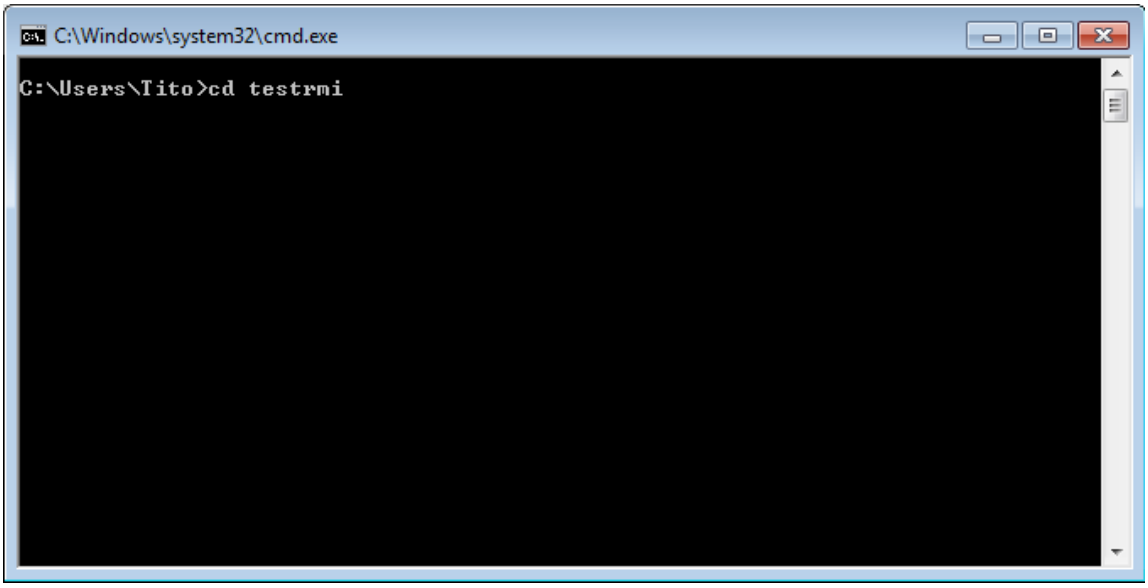
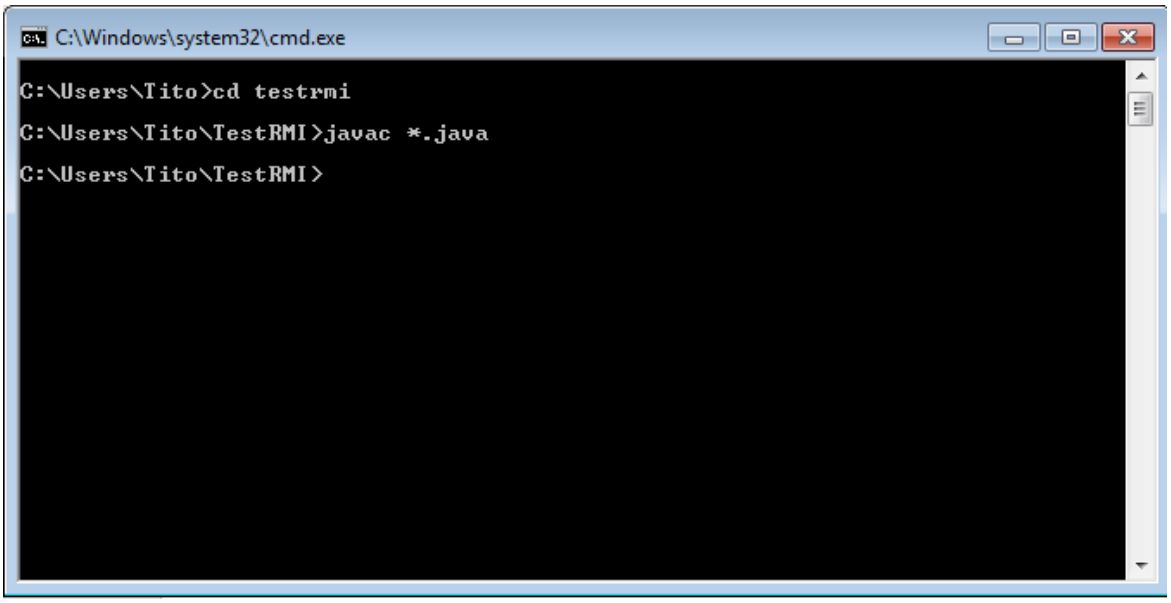
Java RMI Testing

Step-by-step instructions for getting the Java RMI test up and running. For those running on Mac, the process should be almost identical, just with slightly different terminal commands, I suppose. I'll let you figure out the equivalents for Windows cmd commands. For anyone using IntelliJ, you can also follow the same process in the IntelliJ terminal (default Windows shortcut to pull it up is **Alt+F12**). Presumably the process is the same for whatever IDE you're using. Any questions? Ask Tito in Slack.

