SQL financial data analysis

Final project for the SQL module on the data analysis course at Coders Lab

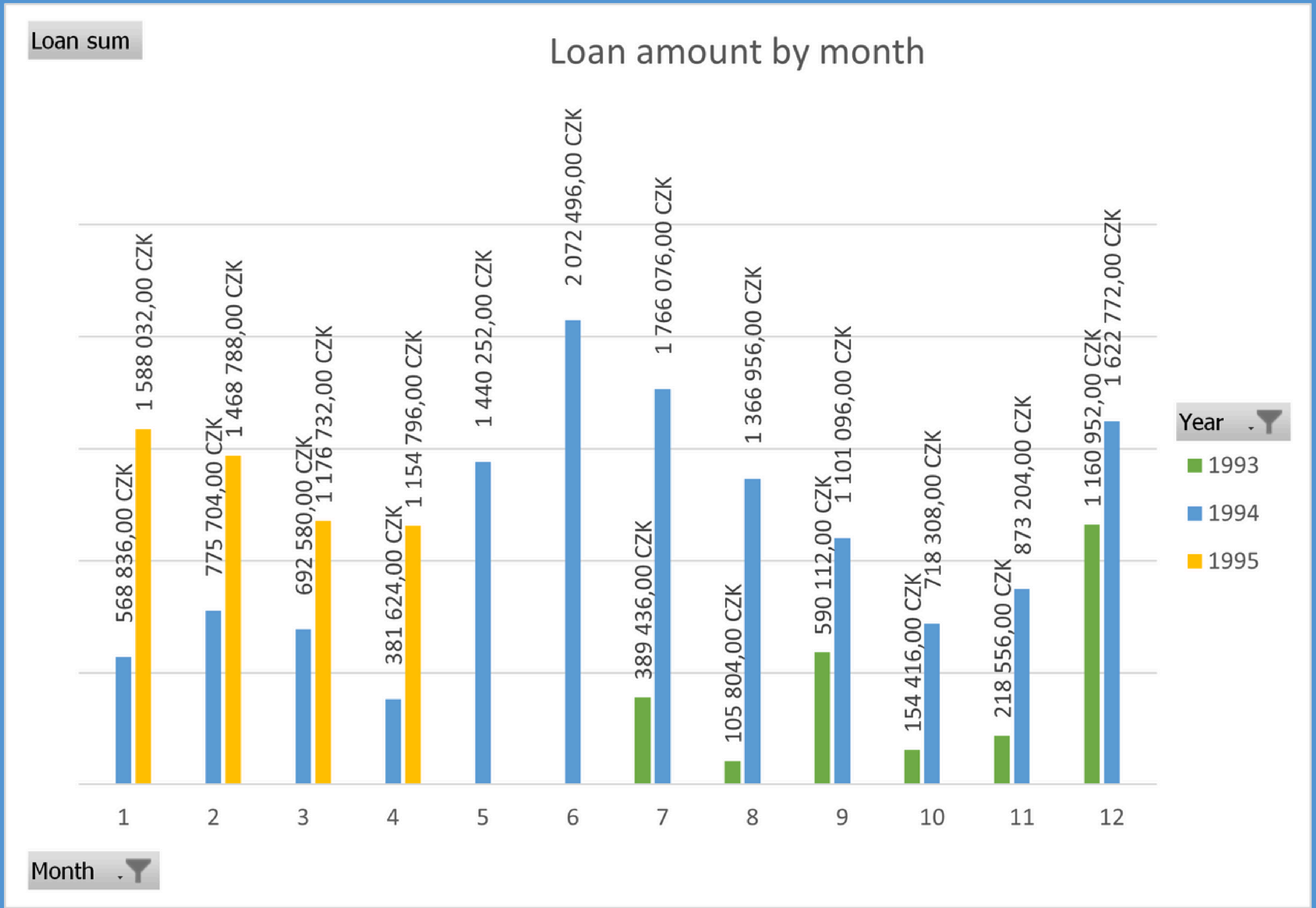
Project summary

The goal of this project was to simulate financial data analysis using SQL queries. The analysis was performed on the Berka dataset, which contains anonymized information from a Czech bank. This dataset includes over 5,300 clients and around 1,000,000 transactions. Additionally, it represents nearly 700 loans and approximately 900 credit cards.

Key insights were extracted from the database using SQL queries of varying complexity, focusing on loans, transactions, and client information. A procedure for automating the reporting of upcoming credit card expirations was also created. The results of the initial analysis can serve as a foundation for further investigations, such as answering specific business questions and improving financial management.

