DB 시스템 및 프로그래밍 과제

20175119 김경태

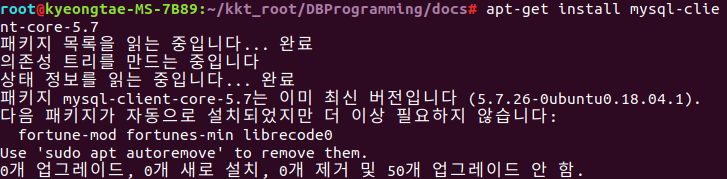
**프로그램 실행 및 개발 환경**

- 우분투 18.04 LTS 64bit.

**1. 사용한 DBMS/대화식 SQL 도구**

- Amazon web service의 RDS에서 MySql 5.7.22 버전을 선택하여 DB 서버를 구축함.

- mysql client는 우분투 18.04 LTS에서 mysql-client-core-5.7 을 설치하여 CLI에서 대화식 Sql을 사용함.





**2. 언어**

- 주언어 : java

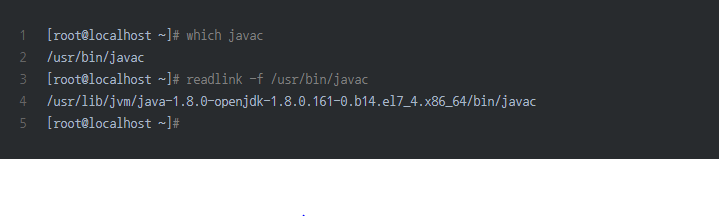
- DB API : JDBC

우분투 18.04 자바 개발환경 세팅 후 JDBC까지 세팅 완료.

**자바 개발환경 세팅**

sudo apt-get install default-jdk을 통해 jdk를 설치해 줌.

그리고 아래 사이트를 참조하여 환경변수를 세팅해준다

\* 이때, 내가 jdk폴더의 이름을 내가 설치한 것으로 알맞게 수정해준다.

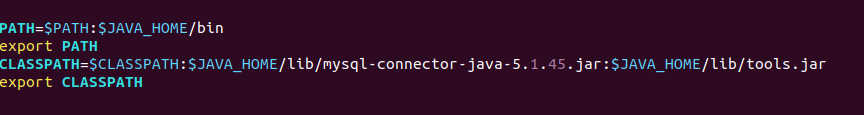
\* 참고한 사이트 : <https://blog.naver.com/PostView.nhn?blogId=opusk&logNo=220985259485>

<https://blog.hanumoka.net/2018/04/30/centOs-20180430-centos-install-jdk/>

**JDBC 드라이버 세팅**

sudo apt-get install libmysql-java 을 통해 jdbc 드라이버를 설치 해준 후 /root/.bashrc 파일에 아래와 같이 환경변수 설정을 추가해 줌.

(이후부터 모두 sudo su로 루트 사용자로 변경하여 작업(자바 프로그램 작성 및 실행 등) 을 진행.)



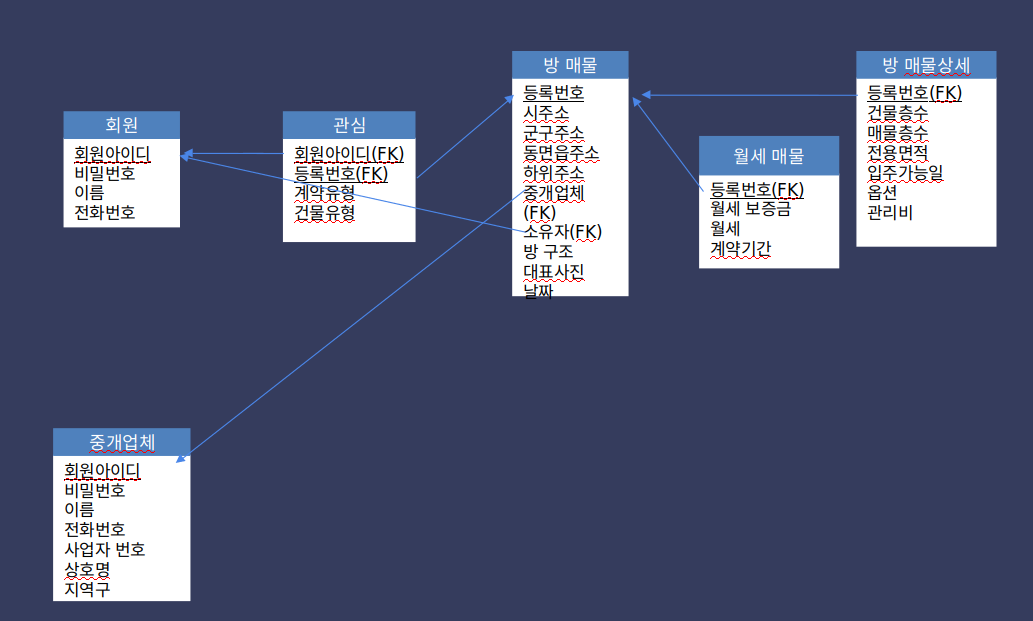
참고한 사이트 : <https://whitememo.tistory.com/201>

