

C++ QT Project

Student 1629

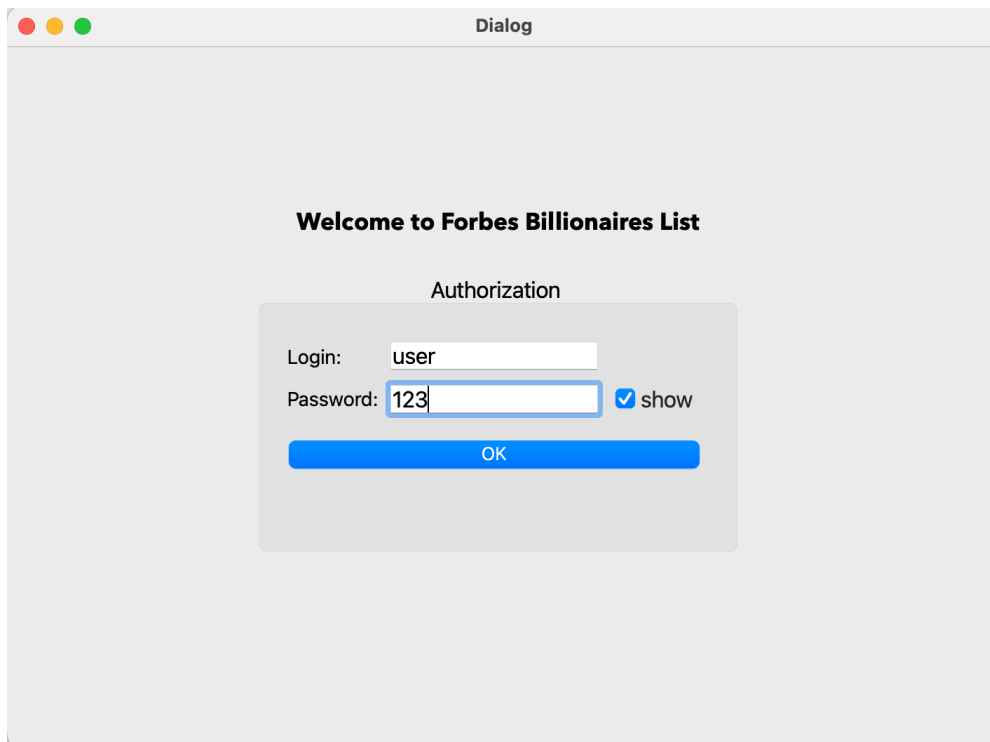
10.06.2023

Problem statement :

I was supposed to create an app that will be used to help Forbes correspondents track and correct the list of the wealthiest people on planet Earth. The app's features are a searching system and a capability to add, update and delete data.

Implementation details:

The app consists of two main windows. The first one(authorise.cpp) is an authorisation window, because only Forbes journalists can have access to this app. For conveniency I set the **login** and **password** to "user" and "123" correspondingly. One can choose to "show" password feature with checkbox implemented with the help of setEchoMode().



Second window is the main one.

- There is a Help button which provides access to "show logo" in a separate window which was made with the help of paintEvent() and working with jpg image.
- All the buttons are not available until a user loads a table. So there is a load button.
- There is a widget of search bar, which enables to find a person by name from the list which is case insensitive.
- There is a comboBox with sorting options: by age, wealth in different ways.
- Add, Edit, Delete, save buttons working in a simple way. When a person is added to a sorted table he is put in a place according to the current sorting.

ranking	name	networth	age	country	source	industry
1	Elon Musk	\$219 B	50	United States	Tesla, SpaceX	Automotive
2	Jeff Bezos	\$171 B	58	United States	Amazon	Technology
3	Bernard Arnau...	\$158 B	73	France	LVMH	Fashion & Retail
4	Bill Gates	\$129 B	66	United States	Microsoft	Technology
5	Warren Buffett	\$118 B	91	United States	Berkshire ...	Finance & ...
6	Larry Page	\$111 B	49	United States	Google	Technology
7	Sergey Brin	\$107 B	48	United States	Google	Technology
8	Larry Ellison	\$106 B	77	United States	software	Technology
9	Steve Ballmer	\$91.4 B	66	United States	Microsoft	Technology
10	Mukesh Ambani	\$90.7 B	64	India	Diversified	Diversified
11	Gautam Adani ...	\$90 B	59	India	Infrastructure, ...	Diversified
12	Michael ...	\$82 B	80	United States	Bloomberg LP	Media & ...
13	Carlos Slim He...	\$81.2 B	82	Mexico	Telecom	Telecom
14	Francoise ...	\$74.8 B	68	France	L'Oréal	Fashion & retail
15	Mark ...	\$67.3 B	37	United States	Facebook	Technology

The CSV data is parsed using the "csv_parser", which provides functions to read and parse CSV files. The `MainWindow` class is responsible for managing the main window of the application. The main window contains a table view that displays CSV data. The data for the table view is provided by the `CSVTable` class, which extends the `QAbstractTableModel`. It implements the necessary functions required by the model to represent the CSV data in a tabular format. These functions include `rowCount`, `columnCount`, `data`, `headerData`. The `CSVTable` class also provides additional functionality such as adding, editing, and deleting data, sorting, and saving the table.

The `DataDialog` class is responsible for creating a dialog window for adding, editing, or deleting data in the CSV table. It extends the `QDialog` class provided by Qt. The dialog contains input fields for the name, net worth, age, country, source, and industry. The input is validated before accepting the dialog, ensuring that all fields are filled match the pattern, e.g. net worth "\$<float> B".

Results and discussion

Overall, the implemented features of the project work, allowing users to view and interact with CSV data through the main window and perform operations such as adding, deleting, sorting, editing, searching, loading data.

Conclusion

In general, the app works. Potential improvements for the project could include enhancing the user interface, providing more advanced sorting options, search.