PROJECT 1 PRESENTATION

Link to the Google Slides (for authors)

Note: the hidden slides represent the presentation outline provided by instructor (“guidelines”)

Project 1
ITIS 6120

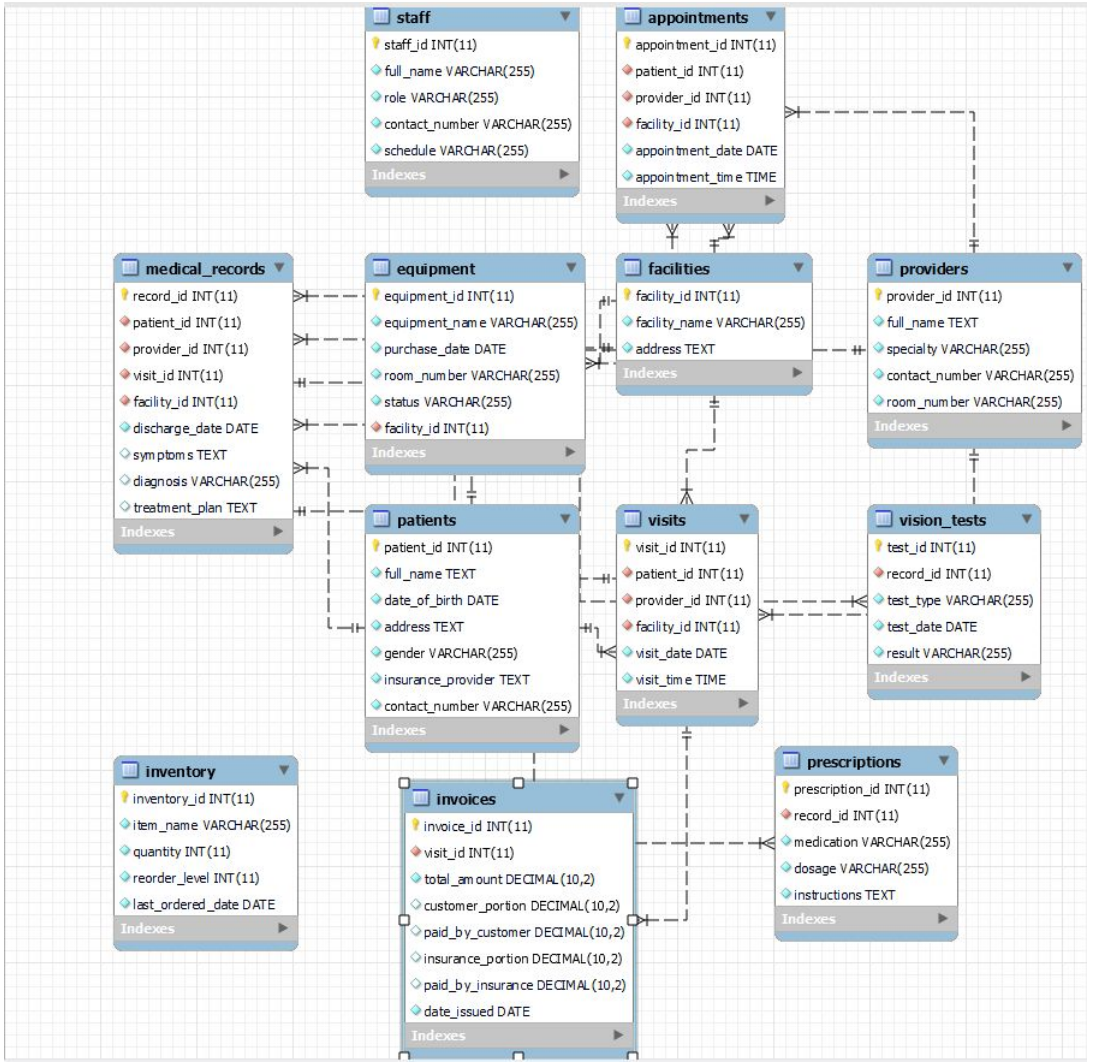
Kelsey Kornegay, Matt Barksdale, Stephanie Karp, Will Lapinel

