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 citation1. 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.

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

Course Number: COMP 5531 **Instructor:** Khaled Jababo

Name: Alexander Newman I.D. # 40183409 Signature: Alex Newman

Name: Emma Langlois I.D. # 40254315 Signature: Emma Langlois

Name: Eric Spensieri I.D. # 26997252 Signature:

Date: 11/04/23

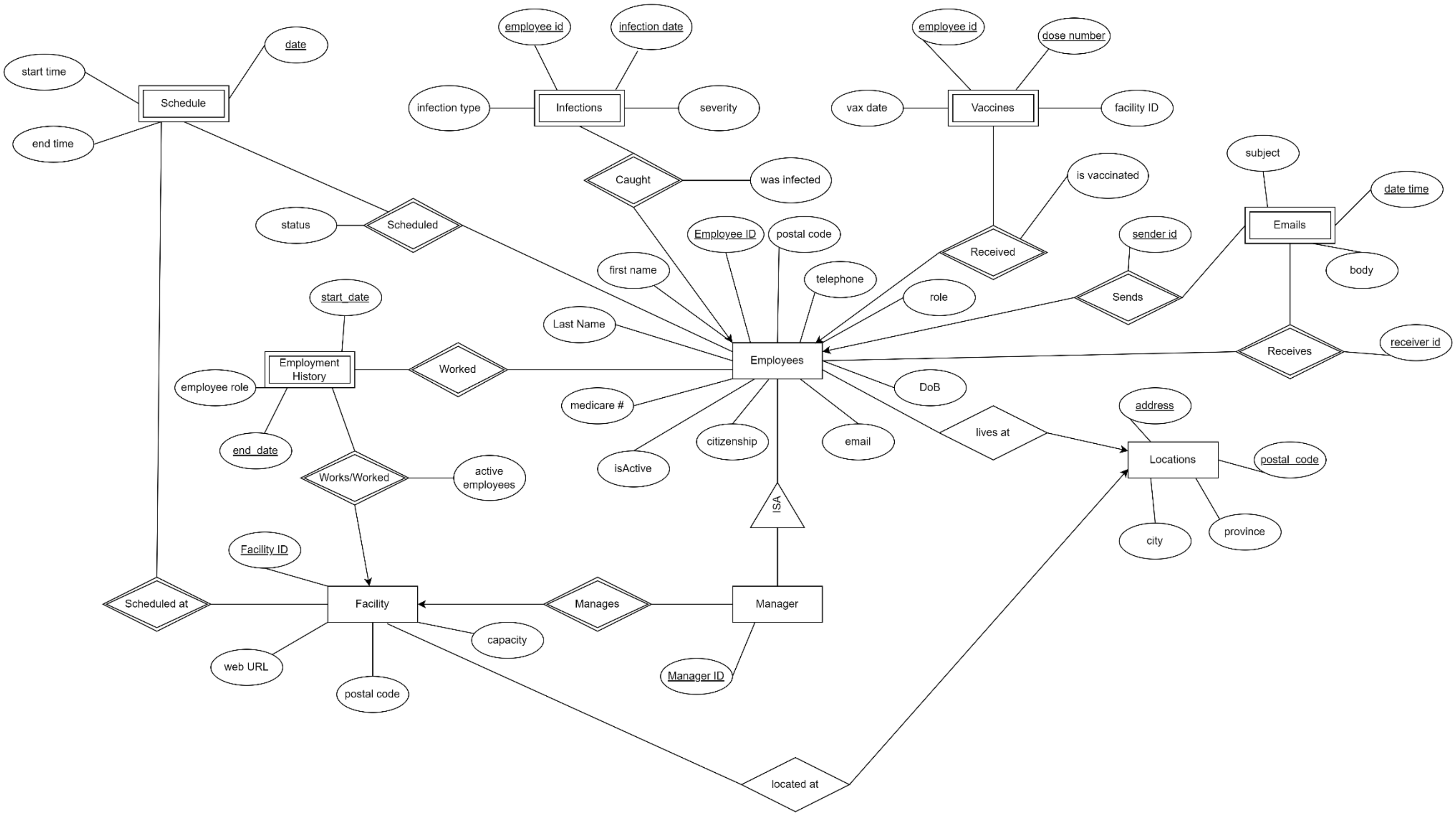
Group account: occ55314

5531 Main Project Report **Submitted April 11, 2023**

Alex Newman
Emma Langlois (#40254315)
Eric Spensieri (#26997252)

Main Project Report

E/R Diagram



Database Architecture

In order to normalize our database from the warmup project, we had to remove some redundancy as well as decompose our employees relation as follows:

Original Relation

Employees (employee_ID, manager_ID, facility_ID, first_name, last_name, isActive, employee_role, isVaccinated, wasInfected, email, telephone, date_of_birth, medicare_number, citizenship, postal_code, address, city, province)

FD's : {

employee_ID -> first_name, last_name, isActive, employee_role, isVaccinated, wasInfected, email, telephone, date_of_birth, medicare_number, citizenship, postal_code

medicare_number -> first_name, last_name, isActive, employee_role, isVaccinated, wasInfected, email, telephone, date_of_birth, employee_ID, citizenship, postal_code, address, city, province

postal_code -> address, city, province }

The postal code relation would put this table in 2NF since it is a transitive dependency with non-prime attributes on the left and right hand sides. In order to normalize this table, we decomposed the employee table into two relations and removed some redundant attributes. See below for the conversion of our diagram to relations and functional dependencies using the E/R approach:

New Relations

Employees (employee_ID, first_name, last_name, isActive, employee_role, isVaccinated, wasInfected, email, telephone, date_of_birth, medicare_number, citizenship, postal_code)

FD's : {

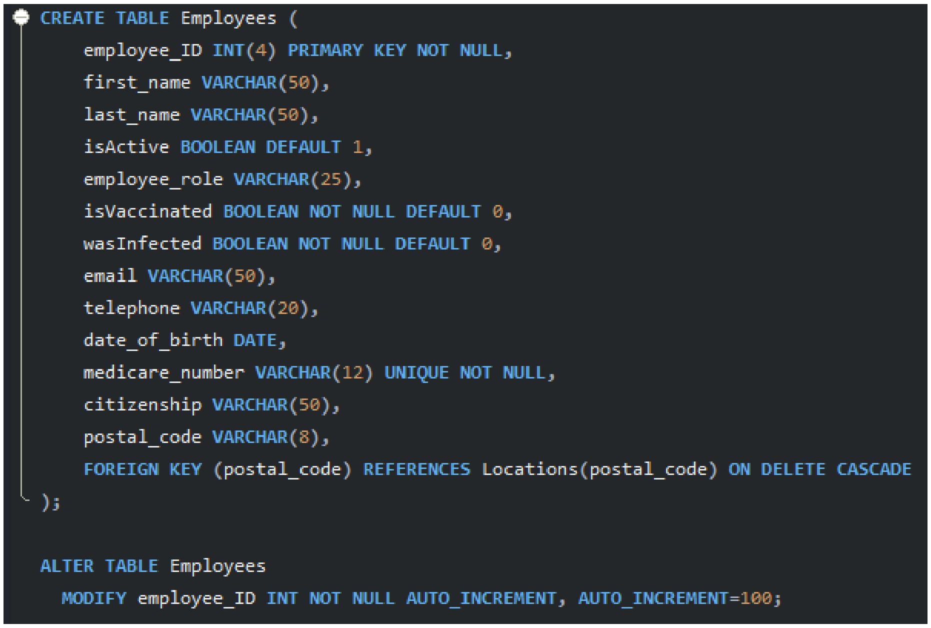
employee_ID -> first_name, last_name, isActive, employee_role, isVaccinated, wasInfected, email, telephone, date_of_birth, medicare_number, citizenship, postal_code ***BCNF***

medicare_number -> first_name, last_name, isActive, employee_role, isVaccinated, wasInfected, email, telephone, date_of_birth, employee_ID, citizenship, postal_code ***BCNF***

}

Keys: employee_ID, medicare_number

Foreign keys: postal_code



```
CREATE TABLE Employees (
    employee_ID INT(4) PRIMARY KEY NOT NULL,
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    isActive BOOLEAN DEFAULT 1,
    employee_role VARCHAR(25),
    isVaccinated BOOLEAN NOT NULL DEFAULT 0,
    wasInfected BOOLEAN NOT NULL DEFAULT 0,
    email VARCHAR(50),
    telephone VARCHAR(20),
    date_of_birth DATE,
    medicare_number VARCHAR(12) UNIQUE NOT NULL,
    citizenship VARCHAR(50),
    postal_code VARCHAR(8),
    FOREIGN KEY (postal_code) REFERENCES Locations(postal_code) ON DELETE CASCADE
);

ALTER TABLE Employees
  MODIFY employee_ID INT NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=100;
```

Locations (<u>postal_code</u>, address, city, province)

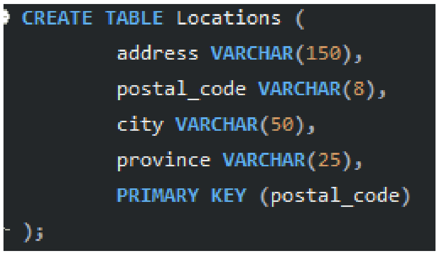
FD's: {

postal_code -> address, city, province ***BCNF***

}

Keys: <u>postal_code</u>

Foreign keys: N/A



```
CREATE TABLE Locations (
        address VARCHAR(150),
        postal_code VARCHAR(8),
        city VARCHAR(50),
        province VARCHAR(25),
        PRIMARY KEY (postal_code)
);
```

Facilities(<u>facility_ID</u>, facility_name, type, active_employees, postal_code, capacity, web_URL)

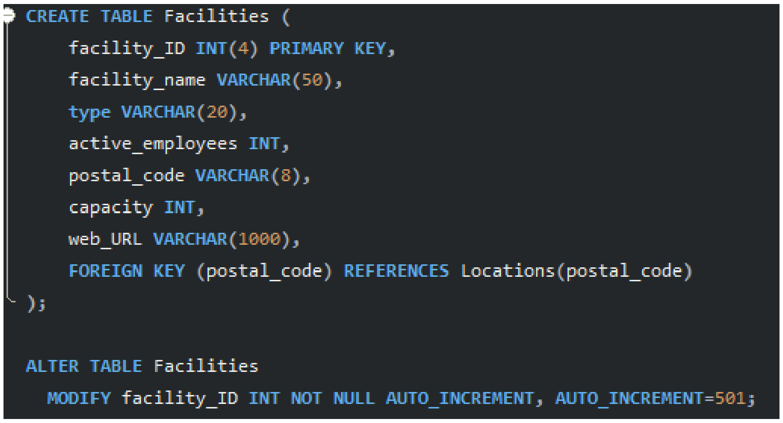
FD's: {

facility_ID -> facility_name, type, active_employees, postal_code, capacity, web_URL ***BCNF***

}

Keys: facility_ID

Foreign keys: postal_code



```sql
CREATE TABLE Facilities (
    facility_ID INT(4) PRIMARY KEY,
    facility_name VARCHAR(50),
    type VARCHAR(20),
    active_employees INT,
    postal_code VARCHAR(8),
    capacity INT,
    web_URL VARCHAR(1000),
    FOREIGN KEY (postal_code) REFERENCES Locations(postal_code)
);

ALTER TABLE Facilities
  MODIFY facility_ID INT NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=501;
```

Infections (employee_ID, nature_of_infection, date_of_infection, severity)

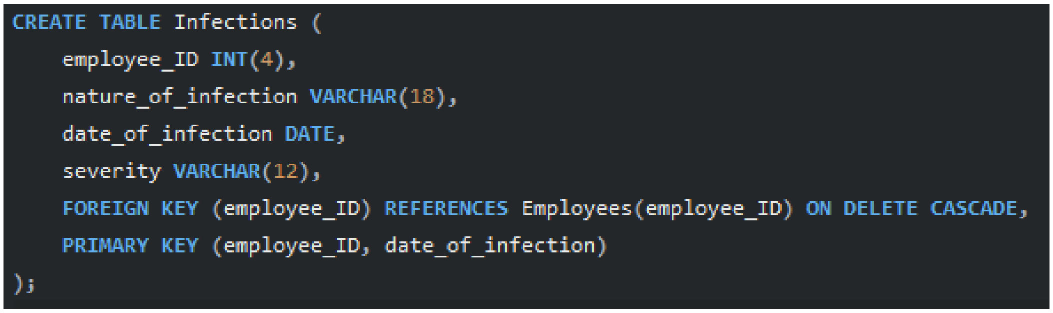
FD's: {

employee_ID, date_of_infection -> nature_of_infection, severity ***BCNF***

}

Candidate Keys: (employee_ID, date_of_infection)

Foreign Keys: employee_ID



```sql
CREATE TABLE Infections (
    employee_ID INT(4),
    nature_of_infection VARCHAR(18),
    date_of_infection DATE,
    severity VARCHAR(12),
    FOREIGN KEY (employee_ID) REFERENCES Employees(employee_ID) ON DELETE CASCADE,
    PRIMARY KEY (employee_ID, date_of_infection)
);
```

Employment_History(employee_ID, facility_ID, employee_role, start_date, end_date)

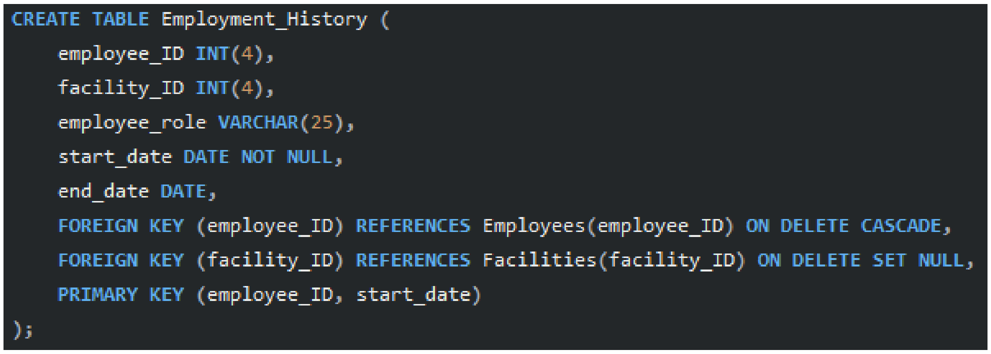
FD's : {

employee_ID, start_date -> facility_ID, employee_role, end_date ***BCNF***

}

Candidate Keys: (employee_ID, start_date)

Foreign keys: employee_ID, facility_ID



```
CREATE TABLE Employment_History (
    employee_ID INT(4),
    facility_ID INT(4),
    employee_role VARCHAR(25),
    start_date DATE NOT NULL,
    end_date DATE,
    FOREIGN KEY (employee_ID) REFERENCES Employees(employee_ID) ON DELETE CASCADE,
    FOREIGN KEY (facility_ID) REFERENCES Facilities(facility_ID) ON DELETE SET NULL,
    PRIMARY KEY (employee_ID, start_date)
);
```

Vaccines(employee_ID, type, vax_date, facility_id, dose_number)

FD's : {

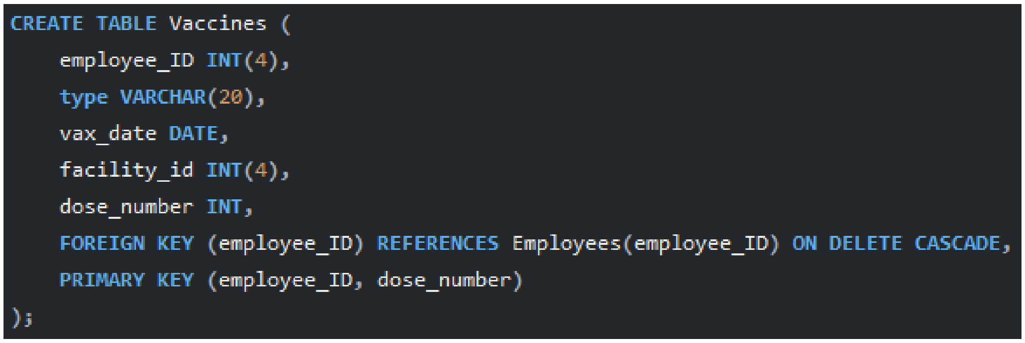
employee_ID, dose_number -> type, vax_date, facility_id ***BCNF***

}

Candidate keys:

(employee_ID, dose_number)

Foreign keys: employee_ID, facility_id



```
CREATE TABLE Vaccines (
    employee_ID INT(4),
    type VARCHAR(20),
    vax_date DATE,
    facility_id INT(4),
    dose_number INT,
    FOREIGN KEY (employee_ID) REFERENCES Employees(employee_ID) ON DELETE CASCADE,
    PRIMARY KEY (employee_ID, dose_number)
);
```

Managers(manager_ID, facility_ID)

FD's: {

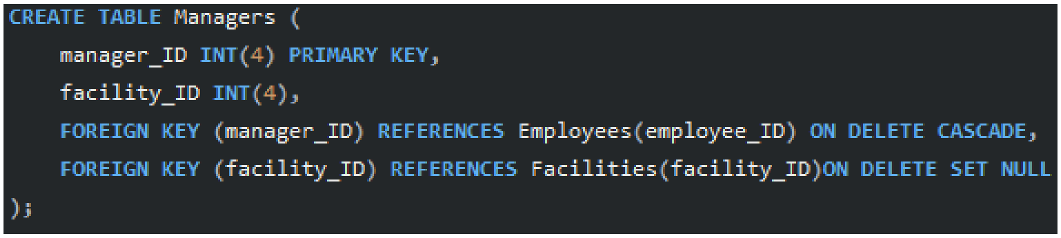
manager_ID -> facility_ID ***BCNF***

facility_ID -> manager_ID ***BCNF***

}

Candidate keys: manager_ID, facility_ID

Foreign keys: manager_ID (from employee_ID), facility_ID



```
CREATE TABLE Managers (
    manager_ID INT(4) PRIMARY KEY,
    facility_ID INT(4),
    FOREIGN KEY (manager_ID) REFERENCES Employees(employee_ID) ON DELETE CASCADE,
    FOREIGN KEY (facility_ID) REFERENCES Facilities(facility_ID)ON DELETE SET NULL
);
```

Emails(receiver_ID, date_Time, sender_ID, subject, body)

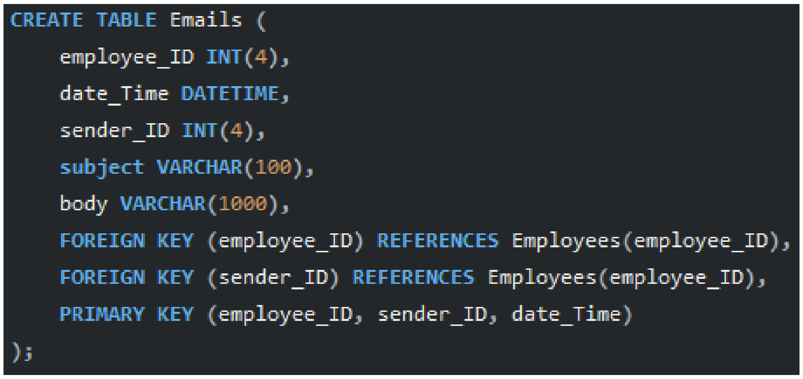
FD's: {

receiver_ID, sender_ID, date_Time -> subject, body ***BCNF***

}

Candidate keys: (receiver_ID_ID, sender_ID, date_Time)