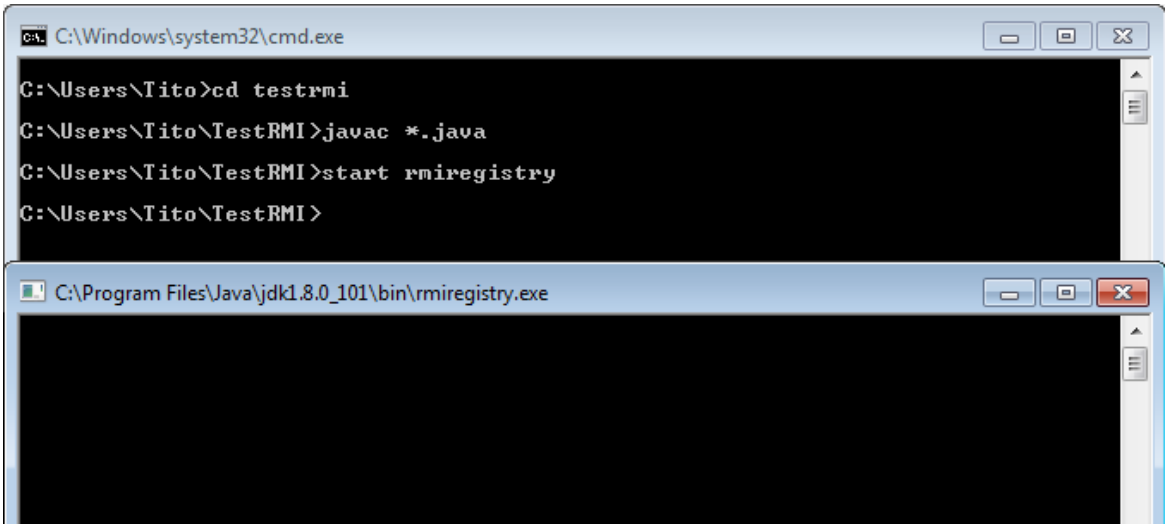
1) Create an empty folder somewhere to house all the .java files and the mysql jdbc .jar file. Once you've got them all downloaded, open your command prompt/terminal and **cd** to that location. In my case, **cd testrmi**



