

# **A Database Application System for Employees of the Health Facilities**

Main Project

By:

Alexandre-Louis Trinh - 40173273

Wen Jie Ng - 40292061

Eliza Bulatao - 40252258

Hui Zhang - 40170679

Group Account: wjc353\_4

Course: COMP 353 - Databases

Due Date: April 15th, 2024

Concordia University

“We certify that this submission is the original work of members of the group and meets the Faculty's Expectations of Originality”

## **Faculty of Engineering and Computer Science** **Expectations of Originality**

This form sets out the requirements for originality for work submitted by students in the Faculty of Engineering and Computer Science. Submissions such as assignments, lab reports, project reports, computer programs and take-home exams must conform to the requirements stated on this form and to the Academic Code of Conduct. The course outline may stipulate additional requirements for the course.

1. Your submissions must be your own original work. Group submissions must be the original work of the students in the group.
2. Direct quotations must not exceed 5% of the content of a report, must be enclosed in quotation marks, and must be attributed to the source by a numerical reference citation<sup>1</sup>. Note that engineering reports rarely contain direct quotations.
3. Material paraphrased or taken from a source must be attributed to the source by a numerical reference citation.
4. Text that is inserted from a web site must be enclosed in quotation marks and attributed to the web site by numerical reference citation.
5. Drawings, diagrams, photos, maps or other visual material taken from a source must be attributed to that source by a numerical reference citation.
6. No part of any assignment, lab report or project report submitted for this course can be submitted for any other course.
7. In preparing your submissions, the work of other past or present students cannot be consulted, used, copied, paraphrased or relied upon in any manner whatsoever.
8. Your submissions must consist entirely of your own or your group's ideas, observations, calculations, information and conclusions, except for statements attributed to sources by numerical citation.
9. Your submissions cannot be edited or revised by any other student.
10. For lab reports, the data must be obtained from your own or your lab group's experimental work.
11. For software, the code must be composed by you or by the group submitting the work, except for code that is attributed to its sources by numerical reference.

You must write one of the following statements on each piece of work that you submit:

For individual work: **"I certify that this submission is my original work and meets the Faculty's Expectations of Originality"**, with your signature, I.D. #, and the date.

For group work: **"We certify that this submission is the original work of members of the group and meets the Faculty's Expectations of Originality"**, with the signatures and I.D. #'s of all the team members and the date.

A signed copy of this form must be submitted to the instructor at the beginning of the semester in each course.

I certify that I have read the requirements set out on this form, and that I am aware of these requirements. I certify that all the work I will submit for this course will comply with these requirements and with additional requirements stated in the course outline.

Course Number: COMP 353

Instructor:

Omid Reza Heidari

Name: Alexandre-Louis Trinh, Wen Jie Ng, Elias Balchao, Hui Zhang

I.D. #

40143273, 40292061, 40252258, 40170679

Signature: Alexandre-Louis Trinh, Wen Jie Ng, Elias Balchao, Hui Zhang

Date:

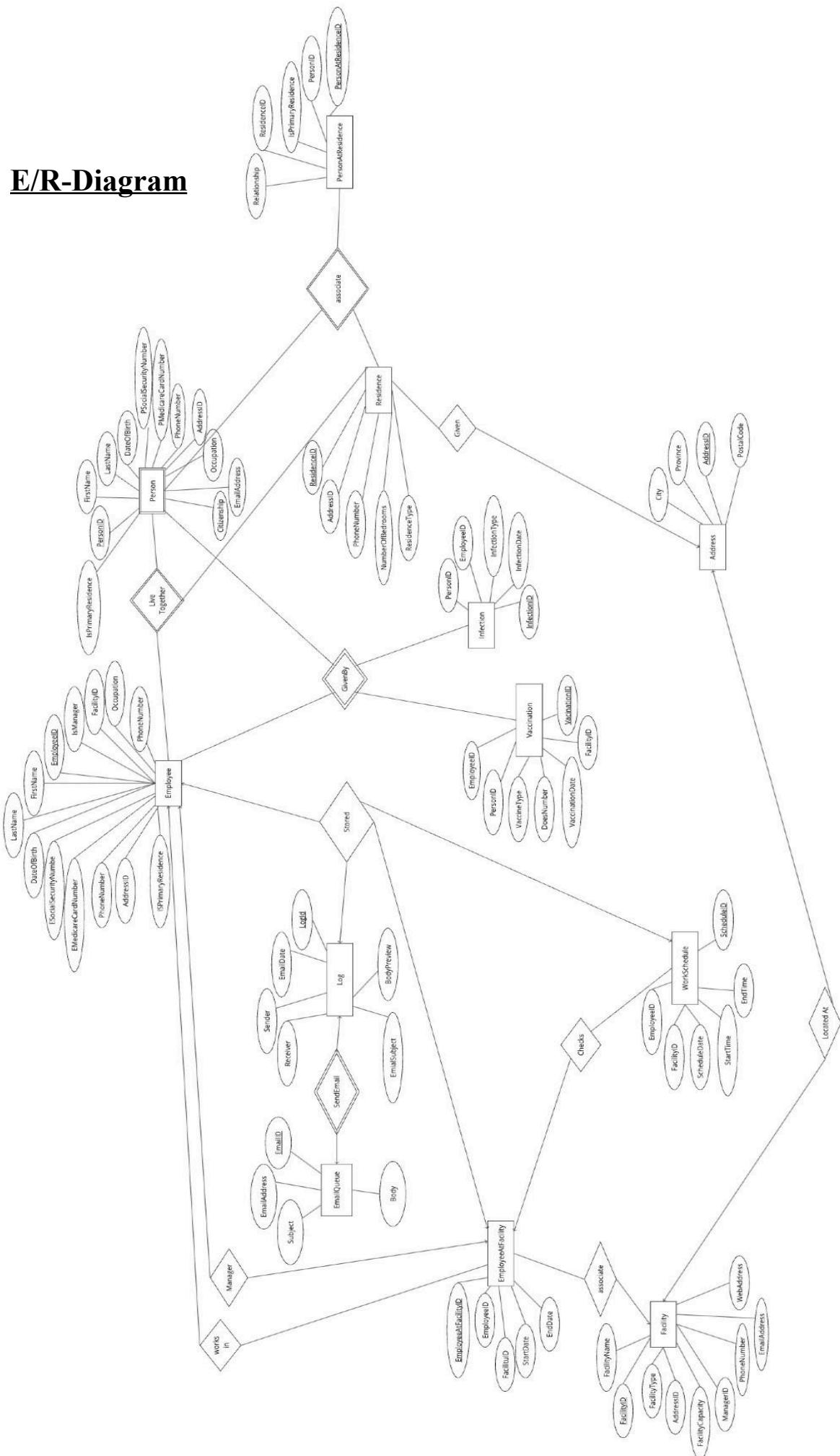
Apr 15 2012

<sup>1</sup> Rules for reference citation can be found in "Form and Style" by Patrick MacDonagh and Jack Bordan, fourth edition, May, 2000, available at <http://www.enca.concordia.ca/scs/Forms/Form&Style.pdf>.

## **Reasonable Assumptions**

For the design of our database, we assumed that Employees and Persons are non-overlapping entities. This means that information on employees of healthcare facilities are stored in the Employee table, whereas the information on the people living in the same residences of these employees are stored in the Person table. As such, we do not have Employee table inheriting the attributes from the Person table.

## E/R-Diagram



## E/R to Relation Conversion, Functional Dependencies

### 1. Address Table:

- Address(AddressID, City, Province, PostalCode)
- Functional Dependencies:
  - AddressID → {City, Province, PostalCode}
- Primary key: AddressID

```
CREATE TABLE Address (
    AddressID INT PRIMARY KEY AUTO_INCREMENT,
    City VARCHAR(255) NOT NULL,
    Province VARCHAR(255) NOT NULL,
    PostalCode VARCHAR(10) NOT NULL UNIQUE
);
```

- Dummy data inserted

```
-- Insert Dummy Data (ADDRESS)
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3G 1A4');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3T 1E2');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H2E 1B4');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H2A 1M4');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3A 3M8');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H4A 3L5');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3W 1Y3');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3G 1P1');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3A 2B4');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H2X 3J6');
INSERT INTO Address(City, Province, PostalCode)
    values ('Toronto', 'ON', 'M5T 2S8');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H1T 2M4');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H1V 1A1');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H4A 2G7');
```

```

INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H4A 2B2');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3G 1Z2');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3C 1X9');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3B 5E6');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H1A 5B6');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H1L 2B3');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H1L 5R5');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H3Y 2T9');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'QC', 'H1C 2E8');
INSERT INTO Address(City, Province, PostalCode)
    values ('Montreal', 'ON', 'K0C 0A9');

#New Insertions Provinces
INSERT INTO Address(City, Province, PostalCode)
    values ('Vancouver', 'BC', 'V5K 1W1');
INSERT INTO Address(City, Province, PostalCode)
    values ('Calgary', 'AB', 'T2P 0B2');
INSERT INTO Address(City, Province, PostalCode)
    values ('Winnipeg', 'MB', 'R3C 1B2');

```

## 2. Person Table:

- Person(PersonID, FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
- Functional Dependencies:
  - PersonID → {FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence}
- Primary key: PersonID
- Foreign key: AddressID is a foreign key referencing the AddressID column in the Address table.

- `CREATE TABLE Person (`

```

PersonID INT PRIMARY KEY AUTO_INCREMENT,
FirstName VARCHAR(50) NOT NULL,
LastName VARCHAR(50) NOT NULL,
DateOfBirth DATE,
PSocialSecurityNumber VARCHAR(50) NOT NULL,
PMedicareCardNumber VARCHAR(50) NOT NULL,
PhoneNumber VARCHAR(20),
AddressID INT,
Citizenship VARCHAR(50),
EmailAddress VARCHAR(255),
Occupation VARCHAR(50),
IsPrimaryResidence BOOLEAN,
FOREIGN KEY (AddressID) REFERENCES Address(AddressID)
);
```

- Dummy data inserted

```
-- Insert Dummy Data (PERSON)
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Mark', 'Castel', '1990-04-30', 'MT1239-3928Q', '939-394-994', '+1 094-939-3943', 13, 'Canadian', 'mark_c@gmail.com', 'Economist', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Steve', 'Gibbons', '1994-03-03', 'MT5129-3928Q', '959-293-959', '+1 003-039-4854', 14, 'Canadian', 'steve_g@gmail.com', 'Waiter', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Castel', 'Pollie', '2001-03-04', 'MT3402-1239Q', '230-293-492', '+1 599-392-9302', 15, 'Canadian', 'castel_p@gmail.com', 'Barista', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Pepper', 'Versalle', '1967-10-20', 'MT0492-9321Q', '048-039-593', '+1 992-239-0403', 16, 'Canadian', 'pepper_v@gmail.com', 'Mechanic', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Gary', 'Connor', '1977-03-30', 'MT0490-1238Q', '592-513-240', '+1 498-382-4903', 17, 'Canadian', 'gary_c@gmail.com', 'Housewife', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Lucas', 'Feeh', '1980-09-02', 'MT0485-1238Q', '503-591-324', '+1 203-284-5930', 18, 'Canadian', 'lucas_f@gmail.com', 'Chef', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Sarah', 'Castel', '2020-12-20', 'MT4343-3040Q', '504-593-492', NULL, 13, 'Canadian', 'sarah_c@gmail.com', 'Student', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Phil', 'Castel', '2023-03-12', 'MT9491-2039Q', '592-394-568', NULL, 13, 'Canadian', 'phil_c@gmail.com', 'Student', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Eve', 'Pollie', '2001-05-30', 'MT0928-1293Q', '390-581-949', '+1 928-394-5963', 15, 'Canadian', 'eve_p@gmail.com', 'Researcher', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Charlie', 'Versalle', '1996-03-28', 'MT9248-0212Q', '421-581-392', '+1 940-356-2934', 16, 'Canadian', 'charlie_v@gmail.com', 'Cashier', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Yunya', 'Xiao', '1996-09-26', 'MT9248-0000Q', '111-111-111', '+1 555-555-5555', 16, 'Canadian', 'yunya@yahoo.ca', 'Firefighter', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Zackary', 'Smith', '2000-10-06', 'MT9248-1111Q', '222-222-222', '+1 777-777-7777', 18, 'Canadian', 'zack@gmail.com', 'Policeman', TRUE);
INSERT INTO Person(FirstName, LastName, DateOfBirth, PSocialSecurityNumber, PMedicareCardNumber, PhoneNumber, AddressID, Citizenship, EmailAddress, Occupation, IsPrimaryResidence)
values ('Odin', 'Skypes', '2000-10-06', 'MT9248-5555Q', '333-333-333', '+1 888-888-8888', 18, 'Canadian', 'odin@gmail.com', 'Student', TRUE);
```