The project utilized tools like MySQL, Excel, and GitHub.

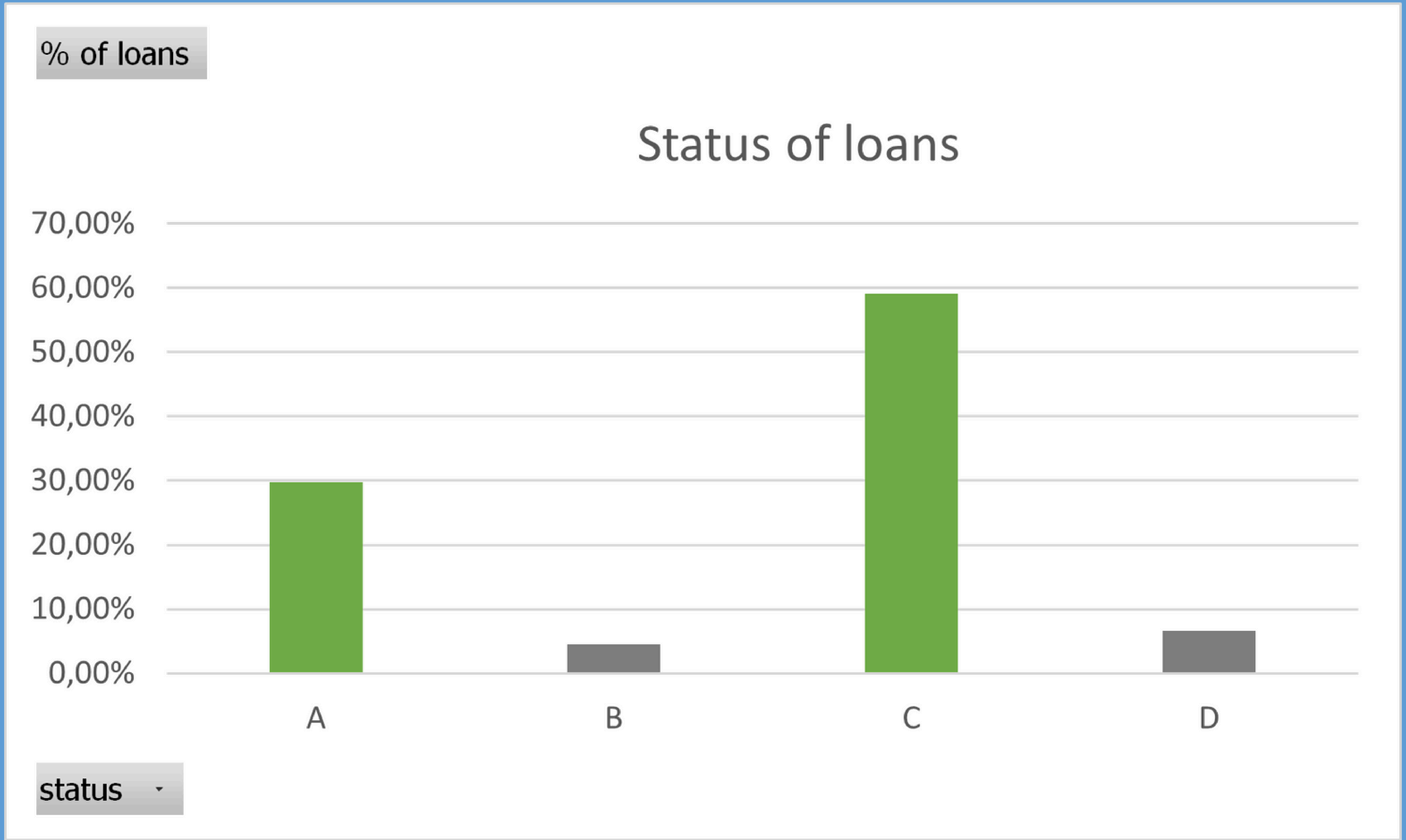
The main objective of the first task was to create an SQL query that would allow the historical analysis of loans granted in different time dimensions: annual, quarterly and monthly. Aggregation and grouping of the data was used, with the ROLLUP function immediately creating overall, annual and quarterly summaries.



