

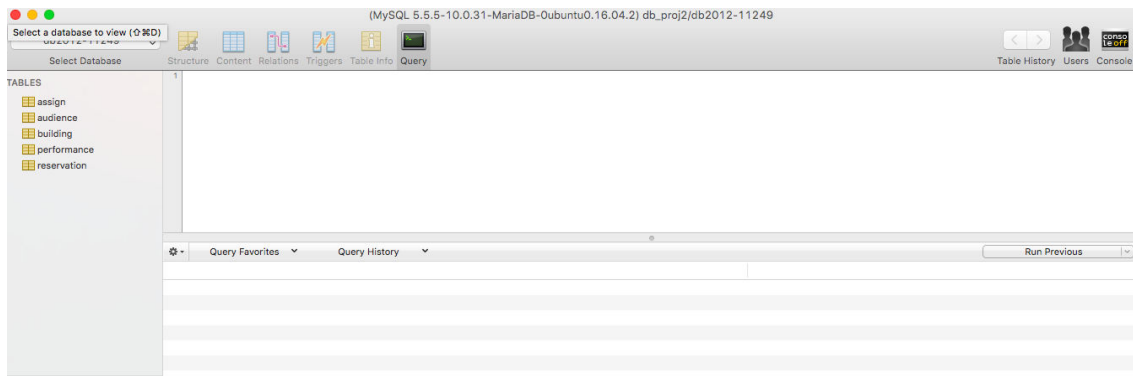
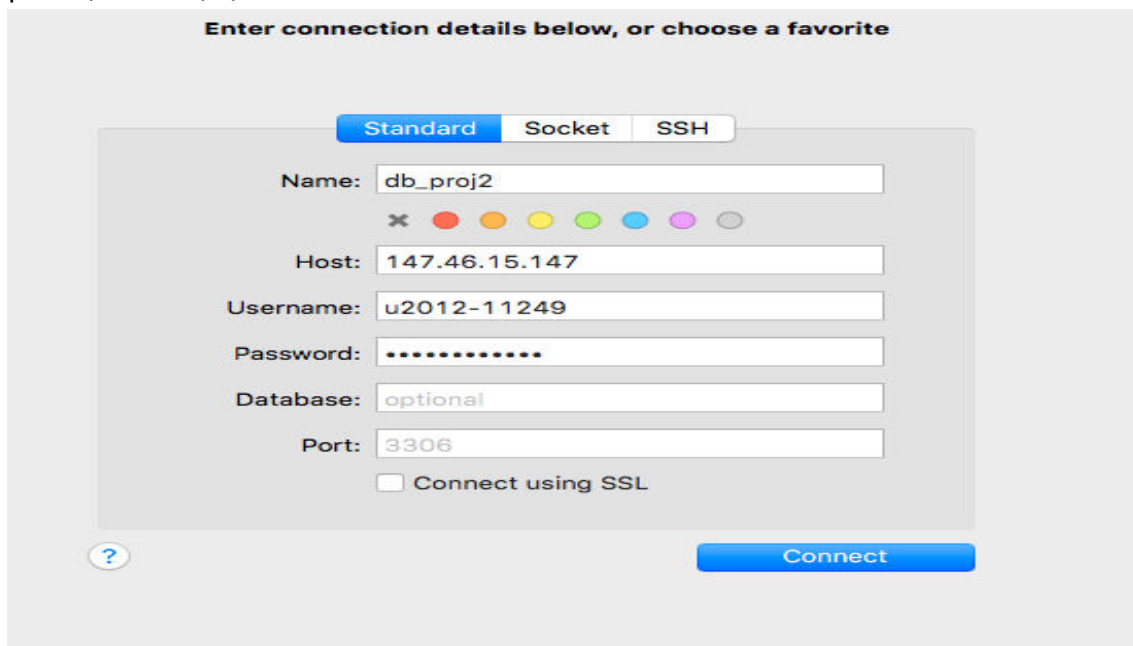
DB2017

Project 2 : Simple Database Application

2012-11249 컴퓨터공학부 설재완

1. 핵심 모듈과 알고리즘에 대한 설명

먼저 remote db 와의 connection 을 열어준다. 그 후 main 에서 command handler()함수를 call 한다. command handler 에는 총 1~14 의 요구사항을 해결해 주는 함수가 정의되어 있다. 각각의 함수에서는 sql 구문을 이용하여 db 와 상호작용을 한다. 특히 'sequel pro'라는 프로그램을 사용하여 실제 table 에 어떤 record 가 들어가 있는지 확인해 볼 수 있었다. 또한 sequel pro 에서 직접 sql 문을 작성하여 결과를 확인할 수 있었다. 아래는 sequel pro 의 모습이다.



2. 구현한 내용에 대한 간략한 설명

각각의 함수에서는 String 으로 sql 을 정의한 후 해당 sql 을 PreparedStatement 로 만든 다음 select 의 경우 executeQuery(), insert, delete 인 경우 executeUpdate()를 call 하게 하였다. 이로써 String 으로부터 시작된 query 가 remote db 에 반영되게 된다. 각각의 함수들은 먼저 필요한 입력을 받고 error check 를 한 후에 error 가 없는 경우에만 query 를 수행하도록 하였다.

3. 가정한 것들

따로 가정한 것은 없다. 다만 스펙에 명시된 것 이외에 예상되는 error 케이스들이 많이 있는데 이것들을 따로 고려하지 않았다. 스펙에 있는 error 케이스는 모두 고려하였으며, 기타사항에 명시된 '여러개의 입력을 받는 경우, 입력을 차례로 받다가 특정 입력에 대해 에러가 있을 시 그 즉시 에러 메시지를 출력하고 해당 명령을 종료함'의 문구를 지키기 위하여 입력을 받는 즉시 error check 를 수행하도록 하였다.

4. 컴파일과 실행방법

이전까지의 프로젝트와 다른점은 .jj 파일이 아닌 .java 파일을 사용했다는 것이다. Eclipse 에서 .java 파일을 컴파일 및 실행하였다. 제출은 Executable jar 파일로, 커맨드라인에서 java -jar PRJ2_2012-11249.jar 명령어로 실행할 수 있다.

5. 프로젝트를 하면서 느낀 점

간단한 에러체크를 하는 경우 create table 에서의 check option 을 사용할 지, 아니면 application level 에서 체크를 해야할지 고민이 생겼다. 특히 일반 함수들 처럼 성공시 0, 실패시 -1 을 return 하는 것이 아니라 ResultSet 이라는 자료구조에 결과를 받아오기 때문에, 실패시에는 SQLException 을 뱉어내기 때문에 java exception handling 에 익숙치 않아 결국 application level 에서 에러를 처리하였다. 아울러 직접 sql 구문을

사용할 기회가 생겨서 sql 에 대해 더욱더 숙달하게 되었다.