**3. 응용분야 및 구현 기능에 대한 설명**

**응용 분야 : 부동산 중개 앱.**

(앱에 로그인하여 매물을 등록하거나 등록된 매물을 볼 수 있다.)

**전체 DB 스키마 중 구현한 기능에 필요한 DB 스키마(서브 스키마).**



**구현 기능**

- 먼저 아래와 같은 과정을 거쳤다고 가정하고 그 이후 방 매물 상세보기 결과를 띄어 주는 것과 그 매물을 관심 매물 리스트에 등록하는 기능을 구현하였다.

**가정.**

1. 부동산 중개 앱에 회원으로 로그인 하여 메뉴에서 방을 클릭 후, 다시 메뉴에서 조회를 클릭한다.
2. 여러 필터 조건을 선택하고 조회 버튼을 클릭한다.
3. 그때 뜨게 되는 방 매물 리스트에서 하나를 골라 상세보기 버튼을 클릭한다.

- 이런 일련의 과정을 거쳤다고 가정했을 때 회원 아이디와 해당 방 매물의 등록번호는 각각 프로그램의 한 변수로 메모리 상에 있기 때문에 main 함수 제일 처음 부분에 String user\_id = “bluesalt752”, String regist\_number = “b34c2ed3-7f7c-11e9-8136-0a9562b69530"를 선언했다.

**구현한 부분.**

이제 이 등록번호와 유저 아이디를 가지고 그 방 매물의 상세보기 버튼을 클릭했을 때 해당 매물에 대한 상세 정보를 동적 select문(3개)을 통해 DB로부터 검색해 띄워준다.

이후 해당 매물을 관심매물 리스트에 등록할 것이냐고 물어보고 yes라고 답하면(숫자 1을 친다.) 이전에 방 매물 상세보기 시 조회한 데이터 중 등록번호(regist\_number)를 가지고 동적 insert문을 통해 해당 매물의 정보(관심매물 등록을 하겠다고 한 현 사용자 id와 그 매물의 등록번호)를 Interest 테이블에 넣어준다.

위의 기능을 구현하기 위해, 이미 방 매물이 등록되어 있고 여러 사용자 및 중개 업체도

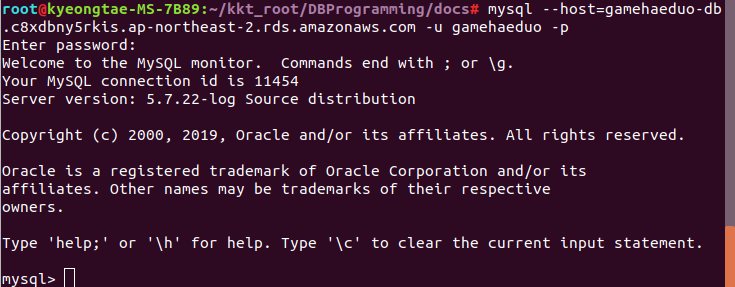
회원가입 한 상황이라고 가정했다. 그리고 방 매물이 한 개 등록된 상태라고 가정하였다.

따라서 이에 따라 미리 테이블들을 채워놓았다.

