Ted Tippets

Final project

CS-499

SNHU

# Narrative for software development

The artifact I am working on is IT-145 I have built the GUI’s for the program. It has the many functions now it is completely running from the GUI. The software will do lots of things for the zoo now. The software has a logon page and depending on the username is where the software will go next. If it is admin it will go to the admin page. The admin page can put users, and passwords into the SQL database. All other users will go the other database pages. The software will put items into the SQL database for the animals, and the habitats.

# Justification for software development

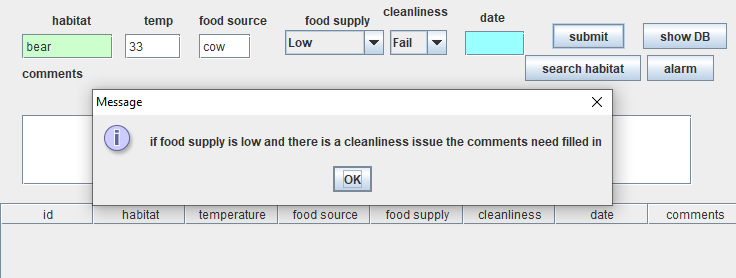
I included this artifact in the eportfolio because I believe it can showcase what I can do. I also figured I could put all three criteria into one artifact this way. I like that the software can handle all the SQL database end. The new GUI’s and the way they work with the program should show that I can make software that is works as intended. The artifact is improved by not being console ran. It can be used from a jar file. The artifact can be added to, and now has a date in it so that the next user can see when the information was put in. The next user can see if everything is OK by searching the DB, or by searching for a specific animal, or habitat. The artifact is very user friendly. It is easy to select from a jcombobox and hit submit to get to the only options available. Then the user is limited to what is needed. For the pass fails that is their only choice that way the program knows if it needs to throw an error in the future for the alarms. The program will show the most current 1000 rows in the database so the user can see more info if needed. They can type an animal and click show just that animal. They can also type the animal and show any current alarms for that animal. The habitat is the same way. The artifact also checks to see if the items are blank. The program also has the option to be updated with a logon if needed also.

# Reflection for Software

As I was enhancing the artifact, I learned that I need to think of all the possibilities of inputs for the software. For example, the if statements to handle the scenarios for the error messages. I cannot just place them in any order. In the habitat the scenario below would only catch the food supply issue if my else if statement started with only covering food, or cleanliness. The first statement in the else if after checking for the items to be filled in had to check for them next.

} else if (d.getSelectedItem().toString().equals("Low") && e1.getSelectedItem().toString().equals("Fail") && g.getText().isEmpty()) {

JOptionPane.showMessageDialog(null, "if food supply is low and there is a cleanliness issue the comments need filled in");. I also struggled with getting this line to work for a little bit. I kept trying to use || instead && and could not figure out why it was not working properly. After reading through that line a few times and realizing I needed to switch to && I got the results I was hoping for.



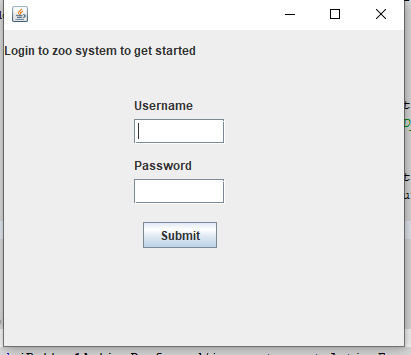
The software was improved by making it a GUI program. The software also has tons of functionality.

# Narrative for Algorithms and data structures

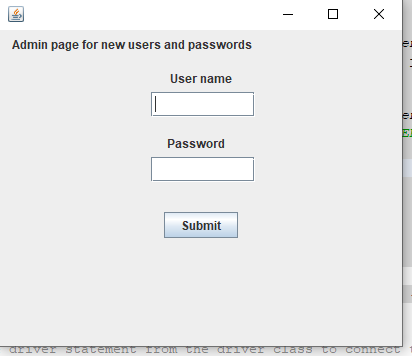
For the algorithm portion of this assignment I created a logon screen to put in an algorithm. The algorithm creates a hashed password that puts into the SQL database so that it is not plain text. The algorithm also creates a hash to verify the password in the SQL database.