2) Next, we'll compile all the java files using **javac *.java**

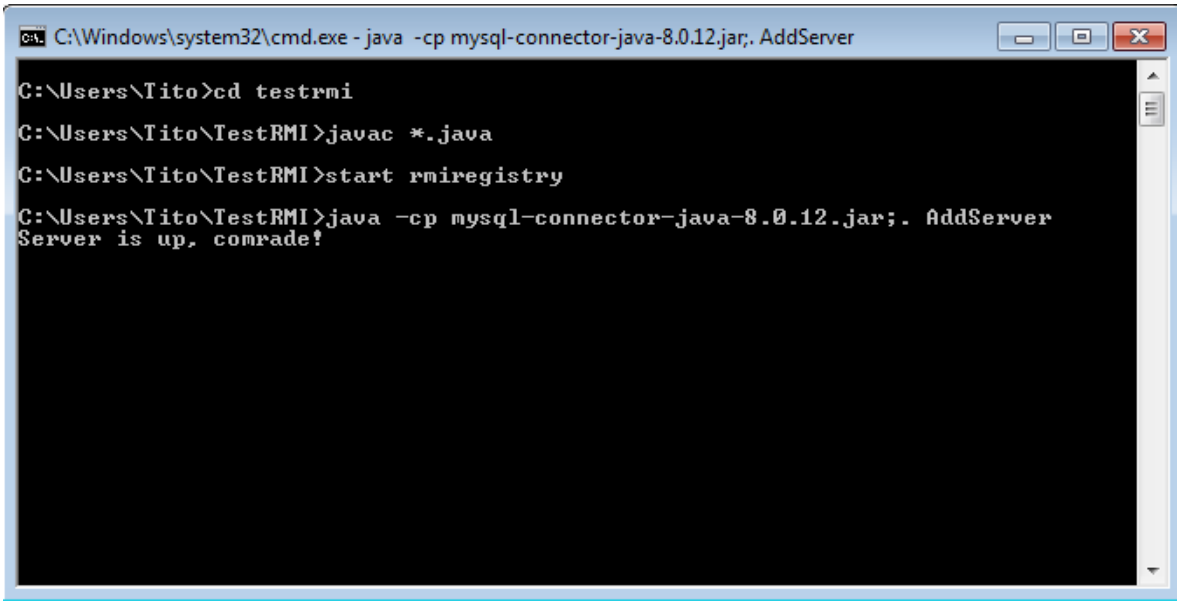


3) Now we have to **start rmiregistry** to bind remote objects to a name, which the client can then use to invoke those methods from the server

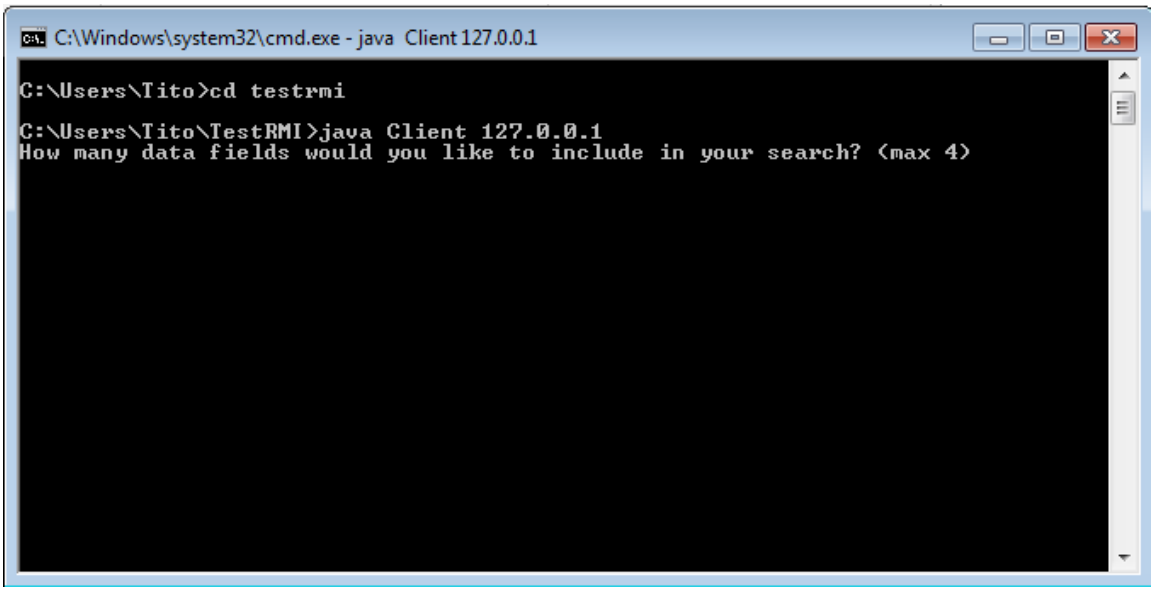


A new blank terminal window will open. That's the registry running; don't close it, just go back to your original terminal window.

