INF1343 Assignment 2: RA, SQL and Accessing a Maternity Clinic Database

Ngoné Lo

1. Patient ID, reason and scheduled time of all scheduled visit(s) for Doctor Avery on March 12, 2019

RA:

$\boldsymbol{\pi}_{\text{pid, reason, TIME(scheduled\_datetime)}}(\boldsymbol{\sigma}_{\text{lname = "Avery" } \wedge \text{ DATE(scheduled\_datetime) = "2019-03-12"}}(staff \bowtie visits))$

SQL:

```
SELECT pid AS patient_id, reason, TIME(scheduled_datetime) AS scheduled_time
FROM visits
WHERE sid = (SELECT sid FROM staff WHERE lname = "Avery") AND DATE(scheduled_datetime)="2019-03-12";
```

OR

```
SELECT pid AS patient_id, reason, TIME(scheduled_datetime) AS scheduled_time
FROM visits NATURAL JOIN staff
WHERE lname = "Avery" AND DATE(scheduled_datetime)="2019-03-12";
```

Result:

Result Grid | Filter Rows:

patient_id	reason	scheduled_time
2018448	ultrasound	10:30:00
2017332	checkup/followup	15:30:00

2. Baby id, delivery type, delivery date, mother's first name and mother's last name of female babies born under Doctor Yang

RA:

$\rho(yang\_sid, \boldsymbol{\pi}_{\text{sid}}(\boldsymbol{\sigma}_{\text{lname="Yang"}}(staff)))$

$\boldsymbol{\pi}_{\text{bid, delivery\_type, DATE(delivery\_datetime), fname, lname}}((\boldsymbol{\sigma}_{\text{babies.gender="F"}}(babies \bowtie_{\text{babies.pid=patients.pid}} patients)) \bowtie yang\_sid)$

SQL:

```
SELECT bid AS baby_id, delivery_type, DATE(delivery_datetime) AS birth_date, fname AS mother_first_name,
        lname AS mother_last_name
FROM babies AS B INNER JOIN patients AS P ON B.pid = P.pid
WHERE sid = (SELECT sid FROM staff WHERE lname = "Yang") AND B.gender="F";
```

Result:

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

baby_id	delivery_type	birth_date	mother_first_name	mother_last_name
2017841	natural	2017-04-08	Aorohi	Kabra
2018726	cesarean	2018-11-24	Karen	Nickerson
2018727	cesarean	2018-11-24	Karen	Nickerson

3. The day with the highest number of visits in March 2019 and its visit count

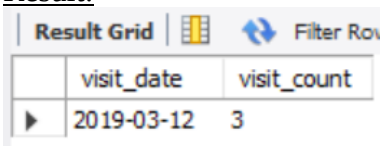
RA: This query cannot be translated in relational algebra because it involves aggregation.

SQL:

```
CREATE
   ALGORITHM = UNDEFINED
   DEFINER = `root`@`localhost`
   SQL SECURITY DEFINER
VIEW `march_visits` AS
   SELECT
      CAST(`v`.`checkedin_datetime` AS DATE) AS `visit_date`,
      COUNT(0) AS `visit_count`
   FROM
      `visits` `v`
   WHERE
      ((MONTH(`v`.`checkedin_datetime`) = '03')
         AND (YEAR(`v`.`checkedin_datetime`) = '2019'))
   GROUP BY CAST(`v`.`checkedin_datetime` AS DATE)

SELECT V1.visit_date, v1.visit_count FROM march_visits AS V1
WHERE V1.visit_count >= (SELECT MAX(V2.visit_count) FROM march_visits AS V2);
```

Result:



visit_date	visit_count
2019-03-12	3

4. Admission history of the patient(s) named Rachael Sloan

RA:

$(\pi_{\text{sid}}(\sigma_{\text{fname="Rachael"} \wedge \text{lname = "Sloan"}}(patients))) \bowtie admissions$

