JAVA FUNDAMENTALS PROJECT

Spring 2018

INFORMATION MANAGEMENT SYSTEM

PROJECT BY

PRASHANTH RAJASEKARAN

Table of Contents

Subject description

The Information Management System (IMS) Project is the final project that was taken with the intent to discover how to program in Java.

This Identity Management system is a not so complex but basic illustration of creating a project adhering to the rules of Business application. The main goal of this project is to manage users of an Information System.

The scope of this project is to create a tool that facilitates:

- Access, create and modify user information
- Persist users data in Derby database
- Robust, capable of good performance IMS

The project was partially completed during the usual tutorial sessions and rest is completed following the guidelines and from the learnings of the tutorial session.

The initial implementation of the project does not contain any UI.

Subject analysis

Major features

a) Authentication:

The IMS project is developed with the motive to be secure and robust. Hence Authentication is considered to be one of the major feature

b) Identity Creation:

Post successful authentication the user is prompted with options to choose from, and Identity creation being the first allows user to create a new identity and store the information of it in derby database.

c) Identity Updating:
The second feature enables the user to modify the identity information that was previously stored in the database.

d) Identity Deletion:
This feature facilitates the user to remove unwanted entries from the IMS.

Derby is used in the backend to store the information collected from the user.

Application Feasibility

Project Problems

Project does not possess any major problems or obstacles that we foresee during the feasibility study as its structure is well-learnt and observed before start.

The Outcome

Except for the UI capabilities, all the requested features (Authentication, Create, Update, and Delete) could be completed in the given span of time using the CLI.

Alternatives

Though there are several possible means are available for IMS project, Java is chosen for its robustness and its ease to write, compile, and debug. More importantly, it allows us to create modular programs and reusable code. Hence, no alternative is considered.

Assessment

The final assessment suggests that this project is less risky, easy to control, and hence an apt solution.

Data description

The project is built with the base of a single user administration, and hence no multiple user login is created in this project. A single user will manage all the identities that are created.

The identities are stored in the derby database, in a single table allowing the ease of access.

Access capabilities:

Read (R) Permission to read identity information.
Write (W) Permission to write to DB, includes also the capability to change.
Delete (D) Permission to delete the entries in the DB.

Expected results

Read and store information about identities.

- API to act on identities data
- With Authentication management
- Functionalities – Create, Update and Delete

Scope of the application

Identity Creation

Create multiple user accounts and store the information in DB. This helps in maintain security and user management in a much effective way. It also allows you to create multiple users that contain the standard user attribute values, to drastically reduce the amount of time you spend in paperwork's in identity management.

Searching Identity

Searching the identities in the DB with any of the following identity values username, email ID, UID etc.

Updating Identity

Updating the identity information in the IMS module thus in turn replacing the existing the details with the new values.

Deleting Identity

Selecting the Identity that needs to be removed from the IMS software easily just by inputting any one of the following values username, email ID, UID etc.

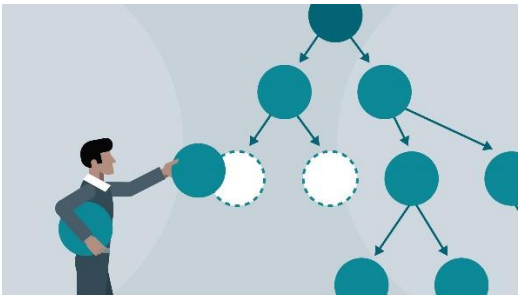
Conception

Chosen Algorithm

Steps:

1. Perform user authentication.
2. If successfully authenticated go to step 4.
3. If it is a failure, go to step 11.
4. Display the welcome message and user display menu.
5. Take Input from the user to select the operation to be performed.
6. When create is selected perform the identity creation operation and go back to step 4.
7. When Search is selected perform the identity search operation and go back to step 4.
8. When Update is selected perform the identity update operation and go back to step 4.
9. When Delete is selected perform the identity deletion operation and go back to step 4.
10. When Exit operation is selected, go to step 11.
11. Exit the Application.