(Interest 테이블 제외 5개 테이블을 모두 채워 놓음.)

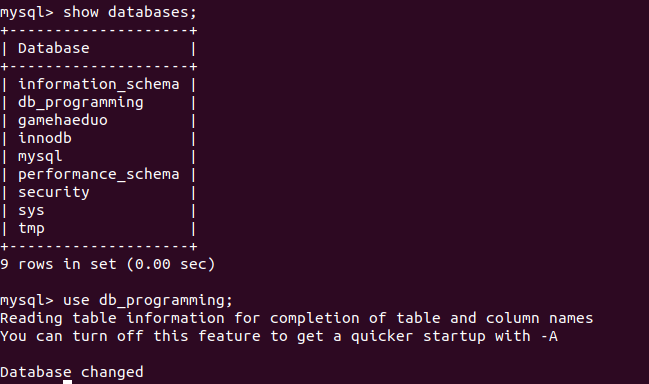
**아래는 이를 수행한 결과 사진 및 설명이다.**

- mysql --host=gamehaeduo-db.c8xdbny5rkis.ap-northeast-2.rds.amazonaws.com -u gamehaeduo –p 명령어를 통해 AWS RDS(mysql server)에 접속.

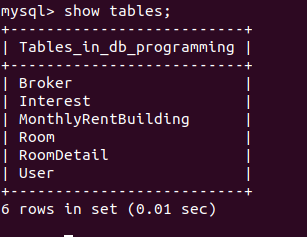


- create database db\_programming; 을 통해 db\_programming 데이터 베이스를 생성하고 그 안에 응용에 필요한 여러 테이블들(6개)을 생성함.

데이터 베이스 목록

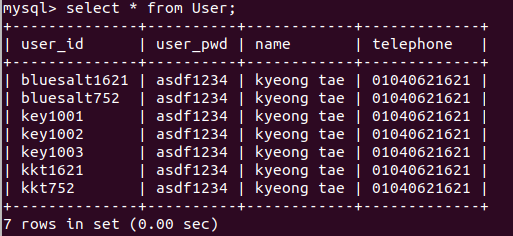


db\_programming DB의 테이블 목록



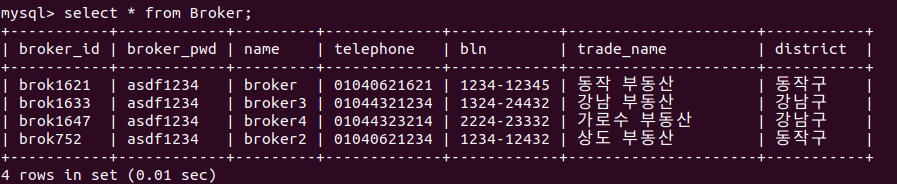
User 테이블(회원 테이블)의 현 상황.

(프로그램 실행 전 일반 회원 회원가입이 되있다고 가정.)



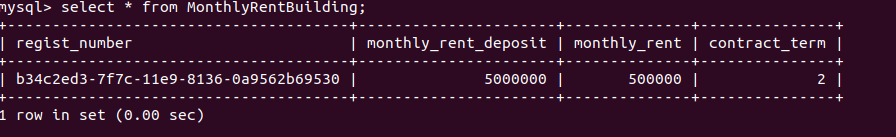
Broker 테이블(중계업체 테이블)의 현 상황.

(프로그램 실행 전 중개업체 회원가입이 되있다고 가정.)



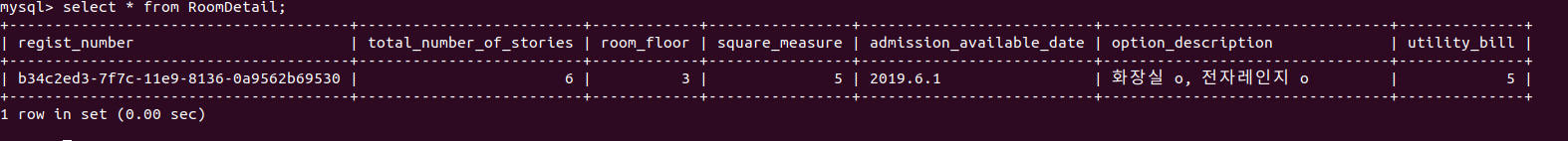
MonthlyRentBuilding(월세 매물 테이블)의 현 상황.