Foreign keys: receiver_ID, sender_ID (both from employee_id)



```
CREATE TABLE Emails (
    employee_ID INT(4),
    date_Time DATETIME,
    sender_ID INT(4),
    subject VARCHAR(100),
    body VARCHAR(1000),
    FOREIGN KEY (employee_ID) REFERENCES Employees(employee_ID),
    FOREIGN KEY (sender_ID) REFERENCES Employees(employee_ID),
    PRIMARY KEY (employee_ID, sender_ID, date_Time)
);
```

Schedule(employee_ID, start_time, end_time, date, facility_ID, status)

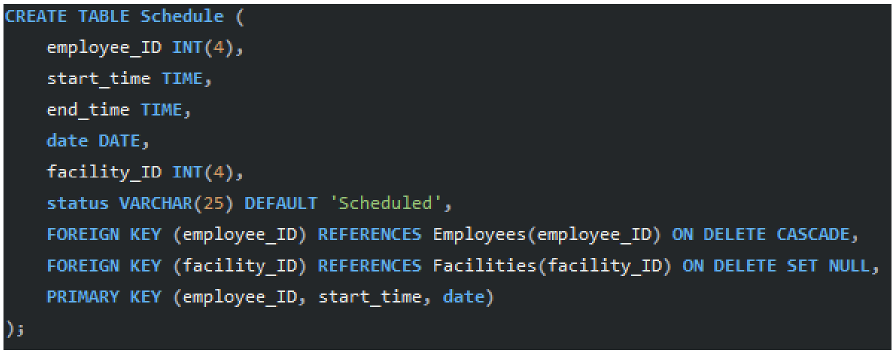
FD's: {

employee_ID, start_time, date -> end_time, facility_ID, status ***BCNF***

}

Candidate keys:(employee_ID, start_time, date)

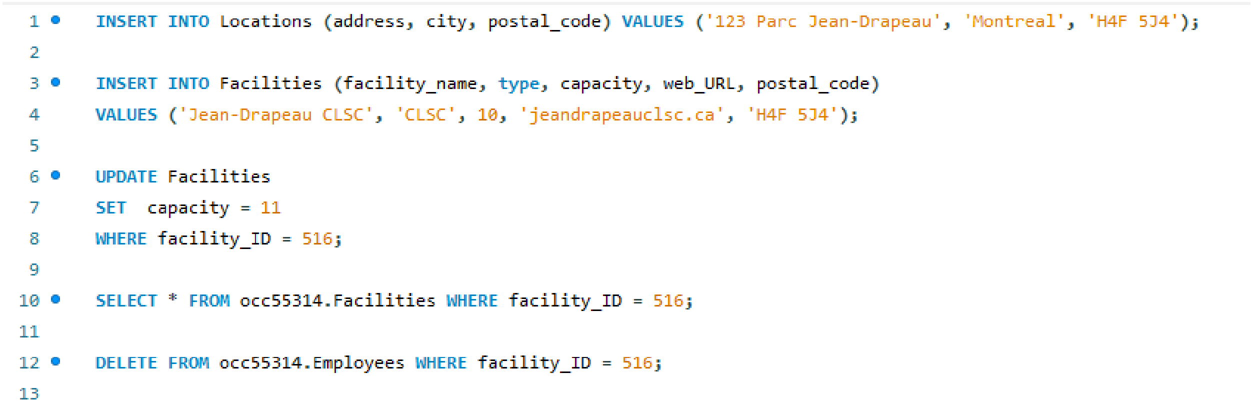
Foreign keys: employee_ID, facility_ID



```sql
CREATE TABLE Schedule (
    employee_ID INT(4),
    start_time TIME,
    end_time TIME,
    date DATE,
    facility_ID INT(4),
    status VARCHAR(25) DEFAULT 'Scheduled',
    FOREIGN KEY (employee_ID) REFERENCES Employees(employee_ID) ON DELETE CASCADE,
    FOREIGN KEY (facility_ID) REFERENCES Facilities(facility_ID) ON DELETE SET NULL,
    PRIMARY KEY (employee_ID, start_time, date)
);
```

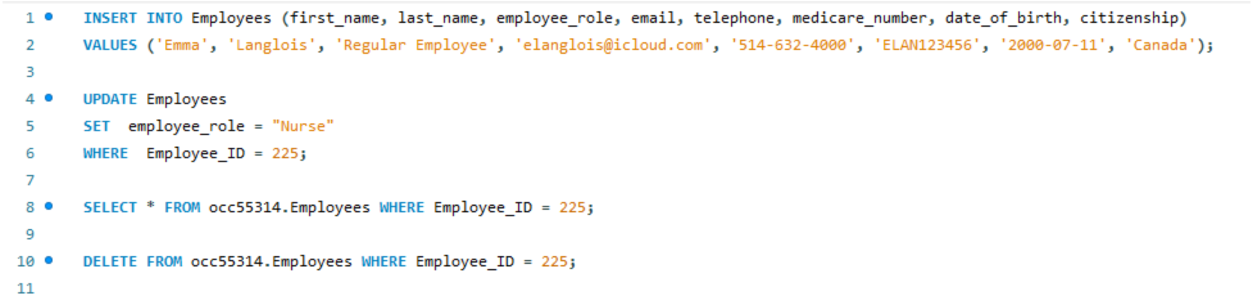
Queries:

1.



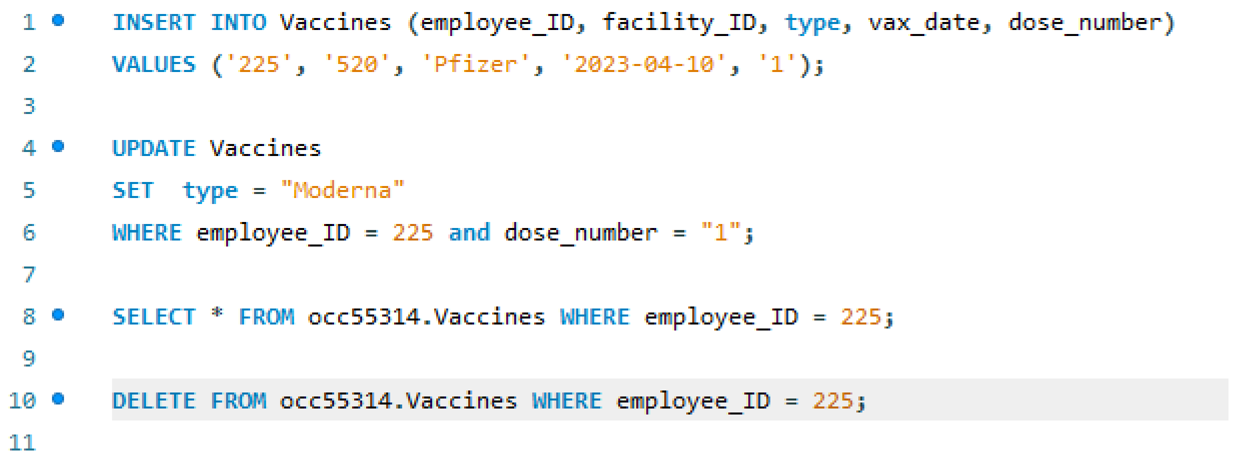
```
INSERT INTO Locations (address, city, postal_code) VALUES ('123 Parc Jean-Drapeau', 'Montreal', 'H4F 5J4');

INSERT INTO Facilities (facility_name, type, capacity, web_URL, postal_code)
VALUES ('Jean-Drapeau CLSC', 'CLSC', 10, 'jeandrapeauclsc.ca', 'H4F 5J4');

UPDATE Facilities
SET  capacity = 11
WHERE facility_ID = 516;

SELECT * FROM occ55314.Facilities WHERE facility_ID = 516;

DELETE FROM occ55314.Employees WHERE facility_ID = 516;
```

2.



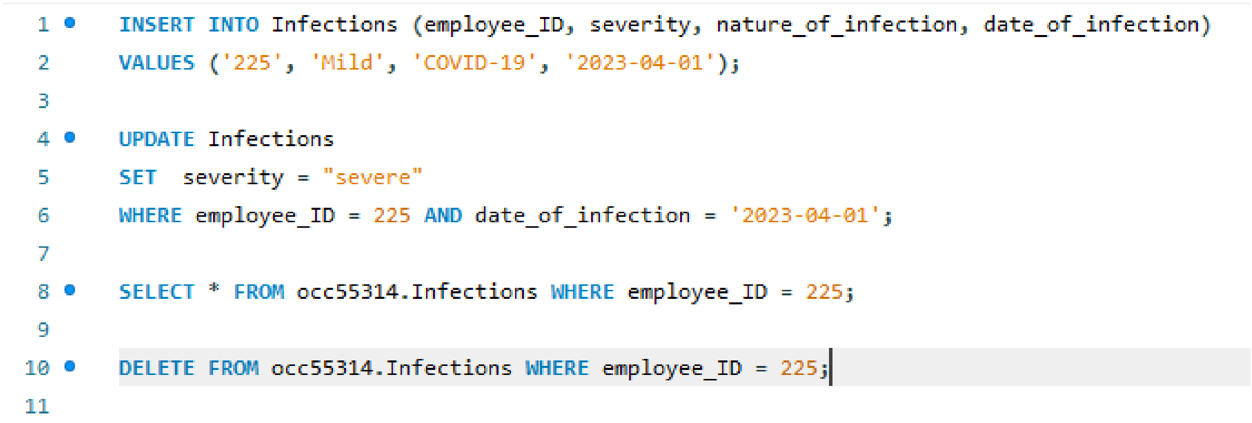
```
INSERT INTO Employees (first_name, last_name, employee_role, email, telephone, medicare_number, date_of_birth, citizenship)
VALUES ('Emma', 'Langlois', 'Regular Employee', 'elanglois@icloud.com', '514-632-4000', 'ELAN123456', '2000-07-11', 'Canada');

UPDATE Employees
SET  employee_role = "Nurse"
WHERE  Employee_ID = 225;

SELECT * FROM occ55314.Employees WHERE Employee_ID = 225;

DELETE FROM occ55314.Employees WHERE Employee_ID = 225;
```

3.



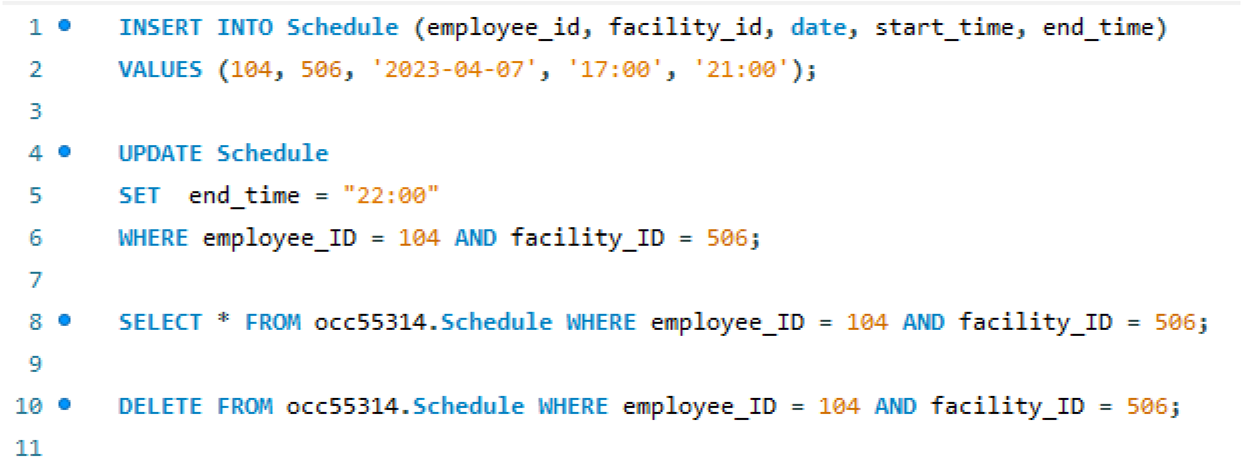
```
INSERT INTO Vaccines (employee_ID, facility_ID, type, vax_date, dose_number)
VALUES ('225', '520', 'Pfizer', '2023-04-10', '1');

UPDATE Vaccines
SET  type = "Moderna"
WHERE employee_ID = 225 and dose_number = "1";

SELECT * FROM occ55314.Vaccines WHERE employee_ID = 225;

DELETE FROM occ55314.Vaccines WHERE employee_ID = 225;
```

4.



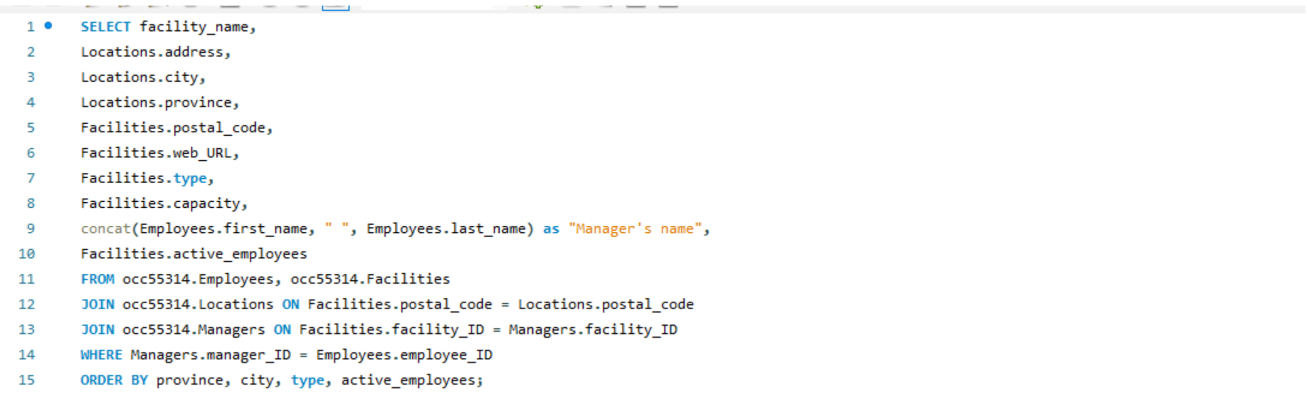
```sql
INSERT INTO Infections (employee_ID, severity, nature_of_infection, date_of_infection)
VALUES ('225', 'Mild', 'COVID-19', '2023-04-01');

UPDATE Infections
SET  severity = "severe"
WHERE employee_ID = 225 AND date_of_infection = '2023-04-01';

SELECT * FROM occ55314.Infections WHERE employee_ID = 225;

DELETE FROM occ55314.Infections WHERE employee_ID = 225;

```

5.

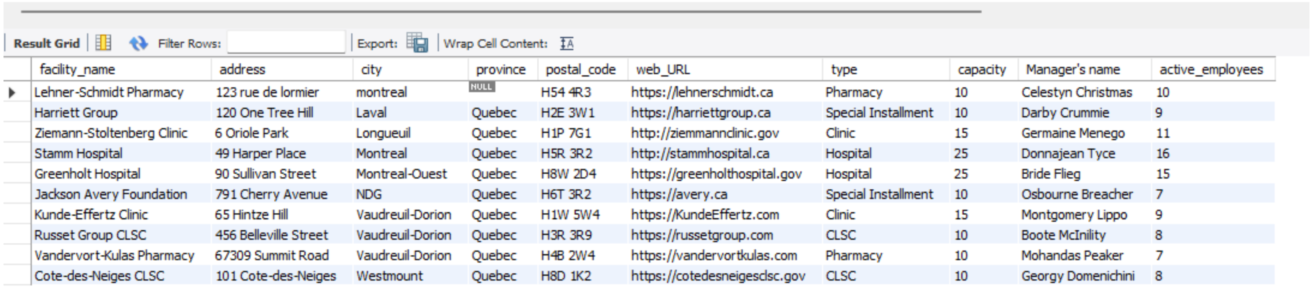


```sql
INSERT INTO Schedule (employee_id, facility_id, date, start_time, end_time)
VALUES (104, 506, '2023-04-07', '17:00', '21:00');

UPDATE Schedule
SET  end_time = "22:00"
WHERE employee_ID = 104 AND facility_ID = 506;

SELECT * FROM occ55314.Schedule WHERE employee_ID = 104 AND facility_ID = 506;

DELETE FROM occ55314.Schedule WHERE employee_ID = 104 AND facility_ID = 506;

```

6.

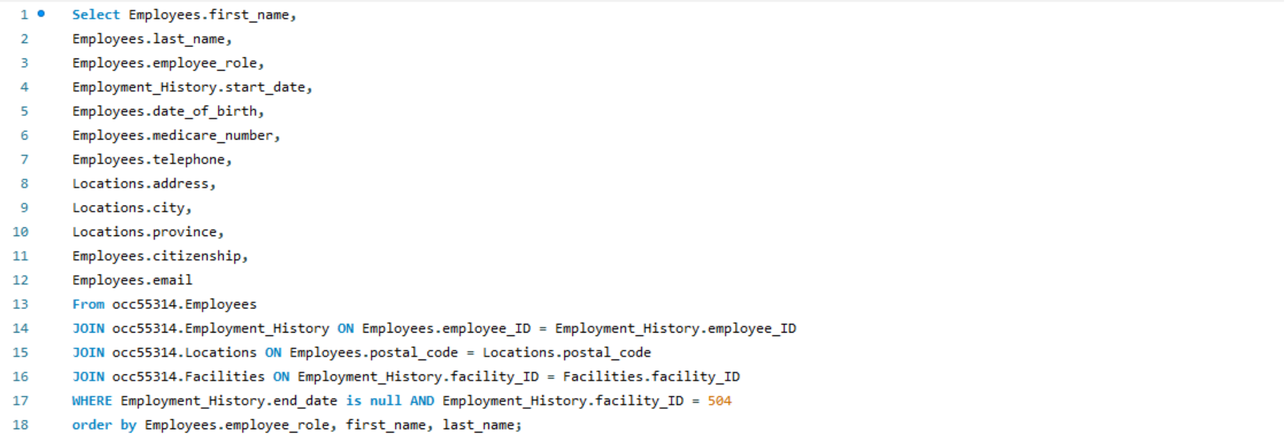


```sql
SELECT facility_name,
Locations.address,
Locations.city,
Locations.province,
Facilities.postal_code,
Facilities.web_URL,
Facilities.type,
Facilities.capacity,
concat(Employees.first_name, " ", Employees.last_name) as "Manager's name",
Facilities.active_employees
FROM occ55314.Employees, occ55314.Facilities
JOIN occ55314.Locations ON Facilities.postal_code = Locations.postal_code
JOIN occ55314.Managers ON Facilities.facility_ID = Managers.facility_ID
WHERE Managers.manager_ID = Employees.employee_ID
ORDER BY province, city, type, active_employees;
```