- What is your database about? Articulate the database

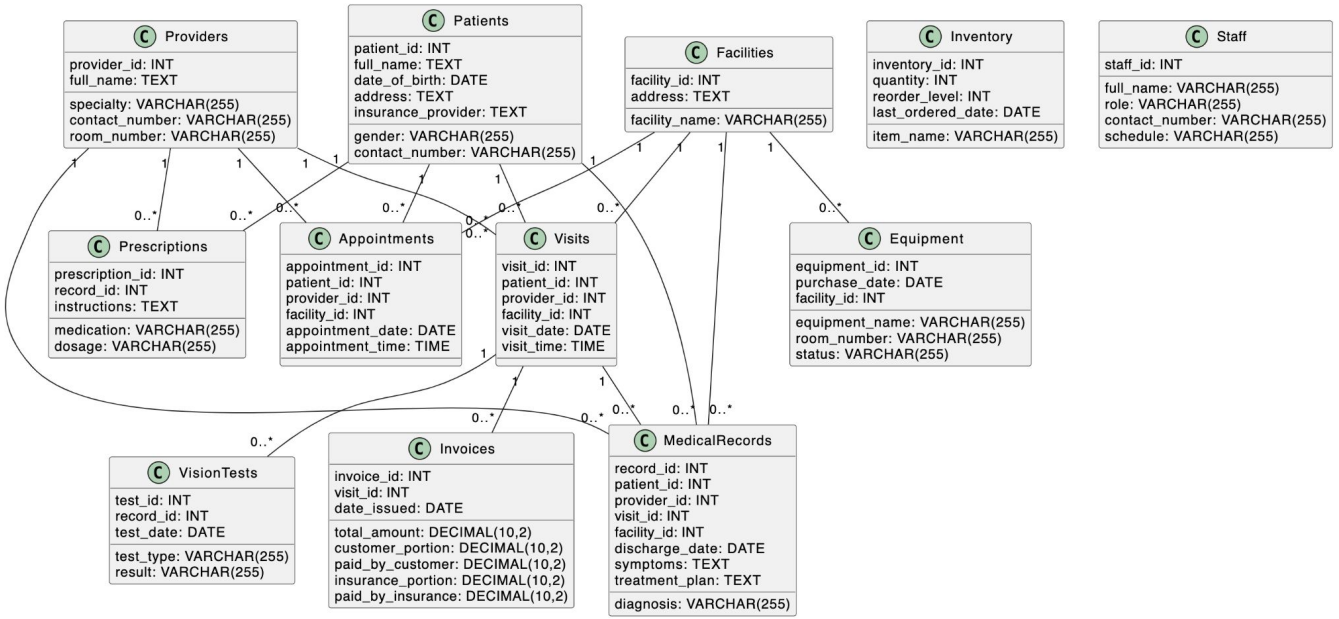
Ophthalmology Database

Our database is designed to support the operations and administration of a hypothetical ophthalmology practice

