# COMP2396B - Assignment 3

Due: 2 April, 2019 23:55

## Introduction

This assignment tests your understanding of **interface** in Java.

You are required to implement an authentication module for a system that manages users' login information. The authentication module allows the user to create and modify their information. It also provides the function to authenticate the user by password.

You need to design the program structure. Program design **will be evaluated** in this assignment. You must make good use of object-oriented programming in order to get all marks for this assignment. You must make use of **interface**, and **all instant variables must be private**. The marking criteria are included at the end of this assignment description.

You are also required to write **JavaDoc** for all non-private classes and non-private class members. **Submissions without JavaDoc will not be marked.**

## Tasks

When the program is executed, the following menu should be displayed:

```
Welcome to the COMP2396 Authentication system!
1. Authenticate user
2. Add user record
3. Edit user record
4. Reset user password
What would you like to perform?
Please enter your command (1-4, or 0 to terminate the system):
```

1. **Hashing interface**

   - Create an Interface class **Hash** to perform hashing (MD5, SHA1, SHA256 etc…)

     - Implement a hash function for the program to use.
     - To use the Java hash utility, please refer to:
       https://docs.oracle.com/javase/8/docs/api/java/security/MessageDigest.html

   - To allow the system to change hashing algorithm without modifying the program logic.

- The hashing interface should be used in the following:

  - Hash the user input password when adding a user.
  - Hash the user input password for the authentication.
  - Hash the user input password when the user change or reset the password.

## 2. Add user record

```
Welcome to the COMP2396 Authentication system!
1. Authenticate user
2. Add user record
3. Edit user record
4. Reset user password
What would you like to perform?
Please enter your command (1-4, or 0 to terminate the system): 2
Please enter your username: Raymond
Please enter your password: 123456
Your password has to fulfil: at least 1 small letter, 1 capital letter,
1 digit!
Please enter your password: 123456Abc
Please re-enter your password: 123456Abc
Please enter your full name: Raymond Chan
Please enter your email address: cbchan@cs.hku.hk
Please enter your Phone number: 12345678
Record added successfully!
Please enter your command (1-4, or 0 to terminate the system):
```

- Define a **User** Object to store the following:

  - Username
  - Hashed password
  - Full Name
  - Email address
  - Phone number
  - Failed login count
  - Last login date
  - Account locked

- Add a user record in the authentication system.

- Use the hashing function to hash the user input password, **the system should store the hashed password instead of the plain text password**.

- The program should check if the password fulfills the following requirement:

  - The password should contain at least 6 characters, with at least 1 small letter, 1 capital letter, 1 digit.

### 3. Modify user record

```
Welcome to the COMP2396 Authentication system!
1. Authenticate user
2. Add user record
3. Edit user record
4. Reset user password
What would you like to perform?
Please enter your command (1-4, or 0 to terminate the system): 3
Please enter your username: Raymond
Please enter your password: 123456Abc
Login success! Hello Raymond!
Please enter your new password: 123456Acb
Please re-enter your new password: 123456Acb
Please enter your new full name: Raymond Chan
Please enter your new email address: cbchan@cs.hku.hk
Record update successfully!
Please enter your command (1-4, or 0 to terminate the system):
```

- To modify the user record, the user has to provide the username and password before he can edit the record.

- If a user failed to login for 3 times, his account will be locked and not allowed to login. (Refer to part 4 for sample output)

- If the failed count is less than 3 and the user can login successfully, the failed count will reset to 0.

- If the failed count is greater than or equal to 3, the user account will be locked, i.e. the user will not be allowed to login again.

- When the user successfully login, the last login date will be updated.

- Allow user to change the Full name, password and email address.

- The program should check if the two new passwords entered are identical. If not identical, print out the information and prompt the user for the two new passwords in the above format again. (You can design this information for passwords mismatch in your own way, but there must be a line of information for it)

### 4. Authentication with error count and last logon date

```
Welcome to the COMP2396 Authentication system!
1. Authenticate user
2. Add user record
3. Edit user record
4. Reset user password
What would you like to add to perform?
```

```
Please enter your command (1-4, or 0 to terminate the system): 1
Please enter your username: Raymond
Please enter your password: 123456Abc
Login success! Hello Raymond!
Please enter your command (1-4, or 0 to terminate the system):
```

- The authentication module should validate if the user can provide correct username and password.

- When the user successfully login, the last login date will be updated.

- If a user failed to login for 3 times, his account will be locked and not allowed to login.

```
Welcome to the COMP2396 Authentication system!
1. Authenticate user
2. Add user record
3. Edit user record
4. Reset user password
What would you like to perform?
Please enter your command (1-4, or 0 to terminate the system): 1
Please enter your username: Raymond
Please enter your password: 147852369
Login failed!
Please enter your username: Raymond
Please enter your password: 147852369
Login failed!
Please enter your username: Raymond
Please enter your password: 147852369
Login failed!
Please enter your username: Raymond
Please enter your password: 147852369
Login failed! Your account has been locked!
Please enter your command (1-4, or 0 to terminate the system):
```

- If the failed count is less than 3 and the user can login successfully, the failed count will reset to 0.

- If the failed count is greater than or equal to 3, the user account will be locked, i.e. the user will not be allowed to login again.

5. **Store the records in a file**

- The system has to read a file "User.txt" when it is executed. If the file does not exist, the program will create a blank "User.txt" automatically.