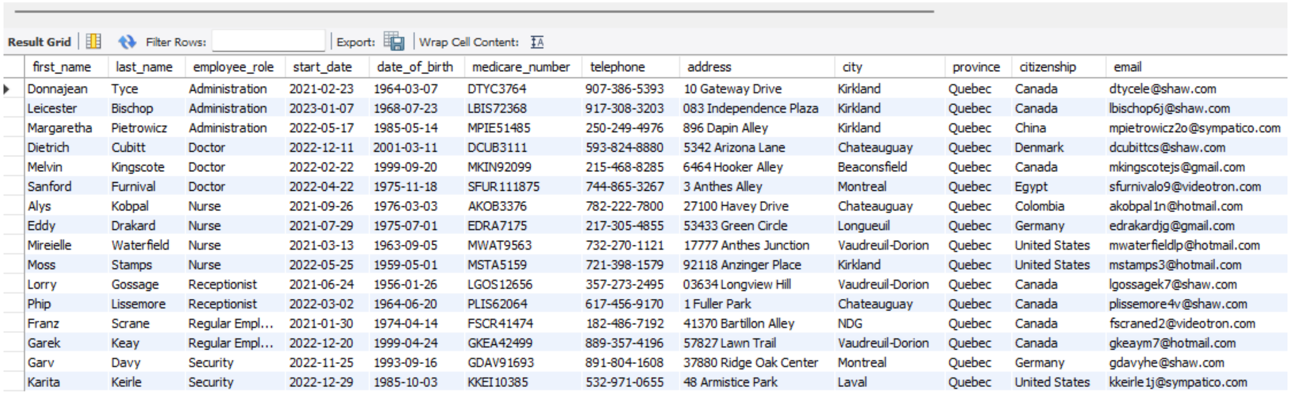


facility_name	address	city	province	postal_code	web_URL	type	capacity	Manager's name	active_employees
Lehner-Schmidt Pharmacy	123 rue de lormier	montreal	NULL	H54 4R3	https://lehnerschmidt.ca	Pharmacy	10	Celestyn Christmas	10
Harriett Group	120 One Tree Hill	Laval	Quebec	H2E 3W1	https://harriettgroup.ca	Special Installment	10	Darby Crummie	9
Ziemann-Stoltenberg Clinic	6 Oriole Park	Longueuil	Quebec	H1P 7G1	http://ziemmannclinic.gov	Clinic	15	Germaine Menego	11
Stamm Hospital	49 Harper Place	Montreal	Quebec	H5R 3R2	http://stammhospital.ca	Hospital	25	Donnajean Tyce	16
Greenholt Hospital	90 Sullivan Street	Montreal-Ouest	Quebec	H8W 2D4	https://greenholthospital.gov	Hospital	25	Bride Flieg	15
Jackson Avery Foundation	791 Cherry Avenue	NDG	Quebec	H6T 3R2	https://avery.ca	Special Installment	10	Osbourne Breacher	7
Kunde-Effertz Clinic	65 Hintze Hill	Vaudreuil-Dorion	Quebec	H1W 5W4	https://KundeEffertz.com	Clinic	15	Montgomery Lippo	9
Russet Group CLSC	456 Belleville Street	Vaudreuil-Dorion	Quebec	H3R 3R9	https://russetgroup.com	CLSC	10	Boote McInility	8
Vandervort-Kulas Pharmacy	67309 Summit Road	Vaudreuil-Dorion	Quebec	H4B 2W4	https://vandervortkulas.com	Pharmacy	10	Mohandas Peaker	7
Cote-des-Neiges CLSC	101 Cote-des-Neiges	Westmount	Quebec	H8D 1K2	https://cotedesneigesclsc.gov	CLSC	10	Georgy Domenichini	8

7.

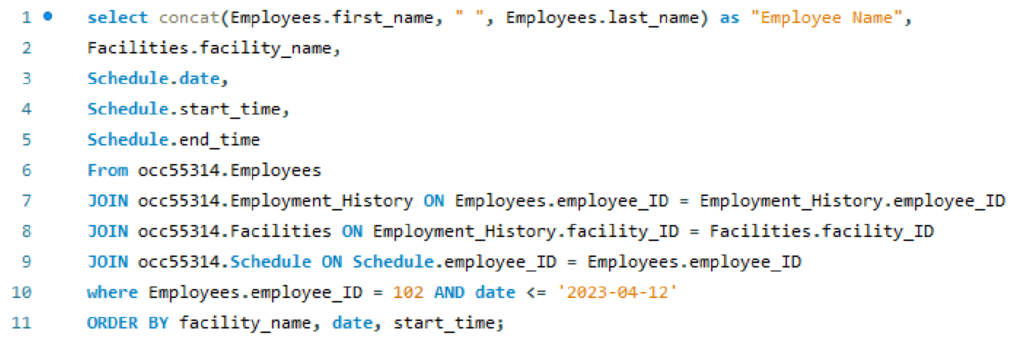


```sql
Select Employees.first_name,
Employees.last_name,
Employees.employee_role,
Employment_History.start_date,
Employees.date_of_birth,
Employees.medicare_number,
Employees.telephone,
Locations.address,
Locations.city,
Locations.province,
Employees.citizenship,
Employees.email
From occ55314.Employees
JOIN occ55314.Employment_History ON Employees.employee_ID = Employment_History.employee_ID
JOIN occ55314.Locations ON Employees.postal_code = Locations.postal_code
JOIN occ55314.Facilities ON Employment_History.facility_ID = Facilities.facility_ID
WHERE Employment_History.end_date is null AND Employment_History.facility_ID = 504
order by Employees.employee_role, first_name, last_name;
```

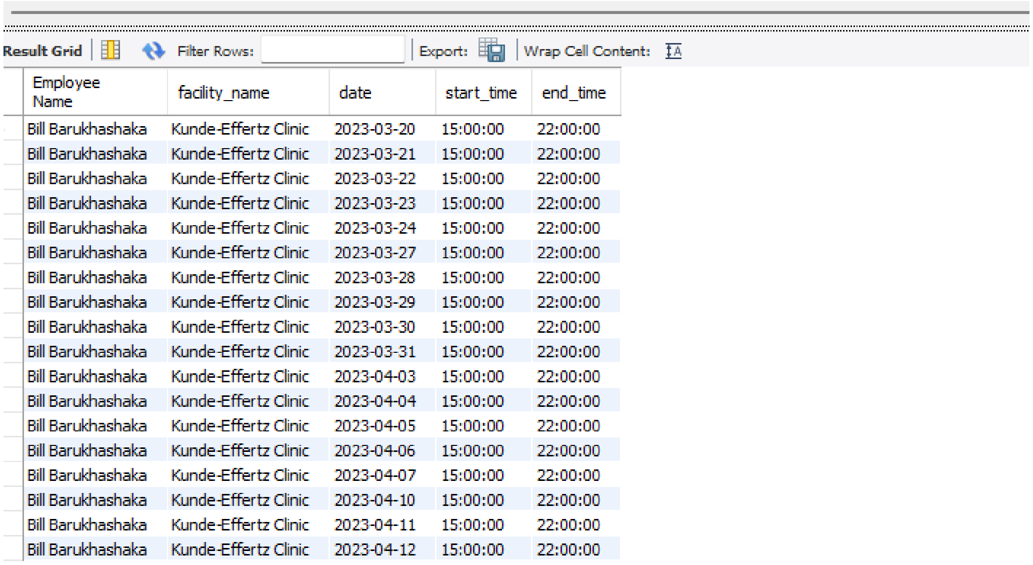


first_name	last_name	employee_role	start_date	date_of_birth	medicare_number	telephone	address	city	province	citizenship	email
Donnajean	Tyce	Administration	2021-02-23	1964-03-07	DTYC3764	907-386-5393	10 Gateway Drive	Kirkland	Quebec	Canada	dtycele@shaw.com
Leicester	Bischop	Administration	2023-01-07	1968-07-23	LBIS72368	917-308-3203	083 Independence Plaza	Kirkland	Quebec	Canada	lbischop6j@shaw.com
Margaretha	Pietrowicz	Administration	2022-05-17	1985-05-14	MPIE51485	250-249-4976	896 Dapin Alley	Kirkland	Quebec	China	mpietrowicz2o@sympatico.com
Dietrich	Cubitt	Doctor	2022-12-11	2001-03-11	DCUB3111	593-824-8880	5342 Arizona Lane	Chateauguay	Quebec	Denmark	dcubittcs@shaw.com
Melvin	Kingscote	Doctor	2022-02-22	1999-09-20	MKIN92099	215-468-8285	6464 Hooker Alley	Beaconsfield	Quebec	Canada	mkingscotejs@gmail.com
Sanford	Furnival	Doctor	2022-04-22	1975-11-18	SFUR111875	744-865-3267	3 Anthes Alley	Montreal	Quebec	Egypt	sfurnivalo9@videotron.com
Alys	Kobpal	Nurse	2021-09-26	1976-03-03	AKOB3376	782-222-7800	27100 Havey Drive	Chateauguay	Quebec	Colombia	akobpal1n@hotmail.com
Eddy	Drakard	Nurse	2021-07-29	1975-07-01	EDRA7175	217-305-4855	53433 Green Circle	Longueuil	Quebec	Germany	edrakardjg@gmail.com
Mireielle	Waterfield	Nurse	2021-03-13	1963-09-05	MWAT9563	732-270-1121	17777 Anthes Junction	Vaudreuil-Dorion	Quebec	United States	mwaterfieldlp@hotmail.com
Moss	Stamps	Nurse	2022-05-25	1959-05-01	MSTA5159	721-398-1579	92118 Anzinger Place	Kirkland	Quebec	United States	mstamps3@hotmail.com
Lorry	Gossage	Receptionist	2021-06-24	1956-01-26	LGOS12656	357-273-2495	03634 Longview Hill	Vaudreuil-Dorion	Quebec	Canada	lgossagek7@shaw.com
Phip	Lissemore	Receptionist	2022-03-02	1964-06-20	PLIS62064	617-456-9170	1 Fuller Park	Chateauguay	Quebec	Canada	plissemore4v@shaw.com
Franz	Scrane	Regular Empl...	2021-01-30	1974-04-14	FSCR41474	182-486-7192	41370 Bartillon Alley	NDG	Quebec	Canada	fscraned2@videotron.com
Garek	Keay	Regular Empl...	2022-12-20	1999-04-24	GKEA42499	889-357-4196	57827 Lawn Trail	Vaudreuil-Dorion	Quebec	Canada	gkeaym7@hotmail.com
Garv	Davy	Security	2022-11-25	1993-09-16	GDAV91693	891-804-1608	37880 Ridge Oak Center	Montreal	Quebec	Germany	gdavyhe@shaw.com
Karita	Keirle	Security	2022-12-29	1985-10-03	KKEI10385	532-971-0655	48 Armistice Park	Laval	Quebec	United States	kkeirle1j@sympatico.com

8.

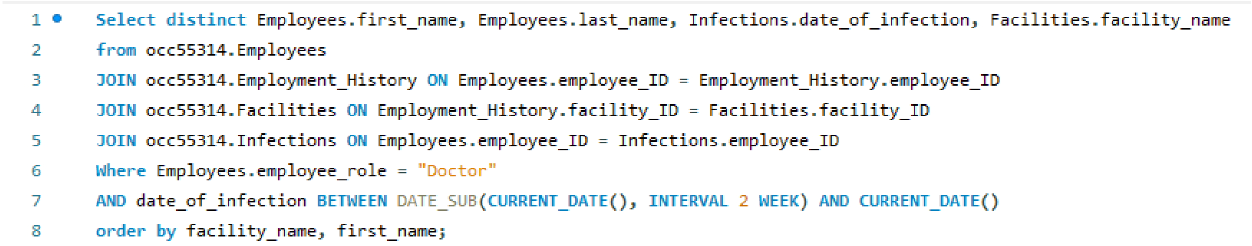


```sql
select concat(Employees.first_name, " ", Employees.last_name) as "Employee Name",
Facilities.facility_name,
Schedule.date,
Schedule.start_time,
Schedule.end_time
From occ55314.Employees
JOIN occ55314.Employment_History ON Employees.employee_ID = Employment_History.employee_ID
JOIN occ55314.Facilities ON Employment_History.facility_ID = Facilities.facility_ID
JOIN occ55314.Schedule ON Schedule.employee_ID = Employees.employee_ID
where Employees.employee_ID = 102 AND date <= '2023-04-12'
ORDER BY facility_name, date, start_time;
```

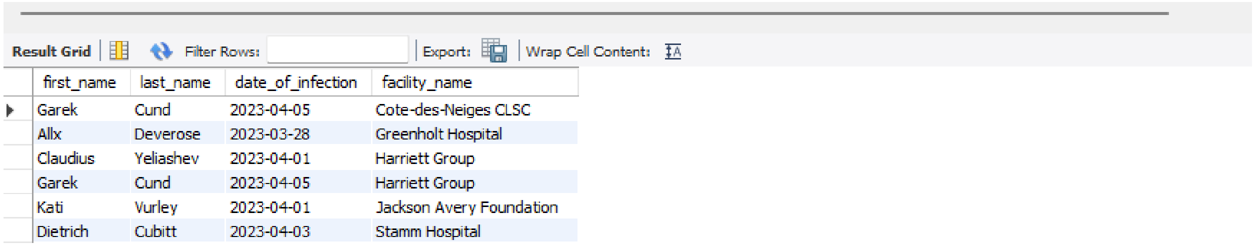


Employee Name	facility_name	date	start_time	end_time
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-20	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-21	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-22	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-23	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-24	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-27	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-28	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-29	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-30	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-03-31	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-04-03	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-04-04	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-04-05	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-04-06	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-04-07	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-04-10	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-04-11	15:00:00	22:00:00
Bill Barukhashaka	Kunde-Effertz Clinic	2023-04-12	15:00:00	22:00:00

9.

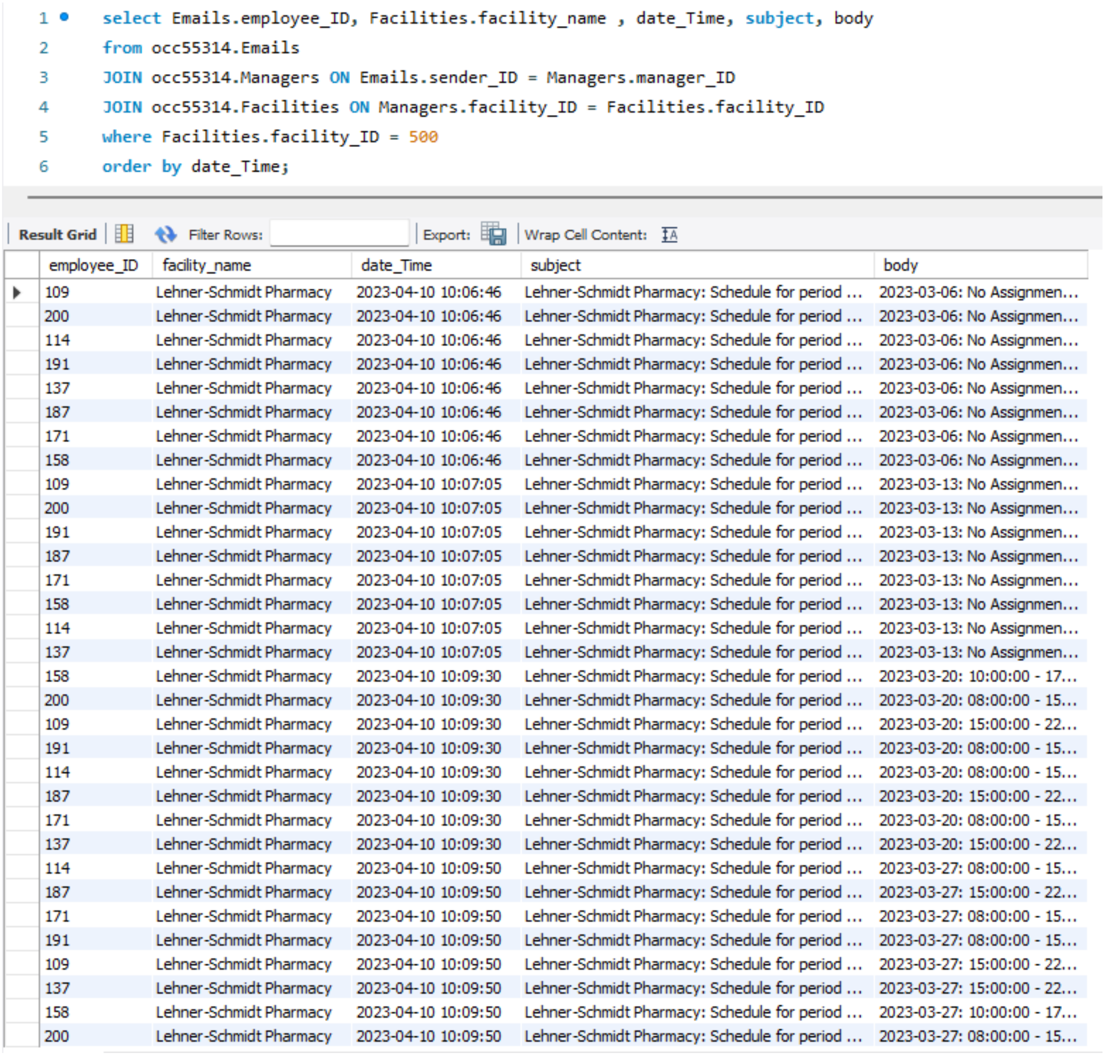


```sql
Select distinct Employees.first_name, Employees.last_name, Infections.date_of_infection, Facilities.facility_name
from occ55314.Employees
JOIN occ55314.Employment_History ON Employees.employee_ID = Employment_History.employee_ID
JOIN occ55314.Facilities ON Employment_History.facility_ID = Facilities.facility_ID
JOIN occ55314.Infections ON Employees.employee_ID = Infections.employee_ID
Where Employees.employee_role = "Doctor"
AND date_of_infection BETWEEN DATE_SUB(CURRENT_DATE(), INTERVAL 2 WEEK) AND CURRENT_DATE()
order by facility_name, first_name;
```



first_name	last_name	date_of_infection	facility_name
Garek	Cund	2023-04-05	Cote-des-Neiges CLSC
Allx	Deverose	2023-03-28	Greenholt Hospital
Claudius	Yeliashev	2023-04-01	Harriett Group
Garek	Cund	2023-04-05	Harriett Group
Kati	Vurley	2023-04-01	Jackson Avery Foundation
Dietrich	Cubitt	2023-04-03	Stamm Hospital

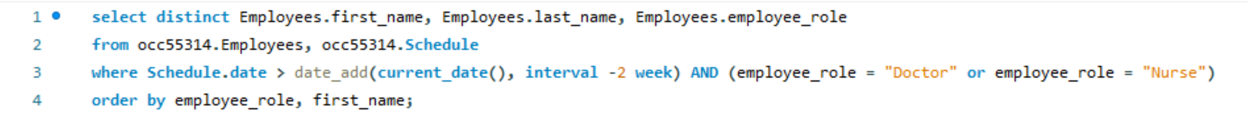
10.



