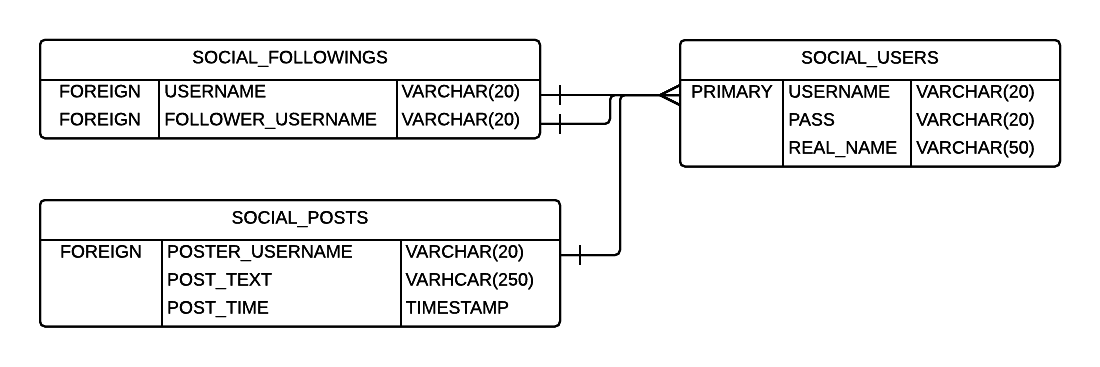
Database Schema:

Social\_Users holds user information. Social\_Followings keeps track of which users a user is following. Social\_Posts holds all posts made by users.