# Justification

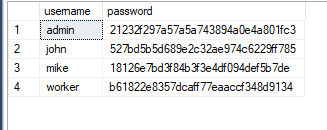
My justification for this algorithm is to add security to the program. I used a hash algorithm to hash passwords going into the database. The software starts with a sign on screen now.

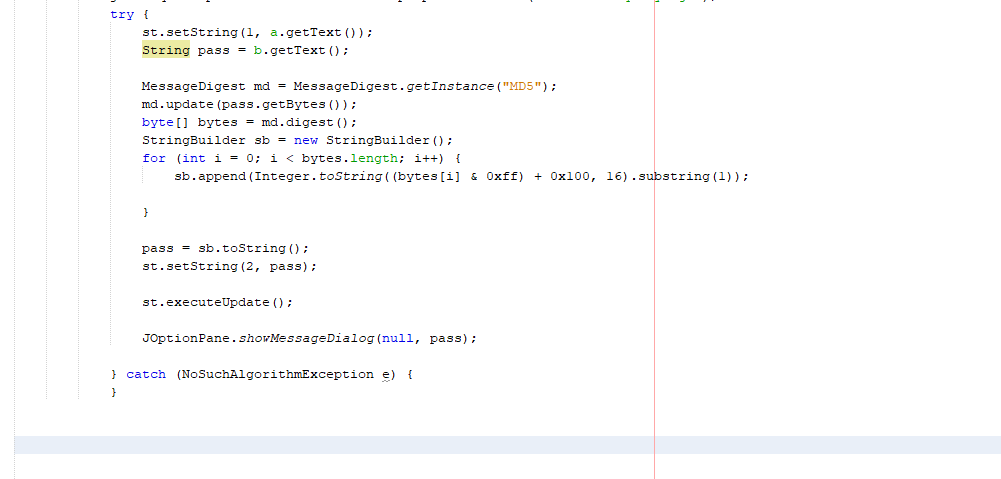


If the user is admin the admin is taken to a screen to add new users and passwords to the database.

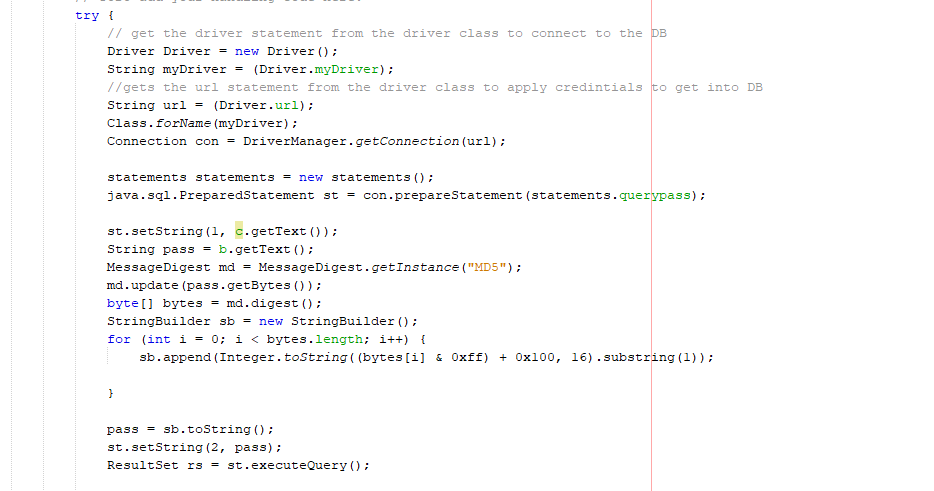


The SQL database show that the passwords are inserted hashed. The username is also the primary key so it will not allow duplicates.



