CSE 3241 Project

Part 2

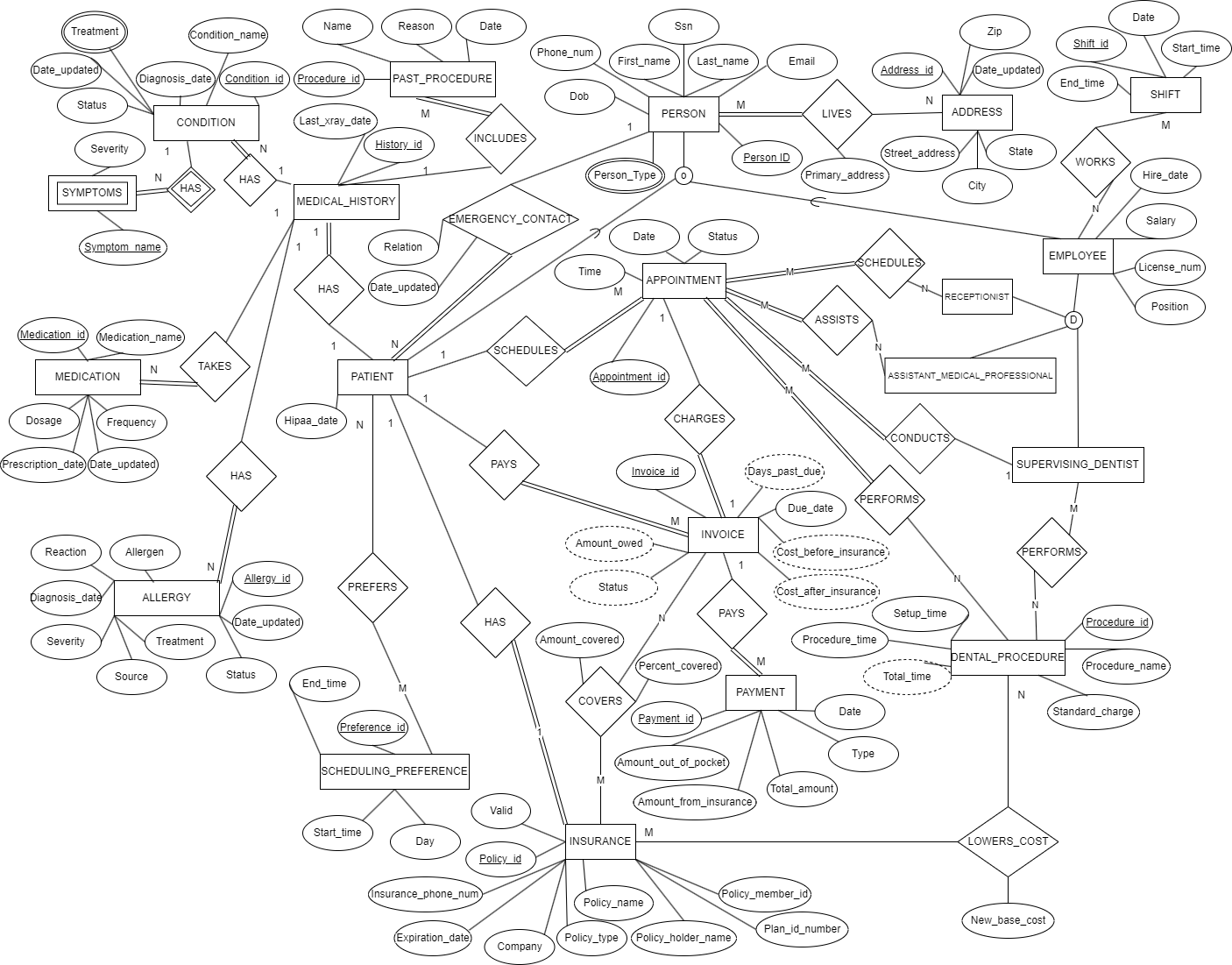
Nico Poggio.1

Casey Curran.232

Connor Stevens.1224

Nathan Johnson.9254

1. EER Diagram:



1. **Relational schema**

**Step 1: (Entities)**

PERSON (PersonID, SSN, First\_Name, Last\_Name, Dob, Phone\_num, Email)

SHIFT(Shift\_id, Start\_time, End\_time, Date)

SCHEDULING\_PREFERENCE(Preference\_id, End\_time, Start\_time, Day)

PAST\_PROCEDURE(Procedure\_id, Name, Reason, Date)