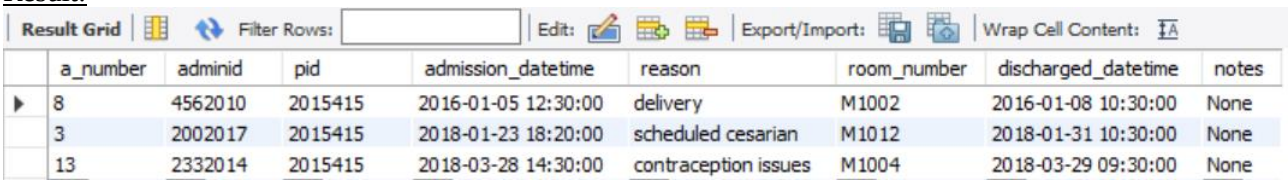
SQL:

```
SELECT * FROM admissions WHERE pid = (SELECT pid FROM patients
                                      WHERE fname = "Rachael" AND lname = "Sloan");
```

OR

```
SELECT * FROM admissions NATURAL JOIN (SELECT pid FROM patients
                                       WHERE fname = "Rachael" AND lname = "Sloan") AS P;
```

Result:



a_number	adminid	pid	admission_datetime	reason	room_number	discharged_datetime	notes
8	4562010	2015415	2016-01-05 12:30:00	delivery	M1002	2016-01-08 10:30:00	None
3	2002017	2015415	2018-01-23 18:20:00	scheduled cesarian	M1012	2018-01-31 10:30:00	None
13	2332014	2015415	2018-03-28 14:30:00	contraception issues	M1004	2018-03-29 09:30:00	None

5. Treatment records of all patients with the last name "Nickerson" who gave birth at the clinic in 2018

RA:

$\rho(pid\_mother\_2018, \boldsymbol{\pi}_{\text{pid}}(\boldsymbol{\sigma}_{\text{YEAR(delivery\_datetime)="2018"}}(babies))$

$\rho(pid\_patients\_Nickerson, \boldsymbol{\pi}_{\text{pid}}(\boldsymbol{\sigma}_{\text{lname="Nickerson"}}(patients))$

$treat \bowtie pid\_mother\_2018 \bowtie pid\_patients\_Nickerson$

SQL:

```
SELECT * FROM treat
WHERE pid IN (SELECT pid FROM patients WHERE lname = "Nickerson") AND
        pid IN (SELECT pid FROM babies WHERE YEAR(delivery_datetime)="2018");
```

OR

```
SELECT DISTINCT *
FROM treat NATURAL JOIN (SELECT pid FROM patients WHERE lname = "Nickerson") AS PPID
            NATURAL JOIN (SELECT pid FROM babies WHERE YEAR(delivery_datetime)="2018") AS BPID;
```

Result:

	sid	pid	treat_datetime	reason	treatment	pregnancy stage	notes
▶	2462015	2017332	2018-05-08 17:25:00	checkin		2nd trimester	None
	1092012	2018034	2018-04-26 12:30:00	checkin	medication	1st trimester	None
	2462015	2018034	2018-09-02 12:15:00	checkin		2nd trimester	None

6. List of unique first name and last name of the patient(s) treated by Doctor Shepperd

RA:

$\rho(shepperd\_treated\_patients, \boldsymbol{\pi}_{\text{pid}}(\boldsymbol{\sigma}_{\text{lname="Shepperd"}}(staff \bowtie treat)))$

$\boldsymbol{\pi}_{\text{fname, lname}}(patients \bowtie shepperd\_treated\_patients)$

SQL:

```
SELECT DISTINCT fname AS patient_first_name, lname AS patient_last_name
FROM patients
WHERE pid IN (SELECT pid
                FROM treat
                WHERE sid = (SELECT sid FROM staff WHERE lname="Shepperd"));
```

OR

```
SELECT DISTINCT fname AS patient_first_name, lname AS patient_last_name
FROM patients NATURAL JOIN (SELECT pid
                            FROM treat NATURAL JOIN staff
                            WHERE lname="Shepperd") AS TPID;
```

Result:

	patient_first_name	patient_last_name
▶	Pamela	Nickerson
	Karen	Nickerson
	Liliana	Nikererson
	Erin	Liu
	David	Thompson

7. Visits checked in by admin Morris for non-consultation and non-checkup/followup reasons

RA:

$\rho(morris\_adminid, \pi_{\text{adminid}}(\sigma_{\text{fname="Morris"}}(admins \bowtie staff)))$

$\sigma_{\text{reason}\neq\text{“consultation”} \wedge \text{reason}\neq\text{“checkup/followup”} \wedge \text{checkin\_datetime}<>\text{NULL}}$ (visits ⋈ morris_adminid)

