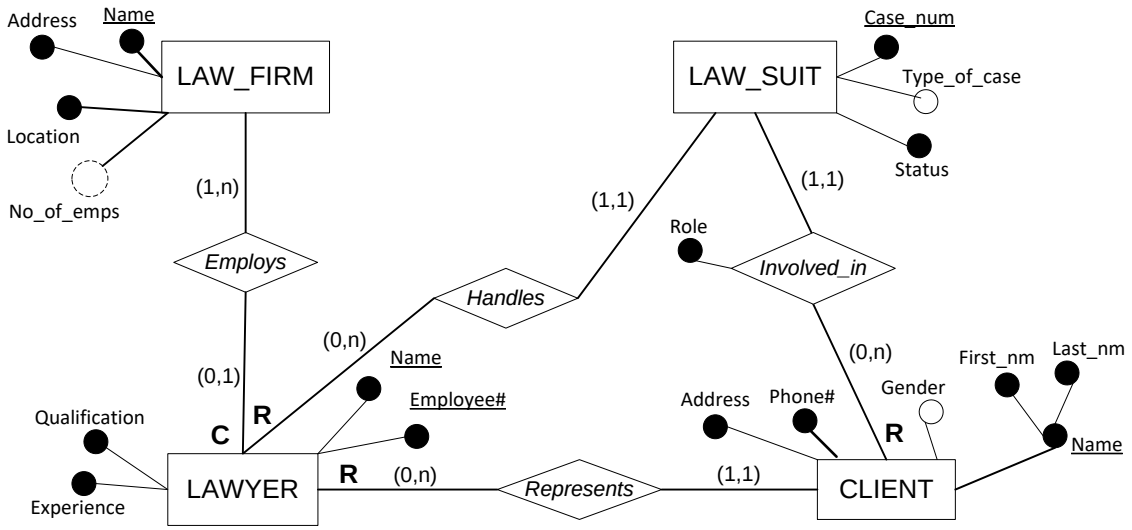
MIS 4386
A Create Tables Opportunity

The purpose of this opportunity is for you to create four tables. The tables themselves are based on the conceptual data model shown below that involves law firms, lawyers, clients, and law suits. Clients use lawyers to represent them in law suits in which they participate. Law firms employ lawyers; however, some lawyers simply engage in private practice. Law firms (which can only be located in the states of Ohio, Texas, or California) are identified by their respective names and contain three other attributes: address, location, and number of employees. Clients have a name, gender, address, and phone number and can be identified by their respective names. Likewise, lawyers are perhaps identified by their respective names too even though the ones working for a law firm may also have a unique employee number of some sort. Some of the other attributes of a lawyer are qualification and experience. Law suits each have a unique case number assigned by the court along with the attributes type of case and current status. Other simple as well as complex business rules pertaining to the scenario will often not be present in the requirements specification; the conceptual modeling grammar (here, ER modeling grammar) may not have constructs to capture some of these simple/complex business rules.

The conceptual data model for this scenario expressed using the ER modeling grammar is displayed in the following figure.



Gender can only be male or female is a simple business rule that cannot be captured in an ER diagram. Allowable values for gender are the letters 'M', 'm', 'F', or 'f'. The role of a client in a law suit can be either that of plaintiff (represented by the letter 'P') or defendant (represented by the letter 'D'). "A lawyer representing a client should handle the case the client is involved in" is an example for a complex business rule that an ER diagram cannot capture. One other thing. Please define Name as the primary key of LAWYER.

Blackboard contains the file **createlawtables.sql**. The purpose of this script is to allow you to concentrate on adding the clauses necessary to create the constraints. In other words, I have already started each create table statement by assigning column names and data types to the various columns associated with each table. **In order for you to properly define the necessary constraints, you will**

need to add some additional column names to some of the create tables statements. Essentially what you will be doing is modifying my create table statements so that they include the relevant primary key, not null, check and referential integrity constraints. **You may also find that you need to reorder the create table statements in this file in order for all four tables to be successfully created**. In other words, at the moment the create table statements are in the following order: the create table statement for the client table comes first followed by the create table statement for the lawyer table, the create table statement for the law_firm table, and the create table statement for the law_suit table.