### 3. Employee Table:

- Employee(EmployeeID, FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
- Functional Dependencies:
  - EmployeeID → {FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship }

- Primary key: EmployeeID
- Foreign key: AddressID is a foreign key referencing the AddressID column in the Address table, and FacilityID is a foreign key referencing the FacilityID column in the Facility table.

```
CREATE TABLE Employee (
    EmployeeID INT PRIMARY KEY AUTO_INCREMENT,
    FirstName VARCHAR(50) NOT NULL,
    LastName VARCHAR(50) NOT NULL,
    DateOfBirth DATE,
    ESocialSecurityNumber VARCHAR(50) NOT NULL,
    EMedicareCardNumber VARCHAR(50) NOT NULL,
    PhoneNumber VARCHAR(20),
    AddressID INT,
    EmailAddress VARCHAR(255),
    Occupation VARCHAR(50),
    FacilityID INT NOT NULL,
    ISManager BOOLEAN DEFAULT FALSE,
    IsPrimaryResidence BOOLEAN,
    Citizenship VARCHAR(50),
    FOREIGN KEY (AddressID) REFERENCES Address(AddressID)
);
```

- Dummy data inserted

```
-- Insert Dummy Data (EMPLOYEE)
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Leah', 'Castel', '1998-09-29', 'HT039-2123Q', '850-959-939', '+1 440-013-0392', 11, 'leah_c@gmail.com', 'Nurse', 1, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Wesley', 'Gibbons', '1997-10-20', 'HT0239-2930Q', '928-359-394', '+1 504-048-3042', 14, 'wesley_g@gmail.com', 'Doctor', 2, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Laurel', 'Pollie', '2000-05-31', 'HT0384-3928Q', '498-394-693', '+1 648-039-3592', 15, 'laurel_p@gmail.com', 'Nurse', 3, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Frederic', 'Versalle', '1999-02-10', 'HT0830-2930Q', '500-038-589', '+1 049-283-2030', 16, 'frederic_v@gmail.com', 'Pharmacist', 4, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Walsh', 'Connor', '2001-01-05', 'HT0234-3928Q', '936-322-384', '+1 043-302-9596', 17, 'walsh_c@gmail.com', 'Receptionist', 5, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Asher', 'Feah', '2000-04-20', 'HT0394-2302Q', '842-830-849', '+1 554-038-3953', 18, 'asher_f@gmail.com', 'Administrative Personnel', 6, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Nicholas', 'Butler', '2000-04-20', 'HT0232-1123Q', '088-039-932', '+1 554-030-3953', 19, 'nicholas_b@gmail.com', 'Security Personnel', 7, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Justin', 'Grant', '1999-02-03', 'HT0923-2934Q', '959-938-941', '+1 800-958-5829', 17, 'justin_g@gmail.com', 'Regular Employee', 8, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Malwina', 'Keating', '1994-03-04', 'HT0594-3928Q', '949-329-312', '+1 048-589-3857', 20, 'malwina_k@gmail.com', 'Nurse', 9, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Keanan', 'Ellast', '1993-04-05', 'HT0934-9230Q', '239-392-003', '+1 438-949-0394', 21, 'keanan_e@gmail.com', 'Doctor', 10, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Elian', 'Champagne', '1999-02-10', 'HT7836-2930Q', '591-038-000', '+1 949-203-2077', 16, 'elian_e@gmail.com', 'Pharmacist', 2, FALSE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Esmerelda', 'Lutin', '1999-03-10', 'A27836-2930Q', '300-038-000', '+1 400-203-2077', 22, 'esmerelda@gmail.com', 'Nurse', 11, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('George', 'Vanier', '1999-04-10', 'A37836-2930Q', '310-038-000', '+1 400-203-2077', 22, 'georgevanier@gmail.com', 'Nurse', 11, FALSE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Justine', 'Samuel', '1999-02-10', 'A7A836-2930Q', '591-038-111', '+1 949-203-2077', 14, 'justine_s@gmail.com', 'Nurse', 2, FALSE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Ali', 'Chanson', '1999-02-10', 'HT7836-211Q', '591-038-222', '+1 949-203-2077', 16, 'aly_c@gmail.com', 'Pharmacist', 2, FALSE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Eric', 'Ateller', '2000-11-11', 'HT7836-222Q', '591-078-133', '+1 049-203-2088', 17, 'ericateller@gmail.com', 'Doctor', 12, TRUE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Jasmine', 'Smith', '2001-11-11', 'HT7836-113Q', '591-038-444', '+1 949-203-1188', 14, 'jasmine_s@gmail.com', 'Nurse', 12, FALSE, TRUE, 'Canadian');

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
VALUES ('Yunya', 'Xiao', '1996-09-26', 'HT0248-0000Q', '111-111-111', '+1 555-555-5555', 16, 'yunya@yahoo.ca', 'Doctor', 13, FALSE, TRUE, 'Canadian');
```

```

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Zackary', 'Smith', '2000-10-08', 'MT9248-1111Q', '+22-222-222', '+1 777-777-777', 18, 'zack@gmail.com', 'Doctor', 12, FALSE, TRUE, 'Canadian');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Odin', 'Skyes', '2000-10-06', 'MT9248-5555Q', '333-333-333', '+1 888-888-8888', 18, 'odin@gmail.com', 'Doctor', 12, FALSE, TRUE, 'Canadian');

-- NEW INSERTIONS LEAH (1) NURSE -> 3 SECONDARY RESIDENCES
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Leah', 'Castel', '1998-09-29', 'MT0393-2323Q', '851-959-939', '+1 448-013-0392', 15, 'leah_c@gmail.com', 'Nurse', 3, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Leah', 'Castel', '1998-09-29', 'MT0393-2323Q', '850-959-939', '+1 448-013-0392', 16, 'leah_c@gmail.com', 'Nurse', 4, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Leah', 'Castel', '1998-09-29', 'MT0393-2323Q', '850-959-939', '+1 448-013-0392', 13, 'leah_c@gmail.com', 'Nurse', 2, FALSE, FALSE, 'USA');

-- NEW INSERTIONS JUSTIN (8) -> 3 SECONDARY RESIDENCES
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Justin', 'Grant', '1995-02-03', 'MT0923-2934Q', '959-938-941', '+1 830-958-9829', 15, 'justin_g@gmail.com', 'Regular Employee', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Justin', 'Grant', '1995-02-03', 'MT0923-2934Q', '959-938-941', '+1 830-958-9829', 16, 'justin_g@gmail.com', 'Regular Employee', 9, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Justin', 'Grant', '1995-02-01', 'MT0923-2940Q', '959-938-941', '+1 830-958-9829', 13, 'justin_g@gmail.com', 'Regular Employee', 10, FALSE, FALSE, 'USA');

-- NEW INSERTIONS WESLEY (3)
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Wesley', 'Gibbons', '1997-10-26', 'MT0239-2930Q', '920-359-394', '+1 584-040-3842', 17, 'wesley_g@gmail.com', 'Doctor', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Wesley', 'Gibbons', '1997-10-26', 'MT0239-2930Q', '920-359-394', '+1 584-040-3842', 18, 'wesley_g@gmail.com', 'Doctor', 2, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Wesley', 'Gibbons', '1997-10-26', 'MT0239-2930Q', '920-359-394', '+1 584-040-3842', 19, 'wesley_g@gmail.com', 'Doctor', 3, FALSE, FALSE, 'USA');

-- NEW INSERTIONS LAUREL NURSE (3)
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Laurel', 'Pollie', '2000-05-31', 'MT0384-3920Q', '498-394-093', '+1 040-039-3592', 16, 'laurel_p@gmail.com', 'Nurse', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Laurel', 'Pollie', '2000-05-31', 'MT0384-3920Q', '498-394-093', '+1 040-039-3592', 17, 'laurel_p@gmail.com', 'Nurse', 2, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Laurel', 'Pollie', '2000-05-31', 'MT0384-3920Q', '498-394-093', '+1 040-039-3592', 18, 'laurel_p@gmail.com', 'Nurse', 3, FALSE, FALSE, 'USA');

-- NEW INSERTIONS ESMERALDA NURSE (12)
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Esmeralda', 'Lutin', '1999-03-18', 'A27836-2930Q', '300-038-000', '+1 400-203-2877', 19, 'Esmeralda@gmail.com', 'Nurse', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Esmeralda', 'Lutin', '1999-03-18', 'A27836-2930Q', '300-038-000', '+1 400-203-2877', 20, 'Esmeralda@gmail.com', 'Nurse', 9, FALSE, FALSE, 'USA');

-- NEW INSERTIONS ESMERALDA (18)
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Esmeralda', 'Lutin', '1999-03-18', 'A27836-2930Q', '300-038-000', '+1 400-203-2877', 21, 'Esmeralda@gmail.com', 'Nurse', 9, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Esmeralda', 'Lutin', '1999-03-18', 'A27836-2930Q', '300-038-000', '+1 400-203-2877', 18, 'Esmeralda@gmail.com', 'Nurse', 11, FALSE, FALSE, 'USA');

-- NEW INSERTIONS MELWINA (1)
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Melwinia', 'Keating', '1994-03-04', 'MT9594-3020Q', '949-329-312', '+1 048-589-3857', 20, 'melwinia_k@gmail.com', 'Nurse', 1, FALSE, FALSE, 'USA');

-- NEW INSERTIONS NURSE 9
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('George', 'Vanler', '1994-03-04', 'MT1111-1111Q', '999-329-311', '+1 333-589-3857', 1, 'georgev@gmail.com', 'Nurse', 9, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('George', 'Vanler', '1994-03-04', 'MT1111-1111Q', '999-329-311', '+1 333-589-3857', 2, 'georgev@gmail.com', 'Nurse', 10, FALSE, FALSE, 'USA');

-- NEW Insertions Out Of Province
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Constantin', 'Valentin', '1993-03-04', 'MT2121-1111Q', '800-329-312', '+1 818-212-3857', 15, 'c_valentin@gmail.com', 'Nurse', 13, TRUE, TRUE, 'Canada');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Sunny', 'Elden', '1993-03-04', 'MT2222-1124Q', '455-329-312', '+1 800-212-3857', 26, 's_elden@gmail.com', 'Doctor', 14, TRUE, TRUE, 'Canada');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Eliane', 'Suprano', '1993-03-04', 'MT3333-1236Q', '466-329-312', '+1 233-212-3857', 27, 'e_suprano@gmail.com', 'Nurse', 15, TRUE, TRUE, 'Canada');

-- New Insert statements for Query 14
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Concordia', 'Suprano', '1993-03-04', 'MT3333-1236Q', '466-329-312', '+1 233-212-3857', 16, 'c_suprano@gmail.com', 'Nurse', 1, FALSE, TRUE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Beethoven', 'Suprano', '1993-03-04', 'MT4321-1236Q', '466-329-312', '+1 233-212-3857', 16, 'b_suprano@gmail.com', 'Doctor', 1, FALSE, TRUE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Just', 'Suprano', '1993-03-04', 'MT4335-1236Q', '466-329-312', '+1 233-212-3857', 16, 'j_suprano@gmail.com', 'Doctor', 1, FALSE, TRUE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Clergy', 'Suprano', '1993-03-04', 'MT4898-1236Q', '466-329-312', '+1 233-212-3857', 16, 'cs_suprano@gmail.com', 'Doctor', 1, FALSE, TRUE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Estaban', 'Suprano', '1993-03-04', 'MT9483-1236Q', '466-329-312', '+1 233-212-3857', 16, 'estaban_suprano@gmail.com', 'Nurse', 1, FALSE, TRUE, 'USA');

-- duplicate
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Concordia', 'Suprano', '1993-03-04', 'MT3333-1236Q', '466-329-312', '+1 233-212-3857', 17, 'c_suprano@gmail.com', 'Nurse', 1, FALSE, TRUE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Beethoven', 'Suprano', '1993-03-04', 'MT4321-1236Q', '466-329-312', '+1 233-212-3857', 17, 'b_suprano@gmail.com', 'Doctor', 1, FALSE, TRUE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Just', 'Suprano', '1993-03-04', 'MT4335-1236Q', '466-329-312', '+1 233-212-3857', 17, 'j_suprano@gmail.com', 'Doctor', 1, FALSE, TRUE, 'USA');

