Connor Shields
CS 340
Project Step 1
4/15/18

Medical Office Database

Project Outline

I will be making a database that represents important information that might be stored in the database of a medical office. In my case, the entities would at least include Patients, Doctors, Appointments, and Insurance Carriers.

Database Outline, in words

The entities in my database are:

1. Patient – patients are the people who go to the medical office for treatment. They have a relation with every other entity. Patients have the following attributes:
    a. id: automatically generated id number, which is the primary key and can't be null
    b. fname: First name of the patient, a string with max 255 chars. Can't be null.
    c. lname: Last name of the patient, a string with max 255 chars. Can't be null.
    d. Primary Provider: Id of the provider to which the patient belongs. Can't be null, and must be a provider that exists in the database.
    e. Insurance Carrier: IDNum of the patient's insurance carrier. Defaults to null
    f. Treatments completed: Number of incomplete appointments the patient is attached to. Defaults to 0, but cannot be null

2. Providers – The doctors who work in the medical clinic. They have a relationship with the patient, as well as the treatments they can perform.
    a. id: autogenerated id, which is the primary key and can't be null
    b. fname: first name of the doctor, a string with max 255 chars. Can't be null.
    c. lname: Last name of the doctor, a string with max 255 chars. Can't be null.
    d. patients: number of patients who are linked to this provider.
    e. specialty: specialty of the doctor. Enumerated data type that cannot be null. Might be important if we decide to use possible treatments for each provider later.
    f. room number: The number of the room that this provider operates in. Cannot be null. Specific to this provider. Must exist in the database.

3. Appointments – the appointments that are performed by a provider for a patient.
    a. idNum: the distinct id number of the appointment.
    b. provider: id of the provider associated with this appointment. Can't be null. Must already exist in the database.

c. patient: id of the patient associated with this appointment. Can't be null. Must already exist in the database.

d. insuranceId: id of the insurance carrier who will pay for part or all of this appointment. Can be null. Defaults to null.

e. cost: total cost of this appointment.

f. Insurance portion: portion that insurance will pay. Cannot be null. Defaults to 0.

g. Patient portion: portion that patient will pay. Cannot be null. Defaults to 0.

h. Status: enumerated data type: defaults to "scheduled". Can also be "complete" or "broken". Cannot be null.

i. Room Number: The number of the room in which this appointment will take place. Cannot be null. Must exist in the database.

4. Insurance Carrier – the carrier that can be associated with a patient.

a. subscribers: number of subscribers to this plan. Cannot be null. Defaults to 0

b. coverage: percent that this plan will cover for every patient appointment. Cannot be null. Defaults to 0 (this is an oversimplification, but works for this purpose. Might add more complexity to this later).

c. name: name of the insurance company. Cannot be null. No default set.

d. Carrier ID: ID that the insurance provides to be used to associate it. Cannot be null. This is the primary key.

5. Room: The room that a provider uses to perform their operations. And integer. Cannot be null.

a. Room Number: an integer. Cannot be null. Does not have a default value.

b. Provider Number: the ID number of the provider associated with this room. Can be null. Can only be occupied by one provider. Must exist in the database.

The Relationships in my database are:

1. A patient has a primary provider. A patient can only have one *primary* provider, but a provider can have many patients. Therefore, the patient and provider entities are a one-to-many relationship

2. Appointments have a patient. An appointment can only have one patient associated with it, but a patient can have many appointments that they're scheduled for. This is a one-to-many relationship between the patient and appointment entities.

3. Patients have insurance: A patient can have many insurance carriers (primary, secondary, tertiary, etc), and and insurance carrier can have many patients associated with it. Therefore, this is a many-to-many relationship.

4. Provider to Room: A provider can only have one room, and a room can only be associated with one provider at a time. Therefore, this is a one-to-one relationship.