Analysis report.xlsx / Question_1_charts

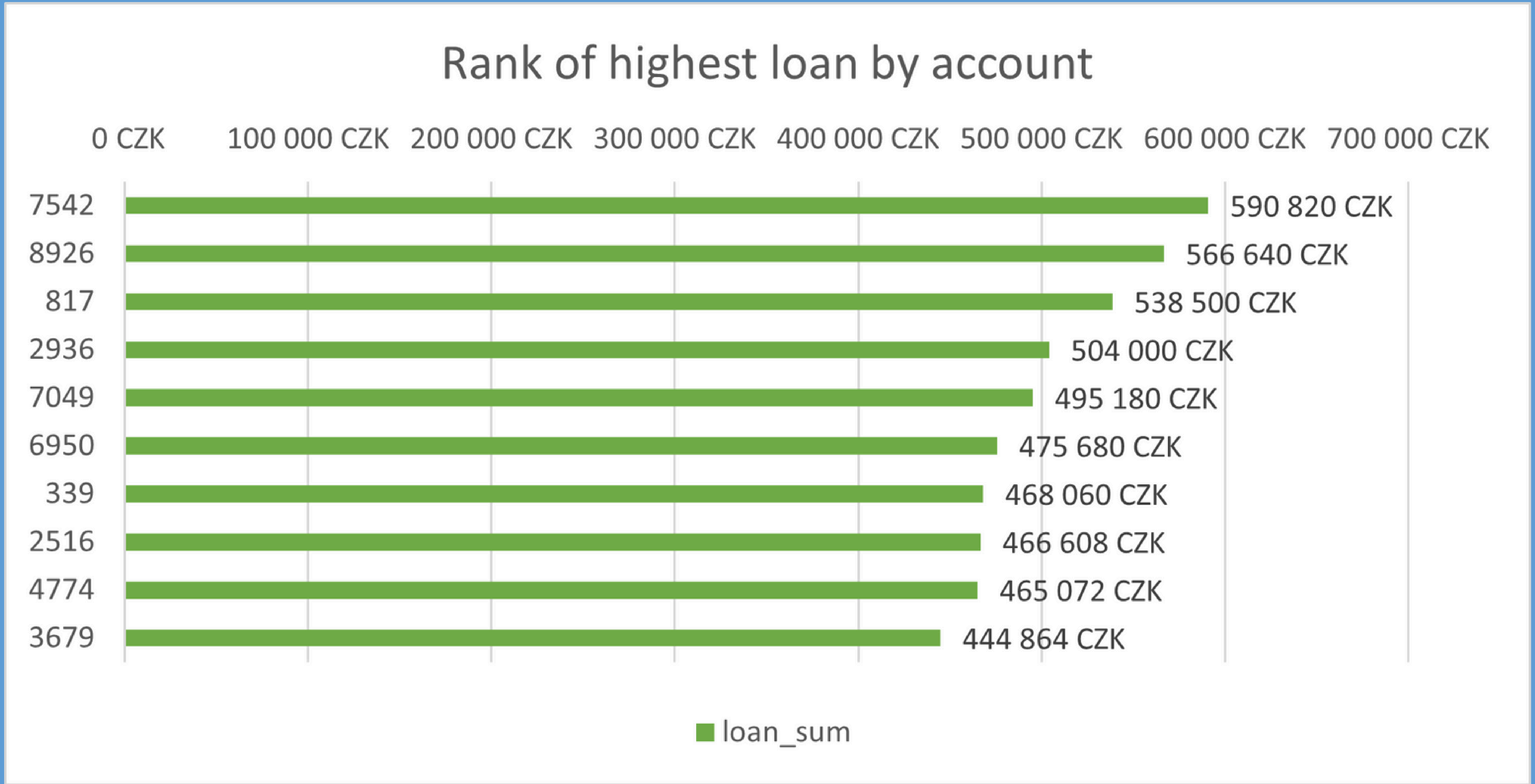
Excel was used to visualise the results of the query in the form of a column chart of the sum of loans granted in each month.

The main objective of the second task was to identify loan statuses that indicate repaid and unrepaid loans. The query grouped data by status and counted the number of loans for each status. Based on the results and given information, it was concluded that statuses "A" and "C" correspond to repaid loans, while statuses "B" and "D" correspond to unrepaid loans.



Excel was used to visualise the results of the query in the form of a column chart of the sum of loans divided by status.