Data structures



The IMS application only has two types of data to handle

a) User Authentication information
b) Identities Stored

User Authentication Information

User authentication information is stored and retrieved from the configuration file.

Since a single user administration is illustrated, it is a onetime hard encoding done at the time of the software creation.

Identities Stored

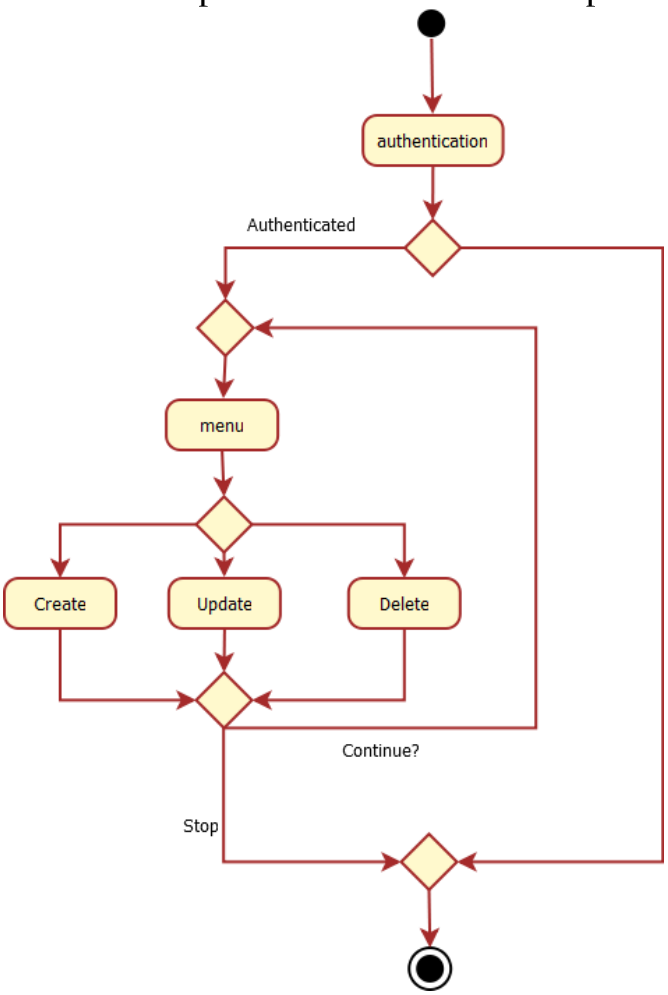
Identities on the other hand are stored in the Derby DB.

Satisfying the requirements by inclusion of following fields:

- UID – Unique User Id is auto-generated and set as a Primary Key in the Identity table.
- Identity Name: Identity name a string containing the name of the identity added.
- Email: Email is taken in as a string value.

