1. Table 4개 이상
- 2. Table 별 Field는 7개 이상
- 3. Table 별 Record는 5건 이상

과제 #1. 입시학원 DB 모델링

Student											
No	Name	Phone nu	number		Address		School		Grade	Parent Phone number	
1	김민찬	010-1547	-9875		경기 수원시 영통구 영통로 498		청명중학교		2	010-1547-7811	
2	김영주	010-6543	-8715		경기 수원시 영통구 영통로 460		청명중학교		3	010-6543-1145	
3	신가람	010-6549	-9652		경기 수원시 영통구 광교중앙로 55		소화초등학교		6	010-6549-3215	
4	주대훈	010-1789	-9432	경기 수원시 영통구 매봉로27번길 11			산남초등학교		5	010-1789-1455	
5	이신형	010-9874	-4962		경기 수원시 영통구 매영로 132		매원고등학교		1	010-9874-7784	
Teacher											
No	Name	Subject Phone		number	er Address		Gender Unive		ersity	Major	
1	김선옥	수학	010-3067-5557		경기 수원시 영통구 영통로290번길 26	female		서울대		수학과	
2	김성수	수학	010-4456-1663		경기 수원시 영통구 청명로 132	male		경인교대		수학교육과	
3	강호동	영어	1 010-7844-		경기 수원시 영통구 청명로 100 male		nale	서울교대		영어교육과	
4	이하늬	국어	010-4213-7744		경기 수원시 영통구 청명북로 33	female		전주교대		국어교육과	
5	김수영	과학 010-6652-811		552-8113	경기 수원시 영통구 매영로310번길 36		male	nale 한양대		화학공학과	
6	Josua Smit	h 영어	010-5177-0913		경기 수원시 영통구 매영로 346	male		Georgia State Univ, USA		English Language and literature	

과제 #1. 입시학원 DB 모델링

Class										
No	Name -		Ti	ime	Price	Class room	Teacher	Grade		
1	고등수학1		20:00~21:20		300,000	1	김성수	고등1		
2	중등수학2 1		16:50~17:50		210,000	2	김성수		중3	
3	영어 A반 15		15:00~16:00		210,000	3	강호동		중급	
4	초등국어		14:30~15:30		150,000	4	이하늬		초등6	
5	중등과학		18:00~19:00		180,000	5	김수영		중2	
6	원어민영어		16:30~17:39		210,000	3	Josua Smith	초	초등~ 중등 전체	
	Payment									
N	0	date		student	payment teacher number		Class number	amount	total amount	
1	202	2020-03-27 13:01		이신형	1		1	300,000	900,000	
2	2 2020-0		27 13:45 김영주		1		2	210,000	630,000	
3	3 2020-03-		27 13:46 김영주		1		1 3		840,000	
4	202	2020-03-28 19:23		신가람	1		1 4		450,000	
5	5 2020-03-2		28 20:21	김민찬	2		2 5		720000	
6	6 2020-03-		29 15:12	주대훈	2		2 6 2		1050000	