Analysis report.xlsx / Question_2

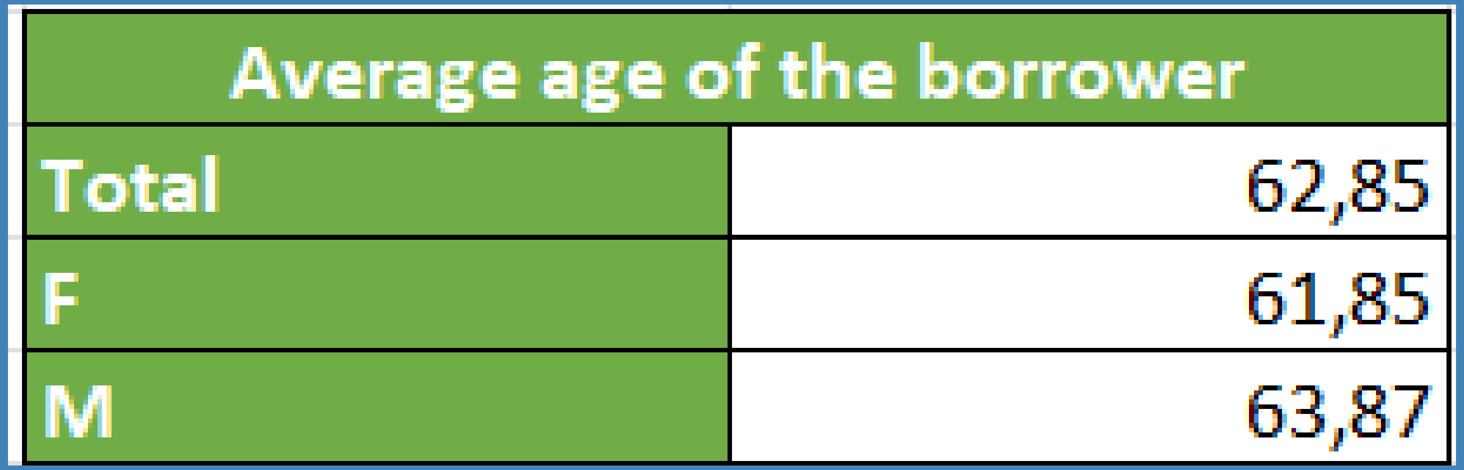


Analysis report.xlsx / Question_3

The main objective of the third task was to rank accounts based on their loan activity, considering only repaid loans. The query used a Common Table Expression (CTE) to calculate loan counts, sums, and averages for each account. Then, the RANK() function was applied to rank accounts by loan sum and loan count.

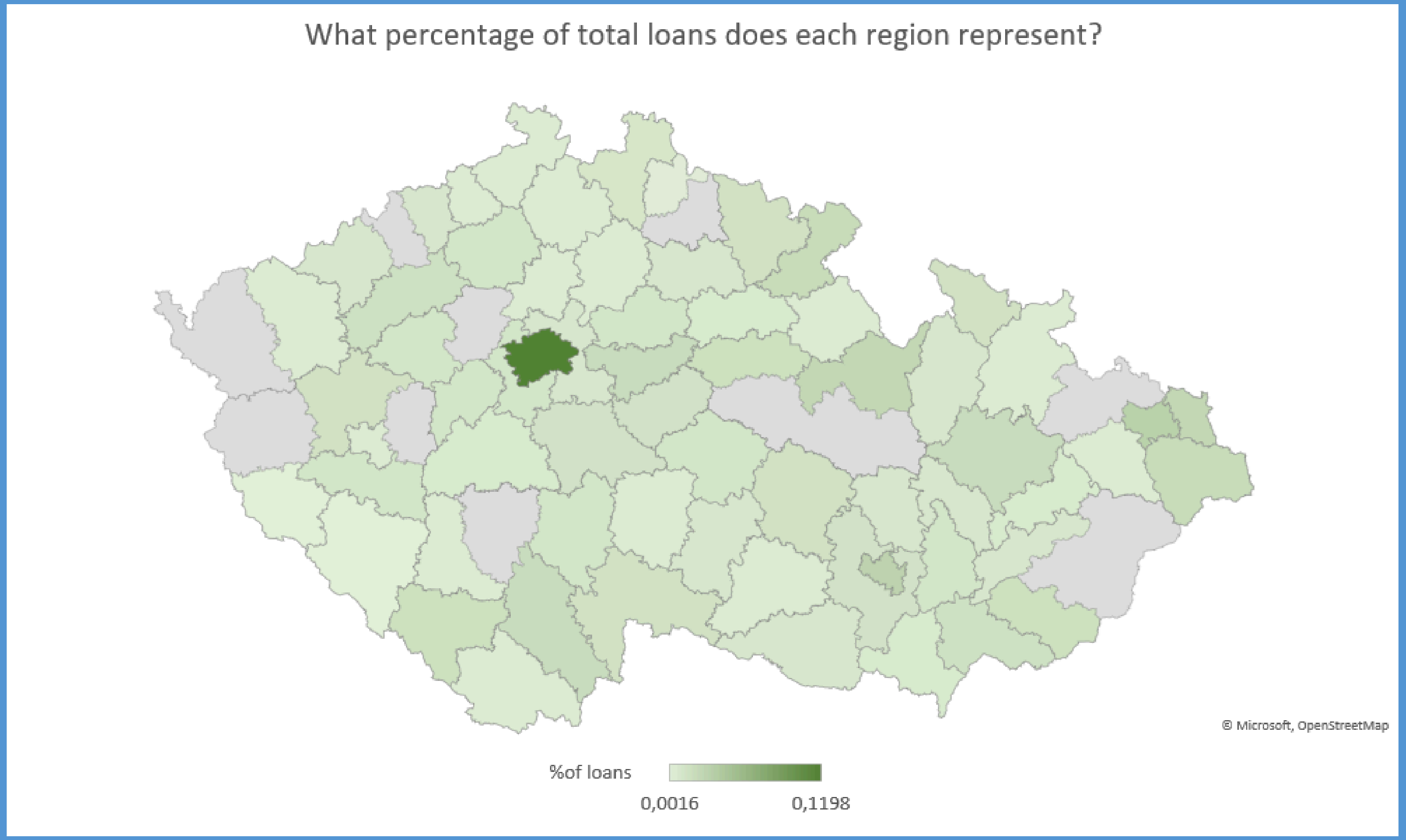
Excel was used to show the 10 accounts with the highest loans sorted from largest to smallest loan.