Global application flow

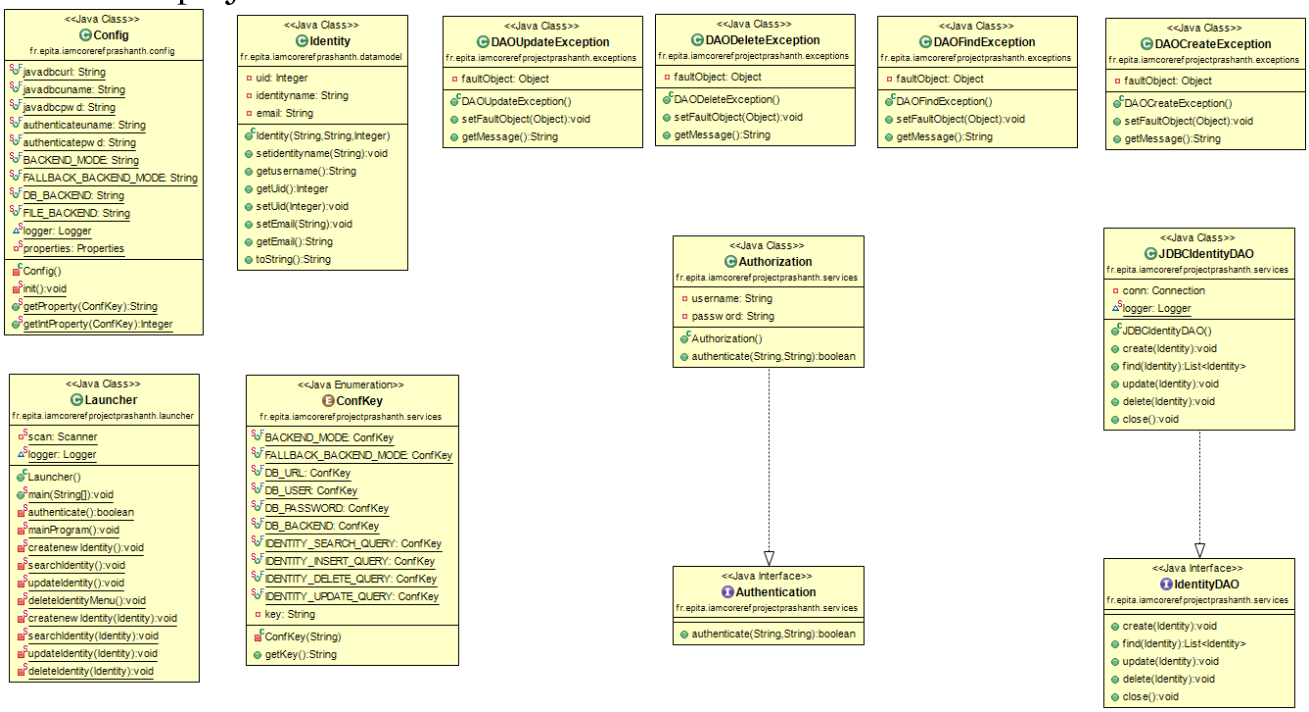
Global application flow is depicted below with the help of the flow chart.



The above depicted algorithm and flowchart is taken into account in creating the IMS module.

Global schema and major features schema

Below is the class schema generated with PlantUML. It clearly depicts the classes used in the project.