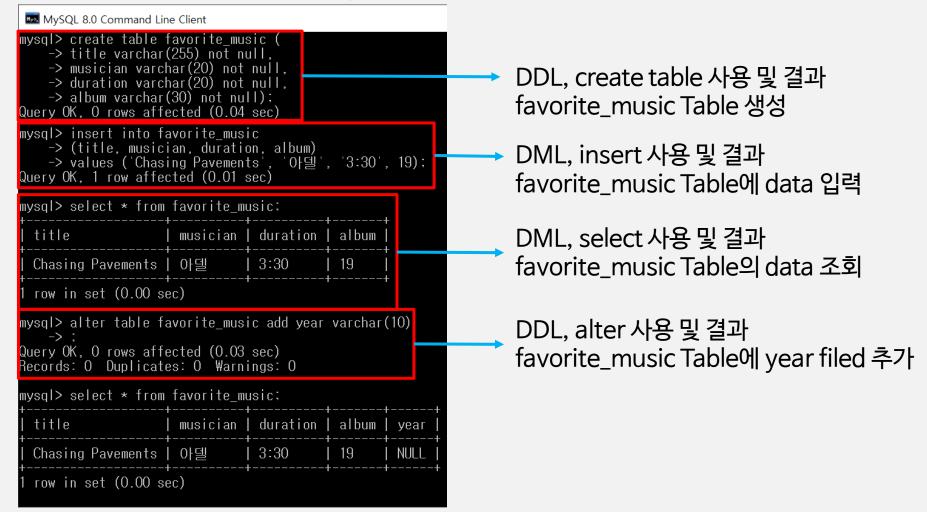


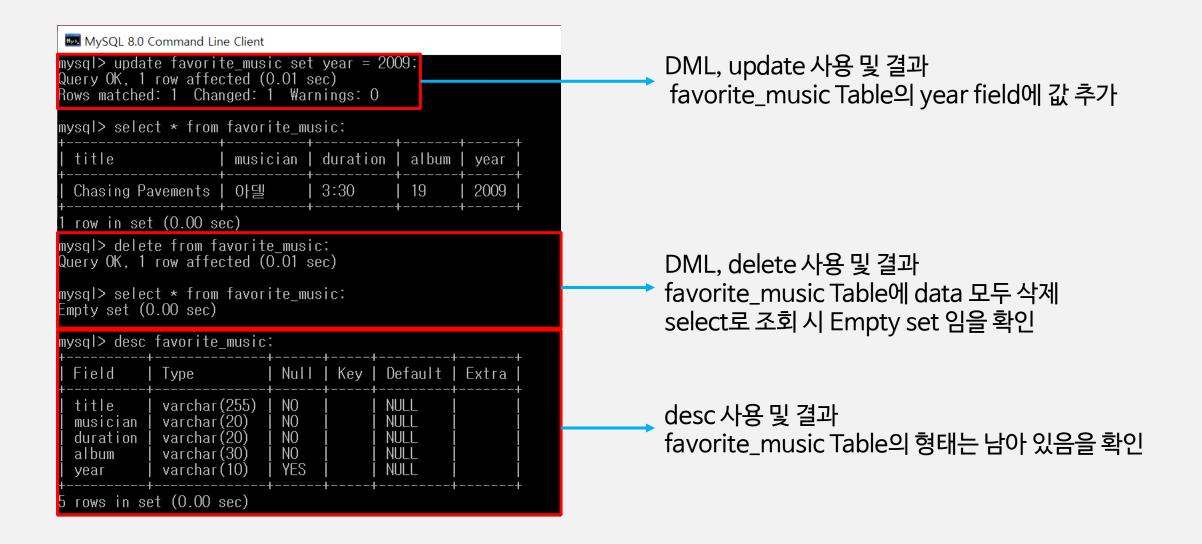
과제 #2. DML, DDL 실습

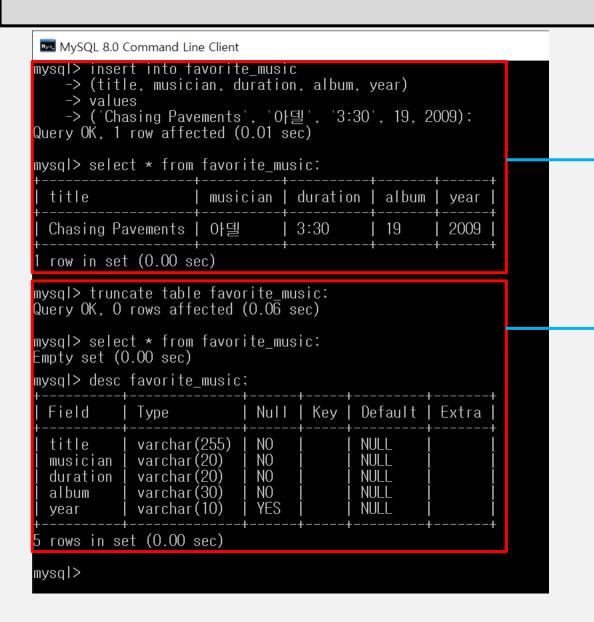
& 명령문 입력 실습

실습 #2. 문제

• 오전 실습 과제로 조사한 DML, DDL 명령어 8건의 예제를 실습 후 화면 캡쳐하기







► 다시 테이블에 값 추가

DDL, truncate 사용 및 결과 favorite_music table의 data가 삭제됨 favorite_music table의 형태는 남아 있음.

```
MySQL 8.0 Command Line Client
mysql> insert into favorite_music
     -> (title, musician, duration, album, year)
    -> values
-> ('Chasing Pavement', 'Adele', '3:30', 19, 2009);
Query OK, 1 row affected (0.01 sec)
mysgl> select * from favorite music;
  title
                         musician | duration | album |
                                                             vear
  Chasing Pavement | Adele
                                      3:30
                                                              2009
  row in set (0.00 sec)
mysql> drop table favorite_music;
Query OK, O rows affected (0.06 sec)
mysql> select * from favorite_music;
ERROR 1146 (42SO2): Table 'music.favorite_music' doesn't exist
mysql> desc favorite_music;
EŔROR 1146 (42SO2): Table 'music.favorite_music' doesn't exist
mysql>
```

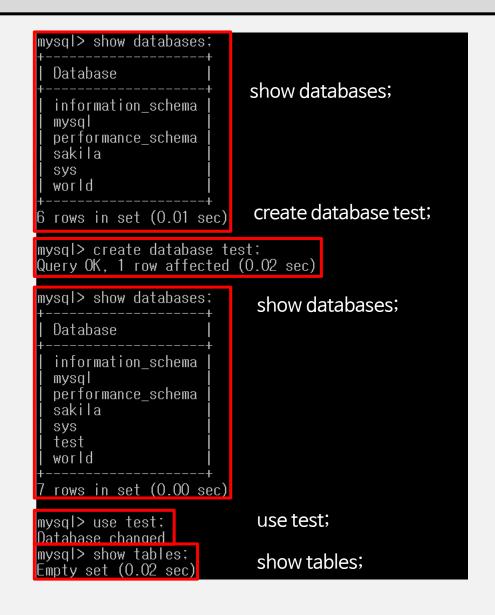
▶ 다시 테이블에 값 추가

DDL, drop 사용 및 결과 favorite_music table의 table 전체가 삭제됨 favorite_music table의 형태도 삭제됨.

과제 #2. 명령문 실습

실습 #2. 문제

- 아래 동작을 하는 명령문을 입력하고 결과화면을 캡쳐하기
 - 1) show databases;
 - 2) create database test;
 - 3) show databases;
 - 4) use test;
 - 5) show tables;
 - 6) create table testtable (id int, name char (10));
 - 7) desc testtable;
 - 8) insert into testtable values (1, "test");
 - 9) select * from testtable;
 - 10) delete from testtable;
 - 11) insert into testtable values (2, "test");
