Name: S M Ragib Rezwan
ID: 103172423
Tutor: Wei Lai
Task1:

Enterprise application has many ways to do that, one of which is the JDBC Realm. Here it uses glassfish server where we also mention the database table so that it knows which table to access. Furthermore, we also need to configure the following settings there:

- User Table: This is the database table that has the loggin details for admin, users, etc.

- Username Column: the column in the table that contains the username selected for the authentication pair

- Password Column: the column in the table that contains the password selected for the authentication pair

- Group name column: the column in the table that contains the group names which determine the level of access they have to the resources present in the website

- Password Encryption Algorithm: the algorithm that will be used to encrypt the data in the table

Basically, when we log in, it at first checks username and password pair against the columns present in the database. After that it checks the group name to decide what sort of access to give us (ie users cant access admin menu and vise versa). This is not only efficient and reliable but also requires the least amount of code to implement (as the glassfish is doing all the work for us)

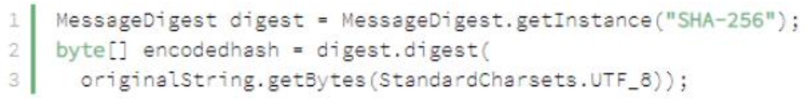
But since most of the work is handled by glassfish, we must be very careful when setting up the configuration details as otherwise the server will give out “strange” errors.

Task 2: Theory part

Here I am going to use EmpId and password as username password pair for authentication and appgroup as group name in order to determine how much access they have and to what resource.

Furthermore, I would set up a hashing algorithm for the password using the following codes:

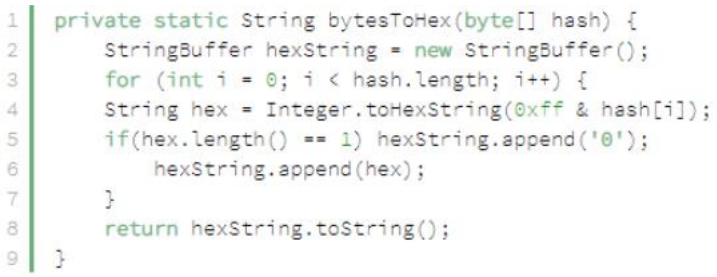
1) First convert the string to Byte



```
MessageDigest digest = MessageDigest.getInstance("SHA-256");
byte[] encodedhash = digest.digest(
  originalString.getBytes(StandardCharsets.UTF_8));
```

2) Then use a custom converter to make the byte into a hex and thus get the hash value in hex

Name: S M Ragib Rezwan
ID: 103172423
Tutor: Wei Lai

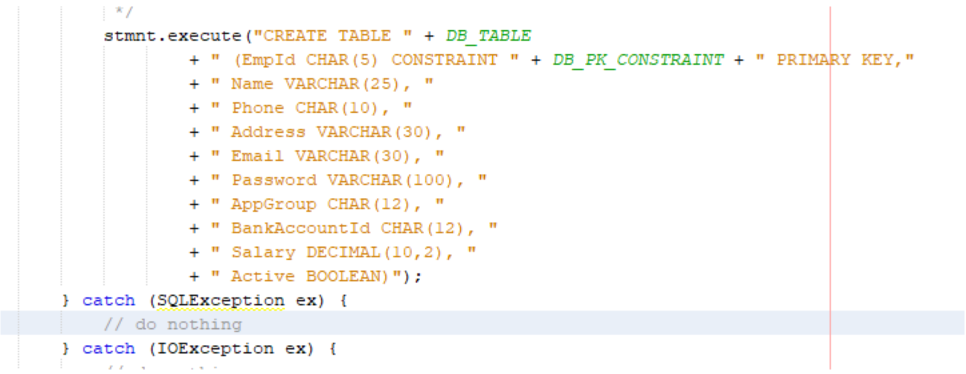


```java
private static String bytesToHex(byte[] hash) {
    StringBuffer hexString = new StringBuffer();
    for (int i = 0; i < hash.length; i++) {
    String hex = Integer.toHexString(0xff & hash[i]);
    if(hex.length() == 1) hexString.append('0');
        hexString.append(hex);
    }
    return hexString.toString();
}
```