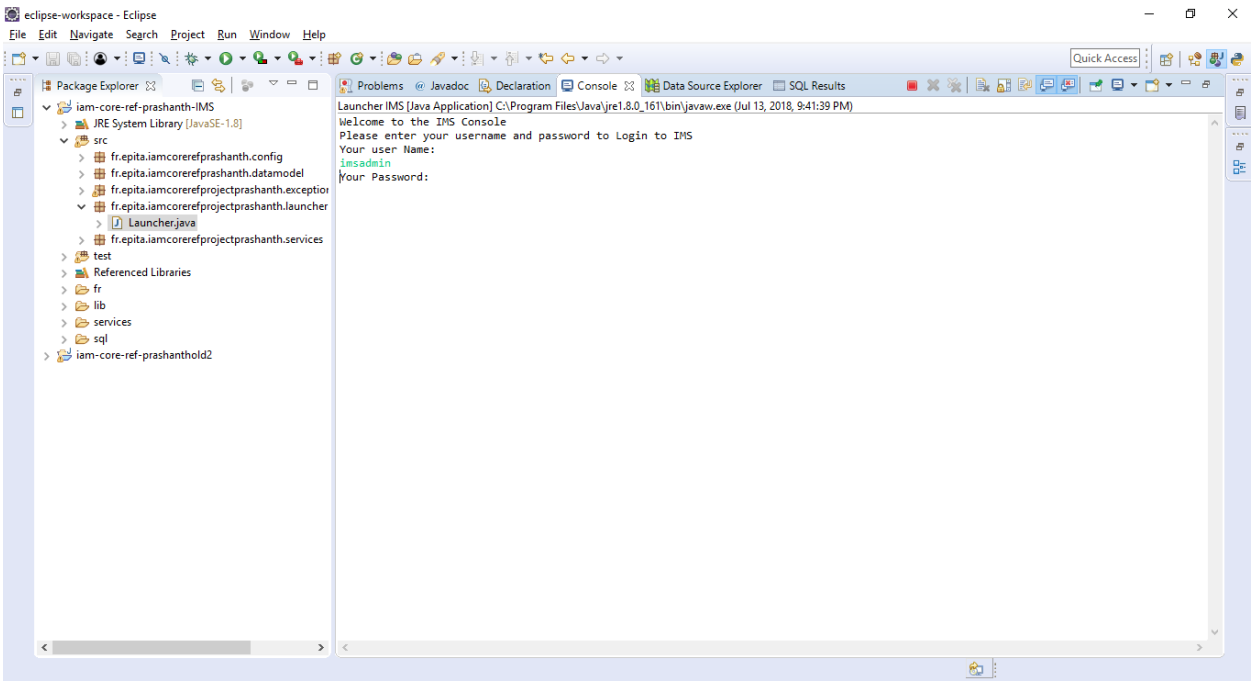


Console operations description

Please find the illustration of above operations with the screen shots to support the descriptive, below:

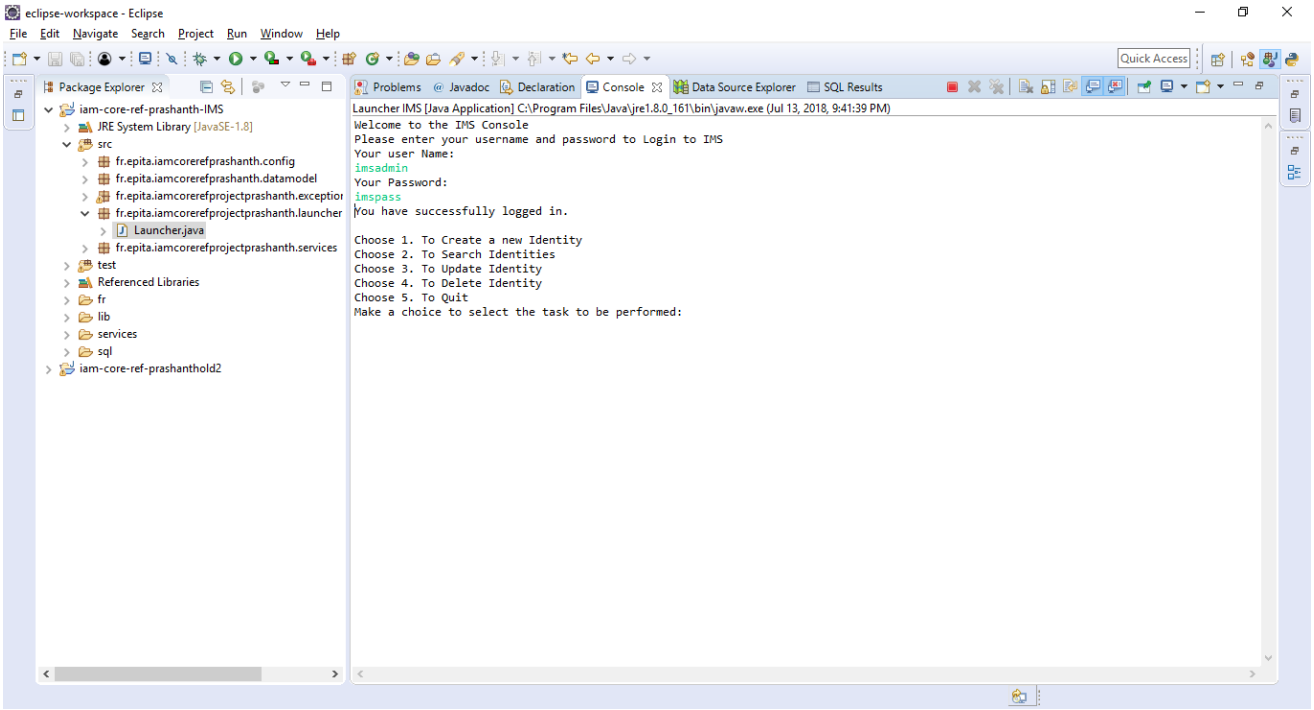
Authentication

The IMS console prompts the user for username and password



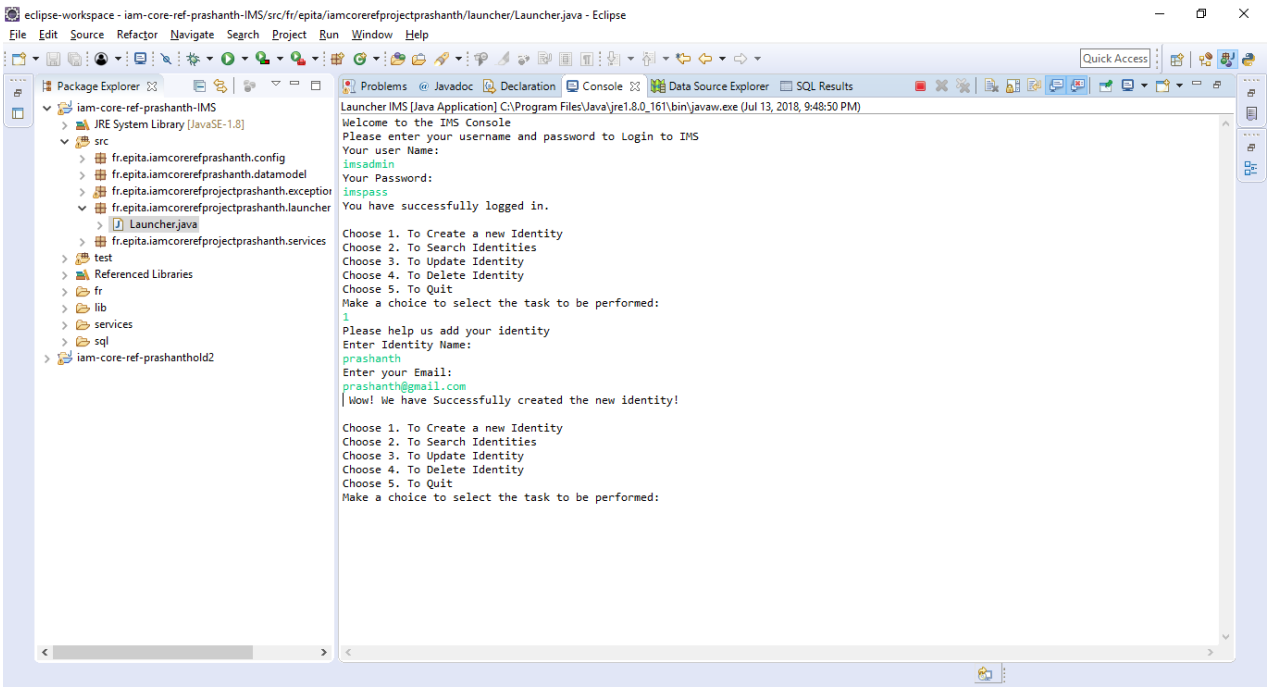
Menu

Once the authentication is successful, it displays the user menu and prompts the user to select from one of the menu options to perform the user intended operation.



