

T아카데미 온라인 강의

데이터베이스 프로그래밍 기초

03. SQL(DML) 1



CONTENTS

1

SQL의 이해와 종류

2

CRUD 이해와 실습

학습목표

1. SQL의 개념과 종류에 대해 이해할 수 있습니다.
2. CRUD 개념을 이해할 수 있습니다.
3. 실습을 통해 CRUD에 대한 SQL문 사용 방법을 습득할 수 있습니다.

1. SQL의 이해와 종류



1. SQL의 이해와 종류

● SQL의 이해

SQL (Structured Query Language)

- 데이터베이스에 있는 필요한 정보를 사용할 수 있도록 도와주는 언어
- 사용방법이나 문법이 다른 언어(Java, C, C#, VB 등) 보다 단순함
- 하나를 배워두면 모든 DBMS에서 사용가능함
- 인터프리터
- 대소문자 구별하지 **않음**(데이터 내용은 구별함)

1. SQL의 이해와 종류

● SQL의 이해



- 사용자나 프로그램이 필요한 데이터를 가져오기 위해 SQL을 작성한다.
- DBMS가 DB에서 사용자가 작성한 SQL을 통해서 데이터를 가져온다.
- 사용자는 SQL을 통해서 가져온 데이터를 볼 수 있다.

1. SQL의 이해와 종류

● SQL의 종류

DML (Data Manipulation Language)

- 테이블의 데이터를 조작하는 기능
- 테이블의 레코드를 CRUD(Create, Retrieve, Update, Delete)

SQL문	내 용
INSERT	데이터베이스 객체에 데이터를 입력한다.
DELETE	데이터베이스 객체에 데이터를 삭제한다.
UPDATE	데이터베이스 객체 안의 데이터를 수정한다.

1. SQL의 이해와 종류

● SQL의 종류

DDL (Data Definition Language)

- DB, 테이블의 스키마를 정의, 수정하는 기능
- 테이블 생성, 컬럼 추가, 타입변경, 각종 제약조건 지정, 수정 등

SQL문	내 용
CREATE	데이터베이스 객체를 생성한다.
DROP	데이터베이스 객체를 삭제한다.
ALTER	기존에 존재하는 데이터베이스 객체를 다시 정의한다.

1. SQL의 이해와 종류

● SQL의 종류

DCL (Data Control Language)

- DB나 테이블의 접근권한이나 CRUD 권한을 정의하는 기능
- 특정 사용자에게 테이블의 조회권한 허가/금지 등

SQL문	내 용
GRANT	데이터베이스 객체에 권한을 부여한다.
REVOKE	이미 부여된 데이터베이스 객체 권한을 취소한다.

2. CRUD 이해와 실습



2. CRUD 이해와 실습

● CRUD (Create, Retrieve, Updata, Delete)

- CRUD는 기본적인 데이터 처리 기능인 Create(생성), Read(읽기), Update(갱신), Delete(삭제)를 묶어서 일컫는 말이다. 사용자 인터페이스가 갖추어야 할 기능(정보의 참조/검색/갱신)을 가리키는 용어로서도 사용된다.

2. CRUD 이해와 실습

● CRUD (Create, Retrieve, Updata, Delete)

▶ CRUD 대신 사용되는 유사 용어

- ABCD

- add(추가), browse(보기), change(변경), delete(삭제)

- ACID

- add(추가), change(변경), inquire(질의), delete(삭제)

- BREAD

- browse(보기), read(읽기), edit(편집), add(추가), delete(삭제)

- VADE(R)

- view(참조), add(추가), delete(삭제), edit(편집), 트랜잭션 처리에서는 restore(복원) 추가

2. CRUD 이해와 실습

● CRUD (Create, Retrieve, Updata, Delete)

▶ CRUD 각 문자와 대응되는 표준 SQL문

이름	조작	SQL
Create	생성	INSERT
Read (또는 Retrieve)	읽기 (또는 인출)	SELECT
Update	갱신	UPDATE
Delete (또는 Destroy)	삭제 (또는 파괴)	DELETE