(프로그램 실행 전 방 매물이 등록되어 있다고 가정.)



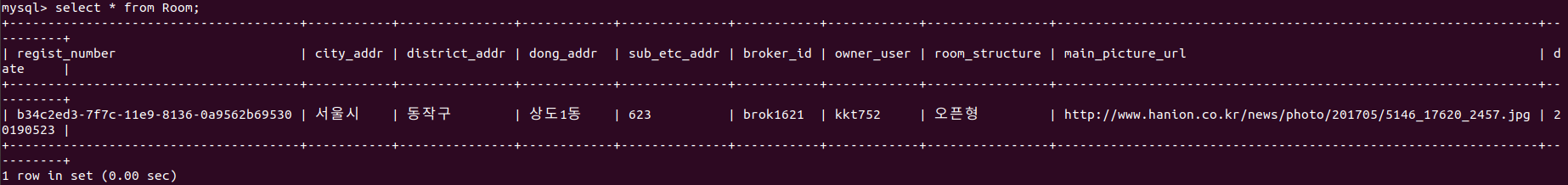
RoomDetail(방매물 상세 테이블)의 현 상황.

(프로그램 실행 전 방 매물이 등록되어 있다고 가정.)



Room(방 매물 테이블)의 현 상황.

(프로그램 실행 전 방 매물이 등록되어 있다고 가정.)



Interest(관심 매물 테이블)의 현 상황.

(프로그램 실행 전 관심매물 테이블이 비어 있는 상황.)



**4. 소스코드(재실행 가능한 포맷으로)**

소스코드를 별도로 첨부파일로 첨부하였으나

github에도 올려놓았습니다.

<https://github.com/cau-school-kkt/DBProgramming>

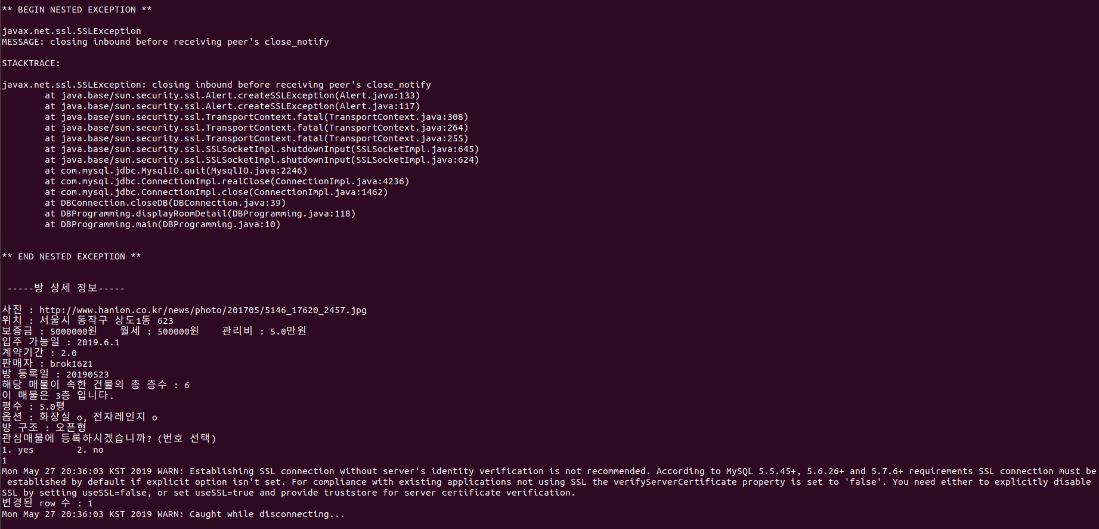
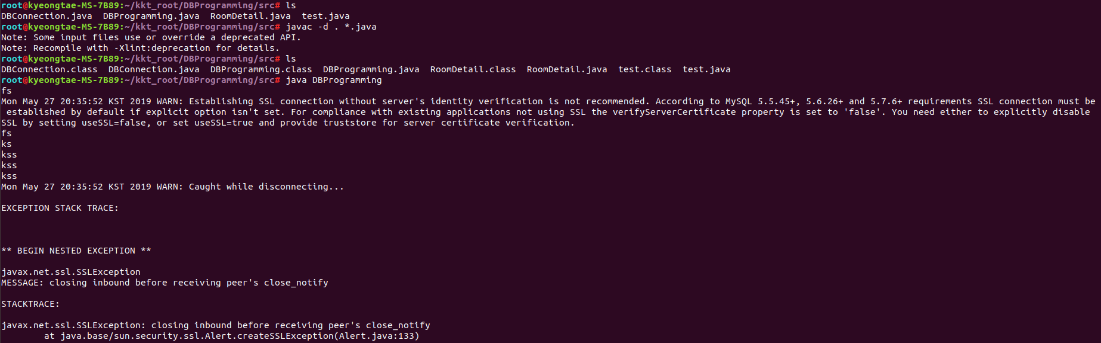
실행 방법

