# CS430 Homework 2

Brendan Nguyen

Due: Wednesday, Mar. 1, 2023

Question 1 (30 points)

A movie platform database contains information about *authors* (identified by `authorid`) and information about *books* (identified by `bookid`). Each book also has a name, a genre, a release year, and a publisher (i.e. publishing company). Authors also have a name, age, phone, and address. Authors write books.
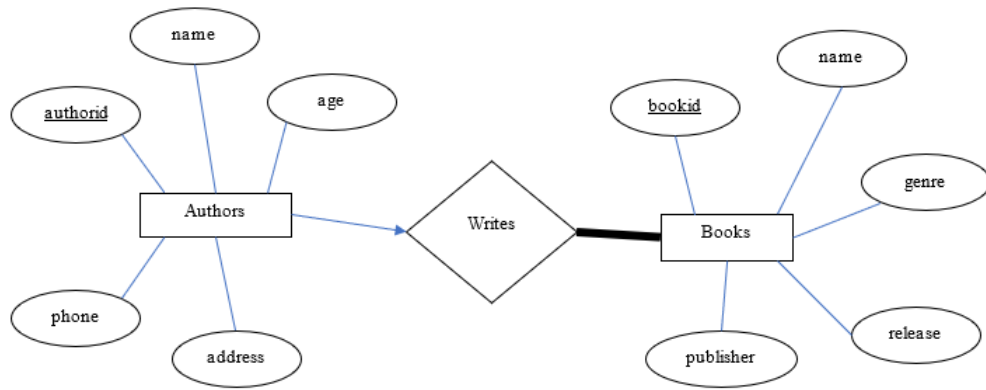
a) Draw the ER diagram of this database (as described in the Question 1 statement). Do not use any other constraints.
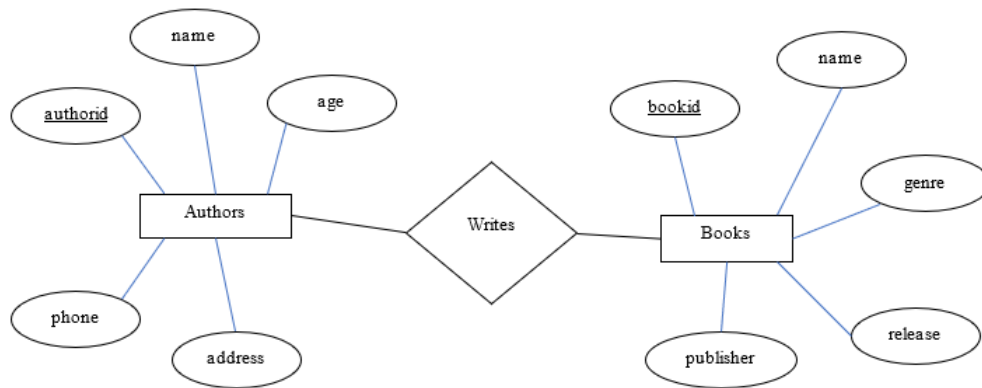
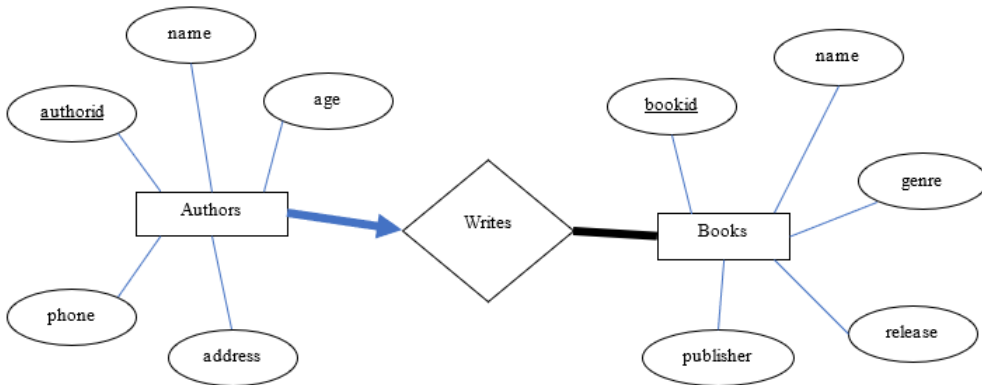