MEDICAL\_HISTORY(History\_id, Last\_xray\_date)

APPOINTMENT(Appointment\_id, Time, Date, Status)

INSURANCE (Policy\_id, Company, Plan\_id\_number, Policy\_holder\_name, Policy\_name, Policy\_member\_id, Policy\_type, Valid, Expiration\_date, Insurance\_phone\_num)

INVOICE (Invoice\_id, Due\_date)

DENTAL\_PROCEDURE(Procedure\_id, Setup\_time, Procedure\_Name, Standard\_charge, Procedure\_time)

PAYMENT(Payment\_id, Total\_amount, Amount\_out\_of\_pocket, Amount\_from\_insurance, Type, Date)

CONDITION(Condition\_id, Condition\_name, Diagnosis\_date, Date\_updated, Status)

ADDRESS(Address\_id, Zip, Date\_updated, Street\_address, State, City)

MEDICATION(Medication\_id, Medication\_name, Dosage, Frequency, Prescription\_date, Date\_updated)

ALLERGY(Allergy\_id, Date\_updated, Status, Treatment, Source, Severity, Diagnosis\_date, Reaction, Allergen)

**Step 2: (Weak entities)**

SYMPTOMS(Condition\_id, Symptom\_name, Severity)

Foreign key Condition\_id references CONDITION

**Step 3: (1 to N)**

APPOINTMENT (AppointmentID, Time, Date, Status, Patient\_id#, Dentist\_id#)

Foreign key *Patient\_id#* references PERSON

Foreign key *Dentist\_id#* references PERSON

INVOICE (InvoiceID, Due\_date, Status, *Patient\_id#)*

Foreign key *Patient\_id#* references PERSON

PAST\_PROCEDURE(Procedure\_id, Name, Reason, Date, History\_id#)

Foreign key *History\_id#* references MEDICAL\_HISTORY

PAYMENT(Payment\_id, Total\_amount, Amount\_out\_of\_pocket, Amount\_from\_insurance, Type, Date, Invoice\_id#)

Foreign key *Invoice\_id#* references INVOICE

CONDITION(Condition\_id, Condition\_name, Diagnosis\_date, Date\_updated, Status, History\_id#)

Foreign key *History\_id#* references MEDICAL\_HISTORY

MEDICATION(Medication\_id, Medication\_name, Dosage, Frequency, Prescription\_date, Date\_updated, History\_id#)

Foreign key *History\_id#* references MEDICAL\_HISTORY

ALLERGY(Allergy\_id, Date\_updated, Status, Treatment, Source, Severity, Diagnosis\_date, Reaction, Allergen, History\_id#)

Foreign key *History\_id#* references MEDICAL\_HISTORY

**Step 4: (1 to 1)**

INVOICE(Invoice\_id, Due\_date, Status, *Appointment\_id#*, *Patient\_id#*)

Foreign key *Appointment\_id#* references APPOINTMENT

Foreign key *Patient\_id#* references PERSON

MEDICAL\_HISTORY (History\_ID, Last\_XRay\_Date, Patient\_id#)

Foreign key Patient\_id# references PERSON

INSURANCE (Policy\_ID, Valid, Insurance\_Phone\_Num, Expiration\_date, Company, Policy\_type, Policy\_holder\_name, Policy\_member\_ID, *PatientID#*)

Foreign key *Patient\_id#* references PERSON

**Step 5: (N to M)**

APPOINTMENT\_RECEPTIONIST(Appointment\_id#, Receptionist\_id#)

Foreign key Appointment\_id# references APPOINTMENT

Foreign key Receiptionist\_id# references PERSON

APPOINTMENT\_ASSISTANT(Appointment\_id#, Assistant\_id#)

Foreign key Appointment\_id# references APPOINTMENT

Foreign key Assitant\_id# references PERSON

APPOINTMENT\_PROCEDURE(Appoint\_id#, Proc\_id#)

Foreign key Appoint\_id# references APPOINTMENT

Foreign key Proc\_id# references DENTAL\_PROCEDURE

PROCEDURE\_DENTIST(Med\_Procedure\_id#, Denist\_id#)

Foreign key Med\_Procedure\_id# references DENTAL\_PROCEDURE

Foreign key Dentist\_id# references PERSON

COVERING\_INSURANCE(Policy\_id#, Invoice\_id#, Amount\_covered, Percent\_covered)

Foreign key Policy\_id# references INSURANCE

Foreign key Invoice\_id# references INVOICE

PATIENT\_PREFERENCE(Preference\_id#, Patient\_id#)