- To use the Java FileReader API, please refer to: https://www.tutorialspoint.com/java/java_filereader_class.htm

- Store the record in JSON format as follow (order doesn't matter):

```
{
  "user_array": [{
      "username": "administrator",
      "hash_password": "41a2e7557fe09833a68f0427c0e9a733",
      "Full Name": "Administrator",
      "Email": "admin@cs.hku.hk",
      "Phone number": 12345678,
      "Fail count": 0,
      "Last Login Date": "2019-03-26",
      "Account locked": false
    },{
      "username": "kpchow",
      "hash_password": "25f9e794323b453885f5181f1b624d0b",
      "Full Name": "Dr. Chow",
      "Email": "chow@cs.hku.hk",
      "Phone number": 12345678,
      "Fail count": 0,
      "Last Login Date": "2019-03-23",
      "Account locked": false
    }, {
      "username": "raymond",
      "hash_password": "25d55ad283aa400af464c76d713c07ad",
      "Full Name": "Raymond Chan",
      "Email": "cbchan@cs.hku.hk",
      "Phone number": 87654321,
      "Fail count": 1,
      "Last Login Date": "2019-03-24",
      "Account locked": false
    }, {
      "username": "mengqi",
      "hash_password": "fcea920f7412b5da7be0cf42b8c93759",
      "Full Name": "Mengqi Peng",
      "Email": "mqpeng@cs.hku.hk",
      "Phone number": 74185296,
      "Fail count": 0,
      "Last Login Date": "2019-03-23",
      "Account locked": false
    }, {
      "username": "changjian",
      "hash_password": "e10adc3949ba59abbe56e057f20f883e",
      "Full Name": "Changjian Li",
      "Email": "cjli@cs.hku.hk",
      "Phone number": 96385274,
      "Fail count": 0,
      "Last Login Date": "2019-03-23",
      "Account locked": false
```

```
        }
      ]
  }
```

- For details of JSON in Java, please refer to:
  https://www.tutorialspoint.com/json/json_java_example.htm

- Please use the jar file json-simple-1.1.1.jar we provided when using JSON.

- The system should store all up-to-date user records back to "User.txt" file
  using PrintStream when the program terminates.

- For PrintStream reference:
  https://docs.oracle.com/javase/8/docs/api/java/io/PrintStream.html

## 6. Reset user password

```
Welcome to the COMP2396 Authentication system!
1. Authenticate user
2. Add user record
3. Edit user record
4. Reset user password
What would you like to perform?
Please enter your command (1-4, or 0 to terminate the system): 4
Please enter the password of administrator: 741852Abc
Please enter the user account need to reset: raymond
Please enter the new password: 789456Abc
Please re-enter the new password: 789456Abd
Password not match! Please enter the new password: 789456Abc
Please re-enter the new password: 789456Abc
Password update successfully!
Please enter your command (1-4, or 0 to terminate the system):
```

- Only the "administrator" account can reset the user password.

```
Welcome to the COMP2396 Authentication system!
1. Authenticate user
2. Add user record
3. Edit user record
4. Reset user password
What would you like to perform?
Please enter your command (1-4, or 0 to terminate the system): 4
Administrator account not exist, please create the administrator account
by setting up a password for it.
Please enter the password: 741852Abc
Please re-enter the password: 741852Abc
Administrator account created successfully!
Please enter your command (1-4, or 0 to terminate the system): 4
Please enter the password of administrator: 741852Abc
```

```
Please enter the user account need to reset: raymond
Please enter the new password: 789456Abc
Please re-enter the new password: 789456Abd
Password not match! Please enter the new password: 789456Abc
Please re-enter the new password: 789456Abc
Password update successfully!
Please enter your command (1-4, or 0 to terminate the system):
```

- If the "administrator" not exists, prompt a message to ask the user to add an "administrator" account.

- The program should check if the password fulfill the following requirement:

  - The password should still contain at least 6 characters, with at least 1 small letter, 1 capital letter, 1 digit.

- The program should check if the password is correct. If incorrect, print out the information and prompt the user for the administrator password again. (You can design this information for incorrect password in your own way, but there must be a line of information)

- The fail count will reset to 0 after the user password is changed.

- The "Account locked" should be reset to false.

**7. Exit the program**

- When "0" is inputted to the system, your program should terminate


## Marking
- **50% marks** are given to the **program design.**

  - You must make use of interface in your program.

  - You should avoid duplicated code as much as possible.

- **40% marks** are given to the **functionality** of your program.

  - Your program output must be identical to what is described in this document, with the exception of the trailing spaces at the end of each line of output. Points will be deducted if your output is not identical to the sample output given in this document.

  - Please ask us on Moodle or by email if you are not sure about any of the requirements stated in this description.

- **10% marks** are given to your **JavaDoc**. A complete JavaDoc includes documentation of every classes, member fields and methods that are not private.

JavaDoc for the main method may be omitted.

- You should know the JavaDoc requirements clearly now. You will either get all marks or none for the JavaDoc written for this assignment. Missing JavaDoc for any individual non-private class or class member will cause you a 0 mark on the JavaDoc.

**Submission:**

Please submit all source files (`*.java`) in a single compressed file (in `.zip` or `.7z`) to Moodle. **Late submission is not allowed and will get a 0 mark. Please do not wait until the last minute to submit, we are not responsible for any late submissions regardless of what reasons you have.**

**Do not submit *.class* file.**

-- END --