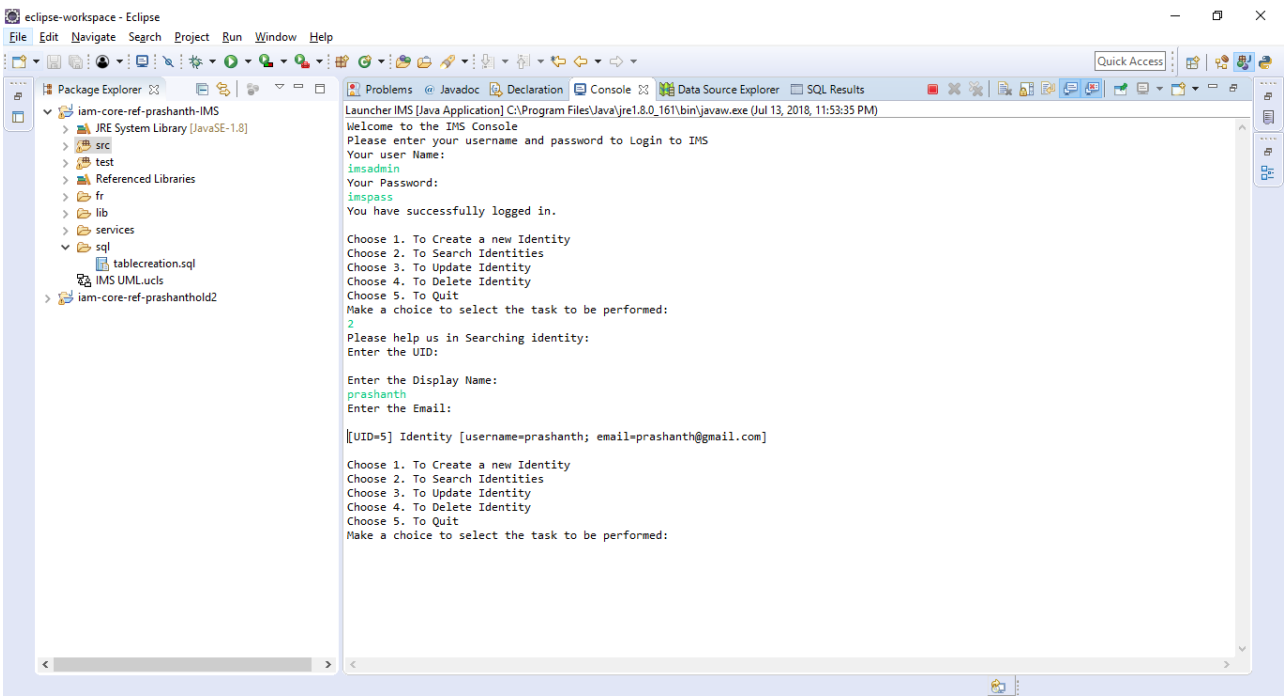
Option 1: To Create a new Identity

When the option 1 is selected Create identity method is called and prompted for identity name, and email id. If the entered values are right. Success message is printed to confirm.