Foreign key Preference\_id# references SCHEDULING\_PREFERENCE

Foreign key Patient\_id# references PERSON

EMPLOYEE\_SHIFT(Shift\_id#, Employee\_id#)

Foreign key Shift\_id# references SHIFT

Foreign key Employee\_id# references PERSON

PERSON\_ADDRESS(Person\_id#, Address\_id#, Primary\_address)

Foreign key Person\_id# references PERSON

Foreign key Address\_id# references ADDRESS

PROCEDURE\_DENTIST(Person\_id#, Procedure\_id#)

Foreign key Person\_id# references PERSON

Foreign key Procedure\_id# references PROCEDURE

INSURANCE\_PROCEDURE(Policy\_id#, Proc\_id#)

Foreign Key Policy\_id# references INSURANCE

Foreign key Proc\_id# references DENTAL\_PROCEDURE`

**Step 6: (Multivalued Attributes)**

TREATMENT(Condition\_id#, Treatment)

Foreign key Condition\_id# references CONDITION

PERSON\_TYPE (Person\_id#, PersonType)

Foreign key Person\_id# references PERSON

**Step 7: (n-ary relationships)**

**No n-ary relationships**

**Step 8: (Specializations)**

PERSON (Person\_id, SSN, First\_Name, Last\_Name, Dob, Phone\_num, Email, Hipaa\_date, Hire\_date, Salary, License\_num, Position, Contact\_id#, Relation, Date\_updated)

Foreign key Contact\_id# references PERSON

Receptionist(Person\_id)

Foreign key Person\_id references PERSON

Assistant\_Medical\_Professional(Person\_id)

Foreign key Person\_id references PERSON

Supervising\_Dentist(Person\_id)

Foreign key Person\_id references PERSON

**Step 9: (Union types)**

No unions

**Final Relational Schema:**

PERSON (Person\_id, SSN, First\_name, Last\_name, Dob, Phone\_num, Email, Hipaa\_date, Hire\_date, Salary, License\_num, Position, Contact\_id#, Relation, Date\_updated)

Foreign key Contact\_id# references PERSON

SHIFT(Shift\_id, Start\_time, End\_time, Date)

SCHEDULING\_PREFERENCE(Preference\_id, End\_time, Start\_time, Day)

PAST\_PROCEDURE(Procedure\_id, Name, Reason, Date, History\_id#)

Foreign key *History\_id#* references MEDICAL\_HISTORY

MEDICAL\_HISTORY (History\_ID, Last\_XRay\_Date, Patient\_id#)

Foreign key Patient\_id# references PERSON

APPOINTMENT (AppointmentID, Time, Date, Status, Patient\_id#, Dentist\_id#)

Foreign key *Patient\_id#* references PERSON

Foreign key *Dentist\_id#* references PERSON

INSURANCE (Policy\_ID, Valid, Insurance\_phone\_num, Expiration\_date, Company, Policy\_type, Policy\_holder\_name, Policy\_member\_id, Plan\_id\_number, *PatientID#*)

Foreign key *Patient\_id#* references PERSON

INVOICE(Invoice\_id, Due\_date, Status, *Appointment\_id#*, *Patient\_id#*)

Foreign key *Appointment\_id#* references APPOINTMENT

Foreign key *Patient\_id#* references PERSON

DENTAL\_PROCEDURE(Procedure\_id, Setup\_time, Procedure\_Name, Standard\_charge, Procedure\_time)

PAYMENT(Payment\_id, Date, Total\_amount, Amount\_out\_of\_pocket, Amount\_from\_insurance, Type, Date, Invoice\_id#)

Foreign key *Invoice\_id#* references INVOICE

CONDITION(Condition\_id, Condition\_name, Diagnosis\_date, Date\_updated, Status, History\_id#)

Foreign key *History\_id#* references MEDICAL\_HISTORY

ADDRESS(Address\_id, Zip, Date\_updated, Street\_address, State, City)

MEDICATION(Medication\_id, Medication\_name, Dosage, Frequency, Prescription\_date, Date\_updated, History\_id#)

Foreign key *History\_id#* references MEDICAL\_HISTORY

ALLERGY(Allergy\_id, Date\_updated, Status, Treatment, Source, Severity, Diagnosis\_date, Reaction, Allergen, History\_id#)

Foreign key *History\_id#* references MEDICAL\_HISTORY

APPOINTMENT\_RECEPTIONIST(Appointment\_id#, Receptionist\_id#)