– 우분투 18.04 LTS에서 위에서 적은 과정을 통해 java 세팅 및 JDBC 세팅이 올바르게되어 있다고 했을 때 압축파일(소스코드 및 클래스 파일(binary))의 압축을 푼 후 java DBProgramming 을 입력하여 실행시킨다.

**5. 프로그램 실행 화면 및 대화식 SQL 도구로**

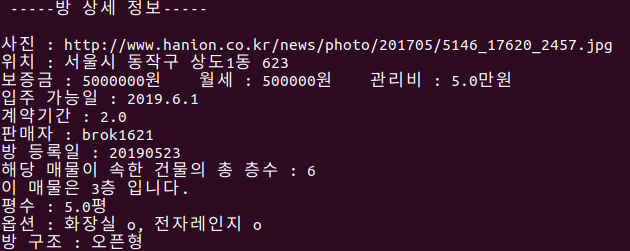
**작동 정확성을 확인한 화면 캡쳐**

프로그램 실행 Process(javac 및 java) 및 결과 화면 전체 캡쳐.



프로그램에서 동적 select문 실행 결과 출력 (Exception 알림 부분을 제외한 캡쳐)

(3개 테이블. Room테이블, MonthlyRentBuilding 테이블, RoomDetail 테이블로부터 해당 매물의 등록번호(메모리 상에 있음)를 이용하여 매물 정보를 가져옴.)



프로그램에서 동적 insert문 실행 결과 출력

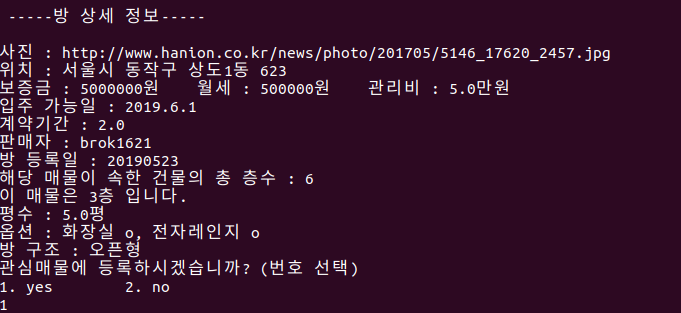
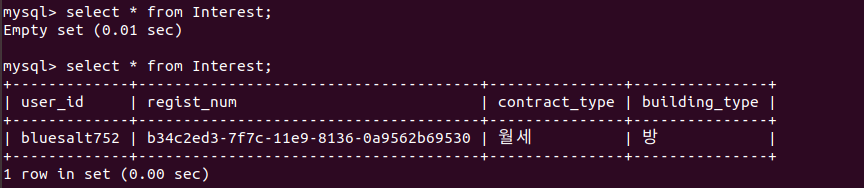


그림 제일 아래에 관심 매물 리스트에 등록 할 것인지 물어본 후 등록 한다고 답하면(숫자 1을 치면) 동적 insert문이 실행되어 아래와 같이 Interest 테이블에 해당 매물에 대한 레코드가 삽입된다.



기존엔 비어 있었지만 프로그램 실행 결과 관심 매물 등록을 하여 레코드가 삽입되었음을 알 수 있다.