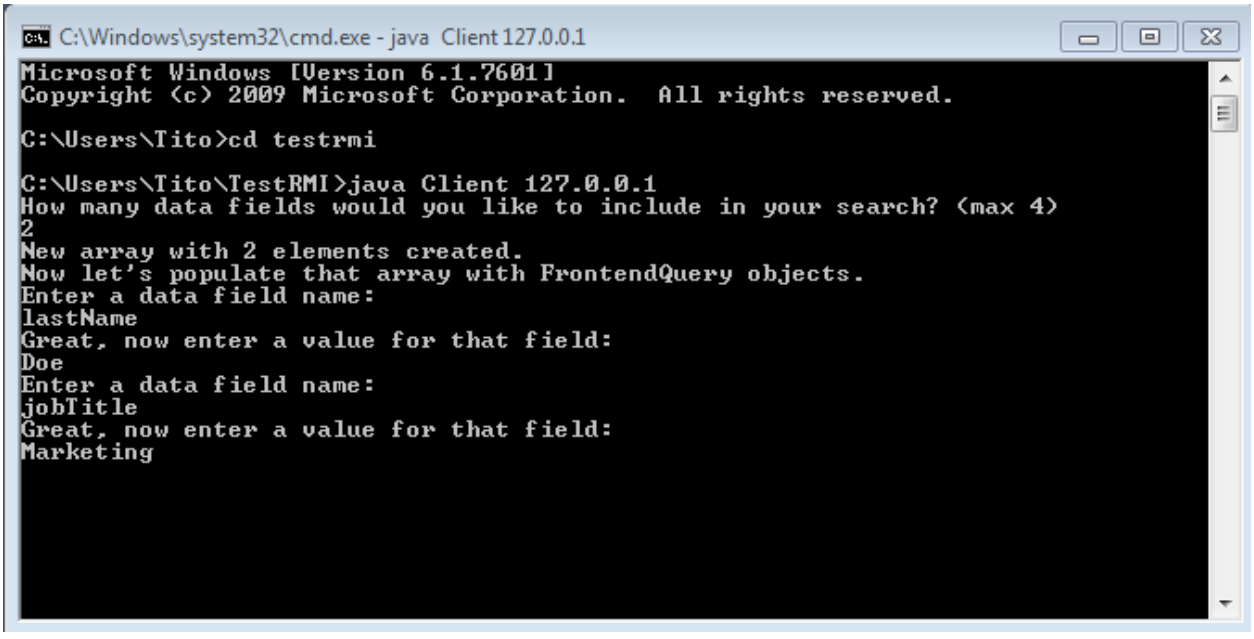
4) Now we can go ahead and run the server. Since we're using the jdbc driver, we need to declare that in the classpath when we run the server application. We can do that by writing **java -cp mysql-connect-java-8.0.12.jar;. AddServer**



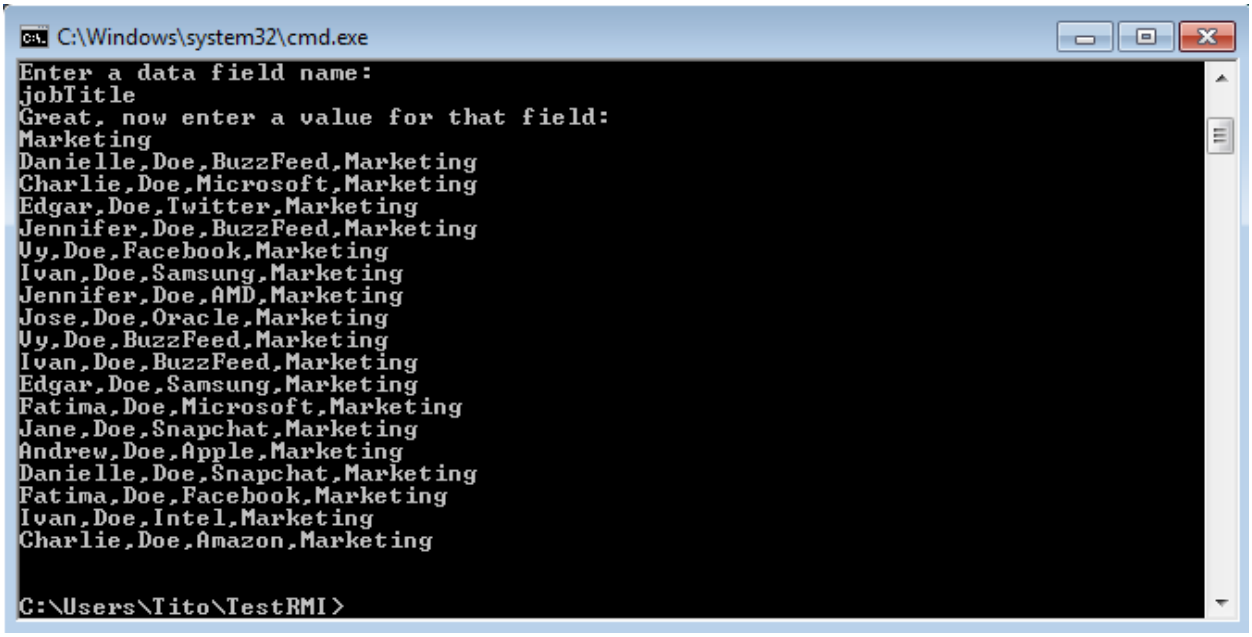
5) Next we'll start up the client application. Open a new terminal window and **cd** into the same directory as before. Then run **java Client 127.0.0.1**



6) Follow the onscreen prompts in the application. Valid data field names include: *firstName*, *lastName*, *workplace*, and *jobTitle*. Any values can be entered for the next prompt, but for a list of data values that are in the test database, refer to the *user-test-data-pool.csv* file.