 - 12) truncate testtable;
 - 13) select * from testtable;



```
mysgl> create table testtable(
                                        create
     -> id int, name char(10)
                                        table testtable
     -> ):
 Query OK, O rows affected (0.13 sec)
                                desc testable;
  nysgl> desc testtable;
  Field | Type
                              | Key | Default | Extra
            int(11)
                        YES
                                        NULL
          NULL
   name
 2 rows in set (0.02 sec)
 mysql> insert into testtable insert into testtable;
-> values(1, "test");
Query OK, 1 row affected (0.01 sec)
 mvsal> select * from testtable;
                               select * from testtable;
       1 | test
  row in set (0.00 sec)
                                -delete from testtable;
 mysql> delete from testtable;
 Query OK, 1 row affected (0.01 sec)
 mysql> insert into testtable insert into testtable;
-> values(2, "test");
Query OK, 1 row affected (0.01 sec)
mysql> truncate testtable; truncate testas...
Query OK, O rows affected (0.05 sec)
select * from testtable;
mysql> select * from testtable;
  mptv set (0.01 sec)
```



과제 #3. 2번 과제 예문들이 어떤 명령에 대한 결과인지 조사하기

과제 #3. 2번 과제 예문들에 사용된 명령들이 어떤 것인지 조사

사용된 명령들

- 1) show databases;
 - SHOW: 데이터베이스 목록이나 테이블 목록 등 다양한 정보를 보기 원할 때 사용하는 명령어

 - SHOW TABLES; ⇒ 현재 데이터베이스의 테이블 목록 모두 출력
 - SHOW TABLES FROM 데이터베이스; ⇒ 특정 데이터베이스의 테이블 목록 출력
- 2) create database test;
 - DDL인 CREATE 명령어 사용해 test 라는 이름의 database를 생성하라는 명령
- 3) use test;
 - 2)에서 생성한 test 데이터 베이스를 사용하도록 설정하는 명령
- 4) show tables;
 - 현재 데이터베이스인 test 데이터베이스 내 존재하는 table을 모두 출력

