Data *Citation* GUI Document
All the classes locate in ***the edu.upenn.cis.citation.ui*** package.
The Database methods lie in the ***edu.upenn.cis.citation.dao*** package under the ***Database*** class,
The main class is called ***QBDApp***, which extends the ***Application*** class.
It has a bunch of attributes which are all different components of the UI.
There are two stages, one of which is the main stage called ***stage***, and the other stage corresponds to the citations is called ***citeStage***.

The main entrance is very simple as shown follows,

```
public static void main(String[] args) {
    launch(args);
}
```
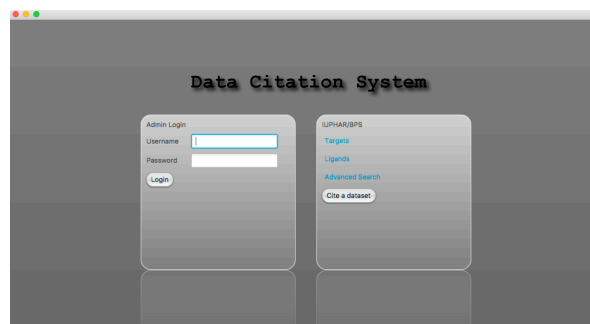
Then the override function public void start(Stage stage) { … } is called to launch the UI.
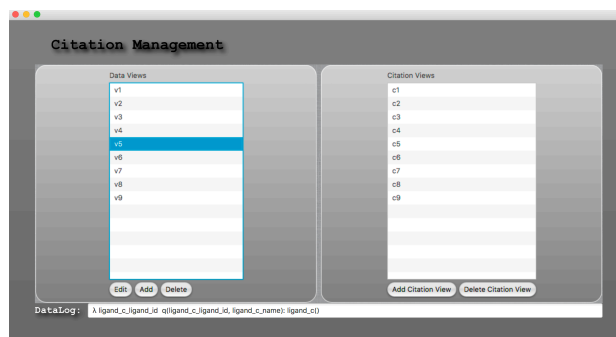The start method does the following things:
Initialize four scenes:
- Login Scene (buildLoginScene())
- QBE DBA Scene (buildQbeSceneDba())
- QBE User Scene (buildQbeSceneUser())
- QBE View Scene (buildViewScene())

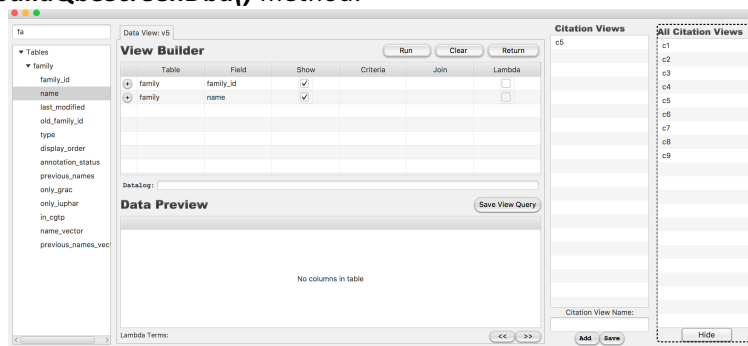Set the current scene to be the login scene as is shown follows.

You can enter username "admin" and password "admin" to log into the dba screen. The place to add authentication is the code ***btnLogin.setOnAction(event -> {}).*** For example, you can add OAuth into the application, which is a potential extension to the user interface.
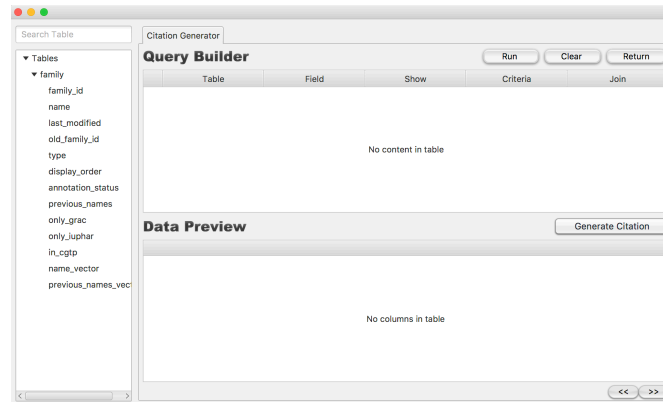

This screen displays the existing data views and citation views. The code for this part is all in the ***buildVIewScene()*** method. You can find all the UI components in the above screen there.

If user selects an data view and the corresponding Datalog is reconstructed and displayed on the screen. If Edit button is clicked it will takes the dba into a data view query and modification screen, which corresponds to the ***buildQbeScreenDba()*** method.

From the login screen, users can click on the "cite a dataset" button to get to the following citation builder screen.

The above screen function just like the query builder before in the dba screen, except for that
- The Datalog is invisible to the users
- Each row in the data preview has a single-row citation
- Users can select multiple rows in the data preview and click the "Generate Citation" button and get the corresponding Multi-row citation.

Note the method that is called to build the search by prefix functionality is implemented in the following methods. The buildSampleTreeView builds the search tree and adds corresponding listeners. The buildSampleTree() takes a samplesTreeView TreeView as the first parameter, the search text as the second parameter and a boolean flag indicating whether to rebuild the tree from the database. If the search text is null, it will display the full content, otherwise it will call the pruneSampleTree to prune the tree so as to narrow down the search content displayed in the left tree view.

Other classes:

The util.java class has some elementary implementation of Datalog conversion and the SQL conversion that is used by the UI to converted the QBE part into its corresponding Datalog/SQL.

The ***TreeNode*** class is the tree node element of the database table search box.

Each instance of ***Entry*** class corresponds a row in the query builder table.

The ***ComboBoxCell*** is used by the cellFactory in the user Data Preview part so that multiple data citation for each row can be displayed nicely in a comboBox (drop down bxo).

The ***style.css*** the styling that's been used by the user interface.