The results will then be return row by row, delimited by commas in the following order: *firstName*, *lastName*, *workplace*, *jobTitle*

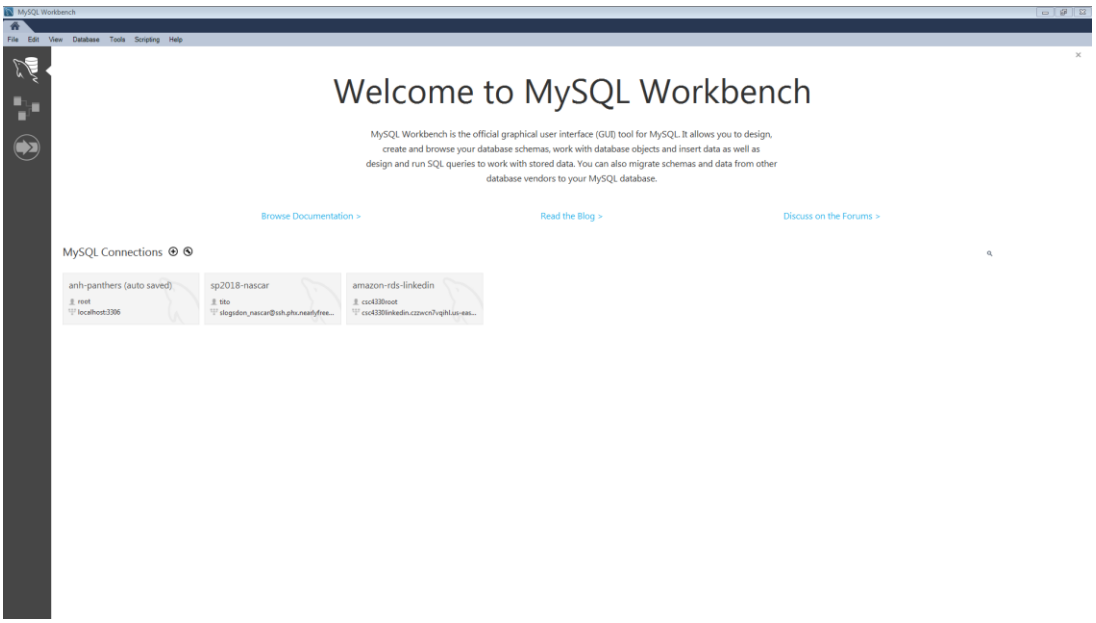


NEXT PAGE: connecting to database in MySQL Workbench

CONNECTING TO THE DATABASE IN MYSQL WORKBENCH

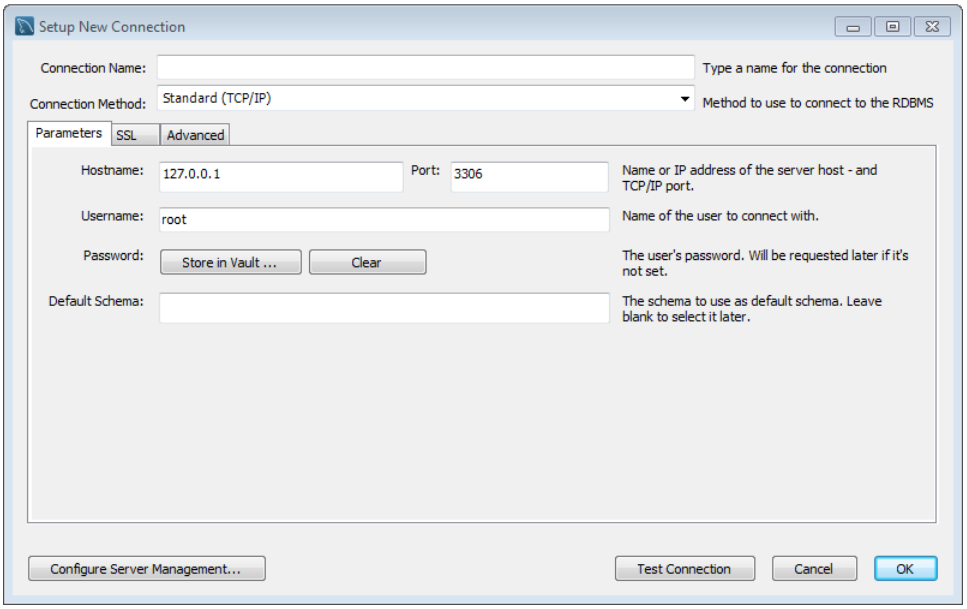
I like to use MySQL Workbench because it's free and easy to use. Need I say more? I've set up a test database on Amazon RDS, but being able to visualize the database in a nice GUI from a desktop application is useful throughout the development process. Here's a quick step-by-step on setting that up. I won't go over how to install MySQL Workbench, just how to establish a connection with the database from the software.

