

Bank app report

Author: Serebryakova Anastasia, 211-1

Submission date: June 13th, 2022

Supervisor: George Piatsky

Problem statement

The main goal of the task is to create a Qt application for bank managers, to help them understand is their client “suitable” to approve a credit card or not. App should load dataset from CSV file. After downloading database, manager should be able to search for specific client’s data, add new client’s data, delete expired data and edit already existed data. After updating some information manager should be able to save file in CSV format.

Project specification

Here is the link: [spec](#)

Implementation details section

Link for repository: [here](#)

General info

In general, I tried to do all the features and requirements of my specification.

In my dataset there are no empty fields, so I thought that I need to put requirements for entering data(add and edit) – you cannot add new client’s data with empty fields. It is really important for bank’s managers to know all the information about their client. Based on this information they should choose – can they approve credit card or not.

Project structure

- ProjectQT
 - about.cpp - custom widget which draws logo and shows my student id
 - about.h
 - about.ui

- add.cpp - new QDialog window which allows to add new item or edit existed
- add.h
- incorrect.cpp - new QDialog window which shows that data wasn't applied correctly
- incorrect.h
- noshow.cpp - new QDialog window which shows that id from search does not exist
- noshow.h
- show.cpp - new QDialog window which shows clients data from searching
- show.h
- mainwindow.cpp - main window(+functionality)
- mainwindow.h
- main.cpp – starts programm
- ProjectQT.pro – builds up a project
- data
 - application_record.csv – dataset with all information
- bank_report.pdf –this report

Main window

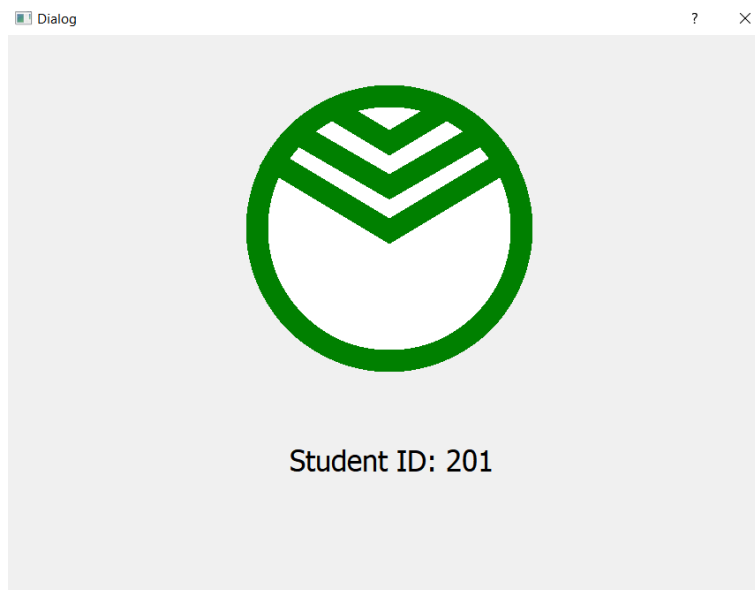
The screenshot shows the ProjectQT application window. At the top, there is a menu bar with 'File' and 'Logo'. Below it is a search bar with the text 'Search' and a dropdown menu set to 'Default'. The main area contains a table with the following columns: ID, CODE_GENDER, FLAG_OWN_CAR, FLAG_OWN_REALT, CNT_CHILDREN, VIT_INCOME_TOTAL, and AE_EDUCATION_I. The table lists 25 rows of client data. To the right of the table, there is a green handwritten label 'dataset' with a bracket pointing to the table. At the bottom of the window, there is a sidebar with three buttons: 'Add client' (labeled with a green '5'), 'Delete' (labeled with a green '6'), and 'Edit' (labeled with a green '7').

| ID | CODE_GENDER | FLAG_OWN_CAR | FLAG_OWN_REALT | CNT_CHILDREN | VIT_INCOME_TOTAL | AE_EDUCATION_I |
|---------|-------------|--------------|----------------|--------------|------------------|------------------|
| 5008804 | M | Y | Y | 0 | 427500 | Higher education |
| 5008805 | M | Y | Y | 0 | 427500 | Higher education |
| 5008806 | M | Y | Y | 0 | 112500 | Secondary / ... |
| 5008808 | F | N | Y | 0 | 270000 | Secondary / ... |
| 5008809 | F | N | Y | 0 | 270000 | Secondary / ... |
| 5008810 | F | N | Y | 0 | 270000 | Secondary / ... |
| 5008811 | F | N | Y | 0 | 270000 | Secondary / ... |
| 5008812 | F | N | Y | 0 | 283500 | Higher education |
| 5008813 | F | N | Y | 0 | 283500 | Higher education |
| 5008814 | F | N | Y | 0 | 283500 | Higher education |
| 5008815 | M | Y | Y | 0 | 270000 | Higher education |
| 5112956 | M | Y | Y | 0 | 270000 | Higher education |
| 6153651 | M | Y | Y | 0 | 270000 | Higher education |
| 5008819 | M | Y | Y | 0 | 135000 | Secondary / ... |
| 5008820 | M | Y | Y | 0 | 135000 | Secondary / ... |
| 5008821 | M | Y | Y | 0 | 135000 | Secondary / ... |
| 5008822 | M | Y | Y | 0 | 135000 | Secondary / ... |
| 5008823 | M | Y | Y | 0 | 135000 | Secondary / ... |
| 5008824 | M | Y | Y | 0 | 135000 | Secondary / ... |
| 5008825 | F | Y | N | 0 | 130500 | Incomplete |