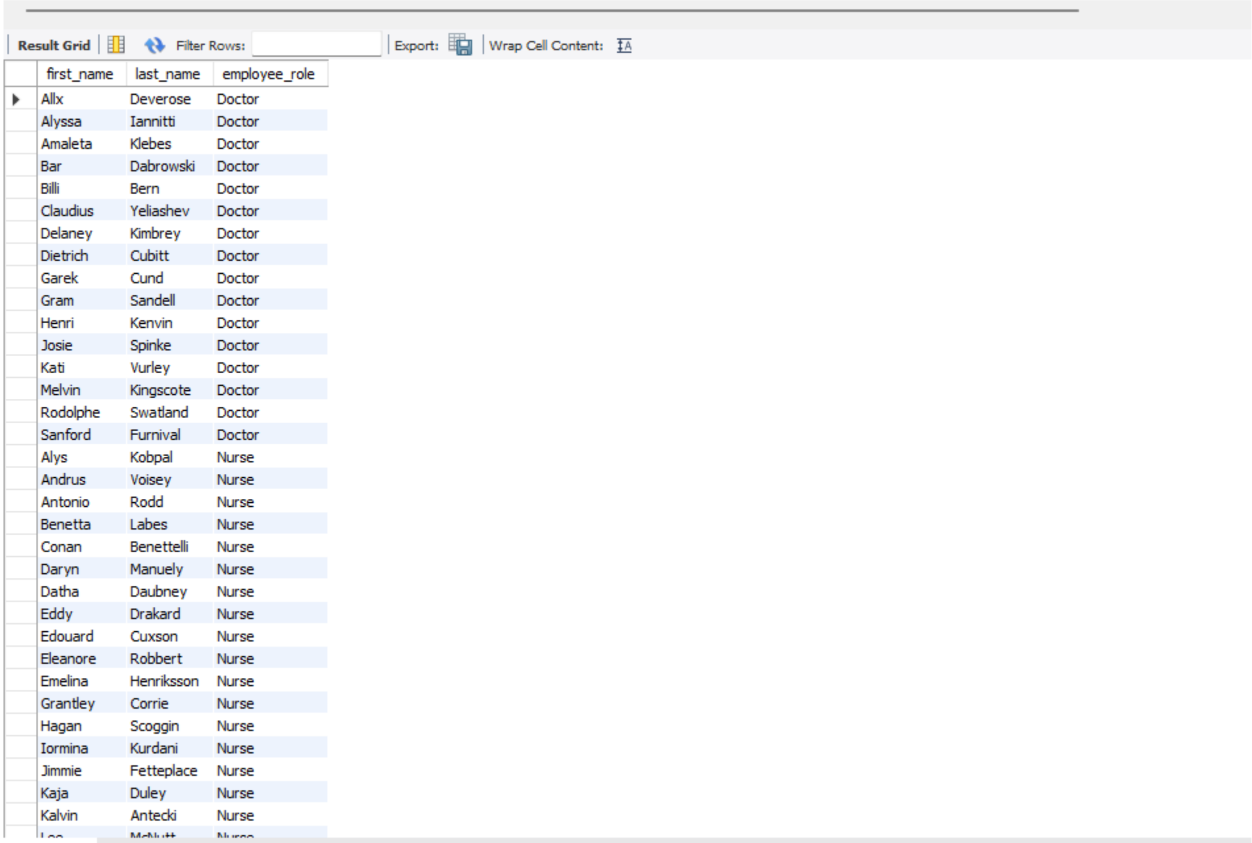
```
select Emails.employee_ID, Facilities.facility_name , date_Time, subject, body
from occ55314.Emails
JOIN occ55314.Managers ON Emails.sender_ID = Managers.manager_ID
JOIN occ55314.Facilities ON Managers.facility_ID = Facilities.facility_ID
where Facilities.facility_ID = 500
order by date_Time;
```

employee_ID	facility_name	date_Time	subject	body
109	Lehner-Schmidt Pharmacy	2023-04-10 10:06:46	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-06: No Assignmen...
200	Lehner-Schmidt Pharmacy	2023-04-10 10:06:46	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-06: No Assignmen...
114	Lehner-Schmidt Pharmacy	2023-04-10 10:06:46	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-06: No Assignmen...
191	Lehner-Schmidt Pharmacy	2023-04-10 10:06:46	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-06: No Assignmen...
137	Lehner-Schmidt Pharmacy	2023-04-10 10:06:46	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-06: No Assignmen...
187	Lehner-Schmidt Pharmacy	2023-04-10 10:06:46	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-06: No Assignmen...
171	Lehner-Schmidt Pharmacy	2023-04-10 10:06:46	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-06: No Assignmen...
158	Lehner-Schmidt Pharmacy	2023-04-10 10:06:46	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-06: No Assignmen...
109	Lehner-Schmidt Pharmacy	2023-04-10 10:07:05	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-13: No Assignmen...
200	Lehner-Schmidt Pharmacy	2023-04-10 10:07:05	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-13: No Assignmen...
191	Lehner-Schmidt Pharmacy	2023-04-10 10:07:05	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-13: No Assignmen...
187	Lehner-Schmidt Pharmacy	2023-04-10 10:07:05	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-13: No Assignmen...
171	Lehner-Schmidt Pharmacy	2023-04-10 10:07:05	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-13: No Assignmen...
158	Lehner-Schmidt Pharmacy	2023-04-10 10:07:05	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-13: No Assignmen...
114	Lehner-Schmidt Pharmacy	2023-04-10 10:07:05	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-13: No Assignmen...
137	Lehner-Schmidt Pharmacy	2023-04-10 10:07:05	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-13: No Assignmen...
158	Lehner-Schmidt Pharmacy	2023-04-10 10:09:30	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-20: 10:00:00 - 17...
200	Lehner-Schmidt Pharmacy	2023-04-10 10:09:30	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-20: 08:00:00 - 15...
109	Lehner-Schmidt Pharmacy	2023-04-10 10:09:30	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-20: 15:00:00 - 22...
191	Lehner-Schmidt Pharmacy	2023-04-10 10:09:30	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-20: 08:00:00 - 15...
114	Lehner-Schmidt Pharmacy	2023-04-10 10:09:30	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-20: 08:00:00 - 15...
187	Lehner-Schmidt Pharmacy	2023-04-10 10:09:30	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-20: 15:00:00 - 22...
171	Lehner-Schmidt Pharmacy	2023-04-10 10:09:30	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-20: 08:00:00 - 15...
137	Lehner-Schmidt Pharmacy	2023-04-10 10:09:30	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-20: 15:00:00 - 22...
114	Lehner-Schmidt Pharmacy	2023-04-10 10:09:50	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-27: 08:00:00 - 15...
187	Lehner-Schmidt Pharmacy	2023-04-10 10:09:50	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-27: 15:00:00 - 22...
171	Lehner-Schmidt Pharmacy	2023-04-10 10:09:50	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-27: 08:00:00 - 15...
191	Lehner-Schmidt Pharmacy	2023-04-10 10:09:50	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-27: 08:00:00 - 15...
109	Lehner-Schmidt Pharmacy	2023-04-10 10:09:50	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-27: 15:00:00 - 22...
137	Lehner-Schmidt Pharmacy	2023-04-10 10:09:50	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-27: 15:00:00 - 22...
158	Lehner-Schmidt Pharmacy	2023-04-10 10:09:50	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-27: 10:00:00 - 17...
200	Lehner-Schmidt Pharmacy	2023-04-10 10:09:50	Lehner-Schmidt Pharmacy: Schedule for period ...	2023-03-27: 08:00:00 - 15...

11.

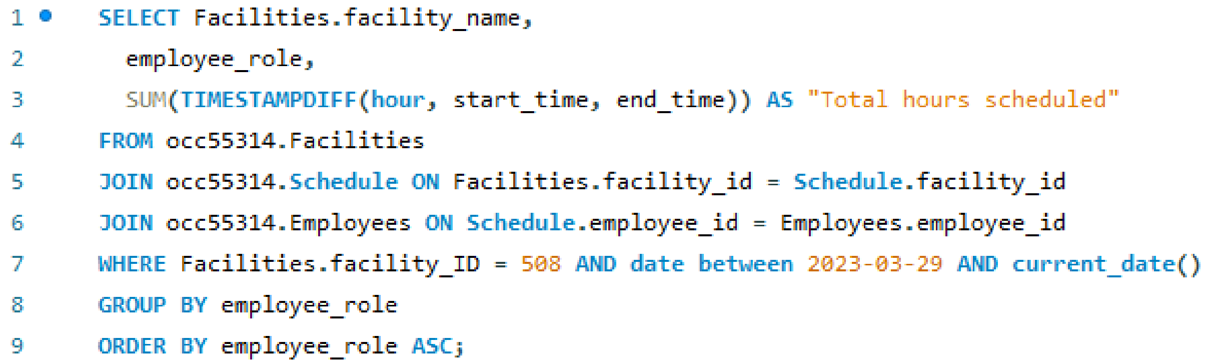


```
select distinct Employees.first_name, Employees.last_name, Employees.employee_role
from occ55314.Employees, occ55314.Schedule
where Schedule.date > date_add(current_date(), interval -2 week) AND (employee_role = "Doctor" or employee_role = "Nurse")
order by employee_role, first_name;
```

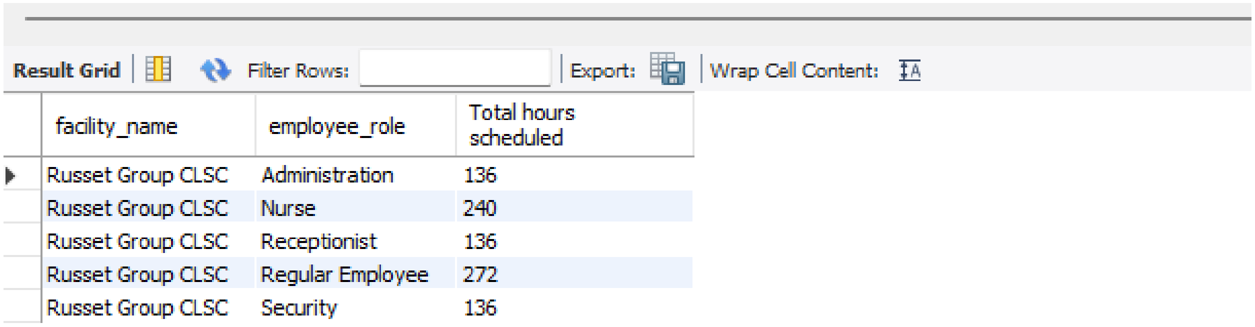


first_name	last_name	employee_role
Allx	Deverose	Doctor
Alyssa	Iannitti	Doctor
Amaleta	Klebes	Doctor
Bar	Dabrowski	Doctor
Billi	Bern	Doctor
Claudius	Yeliashev	Doctor
Delaney	Kimbrey	Doctor
Dietrich	Cubitt	Doctor
Garek	Cund	Doctor
Gram	Sandell	Doctor
Henri	Kenvin	Doctor
Josie	Spinke	Doctor
Kati	Vurley	Doctor
Melvin	Kingscote	Doctor
Rodolphe	Swatland	Doctor
Sanford	Furnival	Doctor
Alys	Kobpal	Nurse
Andrus	Voisey	Nurse
Antonio	Rodd	Nurse
Benetta	Labes	Nurse
Conan	Benettelli	Nurse
Daryn	Manuely	Nurse
Datha	Daubney	Nurse
Eddy	Drakard	Nurse
Edouard	Cuxson	Nurse
Eleanore	Robbert	Nurse
Emelina	Henriksson	Nurse
Grantley	Corrie	Nurse
Hagan	Scoggin	Nurse
Iormina	Kurdani	Nurse
Jimmie	Fetteplace	Nurse
Kaja	Duley	Nurse
Kalvin	Antecki	Nurse
Leo	McNutt	Nurse

12.

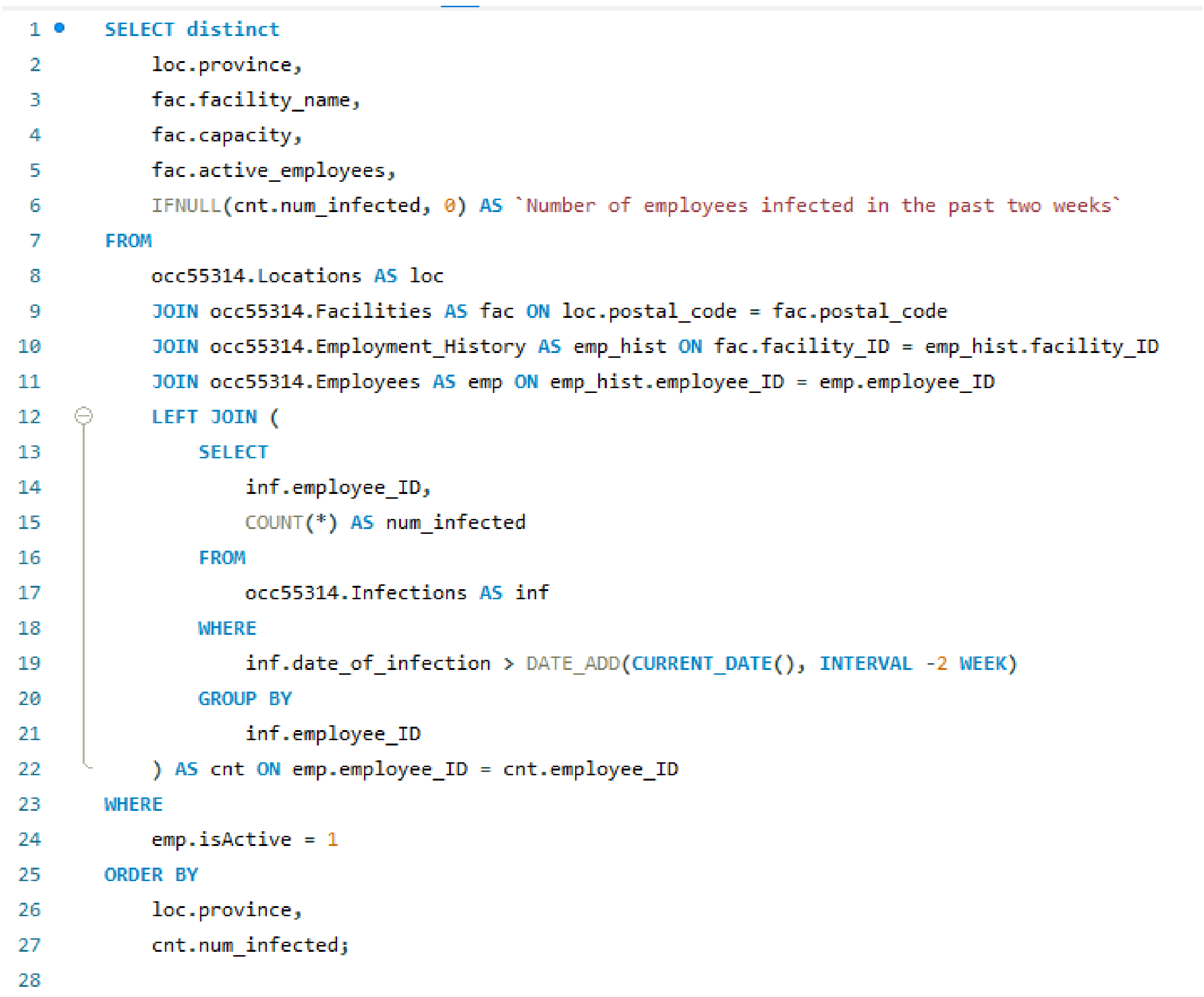


```sql
SELECT Facilities.facility_name,
  employee_role,
  SUM(TIMESTAMPDIFF(hour, start_time, end_time)) AS "Total hours scheduled"
FROM occ55314.Facilities
JOIN occ55314.Schedule ON Facilities.facility_id = Schedule.facility_id
JOIN occ55314.Employees ON Schedule.employee_id = Employees.employee_id
WHERE Facilities.facility_ID = 508 AND date between 2023-03-29 AND current_date()
GROUP BY employee_role
ORDER BY employee_role ASC;
```

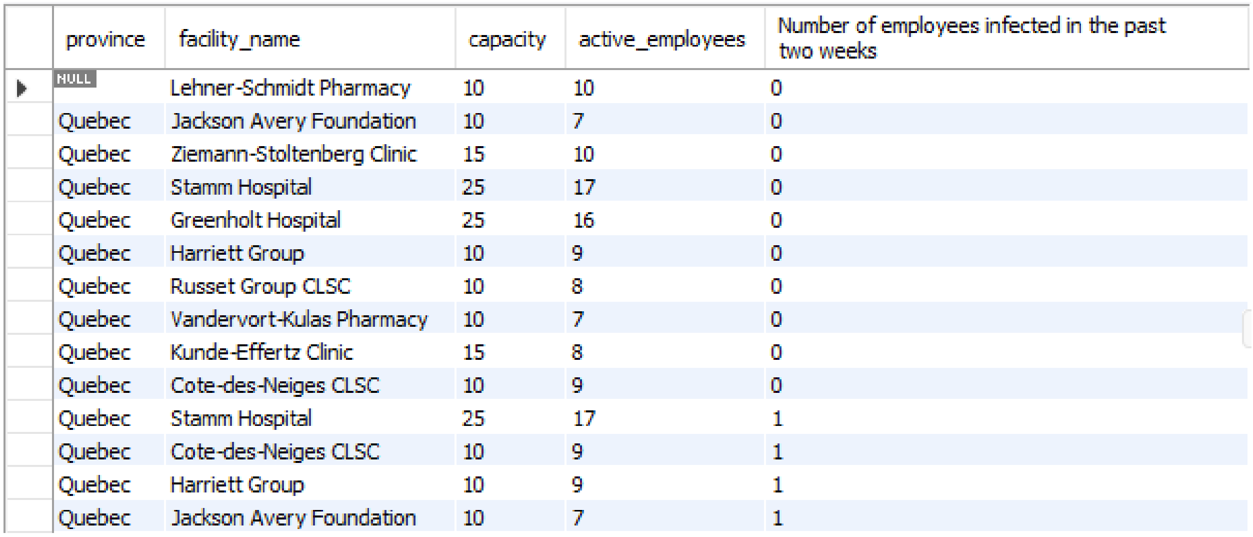


facility_name	employee_role	Total hours scheduled
Russet Group CLSC	Administration	136
Russet Group CLSC	Nurse	240
Russet Group CLSC	Receptionist	136
Russet Group CLSC	Regular Employee	272
Russet Group CLSC	Security	136

13.