Foreign key Appointment\_id# references APPOINTMENT

Foreign key Receiptionist\_id# references PERSON

SYMPTOMS(Condition\_id, Symptom\_name, Severity)

Foreign key Condition\_id references CONDITION

APPOINTMENT\_ASSISTANT(Appointment\_id#, Assistant\_id#)

Foreign key Appointment\_id# references APPOINTMENT

Foreign key Assitant\_id# references PERSON

APPOINTMENT\_PROCEDURE(Appoint\_id#, Proc\_id#)

Foreign key Appoint\_id# references APPOINTMENT

Foreign key Proc\_id# references DENTAL\_PROCEDURE

PROCEDURE\_DENTIST(Med\_Procedure\_id#, Denist\_id#)

Foreign key Med\_Procedure\_id# references DENTAL\_PROCEDURE

Foreign key Dentist\_id# references PERSON

COVERING\_INSURANCE(Policy\_id#, Invoice\_id#, Amount\_covered, Percent\_covered)

Foreign key Policy\_id# references INSURANCE

Foreign key Invoice\_id# references INVOICE

PATIENT\_PREFERENCE(Preference\_id#, Patient\_id#)

Foreign key Preference\_id# references SCHEDULING\_PREFERENCE

Foreign key Patient\_id# references PERSON

EMPLOYEE\_SHIFT(Shift\_id#, Employee\_id#)

Foreign key Shift\_id# references SHIFT

Foreign key Employee\_id# references PERSON

PERSON\_ADDRESS(Person\_id#, Address\_id#, Primary\_address)

Foreign key Person\_id# references PERSON

Foreign key Address\_id# references ADDRESS

TREATMENT(Condition\_id#, Treatment)

Foreign key Condition\_id# references CONDITION

PERSON\_TYPE (Person\_id#, PersonType)

Foreign key Person\_id# references PERSON

RECEPTIONIST(Person\_id#)

Foreign key Person\_id references PERSON

ASSISTANT\_MEDICAL\_PROFESSIONAL (Person\_id#)

Foreign key Person\_id references PERSON

SUPERVISING\_DENTIST(Person\_id#)

Foreign key Person\_id references PERSON

3. Relational Algebra

1. Create a list of patients and the medications they currently take

π First\_name, Last\_name, Medication\_name (σPersonType = “Patient”  (PERSON ⨝ Person\_id = Person\_id# PERSON\_TYPE ⨝ Person\_id = Patient\_id# MEDICATION\_HISTORY ⨝ History\_ID = History\_id# MEDICATION))

1. Display patient information for patients who currently have Delta Dental insurance policy.

π SSN, First\_name, Last\_name, Dob, Phone\_num, Email, Hipaa\_date (σCompany = “Delta Dental” (σPersonType = “Patient”  (PERSON ⨝ Person\_id = Person\_id# PERSON\_TYPE ⨝ Person\_id = PatientID# INSURANCE)))

1. Generate a list of procedures and dates of service performed by doctor Smilow.

π ProcedureName, Date (σ Last\_Name = “Smilow”(PERSON ⨝ Person\_id = Person\_id# SUPERVISING\_DENTIST ⨝ Person\_id = Dentist\_ID# PROCEDURE\_DENTIST ⨝ Med\_Procedure\_id = ProcedureID# DENTAL\_PROCEDURE ⨝ ProcedureID = ProcID# APPOINTMENT\_PROCECURE ⨝ Appoint\_id# = AppointmentID# APPOINTMENT))

1. Print out a list of past due invoices with patient contact information. Past due is defined as over 30 days old with a balance over $10.

π DueDate, Status, FirstName, LastName, Phone\_num, Email (σDays\_Past\_Due > 30 AND Amount\_Owed > 10(PERSON ⨝ Person\_id = Person\_id# PERSON\_TYPE ⨝ Person\_id = Person\_id# INVOICE ))

1. Find the patients who brought the most revenue in the past year.

PATIENT\_SUMS ← Person\_idFSUM Amount (σPersonType = “Patient” (PERSON ⨝Person\_id = Person\_id# PERSON\_TYPE ⨝Person\_id# = Patient\_id# INVOICE ⨝ Invoice\_id = Invoice\_id# PAYMENT))

AVERAGE\_SUM ← FAVERAGE Amount(PATIENT\_SUMS)

σSUM\_Amount > AVERAGE\_SUM\_Average\_Amount(PATIENT\_SUMS)

1. Create a list of doctors who performed less than 5 procedures this year.

RESULT ← DentistID F COUNT ProcedureID (PROCEDURE\_DENTIST)