SQL:

```
SELECT * FROM visits
WHERE adminid = (SELECT adminid FROM admins NATURAL JOIN staff
                 WHERE lname = "Morris")
      AND reason NOT IN ("consultation", "checkup/followup")
      AND checkedin_datetime IS NOT NULL;
```

OR

```
SELECT * FROM visits NATURAL JOIN (SELECT adminid
                                   FROM admins NATURAL JOIN staff
                                   WHERE lname = "Morris") AS A
WHERE reason NOT IN ("consultation", "checkup/followup") AND checkedin_datetime IS NOT NULL;
```

Result:

v_number	adminid	pid	sid	type	reason	checkedin_datetime	scheduled_datetime
100	4562010	2018448	2022014	appointement	ultrasound	2019-03-12 10:30:00	2019-03-12 10:30:00
104	4562010	2018561	2022014	walkin	contraception	2019-03-15 14:30:00	NULL
112	4562010	2018653	1092012	walkin	contraception	2018-12-27 15:00:00	NULL
115	4562010	2018034	2462015	appointment	ultrasound	2018-09-12 15:00:00	2018-09-12 15:00:00

8. The first name, last name and emergency contact information of the mother(s) of all babie(s) born by cesarean in 2018

RA:

$\pi_{\text{fname, lname, emergency\_contact\_name, emergency\_contact\_contact\_phone\_number, emergency\_contact\_relationship}}(\sigma_{\text{delivery\_type=“cesarean”} \wedge \text{YEAR(delivery\_datetime)=“2018”}}(patients \bowtie_{\text{patients.pid=babies.pid}} babies))$

SQL:

```
SELECT fname AS mother_first_name, lname AS mother_last_name,
       emergency_contact_name, emergency_contact_phone_number, emergency_contact_relationship
FROM patients
WHERE pid IN (SELECT pid FROM babies
              WHERE delivery_type = "cesarean" AND YEAR(delivery_datetime) = "2018");
```

OR

```
SELECT DISTINCT fname AS mother_first_name, lname AS mother_last_name,
                emergency_contact_name, emergency_contact_phone_number, emergency_contact_relationship
FROM patients AS P INNER JOIN babies AS B ON P.pid = B.pid
WHERE delivery_type = "cesarean" AND YEAR(delivery_datetime)="2018";
```

Result:

mother_first_name	mother_last_name	emergency_contact_name	emergency_contact_phone_number	emergency_contact_relationship
Rachael	Sloan	Kurt Sloan	457-764-7843	Husband
Karen	Nickerson	Sara Avery	457-563-5686	Friend

9. The sid, first name and last name of nurse(s) who cared for patient(s) named Rachael Sloan between January 24, 2018 and January 27, 2018

RA:

$\rho(sid\_nurse\_care\_rachael, \boldsymbol{\pi}_{\text{sid}}(\boldsymbol{\sigma}_{\text{lname = “Sloan” ∧ fname = “Rachael” ∧ DATE(admission\_datetime) >= “2018-01-24” ∧ DATE(admission\_datetime) <= “2018-01-27”}} (care\_for \bowtie patients)))$

$\boldsymbol{\pi}_{\text{sid, fname, lname}}(staff \bowtie sid\_nurse\_care\_rachael)$

SQL:

```
SELECT sid, fname AS nurse_first_name, lname AS nurse_last_name
FROM staff
WHERE sid IN (SELECT sid FROM care_for NATURAL JOIN patients
                    WHERE fname = "Rachael" AND lname = "Sloan"
                            AND DATE(care_datetime) BETWEEN "2018-01-24" AND "2018-01-27");
```

OR

```
SELECT DISTINCT sid, fname AS nurse_first_name, lname AS nurse_last_name
FROM staff NATURAL JOIN (SELECT sid FROM care_for NATURAL JOIN patients
                                    WHERE fname = "Rachael" AND lname = "Sloan"
                            AND DATE(care_datetime) BETWEEN "2018-01-24" AND "2018-01-27") AS C;
```

Result:

Result Grid | Filter Rows: | Ex

sid	nurse_first_name	nurse_last_name
3242010	Patience	De Luca
4552013	Carrie	Nelson
6022012	Christiane	Brown

10. The sid, first name, last name and number of visits of doctor(s) who got 3 or more visits from patient(s) referred to the clinic by a friend or family member ordered by the number of visits

RA: This query cannot be translated in relational algebra because it involves aggregation.