```

```

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Clergy', 'Suprano', '1993-03-04', 'MT4889-123Q', '466-329-312', '+1 233-212-3857', 17, 'cs_suprano@gmail.com', 'Doctor', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Estaban', 'Suprano', '1993-03-04', 'MT9483-123Q', '466-329-312', '+1 233-212-3857', 17, 'estaban_suprano@gmail.com', 'Nurse', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Concordia', 'Suprano', '1993-03-04', 'MT3333-123Q', '466-329-312', '+1 233-212-3857', 18, 'c_suprano@gmail.com', 'Nurse', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Beethoven', 'Suprano', '1993-03-04', 'MT4231-123Q', '466-329-312', '+1 233-212-3857', 18, 'b_suprano@gmail.com', 'Doctor', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Just', 'Suprano', '1993-03-04', 'MT4335-123Q', '466-329-312', '+1 233-212-3857', 18, 'j_suprano@gmail.com', 'Doctor', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Clergy', 'Suprano', '1993-03-04', 'MT4889-123Q', '466-329-312', '+1 233-212-3857', 18, 'cs_suprano@gmail.com', 'Doctor', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Estaban', 'Suprano', '1993-03-04', 'MT9483-123Q', '466-329-312', '+1 233-212-3857', 18, 'estaban_suprano@gmail.com', 'Nurse', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Concordia', 'Suprano', '1993-03-04', 'MT3333-123Q', '466-329-312', '+1 233-212-3857', 18, 'c_suprano@gmail.com', 'Nurse', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Beethoven', 'Suprano', '1993-03-04', 'MT4231-123Q', '466-329-312', '+1 233-212-3857', 18, 'b_suprano@gmail.com', 'Doctor', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Just', 'Suprano', '1993-03-04', 'MT4335-123Q', '466-329-312', '+1 233-212-3857', 18, 'j_suprano@gmail.com', 'Doctor', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Clergy', 'Suprano', '1993-03-04', 'MT4889-123Q', '466-329-312', '+1 233-212-3857', 18, 'cs_suprano@gmail.com', 'Doctor', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Estaban', 'Suprano', '1993-03-04', 'MT9483-123Q', '466-329-312', '+1 233-212-3857', 18, 'estaban_suprano@gmail.com', 'Nurse', 1, FALSE, FALSE, 'USA');
INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Jordan', 'Suprano', '1993-03-04', 'MT9483-123Q', '466-329-312', '+1 233-212-3857', 18, 'estaban_suprano@gmail.com', 'Doctor', 12, FALSE, FALSE, 'USA');

-- ALEX'S ADDED INSERTS

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('John', 'Doe', '1985-06-15', '123-45-6789', 'ABC123456789', 19, 'john.doe@example.com', 'Doctor', 12, TRUE, TRUE, 'Canadian');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('Jane', 'Smith', '1990-03-25', '587-65-4321', 'XYZ987654321', '+1987654321', 20, 'jane.smith@example.com', 'Doctor', 12, TRUE, TRUE, 'American');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('John', 'Doe', '1985-06-15', '123-45-6789', 'ABC123456789', '+1234567890', 21, 'john.doe@example.com', 'Doctor', 12, TRUE, FALSE, 'Canadian');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('Jane', 'Smith', '1980-03-25', '587-65-4321', 'XYZ987654321', '+1987654321', 22, 'jane.smith@example.com', 'Doctor', 12, TRUE, FALSE, 'American');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('Emily', 'Johnson', '1992-08-20', '345-67-8901', 'DEF345678901', '+112345678901', 19, 'emily.johnson@example.com', 'Nurse', 12, FALSE, TRUE, 'American');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('Michael', 'Williams', '1988-05-12', '678-90-1234', 'GHI678901234', '+1234567895', 22, 'michael.williams@example.com', 'Doctor', 12, FALSE, FALSE, 'Canadian');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('Sophia', 'Brown', '1995-12-03', '981-23-4567', 'JKL9801234567', '+1334456789', 19, 'sophia.brown@example.com', 'Receptionist', 12, FALSE, TRUE, 'British');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('Emily', 'Johnson', '1992-08-20', '345-67-8901', 'DEF345678901', '+112345678901', 21, 'emily.johnson@example.com', 'Nurse', 12, FALSE, FALSE, 'American');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('Michael', 'Williams', '1988-05-12', '678-90-1234', 'GHI678901234', '+1234567895', 22, 'michael.williams@example.com', 'Doctor', 12, FALSE, FALSE, 'Canadian');

INSERT INTO Employee (FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, ISPrimaryResidence, Citizenship)
VALUES ('Sophia', 'Brown', '1995-12-03', '981-23-4567', 'JKL9801234567', '+1334456789', 21, 'sophia.brown@example.com', 'Receptionist', 12, FALSE, FALSE, 'British');

-- ALEX'S ADDED INSERT

INSERT INTO Employee(FirstName, LastName, DateOfBirth, ESocialSecurityNumber, EMedicareCardNumber, PhoneNumber, AddressID, EmailAddress, Occupation, FacilityID, ISManager, IsPrimaryResidence, Citizenship)
    VALUES ('Leah', 'Castel', '1998-09-25', 'MTO983-2323Q', '850-959-939', '+1 440-613-0392', 14, 'leah@gmail.com', 'Nurse', 1, TRUE, FALSE, 'Canadian');

```

#### 4. Residence Table:

- Residence(ResidenceID, AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
  - Functional Dependencies:
    - ResidenceID → {AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType}
  - Primary key: ResidenceID
  - Foreign key: AddressID is a foreign key referencing the AddressID column in the Address table.

```
CREATE TABLE Residence (
    ResidenceID INT PRIMARY KEY AUTO_INCREMENT,
    AddressID INT NOT NULL,
    PhoneNumber VARCHAR(20),
    NumberOfBedrooms INT,
    ResidenceType VARCHAR(50),
    FOREIGN KEY (AddressID) REFERENCES Address(AddressID)
);
```

- Dummy data inserted

```
-- Insert Dummy Data (RESIDENCE)
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (13, '+1 514-892-3691', 1, 'Apartment');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (14, '+1 438-880-2013', 1, 'Apartment');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (15, '+1 514-758-5885', 2, 'Apartment');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (16, '+1 492-592-0958', 5, 'Condominium');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (17, '+1 322-492-9850', 3, 'Condominium');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (18, '+1 403-382-3958', 4, 'Condominium');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (19, '+1 491-239-4051', 4, 'Semidetached');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (20, '+1 514-961-9116', 3, 'House');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (21, '+1 514-926-2938', 2, 'House');
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (22, '+1 416-923-2930', 2, 'House');
-- Toronto
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (23, '+1 416-000-0000', 1, 'Condominium');
-- Vancouver
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (24, '+1 604-000-0000', 2, 'House');
-- Calgary
INSERT INTO Residence(AddressID, PhoneNumber, NumberOfBedrooms, ResidenceType)
    values (25, '+1 403-000-0000', 4, 'House');
```

## 5. Infection Table:

- Infection(InfectionID, PersonID, EmployeeID, InfectionType, InfectionDate)
- Functional Dependencies:
  - InfectionID → {PersonID, EmployeeID, InfectionType, InfectionDate}
- Primary key: InfectionID
- Foreign key: PersonID is a foreign key referencing the PersonID column in the Person table, and EmployeeID is a foreign key referencing the EmployeeID column in the Employee table.

```
CREATE TABLE Infection (
    InfectionID INT PRIMARY KEY AUTO_INCREMENT,
    EmployeeID INT,
    PersonID INT,
    InfectionType VARCHAR(50),
    InfectionDate DATE,
    FOREIGN KEY (PersonID) REFERENCES Person(PersonID),
    FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID)
);
```

- Dummy data inserted

```
-- Insert Dummy Data (INFECTION)
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (1, NULL, 'Covid-19', '2020-05-30');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (4, NULL, 'Delta', '2021-05-02');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (5, NULL, 'Covid-19', '2021-02-04');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (7, NULL, 'Omicron', '2022-02-16');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (9, NULL, 'Delta', '2021-05-16');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (10, NULL, 'SARS-Cov-2', '2024-01-01');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (NULL, 1, 'Omicron', '2022-05-24');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (NULL, 5, 'Omicron', '2022-07-21');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (NULL, 6, 'Delta', '2021-06-20');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (NULL, 10, 'Delta', '2021-07-01');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (11, NULL, 'Covid-19', '2020-05-29');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (12, NULL, 'Covid-19', '2020-05-29');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (14, NULL, 'Covid-19', '2020-05-29');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (17, NULL, 'Covid-19', '2020-05-29');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (16, NULL, 'Covid-19', '2020-05-29');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (1, NULL, 'Omicron', '2020-05-30');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (1, NULL, 'Delta', '2020-05-30');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (20, NULL, 'Covid-19', '2022-02-16');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (20, NULL, 'Delta', '2022-02-16');
```

```

INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (20, NULL, 'Omicron', '2022-02-16');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (20, NULL, 'Influenza', '2022-02-16');
--newline
--NEW INSERTION
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (1, NULL, 'Covid-19', '2024-04-01');
--NEW INSERTION
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (3, NULL, 'Covid-19', '2024-04-02');
--NEW INSERTION
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (12, NULL, 'Covid-19', '2024-04-03');
--NEW INSERTION
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (9, NULL, 'Covid-19', '2024-04-03');
--NEW INSERTION
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (39, NULL, 'Covid-19', '2024-04-03');
--NEW provinces
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (41, NULL, 'Covid-19', '2024-04-03');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (42, NULL, 'Covid-19', '2024-04-03');
INSERT INTO Infection(EmployeeID, PersonID, InfectionType, InfectionDate)
    values (43, NULL, 'Covid-19', '2024-04-03');

-- ALEX'S NEW INSERTION
INSERT INTO Infection (EmployeeID, PersonID, InfectionType, InfectionDate)
    VALUES (2, NULL, 'COVID-19', '2024-04-08');
INSERT INTO Infection (EmployeeID, PersonID, InfectionType, InfectionDate)
    VALUES (65, NULL, 'COVID-19', '2024-04-08');
INSERT INTO Infection (EmployeeID, PersonID, InfectionType, InfectionDate)
    VALUES (19, NULL, 'COVID-19', '2024-04-08');
INSERT INTO Infection (EmployeeID, PersonID, InfectionType, InfectionDate)
    VALUES (70, NULL, 'COVID-19', '2024-04-08');
INSERT INTO Infection (EmployeeID, PersonID, InfectionType, InfectionDate)
    VALUES (11, NULL, 'COVID-19', '2024-04-08');