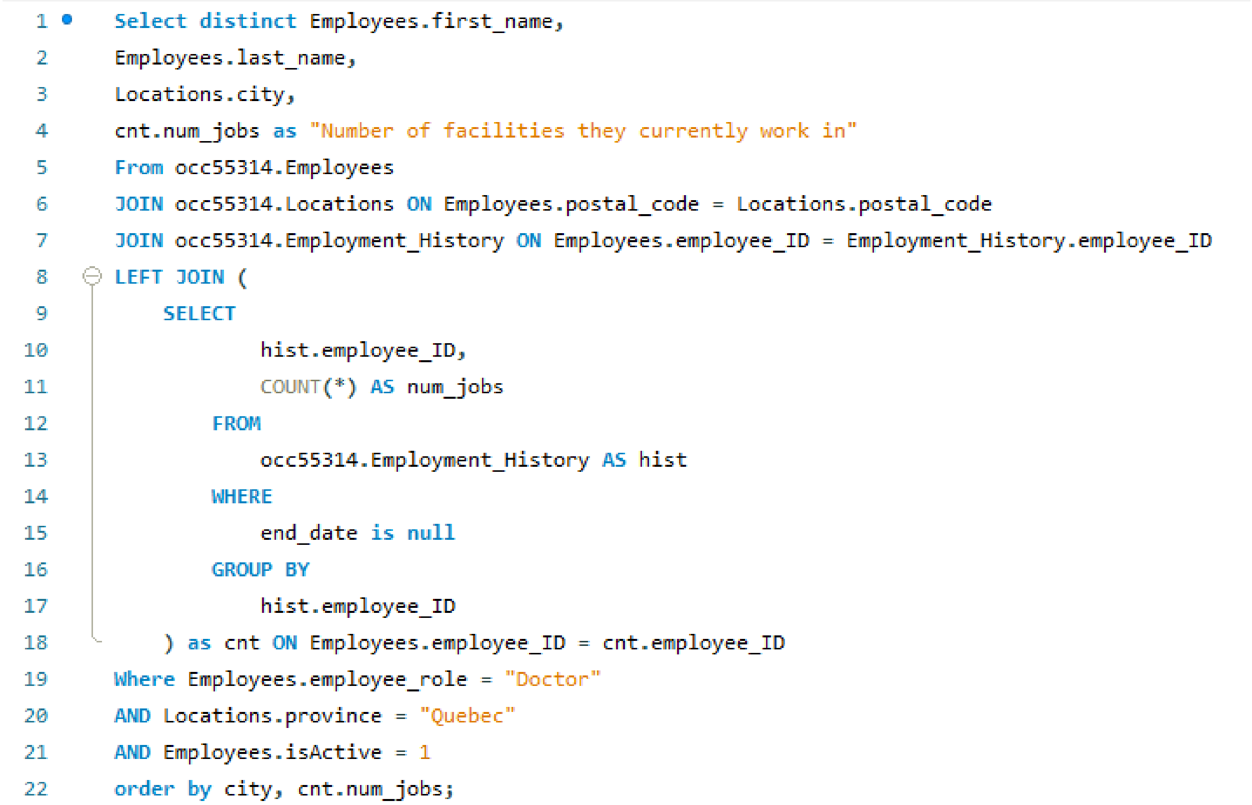


```
SELECT distinct
    loc.province,
    fac.facility_name,
    fac.capacity,
    fac.active_employees,
    IFNULL(cnt.num_infected, 0) AS `Number of employees infected in the past two weeks`
FROM
    occ55314.Locations AS loc
    JOIN occ55314.Facilities AS fac ON loc.postal_code = fac.postal_code
    JOIN occ55314.Employment_History AS emp_hist ON fac.facility_ID = emp_hist.facility_ID
    JOIN occ55314.Employees AS emp ON emp_hist.employee_ID = emp.employee_ID
    LEFT JOIN (
        SELECT
            inf.employee_ID,
            COUNT(*) AS num_infected
        FROM
            occ55314.Infections AS inf
        WHERE
            inf.date_of_infection > DATE_ADD(CURRENT_DATE(), INTERVAL -2 WEEK)
        GROUP BY
            inf.employee_ID
    ) AS cnt ON emp.employee_ID = cnt.employee_ID
WHERE
    emp.isActive = 1
ORDER BY
    loc.province,
    cnt.num_infected;
```

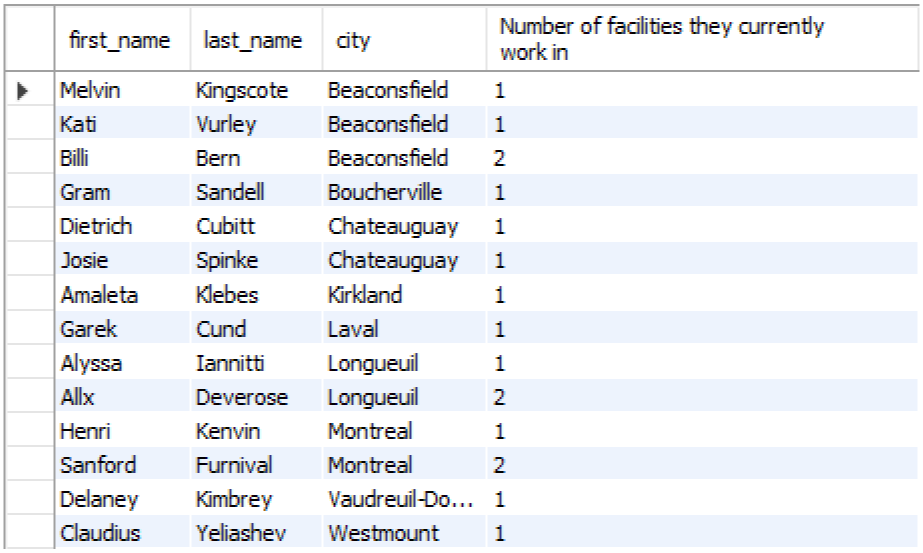


province	facility_name	capacity	active_employees	Number of employees infected in the past two weeks
NULL	Lehner-Schmidt Pharmacy	10	10	0
Quebec	Jackson Avery Foundation	10	7	0
Quebec	Ziemann-Stoltenberg Clinic	15	10	0
Quebec	Stamm Hospital	25	17	0
Quebec	Greenholt Hospital	25	16	0
Quebec	Harriett Group	10	9	0
Quebec	Russet Group CLSC	10	8	0
Quebec	Vandervort-Kulas Pharmacy	10	7	0
Quebec	Kunde-Effertz Clinic	15	8	0
Quebec	Cote-des-Neiges CLSC	10	9	0
Quebec	Stamm Hospital	25	17	1
Quebec	Cote-des-Neiges CLSC	10	9	1
Quebec	Harriett Group	10	9	1
Quebec	Jackson Avery Foundation	10	7	1

14.

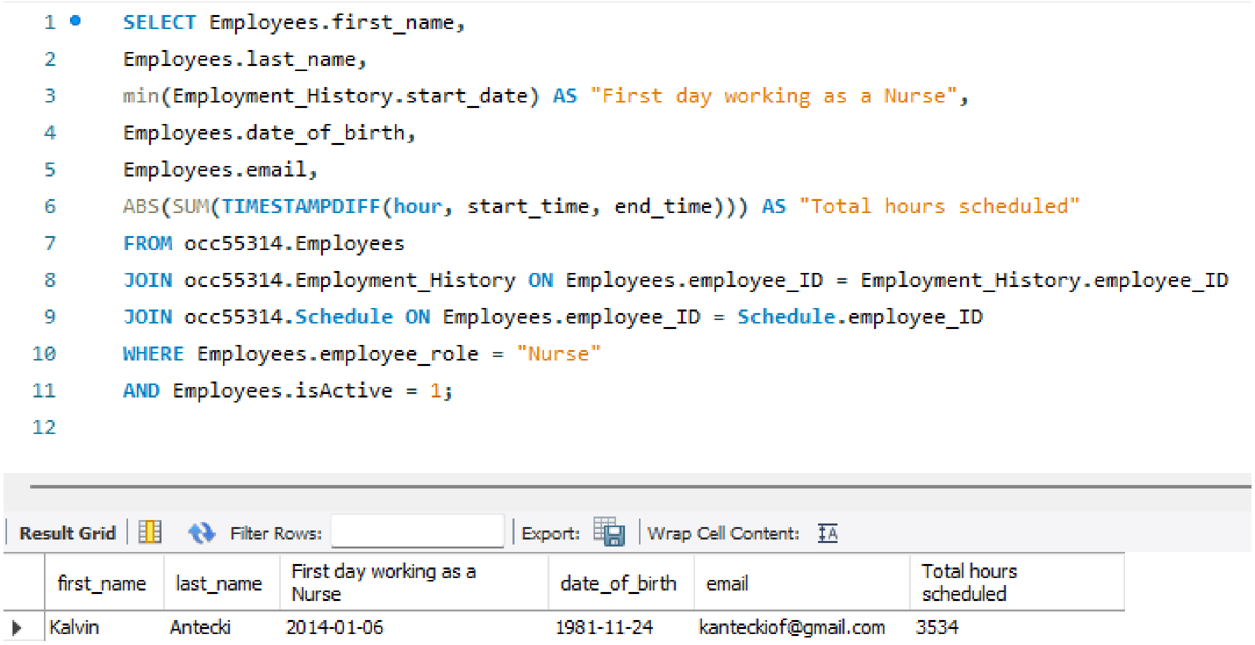


```
Select distinct Employees.first_name,
Employees.last_name,
Locations.city,
cnt.num_jobs as "Number of facilities they currently work in"
From occ55314.Employees
JOIN occ55314.Locations ON Employees.postal_code = Locations.postal_code
JOIN occ55314.Employment_History ON Employees.employee_ID = Employment_History.employee_ID
LEFT JOIN (
    SELECT
            hist.employee_ID,
            COUNT(*) AS num_jobs
        FROM
            occ55314.Employment_History AS hist
        WHERE
            end_date is null
        GROUP BY
            hist.employee_ID
    ) as cnt ON Employees.employee_ID = cnt.employee_ID
Where Employees.employee_role = "Doctor"
AND Locations.province = "Quebec"
AND Employees.isActive = 1
order by city, cnt.num_jobs;
```



	first_name	last_name	city	Number of facilities they currently work in
▶	Melvin	Kingscote	Beaconsfield	1
	Kati	Vurley	Beaconsfield	1
	Billi	Bern	Beaconsfield	2
	Gram	Sandell	Boucherville	1
	Dietrich	Cubitt	Chateauguay	1
	Josie	Spinke	Chateauguay	1
	Amaleta	Klebes	Kirkland	1
	Garek	Cund	Laval	1
	Alyssa	Iannitti	Longueuil	1
	Allx	Deverose	Longueuil	2
	Henri	Kenvin	Montreal	1
	Sanford	Furnival	Montreal	2
	Delaney	Kimbrey	Vaudreuil-Do...	1
	Claudius	Yeliashev	Westmount	1

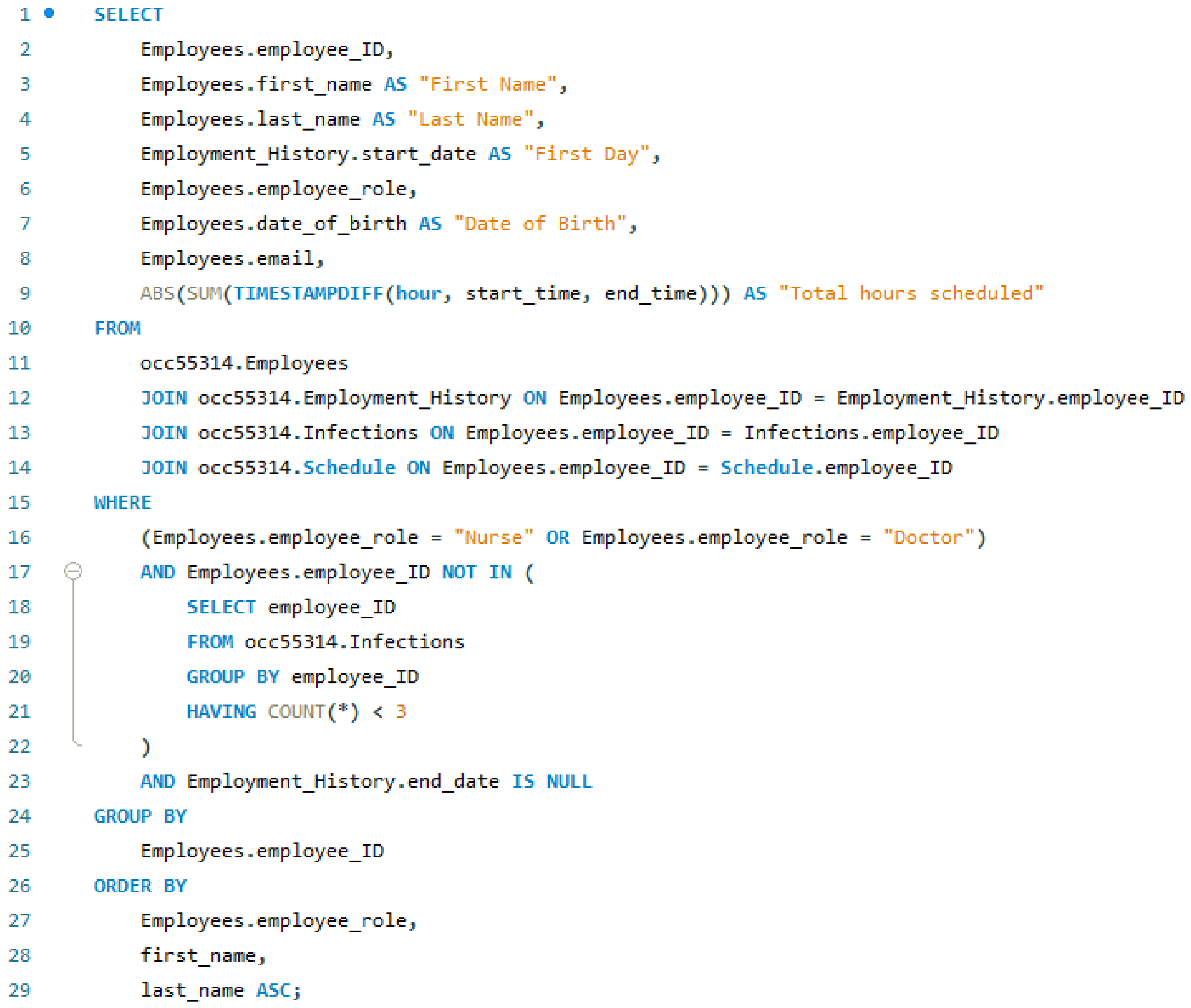
15.



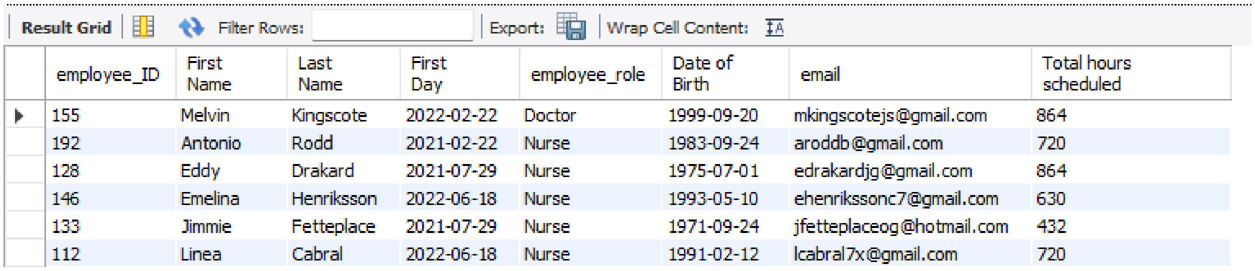
```sql
SELECT Employees.first_name,
Employees.last_name,
min(Employment_History.start_date) AS "First day working as a Nurse",
Employees.date_of_birth,
Employees.email,
ABS(SUM(TIMESTAMPDIFF(hour, start_time, end_time))) AS "Total hours scheduled"
FROM occ55314.Employees
JOIN occ55314.Employment_History ON Employees.employee_ID = Employment_History.employee_ID
JOIN occ55314.Schedule ON Employees.employee_ID = Schedule.employee_ID
WHERE Employees.employee_role = "Nurse"
AND Employees.isActive = 1;
```

first_name	last_name	First day working as a Nurse	date_of_birth	email	Total hours scheduled
Kalvin	Antecki	2014-01-06	1981-11-24	kanteckiof@gmail.com	3534

16.

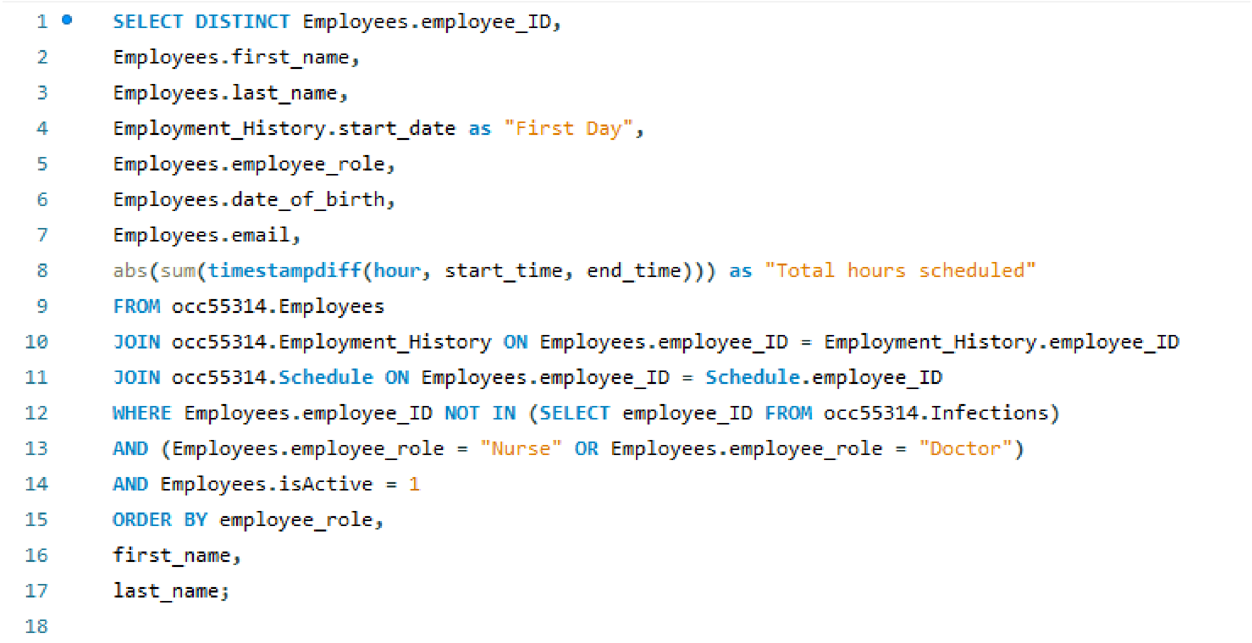


```
SELECT
    Employees.employee_ID,
    Employees.first_name AS "First Name",
    Employees.last_name AS "Last Name",
    Employment_History.start_date AS "First Day",
    Employees.employee_role,
    Employees.date_of_birth AS "Date of Birth",
    Employees.email,
    ABS(SUM(TIMESTAMPDIFF(hour, start_time, end_time))) AS "Total hours scheduled"
FROM
    occ55314.Employees
    JOIN occ55314.Employment_History ON Employees.employee_ID = Employment_History.employee_ID
    JOIN occ55314.Infections ON Employees.employee_ID = Infections.employee_ID
    JOIN occ55314.Schedule ON Employees.employee_ID = Schedule.employee_ID
WHERE
    (Employees.employee_role = "Nurse" OR Employees.employee_role = "Doctor")
    AND Employees.employee_ID NOT IN (
        SELECT employee_ID
        FROM occ55314.Infections
        GROUP BY employee_ID
        HAVING COUNT(*) < 3
    )
    AND Employment_History.end_date IS NULL
GROUP BY
    Employees.employee_ID
ORDER BY
    Employees.employee_role,
    first_name,
    last_name ASC;
```

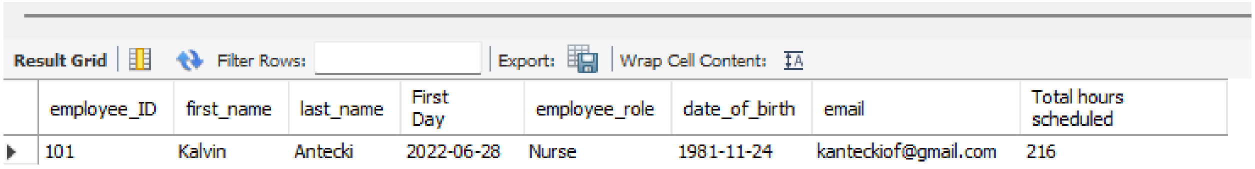


employee_ID	First Name	Last Name	First Day	employee_role	Date of Birth	email	Total hours scheduled
155	Melvin	Kingscote	2022-02-22	Doctor	1999-09-20	mkingscotejs@gmail.com	864
192	Antonio	Rodd	2021-02-22	Nurse	1983-09-24	aroddb@gmail.com	720
128	Eddy	Drakard	2021-07-29	Nurse	1975-07-01	edrakardjg@gmail.com	864
146	Emelina	Henriksson	2022-06-18	Nurse	1993-05-10	ehenrikssonc7@gmail.com	630
133	Jimmie	Fetteplace	2021-07-29	Nurse	1971-09-24	jfetteplaceog@hotmail.com	432
112	Linea	Cabral	2022-06-18	Nurse	1991-02-12	lcabral7x@gmail.com	720

17.