Entity Relationship Diagrams

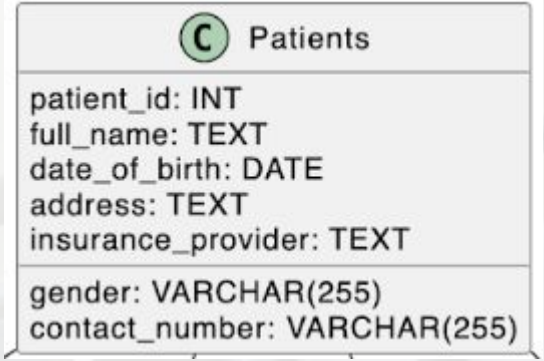


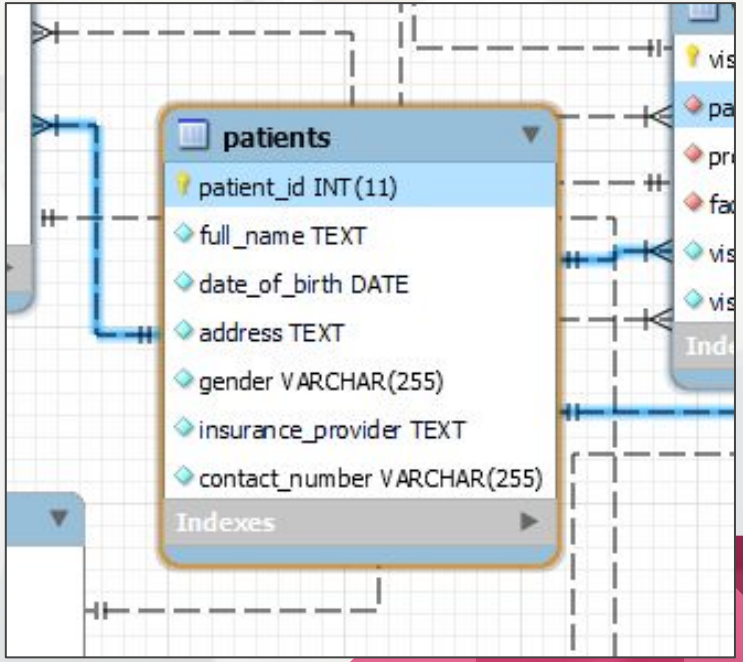
Unified Modeling Language Diagrams



Patients

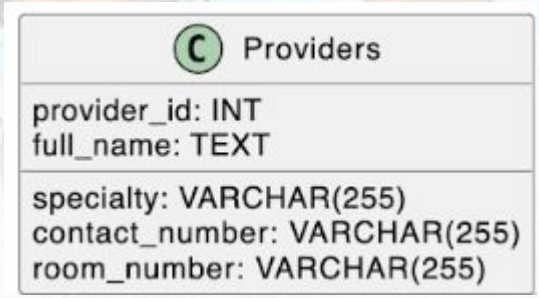
Tracking patients' demographic data

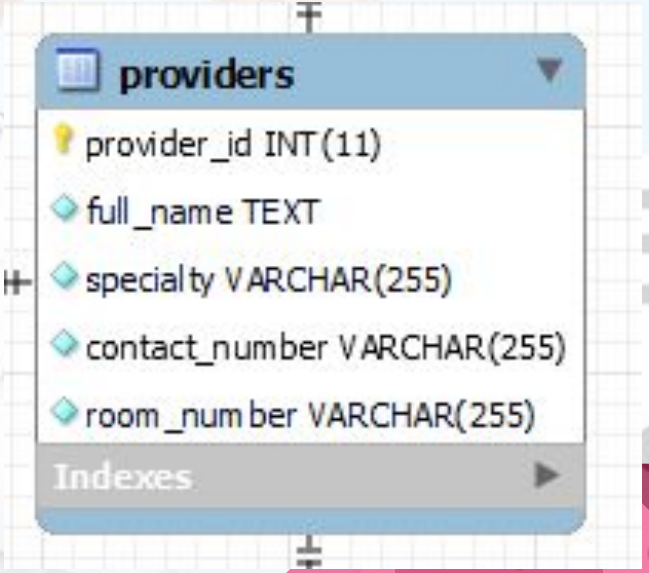




Providers

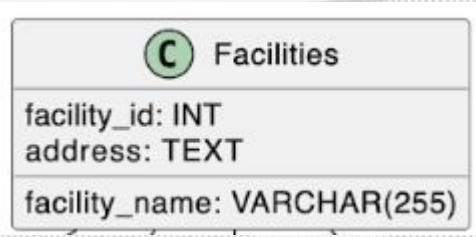
Tracking basic information about providers within the practice