```

## 6. Facility Table:

- Facility( FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, WebAdress, FacilityCapacity, ManagerID)
- Functional Dependencies:
  - FacilityID → {FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, WebAdress, FacilityCapacity, ManagerID}
- Primary key: FacilityID
- Foreign key: AddressID is a foreign key referencing the AddressID column in the Address table, and ManagerID is a foreign key referencing the EmployeeID column in the Employee table.

```

CREATE TABLE Facility (
    FacilityID INT PRIMARY KEY AUTO_INCREMENT,
    FacilityName VARCHAR(255),
    FacilityType VARCHAR(50),
    AddressID INT NOT NULL,
    PhoneNumber VARCHAR(20),
    EmailAddress VARCHAR(255),
    FacilityCapacity INT ,
    WebAddress VARCHAR(255),
    ManagerID INT unique NOT NULL,
    FOREIGN KEY (AddressID) REFERENCES Address(AddressID),
    FOREIGN KEY (ManagerID) REFERENCES Employee(EmployeeID)
);

```

- Dummy data inserted

```

-- Insert Dummy Data (FACILITY)
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (1, 'Montreal General Hospital', 'Hospital', 1, '514-555-1234', 'Info@montrealgeneralhospital.com', 5, 'http://www.montrealgeneralhospital.com', 1);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (2, 'Jewish General Hospital', 'Hospital', 2, '514-555-5678', 'Info@jewishgeneralhospital.com', 5, 'http://www.jewishgeneralhospital.com', 2);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (3, 'CLSC Metro', 'CLSC', 3, '514-555-9012', 'Info@clsclongueuil.com', 4, 'http://www.clsclongueuil.com', 3);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (4, 'CLSC de la Petite-Patrie', 'CLSC', 4, '514-555-3456', 'Info@clsclapetitepatrie.com', 4, 'http://www.clsclapetitepatrie.com', 4);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (5, 'Medisys Executive Health Clinic', 'Clinic', 5, '514-555-7890', 'Info@medisysclinic.com', 3, 'http://www.medisysclinic.com', 5);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (6, 'Montreal Walk-In Clinic - Queen Elizabeth Health Complex', 'Clinic', 6, '514-555-2345', 'Info@montrealwalkinclinic.com', 3, 'http://www.montrealwalkinclinic.com', 6);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (7, 'Jean Coutu Pharmacy', 'Pharmacy', 7, '514-555-0789', 'Info@jeancoutupharmacy.com', 2, 'http://www.jeancoutupharmacy.com', 7);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (8, 'Pharmaprix (Shoppers Drug Mart)', 'Pharmacy', 8, '514-555-1234', 'Info@pharmaprix.com', 2, 'http://www.pharmaprix.com', 8);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (9, 'Montreal Neurological Hospital - Neurology Department', 'Special Installment', 9, '514-555-5678', 'Info@montrealneurologicalhospital.com', 2, 'http://www.montrealneurologicalhospital.com', 9);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (10, 'Dentistry - Clinique Dentaire du Quartier Latin', 'Special Installment', 10, '514-555-9012', 'Info@cliniquedentairequartierlatin.com', 2, 'http://www.cliniquedentairequartierlatin.com', 10);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (11, 'Toronto Western Hospital', 'Hospital', 11, '416-555-3456', 'Info@torontowesternhospital.com', 2, 'http://www.torontowesternhospital.com', 11);
INSERT INTO Facility(FacilityID, FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
VALUES (12, 'Hospital Maisonneuve Rosemont', 'Hospital', 12, '514-555-7890', 'Info@maisonneuverosemonthospital.com', 3, 'http://www.maisonneuverosemonthospital.com', 12);

-- NEW INSERTION
-- In Vancouver
INSERT INTO Facility(FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
values ('Vancouver General Hospital', 'Hospital', 25, '+1 684-875-4111', 'Info@vgh.ca', 500, 'www.vancouvergeneralhospital.ca', 41);

-- In Calgary
INSERT INTO Facility(FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
values ('Calgary Medical Centre', 'Hospital', 26, '+1 403-944-1110', 'Info@calgarmedicalcentre.ca', 400, 'www.calgarymedicalcentre.ca', 42);

-- In Winnipeg
INSERT INTO Facility(FacilityName, FacilityType, AddressID, PhoneNumber, EmailAddress, FacilityCapacity, WebAddress, ManagerID)
values ('Health Sciences Centre Winnipeg', 'Hospital', 27, '+1 204-787-3661', 'Info@hsc.mb.ca', 300, 'www.hsc.mb.ca', 43);

```

## 7. EmployeeAtFacility Table:

- EmployeeAtFacility(EmployeeAtFacilityID, EmployeeID, FacilityID, StartDate, EndDate)
- Functional Dependencies:
  - EmployeeAtFacilityID → {EmployeeID, FacilityID, StartDate, EndDate}
- Primary key: EmployeeAtFacilityID
- Foreign key: EmployeeID is a foreign key referencing the EmployeeID column in the Employee table, and FacilityID is a foreign key referencing the FacilityID column in the Facility table.

```
CREATE TABLE EmployeeAtFacility (
    EmployeeAtFacilityID INT PRIMARY KEY AUTO_INCREMENT,
    EmployeeID INT NOT NULL,
    FacilityID INT NOT NULL,
    StartDate DATE NOT NULL,
    EndDate DATE,
    FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID),
    FOREIGN KEY (FacilityID) REFERENCES Facility(FacilityID)
);
```

- Dummy data inserted

```
-- Insert Dummy Data (EMPLOYEE ASSIGNMENT)
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (1, 1, '2024-01-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (2, 2, '2023-12-30', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (3, 3, '2010-03-20', '2015-08-24');
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (4, 4, '2021-05-03', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (5, 5, '2018-03-05', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (6, 6, '2019-10-10', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (7, 7, '2009-07-10', '2020-01-20');
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (8, 8, '2011-04-12', '2017-01-30');
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (9, 9, '2014-10-03', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (10, 10, '2015-11-22', '2023-01-04');
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (11, 2, '2021-05-03', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (12, 11, '2021-05-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (13, 11, '2021-05-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (14, 2, '2021-05-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (15, 2, '2021-05-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (16, 12, '2021-05-10', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (17, 12, '2021-05-11', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (18, 12, '2019-04-11', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (19, 12, '2023-10-04', NULL);
```

```
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (28, 12, '2022-10-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (25, 12, '2023-10-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (26, 12, '2023-10-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (27, 12, '2022-10-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (28, 12, '2023-10-04', NULL);

-- ALEX'S ADDED INSERTS
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (28, 12, '2023-10-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (64, 12, '2023-10-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (65, 12, '2023-10-04', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (73, 12, '2023-10-04', NULL);

--NEW INSERTION
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (9, 1, '2014-10-03', NULL);
--NEW INSERTION, Justin
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (8, 1, '2014-10-03', NULL);
--NEW INSERTION , Wesley
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (2, 1, '2018-10-03', NULL);
--NEW INSERTION , Laurel
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (3, 1, '2017-10-03', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (3, 2, '2017-10-03', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (3, 3, '2017-10-03', NULL);
```

```
##NEW INSERTION , Esmeralda
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (12, 1, '2017-10-03', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (12, 9, '2017-10-03', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (12, 11, '2017-10-03', NULL);
##NEW LEAH
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (1, 2, '2019-01-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (1, 3, '2019-01-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (1, 4, '2019-01-02', NULL);
##NEW NURSES
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (39, 9, '2019-02-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (39, 10, '2019-02-02', NULL);
##New Out of Province
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (41, 13, '2019-02-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (42, 14, '2019-02-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (43, 15, '2019-02-02', NULL);
##New again
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (44, 1, '2019-02-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (45, 1, '2019-02-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (46, 1, '2019-02-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (47, 1, '2019-02-02', NULL);
INSERT INTO EmployeeAtFacility(EmployeeID, FacilityID, StartDate, EndDate)
    values (48, 1, '2019-02-02', NULL);
```

## 8. PersonAtResidence Table:

- PersonAtResidence( PersonAtResidenceID, PersonID, ResidenceID, Relationship, IsPrimaryResidence )
- Functional Dependencies:
  - PersonAtResidenceID → {PersonID, ResidenceID, Relationship, IsPrimaryResidence}
- Primary key: PersonAtResidenceID
- Foreign key: PersonID is a foreign key referencing the PersonID column in the Person table, and ResidenceID is a foreign key referencing the ResidenceID column in the Residence table.

```
CREATE TABLE PersonAtResidence (
    PersonAtResidenceID INT PRIMARY KEY AUTO_INCREMENT,
    PersonID INT NOT NULL,
    ResidenceID INT NOT NULL,
    Relationship VARCHAR(50),
    IsPrimaryResidence BOOLEAN,
    FOREIGN KEY (PersonID) REFERENCES Person(PersonID),
    FOREIGN KEY (ResidenceID) REFERENCES Residence(ResidenceID),
    UNIQUE (PersonID, ResidenceID)
);
```

- Dummy data inserted

```
-- Insert Dummy Data (PERSONATRESIDENCE)
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(1, 1, "Partner", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(2, 2, "Non-Parent Dependent", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(3, 3, "Non-Parent Dependent", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(4, 4, "Parent", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(5, 5, "Parent", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(6, 6, "Partner", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(7, 1, "Children", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(8, 1, "Children", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(9, 3, "Roommate", TRUE);
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(10, 4, "Roommate", TRUE);
-- ALEX'S ADDED INSERTS
INSERT INTO PersonAtResidence(PersonID, ResidenceID, Relationship, IsPrimaryResidence)
    values(5, 2, "Non-Parent Dependent", TRUE);
```

## 9. Vaccination Table:

- Vaccination(VaccinationID, EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
- Functional Dependencies:
  - VaccinationID → {EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID}
- Primary key: VaccinationID
- Foreign key: PersonID is a foreign key referencing the PersonID column in the Person table, and FacilityID is a foreign key referencing the FacilityID column in the Facility table.

```
CREATE TABLE Vaccination(
    VaccinationID INT PRIMARY KEY AUTO_INCREMENT,
    EmployeeID INT,
    PersonID INT,
    VaccineType VARCHAR(50),
    DoseNumber INT,
    VaccinationDate DATE,
    FacilityID INT,
    FOREIGN KEY (PersonID) REFERENCES Person(PersonID),
    FOREIGN KEY (FacilityID) REFERENCES Facility(FacilityID)
);
```

- Dummy data inserted

```
-- Insert Dummy Data (VACCINATION)
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (1, NULL, 'Pfizer', 2, '2020-06-30', 1);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (2, NULL, 'Moderna', 3, '2020-07-01', 2);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (3, NULL, 'AstraZeneca', 2, '2020-07-02', 3);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (4, NULL, 'Johnson&Johnson', 3, '2020-07-03', 4);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (5, NULL, 'Pfizer', 2, '2020-06-30', 5);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (6, NULL, 'Moderna', 3, '2020-07-01', 6);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (7, NULL, 'AstraZeneca', 2, '2020-07-02', 7);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (8, NULL, 'Johnson&Johnson', 1, '2020-07-03', 8);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (9, NULL, 'Pfizer', 2, '2020-06-30', 9);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (10, NULL, 'Moderna', 3, '2020-07-01', 10);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (NULL, 1, 'AstraZeneca', 4, '2020-07-02', 1);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (NULL, 2, 'Johnson&Johnson', 2, '2020-07-03', 2);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (NULL, 3, 'Pfizer', 2, '2020-06-30', 1);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (NULL, 4, 'Moderna', 2, '2020-07-01', 2);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
values (NULL, 5, 'AstraZeneca', 3, '2020-07-02', 5);
```

```

INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (NULL, 6, 'Johnson&Johnson', 1, '2020-07-03', 6);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (NULL, 7, 'Pfizer', 1, '2020-06-30', 6);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (NULL, 8, 'Moderna', 2, '2020-07-01', 8);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (NULL, 9, 'AstraZeneca', 3, '2020-07-02', 4);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (NULL, 10, 'Johnson&Johnson', 4, '2020-07-03', 10);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (11, NULL, 'Johnson&Johnson', 4, '2020-07-03', 2);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (12, NULL, 'Moderna', 3, '2020-07-04', 11);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (13, NULL, 'Johnson&Johnson', 4, '2020-07-05', 11);
--new
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (39, NULL, 'Johnson&Johnson', 3, '2020-07-05', 9);
INSERT INTO Vaccination(EmployeeID, PersonID, VaccineType, DoseNumber, VaccinationDate, FacilityID)
    values (73, NULL, 'Johnson&Johnson', 3, '2020-07-05', 9);