σ COUNT(ProcedureID) < 5 (RESULT)

1. Find the highest paying procedures, procedure price, and the total number of those procedures performed.

MAX\_RESULT ← Procedure\_idFMAX Standard\_charge(DENTAL\_PROCEDURE)

SECOND\_MAX ← Procedure\_idFMAX Standard\_charge(σ ProcedureID < MAX\_RESULT\_Standard\_charge(DENTAL\_PROCEDURE))

RESULT ← Procedure\_name, Standard\_chargeFCOUNT Appointment\_id((σProcedure\_id = MAX\_RESULT\_Procedure\_id(DENTAL\_PROCEDURE)) ∪ (σProcedure\_id = SECOND\_MAX\_Procedure\_id(DENTAL\_PROCEDURE)) ⟕ProcID = Procedure\_id(APPOINTMENT\_PROCEDURE)))

1. Create a list of all payment types accepted, number of times each of them was used, and total amount charged to that type of payment

TypeFCOUNT Payment\_id, SUM Amount(PAYMENT)

1. List ids and names of insurance plans ever used by patients and how many patients have that plan.

Policy\_id, Policy\_nameFCOUNT Person\_id(INSURANCE)

4. Additional queries:

1. Outer join

List of patients where the due date of their invoice is on the same day as their appointment.

PERSON ⟕Person\_id = Patient\_id#(σDue\_Date == Date(INVOICE⨝Appointment\_id# = Appointment\_idAPPOINTMENT))

1. Aggregate function

Procedure with the shortest time.

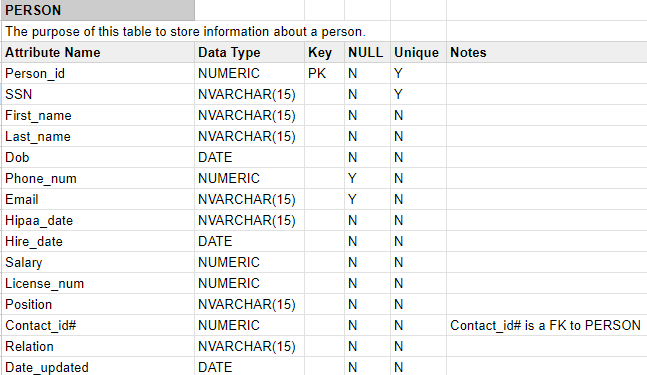
NameFMin Procedure\_Time(DENTAL\_PROCEDURE)

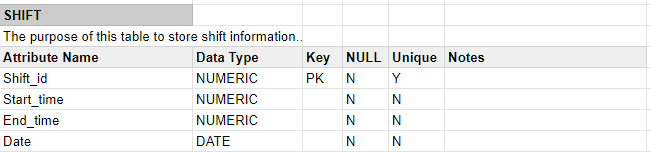
1. Extra entities

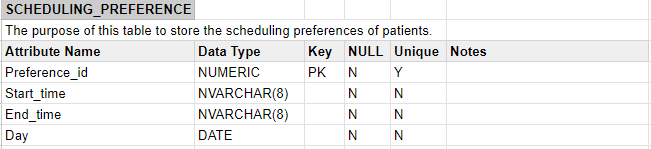
Scheduling preference is inside of an available shift time

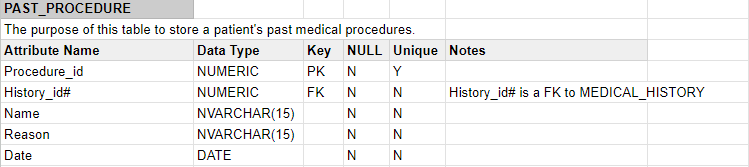
σStart\_Time >= Start\_Time AND End\_Time <= End\_Time(SCHEDULING\_PREFERENCE ⟗ Day = DateSHIFT)