[출처] 위키백과(<https://ko.wikipedia.org/wiki/CRUD>)

▶ Create ; 데이터베이스 객체 생성

- INSERT INTO
- 새로운 레코드를 추가

2. CRUD 이해와 실습

● CRUD (Create, Retrieve, Updata, Delete)

Update ; 데이터베이스 객체 안의 데이터 수정

- UPDATE
- 특정 조건의 레코드(들)의 컬럼 값을 수정

Delete ; 데이터베이스 객체의 데이터 삭제

- DELETE
- 특정 조건의 레코드(들)을 삭제

Retrieve(Search) ; 데이터베이스 객체 안의 데이터 검색

- SELECT
- 조건을 만족하는 레코드(들)를 찾아 특정 컬럼 값(모두 표시하려면 *)을 표시

2. CRUD 이해와 실습

MySQL 이용한 CRUD 실습

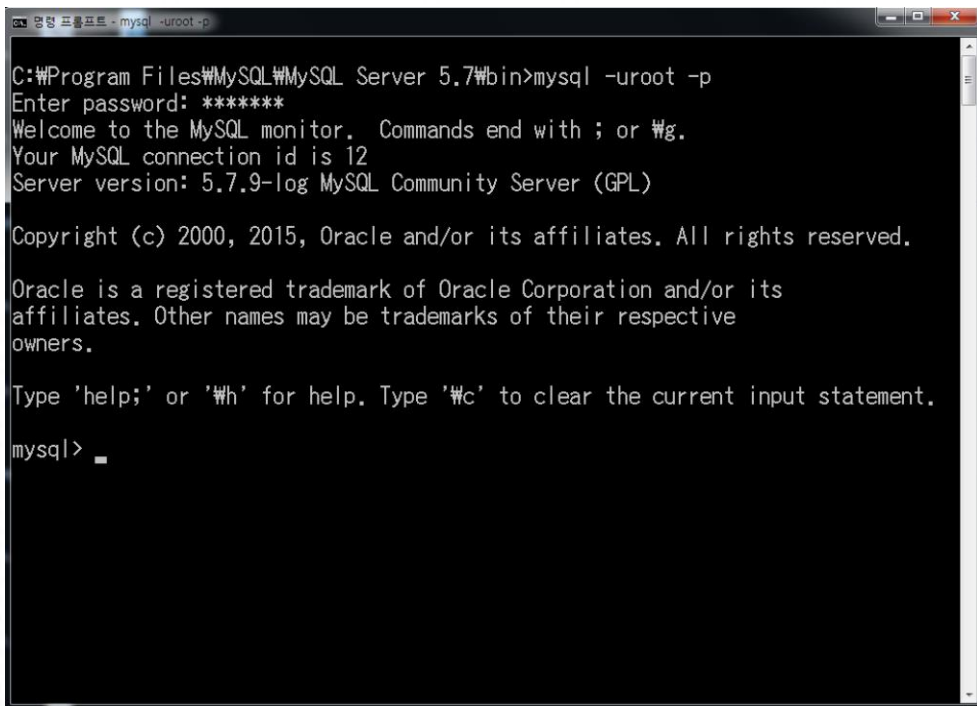
MySQL 실습 구성

- MySQL 접속
`mysql -uroot -p [비밀번호]`
 - `use world;`
 - `show tables ;`
 - `desc city;`
-