```

## 10. WorkSchedule Table:

- WorkSchedule(ScheduleID, EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
- Functional Dependencies:
  - ScheduleID → {EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime}
- Primary key: ScheduleID

```

CREATE TABLE WorkSchedule (
    ScheduleID INT PRIMARY KEY AUTO_INCREMENT ,
    EmployeeID INT NOT NULL,
    FacilityID INT NOT NULL,
    ScheduleDate DATE,
    StartTime TIME NOT NULL,
    EndTime TIME,
    CONSTRAINT check_dates
        CHECK (StartTime < EndTime)
);

```

- Dummy data inserted

```

-- Insert Dummy Data (WORKSCHEDULE)
INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
    values(1, 1, '2020-08-31', '10:00', '14:00');
INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
    values(1, 1, '2020-08-21', '10:00', '14:00');
INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
    values(2, 1, '2020-08-21', '12:00', '16:00');
INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
    values(3, 1, '2020-08-21', '8:00', '12:00');

```

```

    ||new for emp9
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(9, 9, '2020-07-21', '14:45', '16:30');

    ||new
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(12, 9, '2020-07-21', '14:45', '16:30');

    ||new
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(39, 9, '2020-07-21', '12:00', '18:00');

    ||New provinces
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(41, 13, '2020-07-21', '12:00', '18:00');
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(42, 14, '2020-07-21', '12:00', '18:00');
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(43, 15, '2020-07-21', '12:00', '18:00');
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(12, 11, '2020-07-21', '8:00', '12:00');

    ||New again
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(44, 1, '2024-04-01', '8:00', '12:00');
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(45, 1, '2024-04-01', '8:00', '12:00');
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(46, 1, '2024-04-01', '8:00', '12:00');
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(47, 1, '2024-04-01', '8:00', '12:00');
    INSERT INTO WorkSchedule(EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
        values(48, 1, '2024-04-01', '8:00', '12:00');

    INSERT INTO WorkSchedule (EmployeeID, FacilityID, ScheduleDate, StartTime, EndTime)
VALUES
    (73, 12, '2024-04-10', '09:00:00', '12:00:00'),
    (73, 12, '2024-04-15', '13:00:00', '17:00:00'),
    (73, 12, '2024-04-16', '10:00:00', '14:00:00'),
    (73, 12, '2024-04-20', '08:30:00', '12:30:00'),
    (73, 12, '2024-04-25', '09:30:00', '13:30:00');

```

## 11. Log Table:

- Log (LogID, EmailDate, Sender, Receiver, EmailSubject, BodyPreview)
- Functional Dependencies:
  - LogID → {EmailDate, Sender, Receiver, EmailSubject, BodyPreview}
- Primary key: LogID

```

CREATE TABLE Log (
    LogID INT PRIMARY KEY AUTO_INCREMENT,
    EmailDate DATE,
    Sender VARCHAR(255),
    Receiver VARCHAR(255),
    EmailSubject VARCHAR(255),
    BodyPreview VARCHAR(100)
);

```

12. EmailQueue table:

- EmailQueue (EmailID, EmailAddress, Subject, Body)
- Functional Dependencies:
  - EmailID → { EmailAddress, Subject, Body}
- Primary key: EmailID

```
↳ CREATE TABLE EmailQueue (
    EmailID INT PRIMARY KEY AUTO_INCREMENT,
    EmailAddress VARCHAR(255),
    Subject VARCHAR(255),
    Body VARCHAR(255)
    );
```

## **Normalization**

### 1. Address Table

- The primary key ‘AddressID’ is the super key, and all other attributes (‘City’, ‘Province’, ‘PostalCode’) are fully dependent on the primary key.

### 2. Person Table

- The primary key ‘PersonID’ is the super key, and all other attributes are fully dependent on the primary key.

### 3. Employee Table

- The primary key ‘EmployeeID’ is the super key, and all other attributes are fully dependent on the primary key.

### 4. Residence Table

- The primary key ‘ResidenceID’ is the super key, and all other attributes are fully dependent on the primary key.

### 5. Infection Table

- The primary key ‘InfectionID’ is the super key, and all other attributes are fully dependent on the primary key.

### 6. Facility Table

- The primary key ‘FacilityID’ is the super key, and all other attributes are fully dependent on the primary key.

### 7. EmployeeAtFacility Table

- The primary key ‘EmployeeAtFacilityID’ is the super key, and all other attributes are fully dependent on the primary key.

## 8. PersonAtResidence Table

- The primary key ‘PersonAtResidenceID’ is the super key, and all other attributes are fully dependent on the primary key.

## 9. Vaccination Table

- The primary key ‘VaccinationID’ is the super key, and all other attributes are fully dependent on the primary key.

## 10. WorkSchedule Table

- The primary key ‘ScheduleID’ is the super key, and all other attributes are fully dependent on the primary key.

## 11. Log Table

- The primary key ‘LogID’ is the super key, and all other attributes are fully dependent on the primary key.

## 12. EmailQueue

- The primary key ‘EmialID’ is the super key, and all other attributes are fully dependent on the primary key.

-3NF?

The provided database schema appears to be in Third Normal Form (3NF). Each table has a primary key, and all non-key attributes are fully dependent on the primary key. There are no transitive dependencies, and all functional dependencies are based on the primary key.

- BCNF?

All the tables in the provided database schema are in Boyce-Codd Normal Form (BCNF).

The primary keys of each table are the super keys, and all non-key attributes are fully dependent on the primary keys, satisfying the BCNF requirements.

## Functionalities Implemented

### SQL: Queries

- (8)

```
-- QUERY 8
SELECT
    F.FacilityName,
    A.City,
    A.Province,
    A.PostalCode,
    F.PhoneNumber,
    F.WebAddress,
    F.FacilityType,
    F.FacilityCapacity,
    CONCAT(E.FirstName, ' ', E.LastName) AS General_Manager,
    COUNT(EF.EmployeeID) AS Num_Employees,
    SUM(CASE WHEN E.Occupation = 'Doctor' THEN 1 ELSE 0 END) AS Num_Doctors,
    SUM(CASE WHEN E.Occupation = 'Nurse' THEN 1 ELSE 0 END) AS Num_Nurses
FROM
    Facility F
JOIN
    Address A ON F.AddressID = A.AddressID
JOIN
    Employee E ON F.ManagerID = E.EmployeeID
LEFT JOIN
    EmployeeAtFacility EF ON F.FacilityID = EF.FacilityID
GROUP BY
    F.FacilityID
ORDER BY
    A.Province ASC, A.City ASC, F.FacilityType ASC, Num_Doctors ASC;
```

### Result:

FacilityName	City	Province	PostalCode	PhoneNumber	WebAddress	FacilityType	FacilityCapacity	General_Manager	Num_Employees	Num_Doctors	Num_Nurses
Calgary Medical Centre	Calgary	AB	T2P 0B2	+1 403-944-1110	www.calgarymedicalcentre.ca	Hospital	400	Eliane Suprano	1	0	1
Vancouver General Hospital	Vancouver	BC	V5K 1W1	+1 604-875-4111	www.vancouvergeneralhospital.ca	Hospital	500	Sammy Elden	1	1	0
Health Sciences Centre Winnipeg	Winnipeg	MB	R3C 1B2	+1 204-787-3661	www.hsc.mb.ca	Hospital	300	Concordia Suprano	1	0	1
Toronto Western Hospital	Toronto	ON	M5T 2S8	416-555-3456	http://www.torontowesternhospital.com	Hospital	2	Esmerelda Lutin	3	0	3
Medisys Executive Health Clinic	Montreal	QC	H3A 3M8	514-555-7890	http://www.medisysclinic.com	Clinic	3	Walsh Connor	1	0	0
Montreal Walk-In Clinic - Queen Elizabeth Healt...	Montreal	QC	H4A 3L5	514-555-2345	http://www.montrealwalkinclinic.com	Clinic	3	Asher Feah	1	0	0
CLSC Metro	Montreal	QC	H2E 1B4	514-555-9012	http://www.clsclongueuil.com	CLSC	4	Laurel Pollie	3	0	3
CLSC de la Petite-Patrie	Montreal	QC	H2A 1M4	514-555-3456	http://www.clscpetitepatrie.com	CLSC	4	Frederic Versalle	2	0	0
Montreal General Hospital	Montreal	QC	H3G 1A4	514-555-1234	http://www.montrealgeneralhospital.com	Hospital	5	Leah Castel	11	0	11
Jewish General Hospital	Montreal	QC	H3T 1E2	514-555-5678	http://www.jewishgeneralhospital.com	Hospital	5	Wesley Gibbons	6	6	0
Hospital Maisonneuve Rosemont	Montreal	QC	H1T 2M4	514-555-7890	http://www.maisonneuveurosemonthosp...	Hospital	3	Eric Atelier	13	13	0
Jean Coutu Pharmacy	Montreal	QC	H3W 1Y3	514-555-6789	http://www.jeancoutupharmacy.com	Pharmacy	2	Nicholas Butler	1	0	0
Pharmaprix (Shoppers Drug Mart)	Montreal	QC	H3G 1P1	514-555-1234	http://www.pharmaprix.com	Pharmacy	2	Justin Grant	1	0	0
Montreal Neurological Hospital - Neurology Dep...	Montreal	QC	H3A 2B4	514-555-5678	http://www.montrealneurologicalhospit...	Special In...	2	Malwina Keating	3	0	3
Dentistry - Clinique Dentaire du Quartier Latin	Montreal	QC	H2X 3J6	514-555-9012	http://www.cliniquedentairequartierlatin...	Special In...	2	Keenan Ellast	2	2	0

- (9)

```
-- QUERY 9
SELECT
    E.FirstName, E.LastName, E.DateOfBirth, E.EMedicareCardNumber,
    E.PhoneNumber, E.AddressID, A.City, A.Province , A.PostalCode,
    E.Citizenship, E.EmailAddress,
    (
        SELECT COUNT(*)
        FROM Employee er
        WHERE er.ESocialSecurityNumber = E.ESocialSecurityNumber
        AND er.IsPrimaryResidence = FALSE
    ) AS NumOfSecondaryResidences
FROM Employee E
JOIN EmployeeAtFacility EF ON E.EmployeeID = EF.EmployeeID
JOIN Facility F ON EF.FacilityID = F.FacilityID
JOIN Address A ON E.AddressID = A.AddressID
WHERE F.FacilityID = 1 AND EndDate is NULL AND IsPrimaryResidence = TRUE
GROUP BY E.EmployeeID,EF.StartDate
HAVING NumOfSecondaryResidences >= 1
ORDER BY EF.StartDate DESC, E.FirstName, E.LastName;
```

