

5장 . 테이블 생성

1. MySQL RDBMS의 제한 조건
2. 새로운 테이블의 생성
3. 기본 키
4. 테이블 명세와 소유자
5. 테이블과 열의 이름
6. 테이블 삭제
7. 열의 추가
8. 테이블 구조 변경
9. 테이블 복사
10. 이름변경
11. 데이터 사전

5.1 MySQL RDBMS의 제한 조건

- 데이터베이스가 포함할 수 있는 Table의 수 : 무한대
- 테이블의 행의 갯수 : 무한대
- 테이블이 포함 할 수 있는 열의 갯수(필드수) : 254개까지
- 1개의 Row의 최대문자수(전체) : 130,306 Byte(Long 필드(2GB)는 제외)
- 문자열필드의 최대바이트수 : 255
- 숫자열필드의 최대Byte수 : 38
- 1개의 테이블에 만들 수 있는 Index수 : 무한대
- 쿼리시 결합 가능한 필드의 수 : 무한대
- SubQuery의 하위단계별수 : 255개까지
- 명칭의 문자수 : 30Byte

● DDL : Data Definition Language 명령문의 종류

- CREATE TABLE : 테이블을 생성
- ALTER TABLE : 테이블과 열(Column) 변경
- DROP TABLE : 테이블 삭제
- RENAME : 테이블 이름 변경

5.2 새로운 테이블의 생성

CREATE TABLE 명령문은 데이터 행이 저장될 수 있는 테이블을 생성

SUBJECT(교과목) 테이블을 생성하는 CREATE TABLE 명령문의 예제

```
mysql> create table subject(  
    -> sub_code char(5) Not null,           #과목번호  
    -> sub_name varchar(50) Not null,       #과목명  
    -> sub_ename varchar(50),               #영문과목명  
    -> create_year char(4),                 #개설년도  
    -> primary key (sub_code)  
    -> )engine = innodb;
```

- 테이블 이름은 SUBJECT : DB내에 유일한 이름, 입력한 사람이 소유자(OWNER)
- 열의 정의(column definition)는 sub_code, sub_name ...등
- 열에 NOT NULL 무결성 규칙이 지정되면 반듯이 데이터 값 입력
- sub_ename처럼 NOT NULL이 생략되면 행 삽입시 데이터 값 생략해도 됨
- primary key 정의는 테이블생성시 정의된 열(column)의 이름 사용

5.3 기본 키

- 기본 키(primary key)는 값이 항상 유일한 열 또는 열의 집합
- 기본 키로 사용하는 열에는 NULL값을 허용하지 않는다.
- 여러 개의 열을 사용하여 기본 키를 정의 : 조합키(composite primary key)

[예제 5-1] 과목명, 학생이름, 과목인원, 과목 개설 일자를 기록하기 위한 DIPLOMAS 테이블을 생성하라. COURSE, STUDENT, END_DATE 열을 사용하여 조합 기본 키를 생성한다.

```
CREATE TABLE DIPLOMAS
(COURSE VARCHAR(20) NOT NULL,
 STUDENT VARCHAR(10) NOT NULL,
 COU_NUM INT(2),
 END_DATE DATE NOT NULL,
 PRIMARY KEY (COURSE, STUDENT, END_DATE));
```

- 3개의 열을 사용하여 기본 키 정의 : 학생은 지정된 일자에 한 과목에 대하여 취득 가능
- 각 테이블에는 오직 하나의 기본 키만 정의할 수 있다.
- 관계형 모델에서는 각 테이블에 하나의 기본 키를 정의하도록 하고 있다.