Option 2: To Search Identities

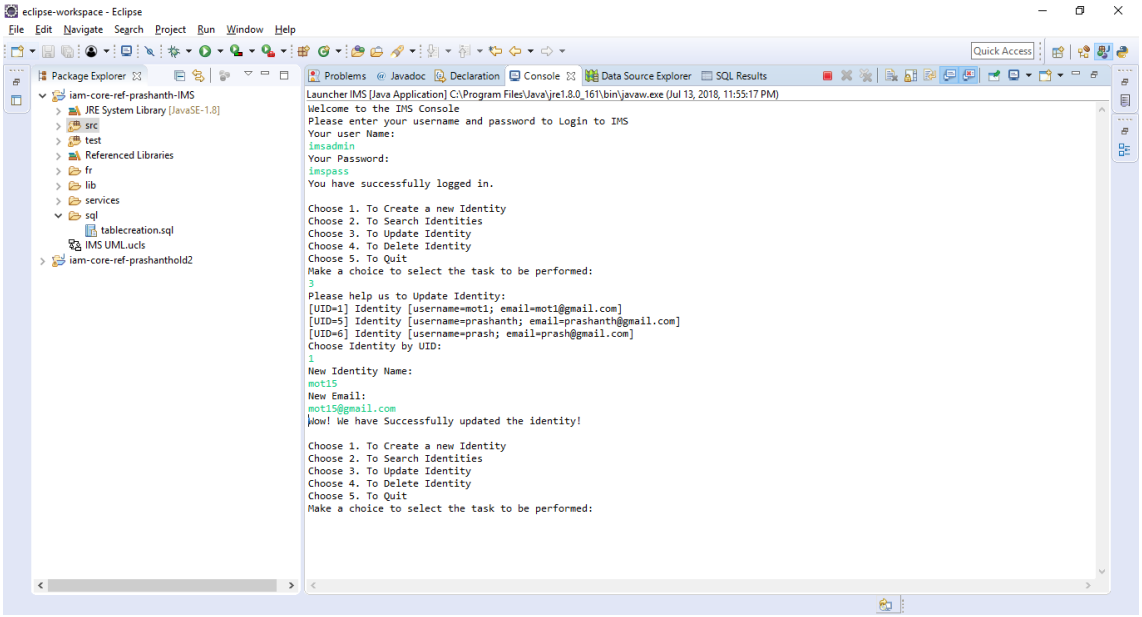
When the option 2 is selected Search identity method is called and prompted for UID, identity name, or email id. If a suitable match is found then it returns the mined value from DB.



Option 3: To Update Identity

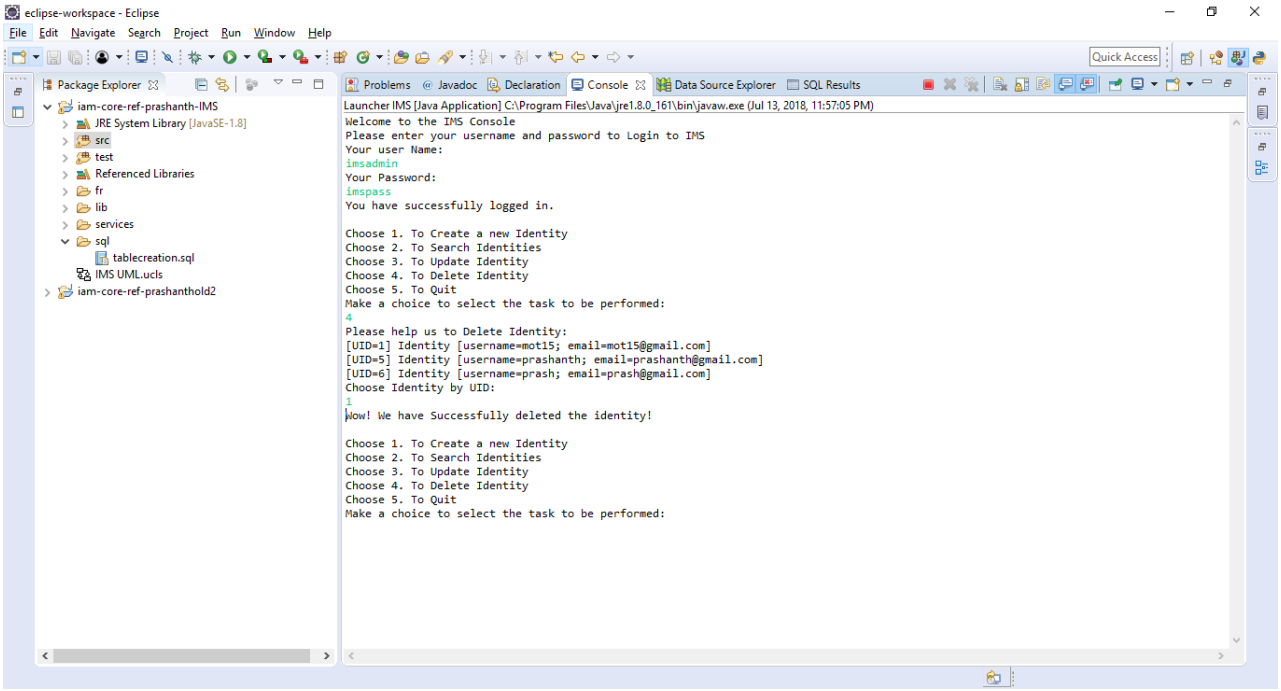
On selecting option 3, all the available identities in the DB is displayed to the user for him to make a selection and then gets the input value of UID, identity name or email

