

## SQL Queries

---

Consider the Bank Database.

### Bank Database

*branch* (*branch-name*, *branch-city*, *assets*)

*customer* (*customer-name*, *customer-street*, *customer-city*)

*account* (*account-number*, *branch-name*, *balance*)

*loan* (*loan-number*, *branch-name*, *amount*)

*depositor* (*customer-name*, *account-number*)

*borrower* (*customer-name*, *loan-number*)

*employee* (*employee-name*, *branch-name*, *salary*)

## **Do the following:**

1. Create the Bank database. Bank database schema is available on the blackboard.
2. Populate the Bank Database, using the data records available on the blackboard.
3. Do the following queries:

For each query:

Write the question

Write the SQL statement

Provide the output. For update queries (insert, delete, replace), display the table(s) before the query and after the query.

## **Retrieval Queries**

1. Find all loan number for loans made at the Perryridge branch with loan amounts greater than \$1100.
2. Find the loan number of those loans with loan amounts between \$1,000 and \$1,500 (that is,  $\geq \$1,000$  and  $\leq \$1,500$ )
3. Find the names of all branches that have greater assets than some branch located in Brooklyn.
4. Find the customer names and their loan numbers for all customers having a loan at some branch.
5. Find all customers who have a loan, an account, or both:
6. Find all customers who have an account but no loan.  
(no minus operator provided in mysql)
7. Find the number of depositors for each branch.
8. Find the names of all branches where the average account balance is more than \$500.
9. Find all customers who have both an account and a loan at the bank.
10. Find all customers who have a loan at the bank but do not have an account at the bank
11. Find the names of all branches that have greater assets than all branches located in Horseneck. (using both non-nested and nested select statement)
12. 1 query of your choice involving aggregate functions
13. 1 query of your choice involving group by feature.

## **Insert Queries**

Do 2 insert queries requiring multiple records insertion as follow:

1. Create a HighLoan table with loan amount  $\geq 1500$ .
2. Create a HighSalaryEmployee table with employee having salary more than 2000.
3. 1 more query (meaningful) of your choice on any table.

## **Update Queries**

1. Increase all accounts with balances over \$800 by 7%, all other accounts receive 8%.
2. Do 2 update queries, each involving 2 tables.
3. 1 more update query of your choice on any table.

## **Delete Queries**

1. Delete the record of all accounts with balances below the average at the bank.
2. Do 2 update queries, each involving 2 tables.
3. 1 more delete query of your choice from any table.

## **Views Queries**

1. A view consisting of branches and their customers
2. Create a view of HQEmployee who work in downtown branch.
3. Do one insert, delete, update, and select queries on HQEmployee view.

## **Complex Queries: provide results**

1. 1 select query involving 3 tables
2. 1 Delete query involving 3 tables
3. 1 Update query involving 3 tables

**Submit your Homework 2 as a PDF file as**

**YourName.pdf**

**Including. All your questions, SQL statements, and results.**