2. CRUD 이해와 실습

MySQL 이용한 CRUD 실습

MySQL 실행화면



```
C:\Program Files\MySQL\MySQL Server 5.7\bin>mysql -uroot -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 5.7.9-log MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

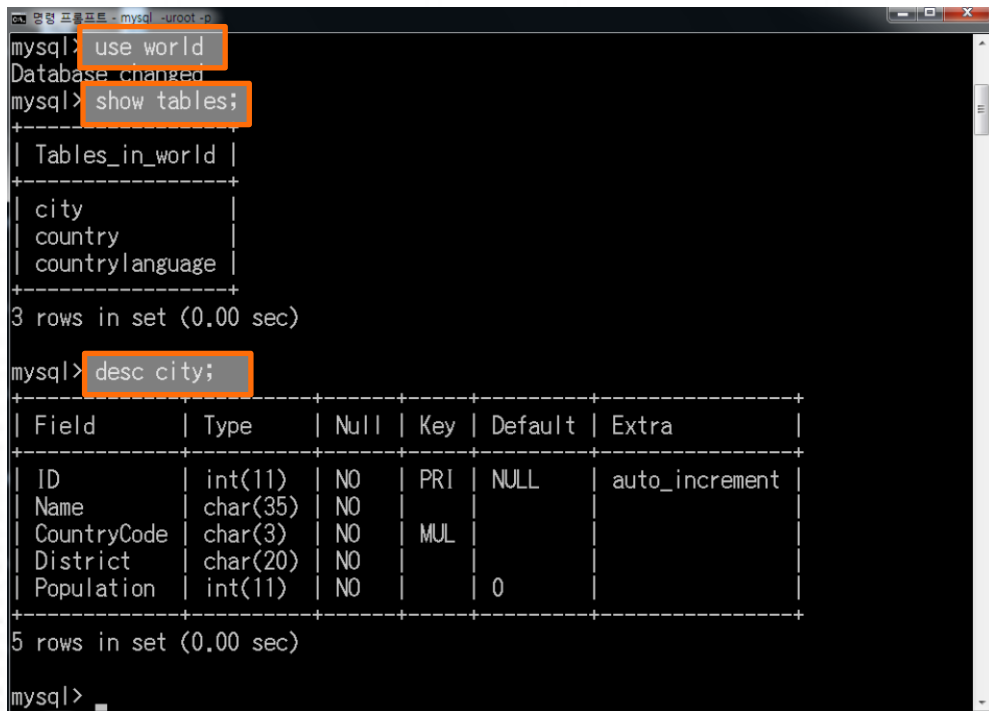
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> _
```


2. CRUD 이해와 실습

MySQL 이용한 CRUD 실습

MySQL 실행화면



```
mysql> use world
Database changed
mysql> show tables;
+-----+
| Tables_in_world |
+-----+
| city              |
| country           |
| countrylanguage   |
+-----+
3 rows in set (0.00 sec)

mysql> desc city;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| ID     | int(11) | NO | PRI | NULL | auto_increment |
| Name   | char(35) | NO |   |   |   |
| CountryCode | char(3) | NO | MUL |   |   |
| District | char(20) | NO |   |   |   |
| Population | int(11) | NO |   | 0 |   |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

mysql>
```

2. CRUD 이해와 실습

● SELECT 명령문

▪ SELECT 컬럼명 FROM 테이블명 WHERE 조건절 ;

▶ 예제1

```
desc city;
```

▶ 예제2

```
select * from city;  
select * from city where CountryCode='KOR';  
select * from city where Population > 5000000;
```