to pick the identity to be updated. If the Id introduced by the user is valid, then the identity is updated. else, the process is aborted.



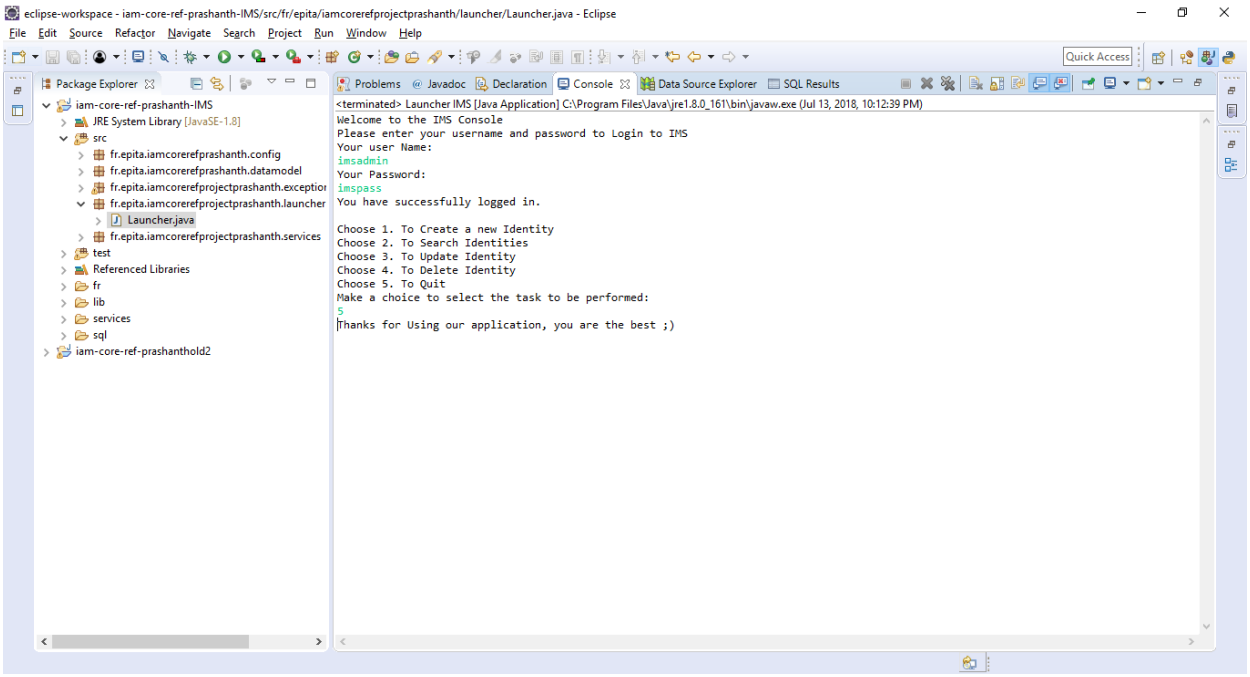
Option 4: To Delete Identity

On selecting option 4, all the available identities in the DB is displayed to the user for him to make a selection and then gets the input value of UID, identity name or email to pick the identity to be deleted. If the Id introduced by the user is valid, then the identity is deleted. else, the process is aborted.



Option 5: To Quit

When option 5 is selected, it terminates the program with a message.



End of Document