b) Modify the diagram from a) further to add the constraint that each book must be written by at least one author.
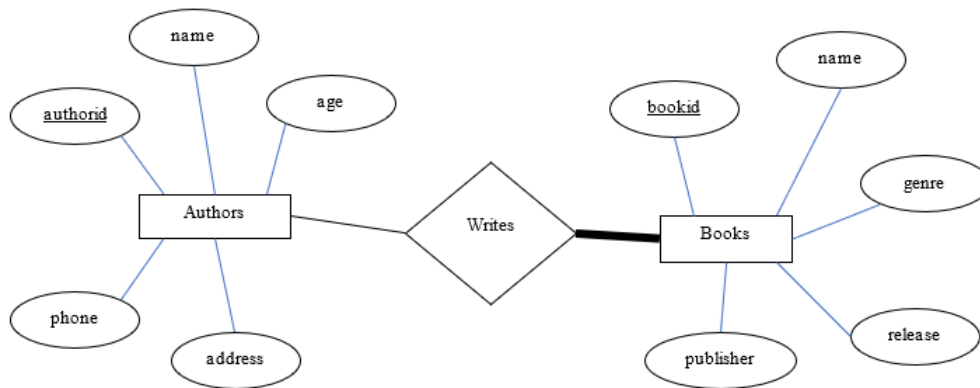
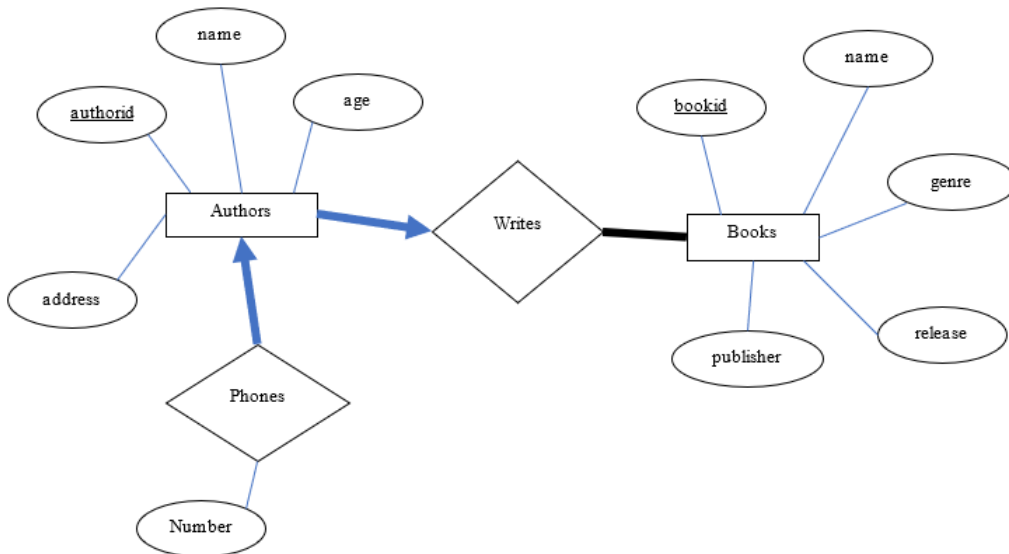c) Modify the diagram from b) further to add the constraint that each author must write at most one book.
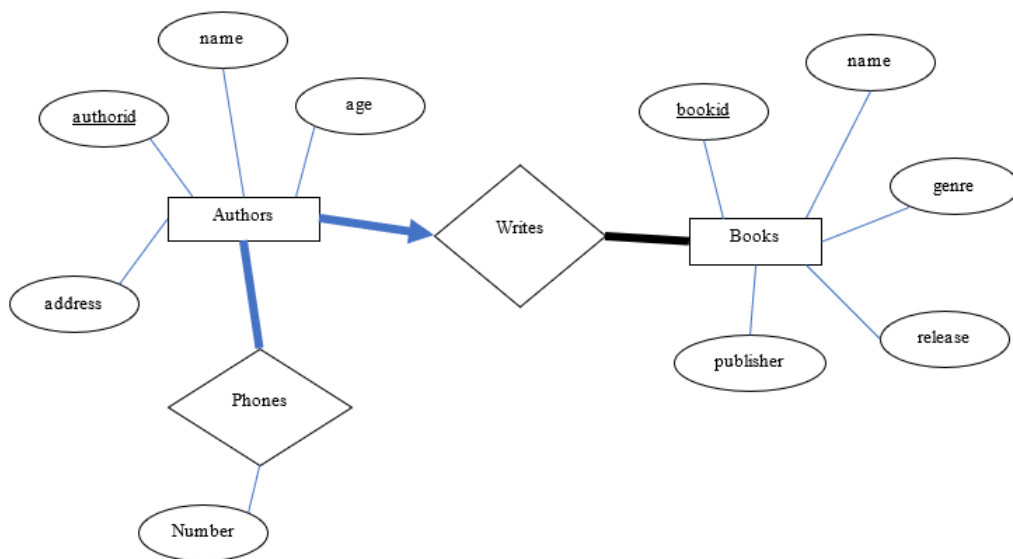
d) Modify the diagram from c) further to add the constraint such that each book must be written by exactly one author.

e) Modify the diagram from d) further such that each author can have multiple phone numbers.

f) Modify the diagram from e) further such that each author must have at least one phone number.