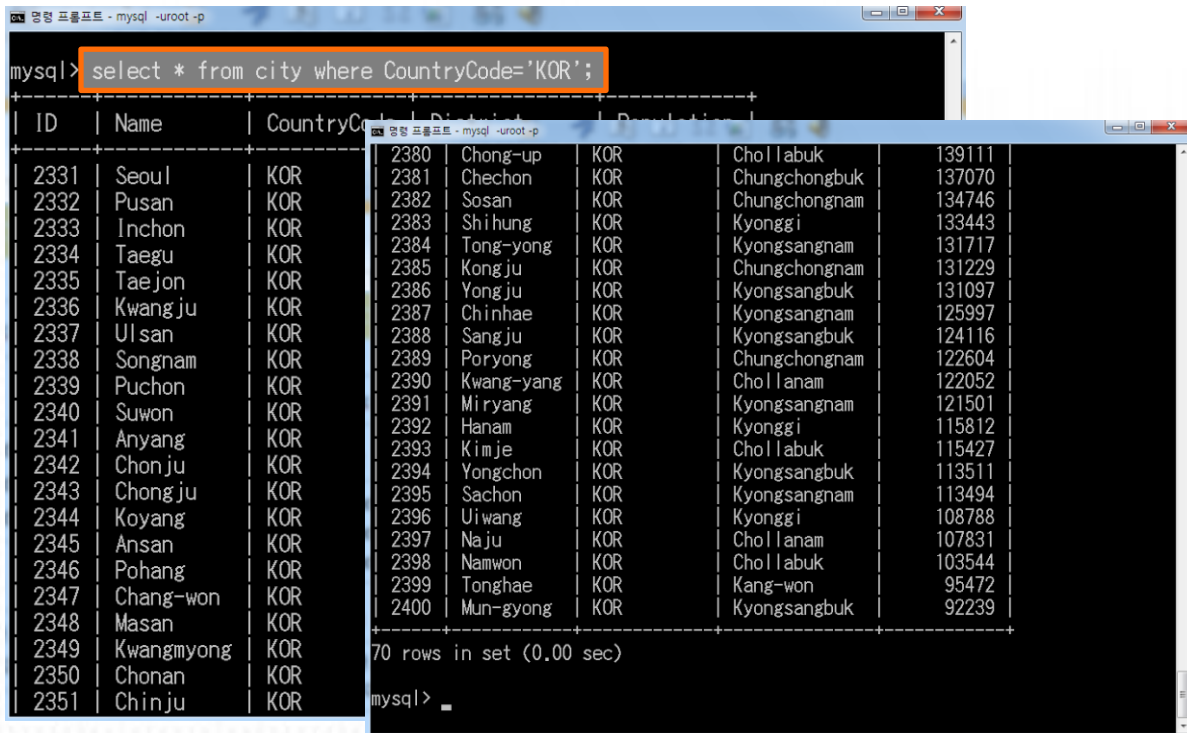
▶ 예제3

국가 코드가 'CHN'으로 되어 있는 도시[들]의 '이름'
인구가 1,000만 이상인 도시[들]의 '이름 '

2. CRUD 이해와 실습

SELECT 명령문

SELECT 명령문 결과 화면



```
mysql> select * from city where CountryCode='KOR';
```

ID	Name	CountryCode	District	Population
2331	Seoul	KOR		
2332	Pusan	KOR		
2333	Inchon	KOR		
2334	Taegu	KOR		
2335	Taejon	KOR		
2336	Kwangju	KOR		
2337	Ulsan	KOR		
2338	Songnam	KOR		
2339	Puchon	KOR		
2340	Suwon	KOR		
2341	Anyang	KOR		
2342	Chonju	KOR		
2343	Chongju	KOR		
2344	Koyang	KOR		
2345	Ansan	KOR		
2346	Pohang	KOR		
2347	Changwon	KOR		
2348	Masan	KOR		
2349	Kwangmyong	KOR		
2350	Chonan	KOR		
2351	Chinju	KOR		
2380	Chong-up	KOR	Chollabuk	139111
2381	Chechon	KOR	Chungchongbuk	137070
2382	Sosan	KOR	Chungchongnam	134746
2383	Shihung	KOR	Kyonggi	133443
2384	Tong-yong	KOR	Kyongsangnam	131717
2385	Kongju	KOR	Chungchongnam	131229
2386	Yongju	KOR	Kyongsangbuk	131097
2387	Chinhae	KOR	Kyongsangnam	125997
2388	Sangju	KOR	Kyongsangbuk	124116
2389	Poryong	KOR	Chungchongnam	122604
2390	Kwang-yang	KOR	Chollanam	122052
2391	Miryang	KOR	Kyongsangnam	121501
2392	Hanam	KOR	Kyonggi	115812
2393	Kimje	KOR	Chollabuk	115427
2394	Yongchon	KOR	Kyongsangbuk	113511
2395	Sachon	KOR	Kyongsangnam	113494
2396	Uiwang	KOR	Kyonggi	108788
2397	Naju	KOR	Chollanam	107831
2398	Namwon	KOR	Chollabuk	103544
2399	Tonghae	KOR	Kangwon	95472
2400	Mun-gyong	KOR	Kyongsangbuk	92239