## Result:

FirstName	LastName	DateOfBirth	EMedicareCardNum...	PhoneNumber	AddressID	City	Province	PostalCode	Citizenship	EmailAddress	NumOfSecondaryResidenc...
Leah	Castel	1998-09-29	850-959-939	+1 440-013-0392	13	Montreal	QC	H1V 1A1	Canadian	leah_c@gmail.com	4
Bethoven	Suprano	1993-03-04	466-329-312	+1 233-212-3857	16	Montreal	QC	H3G 1Z2	USA	b_suprano@gmail.com	3
Clergy	Suprano	1993-03-04	466-329-312	+1 233-212-3857	16	Montreal	QC	H3G 1Z2	USA	cs_suprano@gmail.com	3
Estaban	Suprano	1993-03-04	466-329-312	+1 233-212-3857	16	Montreal	QC	H3G 1Z2	USA	estaban_suprano@gmail.com	4
Just	Suprano	1993-03-04	466-329-312	+1 233-212-3857	16	Montreal	QC	H3G 1Z2	USA	j_suprano@gmail.com	3
Wesley	Gibbons	1997-10-20	920-359-394	+1 504-040-3042	14	Montreal	QC	H4A 2G7	Canadian	wesley_g@gmail.com	3
Esmerelda	Lutin	1999-03-10	300-038-000	+1 400-203-2077	22	Montreal	QC	H3Y 2T9	Canadian	Esmerelda@gmail.com	4
Laurel	Pollie	2000-05-31	498-394-093	+1 040-039-3592	15	Montreal	QC	H4A 2B2	Canadian	laurel_p@gmail.com	3
Justin	Grant	1995-02-03	959-938-941	+1 830-958-9829	19	Montreal	QC	H1A 5B6	Canadian	justin_g@gmail.com	3
Malwina	Keating	1994-03-04	949-329-312	+1 048-589-3857	20	Montreal	QC	H1L 2B3	Canadian	malwina_k@gmail.com	1

- (10)

```
-- Query 10
SELECT F.FacilityName, S.ScheduleDate, S.StartTime, S.EndTime
FROM WorkSchedule S
JOIN Facility F ON S.FacilityID = F.FacilityID
JOIN Employee E ON S.EmployeeID = E.EmployeeID
WHERE E.EmployeeID = 73
AND S.ScheduleDate BETWEEN '2024-04-10' AND '2024-04-25'
ORDER BY F.FacilityName ASC, S.ScheduleDate, S.StartTime;
```

Result:

FacilityName	ScheduleDate	StartTime	EndTime
Hospital Maisonneuve Rosemont	2024-04-10	09:00:00	12:00:00
Hospital Maisonneuve Rosemont	2024-04-15	13:00:00	17:00:00
Hospital Maisonneuve Rosemont	2024-04-16	10:00:00	14:00:00
Hospital Maisonneuve Rosemont	2024-04-20	08:30:00	12:30:00
Hospital Maisonneuve Rosemont	2024-04-25	09:30:00	13:30:00

- (11)

```
-- Query 11
SELECT
    MAX(E.FirstName) AS EmployeeFirstName,
    MAX(E.LastName) AS EmployeeLastName,
    MAX(A.City) AS AddressCity,
    MAX(A.Province) AS AddressProvince,
    MAX(A.PostalCode) AS AddressPostalCode,
    MAX(R.ResidenceType) AS ResidenceType,
    P.FirstName AS PersonFirstName,
    P.LastName AS PersonLastName,
    MAX(P.Occupation) AS Occupation,
    MAX(PAR.Relationship) AS Relationship,
    MAX(E.IsPrimaryResidence) AS EmployeePrimaryResidence
FROM Employee E
JOIN Address A ON E.AddressID = A.AddressID
JOIN Residence R ON A.AddressID = R.AddressID
JOIN PersonAtResidence PAR ON R.ResidenceID = PAR.ResidenceID
JOIN Person P ON PAR.PersonID = P.PersonID
WHERE E.FirstName = 'Leah' AND E.LastName = 'Castel'
GROUP BY P.FirstName, P.LastName
ORDER BY AddressCity, AddressProvince, AddressPostalCode, P.FirstName, P.LastName;
```

Result:

EmployeeFirstName	EmployeeLastName	AddressCity	AddressProvince	AddressPostalCode	ResidenceType	PersonFirstName	PersonLastName	Occupation	Relationship	EmployeePrimaryResiden...
Leah	Castel	Montreal	QC	H1V 1A1	Apartment	Mark	Castel	Economist	Partner	1
Leah	Castel	Montreal	QC	H1V 1A1	Apartment	Phil	Castel	Student	Children	1
Leah	Castel	Montreal	QC	H1V 1A1	Apartment	Sarah	Castel	Student	Children	1
Leah	Castel	Montreal	QC	H3G 1Z2	Condominium	Charlie	Versalle	Cashier	Roommate	0
Leah	Castel	Montreal	QC	H3G 1Z2	Condominium	Pepper	Versalle	Mechanic	Parent	0
Leah	Castel	Montreal	QC	H3G 1Z2	Condominium	Pollie	Barista	Non-Parent Dependent	0	
Leah	Castel	Montreal	QC	H4A 2B2	Apartment	Castel	Eve	Researcher	Roommate	0
Leah	Castel	Montreal	QC	H4A 2B2	Apartment	Pollie	Gibbons	Housewife	Non-Parent Dependent	0
Leah	Castel	Montreal	QC	H4A 2G7	Apartment	Gary	Connor	Waiter	Non-Parent Dependent	0
Leah	Castel	Montreal	QC	H4A 2G7	Apartment	Steve	Gibbons	Waiter	Non-Parent Dependent	0

- (12)

```
-- QUERY 12
SELECT E.FirstName, E.LastName, I.InfectionDate, F.FacilityName,
(
    SELECT COUNT(*)
    FROM Employee er
    WHERE er.ESocialSecurityNumber = E.ESocialSecurityNumber
    AND er.IsPrimaryResidence = FALSE
) AS NumOfSecondaryResidences
FROM Infection I
JOIN Employee E ON I.EmployeeID = E.EmployeeID
JOIN EmployeeAtFacility EF ON E.EmployeeID = EF.EmployeeID
JOIN Facility F ON EF.FacilityID = F.FacilityID
WHERE E.Occupation = 'Doctor' AND I.InfectionType = 'COVID-19'
AND I.InfectionDate BETWEEN DATE_SUB(NOW(), INTERVAL 2 WEEK) AND NOW()
ORDER BY F.FacilityName ASC, NumOfSecondaryResidences ASC;
```

Result:

FirstName	LastName	InfectionDa...	FacilityName	NumOfSecondaryResidenc...
Zackary	Smith	2024-04-08	Hospital Maisonneuve Rosemont	0
Jane	Smith	2024-04-08	Hospital Maisonneuve Rosemont	1
Wesley	Gibbons	2024-04-08	Jewish General Hospital	3
Wesley	Gibbons	2024-04-08	Montreal General Hospital	3
Sammy	Elden	2024-04-03	Vancouver General Hospital	0

- (13)

```
SELECT * FROM Log
WHERE EmailSubject LIKE 'Assignment Cancellation'
AND Sender = 'info@maisonneuverosemonthospital.com'
AND EmailDate BETWEEN '2024-04-10' AND '2024-04-12'
ORDER BY EmailDate DESC;
```

Result:

LogID	EmailDate	Sender	Receiver	EmailSubject	BodyPreview
116	2024-04-12	info@maisonneuverosemonthospital.com	employee4@example.com	Assignment Cancellation	Your scheduled shift has been cancelled. We ap...
117	2024-04-12	info@maisonneuverosemonthospital.com	employee5@example.com	Assignment Cancellation	Your scheduled shift has been cancelled. We ap...
114	2024-04-11	info@maisonneuverosemonthospital.com	employee2@example.com	Assignment Cancellation	Your scheduled shift has been cancelled. We ap...
115	2024-04-11	info@maisonneuverosemonthospital.com	employee3@example.com	Assignment Cancellation	Your scheduled shift has been cancelled. We ap...
113	2024-04-10	info@maisonneuverosemonthospital.com	employee1@example.com	Assignment Cancellation	Your scheduled shift has been cancelled. We ap...
NULL	NULL	NULL	NULL	NULL	NULL

- (14)

```
#14
SELECT e.ESocialSecurityNumber, e.FirstName, e.LastName, e.Occupation,
    (
        SELECT COUNT(*)
        FROM Employee er
        WHERE er.ESocialSecurityNumber = e.ESocialSecurityNumber
            AND er.IsPrimaryResidence = FALSE
    ) AS TotalNumOfSecondaryResidences
    FROM Employee e
    JOIN EmployeeAtFacility eaf ON eaf.EmployeeID = e.EmployeeID
    JOIN WorkSchedule ws ON eaf.EmployeeID = ws.EmployeeID
    WHERE eaf.FacilityID = 1
        AND ws.ScheduleDate >= DATE_SUB(CURDATE(), INTERVAL 4 WEEK)
        AND eaf.EndDate IS NULL
    GROUP BY e.ESocialSecurityNumber, e.FirstName, e.LastName, e.Occupation, TotalNumOfSecondaryResidences
    HAVING TotalNumOfSecondaryResidences >= 3
    ORDER BY e.Occupation;
```

Result:

ESocialSecurityNumber	FirstName	LastName	Occupation	TotalNumOfSecondaryResidences
MT4321-1236Q	Bethoven	Suprano	Doctor	3
MT4335-1236Q	Just	Suprano	Doctor	3
MT4898-1236Q	Clergy	Suprano	Doctor	3
MT9483-1236Q	Estaban	Suprano	Nurse	4
MT3333-1236Q	Concordia	Suprano	Nurse	3

- (15)

```
#15)
SELECT e.FirstName, e.LastName, e.DateOfBirth, e.EmailAddress,
infectionTotal.Total AS "Total Number of Covid-19 Infections",
MIN(eaf.StartDate) AS "First Day Worked",
v.DoseNumber AS "Total Vaccines Received",
(
    SELECT CONCAT(
        FLOOR(SUM(TIMESTAMPDIFF(MINUTE, StartTime, EndTime)) / 60), ' hour ',
        MOD(SUM(TIMESTAMPDIFF(MINUTE, StartTime, EndTime)), 60), ' minutes'
    )
    FROM WorkSchedule w
    WHERE w.EmployeeID = e.EmployeeID
) AS "Total Number of Hours Scheduled",
(
    SELECT COUNT(*)
    FROM Employee er
    WHERE er.ESocialSecurityNumber = e.ESocialSecurityNumber
        AND er.IsPrimaryResidence = FALSE
) AS TotalNumOfSecondaryResidences
FROM Employee e
JOIN (
    SELECT EmployeeID, COUNT(*) AS Total
    FROM Infection
    WHERE InfectionType = 'Covid-19' AND InfectionDate >= DATE_SUB(CURDATE(), INTERVAL 2 WEEK)
    GROUP BY EmployeeID
) infectionTotal ON e.EmployeeID = infectionTotal.EmployeeID
JOIN EmployeeAtFacility eaf ON e.EmployeeID = eaf.EmployeeID
JOIN WorkSchedule ws ON e.EmployeeID = ws.EmployeeID
JOIN Vaccination v ON e.EmployeeID = v.EmployeeID
WHERE e.Occupation = 'Nurse' AND eaf.EndDate IS NULL
GROUP BY e.EmployeeID, e.FirstName, e.LastName, v.DoseNumber
HAVING COUNT(DISTINCT eaf.FacilityID) >= 2
ORDER BY MIN(eaf.StartDate), e.FirstName, e.LastName;
```

## Result:

FirstName	LastName	DateOfBirth	EmailAddress	Total Number of Covid-19 Infections	First Day Worked	Total Vaccines Received	Total Number of Hours Scheduled	TotalNumOfSecondaryResidences
Malwina	Keating	1994-03-04	malwina_k@gmail.com	1	2014-10-03	2	1 hour 45 minutes	1
George	Vanier	1994-03-04	georgev@gmail.com	1	2019-02-02	3	6 hour 0 minutes	1
Laurel	Pollie	2000-05-31	laurel_p@gmail.com	1	2017-10-03	2	4 hour 0 minutes	3
Esmereilda	Lutin	1999-03-10	Esmerelda@gmail.com	1	2017-10-03	3	5 hour 45 minutes	4
Leah	Castel	1998-09-29	leah_c@gmail.com	1	2019-01-02	2	8 hour 0 minutes	4