- 기본 키(primary key) 없이 테이블 생성가능
- 기본 키로 테이블을 갱신할 때 유일성을 조사

- 무결성 규칙(first integrity rule) 또는 엔티티 무결성 규칙(entity integrity rule)

- 유일성 규칙(uniqueness rule) : 테이블에서 서로 다른 행은 동일한 값을 기본 키로 가질 수 없다.
- 최소화 규칙(minimal rule) : 기본 키는 불필요하게 많은 열로 구성하지 않아야 한다.
 - STU_NO과 STU_NAME을 기본 키로 정의했다고 가정해 보자. 우리는 학생의 학번이 유일하다는 것을 이미 알고 있다. 따라서 이러한 경우에 기본 키는 필요 이상의 열을 가지고 있으므로 최소화 규칙을 만족하지 않는다.
- 기본 키 열 목록에서 열의 이름은 오직 한번만 나타나야 한다.
- 기본 키에 포함되는 열의 집단은 NULL 값을 가지지 않아야 한다.

● DIPLOMAS 테이블 데이터 입력

[예제 5-2] DIPLOMAS 테이블에 다음의 데이터를 입력하라.

```
mysql> insert into diplomas values('웹프로그래밍','공자',2,'2007/07/25');  
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into diplomas values('웹프로그래밍','맹자',3,'2007/07/25');  
Query OK, 1 row affected (0.00 sec)
```

```
mysql> SELECT * FROM DIPLOMAS;
```

COURSE	STUDENT	COU_NUM	END_DATE
웹프로그래밍	공자	2	2007-07-25
웹프로그래밍	맹자	3	2007-07-25

2 rows in set (0.00 sec)

```
mysql>
```

5.4 테이블 명세와 소유자

- SOL라는 사용자가 소유자인 STUDENT 테이블을 접근하는 질의

```
SELECT * FROM SOL.STUDENT;
```

소유자 테이블이름

- 사용자 자신이 소유자인 STUDENT 테이블을 접근하는 질의(소유자 생략)

```
SELECT * FROM STUDENT;
```

[예제 5-2] 사용자가 SOL일 때 LEE의 HELP 테이블을 생성하라.

```
CREATE TABLE LEE.HELP  
( ... );
```

5.5 테이블과 열의 이름

-사용자는 테이블과 열의 이름을 자유롭게 사용할 수 있다.

● 테이블과 열의 이름 사용시 제약사항

- MySQL에서 하나의 스키마는 2개의 테이블을 동일한 이름으로 가질 수 없다.
- 하나의 테이블에서 2개의 열이 동일한 이름을 가질 수 없다.
- 테이블 또는 열의 이름 길이는 64자로 제한되어 있다.
- 이름은 문자, 숫자, 특수 기호 _(underscore), \$, #로 구성, 첫 문자는 문자로
- 테이블과 열의 이름으로 예약어를 사용할 수 없다.

● 테이블과 열의 이름은 의미 있는 이름을 선택

- 테이블과 열의 이름을 짧게 사용하라.
- 정보의 의미가 있는 이름은 사용하지 말라.
(STUDENT_2 보다는 STUDENT를 사용, 2는 인덱스의 수로 사용)
- 일관성을 유지하라. (stu_no 사용)
- 너무 긴 이름은 피하라. (STUDENTNAME 보다는 STU_NAME 사용)
- 가능하다면 열의 이름들은 각 테이블에서 사용된 동일한 이름으로 사용하라.
(STUDENT 테이블에서 STU_NO, ATTEND 테이블에서 STU_NO)

5.6 테이블의 삭제

- DROP TABLE 명령문은 테이블을 삭제하기 위해서 사용
- 카탈로그 테이블로부터 테이블의 명세표, 인덱스, 데이터 값 그리고 테이블에 부여된 권한을 제거

[예제 5-3] 예제 5-1에서 생성된 DIPLOMAS 테이블을 삭제하라.

```
mysql> DROP TABLE DIPLOMAS;
```

5.7 테이블 구조 변경(열의 추가)

- 테이블의 구조를 변경은 ALTER TABLE 명령문을 사용

[예제 5-4] DIPLOMAS 테이블을 출력하라.

```
mysql> select * from diplomas;
```

COURSE	STUDENT	COU_NUM	END_DATE
웹프로그래밍	공자	2	2007-07-25
웹프로그래밍	맹자	3	2007-07-25

```
2 rows in set (0.00 sec)
```

