Problem Statement:

My goal was to create an authentication system based on a user gaining access to specified areas of a computer system based on their roles, determined by the credentials they enter when initially logging in. Users should also have the option to exit this program via inputs or switch roles via logging in after a successful attempt has occurred.

Overall Progress:

The live chat helped me considerably this week. I was on the path to hardcoding roles and realized this did not mesh with the intention of the assignment. It made it a lot more static vs the dynamic nature of what good code should be. The start of my confusion came when incorporating the MD5 hash file.

Pseudocode:

Pseudocode for Option 1: Authentication System

WHILE username and password attempts are less than/equal to 3

WHILE username DOES NOT equal exit

READ username

READ password

CHECK password using MD5 hash

CHECK username with credential file

CHECK password with credential file

IF Username/Password are correct

ALL user access to specified role

IF Username/Password are incorrect

INCREMENT attempts

IF username and password attempts are more than 3

NOTIFY user of 3 incorrect attempts

EXIT program

IF user input equals EXIT

EXIT Program

Methods and Classes:

As shown in the tutorial/weekly discussion video, I utilized a User Role class, used for getting the role name from the credentials file and returning it, stripped of extraneous whitespaces/tabs. It then sets the role to the “roleInfo” variable. This is then used to select the proper access file, depending on which user logs in.

I also used the MD5Digest class. This is used to run the input password through an MD5 hash and format it appropriately. This is used to verify against the credential file to ensure the proper password input and the password input space isn’t blank, via the “j” variable.

Error Documentation:

My biggest point of confusion came with the MD5Digest class. I found myself researching inputs into functions, the returns from functions and the proper syntax in Java. I also had some issues with exiting the program after a user successfully logged in. This was due to me not providing a way to input other text following a proper login.

Solution Documentation:

Reviewing Zybooks and paying close attention to the tutorials helped me solve my issues and align the proper variables with what I was attempting to pass to the class and return from it. Based on reviewing several www.stackoverflow.com issues and the provided tutorials, I found my loops were not properly nested and led to issues that I had yet to discover as well.