■ 데이터 조작어 [DML : Data Manipulation Language]

사용자(응용프로그램)와 DBMS 간의 인터페이스 역할을 제공하며, 질의어(SQL)를 통해 조작한다.

0) table 생성 - CREATE

 # 실습을 위해 간단한 test 테이블을 생성한다.

```
mysql> create table test(
    -> name varchar(10) not null,
    -> age varchar(5) not null,
    -> phone varchar(20),
    -> no int not null auto_increment primary key);
Query OK, 0 rows affected (0.50 sec)

mysql> desc test;
+-------+-------------+------+-----+---------+----------------+
| Field | Type        | Null | Key | Default | Extra          |
+-------+-------------+------+-----+---------+----------------+
| name  | varchar(10) | NO   |     | NULL    |                |
| age   | varchar(5)  | NO   |     | NULL    |                |
| phone | varchar(20) | YES  |     | NULL    |                |
| no    | int(11)     | NO   | PRI | NULL    | auto_increment |
+-------+-------------+------+-----+---------+----------------+
4 rows in set (0.01 sec)
```

1) 레코드 삽입 - INSERT

mysql> insert into [테이블 명](Column_List) values(데이터1, 데이터2, ...);

 # Column을 선택하여 넣는 실습과 전체의 Column 의 값을 넣는 실습을 했다.

```
mysql> insert into test(name, age) values('minki', '25');
Query OK, 1 row affected (0.07 sec)

mysql> insert into test values('hyewon', '23', '010-9999-9999', 2);
Query OK, 1 row affected (0.04 sec)

mysql> select * from test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 25  | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)
```

2) 레코드 선택 - SELECT

mysql> select [Column1, Column2...N] from [테이블 명1..N] [조건1]..[조건N];

(1) 전체의 레코드를 출력하는 쿼리문
(2) 레코드를 선택하여 출력하는 쿼리문
(3) 레코드의 순서를 변경하여 출력하는 쿼리문 (Column의 입력순서에 따라 출력된다.)

```
mysql> select * from test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 123 | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)

mysql> select name, age from test;
+--------+-----+
| name   | age |
+--------+-----+
| minki  | 123 |
| hyewon | 23  |
+--------+-----+
2 rows in set (0.00 sec)

mysql> select no, phone, age, name from test;
+----+---------------+-----+--------+
| no | phone         | age | name   |
+----+---------------+-----+--------+
|  1 | NULL          | 123 | minki  |
|  2 | 010-9999-9999 | 23  | hyewon |
+----+---------------+-----+--------+
2 rows in set (0.00 sec)
```

 # ORDER BY [Column 명] [DESC(내림) | ASC(오름)]
(Column을 많이 선택할 수 있다.)

```
mysql> select * from test order by age desc;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| hyewon | 23  | 010-9999-9999 |  2 |
| minki  | 123 | NULL          |  1 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)

mysql> select * from test order by age asc;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 123 | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)
```

3) 레코드 삭제 - DELETE / TRUNCATE

mysql> delete from [테이블 명] where [조건];

 # 레코드 한줄 삭제 실습

```
mysql> select * from test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 25  | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
| minki2 | 55  | 000000        |  3 |
+--------+-----+---------------+----+
3 rows in set (0.00 sec)

mysql> delete from test where name='minki2';
Query OK, 1 row affected (0.07 sec)

mysql> select * from test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 25  | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)
```

 # 레코드 전체삭제 실습

```
mysql> select * from test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 123 | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)

mysql> delete from test;
Query OK, 2 rows affected (0.08 sec)

mysql> select * from test;
Empty set (0.00 sec)
```

[심화] - 테이블 생성

 # 테이블 생성시 조건을 삽입하여 테이블을 복사한다.
 # Select 다음 원하는 Column 을 선택하면 그 조건에 맞는 테이블만 생성된다.

```
mysql> show tables;
+----------------+
| Tables_in_test |
+----------------+
| test           |
+----------------+
1 row in set (0.20 sec)

mysql> create table test_test as select * from test;
Query OK, 2 rows affected (1.72 sec)
Records: 2  Duplicates: 0  Warnings: 0

mysql> show tables;
+----------------+
| Tables_in_test |
+----------------+
| test           |
| test_test      |
+----------------+
2 rows in set (0.00 sec)

mysql> select * from test_test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 123 | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)
```

mysql> truncate [테이블 명]

 # truncate 명령을 사용해 레코드를 전부 지우는 실습

 # delete from [테이블 명] 보다 시스템 부하가 적다.

```
mysql> select * from test_test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 123 | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)

mysql> truncate test_test;
Query OK, 0 rows affected (0.51 sec)

mysql> select * from test_test;
Empty set (0.00 sec)
```

4) 레코드 변경 - UPDATE

mysql> update [테이블 명] set [Column]=데이터 where [조건]

 # 레코드 데이터 변경 실습

```
mysql> select * from test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 25  | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)

mysql> update test set name='minkiiiii' where no=1;
Query OK, 1 row affected (1.06 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from test;
+-----------+-----+---------------+----+
| name      | age | phone         | no |
+-----------+-----+---------------+----+
| minkiiiii | 25  | NULL          |  1 |
| hyewon    | 23  | 010-9999-9999 |  2 |
+-----------+-----+---------------+----+
2 rows in set (0.00 sec)
```

2개 이상 Column 값 변경 실습

```
mysql> select * from test;
+-----------+-----+---------------+----+
| name      | age | phone         | no |
+-----------+-----+---------------+----+
| minkiiiii | 25  | NULL          |  1 |
| hyewon    | 23  | 010-9999-9999 |  2 |
+-----------+-----+---------------+----+
2 rows in set (0.00 sec)

mysql> update test set age='123', name='minki' where no=1;
Query OK, 1 row affected (0.06 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from test;
+--------+-----+---------------+----+
| name   | age | phone         | no |
+--------+-----+---------------+----+
| minki  | 123 | NULL          |  1 |
| hyewon | 23  | 010-9999-9999 |  2 |
+--------+-----+---------------+----+
2 rows in set (0.00 sec)
```