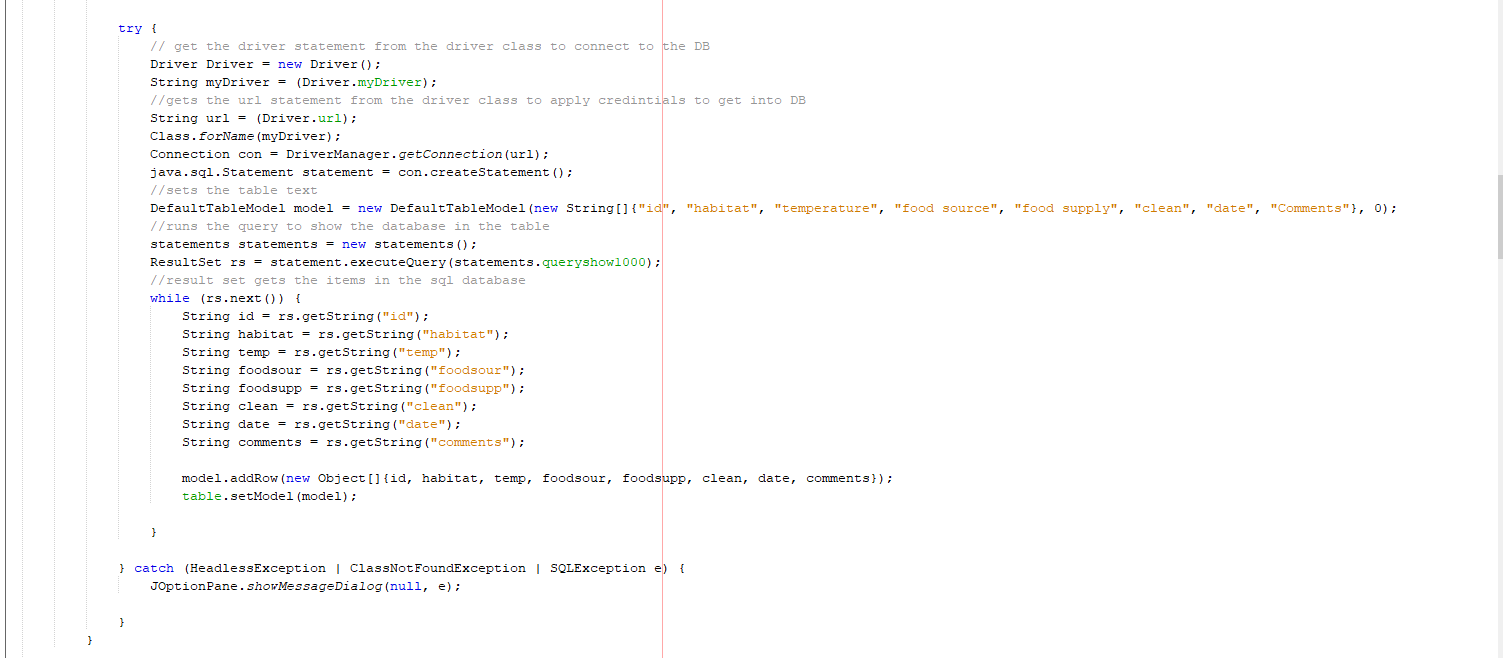
My algorithm for inserting the hash into the database is

My algorithm for validating the password hash is





For some data structures I have the SQL database storing information and am using the following data structure to query the data and retrieve it.



# Reflection

I learned how to insert more security into my programming with this portion of the assignment. It was challenging for me to get it to work correctly. I had a hard time getting this line to work the way I wanted. sb.append(Integer.toString((bytes[i] & 0xff) + 0x100, 16).substring(1));. After getting that to work I had a hard time with the hash working. I ended up having to use my pass variable, and add the stringbuilder.toString for it to work. I was pleased with the results when I finished this portion though. The software is growing and improving with each enhancement. I learned the importance of not keeping passwords stored in plain text, and how to prevent security issues from an unauthorized person getting into the database.

# Narrative for SQL database

I incorporated the SQL database into the zookeeper program. This replaced the text file in the program. These enhancements made the database be able to store lots of data over time.

# Justification

The reason for utilizing the SQL database into the software is for the storage of data. I can store usernames, and passwords into the database. I can also store all the data that is needed for the software. The SQL database makes it so that historical data can be seen. The end user can eventually look for trends in health, or environmental concerns.

# Reflection

During this portion of the enhancement I learned lots about using SQL databases with software. Using SQL databases with the software make the handling of information very user friendly. I found that if I want to set the identity for the ID from the software, I must handle that by turning the identity set on for each insert. I did that to because the software is picking the next number based on the most recent number inserted into the database. If I wanted to insert a delete function in the future this would keep the ID going in order. I learned using java to handle the SQL queries that it is useful to use ? for the values as a placement holder that will be added later with the prepared statement.