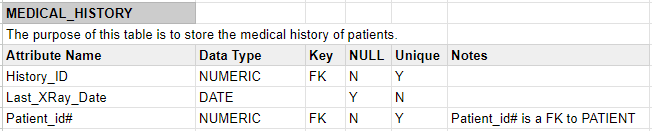
5. Specification Sheets

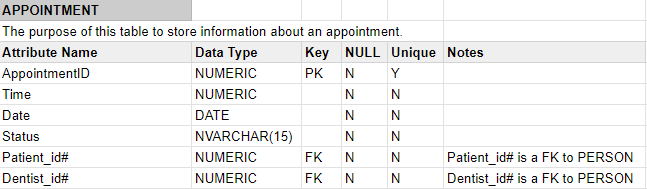


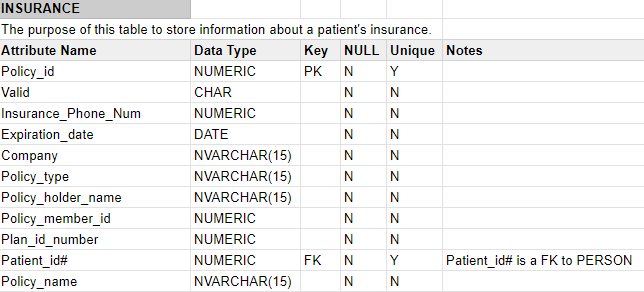


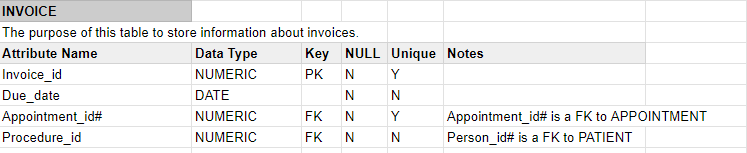


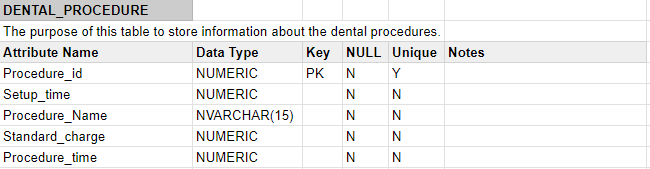


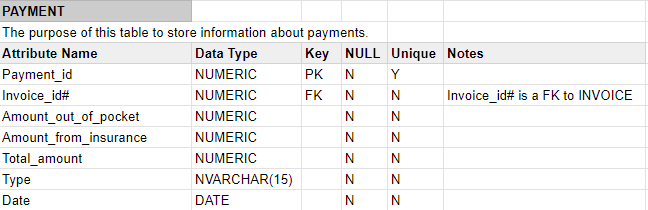


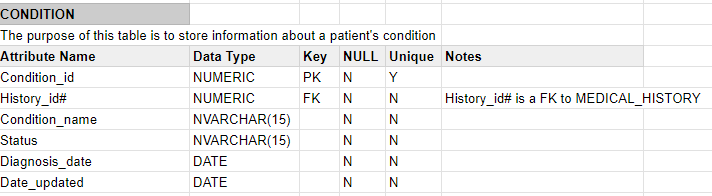


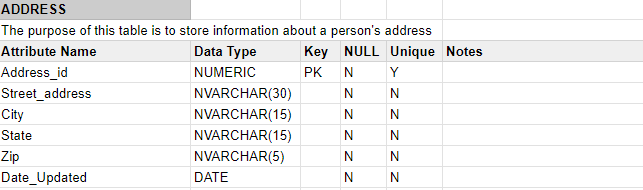


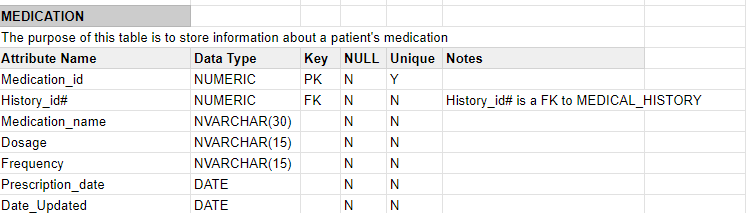


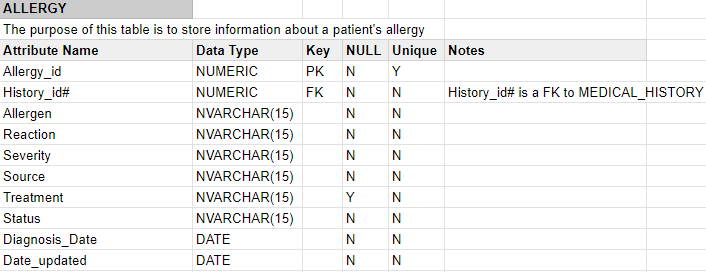


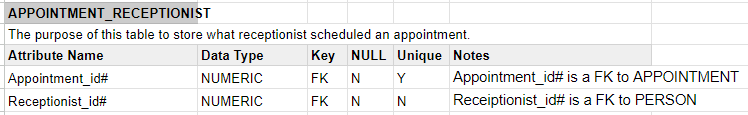


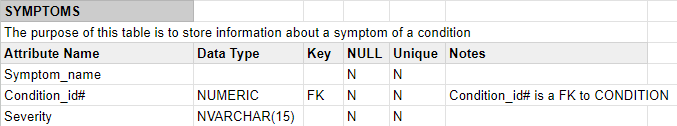


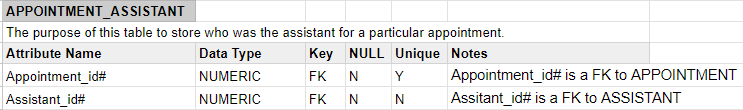


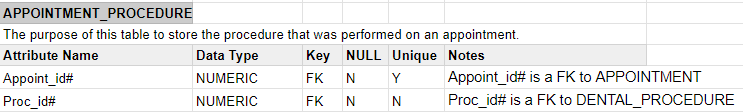


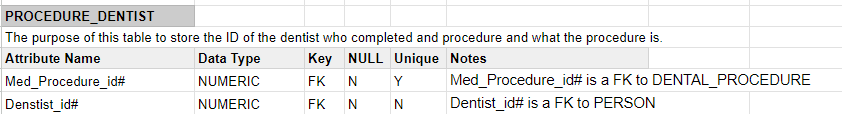


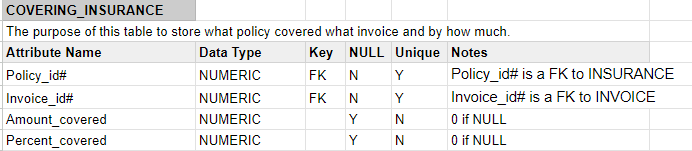


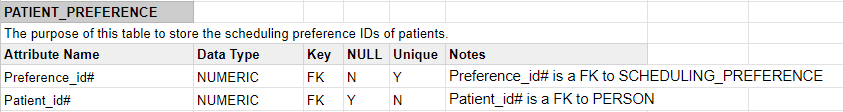


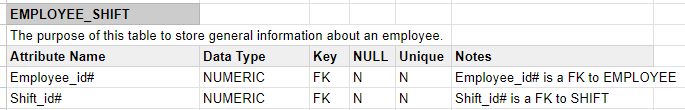


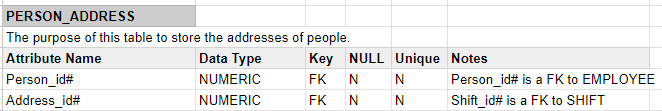


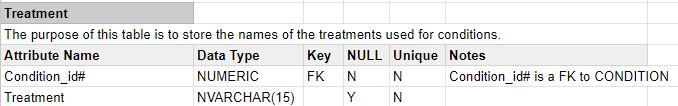


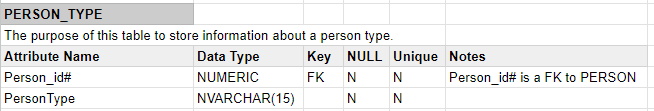


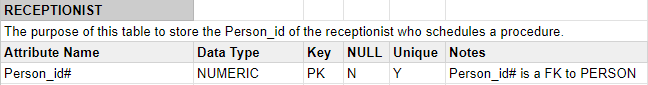