Task 2: Practical part + outputs

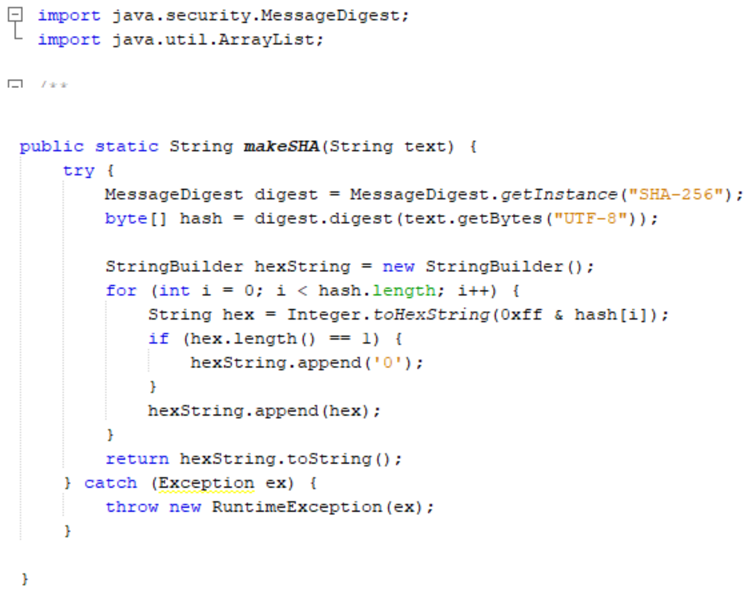
Screenshot for the adjustments I made in my code to make it work for me:

Table password size adjustment:



```java
        */
        stmnt.execute("CREATE TABLE " + DB_TABLE
                + " (EmpId CHAR(5) CONSTRAINT " + DB_PK_CONSTRAINT + " PRIMARY KEY,"
                + " Name VARCHAR(25), "
                + " Phone CHAR(10), "
                + " Address VARCHAR(30), "
                + " Email VARCHAR(30), "
                + " Password VARCHAR(100), "
                + " AppGroup CHAR(12), "
                + " BankAccountId CHAR(12), "
                + " Salary DECIMAL(10,2), "
                + " Active BOOLEAN)");
    } catch (SQLException ex) {
        // do nothing
    } catch (IOException ex) {
```

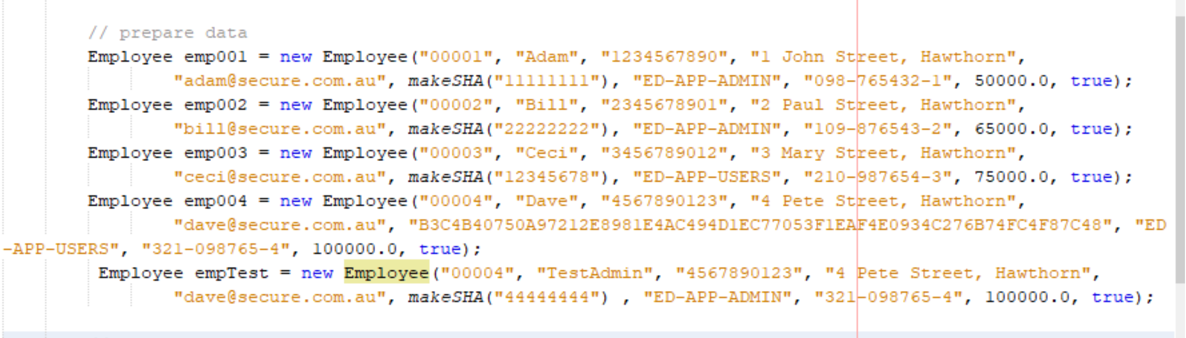
Method to turn text into sha256



```java
import java.security.MessageDigest;
import java.util.ArrayList;

/**

public static String makeSHA(String text) {
    try {
        MessageDigest digest = MessageDigest.getInstance("SHA-256");
        byte[] hash = digest.digest(text.getBytes("UTF-8"));

        StringBuilder hexString = new StringBuilder();
        for (int i = 0; i < hash.length; i++) {
            String hex = Integer.toHexString(0xff & hash[i]);
            if (hex.length() == 1) {
                hexString.append('0');
            }
            hexString.append(hex);
        }
        return hexString.toString();
    } catch (Exception ex) {
        throw new RuntimeException(ex);
    }

}
```

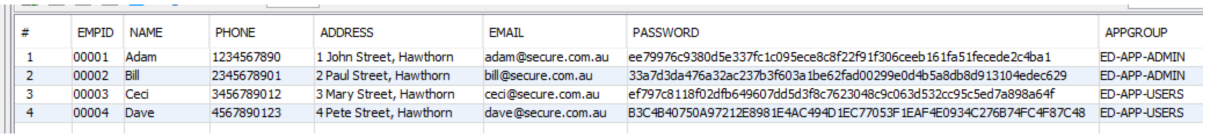
Name: S M Ragib Rezwan
ID: 103172423
Tutor: Wei Lai

