

# <Term Project : Database>



제출일	2022.11.22	전공	소프트웨어학과
과목	데이터시스템프로그래밍	학번	2020039096
담당교수	노서영 교수님	이름	백인혁

## 1. putty로 centos7에 접속하여 작업한다

baekinhyeok@mysql:~

```
login as: baekinhyeok
baekinhyeok@192.168.89.6's password:
Last failed login: Tue Nov 15 21:57:40 KST 2022 from 192.168.89.1 on ssh:notty
There was 1 failed login attempt since the last successful login.
Last login: Tue Nov 15 20:41:33 2022 from 192.168.89.1
```

## 2. root 계정으로 mysql에 접근한다

```
[baekinhyeok@mysql ~]$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.7.39 MySQL Community Server (GPL)

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

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.
```

## 3. ApartmentComplex라는 이름을 가진 Database가 있다면 삭제한다.

```
mysql> DROP DATABASE IF EXISTS ApartmentComplex;
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

## 4. ApartmentComplex라는 이름을 가진 로컬 사용자가 있다면 삭제한다.

```
mysql> DROP USER IF EXISTS ApartmentComplex@localhost;
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

## 5. ApartmentComplex라는 이름을 가진 로컬 사용자를 생성한다. 비밀번호는 ApartmentComplex이다.

```
mysql> create user ApartmentComplex@localhost identified WITH mysql_native_password by 'ApartmentComplex';
Query OK, 0 rows affected (0.00 sec)
```

## 6. ApartmentComplex Database를 생성한다.

```
mysql> create database ApartmentComplex;
Query OK, 1 row affected (0.00 sec)
```

## 7. 로컬 사용자에게 ApartmentComplex Database에 대한 모든 권한을 부여한다.

```
mysql> grant all privileges on ApartmentComplex.* to ApartmentComplex@localhost with grant option;
Query OK, 0 rows affected (0.00 sec)
```

## 8. ApartmentComplex를 사용한다

```
mysql> use ApartmentComplex
Database changed
```

## 9. 동명의 테이블이 있을 것을 방지하기 위해 사용할 이름을 가진 테이블들을 삭제한다.

```
mysql> DROP TABLE IF EXISTS Manager;
Query OK, 0 rows affected, 1 warning (0.00 sec)
mysql> DROP TABLE IF EXISTS Office;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> DROP TABLE IF EXISTS Apartment;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> DROP TABLE IF EXISTS Facility;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> DROP TABLE IF EXISTS President;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> DROP TABLE IF EXISTS Work;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> DROP TABLE IF EXISTS Mangae;
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> DROP TABLE IF EXISTS Represent;
Query OK, 0 rows affected, 1 warning (0.00 sec)
```

10. 데이터베이스에서 사용되는 기본 요소들에 대한 테이블들부터 생성한다.

#### 10-1. 관리자(Manager) 테이블 생성

```
mysql> CREATE TABLE Manager(  
-> mid INTEGER PRIMARY KEY,  
-> mname VARCHAR(30),  
-> mage INTEGER,  
-> mtel VARCHAR(20)  
-> );  
Query OK, 0 rows affected (0.00 sec)
```

#### 10-2. 사무실(Office) 테이블 생성

```
mysql> CREATE TABLE Office(  
-> ocode INTEGER PRIMARY KEY,  
-> oname VARCHAR(20),  
-> mcount INTEGER  
-> );  
Query OK, 0 rows affected (0.01 sec)
```

#### 10-3. 아파트(Apartment) 테이블 생성

```
mysql> CREATE TABLE Apartment(  
-> acode INTEGER PRIMARY KEY,  
-> aname VARCHAR(20),  
-> afloor INTEGER,  
-> aarea INTEGER,  
-> housecount INTEGER  
-> );  
Query OK, 0 rows affected (0.01 sec)
```

#### 10-4. 시설(Facility) 테이블 생성

```
mysql> CREATE TABLE Facility(  
-> fcode INTEGER PRIMARY KEY,  
-> fname VARCHAR(20),  
-> ffloor INTEGER,  
-> farea INTEGER,  
-> buildingtype VARCHAR(10)  
-> );  
Query OK, 0 rows affected (0.01 sec)
```

#### 10-5. 대표자(President) 테이블 생성

```
mysql> CREATE TABLE President(  
-> pname VARCHAR(20) PRIMARY KEY,  
-> page INTEGER,  
-> pjob VARCHAR(30)  
-> );  
Query OK, 0 rows affected (0.00 sec)
```

11. 테이블 간 관계를 정의하는 테이블들을 생성한다.

#### 11-1. 관리자와 아파트를 연결하는 ManageApartment

```
mysql> CREATE TABLE ManageApartment(  
-> mid Integer PRIMARY KEY,  
-> acode INTEGER  
-> );  
Query OK, 0 rows affected (0.01 sec)
```

#### 11-2. 관리자와 시설을 연결하는 ManageFacility

```
mysql> CREATE TABLE ManageFacility(  
-> mid INTEGER PRIMARY KEY,  
-> fcode INTEGER  
-> );  
Query OK, 0 rows affected (0.01 sec)
```

11-3. 관리자와 사무실을 연결하는 WorkIn

```
mysql> CREATE TABLE WorkIn(  
-> mid INTEGER PRIMARY KEY,  
-> ocode INTEGER  
-> );  
Query OK, 0 rows affected (0.00 sec)
```

11-4. 아파트와 대표자를 연결하는 Represent

```
mysql> CREATE TABLE Represent(  
-> acode INTEGER PRIMARY KEY,  
-> pname VARCHAR(20)  
-> );  
Query OK, 0 rows affected (0.01 sec)
```

12. 변동사항이 없는 아파트에 대한 정보는 미리 입력했다. 아파트는 101동부터 106동까지 6개 동으로 고정되어있다.

```
mysql> SELECT * FROM Apartment;  
+-----+-----+-----+-----+-----+  
| acode | aname  | afloor | aarea | housecount |  
+-----+-----+-----+-----+-----+  
| 1     | 101동  | 35     | 24    | 140        |  
| 2     | 102동  | 40     | 33    | 160        |  
| 3     | 103동  | 30     | 33    | 120        |  
| 4     | 104동  | 40     | 24    | 160        |  
| 5     | 105동  | 35     | 45    | 140        |  
| 6     | 106동  | 20     | 62    | 40         |  
+-----+-----+-----+-----+-----+  
6 rows in set (0.00 sec)
```

13. 아파트 단지의 사무실은 다음과 같이 고정되어있다.

```
mysql> SELECT * FROM Office;  
+-----+-----+-----+  
| ocode | oname  | mcount |  
+-----+-----+-----+  
| 112   | 보 안  | 0      |  
| 113   | 미 화  | 0      |  
| 114   | 전 기  | 0      |  
| 115   | 수 도  | 0      |  
| 116   | 행 정  | 0      |  
+-----+-----+-----+  
5 rows in set (0.00 sec)
```