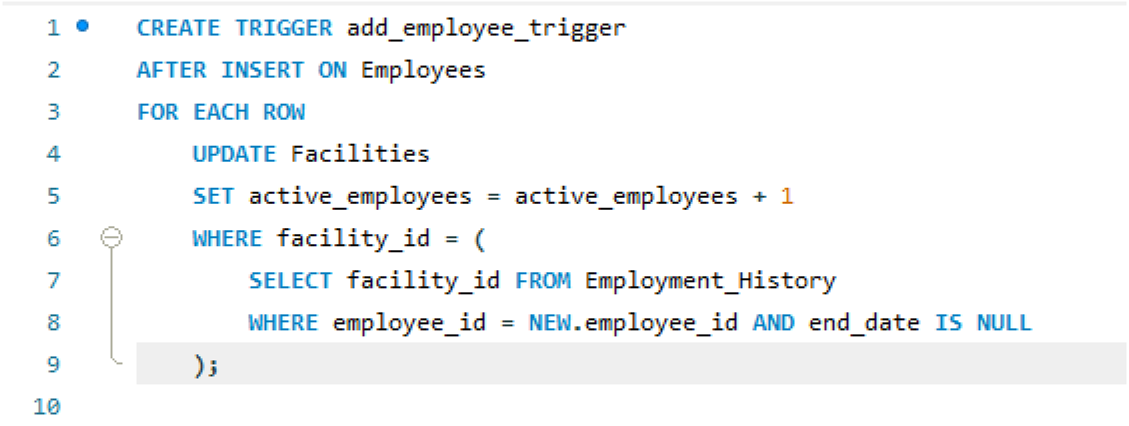


```sql
SELECT DISTINCT Employees.employee_ID,
Employees.first_name,
Employees.last_name,
Employment_History.start_date as "First Day",
Employees.employee_role,
Employees.date_of_birth,
Employees.email,
abs(sum(timestampdiff(hour, start_time, end_time))) as "Total hours scheduled"
FROM occ55314.Employees
JOIN occ55314.Employment_History ON Employees.employee_ID = Employment_History.employee_ID
JOIN occ55314.Schedule ON Employees.employee_ID = Schedule.employee_ID
WHERE Employees.employee_ID NOT IN (SELECT employee_ID FROM occ55314.Infections)
AND (Employees.employee_role = "Nurse" OR Employees.employee_role = "Doctor")
AND Employees.isActive = 1
ORDER BY employee_role,
first_name,
last_name;
```



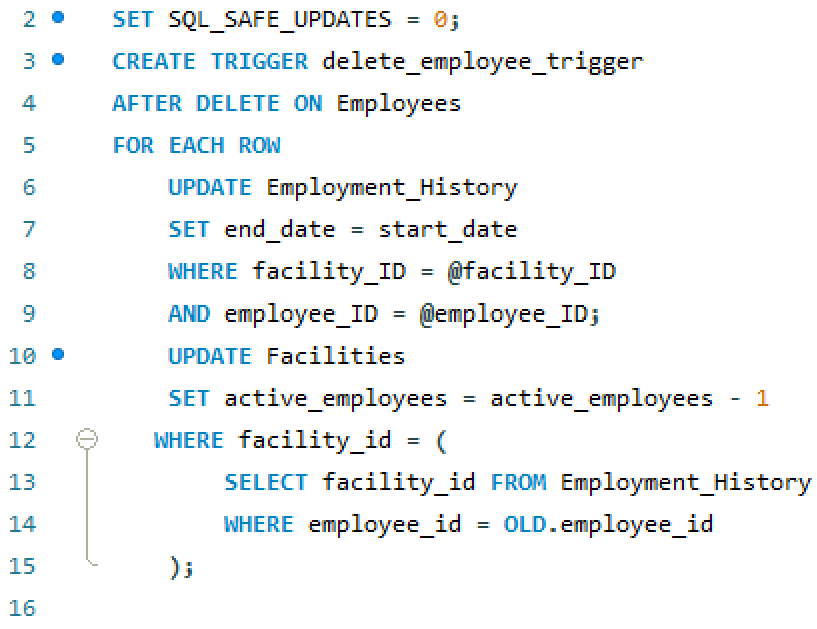
employee_ID	first_name	last_name	First Day	employee_role	date_of_birth	email	Total hours scheduled
101	Kalvin	Antecki	2022-06-28	Nurse	1981-11-24	kanteckiof@gmail.com	216

18. add_employee_trigger : increments active employee number in the facility table when an employee is created:



```sql
CREATE TRIGGER add_employee_trigger
AFTER INSERT ON Employees
FOR EACH ROW
    UPDATE Facilities
    SET active_employees = active_employees + 1
    WHERE facility_id = (
        SELECT facility_id FROM Employment_History
        WHERE employee_id = NEW.employee_id AND end_date IS NULL
    );
```

delete_employee_trigger: Once deleted from Employees, this trigger deletes the record from Employment_History and decreases the number of active employees in the facility by 1.



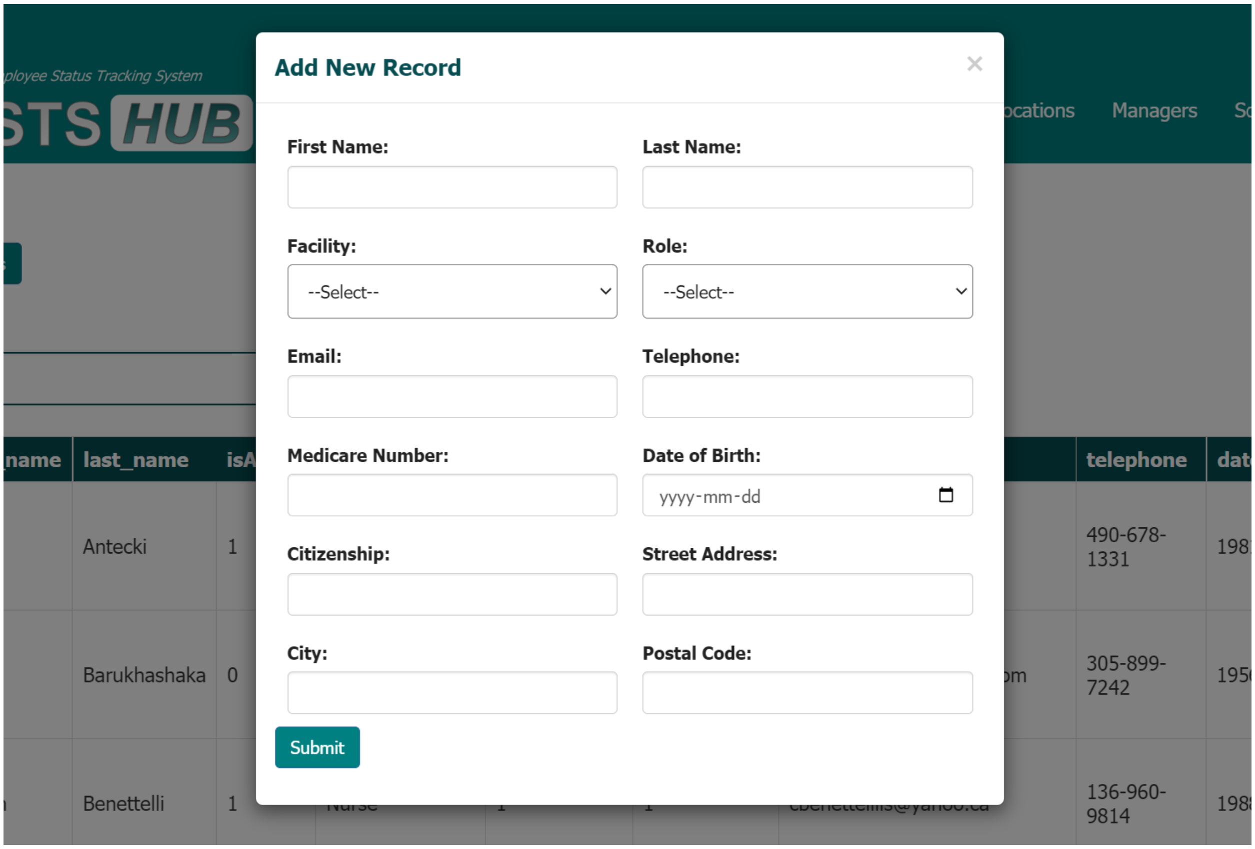
```
SET SQL_SAFE_UPDATES = 0;
CREATE TRIGGER delete_employee_trigger
AFTER DELETE ON Employees
FOR EACH ROW
    UPDATE Employment_History
    SET end_date = start_date
    WHERE facility_ID = @facility_ID
    AND employee_ID = @employee_ID;
    UPDATE Facilities
    SET active_employees = active_employees - 1
   WHERE facility_id = (
        SELECT facility_id FROM Employment_History
        WHERE employee_id = OLD.employee_id
    );
```

19. HFESTS Hub Functionality & Data Integrity Constraints

The HFESTS Hub online graphical interface facilitates interactions with the HFESTS Database. The interface consists of a navigation menu enabling access to all relations within the database schema and a keyword search bar that allows the user to search all relation attributes.

The individual relation relation windows each feature their own functionalities, described below.

1. **Employees Window:** Allows creation of a new employee record along with deletion and update rights for existing records. New Employee ID numbers are auto-assigned by the system.



Add New Record

First Name:

Last Name:

Facility: --Select--

Role: --Select--

Email:

Telephone:

Medicare Number:

Date of Birth: yyyy-mm-dd

Citizenship:

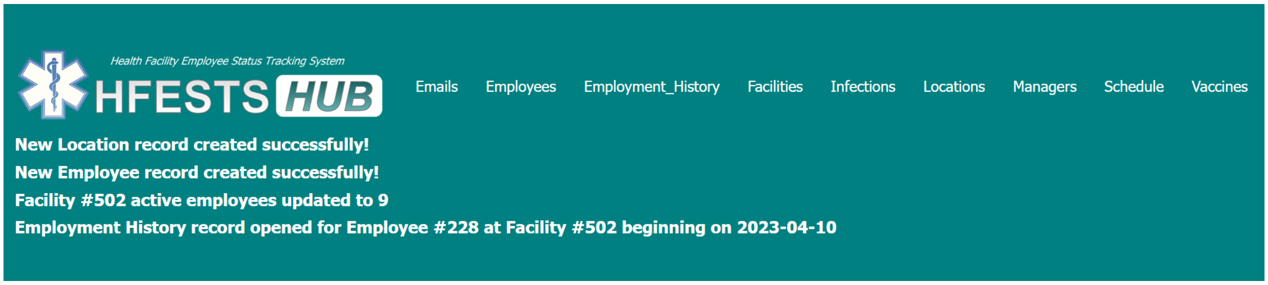
Street Address:

City:

Postal Code:

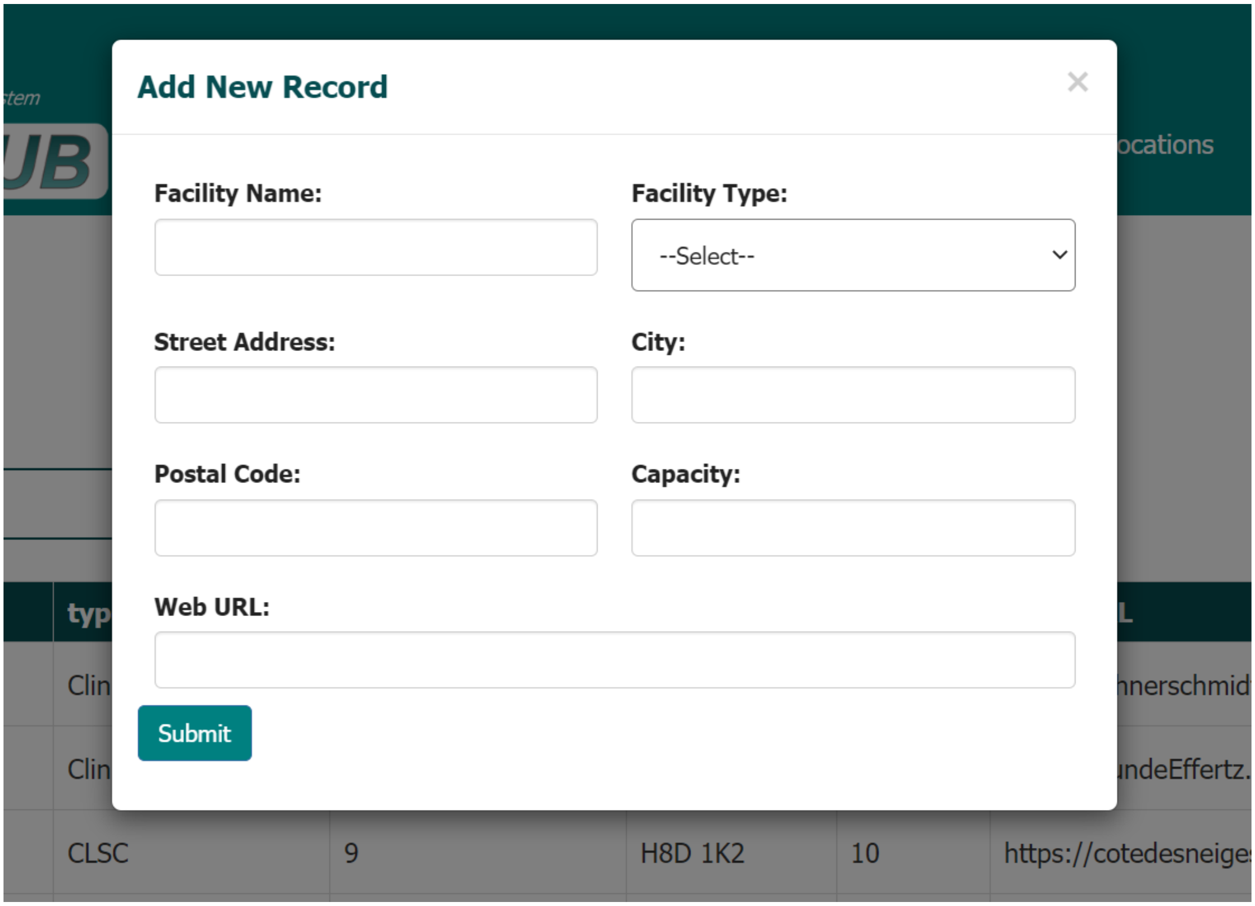
Adding a new employee record automatically creates a new Location record to store the Employee's address (if not already tracked in the system), opens a new Employment History record at the selected facility with start date as the current date, and increments the number of Active Employees tracked in the Facilities table for the selected facility.

If the selected Facility is currently at capacity, the system will reject the employee record.



New Location record created successfully!
New Employee record created successfully!
Facility #502 active employees updated to 9
Employment History record opened for Employee #228 at Facility #502 beginning on 2023-04-10

2. **Facilities Window:** Allows creation of a new facility record along with deletion and update rights for existing records. New Facility ID numbers are auto assigned by the system.



Add New Record

Facility Name:

Facility Type: --Select--

Street Address:

City:

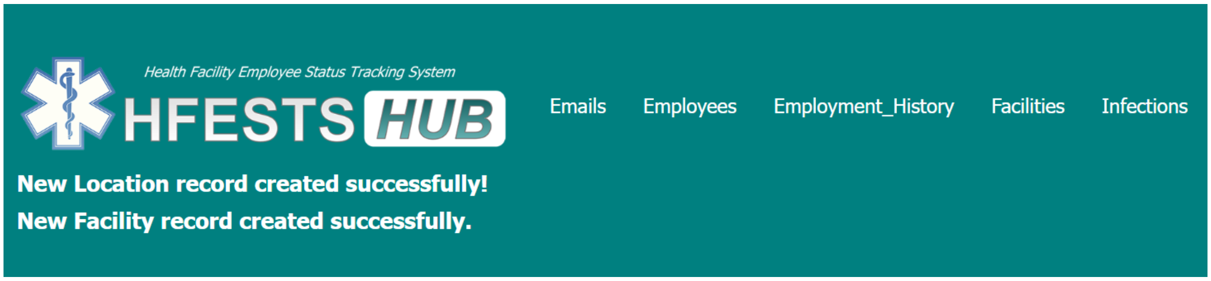
Postal Code:

Capacity:

Web URL:

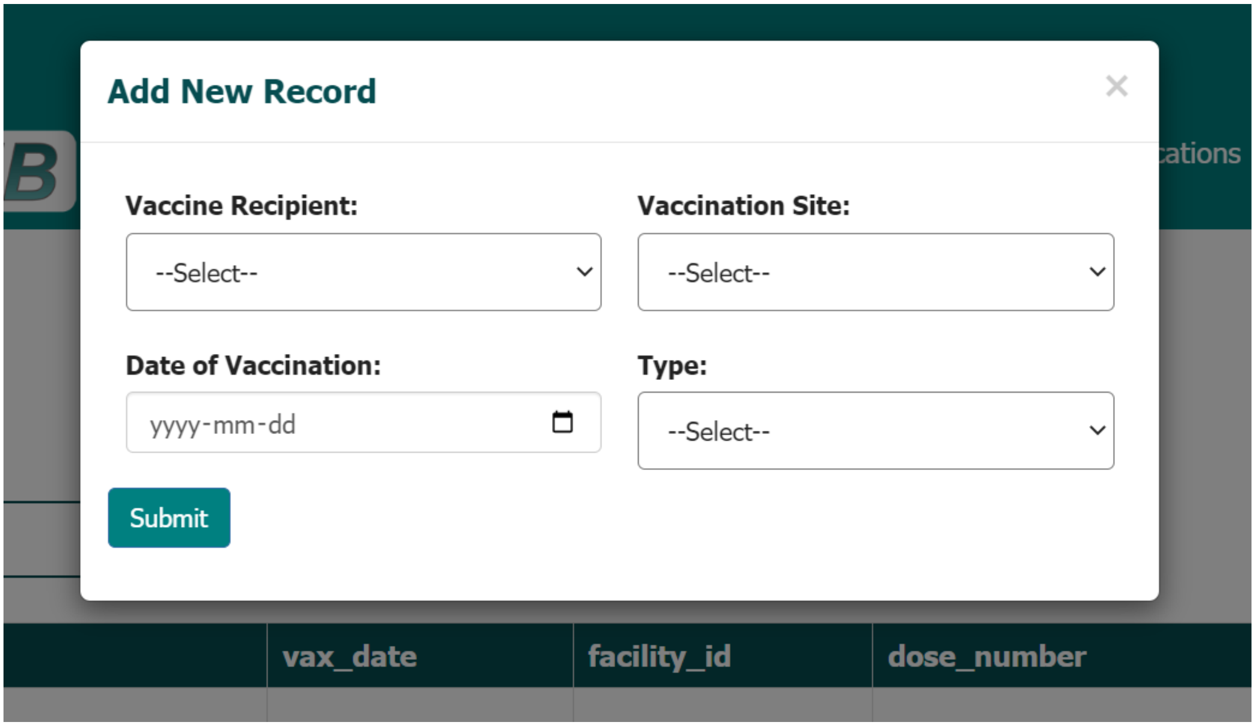
Submit

Adding a new facility record also automatically creates a new Location record to store the Facility's address (if not already tracked in the system).