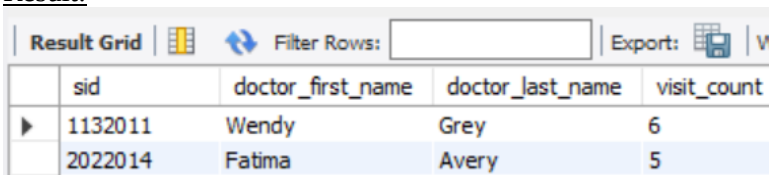
SQL:

```
SELECT S.sid, fname AS doctor_first_name, lname AS doctor_last_name, COUNT(S.sid) AS visit_count
FROM staff AS S INNER JOIN visits AS V ON S.sid = V.sid
WHERE pid IN (SELECT pid FROM patients WHERE referred_by IN ("Friend", "Family"))
GROUP BY S.sid
HAVING visit_count >= 3
ORDER BY visit_count DESC;
```

OR

```
SELECT S.sid, fname AS doctor_first_name, lname AS doctor_last_name, COUNT(S.sid) AS visit_count
FROM (staff AS S INNER JOIN visits AS V ON S.sid = V.sid)
        NATURAL JOIN (SELECT pid FROM patients WHERE referred_by IN ("Friend", "Family")) AS P
GROUP BY S.sid
HAVING visit_count >= 3
ORDER BY visit_count DESC;
```

Result:



sid	doctor_first_name	doctor_last_name	visit_count
1132011	Wendy	Grey	6
2022014	Fatima	Avery	5

Insertions

11. Insert record of nurse Carrie Nelson (4552013) caring for patient Erin Liu (2018448) on 2019-09-28 15:20:00 in the care_for relation

```
INSERT INTO care_for(care_datetime,sid,pid) VALUES ("2019-09-28 15:20:00",4552013,2018448);
```

12. Suppose we do have a new relation, freq_staff_patient, where we store the frquent interactions between staff and patient(s). The DDL to create freq_staff_patient is as follow:

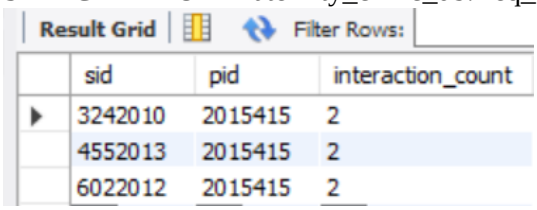
```
CREATE TABLE `freq_staff_patient` (
  `sid` int(7) NOT NULL,
  `pid` int(7) NOT NULL,
  `interaction_count` int(7) NOT NULL,
  PRIMARY KEY (`sid`,`pid`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

Insert the result of the query for nurse(s) and patient(s) with 2 or more interactions into freq_staff_patient

```
INSERT INTO freq_staff_patient (SELECT sid, pid, COUNT(*) AS interaction_count
                                FROM care_for
                                GROUP BY sid, pid
                                HAVING interaction_count >= 2
                               ORDER BY interaction_count);
```

Result:

```
SELECT * FROM maternity_clinic_db.freq_staff_patient;
```



sid	pid	interaction_count
3242010	2015415	2
4552013	2015415	2
6022012	2015415	2

Updates

13. Update left_date(current date), phone number(457-432-3684) and sid(7262011) for staff (nurse) Colin Walker. This should update Colin Walker's sid on the nurses and care_for relations as well.

```
UPDATE staff SET left_date = CURDATE(), phone_number = "457-432-3684", sid = "7262011"
WHERE fname="Colin" AND lname = "Walker";
```

14. Update all babies' delivery type that are natural to assisted childbirth. In other words, replace “natural” with “assisted childbirth” in babies’ delivery_type

