

Lab Session 5: SQL Queries

In this lab, we will use our simple bank database as an example to practice different kinds of SQL queries.

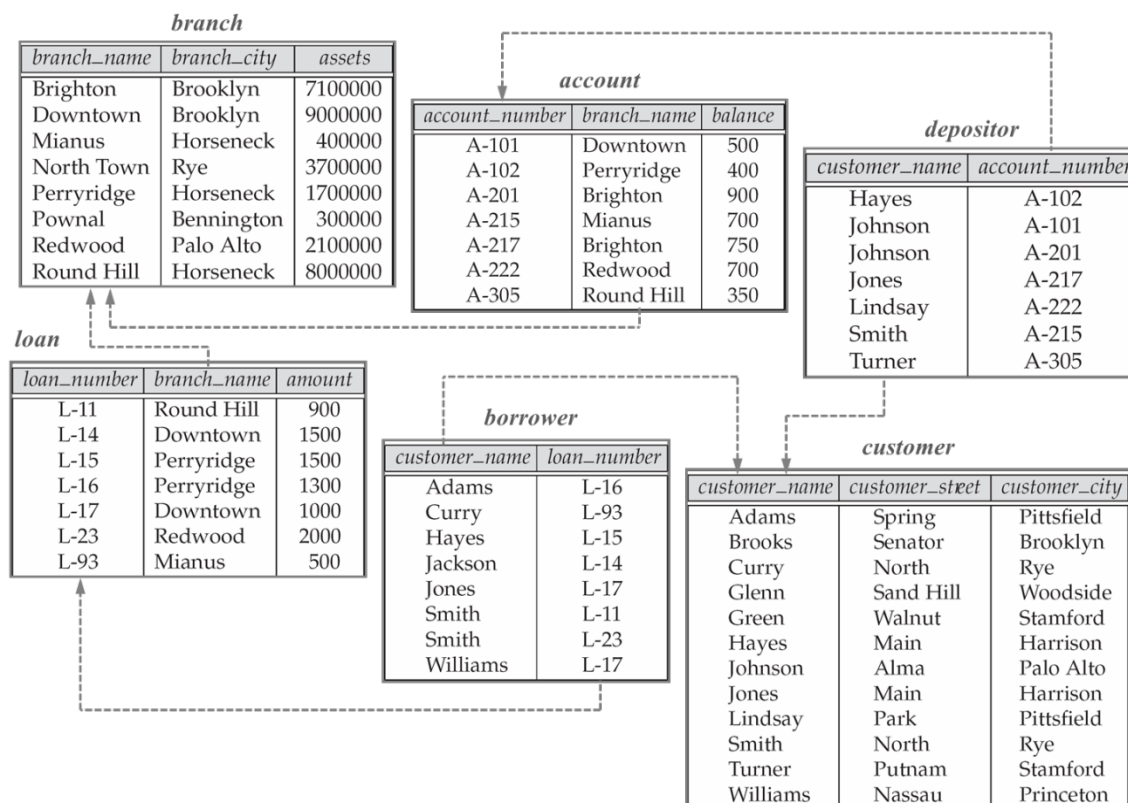


Figure 1. Bank database

As usual, this lab assumes that this database has already been created on the server (db.tecnico.ulisboa.pt), and can be accessed with a tool such as **mysql command**.

Querying the database

Write an SQL query to answer each of the following questions:

1. How many customers live in the same city of a branch where they have a loan?
2. Find the average balance of the accounts of each customer.

3. Find the average balance of the accounts of each customer, but only for branches in the city of Horseneck.
4. Considering the branches in the city of Horseneck, determine the total amount of money in the accounts of those branches.
5. For each city, determine the total amount of money in the accounts of branches in that city. (In the results, indicate the city and the total amount of money)
6. Show, in alphabetical order, the names of branches that have 2 or more loans.
7. Same question as before, but show also the sum of the loans for each branch.
8. Someone has noticed that there are some branches without any account. Using an outer join, find those branches. In the result, show the name and city of those branches which have no accounts.
9. Someone has noticed that there are some customers without any account. Using the **NOT IN** operator, find the customers who have no accounts.
10. Consider the set of branches which have either accounts or loans (or both). Find which branches are not in that set. Use **UNION** and **NOT IN** in the query.
11. Now consider the set of branches which have both accounts and loans. To find which branches are not in that set, how would you change the previous query? Try it.
12. Consider the set of cities where the bank has branches. Now find the set of customers who live in those cities. Use the **IN** operator.
13. What is the loan with the highest amount? In the result, show only the number and the amount for that loan. Use the **ALL** operator.
14. Find the total amount of money that each customer owes to the bank (i.e. considering all the loans of that customer). In the result, show the customer name and the total amount of money for each customer name.
15. Now find the customer who owes more money to the bank (considering the all loans of that customer). Show only the name of that customer. Use the **ALL** operator.