Moving forward, I have decided that it might work best if each member of the class belonged to a three-person group. I have taken it upon myself to create the groups since not everyone in the class may feel comfortable asking others (or even knowing many others) in the class to work with him or her. Besides often in the real world people find themselves thrown together on a project with people they barely know and in some cases with whom they have never met. My purpose in creating the groups is for each member of the class to have, in addition to me and Hanup, at least two others in the class with whom he or she can communicate. Sending files back and forth within a given group, if necessary, is OK. Sending files back and forth between groups **is not necessary**. **The composition of the groups is available on Blackboard in the Database Creation section.**

Of course once we have created the tables, we need to test them to make sure that our table and column definitions allow for the entry of only valid data. For example, it should not be possible to insert two law firms with the same name or two law suits with the same case number. In order to give the tables you have created a thorough test, I have created some test data in the form of insert statements and delete statements. This test data is available on Blackboard and stored in the file **lawfirmcreatedata.sql**. If you have defined your constraints properly, some of the insert statements will successfully insert a row into a table. On the other hand, some of the insert statements should fail because the data they contain violate one of the constraints associated with the table. Likewise, some of the delete and update statements will be successful while others should fail.

Note: You need not make any changes to the lawfirmcreatedata.sql file. This file was "seeded" with errors in order to test whether your create table statements handle all of the necessary constraints. You will need to review the structure of the data in the insert statements in order to determine where to place the foreign key attributes.

Deliverables

1. Please store your modifications to my create table statements in the **createlawtables.sql** file. When all of your create table statements work to your satisfaction (a) add a spool command as the first line in this file and (b) execute this file by entering in SQL*Plus the command:

   start [path to the location of your file] createlawtables.sql

   The spooled output will make it possible for me to review your modified create table statements.

2. Get into SQL*Plus and add a spool command as the first line of the **lawfirmcreatedata.sql file**. Then enter the command

   start [path to the location of your file] lawfirmcreatedata.sql

   The spooled output will make it possible for me to see if you have properly defined all of the tables. When you submit the printed spooled output please highlight the INSERT statements and DELETE statements that generate the error messages.

   Suggestion
   Whenever you make changes to createlawtables.sql, remember to execute the file before executing lawfirmcreatedata.sql again.

3. Write a query that displays the following information about each constraint you have created. Your column headings should appear exactly as follows:

```
CONSTRAINT_NAME CONSTRAINT_TYPE TABLE_NAME      COLUMN_NAME   POSITION   LAST_CHANGE
--------------- --------------- --------------- ------------ ---------- ------------
```

   You will need to use the two data dictionary tables found in the handout material to handle this query.

Creating a New User Prior To Beginning the Create Tables Opportunity

Begin by logging on to SQL*Plus as usual using the username and password combination that you established when installing Oracle 19c

The create user command creates a new user with the name cougar and password cougar. Please choose a username and password that is convenient with you. I suggest making the username and password the same. It is easier to remember.

```
SQL> create user cougar identified by cougar;

User created.
```

The grant command makes it possible for cougar to log on to the database and also create various type of Oracle objects such as tables and constraints.

```
SQL> grant connect, resource to cougar;

Grant succeeded.
```

This connect command allows cougar to log on to the database.

```
SQL> connect cougar/cougar
Connected.
```

The following start command allows to you create the initial version of the four tables for the Create Tables Opportunity. Of course createlawtables.sql will be located in a folder of your choosing.

```
SQL> start e:\mis4386fall22\databasecreation\createlawtables
SQL> drop table lawyer cascade constraints purge;
drop table lawyer cascade constraints purge
           *
ERROR at line 1:
ORA-00942: table or view does not exist


SQL> drop table law_firm cascade constraints purge;
drop table law_firm cascade constraints purge
           *
ERROR at line 1:
ORA-00942: table or view does not exist


SQL> drop table law_suit cascade constraints purge;
drop table law_suit cascade constraints purge
           *
ERROR at line 1:
ORA-00942: table or view does not exist


SQL> drop table client cascade constraints purge;
drop table client cascade constraints purge
           *
ERROR at line 1:
ORA-00942: table or view does not exist
```

```
SQL>
```

Please complete the Create Tables Opportunity using the new username and password combination that you have created.

```
SQL>
CREATE TABLE client
(Cl_fname             varchar2(15),
Cl_lname              varchar2(23),
Cl_gender             varchar2(1),
Cl_ph#                   number(10),
Cl_address            varchar2(30)
);

CREATE TABLE lawyer
(Lyr_name             varchar2(30),
Lyr_emp#             varchar2(11),
Lyr_qual               varchar2(37),
Lyr_exp                 smallint
);

CREATE TABLE law_firm
(Lf_name              varchar2(45),
Lf_address            varchar2(30),
Lf_location    varchar2(30)
);

CREATE TABLE law_suit
(Ls_case_num       varchar2(17),
Ls_case_type       varchar2(11),
Ls_status              varchar2(7)
);
```