```
UPDATE babies SET delivery_type = "assisted childbirth"
WHERE delivery_type="natural";
```

Deletions

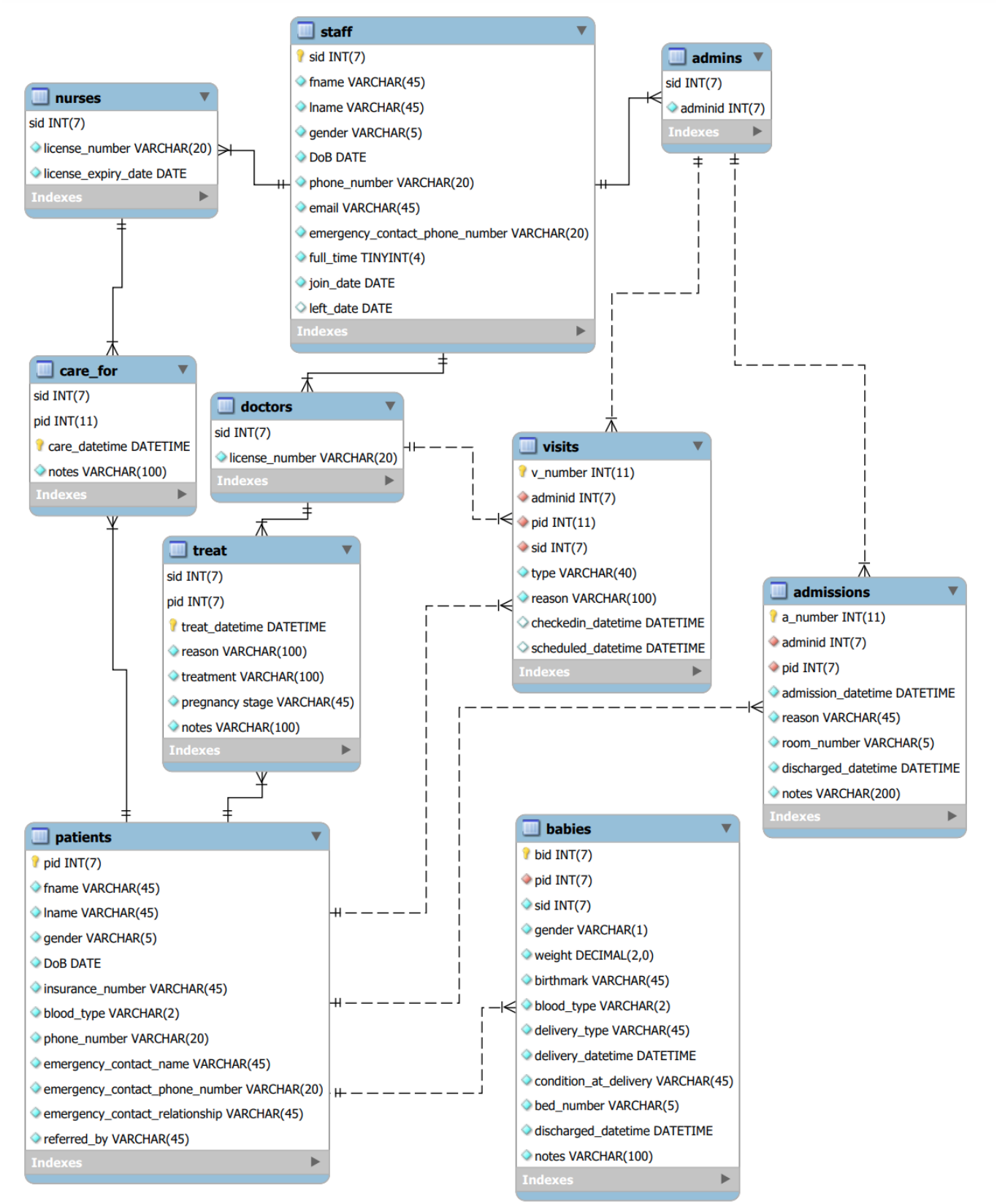
15. Delete admission record(s) of patient(s) whose first name start with J

```
DELETE FROM admissions WHERE pid IN (SELECT pid FROM patients WHERE fname LIKE "J%");
```

16. Delete all tuples in the relation freq_staff_patient. In other words, clear it!!!