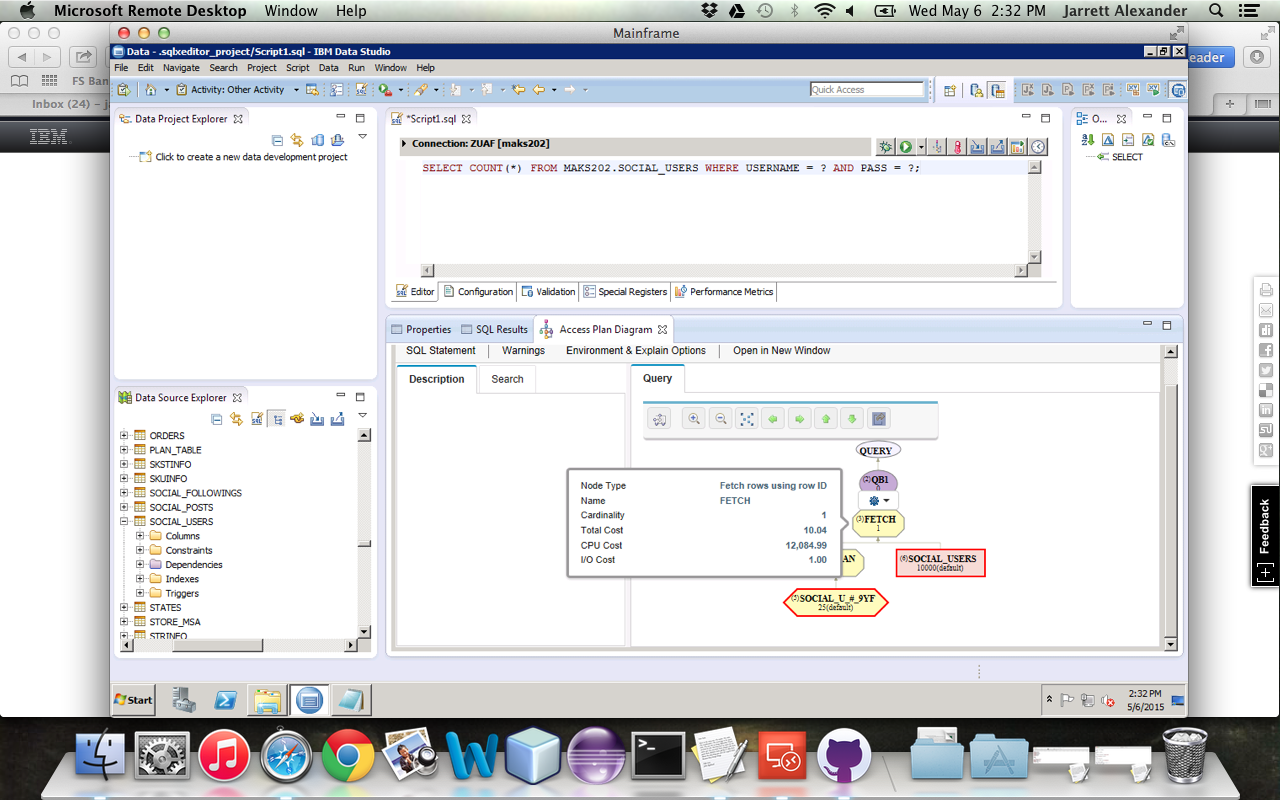


Indexing Information:

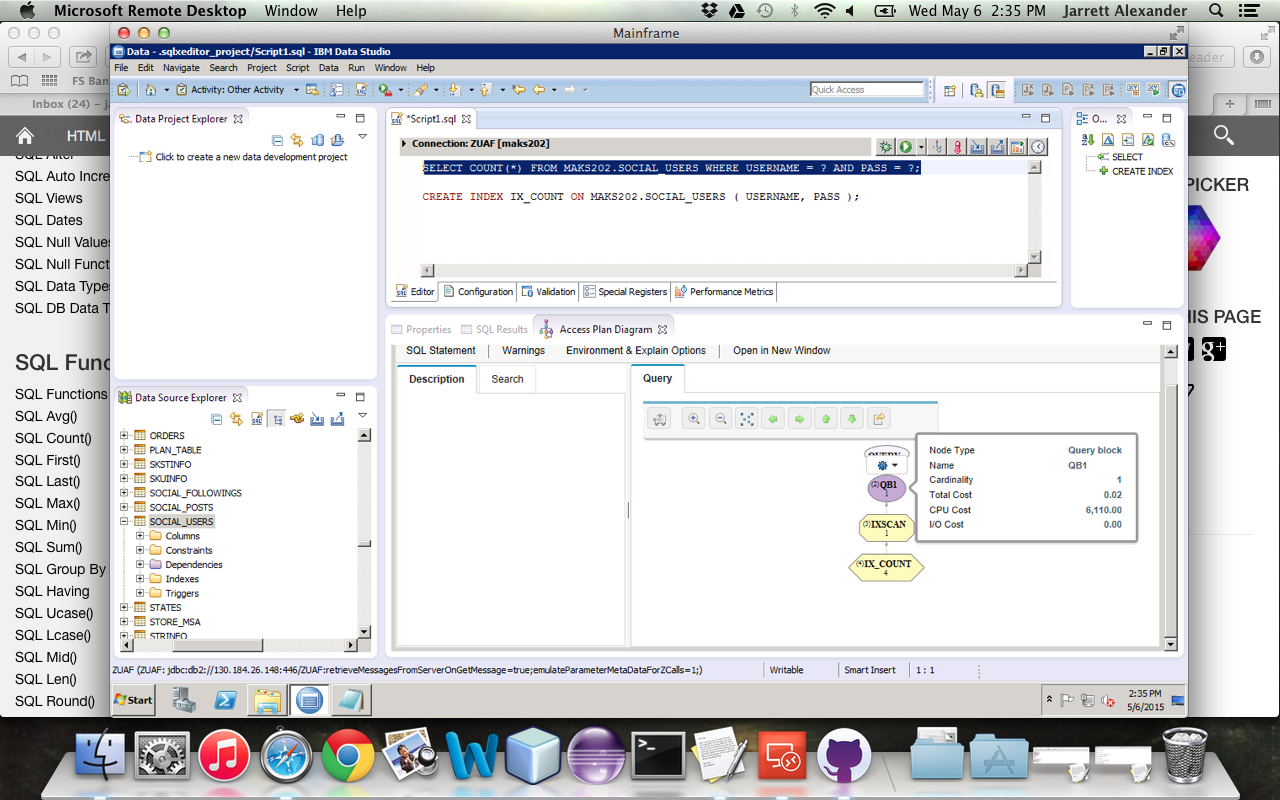
Social\_Users Table:

SELECT COUNT(\*) FROM MAKS202.SOCIAL\_USERS WHERE USERNAME = ? AND PASS = ?;

Authorization for user login. Before index:

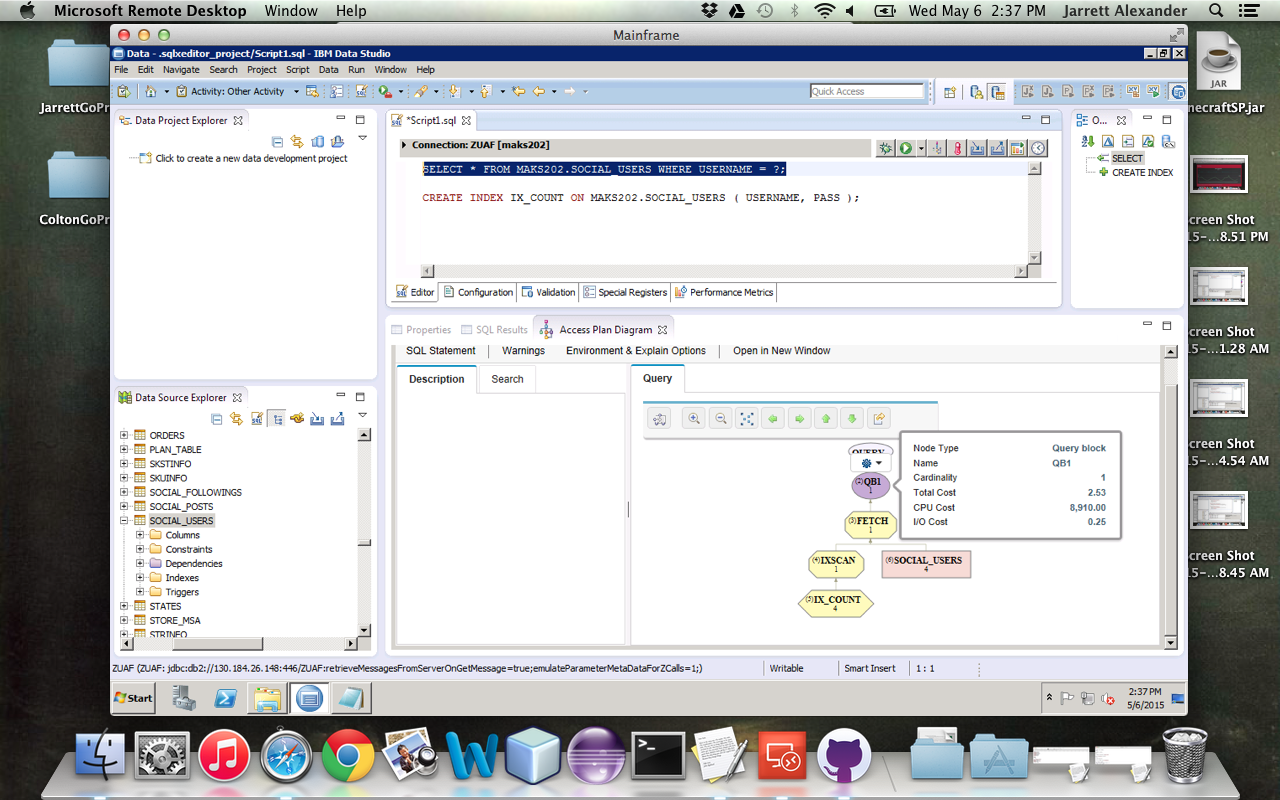


After index:

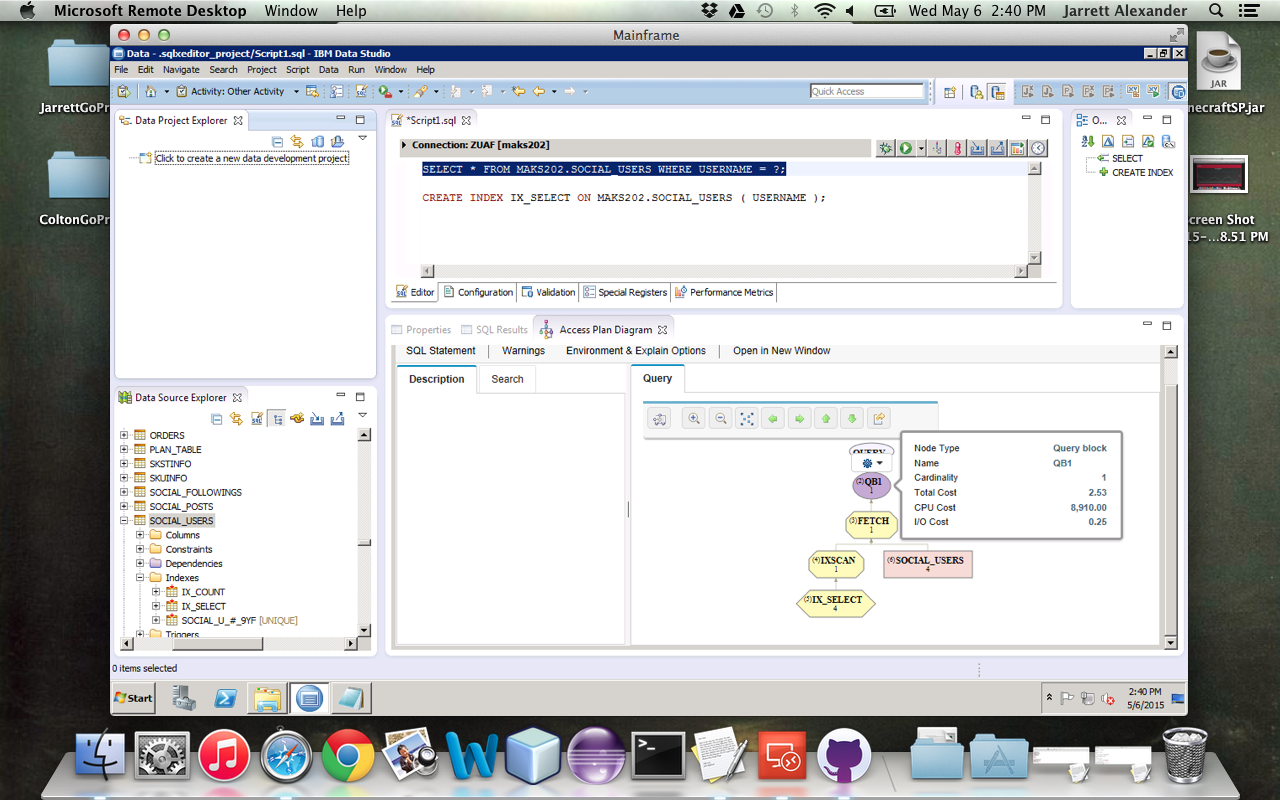


SELECT \* FROM MAKS202.SOCIAL\_USERS WHERE USERNAME = ?;

Retrieving a user by username. Before index:



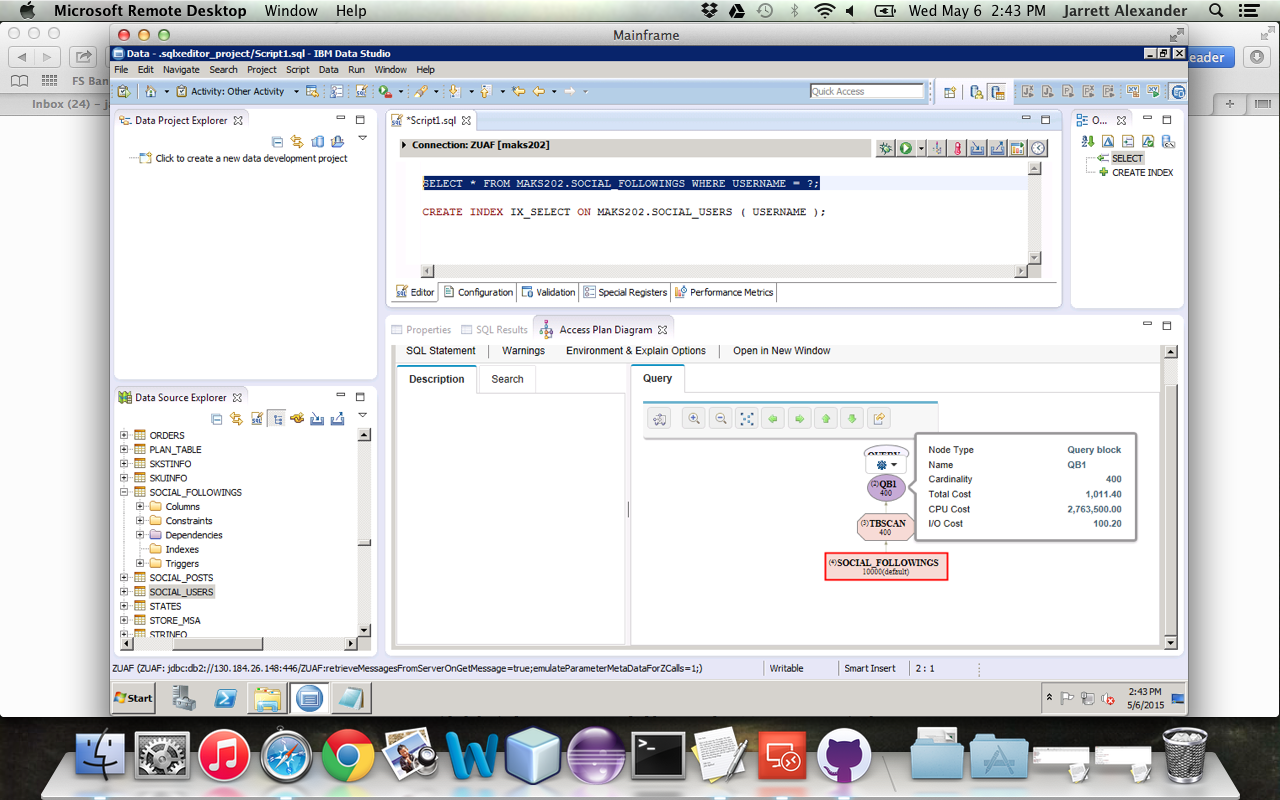
After index. I believe there’s not a change because username is the primary key for this table.



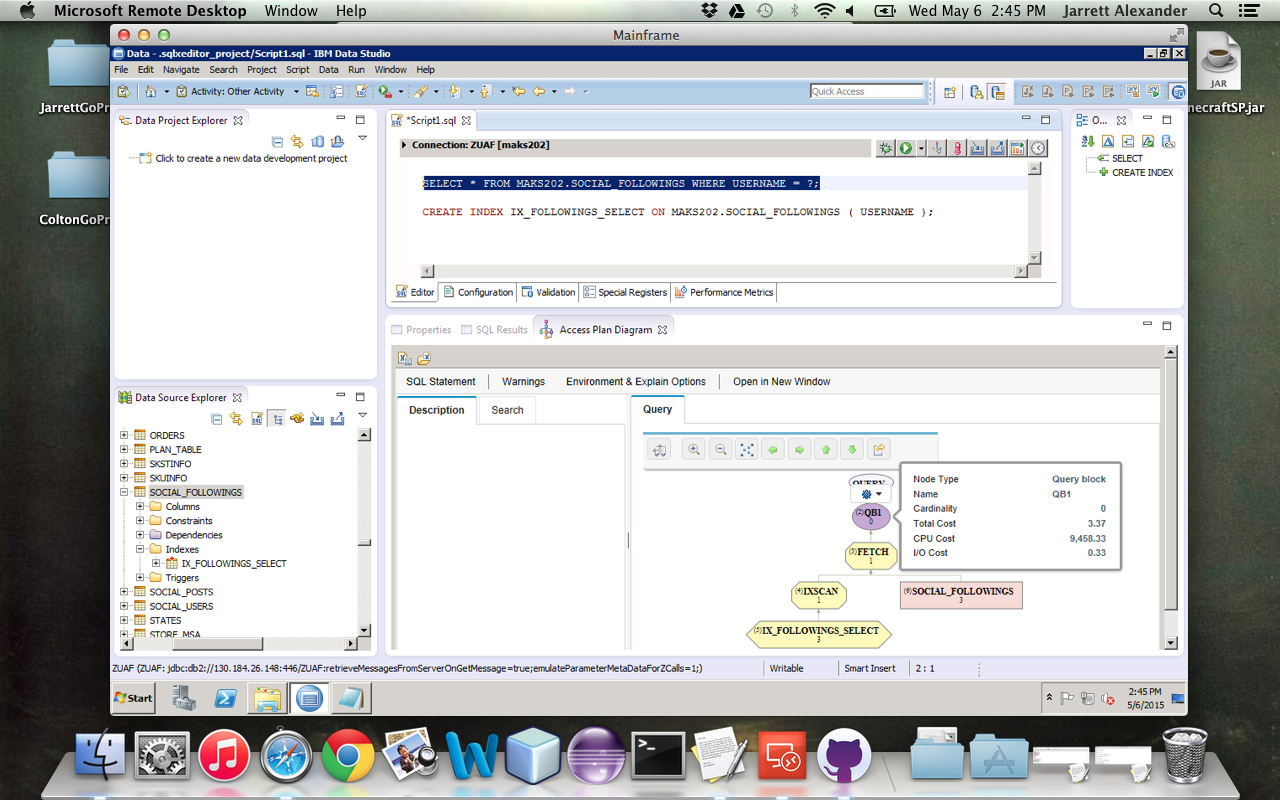
Social\_Followings table:

SELECT \* FROM MAKS202.SOCIAL\_FOLLOWINGS WHERE USERNAME = ?;

Retrieving following info by username. Before index:



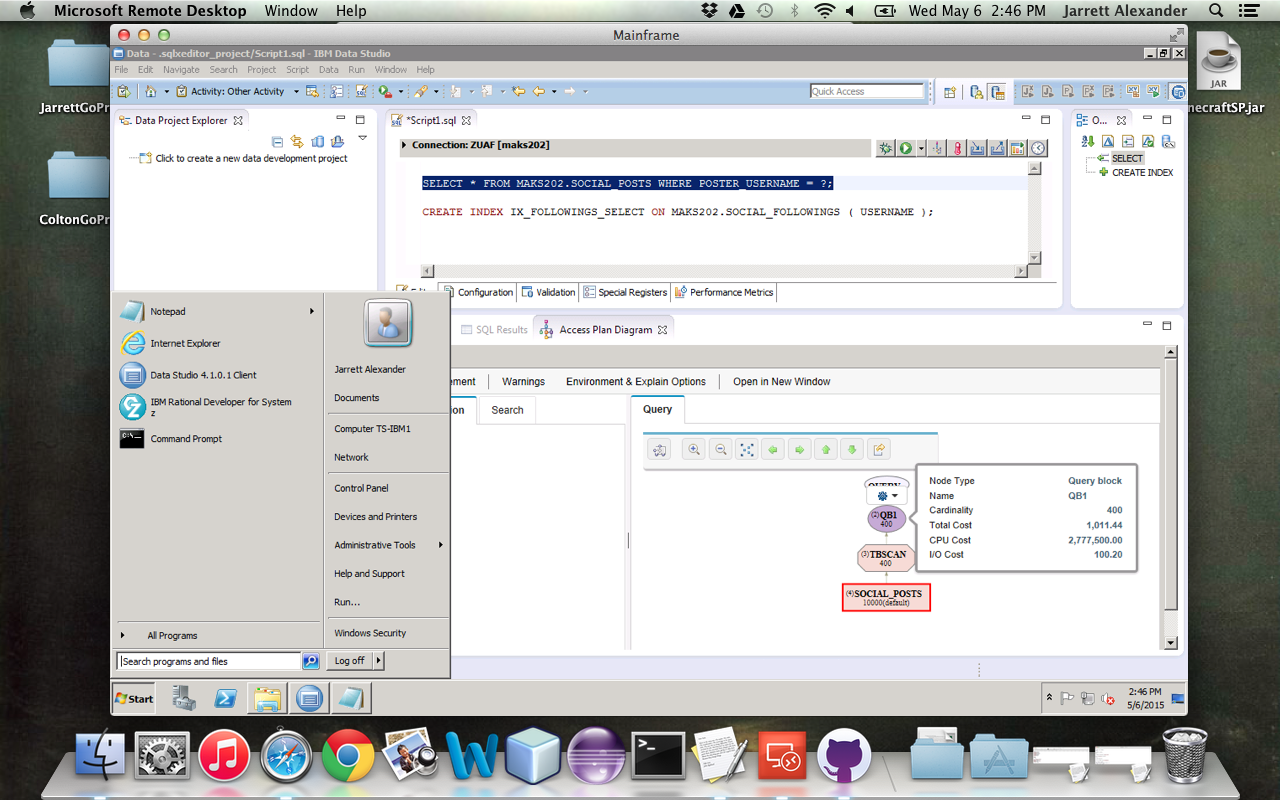
After index:



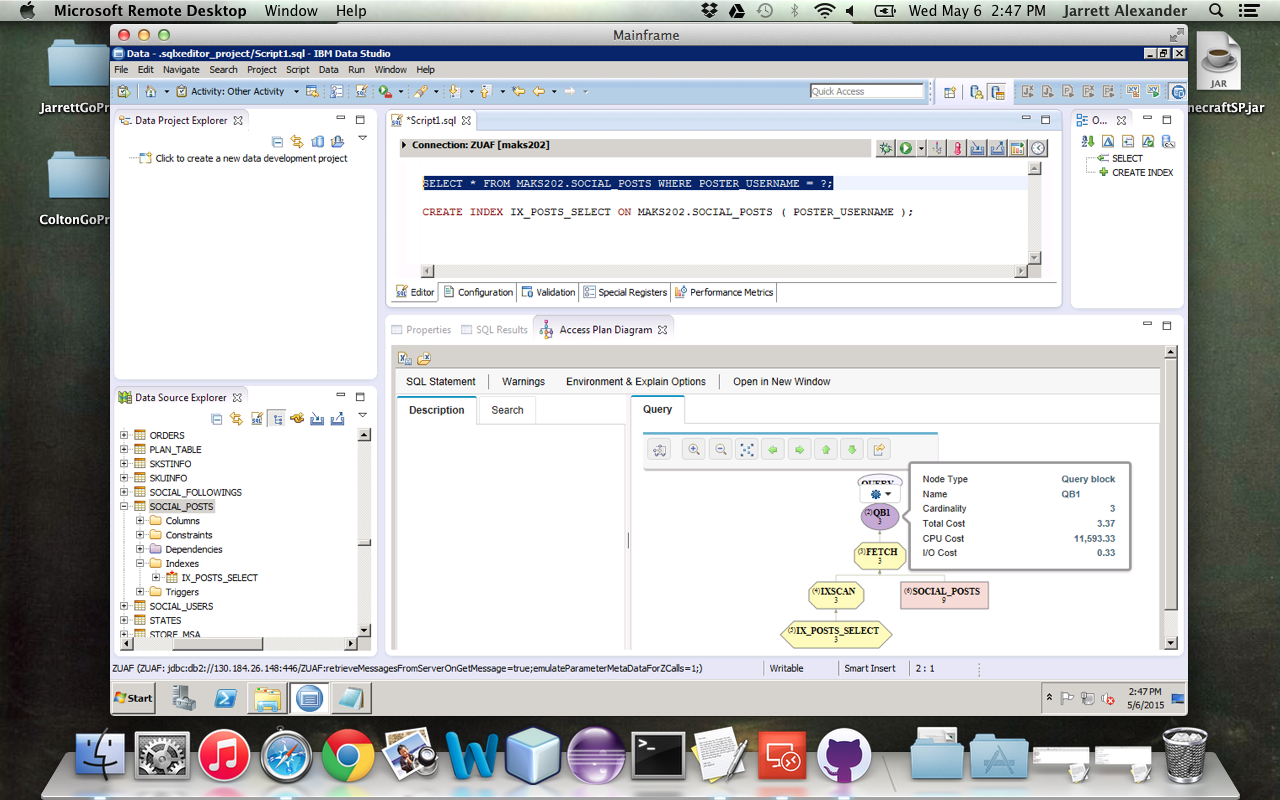
Social\_Posts table:

SELECT \* FROM MAKS202.SOCIAL\_POSTS WHERE POSTER\_USERNAME = ?;

Retrieve all posts by user. Before index:



After index:



SQL Not Indexed:

I read that insert statements run slower the more indexes there are on a table, so I did not index the insert statements.

INSERT INTO MAKS202.SOCIAL\_USERS( USERNAME, PASS, REAL\_NAME ) VALUES ( ?, ?, ? );

INSERT INTO MAKS202.SOCIAL\_FOLLOWINGS VALUES ( ?, ? );

INSERT INTO MAKS202.SOCIAL\_POSTS VALUES ( ?, ?, CURRENT\_TIMESTAMP );