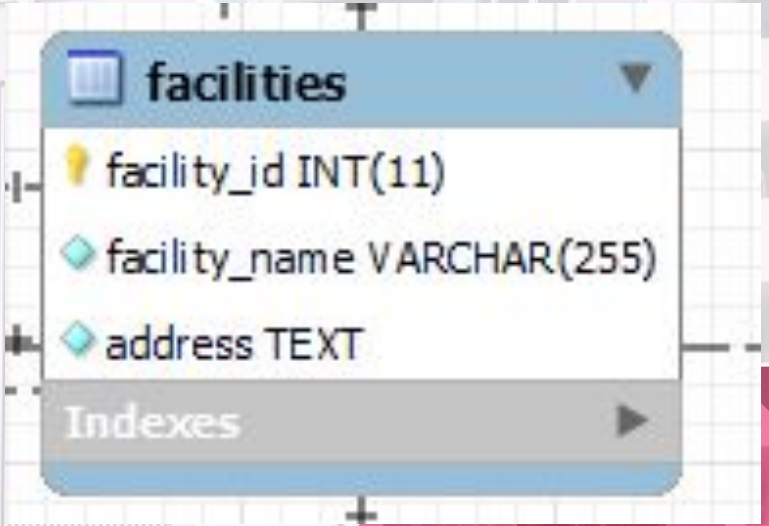




Facilities

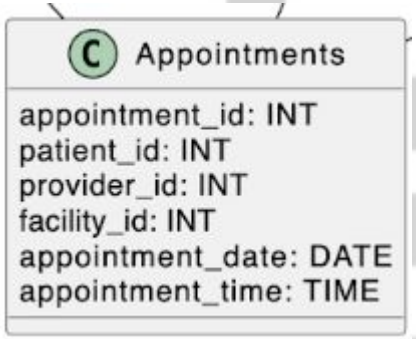
Tracking facilities (i.e. locations) of the practice

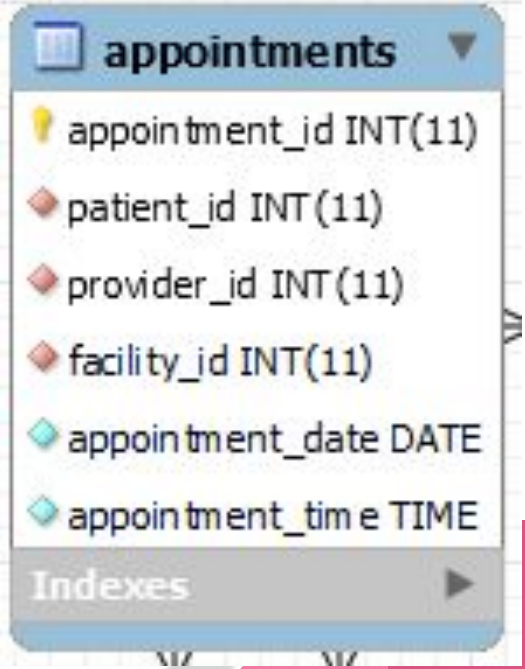




Appointments

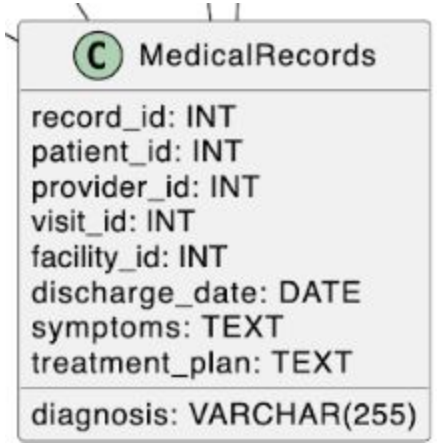
For tracking patients' appointments

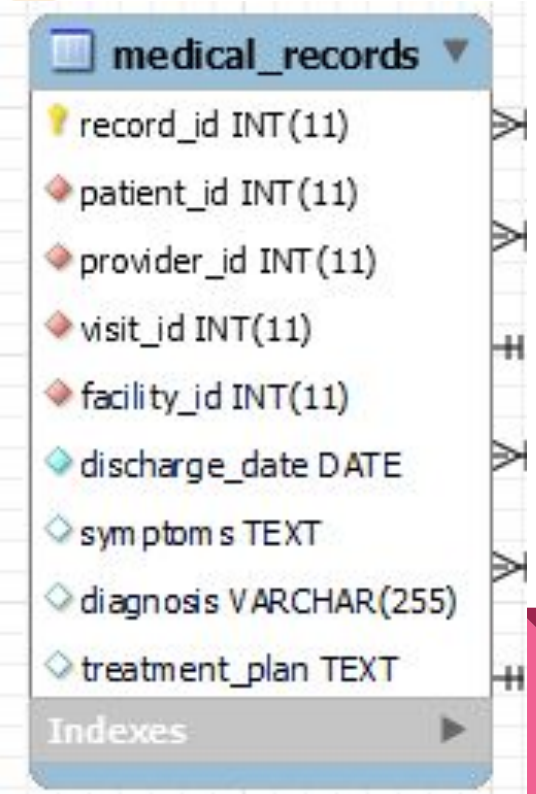




Medical Records

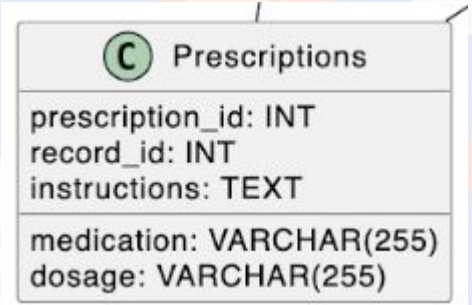
For tracking medical information (linked to visit)