```
DELETE FROM freq_staff_patient;
```

EERD of the Maternity Clinic Database



DDL of the Maternity Clinic Database Tables

```
CREATE TABLE `staff` (
  `sid` int(7) NOT NULL,
  `fname` varchar(45) NOT NULL,
  `lname` varchar(45) NOT NULL,
  `gender` varchar(5) NOT NULL,
  `DoB` date NOT NULL,
  `phone_number` varchar(20) NOT NULL,
  `email` varchar(45) NOT NULL,
  `emergency_contact_phone_number` varchar(20) NOT NULL,
  `full_time` tinyint(4) NOT NULL DEFAULT '1',
  `join_date` date NOT NULL,
  `left_date` date DEFAULT NULL,
  PRIMARY KEY (`sid`)
)

CREATE TABLE `doctors` (
  `sid` int(7) NOT NULL,
  `license_number` varchar(20) NOT NULL,
  PRIMARY KEY (`sid`),
  UNIQUE KEY `unique_license_number` (`license_number`),
  CONSTRAINT `doctor_is_staff` FOREIGN KEY (`sid`) REFERENCES `staff` (`sid`) ON DELETE
CASCADE ON UPDATE CASCADE
)

CREATE TABLE `nurses` (
  `sid` int(7) NOT NULL,
  `license_number` varchar(20) NOT NULL,
  `license_expiry_date` date NOT NULL,
  PRIMARY KEY (`sid`),
  UNIQUE KEY `unique_license_number` (`license_number`),
  CONSTRAINT `nurse_is_staff` FOREIGN KEY (`sid`) REFERENCES `staff` (`sid`) ON DELETE
CASCADE ON UPDATE CASCADE
)

CREATE TABLE `admins` (
  `sid` int(7) NOT NULL,
  `adminid` int(7) NOT NULL,
  PRIMARY KEY (`sid`),
  UNIQUE KEY `unique_adminid` (`adminid`),
  CONSTRAINT `admin_is_staff` FOREIGN KEY (`sid`) REFERENCES `staff` (`sid`) ON DELETE
CASCADE ON UPDATE CASCADE
)
```

```
CREATE TABLE `patients` (
  `pid` int(7) NOT NULL,
  `fname` varchar(45) NOT NULL,
  `lname` varchar(45) NOT NULL,
  `gender` varchar(5) NOT NULL DEFAULT 'F',
  `DoB` date NOT NULL,
  `insurance_number` varchar(45) NOT NULL,
  `blood_type` varchar(2) NOT NULL,
  `phone_number` varchar(20) NOT NULL,
  `emergency_contact_name` varchar(45) NOT NULL,
  `emergency_contact_phone_number` varchar(20) NOT NULL,
  `emergency_contact_relationship` varchar(45) NOT NULL,
  `referred_by` varchar(45) NOT NULL DEFAULT 'self',
  PRIMARY KEY (`pid`),
  UNIQUE KEY `unique_insurance_number` (`insurance_number`) /*!80000 INVISIBLE */
)

CREATE TABLE `babies` (
  `bid` int(7) NOT NULL,
  `pid` int(7) NOT NULL,
  `sid` int(7) NOT NULL,
  `gender` varchar(1) NOT NULL,
  `weight` decimal(2,0) NOT NULL,
  `birthmark` varchar(45) NOT NULL DEFAULT 'None',
  `blood_type` varchar(2) NOT NULL,
  `delivery_type` varchar(45) NOT NULL,
  `delivery_datetime` datetime NOT NULL,
  `condition_at_delivery` varchar(45) NOT NULL,
  `bed_number` varchar(5) NOT NULL,
  `discharged_datetime` datetime NOT NULL,
  `notes` varchar(100) NOT NULL DEFAULT 'None',
  PRIMARY KEY (`bid`),
  UNIQUE KEY `unique_mother_delivery_datetime` (`pid`,`delivery_datetime`),
  KEY `birthed_by_idx` (`pid`),
  CONSTRAINT `birthed_by` FOREIGN KEY (`pid`) REFERENCES `patients` (`pid`) ON DELETE
CASCADE ON UPDATE CASCADE
)
```