- (16)

```
#16
SELECT e.Occupation,
COUNT(DISTINCT e.EmployeeID) AS "Total Number of Employees Currently Working",
COUNT(DISTINCT i.EmployeeID) AS "Total Number of Employees That Currently Have Covid-19"
FROM Employee e
JOIN EmployeeAtFacility eaf ON e.EmployeeID = eaf.EmployeeID
LEFT JOIN Infection i ON e.EmployeeID = i.EmployeeID AND i.InfectionType IN ('Covid-19')
AND i.InfectionDate >= DATE_SUB(CURDATE(), INTERVAL 2 WEEK)
WHERE eaf.EndDate IS NULL
GROUP BY e.Occupation;
```

Result:

Occupation	Total Number of Employees Currently Working	Total Number of Employees That Currently H...
Administrative Personnel	1	0
Doctor	13	4
Nurse	12	7
Pharmacist	3	1
Receptionist	2	0
Regular Employee	3	0

- (17)

```
#17
SELECT e.Occupation,
COUNT(DISTINCT e.EmployeeID) AS "Total Number of Employees Currently Working",
COUNT(DISTINCT i.EmployeeID) AS "Total Number of Employees That Never Caught Covid-19"
FROM Employee e
JOIN EmployeeAtFacility eaf ON e.EmployeeID = eaf.EmployeeID
LEFT JOIN Infection i ON e.EmployeeID = i.EmployeeID AND i.InfectionType NOT IN ('Covid-19')
WHERE eaf.EndDate IS NULL
GROUP BY e.Occupation;
```

Result:

Occupation	Total Number of Employees Currently Working	Total Number of Employees That Never Caught Covid-19
Administrative Personnel	1	0
Doctor	13	1
Nurse	12	2
Pharmacist	3	1
Receptionist	2	0
Regular Employee	3	0

- (18)

```
#18
SELECT COUNT(DISTINCT f.FacilityID) AS "total number of facilities", a.Province, COUNT(DISTINCT eaf.EmployeeID)
AS "Total Number of Employees Currently Working",
COUNT(DISTINCT i.EmployeeID) AS "Total Number of Working & Infected Employees by Covid-19",
(SELECT SUM(f2.FacilityCapacity)
FROM Facility f2
JOIN Address a2 ON f2.AddressID = a2.AddressID
WHERE a2.Province = a.Province) AS "Max Capacity of Total Province Facility" ,
(SELECT SUM(TIMESTAMPDIFF(HOUR, ws.StartTime, ws.EndTime))
FROM WorkSchedule ws
JOIN EmployeeAtFacility eaff ON ws.EmployeeID = eaff.EmployeeID
JOIN Facility f3 ON eaff.FacilityID = f3.FacilityID
JOIN Address a3 ON f3.AddressID = a3.AddressID
WHERE a3.Province = a.Province
AND ws.ScheduleDate = '2020-07-21'
AND eaff.EndDate IS NULL)AS "Total Hours Worked On 2020-07-21"
FROM Facility f
JOIN Address a ON a.AddressID = f.AddressID -- AND f.FacilityID <=15
LEFT JOIN Employee e ON e.FacilityID = f.FacilityID
LEFT JOIN Infection i ON e.EmployeeID = i.EmployeeID AND i.InfectionType IN ('Covid-19') AND i.InfectionDate >= DATE_SUB(CURDATE(), INTERVAL 2 WEEK)
LEFT JOIN EmployeeAtFacility eaf ON eaf.EmployeeID = e.EmployeeID AND eaf.EndDate IS NULL
GROUP BY a.Province
ORDER BY a.Province ;
```

### Result:

total number of facilities	Province	Total Number of Employees Currently Working	Total Number of Working & Infected Employees by Covid-19	Max Capacity of Total Province Facility	Total Hours Worked On 2020-07-21
1	AB	1	1	400	6
1	BC	0	0	500	6
1	MB	1	1	300	6
1	ON	2	1	2	10
11	QC	30	10	35	24

## Triggers Used

1. Only one manager allowed per facility

```
CREATE TRIGGER PreventMultipleManagers
BEFORE INSERT ON Employee
FOR EACH ROW
BEGIN
    DECLARE managerCount INT;

    SELECT COUNT(*)
    INTO managerCount
    FROM Employee
    WHERE FacilityID = NEW.FacilityID AND IsManager = TRUE;

    IF managerCount > 0 AND NEW.IsManager = TRUE THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Only one manager allowed per facility.';
    END IF;
END;
```

2. It checks if there are any existing schedules for the same employee on the same date that overlap with the new schedule's time frame. If any overlapping schedule is found, it raises an error using the SIGNAL statement to prevent the insertion of conflicting schedules.

```

CREATE TRIGGER trg_NoOverlappingSchedules BEFORE INSERT ON WorkSchedule
FOR EACH ROW
BEGIN
    IF EXISTS (
        SELECT 1
        FROM WorkSchedule ws1
        WHERE ws1.EmployeeID = NEW.EmployeeID
        AND ws1.ScheduleDate = NEW.ScheduleDate
        AND (
            (NEW.StartTime BETWEEN ws1.StartTime AND ws1.EndTime)
            OR (NEW.EndTime BETWEEN ws1.StartTime AND ws1.EndTime)
            OR (ws1.StartTime BETWEEN NEW.StartTime AND NEW.EndTime)
            OR (ws1.EndTime BETWEEN NEW.StartTime AND NEW.EndTime)
        )
        AND ws1.ScheduleID != NEW.ScheduleID
    ) THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Cannot insert overlapping schedules';
    END IF;
END;

```

3. This trigger ensures that there is a minimum gap of two hours between consecutive schedules for the same employee on the same date. It checks if there are any existing schedules that violate this rule and raises an error if such schedules are found.

```

CREATE TRIGGER trg_MinTwoHoursBetweenSchedules BEFORE INSERT ON WorkSchedule
FOR EACH ROW
BEGIN
    IF EXISTS (
        SELECT 1
        FROM WorkSchedule ws2
        WHERE ws2.EmployeeID = NEW.EmployeeID
        AND ws2.ScheduleDate = NEW.ScheduleDate
        AND (
            (NEW.StartTime BETWEEN ws2.StartTime AND ADDTIME(ws2.EndTime, '02:00:00'))
            OR (NEW.EndTime BETWEEN ws2.StartTime AND ADDTIME(ws2.EndTime, '02:00:00'))
            OR (ws2.StartTime BETWEEN NEW.StartTime AND ADDTIME(NEW.EndTime, '-02:00:00'))
            OR (ws2.EndTime BETWEEN NEW.StartTime AND ADDTIME(NEW.EndTime, '-02:00:00'))
        )
        AND ws2.ScheduleID != NEW.ScheduleID
    ) THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Minimum two hours required between schedules';
    END IF;
END;

```

4. This trigger ensures that the ScheduleDate falls within a four-week range starting from the current date.

```

CREATE TRIGGER trg_SupportFourWeeksAhead BEFORE INSERT ON WorkSchedule
FOR EACH ROW
BEGIN
    IF NEW.ScheduleDate > DATE_ADD(CURRENT_DATE, INTERVAL 28 DAY) THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Schedule date must be within four weeks ahead of the current date';
    END IF;
END //

DROP TRIGGER IF EXISTS trg_PreventInfectedScheduling;

```

5. This trigger checks if the employee has been infected with COVID-19 within the last two weeks before the scheduled date. If infected, it prevents scheduling.

```

CREATE TRIGGER trg_PreventInfectedScheduling BEFORE INSERT ON WorkSchedule
FOR EACH ROW
BEGIN
    DECLARE is_infected BOOLEAN;
    DECLARE emp_occupation VARCHAR(50);

    -- Get the occupation of the employee
    SELECT Occupation INTO emp_occupation
    FROM Employee
    WHERE EmployeeID = NEW.EmployeeID;

    -- Check if the employee is a nurse or doctor
    IF emp_occupation IN ('Nurse', 'Doctor') THEN
        -- Check if the employee is infected
        SELECT TRUE INTO is_infected
        FROM Infection
        WHERE EmployeeID = NEW.EmployeeID
        AND InfectionType IN ('COVID-19')
        AND NEW.ScheduleDate <= DATE_ADD(InfestationDate, INTERVAL 14 DAY)
        AND NEW.ScheduleDate >= InfestationDate
        LIMIT 1;

        -- If infected, prevent scheduling
        IF is_infected THEN
            SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Nurses and doctors infected with COVID-19 cannot be scheduled for work';
        END IF;
    END IF;
END //

```

6. This trigger checks if the employee has been vaccinated with at least one vaccine for COVID-19 within the last six months before the scheduled date. If not vaccinated, it prevents scheduling.

```

CREATE TRIGGER trg_RequireVaccination BEFORE INSERT ON WorkSchedule
FOR EACH ROW
BEGIN
    DECLARE not_vaccinated BOOLEAN;

    SELECT TRUE INTO not_vaccinated
    FROM Vaccination
    WHERE EmployeeID = NEW.EmployeeID
    AND VaccineType IN ('Pfizer', 'Moderna', 'AstraZeneca', 'Johnson & Johnson')
    AND (NEW.ScheduleDate > DATE_ADD(VaccinationDate, INTERVAL 6 MONTH)
    OR NEW.ScheduleDate < VaccinationDate)
    LIMIT 1;

    IF not_vaccinated THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Employee must be vaccinated to be scheduled for work';
    END IF;
END //

```

7. This trigger will cancel all assignments for an infected employee for two weeks from the date of infection.

```

CREATE TRIGGER trg_CancelAssignments AFTER INSERT ON Infection
FOR EACH ROW
BEGIN
    DECLARE infected_employee_id INT;

    IF NEW.InfectionType = 'COVID-19' AND NEW.EmployeeID IS NOT NULL THEN
        SELECT EmployeeID INTO infected_employee_id FROM Employee WHERE EmployeeID = NEW.EmployeeID;

        -- Cancel assignments for the infected employee for two weeks
        DELETE FROM WorkSchedule
        WHERE EmployeeID = NEW.EmployeeID
        AND ScheduleDate <= DATE_ADD(NEW.InfectionDate, INTERVAL 14 DAY)
        AND ScheduleDate >= NEW.InfectionDate;
    END IF;
END //

```

8. This trigger will send warning emails to colleagues who have been in contact with the infected employee.

```

CREATE TRIGGER trg_SendWarningEmails AFTER INSERT ON Infection
FOR EACH ROW
BEGIN
    DECLARE sender_facility VARCHAR(255);

    SELECT EmailAddress INTO sender_facility
    FROM Facility f INNER JOIN EmployeeAtFacility eaf ON f.FacilityID = eaf.FacilityID
    WHERE eaf.EmployeeID = NEW.EmployeeID
    LIMIT 1;

    IF NEW.InfectionType = 'COVID-19' AND NEW.EmployeeID IS NOT NULL THEN
        -- Get colleagues who worked with the infected employee in the past two weeks
        INSERT INTO EmailQueue (EmailAddress, Subject, Body)
        SELECT DISTINCT e.EmailAddress, 'Warning', CONCAT('One of your colleagues with whom you worked in the past two weeks has been infected with COVID-19.') AS Body
        FROM WorkSchedule ws
        INNER JOIN Employee e ON ws.EmployeeID = e.EmployeeID
        WHERE ScheduleDate >= DATE_SUB(NEW.InfectionDate, INTERVAL 14 DAY)
        AND ScheduleDate <= NEW.InfectionDate;

        -- Send Email about the close contact of other employees with infected employee
        INSERT INTO Log (EmailDate, Sender, Receiver, EmailSubject, BodyPreview)
        SELECT DISTINCT CURRENT_DATE AS EmailDate, sender_facility AS Sender, e.EmailAddress AS Receiver, 'Warning' AS EmailSubject,
        CONCAT('One of your colleagues with whom you worked in the past two weeks has been infected with COVID-19.') AS BodyPreview
        FROM WorkSchedule ws
        INNER JOIN Employee e ON ws.EmployeeID = e.EmployeeID
        WHERE ScheduleDate >= DATE_SUB(NEW.InfectionDate, INTERVAL 14 DAY)
        AND ScheduleDate <= NEW.InfectionDate;
    END IF;
END;