Question 2 (30 points)

Given a database that stores information about *Persons* (identified by `id`) and *Countries* (identified by `countryname`). A country also has a capital, a president, and a continent. A person also has a first name, a last name, date of birth (dob). Persons visit Countries.

NOTES: the ER diagram should strictly follow the notations used in class. No other notations will receive any points.

a) Draw the ER diagram that describes this database. Do not add any additional constraints.
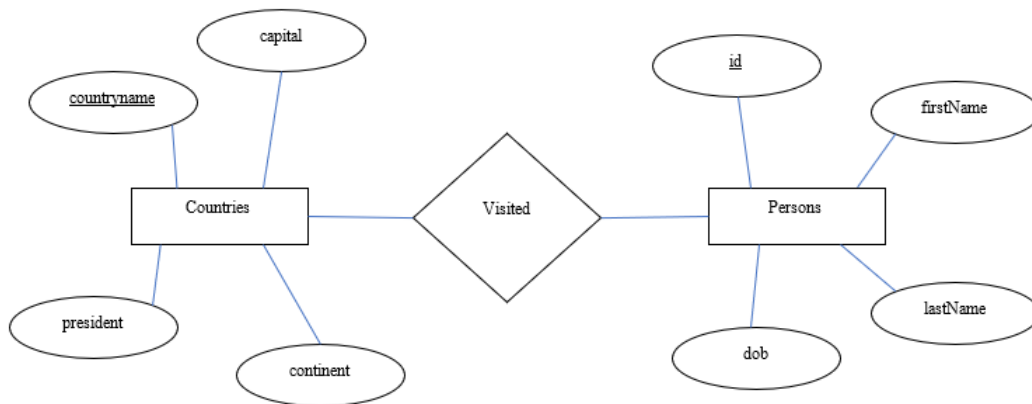
b) Write the database schema for this ER diagram.

- *Countries* (countryname: string, capital: string, president: string, continent: string)

- *Persons* (id: int, firstName: string, lastName: string, dob: date)

- *Visited* (id: int, countryname: string)

c) Write the SQL `CREATE TABLE` statements for all tables identified for this database.
   The create statements have to work when ran against the Oracle database.
   The create statements have to be written in an order such that if executed in that order will not cause any error.

```sql
CREATE TABLE Countries (
    countryname     VARCHAR(20) PRIMARY KEY,
    capital         VARCHAR(20),
    president       VARCHAR(20),
    continent       VARCHAR(20)
);

CREATE TABLE Persons (
    id          NUMBER(9) PRIMARY KEY,
    firstName   VARCHAR(20),
    lastName    VARCHAR(20),
    dob         DATE
);

CREATE TABLE Visited (
    id          NUMBER(9),
    countryname VARCHAR(20),
    PRIMARY KEY(id, countryname),
    FOREIGN KEY(id) REFERENCES Persons,
    FOREIGN KEY(countryname) REFERENCES Countries
);
```

d) Write the SQL statement to update the table *Persons* to add two new columns: city and state.

```sql
ALTER TABLE Persons ADD (
    city    VARCHAR(50),
    state   VARCHAR(50)
);
```

e) Write the SQL statements to remove these three tables from the database. Write the statements in an order such that if executed in that order no error will occur.

```sql
DROP TABLE Visited;
DROP TABLE Persons;
DROP TABLE Countries;
```

Question 3 (40 points)

(provide your answers to this in an SQL file)

Given the following schema:

- *songs* (songid: int, title: string, release: date)

- *singers* (singerid: int, name: string, age: real, city: string, state: string)

- *singsin* (singerid: int, songid: int)

The primary keys are uld in each relation. Relation *singers* contains information about singers. Relation *songs* contain information about songs. Relation *singsin* contains information about singers singing songs.

a) Write the SQL statements that create tables *songs*, *singers* and *singsin*. Don't forget to define the key constraints.

b) Write the SQL query that extracts all the names of the singers from Boston. Sort the result by name in ascending order.

c) Write the SQL query that extracts information about the singers and the songs they sing. Each record in the result should contain the name and age of the singer, the title of the song, and the release date of the song. Sort the result by the name of the singers in descending order.

d) Write the SQL query that finds how many singers from Los Angeles, CA are in the database.

e) Write the SQL that extracts information about singers whose name ends with the letter "a". Sort the result by the state of the singers, in descending order.

f) Write the SQL that extracts the name and state for all singers who played in a song that has a title that contains the word happy. The query should be case insensitive with regard to the case of the letters from the title of the song.

g) Write an SQL query to extract the name, city, and state of singers that sang a song that was released after Aug 10, 2022.

h) Write the SQL to extract the names of the oldest songs.

i) Write the SQL to extract the number of unique singers that sang some songs between Jan 1st 2022 and May 31st 2022 (including these dates).

j) Write the SQL that extracts the number of songs released in the year 2020.