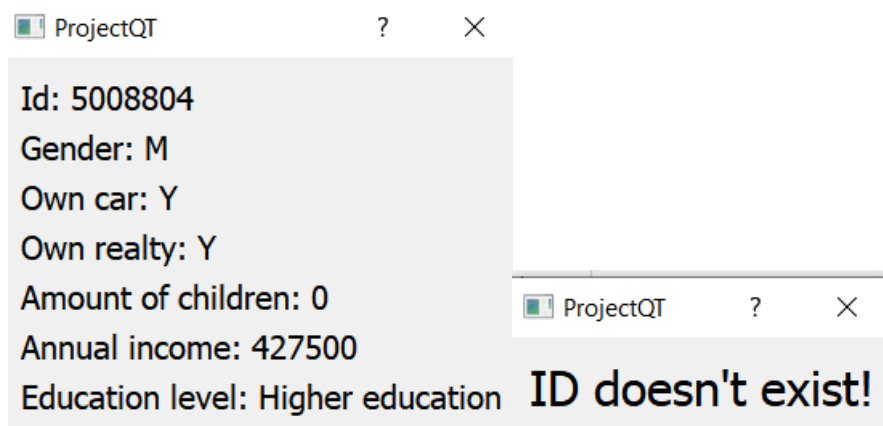
Navigation buttons:

- Add client 5
- Delete 6
- Edit 7

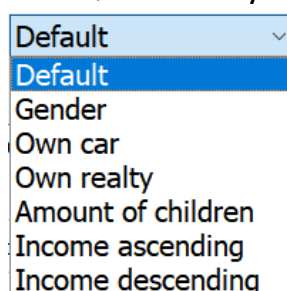
1. In file menu you have 2 options: you can open data in CSV file format; you can save your table as CSV file.
2. You can find logo of my bank logo in the menu bar “Logo” :



3. You can search for a specific client’s data by id. If such id doesn’t exist, new window will appear and say, that such id doesn’t exist. If it exists, there will appear another window with information about the client.

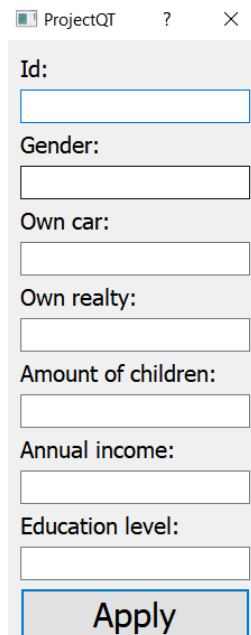


4. You can sort dataset in 6 different ways + default sort using QComboBox and QFilterProxyModel:



5. You can add new clients to your dataset using this button, after pushing it there will appear new window in which you enter data. Data is

checked for being valid, if it is not, there will appear window, which says, that data is not valid:



ProjectQT ? X

Id:

Gender:

Own car:

Own realty:

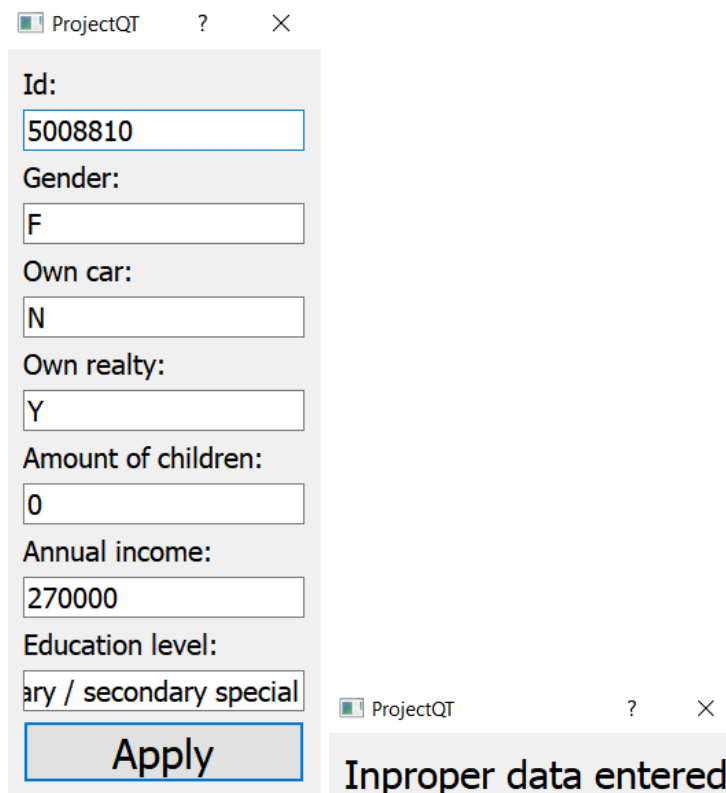
Amount of children:

Annual income:

Education level:

Apply

6. This QPushButton deletes selected rows.
7. This QPushButton edits information about specific client. Data is checked for being valid, if it is not, there will appear window, which says, that data is not valid:



ProjectQT ? X

Id:

Gender:

Own car:

Own realty:

Amount of children:

Annual income:

Education level:

Apply

ProjectQT ? X

Inproper data entered

Results and discussion

My specification had a huge amount of windows, so I made less windows(with permission of my supervisor) to make it more simple and comfortable to use. Search and delete were made in the same way as it was described in the specification, sorting has 3 more parameters(and has default) and add/edit have restrictions on data entry(you cannot enter empty fields). Now it is a little bit more useful than it was asked in spec.

Conclusion

I think the main goal of the project was achieved. I also added a couple more features to make it more usable for real bank managers. But there are still things, which can be improved as well.

For example it would be great to improve my app's design to make it look more familiar to manager's eyes. Also it would be interesting to make:

- Search by educational type is optional
- New column which shows that client "bad" or "good" depending on his other parameters instead of thinking, manager will see conclusion about client already made by a computer