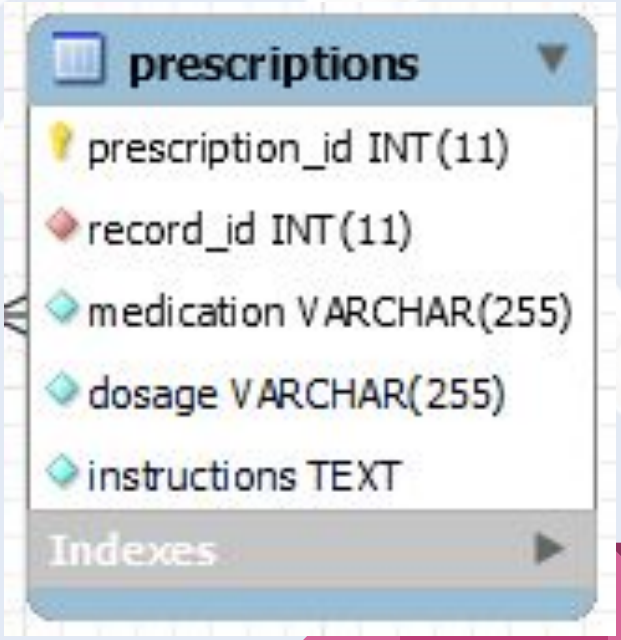




Prescriptions

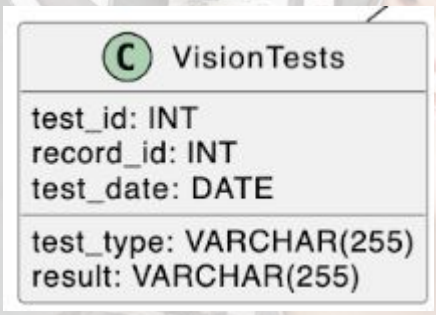
For tracking prescription information

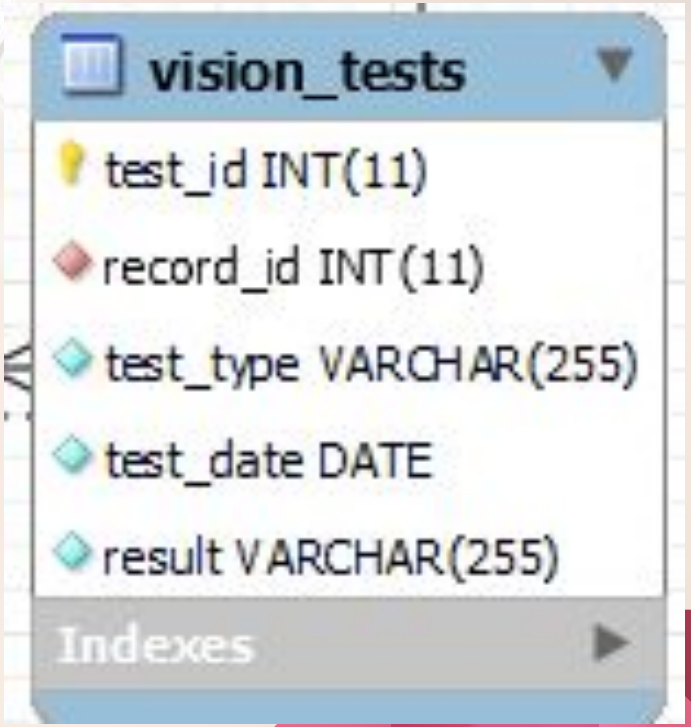




Vision Tests

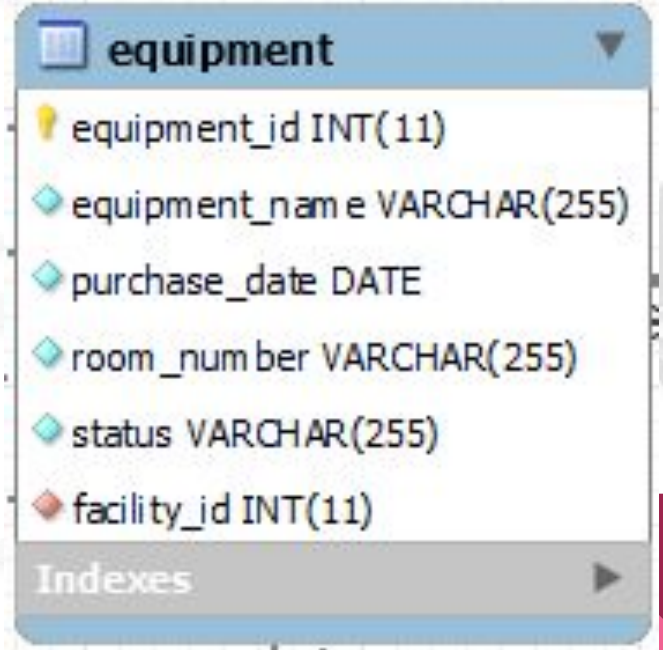
