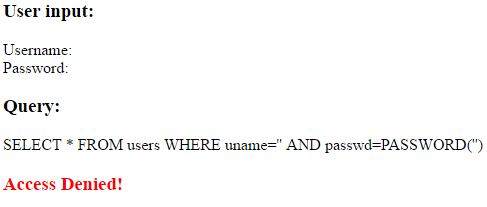
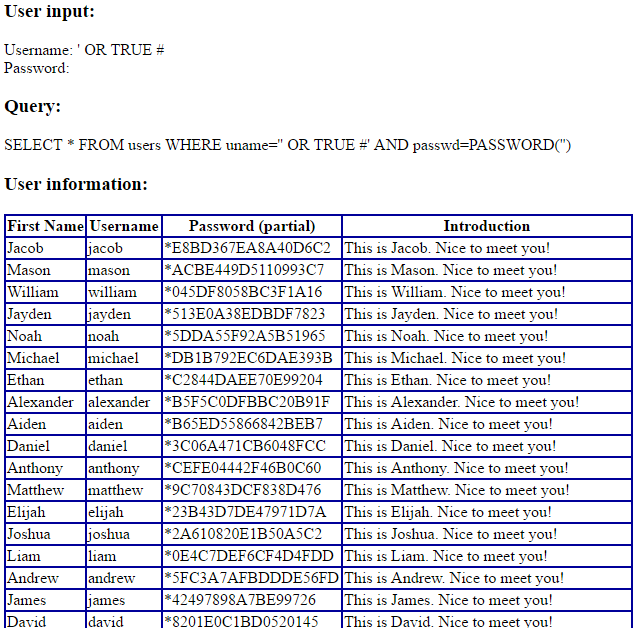
\*\*INITIAL NOTE\*\*

Hello Christopher. You’ll find throughout this report that I failed to read the assignment before beginning with the task. The assignment clearly outlines the database schema, the password function, and other useful information. Like I said, I did not read this until after I was ready to turn in the assignment. I did leave all of my work though, for your enjoyment.

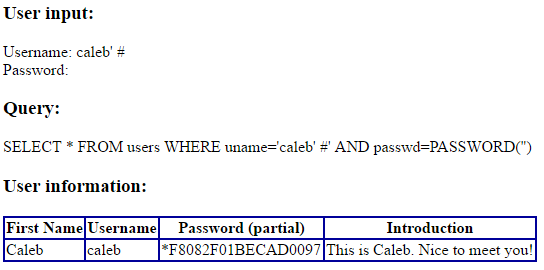
To begin, I simply clicked login, which too me to a redirect page which printed out the query of which I put in the database.



Next, I tried to see if I could access any users. I into the username field: ‘ OR TRUE #. This command ended the username selection, added another predicate statement, which was always true. This injection allowed me to receive the entire list of users in the database. **(steal all records in table)**

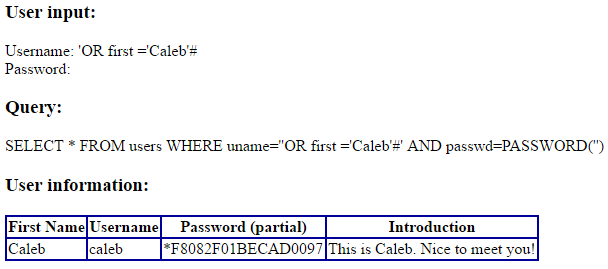


With this information, I could **impersonate any user without providing their password** with an injection into the “username” field such as: [uname]’ #



However, what if I wasn’t able to known the list of users, and I only had their first name? I could use try to target their first name field instead…I tried all the combination of the column names, such as “fname, name, firstName, FirstName, and Name” until I finally figured out the column name was “first”. I was then able **to impersonate a user by only knowing their name** using the command:

‘ OR first=’[name]’#.



So, the last task is to insert a record. Ok, shouldn’t be too bad. I’ll end the query with a semicolon, and insert a record using the insert command. After a few attempts, I was **able to insert a record using the command:**

';INSERT INTO users VALUES ([first], [uname], [passwd], [introduction])#

// For my user, I entered the exact command below:

';INSERT INTO users VALUES ('Jay', 'jaysanco', 'wtfdyjfsamylbihyk', 'https://www.youtube.com/watch?v=dQw4w9WgXcQ')#



My record was inserted, but the password was inserted as plaintext…this would look fishy in the system. So I did some research, and PWDENCRYPT looked like a promising functions.

';INSERT INTO users VALUES ('Jay', 'jaysanco', PWDENCRYPT('wtfdyjfsamylbihyk'), 'pls give hash')#

However, this didn’t work. Next, I tried using the built in MD5 function, which gave a weakly encrypted version of my navy seal password.



It was at this point I realized I wasn’t thinking correctly. I noticed that the input query gave a user defined function “PASSWORD”, which was responsible for hashing the input password. And, ta-da, we got it! I used the command:

';INSERT INTO users VALUES ('Jay', 'jaysanco', PASSWORD('wtfdyjfsamylbihyk'), 'pls give hash')#