Calling it in method to input data to table



```
        // prepare data
        Employee emp001 = new Employee("00001", "Adam", "1234567890", "1 John Street, Hawthorn",
                "adam@secure.com.au", makeSHA("11111111"), "ED-APP-ADMIN", "098-765432-1", 50000.0, true);
        Employee emp002 = new Employee("00002", "Bill", "2345678901", "2 Paul Street, Hawthorn",
                "bill@secure.com.au", makeSHA("22222222"), "ED-APP-ADMIN", "109-876543-2", 65000.0, true);
        Employee emp003 = new Employee("00003", "Ceci", "3456789012", "3 Mary Street, Hawthorn",
                "ceci@secure.com.au", makeSHA("12345678"), "ED-APP-USERS", "210-987654-3", 75000.0, true);
        Employee emp004 = new Employee("00004", "Dave", "4567890123", "4 Pete Street, Hawthorn",
                "dave@secure.com.au", "B3C4B40750A97212E8981E4AC494D1EC77053F1EAF4E0934C276B74FC4F87C48", "ED
-APP-USERS", "321-098765-4", 100000.0, true);
         Employee empTest = new Employee("00004", "TestAdmin", "4567890123", "4 Pete Street, Hawthorn",
                "dave@secure.com.au", makeSHA("44444444") , "ED-APP-ADMIN", "321-098765-4", 100000.0, true);
```

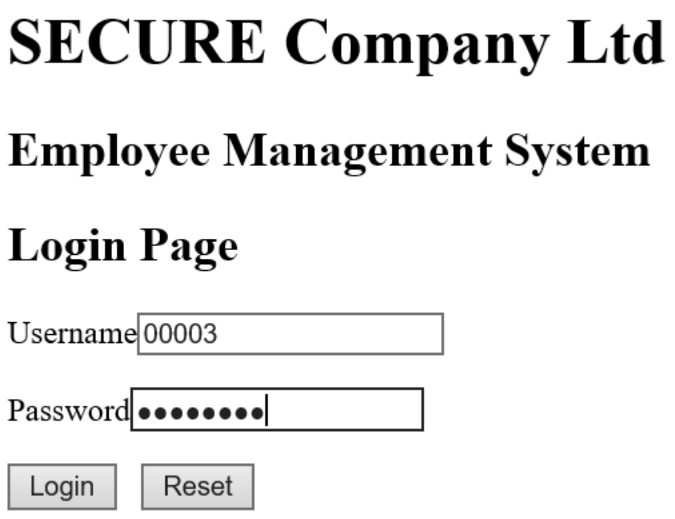
Output for table:



#	EMPID	NAME	PHONE	ADDRESS	EMAIL	PASSWORD	APPGROUP
1	00001	Adam	1234567890	1 John Street, Hawthorn	adam@secure.com.au	ee79976c9380d5e337fc1c095ece8c8f22f91f306ceeb161fa51fecede2c4ba1	ED-APP-ADMIN
2	00002	Bill	2345678901	2 Paul Street, Hawthorn	bill@secure.com.au	33a7d3da476a32ac237b3f603a1be62fad00299e0d4b5a8db8d913104edec629	ED-APP-ADMIN
3	00003	Ceci	3456789012	3 Mary Street, Hawthorn	ceci@secure.com.au	ef797c8118f02dfb649607dd5d3f8c7623048c9c063d532cc95c5ed7a898a64f	ED-APP-USERS
4	00004	Dave	4567890123	4 Pete Street, Hawthorn	dave@secure.com.au	B3C4B40750A97212E8981E4AC494D1EC77053F1EAF4E0934C276B74FC4F87C48	ED-APP-USERS

Also, website output for the codes:

Access by employee on employee page



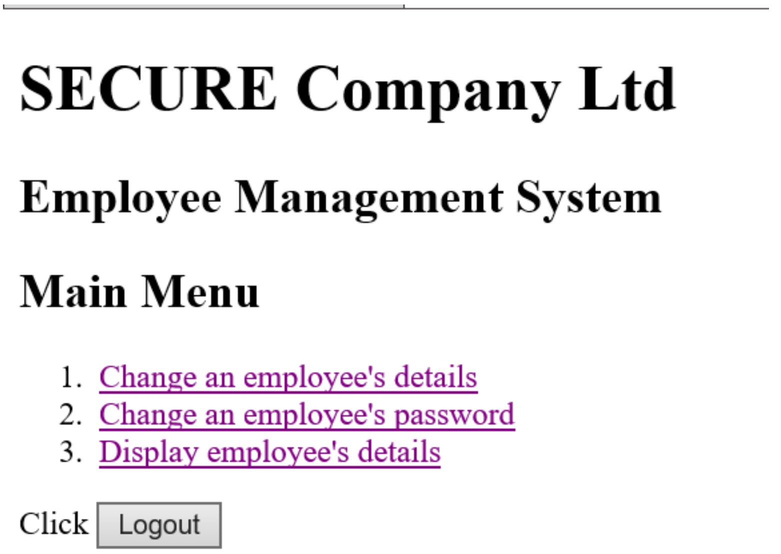
SECURE Company Ltd

Employee Management System

Login Page

Username: 00003

Password: ••••••••



SECURE Company Ltd

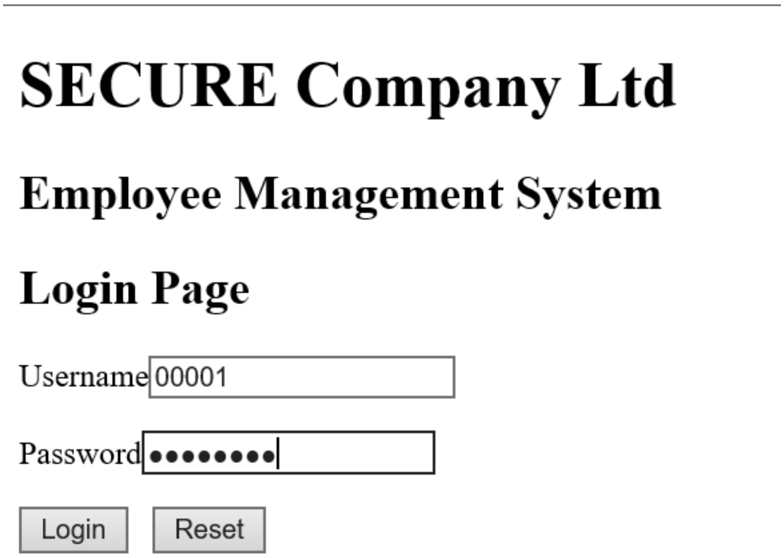
Employee Management System

Main Menu

1. Change an employee's details
2. Change an employee's password
3. Display employee's details

Click

Access by admin on employee page:



SECURE Company Ltd

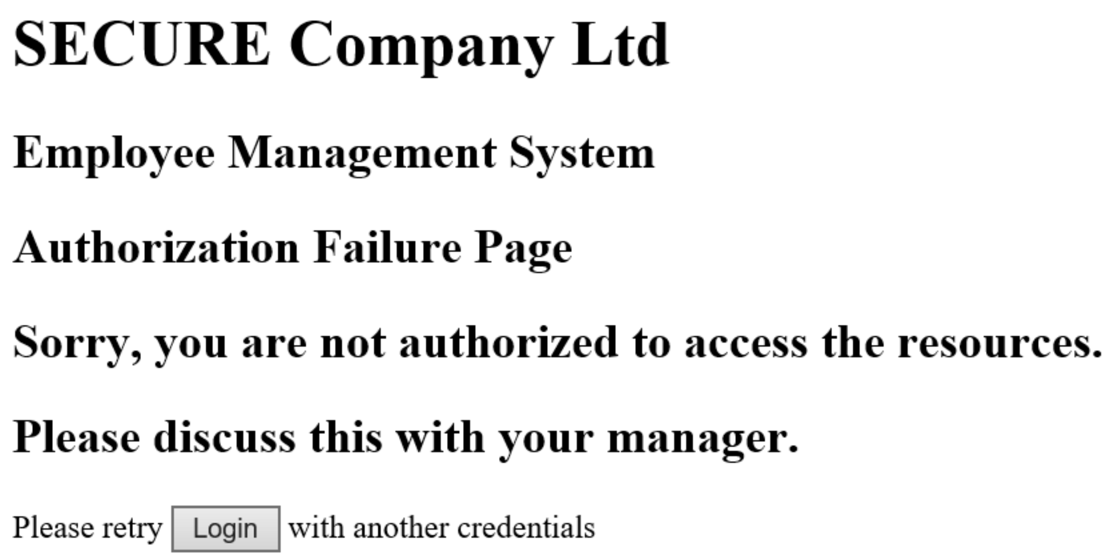
Employee Management System

Login Page

Username 00001

Password ••••••••

Name: S M Ragib Rezwan
ID: 103172423
Tutor: Wei Lai



SECURE Company Ltd

Employee Management System

Authorization Failure Page

Sorry, you are not authorized to access the resources.

Please discuss this with your manager.

Please retry Login with another credentials