MySQL connection from RStudio.

There were some non-trivial setup steps.

I ssh'd into the server with "ssh normie@134.122.18.100" pw = generic. I logged into mysql from there with "mysql -u normie -p" pw = doge. Then, I created two databases: "skills" and "cars". You can do this from Workbench, as well, using the regular TCP/IP protocol, no ssh required.

Issue being, you need to declare a dbname in the connection in the first code chunk, or else R pukes all over itself. I got rid of the individual usernames because it was too damn messy.

I guess, I will leave the server up and running for a bit in case people from the class want to play around with it. Maybe, I'll lock down the permissions a bit more at that point.

This is the moneymaker:

```
library(RMySQL)
```

```
## Loading required package: DBI
```

```
drv = dbDriver("MySQL")
con = dbConnect(drv,
                host = "134.122.18.100",
                port = 3306,
                user = "normie",
                password = "doge",
                dbname = "cars")
```

"Good 'ole mtcars!"

We make a column called model for the row names, which contain the model name of each car.

```
tester <- mtcars
tester$model <- row.names(tester)
```

We create a table in the "cars" DB called "mtcars"

```
query <- "CREATE TABLE mtcars(
  mpg NUMERIC,
  cyl INT,
  disp INT,
  hp INT,
  drat NUMERIC,
  wt NUMERIC,
  qsec NUMERIC,
  vs INT,
  am INT,
  gear INT,
  carb INT,
  model TEXT);"

full_send <- dbSendQuery(con, query)

dbListTables(con)
```

```
## [1] "mtcars"
```

Now we can populate the table. I suppose it's smarter to do this all in one query, but I'm no expert in MySQL… YET.

```
columns <- colnames(tester)

dbWriteTable(con,
             value = tester,
             row.names = FALSE,
             name = "mtcars",
             append = TRUE)
```

```
## [1] TRUE
```

When I first tried this query, I got the following error:

```
Error in .local(conn, statement, ...) :
  could not run statement: Loading local data is disabled; this must be enabled on both the client and s
```

Fixing this required editing /etc/mysql/mysql.conf.d/mysqld.cnf to allow local infile loading. This conf file should be in the same location on Mac, I think, but on a Windows server, there will be all manner of other hoops. I first ran “SET GLOBAL local_infile=1;” from the root mysql user, which seemed to work, but it was not persistent. Set the variable in the [mysqld] section of mysqld.cnf.

Finishing up.

We can drop tables from an R environment using dbRemoveTable(). Always close your connection when you are finished. :^)

```
dbRemoveTable(con, "mtcars")
```

```
## [1] TRUE
```

```
dbDisconnect(con)
```

```
## [1] TRUE
```