```
CREATE TABLE `visits` (
  `v_number` int(11) NOT NULL AUTO_INCREMENT,
  `adminid` int(7) NOT NULL,
  `pid` int(11) NOT NULL,
  `sid` int(7) NOT NULL,
  `type` varchar(40) NOT NULL,
  `reason` varchar(100) NOT NULL,
  `checkedin_datetime` datetime DEFAULT NULL,
  `scheduled_datetime` datetime DEFAULT NULL,
  PRIMARY KEY (`v_number`),
  UNIQUE KEY `unique_sid_checkedin` (`sid`,`checkedin_datetime`),
  UNIQUE KEY `unique_sid_scheduled` (`sid`,`scheduled_datetime`) /*!80000 INVISIBLE */,
  UNIQUE KEY `unique_pid_checkedin` (`pid`,`checkedin_datetime`) /*!80000 INVISIBLE */,
  UNIQUE KEY `unique_pid_scheduled` (`pid`,`scheduled_datetime`),
  KEY `visit_doctor_idx` (`sid`),
  KEY `visit_patient_idx` (`pid`),
  KEY `visit_admin_idx` (`adminid`),
  CONSTRAINT `visit_admin` FOREIGN KEY (`adminid`) REFERENCES `admins` (`adminid`) ON
DELETE CASCADE ON UPDATE CASCADE,
  CONSTRAINT `visit_doctor` FOREIGN KEY (`sid`) REFERENCES `doctors` (`sid`) ON DELETE
CASCADE ON UPDATE CASCADE,
  CONSTRAINT `visit_patient` FOREIGN KEY (`pid`) REFERENCES `patients` (`pid`) ON DELETE
CASCADE ON UPDATE CASCADE
)

CREATE TABLE `admissions` (
  `a_number` int(11) NOT NULL AUTO_INCREMENT,
  `adminid` int(7) NOT NULL,
  `pid` int(7) NOT NULL,
  `admission_datetime` datetime NOT NULL,
  `reason` varchar(45) NOT NULL,
  `room_number` varchar(5) NOT NULL,
  `discharged_datetime` datetime NOT NULL,
  `notes` varchar(200) NOT NULL DEFAULT 'None',
  PRIMARY KEY (`a_number`),
  UNIQUE KEY `unique_pid_admission_datetime` (`pid`,`admission_datetime`),
  KEY `admission_admin_idx` (`adminid`),
  KEY `admission_patient_idx` (`pid`),
  CONSTRAINT `admission_admin` FOREIGN KEY (`adminid`) REFERENCES `admins` (`adminid`)
ON DELETE CASCADE ON UPDATE CASCADE,
  CONSTRAINT `admission_patient` FOREIGN KEY (`pid`) REFERENCES `patients` (`pid`) ON
DELETE CASCADE ON UPDATE CASCADE
)
```

```
CREATE TABLE `treat` (
  `sid` int(7) NOT NULL,
  `pid` int(7) NOT NULL,
  `treat_datetime` datetime NOT NULL,
  `reason` varchar(100) NOT NULL,
  `treatment` varchar(100) NOT NULL DEFAULT 'N/A',
  `pregnancy stage` varchar(45) NOT NULL DEFAULT 'N/A',
  `notes` varchar(100) NOT NULL DEFAULT 'None',
  PRIMARY KEY (`sid`,`pid`,`treat_datetime`),
  UNIQUE KEY `unique_doctor_datetime` (`sid`,`treat_datetime`),
  KEY `doctor_treat_idx` (`sid`),
  KEY `patient_is_treated_by_idx` (`pid`),
  CONSTRAINT `doctor_treat` FOREIGN KEY (`sid`) REFERENCES `doctors` (`sid`) ON DELETE
CASCADE ON UPDATE CASCADE,
  CONSTRAINT `patient_is_treated` FOREIGN KEY (`pid`) REFERENCES `patients` (`pid`) ON
DELETE CASCADE ON UPDATE CASCADE
)

CREATE TABLE `care_for` (
  `sid` int(7) NOT NULL,
  `pid` int(11) NOT NULL,
  `care_datetime` datetime NOT NULL,
  `notes` varchar(100) NOT NULL DEFAULT 'None',
  PRIMARY KEY (`sid`,`pid`,`care_datetime`),
  KEY `patient_care_idx` (`pid`) /*!80000 INVISIBLE */,
  KEY `nurse_care_idx` (`sid`),
  CONSTRAINT `nurse_care` FOREIGN KEY (`sid`) REFERENCES `nurses` (`sid`) ON DELETE
CASCADE ON UPDATE CASCADE,
  CONSTRAINT `patient_care` FOREIGN KEY (`pid`) REFERENCES `patients` (`pid`) ON DELETE
CASCADE ON UPDATE CASCADE
)
```