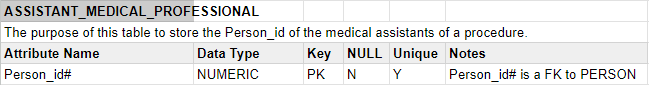


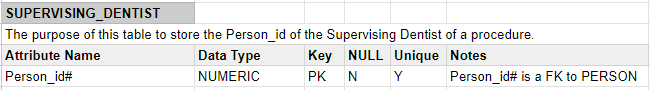




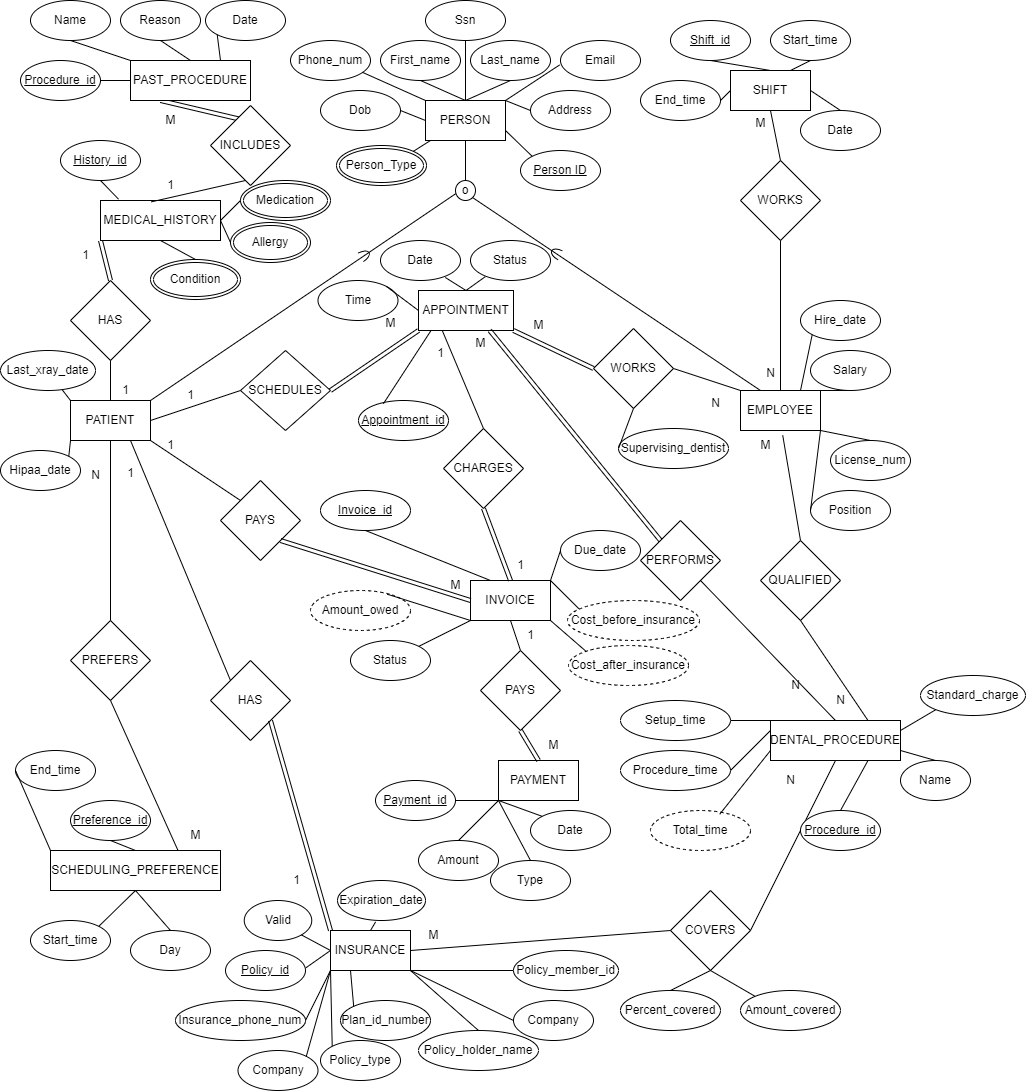








6. Original ERD and comments



Comments we received:

-PATIENT and EMPLOYEE are overlapping. We mistakenly listed it as disjoint under entities and relationships.

-EMPLOYEE entity should have specialization entities.

Added RECEPTIONIST, ASSISTANT\_MEDICAL\_PROFESSIONAL, and SUPERVISING\_DENTIST specializations

-DENTAL\_PROCEDURE could be charged differently based on different policy types

Added the LOWERS\_COST relationship between INSURANCE and DENTAL\_PROCEDURE, so that the base cost for each operation can change according to policy.

-PAYMENT can be split up between INSURANCE and PATIENT

For PAYMENT, replaced the Amount attribute with two attributes: Amount\_out\_of\_pocket and Amount\_from\_insurance

Team Member Contributions:

Nico Poggio.1: Worked on the first draft of the relational schema. Created and adjusted the relational algebra queries.

Casey Curran.232: Worked on building on relational schema, worked on relational algebra queries, worked on additional queries, built out specification sheets, helped with ERD changes

Connor Stevens.1224: Revised the EERD according to the feedback from part 1. Updated the relational schema and specification sheet to reflect these changes. Helped with relational algebra.

Nathan Johnson.9254: Created some of the entities in #2 from steps 1-9. Created first draft of #4 answers. Created ~100 tables in the first version of the specification sheet.

As a team, we were able to effectively communicate using Discord and setting up specific times to meet and do work. As individuals, we would split up and delegate what we were working on. However, we could always text each other with any questions/concerns while we worked through the project if anything came up while we were not on a call.