CS 3380 Lab Assignment 2

Directions : Use the following Entity–relationship model to create a set of tables within your database that conform to the information conveyed by the model. You must submit a single `lab2.sql` text file that contains SQL CREATE TABLE statements based on the model shown below with appropriate primary and foreign key declarations.

Initially, this instruction might sound somewhat odd, but you'll be thankful that it's here later. The very first thing that you should write in your `lab2.sql` file is the following statement.

```
DROP SCHEMA IF EXISTS lab2 CASCADE;
```

This statement will cause the lab2 schema to be dropped and all tables within that schema to be dropped as well. This statement will be useful because as you develop the SQL statements required for this lab assignment, you will want to execute your script multiple times. The effect of this `DROP SCHEMA` statement let's you start with a "clean slate" each time you import your SQL script.

Your SQL file should then create a schema named `lab2` into which all tables are created. (Hint: After creating the schema, if you modify your `search_path` variable, to list `lab2` first, then newly created tables will be added to that schema instead of the public schema. Alternatively, you can create tables using schema-qualified names such as `lab2.doctor`).

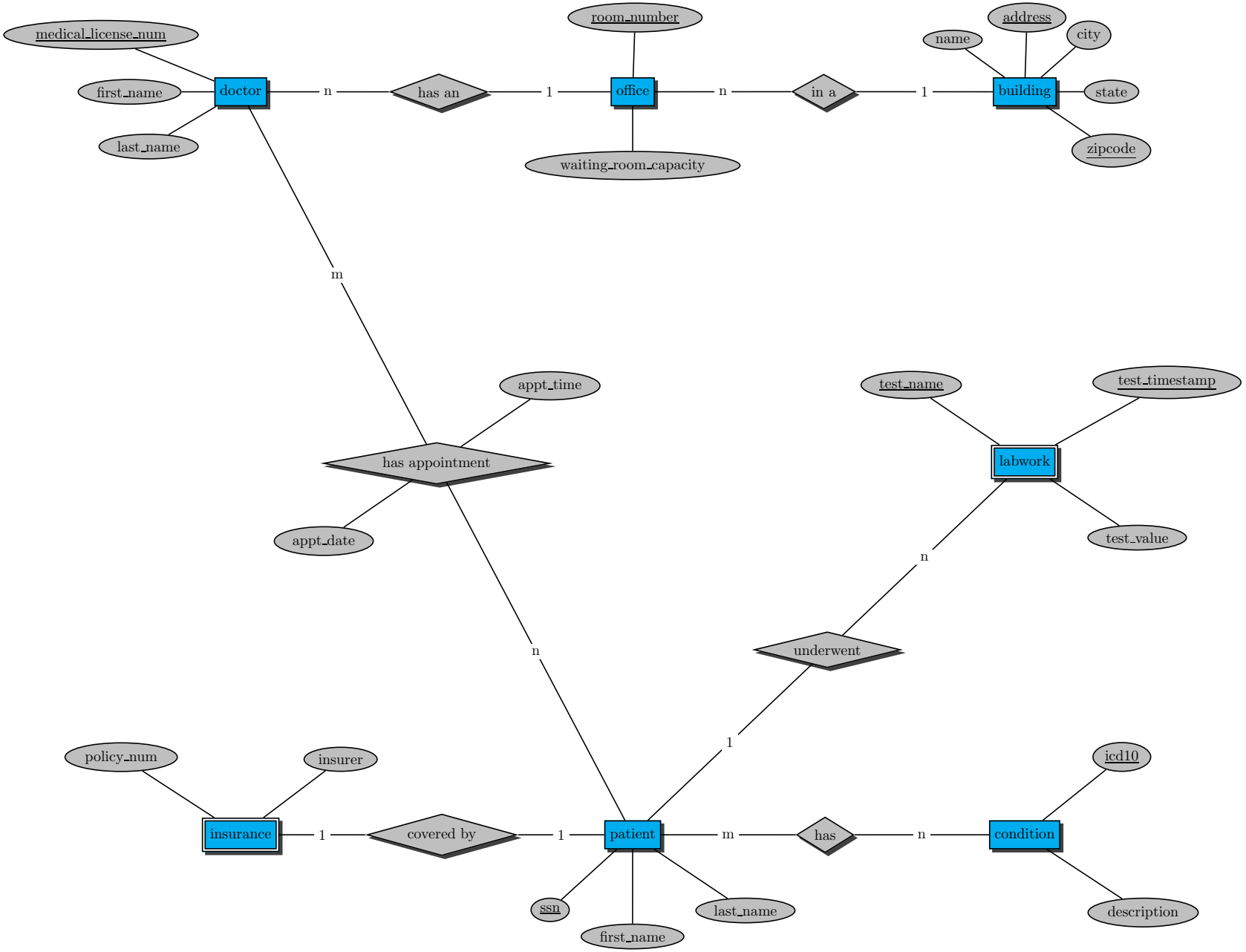
Next, add your `CREATE TABLE` statements for all of the required tables based on the ERD that follows.

Finally, add at least 3 records to every table using `INSERT` statements. You are free to insert whatever data you'd like into the tables. Make reasonable assumptions for appropriate data types for each attribute. (I believe the only non-obvious field attribute shown is "icd10". This is an alphanumeric code for various medical conditions. Do a quick Google search to determine what an appropriate data type might be for this field.) Don't forget to add any appropriate foreign key constraints.

This assignment must be submitted via Blackboard by **Sunday, February 8 at 11:59 PM**.

Extra Credit: Imagine that you wanted to delete a patient record from your database. However, that patient has insurance, condition and labwork information associated with it via foreign key constraints. How could you modify your table definitions to allow the deletion of a patient record to also remove any associated insurance, condition and labwork information. (Read section section 5.3.5 in the PostgreSQL documentation; it discusses the concept of foreign keys in depth.)

After you figure out what needs to be done modify your `CREATE TABLE` statements appropriately. If you successfully implement this you can earn 10 points of extra credit.


medical_license_num
first_name
last_name
doctor
n
has an
1
room_number
office
waiting_room_capacity
n
in a
1
address
name
city
building
state
zipcode
m
has appointment
appt_time
appt_date
n
test_name
test_timestamp
labwork
test_value
n
underwent
1
policy_num
insurer
insurance
1
covered by
1
patient
ssn
first_name
last_name
m
has
n
icd10
condition
description