1) When you open MySQL Workbench, you should see a home screen like this



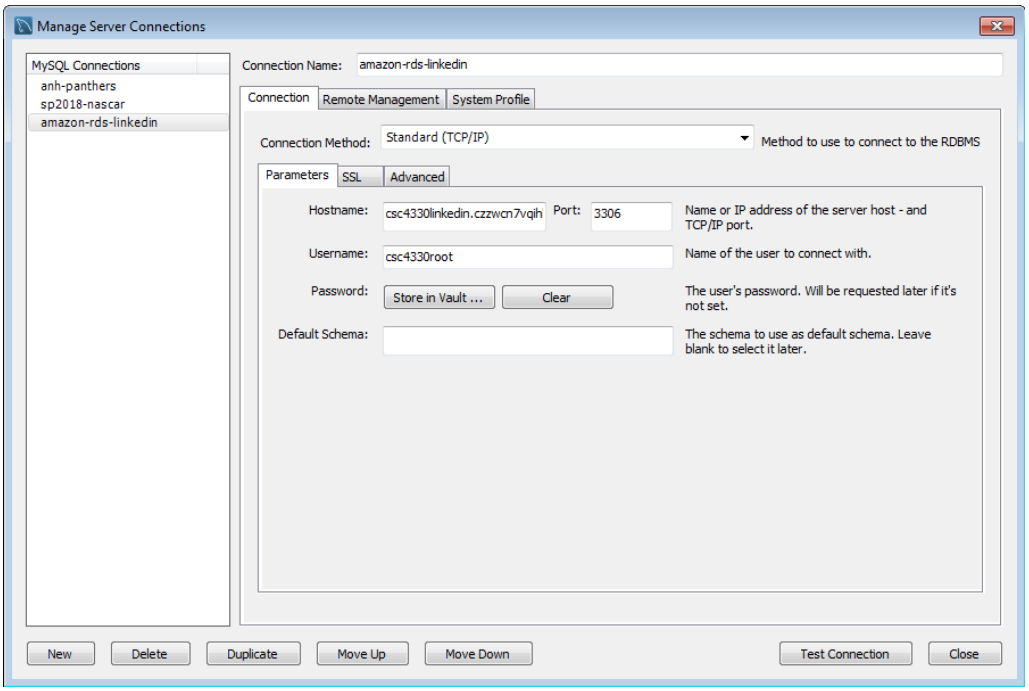
Click the little **+** symbol next to *MySQL Connections*

2) A window like this should pop up

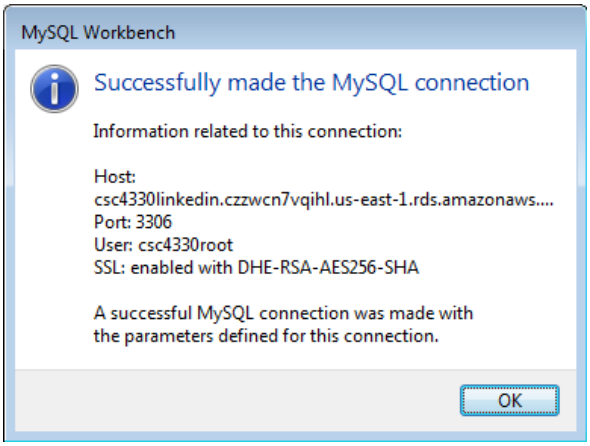


In the *Connection Name* field, enter whatever name you like for the connection. Leave the *Connection Method* as *Standard (TCP/IP)*. In the *Parameters* tab, set the *Hostname* to **csc4330linkedin.czzwcn7vqihl.us-east-1.rds.amazonaws.com**
Set the *Port* to **3306**
Username should be **csc4330root**
If you select *Store in Vault* next to *Password*, you won't have to type the password every time you're going to connect. The password is **LinkedIn2daDB!**

3) Once that information is entered, click *Test Connection*.



You should see a success message like this:



Click *OK*, then click *OK* again on the main window.

4) Your connection is now set up, and you'll see it appear on your home screen in Workbench. Just click on the button for that connection, and it'll open right up to a GUI for managing the database. Simply double-click **testdb1** in the left pane to begin working with the test data through Workbench.