70 rows in set (0.00 sec)

```
mysql>
```

2. CRUD 이해와 실습

SELECT 명령문

SELECT 명령문 결과 화면



```
mysql> select * from city where Population > 5000000;
```

ID	Name	Country	Population
206	S?o Paulo	B	5064000
207	Rio de Janeiro	B	9981619
456	London	C	8591309
608	Cairo	E	9269265
939	Jakarta	I	5063499
1024	Mumbai (Bombay)	I	6464693
1025	Delhi	I	6320174
1380	Teheran	I	8787958
1532	Tokyo	J	8389200
2298	Kinshasa	COD	8008278
2331	Seoul	KOR	
2515	Ciudad de M??xico	MEX	
2822	Karachi	PAK	
2823	Lahore	PAK	
2890	Lima	PER	
3320	Bangkok	THA	
3357	Istanbul	TUR	
3580	Moscow	RUS	
3793	New York	USA	

24 rows in set (0.00 sec)

```
mysql>
```

2. CRUD 이해와 실습

● INSERT INTO 명령문

▪ INSERT INTO 테이블명(컬럼명) VALUES (값);

▶ 예제

```
insert into city (ID, Name, CountryCode, District, Population) values  
(10000, 'Sample', 'SPL', 'Test', 1000000);  
insert into city values (20000, 'SampleTest', 'SPL', 'Test', 2000000);
```

▶ 결과 확인

```
select * from city where ID=10000;  
select * from city where ID=20000;
```

2. CRUD 이해와 실습

● INSERT INTO 명령문

▶ INSERT INTO 명령문 결과 화면

```
mysql> insert into city (ID, Name, CountryCode, District, Population) values (10000, 'SampleCity', 'KOR', 'Seoul', 1000000);
Query OK, 1 row affected (0.05 sec)
```

```
mysql> select * from city where id = 10000;
```

ID	Name	CountryCode	District	Population
10000	SampleCity	KOR	Seoul	1000000

```
1 row in set (0.00 sec)
```

```
mysql> insert into city values (20000, 'SampleCity2', 'KOR', 'Seoul', 2000000);
Query OK, 1 row affected (0.05 sec)
```

```
mysql> select * from city where id = 20000;
```

ID	Name	CountryCode	District	Population
20000	SampleCity2	KOR	Seoul	2000000

```
1 row in set (0.00 sec)
```

```
mysql>
```

2. CRUD 이해와 실습

● UPDATE 명령문

- UPDATE 테이블명 SET 컬럼명=값, ... WHERE 조건절;

▶ 예제

```
update city set name='SampleRevised' where id = 10000;
```

▶ 결과 확인

```
select * from city where ID=10000;
```

2. CRUD 이해와 실습

● UPDATE 명령문

▶ UPDATE 명령문 결과 화면

```
mysql> update city set name='SampleCityRevised' where id = 10000;  
Query OK, 1 row affected (0.01 sec)  
Rows matched: 1 Changed: 1 Warnings: 0  
  
mysql> update city set Population = 3000000 where id = 20000;  
Query OK, 1 row affected (0.00 sec)  
Rows matched: 1 Changed: 1 Warnings: 0  
  
mysql> select * from city where id = 10000;  
+-----+-----+-----+-----+-----+  
| ID   | Name           | CountryCode | District | Population |  
+-----+-----+-----+-----+-----+  
| 10000 | SampleCityRevised | KOR         | Seoul    | 1000000    |  
+-----+-----+-----+-----+-----+  
1 row in set (0.00 sec)  
  
mysql> select * from city where id = 20000;  
+-----+-----+-----+-----+-----+  
| ID   | Name           | CountryCode | District | Population |  
+-----+-----+-----+-----+-----+  
| 20000 | SampleCity2    | KOR         | Seoul    | 3000000    |  
+-----+-----+-----+-----+-----+  
1 row in set (0.00 sec)  
  
mysql>
```


2. CRUD 이해와 실습

• DELETE 명령문

▪ DELETE FROM 테이블명 WHERE 조건절;

▶ 예제

```
delete from city where ID = 10000;  
delete from city where ID = 20000;
```