For tracking test results





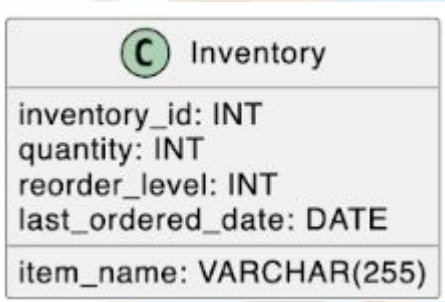
Equipment

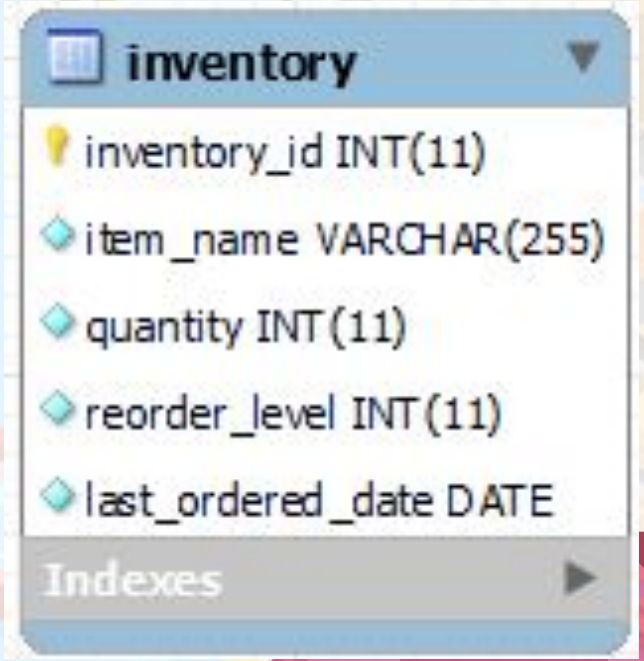
For tracking high-value equipment



Inventory

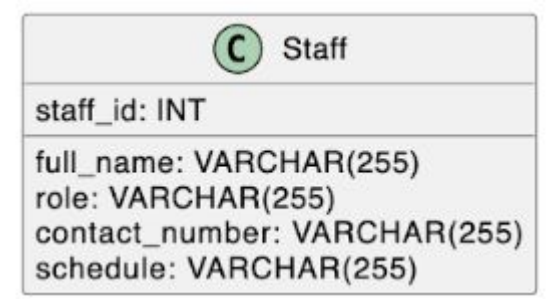
For tracking consumable supplies (non-equipment)

