Warm-up Project

Group ID: td_comp353_2

Yassin Bah 40077524

Joel Dusablon Senecal 40035704

Feng Zhao 40021856

Alireza Sari 40032394

September 29, 2018

1 Database Design

1.1 Assumptions

In developping and designing this database, certain assumptions have been made. The goal of this section is to list them in order to help clarify why the database is created the way it is.

1.1.1 Branch

A branch has a unique ID that allows for the distinction of if multiple branches are in the same area. A branch needs to have one manager at all times. In relational terms, it means the attribute cannont be null in the Branch table. All the branches of the bank, including the head office, should be in this table. The head office is denoted by a flag that is set to 1 for the record of the head office. Furthermore, the manager of the head office represents the president of the bank. Following this assumption, the president of the bank is also an employee and has an equivalent entry in the Employee table.

1.1.2 Employee

Every employee has a unique identifier. Basic assumptions about employees are that each record must have their first name, last name, starting date and a branch ID that cannot be null at the onset. Employee only works at one branch and that branch has to be open. That means that the branch ID for in the Employee table cannot be null at any time. All service general managers work at the head office. In order to know the general manager of each service, the Service table needs to be looked up.

1.1.3 Client

Clients also have unique identifier. Like employees, clients have a first name, a last name and a branch attributes that cannot be null. A client needs to be associated with one branch at all times.

1.1.4 Account

Accounts belong to a specific client and may not be shared. Clients, however, may have multiple accounts linked to them. An account has to be associated with a current client of the bank. An account can only have one option associated with it.Credit limits are associated with accounts rather than clients. Reason being is that a client may have a business account and a personal account, but the credit limit for either account might be different. A similar situation occurs with the interest rate, they can vary depending on multiple factors and may hold a range of values.

1.1.5 Services

As stated previously, the services contain a list of services that the bank offers as well as the ID of the general manager for said service. All general managers are thus also considered employees.

1.1.6 Charge Plan

2 Schema

3 Queries

(a) All of the tables

(b) List of all the branches grouped by city and ordered by oldest branch.

(c) List of all clients with DOB between 1990 and 2017.

(d) List all clients of a branch who has either a checking or savings account of balance more than CND 10,000.00.

(e) List of all clients of a branch who has a line of credit of limit CND 25,000.00 with an interest rate of 7.5% or below.

```
/*
 * creation_commands.sql
 * Copyright (C) 2018 Sebastien Bah <sebastien.bah@gmail.com>
 *
 * Distributed under terms of the MIT license.
 */


/* Branch */
CREATE TABLE Branch (
        branch_id       int not null auto_increment,
        province        varchar(255) not null,
        city            varchar(255) not null,
        street          varchar(255) not null,
        phone           varchar(255),
        fax             varchar(255),
        opening_date    date not null,
        manager_id      int not null,
        isHeadOffice    tinyint(1),
        primary key(branch_id),
```

(f) List details of a client named Roberto.

(g) List of all clients of 'Cote Des Neiges' branch.

(h) List of clients who have at least 1,000,000 CDN dollar in their savings account.

(i) List of all the services along with the general manager for each service.

(j) Complete details of the president of the bank.

```
        foreign key(manager_id) references Employee(title_id)
);


/* Employee */
CREATE TABLE Employee (
        employee_id     int not null auto_increment,
        firstName       varchar(255) not null,
        lastName        varchar(255) not null,
        addr            varchar(255),
        start_date      date not null,
        salary          decimal(14,2),
        email           varchar(255),
        phone           varchar(255),
        branch_id       int not null,
        -- position_id      int not null,
        foreign key(branch_id) references Branch(branch_id),
        -- foreign key(position_id) references Positions(position_id),
        primary key(employee_id)
);
```