New Location record created successfully!
New Facility record created successfully.

3. **Vaccines Window:** Allows a new vaccination to be recorded in the system for a current employee as well as deletion and update rights for existing Vaccine records. The system automatically assigns the dose number based on the number of previously tracked vaccinations against the Employee's ID.



Add New Record

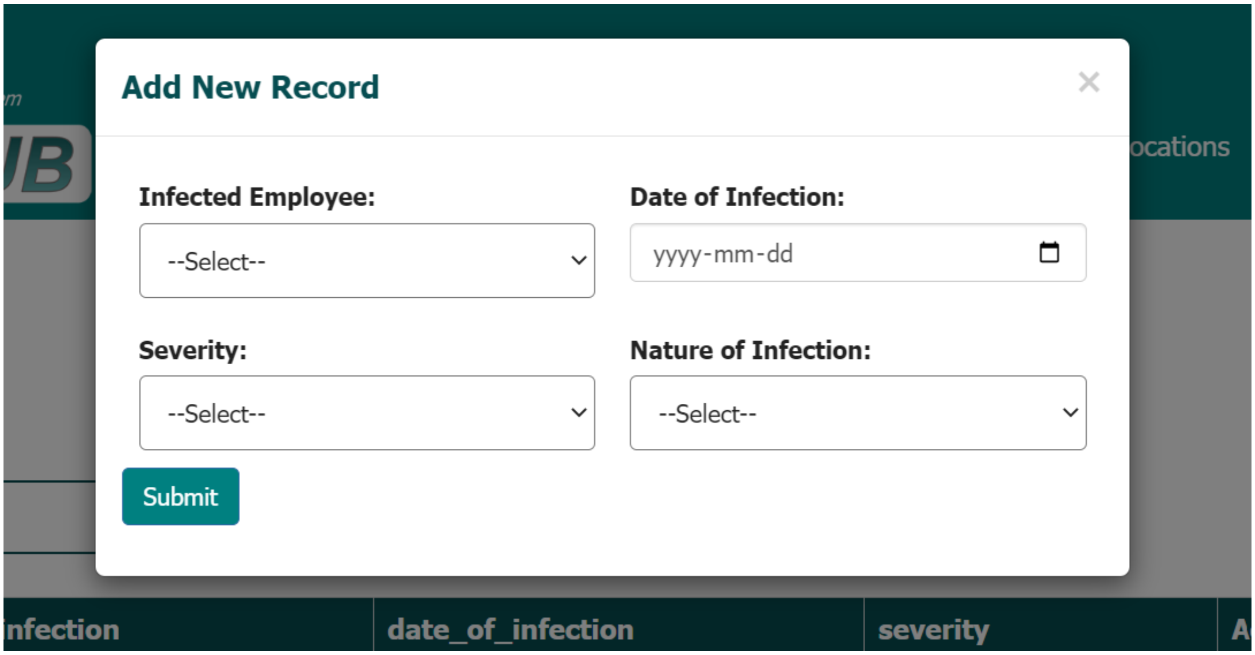
Vaccine Recipient: --Select--

Vaccination Site: --Select--

Date of Vaccination: yyyy-mm-dd

Type: --Select--

4. **Infections Window:** Allows a new infection to be recorded in the system for a current employee as well as deletion and update rights for existing Infection records.



Add New Record

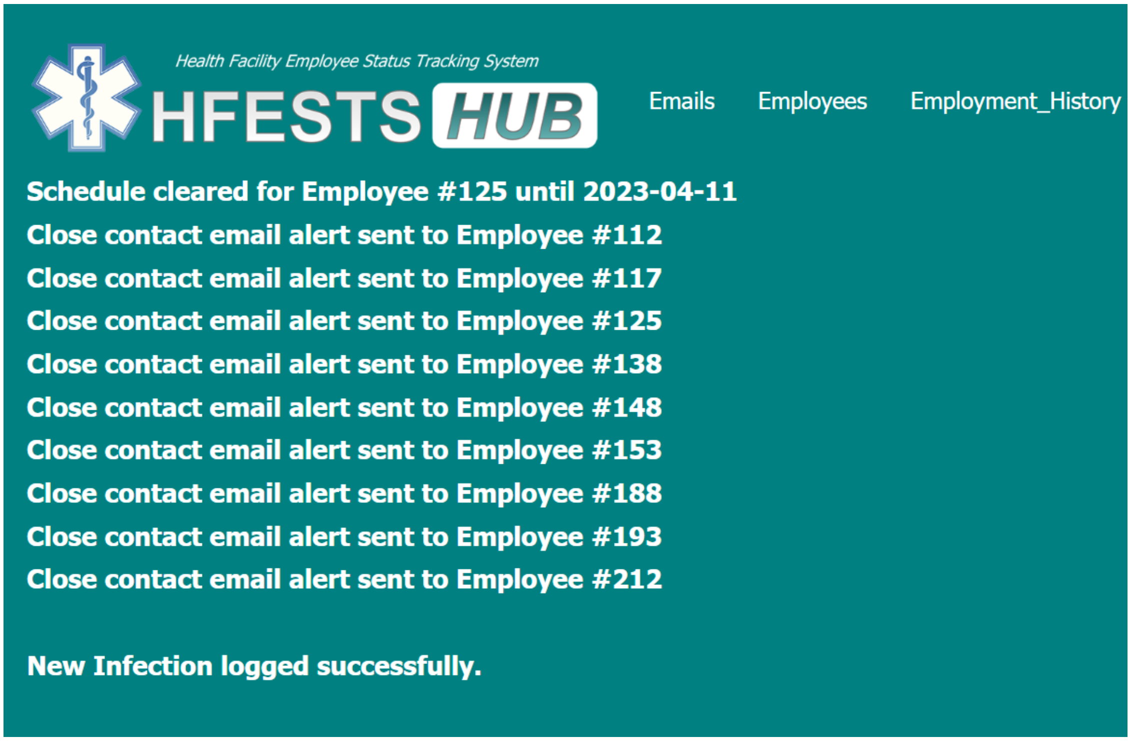
Infected Employee: --Select--

Date of Infection: yyyy-mm-dd

Severity: --Select--

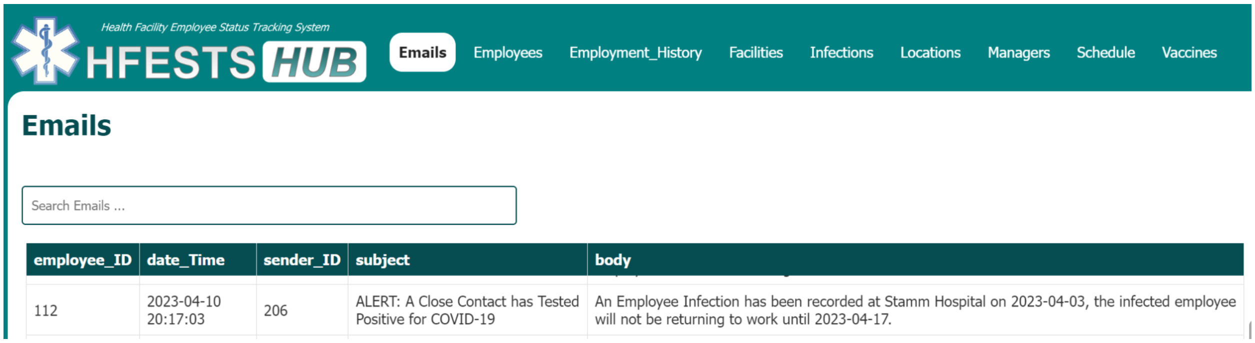
Nature of Infection: --Select--

When an Infection is logged for either a Nurse or Doctor, the system automatically cancels the scheduled shifts for the infected employee for two weeks from the date of the infection. The system will also query the database for a list of all Nurses and Doctors who worked at the same facility on the same day as the infected employee within two weeks of the reported infection and send them an alert by email. Below is an example of an alert email, logged in the Emails table.



Schedule cleared for Employee #125 until 2023-04-11
Close contact email alert sent to Employee #112
Close contact email alert sent to Employee #117
Close contact email alert sent to Employee #125
Close contact email alert sent to Employee #138
Close contact email alert sent to Employee #148
Close contact email alert sent to Employee #153
Close contact email alert sent to Employee #188
Close contact email alert sent to Employee #193
Close contact email alert sent to Employee #212

New Infection logged successfully.

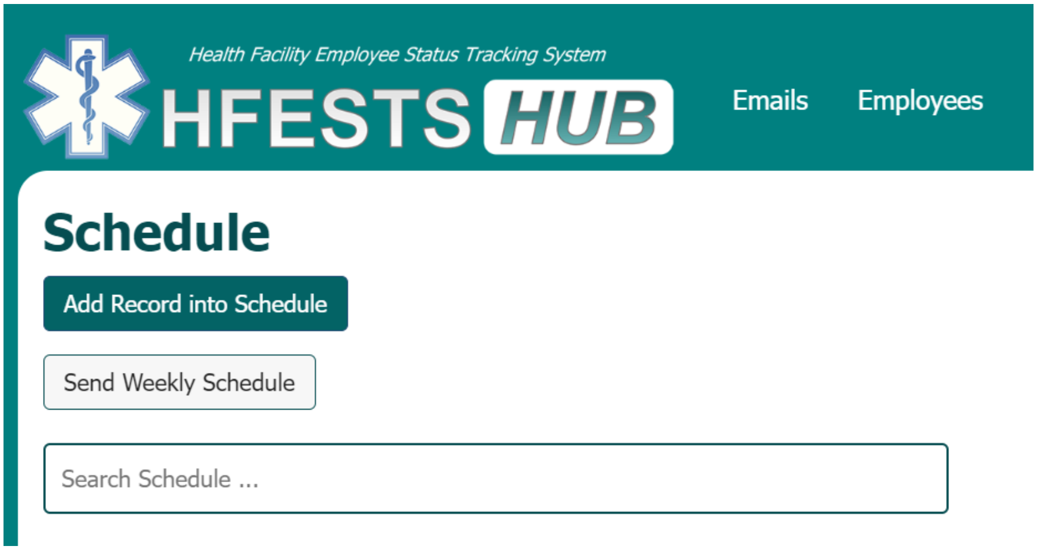


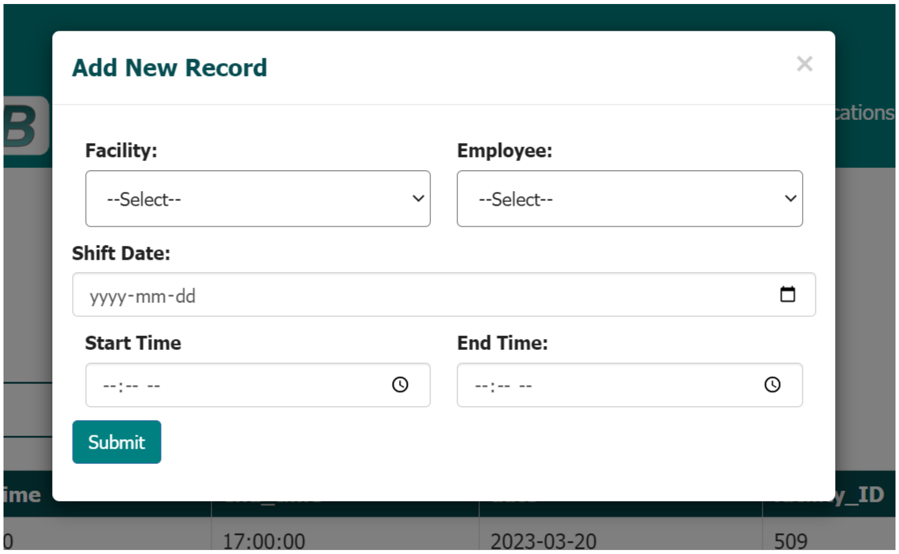
Emails

Search Emails ...

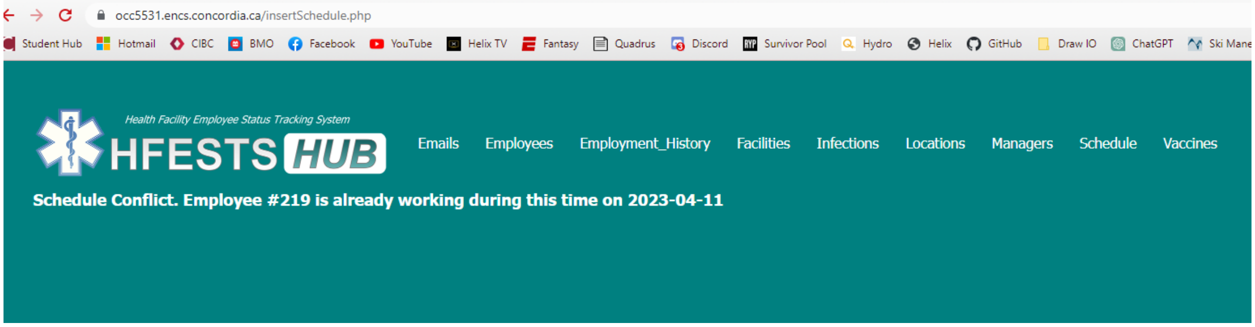
employee_ID	date_Time	sender_ID	subject	body
112	2023-04-10 20:17:03	206	ALERT: A Close Contact has Tested Positive for COVID-19	An Employee Infection has been recorded at Stamm Hospital on 2023-04-03, the infected employee will not be returning to work until 2023-04-17.

5. **Schedule Window:** Allows managers to schedule shifts for their employees and send out email alerts to all active employees for a given facility for the upcoming week.



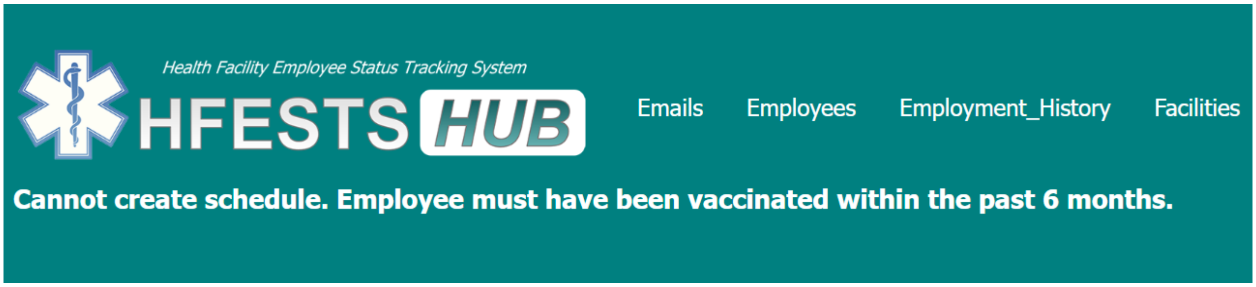


When scheduling an employee, the system will reject requests for a new shift if the employee is already scheduled on the same day within an hour of the new shift.



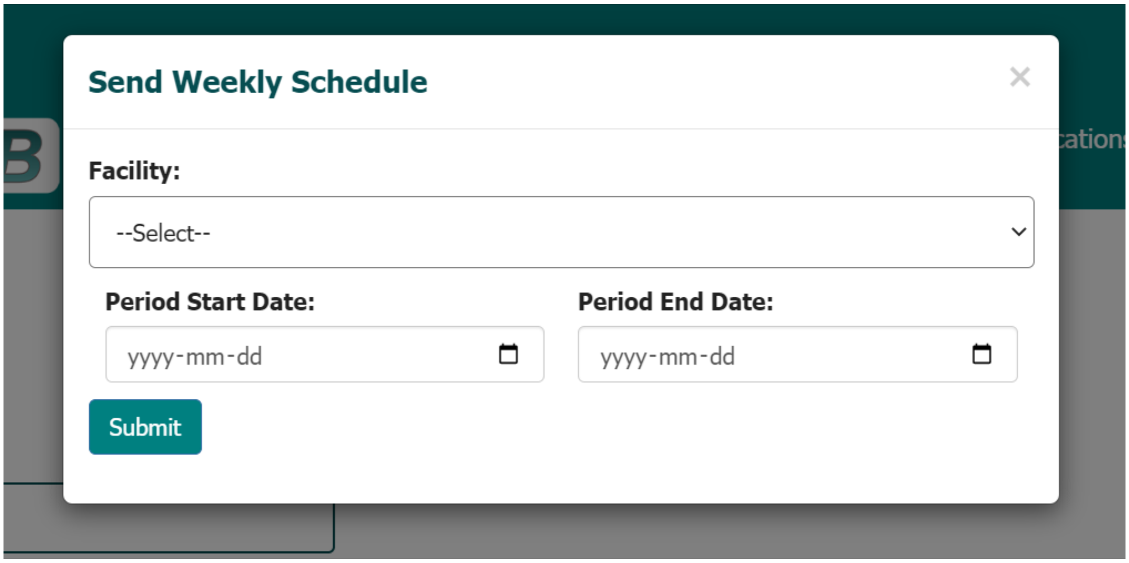
Schedule Conflict. Employee #219 is already working during this time on 2023-04-11

As well, the system rejects schedule requests for employees who are either unvaccinated or have not received a vaccine within the last 6 months.



Cannot create schedule. Employee must have been vaccinated within the past 6 months.

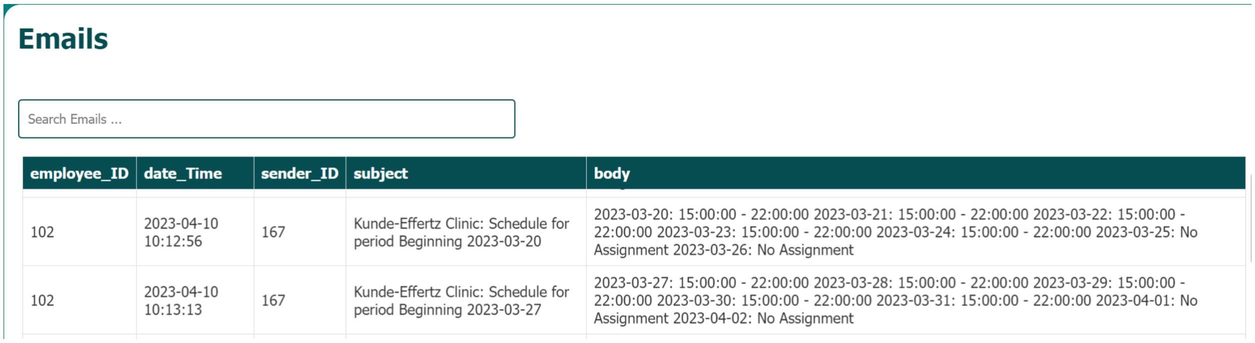
To send a weekly schedule, the manager clicks the 'Send Weekly Schedule' button and selects their Facility from the dropdown along with the start and end dates of the schedule period. Emails are then auto-generated and sent to the active employees of the selected facility.