```

9. This trigger will send an email to the infected employee about the cancellation of assignments.

```

CREATE TRIGGER trg_SendCancellationEmail AFTER INSERT ON Infection
FOR EACH ROW
BEGIN
    DECLARE infected_email VARCHAR(255);
    DECLARE affected_addressid VARCHAR(255);
    DECLARE sender_facility VARCHAR(255);

    SELECT EmailAddress INTO sender_facility
    FROM Facility f INNER JOIN EmployeeAtFacility eaf ON f.FacilityID = eaf.FacilityID
    WHERE eaf.EmployeeID = NEW.EmployeeID
    LIMIT 1;

    IF NEW.InfectionType = 'COVID-19' AND NEW.EmployeeID IS NOT NULL THEN -- Only trigger if COVID-19 Infection from an Employee
        -- Get the infected employee's email address
        SELECT EmailAddress INTO infected_email FROM Employee WHERE EmployeeID = NEW.EmployeeID;

        -- Send email about the cancellation of assignments to the infected employee
        INSERT INTO EmailQueue (EmailAddress, Subject, Body)
        VALUES (infected_email, 'Assignment Cancellation', 'Your assignments have been cancelled due to COVID-19 infection. Please contact your manager for further instructions.');

        INSERT INTO Log (EmailDate, Sender, Receiver, EmailSubject, BodyPreview)
        VALUES (CURRENT_DATE, sender_facility, infected_email, 'Assignment Cancellation', 'Your assignments have been cancelled due to COVID-19 infection. Please contact your manager for further instructions.');
    END IF;
END;

```

## Additional Features

- Weekly Sunday Schedule Email Blast

```
-- Event for Sunday Weekly Email Blast for Schedule
DROP EVENT IF EXISTS event_SundayEmailBlast;
// 

-- SET GLOBAL event_scheduler = ON;
SHOW processlist; //

CREATE EVENT event_SundayEmailBlast
ON SCHEDULE
-- Actual:
EVERY 7 DAY STARTS DATE '2024-04-09' ENDS DATE '2024-05-01'
-- Demo Purposes:
-- EVERY 1 MINUTE STARTS CURRENT_TIMESTAMP ENDS DATE '2024-04-12'

DO
BEGIN
    DECLARE start_date VARCHAR(255);
    DECLARE end_date VARCHAR(255);
    DECLARE subj VARCHAR(255);

    SELECT DATE_FORMAT(CURRENT_DATE + INTERVAL 1 DAY, "%W %d-%b-%Y") INTO start_date;
    SELECT DATE_FORMAT(start_date + INTERVAL 7 DAY, "%W %d-%b-%Y") INTO end_date;

    INSERT INTO EmailQueue (EmailAddress, Subject, Body)
    SELECT e.EmailAddress, CONCAT(f.FacilityName, " Schedule for ", start_date, " to ", end_date) AS "Subject",
        COALESCE(CONCAT(f.FacilityName, " ", a.PostalCode, " ", a.City, " ", a.Province, CHAR(10 using utf8mb4),
            e.Occupation, " ", e.FirstName, " ", e.LastName, " ", e.EmailAddress, CHAR(10 using utf8mb4),
            GROUP_CONCAT(ws.ScheduleDate SEPARATOR " | "), CHAR(10 using utf8mb4), GROUP_CONCAT(ws.StartTime SEPARATOR " | "),
            CHAR(10 using utf8mb4), GROUP_CONCAT(ws.EndTime SEPARATOR " | ")), "No Assignment") AS "Body"
    FROM Employee e
    JOIN EmployeeAtFacility eaf ON e.EmployeeID = eaf.EmployeeID
    JOIN Facility f ON eaf.FacilityID = f.FacilityID
    JOIN Address a ON f.AddressID = a.AddressID
    LEFT JOIN WorkSchedule ws ON e.EmployeeID = ws.EmployeeID
    WHERE (ws.ScheduleDate >= CURRENT_DATE) AND (ws.ScheduleDate <= CURRENT_DATE + INTERVAL 7 DAY)
    GROUP BY e.EmailAddress;

    INSERT INTO Log (EmailDate, Sender, Receiver, EmailSubject, BodyPreview)
    SELECT CURRENT_DATE AS "EmailDate", f.EmailAddress AS "Sender", e.EmailAddress AS "Receiver",
        CONCAT(f.FacilityName, " Schedule for ", start_date, " to ", end_date) AS "EmailSubject",
        SUBSTRING(COALESCE(CONCAT(f.FacilityName, " ", a.PostalCode, " ", a.City, " ", a.Province, CHAR(10 using utf8mb4),
            e.Occupation, " ", e.FirstName, " ", e.LastName, " ", e.EmailAddress, CHAR(10 using utf8mb4),
            GROUP_CONCAT(ws.ScheduleDate SEPARATOR " | "), CHAR(10 using utf8mb4), GROUP_CONCAT(ws.StartTime SEPARATOR " | "),
            CHAR(10 using utf8mb4), GROUP_CONCAT(ws.EndTime SEPARATOR " | ")), "No Assignment"), 1, 100) AS "BodyPreview"
    FROM Employee e
    JOIN EmployeeAtFacility eaf ON e.EmployeeID = eaf.EmployeeID
    JOIN Facility f ON eaf.FacilityID = f.FacilityID
    JOIN Address a ON f.AddressID = a.AddressID
    LEFT JOIN WorkSchedule ws ON e.EmployeeID = ws.EmployeeID
    WHERE (ws.ScheduleDate >= CURRENT_DATE) AND (ws.ScheduleDate <= CURRENT_DATE + INTERVAL 7 DAY)
    GROUP BY e.EmailAddress;
END;
//
```

- Snapshots of the web interface

## **Database application system for employees of the health facilities**

### **Insert new Facility**

Facility Name:

Facility Type:

Address ID:

Phone Number:

Email Address:

Facility Capacity:

Web Address:

Manager ID:

### **Insert new Employee**

First Name:

Last Name:

Date of Birth:

Social Security Number:

Medicare Card Number:

Phone Number:

Address ID:

Email Address:

Occupation:

Facility ID:

Is Manager:

Is Primary Residence:

Citizenship:

- Table display

Connected successfully to database: wjc353\_4

Employee ID	First Name	Last Name	Date of Birth	Social Security Number	Medicare Card Number	Phone Number	Address ID	Email Address	Occupation	Facility ID	Is Manager	Is Primary Residence	Citizenship
1	Leah	Castel	1998-09-29	MT0393-2323Q	850-959-939	+1 440-013-0392	13	leah_c@gmail.com	Nurse	1	1	1	Canadian
2	Wesley	Gibbons	1997-10-20	MT0239-2930Q	920-359-394	+1 504-040-3042	14	wesley_g@gmail.com	Doctor	2	1	1	Canadian
3	Laurel	Pollie	2000-05-31	MT0384-3920Q	498-394-093	+1 040-039-3592	15	laurel_p@gmail.com	Nurse	3	1	1	Canadian
4	Frederic	Versalle	1999-02-10	MT0830-2930Q	590-038-589	+1 949-203-2039	16	frederic_v@gmail.com	Pharmacist	4	1	1	Canadian
5	Walsh	Connor	2001-01-05	MT0234-3920Q	930-322-384	+1 043-302-9596	17	walsh_c@gmail.com	Receptionist	5	1	1	Canadian
6	Asher	Feah	2000-04-20	MT0394-2302Q	042-830-849	+1 554-030-3953	18	asher_f@gmail.com	Administrative Personnel	6	1	1	Canadian
7	Nicholas	Butler	2000-04-20	MT0232-1123Q	080-039-932	+1 554-030-3953	18	nicholas_b@gmail.com	Security Personnel	7	1	1	Canadian
8	Justin	Grant	1995-02-03	MT0923-2934Q	959-938-941	+1 830-958-9829	19	justin_g@gmail.com	Regular Employee	8	1	1	Canadian
9	Malwina	Keating	1994-03-04	MT9594-3920Q	949-329-312	+1 048-589-3857	20	malwina_k@gmail.com	Nurse	9	1	1	Canadian
10	Keenan	Ellast	1993-04-05	MT0934-9239Q	239-392-003	+1 438-949-0394	21	keenan_e@gmail.com	Doctor	10	1	1	Canadian
11	Eliane	Champagne	1999-02-10	MT7836-2930Q	591-038-000	+1 949-203-2077	16	Eliane_v@gmail.com	Pharmacist	2	0	1	Canadian

- Queries 8-18

### Test Queries

[Execute Query 8](#) [Execute Query 9](#) [Execute Query 10](#) [Execute Query 11](#) [Execute Query 12](#) [Execute Query 13](#) [Execute Query 14](#) [Execute Query 15](#) [Execute Query 16](#) [Execute Query 17](#)  
[Execute Query 18](#)

- Table display example for query 8:

Connected successfully to database: wjc353\_4

Facility Name	City	Province	Postal Code	Phone Number	Web Address	Facility Type	Facility Capacity	General Manager	Number of Employees	Number of Doctors	Number of Nurses
Calgary Medical Centre	Calgary	AB	T2P 0B2	+1 403-944-1110	www.calgarymedicalcentre.ca	Hospital	400	Eliane Suprano	1	0	1
Vancouver General Hospital	Vancouver	BC	V5K 1W1	+1 604-875-4111	www.vancouvergeneralhospital.ca	Hospital	500	Sammy Elden	1	1	0
Health Sciences Centre Winnipeg	Winnipeg	MB	R3C 1B2	+1 204-787-3661	www.hsc.mb.ca	Hospital	300	Concordia Suprano	1	0	1
Toronto Western Hospital	Toronto	ON	M5T 2S8	416-555-3456	http://www.torontowesternhospital.com	Hospital	2	Esmerelda Lutun	3	0	3
Medisys Executive Health Clinic	Montreal	QC	H3A 3M8	514-555-7890	http://www.medisysclinic.com	Clinic	3	Walsh Connor	1	0	0
Montreal Walk-In Clinic - Queen Elizabeth Health Complex	Montreal	QC	H4A 3L5	514-555-2345	http://www.montrealwalkinclinic.com	Clinic	3	Asher Feah	1	0	0
CLSC Metro	Montreal	QC	H2E 1B4	514-555-9012	http://www.clsclongueuil.com	CLSC	4	Laurel Pollie	3	0	3
CLSC de la Petite-Patrie	Montreal	QC	H2A 1M4	514-555-3456	http://www.clsclapetitepatrie.com	CLSC	4	Frederic Versalle	2	0	0
Montreal General Hospital	Montreal	QC	H3G 1A4	514-555-1234	http://www.montrealgeneralhospital.com	Hospital	5	Leah Castel	11	0	11
test5	Montreal	QC	H3G 1A4	1	1	hospital	5	Just Suprano	0	1	0
Jewish General Hospital	Montreal	QC	H3T 1E2	514-555-5678	http://www.jewishgeneralhospital.com	Hospital	5	Wesley Gibbons	6	6	0

## **Contributions**

Alex

- Queries 8-13
- Web User Interface Development and Implementation

Eliza

- Queries 14-18

Hui

- E/R Diagram
- Create tables
- Analyzing relations in the database is 3NF and BCNF
- Triggers for 1 to 7

Wen Jie (Luke)

- Insertion of Dummy Data
- Triggers for EmailQueue, Log insertion
- Event Scheduler
- Testing of Integrity Constraints of tables