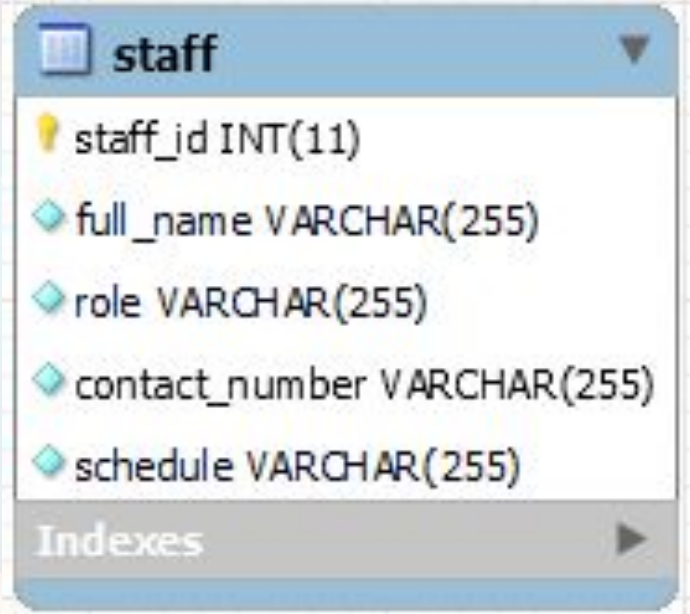




Staff

Basic information for other employees; e.g. nurses





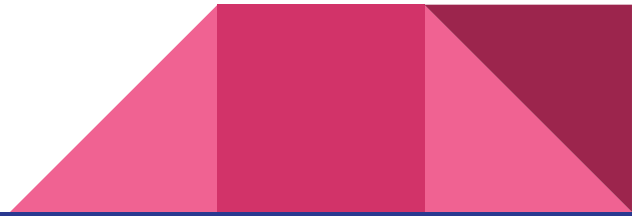
Prove tables are in BCNF form

All tables are in BCNF

Boyce-Codd Normal Form (BCNF) requires that each table has a primary key (or a composite primary key in the case of linked tables), and all attributes are fully functionally dependent on their respective primary key.

There are no partial dependencies or transitive dependencies that would violate the rules of BCNF.

All tables are in BCNF.



- What information can be obtained?

Supporting queries

A sample of the many queries that would be needed to support operations.

The data in these queries are fictitious.

Supporting Query 1

Spec A: Should allow users to enter patient demographic information (including address and insurance)

INSERT INTO patients (full_name, date_of_birth, address, gender, insurance_provider, contact_number)

VALUES ('Alice Johnson', '1990-05-15', '123 Main St, Springfield', 'female', 'HealthPlus Insurances', '555-0201');

Supporting Query 2

Spec B: Should allow users to enter provider information (including specialty)

```
INSERT INTO providers (full_name, specialty, contact_number, room_number)
VALUES ('Dr. Sarah Lee', 'Oncology', '555-0301', '201');
```

Supporting Query 3 and 4

Spec C: Should allow users to enter visit information (including time and facility)

3. Appointments:

INSERT INTO appointments (patient_id, provider_id, facility_id, appointment_date, appointment_time)

VALUES (1, 1, 1, '2024-03-01', '09:00:00');

4. Visits and medical record information:

INSERT INTO medical_records (patient_id, provider_id, visit_id, facility_id, discharge_date, symptoms, diagnosis, treatment_plan)

VALUES (1, 3, 2, 1, '2024-03-02', 'Cough and fever', 'Common Cold', 'Rest');

Note on Spec D:

Spec D: should allow users to enter clinical care information (including recording of signs and symptoms, discharge diagnosis and prescriptions, and orders and results of exams, tests, and procedures)

This spec is satisfied by Query 4.

Supporting Query 5

SPEC E: Other pertinent information depending on scenarios, for example, clinics will need to manage appointments and exam rooms, emergency department will need to manage information about beds. All clinics will also need to manage supplies and billing.

SUPPORTING QUERY 5: Record a prescription

INSERT INTO prescriptions (record_id, medication, dosage, instructions)

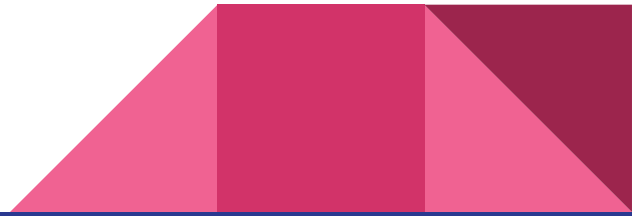
VALUES (1, 'Tylenol', '500mg', 'Take with food once a day');

Supporting Query 6

SUPPORTING QUERY 6: Record results of a vision test

INSERT INTO vision_tests (record_id, test_type, test_date, result)