The fourth and fifth tasks analyzed customer demographics based on repaid loans. Task four calculated the total balance of repaid loans by gender. Task five expanded the analysis to include borrower age, allowing for a comparison of loan repayment behavior between men and women and the calculation of average borrower age. This combined approach provided a deeper understanding of customer loan repayment behavior, enabling comparisons between men and women in terms of both loan amounts and the average age of borrowers.



Average age of the borrower	
Total	62,85
F	61,85
M	63,87

The data derived from these queries can be further refined and analyzed. For instance, grouping customers into age groups can help identify patterns within specific demographics. The accompanying 'Analysis report.xlsx' file, specifically the 'Question_4' sheet, demonstrates how a pivot table can effectively visualize such trends and insights.


What percentage of total loans does each region represent?
© Microsoft, OpenStreetMap
%of loans
0,0016
0,1198

Analysis report.xlsx / Question_6

Building upon the previous analysis, tasks 6 and 7 focused on regional differences in loan repayment behavior. SQL queries were employed to join multiple tables and calculate metrics such as the number of clients, total loan amount, and loan count for each district. The results highlighted regions with the highest customer concentration and the largest total loan value. By using window functions like SUM() OVER(), the analysis also determined the percentage contribution of each region to the overall loan portfolio, providing valuable insights for resource allocation and strategic planning.

The use of an Excel map to visualise data on repaid loans makes it much easier to interpret the results of the analysis and allows practical conclusions to be drawn for the business.

Tasks eight and nine focused on detailed analysis of customer profiles, with particular emphasis on credit activity and demographics.

In Task eight, an SQL query was created to select customers who met certain criteria: a high account balance, a high number of loans and a young age. Analysis of the results showed that there were no customers who met all three criteria at the same time.

Task nine went one step further and conducted a more detailed analysis to determine which of the conditions was the most restrictive. It turned out that the age and number of loans conditions were too restrictive and excluded too large a proportion of potential customers.

Results from this part of the analysis underline the importance of carefully considering and reviewing criteria. Restrictive criteria may lead to poor results or select an inappropriate group of customers.

The last task required the creation of a procedure to automatically generate a report on credit cards approaching their expiry date. The procedure uses SQL queries to combine data from different tables, calculate the expiry date and filter the results according to a specific criterion.

The benefits of such a procedure are:

- Automation of the process: The procedure allows reports to be generated automatically, eliminating the need for manual queries.
- Up-to-date data: The report always contains the most recent credit card information.
- Improved customer service: The process allows you to proactively inform customers of impending card expiry dates, which can improve customer satisfaction.

Project repository

https://github.com/ReviIsCoding/CodersLab_Subprojects/tree/main/SQL_module_final_project