[예제 5-5] DIPLOMAS테이블에 성별 SEX를 CHAR(2)로 열을 추가

```
mysql> ALTER TABLE DIPLOMAS ADD SEX CHAR(2);
```

```
mysql> select * from diplomas;
```

COURSE	STUDENT	COU_NUM	END_DATE	SEX
웹프로그래밍	공자	2	2007-07-25	NULL
웹프로그래밍	맹자	3	2007-07-25	NULL

2 rows in set (0.00 sec)

5.8 테이블 구조 변경

● ALTER TABLE 명령문의 기능

- 테이블에 새로운 열을 추가한다.
- 기존에 존재하는 열을 삭제한다.
- 자료형의 길이를 변경한다.
- 기존에 존재하는 열의 자료형을 특별한 조건하에서 변경할 수 있다.
- 기본 키와 같은 새로운 무결성 규칙을 추가할 수 있다.
- 기존에 존재하는 무결성 규칙을 삭제할 수 있다.

[예제 5-5] DIPLOMAS 테이블에 성별 GENDER를 CHAR(2)로 열을 추가하라.

```
mysql> ALTER TABLE DIPLOMAS ADD GENDER CHAR(2);
```

DIPLOMAS 테이블을 확인하여 보자.

```
mysql> select * from diplomas;
```

COURSE	STUDENT	COU_NUM	END_DATE	GENDER
웹프로그래밍	공자	2	2007-07-25	NULL
웹프로그래밍	맹자	3	2007-07-25	NULL

2 rows in set (0.00 sec)

[예제 5-6] DIPLOMAS 테이블의 GENDER 열의 길이를 2에서 4로 증가시켜라.

```
mysql> ALTER TABLE DIPLOMAS MODIFY GENDER CHAR(4);
```

[예제5-7] DIPLOMAS테이블에서 GENDER의 자료형을 CHAR에서 INT로 변경하라.

```
mysql> ALTER TABLE DIPLOMAS MODIFY GENDER INT;
```

[예제 5-8] DIPLOMAS 테이블에 기본 키(STUDENT)를 추가하라.

```
mysql> ALTER TABLE DIPLOMAS ADD PRIMARY KEY (STUDENT);
```

5.9 테이블 복사

-CREATE TABLE문과 AS 키워드를 이용하여 새로운 테이블을 복사

[예제 5-9] STUDENT 테이블의 Owner가 “**SOL**”인 경우 STUDENT 테이블과 동일한 테이블 구조와 데이터를 가지고 있는 Owner가 “**SOL**”인 사람이 STUDENT1 테이블을 복사하라.

```
mysql> CREATE TABLE STUDENT1 AS SELECT * FROM STUDENT;
```

[예제 5-10] STUDENT 테이블의 Owner가 “**SOL**”인 경우 사용자 Owner가 “**LE**”인 STUDENT2 테이블로 동일한 테이블 구조와 데이터를 복사하라.

```
mysql> CREATE TABLE LEE.STUDENT2 AS SELECT * FROM SOL.STUDENT;
```

[예제 5-11] SUBJECT테이블에서 SUB_CODE와 SUB_NAME으로 구성된 신규 SUBJECT_COPY 테이블을 생성하라.

```
mysql> CREATE TABLE SUBJECT_COPY AS SELECT SUB_CODE, SUB_NAME FROM SUBJECT;
```

[예제 5-12] SUBJECT_COPY 테이블의 모든 열을 조회하라.

```
mysql> select * from SUBJECT_COPY;
```

SUB_CODE	SUB_NAME
4001	데이터베이스 응용
4002	웹사이트 구축
4003	소프트웨어공학
4004	웹프로그래밍
4005	컴퓨터구조
4006	정보처리실무
4007	UML
4008	운영체제
4009	객체지향프로그래밍
4010	윈도우즈 프로그래밍
4011	자바프로그래밍
4012	파이썬 프로그래밍
4013	스크래치 프로그래밍

13 rows in set (0.00 sec)