VALUES (1, 'Visual Acuity', '2024-03-02', '20/20');



Supporting Query 7

SUPPORTING QUERY 7: Record an invoice

INSERT INTO invoices (visit_id, total_amount, customer_portion, paid_by_customer, insurance_portion, paid_by_insurance, date_issued)

VALUES (1, 100.00, 50.00, 50.00, 50.00, 50.00, '2024-03-02');

Supporting Query 8

SUPPORTING QUERY 8: Update quantity of inventory item

```
UPDATE inventory

SET quantity = 10

WHERE inventory_id = 1;
```

Supporting Query 9

SPEC F: Your database should support editing of existing records to correct data entry mistakes or legitimate changes of information (e.g. change of address or insurance).

SUPPORTING QUERY 9: Update patient demographic information

```
UPDATE patients

SET address = '456 Elm St, Springfield'

WHERE patient_id = 2;
```

Supporting Query 10

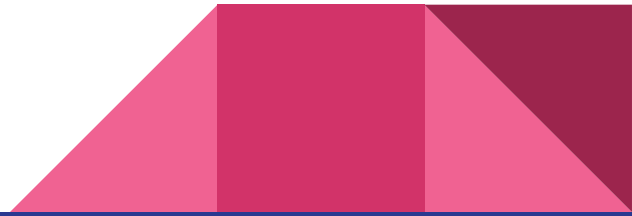
SPEC G: searching of patient records based on name, ID, and possibly other information such as visit dates.

SUPPORTING QUERY 10: Search for patients based on name

```
SELECT *

FROM patients

WHERE full_name = 'John Doe';
```



Supporting Query 11

SUPPORTING QUERY 11: Search for patients based on ID

```
SELECT *

FROM patients

WHERE patient_id = 2;
```

Supporting Query 12

SUPPORTING QUERY 12: search for records based on visit date:

```
SELECT *

FROM patients

JOIN medical_records ON patients.patient_id = medical_records.patient_id

JOIN visits ON medical_records.visit_id = visits.visit_id

WHERE visits.visit_date = '2024-03-25';
```

Supporting Query 13

SPEC H: Your database should support reporting functions such as listing of all patients who satisfy certain selection criteria, such as those who have been given certain diagnosis, or who visited on certain days, or who have been seen by certain doctor, or combinations of these such as, the diagnoses of patients who visited the clinic twice within the shortest time interval.

SUPPORTING QUERY 13: List all patients who have been given a certain diagnosis

```
SELECT *

FROM patients

JOIN medical_records ON patients.patient_id = medical_records.patient_id

WHERE medical_records.diagnosis = 'Myopia';
```

Supporting Query 14

SUPPORTING QUERY 14: List all patients who visited on certain days

```
SELECT *

FROM patients

JOIN medical_records ON patients.patient_id =
medical_records.patient_id

JOIN visits on medical_records.visit_id = visits.visit_id

WHERE visits.visit_date = '2024-03-25';
```

Supporting Query 15

SUPPORTING QUERY 15: List all patients who have been seen by a certain doctor

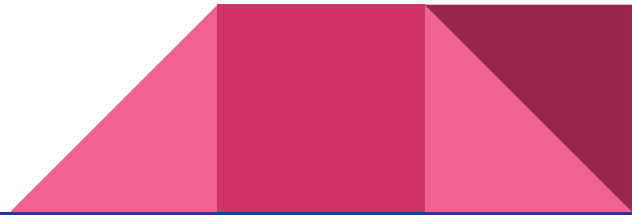
```
SELECT *

FROM patients

JOIN medical_records ON patients.patient_id = medical_records.patient_id

JOIN providers ON medical_records.provider_id = providers.provider_id

WHERE providers.full_name = 'Dr. Iris Clearview';
```



Supporting Query 16

SUPPORTING QUERY 16: List all patients who have visited the clinic twice within the shortest time interval

PLACEHOLDER

- Explain UML data model (can be incorporated with the explaining the database)

UML Models and ER Diagrams included previously

- Explain ER diagram (no need to spend too much time on this part)

Included in previous slides

Show database and test data in SQL (it’s okay to not spend too much on this part)

Supporting functionalities

Included in previous slides