과제 #3. 2번 과제 예문들에 사용된 명령들이 어떤 것인지 조사

사용된 명령들

- 5) create table testtable (id int, name char (10));
 - 현재 데이터베이스에 testtable 이름을 갖는 table 생성
 - testtable은 INT형의 id FIELD, CHAR 형 10bytes크기의 name FIELD를 갖는다
- 6) desc testtable;
 - DESCRIBE을 줄여서 DESC로 사용할 수 있다. ORDER BY절의 desc와 구별
 - testtable의 구조에 대한 정보를 출력 ⇒ 테이블의 FIELD이름, NULL 조건 여부, 데이터 타입 확인 가능
- 7) insert into testtable

```
values(1, "test");
```

- DML 인 INSERT 명령어를 사용해 testtable table에 (1, "test") 데이터를 삽입하라는 명령
- 8) select * from testtable;
 - DML인 SELECT 명령어를 사용해 testtable table의 모든 항목을 조회하는 명령어
 - 사용시 testtable의 모든 내용이 console에 출력된다.

과제 #3. 2번 과제 예문들에 사용된 명령들이 어떤 것인지 조사

사용된 명령들

- 9) delete from testtable;
 - DML인 delete 명령어를 사용해 testtable table의 내용을 모두 삭제하라는 명령
 - testtable table 자체가 삭제되는 것은 아니기 때문에 describe 사용 시 table의 구조가 출력된다.

10) truncate testtable;

- DDL인 TRUNCATE 명령어 사용해 testtable table의 내용을 모두 삭제하라는 명령
- DELETE와 마찬가지로 table 자체가 삭제되는 것은 아니므로 DESCRIBE 사용 시 table의 구조가 출력된다.

※ DELETE와 TRUNCATE의 차이점

- 두 명령어 모두 해당 테이블의 데이터를 삭제하지만 DELETE는 뒤에 WHERE 절을 통해 조건에 맞는 일부 데이터만 삭제하는 것이 가능하고 TRUNCATE는 조건절의 부여 없이 테이블의 모든 데이터를 삭제한다.
- 가장 큰 차이점은 데이터를 삭제하는 방식이다.
 DELETE는 데이터를 한 줄 씩 순차적으로 삭제하고 TRUNCATE는 테이블을 DROP 후 CREATE를 수행한다. 이는 두명령어 간의 목적 차이로 발생한다. DELETE는 해당 데이터를 삭제하고 삭제한 공간을 재사용하기 위한 명령어이고 TRUNCATE는 테이블 명세만을 남기고 데이터가 존재하던 공간마저 제거하기 위한 명령어이다.
- 두 명령어의 실행 속도는 TRUNCATE가 더 빠르지만, 복구가 불가능하다는 제한사항이 존재한다.



과제 #4. Data type 조사

과제 #4. Data type 조사

• INT, FLOAT, DOUBLE, CHAR

자료형	크기	범위				
INT	4 Bytes	signed: -2147483648 ~ 2144483647 unsigned: 0 ~ 4294967295				
FLOAT	4 Bytes	±1.175494351E-38 ~ ±3.402823466E+38				
DOUBLE	8 Bytes	± 1.7976931348623157E-308 ~ ± 2.2250738585072014E+308				
CHAR	(지정한 글자수) bytes (한글 제외)	0 ~ 255 글자				

X CHAR, VARCHAR

- 둘 다 문자열 자료형이나 CHAR는 고정형 문자열, VARCHAR는 가변형 문자열이다.
- 둘 다 사용 시 길이를 명시해주어야 한다.
- CHAR의 경우, CHAR(20)일 때 2 byte의 문자만 넣어도 20byte 만큼을 차지하지만
- VARCHAR의 경우, VARCHAR(20)일 때 2byte의 문자를 넣으면 2byte 만큼만 차지하게 된다.
- 고정형인 CHAR 타입은 추후 연산이 필요 없기 때문에 검색속도 및 읽는 속도가 다른 타입보다 월등히 빠르다.