새롭게 생성된 “SUBJECT_COPY” 테이블의 데이터 유형을 알아보자.

```
mysql> desc SUBJECT_COPY;
```

Field	Type	Null	Key	Default	Extra
SUB_CODE	char(5)	NO		NULL	
SUB_NAME	varchar(50)	NO		NULL	

2 rows in set (0.00 sec)

5.10 테이블 이름 변경

- 테이블, 뷰, 시노님, 시퀀스 등의 MySQL 객체의 이름을 바꿀 수 있다.
- RENAME 명령문의 형식

RENAME OLDNAME TO NEWNAME;

[예제 5-13] “SUBJECT_COPY” 테이블을 “TEST_SUBJECT” 테이블명으로 변경하라.

```
mysql> alter table subject_copy rename test_subject;  
Query OK, 0 rows affected (0.02 sec)
```

5.11 테이블과 데이터 사전

1. 테이블 및 데이터 사전 정보 보기

```
mysql> use information_schema
Database changed
```

```
mysql> show tables;
```

Tables_in_information_schema
CHARACTER_SETS
COLLATION_CHARACTER_SET_APPLICABILITY
COLLATIONS
COLUMN_PRIVILEGES
COLUMN_STATISTICS
COLUMNS
ENGINES
EVENTS
VIEW_ROUTINE_USAGE
VIEW_TABLE_USAGE
VIEWS

```
66 rows in set (0.01 sec)
```

```
mysql> use mysql
Database changed
mysql> show tables;
```

Tables_in_mysql
columns_priv
component
db
default_roles
engine_cost
func
general_log
time_zone_name
time_zone_transition
time_zone_transition_type
user

```
33 rows in set (0.00 sec)
```


2. TABLE 뷰

```
mysql> desc tables;
```

Field	Type	Null	Key	Default	Extra
TABLE_CATALOG	varchar(64)	YES		NULL	
TABLE_SCHEMA	varchar(64)	YES		NULL	
TABLE_NAME	varchar(64)	YES		NULL	
TABLE_TYPE	enum('BASE TABLE','VIEW','SYSTEM VIEW')	NO		NULL	
ENGINE	varchar(64)	YES		NULL	
VERSION	int(2)	YES		NULL	
ROW_FORMAT	enum('Fixed','Dynamic','Compressed','Redundant','Compact','Paged')	YES		NULL	
TABLE_ROWS	bigint(21) unsigned	YES		NULL	
AVG_ROW_LENGTH	bigint(21) unsigned	YES		NULL	
DATA_LENGTH	bigint(21) unsigned	YES		NULL	
MAX_DATA_LENGTH	bigint(21) unsigned	YES		NULL	
INDEX_LENGTH	bigint(21) unsigned	YES		NULL	
DATA_FREE	bigint(21) unsigned	YES		NULL	
AUTO_INCREMENT	bigint(21) unsigned	YES		NULL	
CREATE_TIME	timestamp	NO		NULL	
UPDATE_TIME	datetime	YES		NULL	
CHECK_TIME	datetime	YES		NULL	
TABLE_COLLATION	varchar(64)	YES		NULL	
CHECKSUM	bigint(21)	YES		NULL	
CREATE_OPTIONS	varchar(256)	YES		NULL	
TABLE_COMMENT	varchar(256)	YES		NULL	

21 rows in set (0.01 sec)