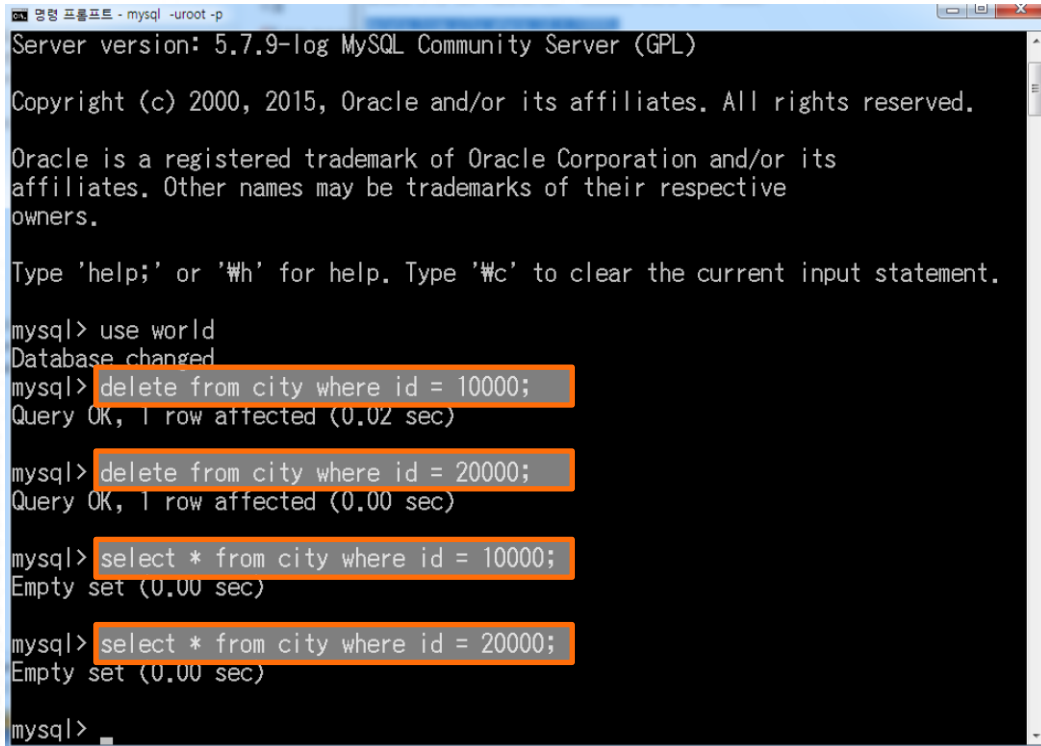
▶ 결과 확인

```
select * from city where ID=10000;  
select * from city where ID=20000;
```

2. CRUD 이해와 실습

● DELETE 명령문

▶ DELETE 명령문 결과 화면



```
mysql -uroot -p
Server version: 5.7.9-log MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or 'h' for help. Type 'q' to quit.

mysql> use world
Database changed
mysql> delete from city where id = 10000;
Query OK, 1 row affected (0.02 sec)

mysql> delete from city where id = 20000;
Query OK, 1 row affected (0.00 sec)

mysql> select * from city where id = 10000;
Empty set (0.00 sec)

mysql> select * from city where id = 20000;
Empty set (0.00 sec)

mysql>
```

학습정리



- 지금까지 'SQL(DML) 1' 에 대해 살펴보았습니다.

DML (Data Manipulation Language)

SQL 종류의 하나로, 데이터의 삽입(INSERT), 삭제(DELETE), 갱신(UPDATE) 등을 처리함

CRUD (Create, Retrieve, Update, Delete)

기본적인 데이터 처리 기능인 Create(생성), Read(읽기), Update(갱신), Delete(삭제)를 묶어서 일컫는 말

- SELECT 컬럼명 FROM 테이블명 WHERE 조건절;
- INSERT INTO 테이블명(컬럼명) values (값);
- UPDATE 테이블명 set 컬럼명=값, ... where 조건절;
- DELETE from 테이블명 where 조건절;