Facility: --Select--

Period Start Date: yyyy-mm-dd

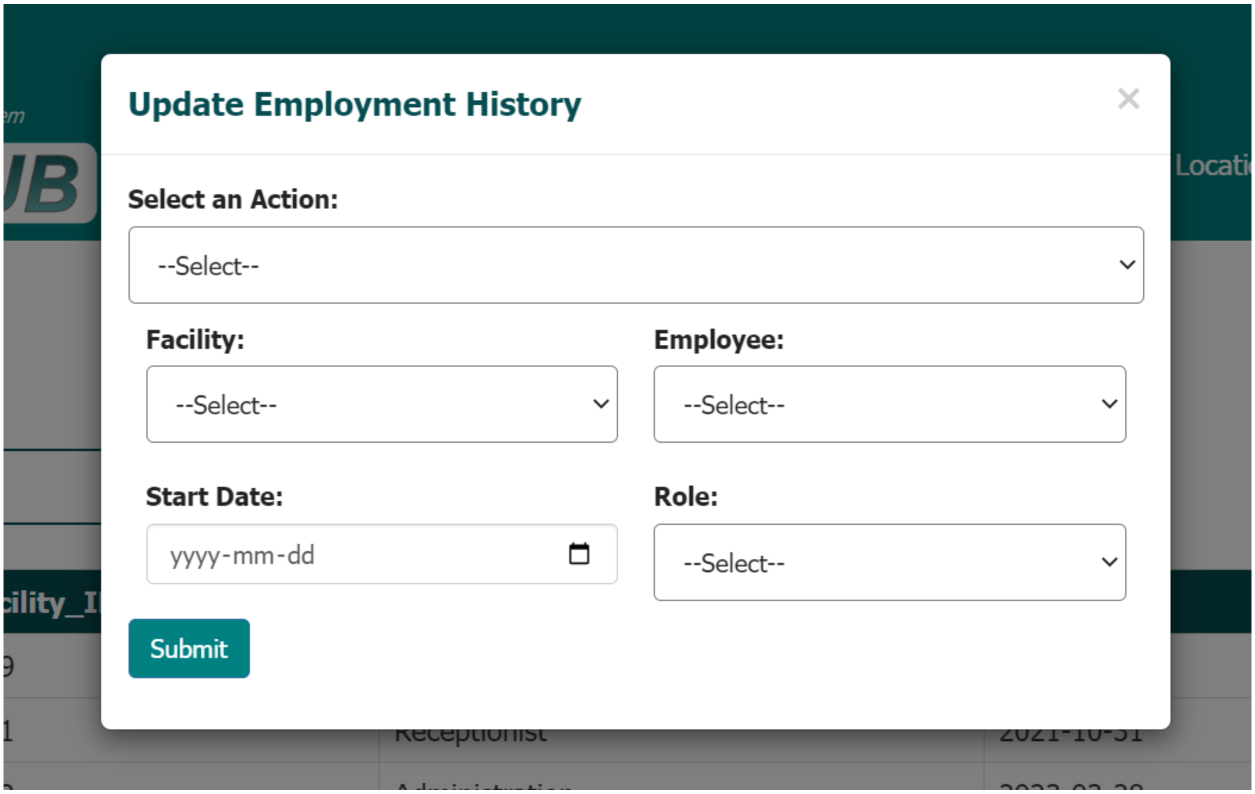
Period End Date: yyyy-mm-dd



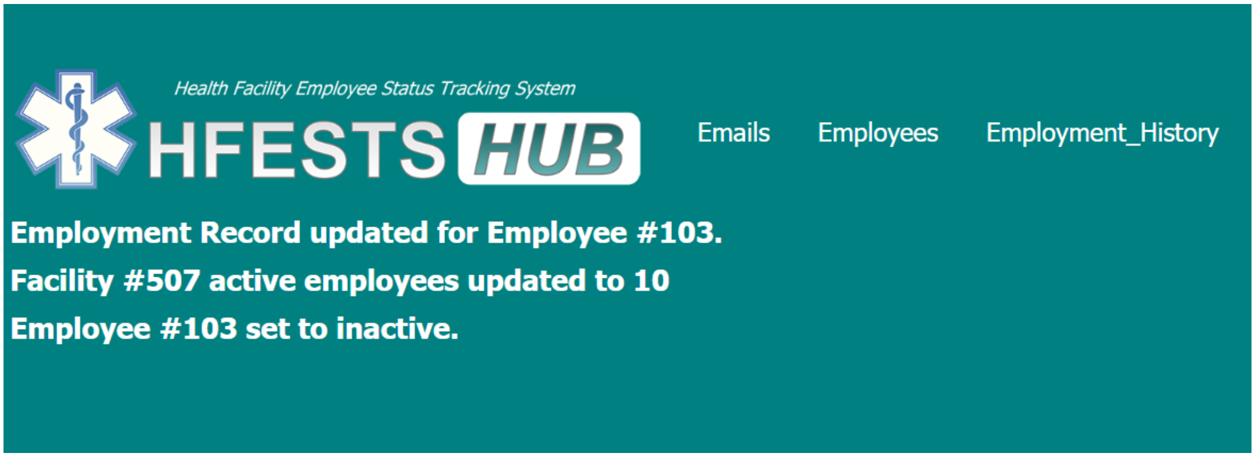
Emails

employee_ID	date_Time	sender_ID	subject	body
102	2023-04-10 10:12:56	167	Kunde-Effertz Clinic: Schedule for period Beginning 2023-03-20	2023-03-20: 15:00:00 - 22:00:00 2023-03-21: 15:00:00 - 22:00:00 2023-03-22: 15:00:00 - 22:00:00 2023-03-23: 15:00:00 - 22:00:00 2023-03-24: 15:00:00 - 22:00:00 2023-03-25: No Assignment 2023-03-26: No Assignment
102	2023-04-10 10:13:13	167	Kunde-Effertz Clinic: Schedule for period Beginning 2023-03-27	2023-03-27: 15:00:00 - 22:00:00 2023-03-28: 15:00:00 - 22:00:00 2023-03-29: 15:00:00 - 22:00:00 2023-03-30: 15:00:00 - 22:00:00 2023-03-31: 15:00:00 - 22:00:00 2023-04-01: No Assignment 2023-04-02: No Assignment

6. **Employment History Window**: Allows the user to terminate an open employment record or open a new employment record for an existing employee.



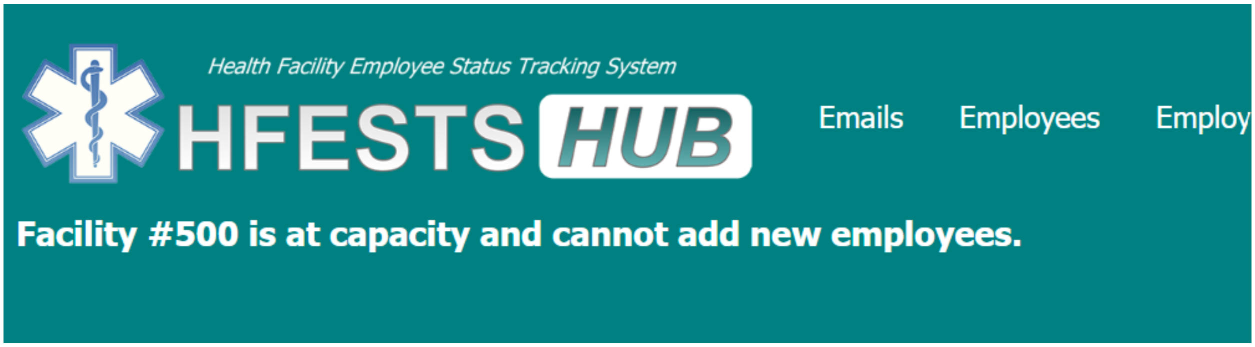
If the user terminates an existing employment record, the system closes the record with the current date as the End Date, decrements the number of active employees against the affected facility, and sets the employee's status to inactive if no other employment record remains open for the employee.



Employment Record updated for Employee #103.
Facility #507 active employees updated to 10
Employee #103 set to inactive.

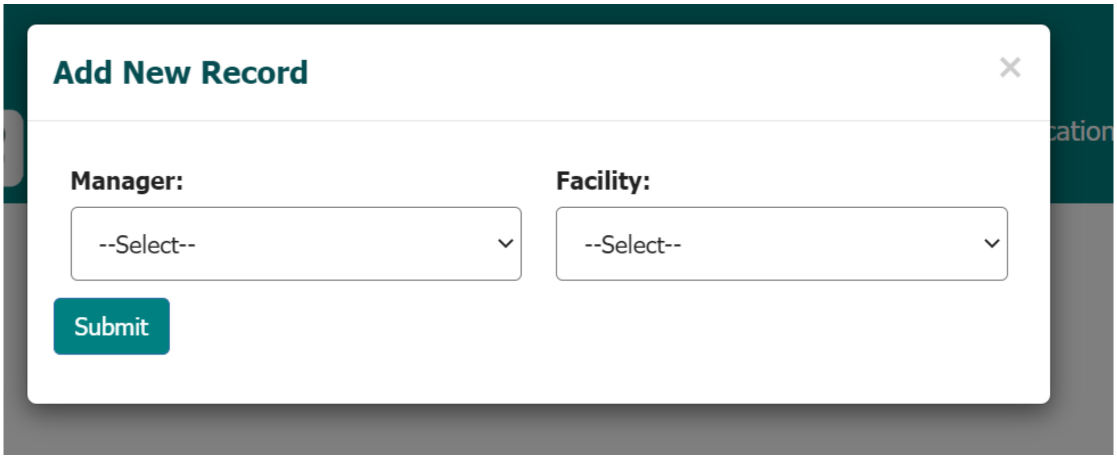
Similarly, if the user chooses to add a new employment record, the employee's status is refreshed to active and the facility's active employee count is incremented.

If the chosen facility is currently at capacity, the employee assignment is rejected.



Facility #500 is at capacity and cannot add new employees.

7. **Managers Window:** Allows a manager to be assigned to an existing facility and automatically updates the role of the new manager to 'Administration'.

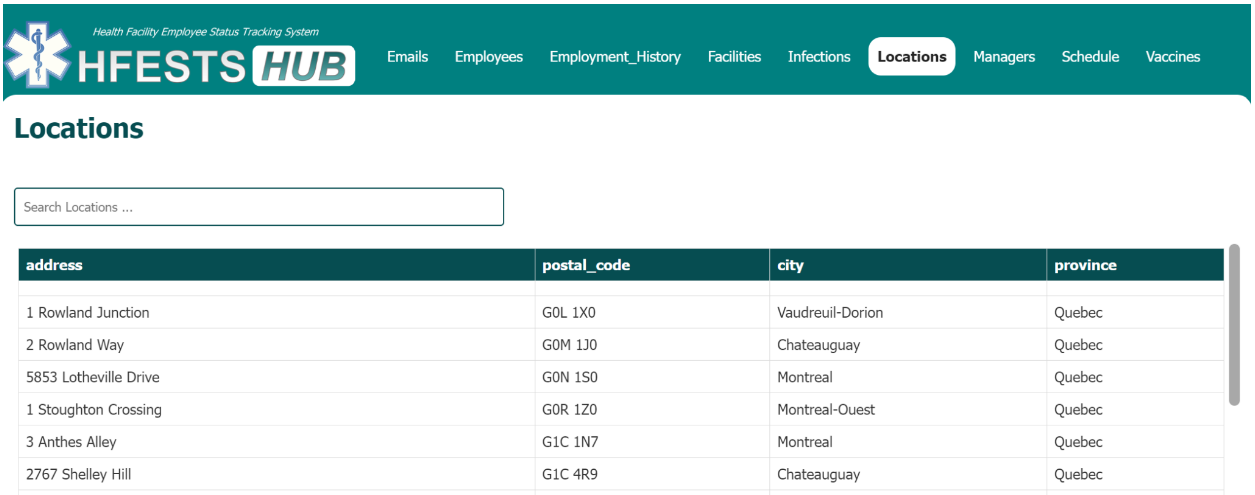


Add New Record

Manager: --Select--

Facility: --Select--

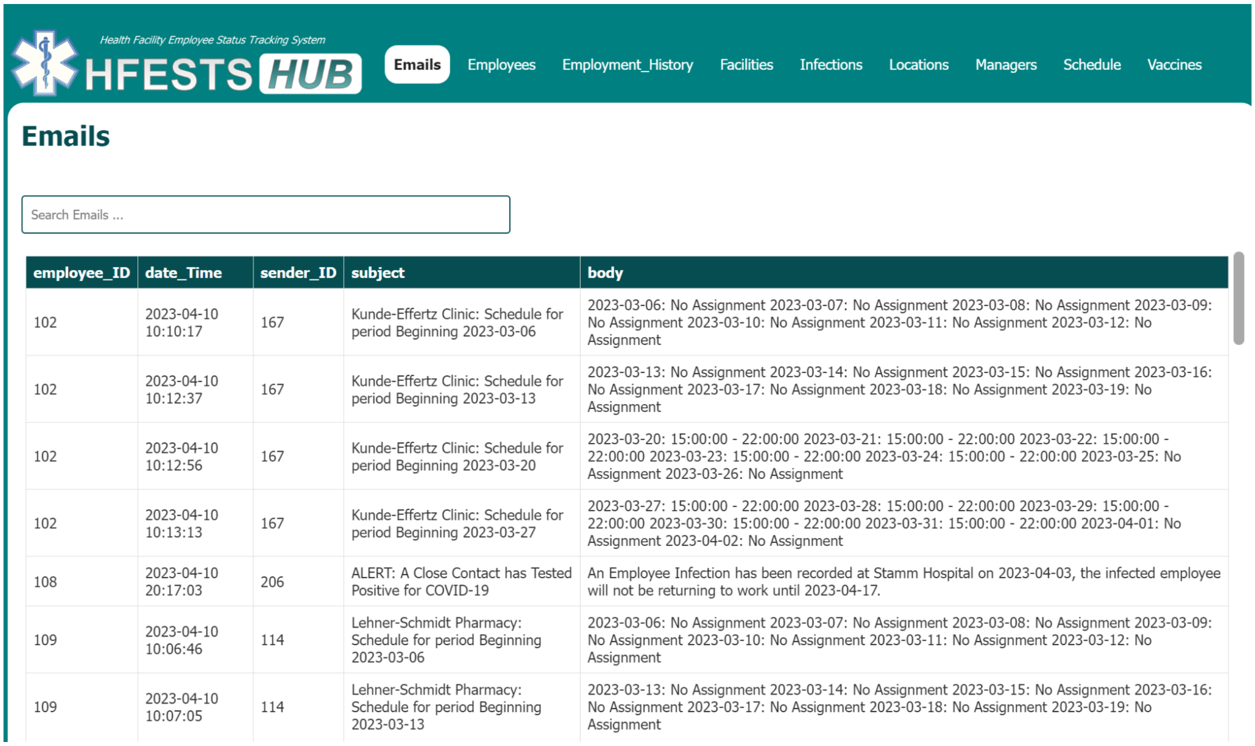
8. **Locations Window:** Displays all Facility & Employee addresses in the HFESTS Hub.



Locations

address	postal_code	city	province
1 Rowland Junction	G0L 1X0	Vaudreuil-Dorion	Quebec
2 Rowland Way	G0M 1J0	Chateauguay	Quebec
5853 Lotheville Drive	G0N 1S0	Montreal	Quebec
1 Stoughton Crossing	G0R 1Z0	Montreal-Ouest	Quebec
3 Anthes Alley	G1C 1N7	Montreal	Quebec
2767 Shelley Hill	G1C 4R9	Chateauguay	Quebec

9. **Emails Window:** Displays all emails logged by the HFESTS Hub alert system.



Emails

employee_ID	date_Time	sender_ID	subject	body
102	2023-04-10 10:10:17	167	Kunde-Effertz Clinic: Schedule for period Beginning 2023-03-06	2023-03-06: No Assignment 2023-03-07: No Assignment 2023-03-08: No Assignment 2023-03-09: No Assignment 2023-03-10: No Assignment 2023-03-11: No Assignment 2023-03-12: No Assignment
102	2023-04-10 10:12:37	167	Kunde-Effertz Clinic: Schedule for period Beginning 2023-03-13	2023-03-13: No Assignment 2023-03-14: No Assignment 2023-03-15: No Assignment 2023-03-16: No Assignment 2023-03-17: No Assignment 2023-03-18: No Assignment 2023-03-19: No Assignment
102	2023-04-10 10:12:56	167	Kunde-Effertz Clinic: Schedule for period Beginning 2023-03-20	2023-03-20: 15:00:00 - 22:00:00 2023-03-21: 15:00:00 - 22:00:00 2023-03-22: 15:00:00 - 22:00:00 2023-03-23: 15:00:00 - 22:00:00 2023-03-24: 15:00:00 - 22:00:00 2023-03-25: No Assignment 2023-03-26: No Assignment
102	2023-04-10 10:13:13	167	Kunde-Effertz Clinic: Schedule for period Beginning 2023-03-27	2023-03-27: 15:00:00 - 22:00:00 2023-03-28: 15:00:00 - 22:00:00 2023-03-29: 15:00:00 - 22:00:00 2023-03-30: 15:00:00 - 22:00:00 2023-03-31: 15:00:00 - 22:00:00 2023-04-01: No Assignment 2023-04-02: No Assignment
108	2023-04-10 20:17:03	206	ALERT: A Close Contact has Tested Positive for COVID-19	An Employee Infection has been recorded at Stamm Hospital on 2023-04-03, the infected employee will not be returning to work until 2023-04-17.
109	2023-04-10 10:06:46	114	Lehner-Schmidt Pharmacy: Schedule for period Beginning 2023-03-06	2023-03-06: No Assignment 2023-03-07: No Assignment 2023-03-08: No Assignment 2023-03-09: No Assignment 2023-03-10: No Assignment 2023-03-11: No Assignment 2023-03-12: No Assignment
109	2023-04-10 10:07:05	114	Lehner-Schmidt Pharmacy: Schedule for period Beginning 2023-03-13	2023-03-13: No Assignment 2023-03-14: No Assignment 2023-03-15: No Assignment 2023-03-16: No Assignment 2023-03-17: No Assignment 2023-03-18: No Assignment 2023-03-19: No Assignment

Contribution Log

Name	Student ID	Contribution
Emma Langlois	40254315	→ Schema Design & Refinement → Mock Data Generation → SQL Query Design & Testing → SQL Trigger Implementation → Report Template
Alex Newman	40183409	→ Schema Design & Refinement → Mock Data Generation → E/R Diagram & Relational conversion → FD Analysis & Normalization → Interface Functionality Implementation & Testing

Eric Spensieri	26997252	➔ Schema Design & Refinement ➔ Mock Data Generation ➔ Interface Design & Development ➔ Interface Functionality Implementation & Testing