3. COLUMN 뷰

```
mysql> desc columns;
```

Field	Type	Null	Key	Default	Extra
TABLE_CATALOG	varchar(64)	YES		NULL	
TABLE_SCHEMA	varchar(64)	YES		NULL	
TABLE_NAME	varchar(64)	YES		NULL	
COLUMN_NAME	varchar(64)	YES		NULL	
ORDINAL_POSITION	int(10) unsigned	NO		NULL	
COLUMN_DEFAULT	text	YES		NULL	
IS_NULLABLE	varchar(3)	NO			
DATA_TYPE	longtext	YES		NULL	
CHARACTER_MAXIMUM_LENGTH	bigint(21)	YES		NULL	
CHARACTER_OCTET_LENGTH	bigint(21)	YES		NULL	
NUMERIC_PRECISION	bigint(10) unsigned	YES		NULL	
NUMERIC_SCALE	bigint(10) unsigned	YES		NULL	
DATETIME_PRECISION	int(10) unsigned	YES		NULL	
CHARACTER_SET_NAME	varchar(64)	YES		NULL	
COLLATION_NAME	varchar(64)	YES		NULL	
COLUMN_TYPE	mediumtext	NO		NULL	
COLUMN_KEY	enum('', 'PRI', 'UNI', 'MUL')	NO		NULL	
EXTRA	varchar(60)	YES		NULL	
PRIVILEGES	varchar(154)	YES		NULL	
COLUMN_COMMENT	text	NO		NULL	
GENERATION_EXPRESSION	longtext	NO		NULL	
SRS_ID	int(10) unsigned	YES		NULL	

```
22 rows in set (0.00 sec)
```

4. USERS 뷰

```
mysql> use mysql
Database changed
mysql> desc user;
```

Field	Type	Null	Key	Default	Extra
Host	char(60)	NO	PRI		
User	char(32)	NO	PRI		
Select_priv	enum('N','Y')	NO		N	
Insert_priv	enum('N','Y')	NO		N	
Update_priv	enum('N','Y')	NO		N	
Delete_priv	enum('N','Y')	NO		N	
Create_priv	enum('N','Y')	NO		N	
Drop_priv	enum('N','Y')	NO		N	
Reload_priv	enum('N','Y')	NO		N	
Shutdown_priv	enum('N','Y')	NO		N	
Process_priv	enum('N','Y')	NO		N	
File_priv	enum('N','Y')	NO		N	
Grant_priv	enum('N','Y')	NO		N	
	:				
	:				
	:				
Password_reuse_time	smallint(5) unsigned	YES		NULL	
Password_require_current	enum('N','Y')	YES		NULL	
User_attributes	json	YES		NULL	

51 rows in set (0.01 sec)

[예제 5-14] 사용자 “ROOT” 사용자(USER), 인증문자열(AUTHENTICATION_STRING), 파일권한(FILE_PRIV)을 출력하라.

```
mysql> select user, authentication_string, file_priv
      -> from user
      -> where user = 'root';
```

user	authentication_string	file_priv
root	\$A\$005\$ 0/.WqU2>S APYS tMgMG/bRsRQOX.ERxy5TnjDTP2fMLb1PgqnBQy6nPB3	Y

1 row in set (0.00 sec)

5. Trigger 뷰

```
mysql> use information_schema
Database changed
mysql> desc triggers;
```

Field	Type	Null	Key	Default	Extra
TRIGGER_CATALOG	varchar(64)	YES		NULL	
TRIGGER_SCHEMA	varchar(64)	NO		NULL	
TRIGGER_NAME	varchar(64)	NO		NULL	
COLLATION_CONNECT	varchar(64)	NO		NULL	
DATABASE_COLLATION	varchar(64)	NO		NULL	

```
22 rows in set (0.01 sec)
```

6. VIEW 뷰

```
mysql> desc views;
```

Field	Type	Null	Key	Default	Extra
TABLE_CATALOG	varchar(64)	YES		NULL	
TABLE_SCHEMA	varchar(64)	YES		NULL	
TABLE_NAME	varchar(64)	YES		NULL	
VIEW_DEFINITION	longtext	YES		NULL	
CHECK_OPTION	enum('NONE', 'LOCAL', 'CASCADED')	YES		NULL	
IS_UPDATABLE	enum('NO', 'YES')	YES		NULL	
DEFINER	varchar(93)	YES		NULL	
SECURITY_TYPE	varchar(7)	YES		NULL	
CHARACTER_SET_CLIENT	varchar(64)	NO		NULL	
COLLATION_CONNECTION	varchar(64)	NO		NULL	

10 rows in set (0.00 sec)