1. (25 pts)

The following is the requirement descriptions for a Coronavirus Testing Database for a state.

Create the ER diagram for this database.

- The database will keep track of the residents, testing centers, zip codes, cities and counties in the state.
- For each resident, we will keep track of a unique resident identifier, a unique SSN, his/her name, street address, multiple phones, birthdate, and age. The name is composed of first name and last name. Each resident may have a various number of phones. The age is computed based on the birthdate and the current date.
- For each testing center, we will keep track of a unique testing center identifier, name, phone, and street address.
- For each zip code, we keep track of a unique zip code identifier (a simple number), and the unique zip code. For example, if there are only 60 zip codes stored in the database, then the zip code identifier goes from 1 to 60. The actual zip code could be like 19122, 19123, etc.
- For each city, we keep track of a unique city identifier, the city name, and the population in the city.
- For each county, we keep track of a unique county identifier, the unique county name, and the population in the county.
- Each resident in the database must get tested in at least one testing center and could get tested in multiple testing centers. Each testing center in the database must offer testing to at least one resident, and could offer testing to many residents. For each testing that a resident gets in a testing center, we will record the date when the testing occurs, and the testing result.
- **Choose one between the following two requirements**
    - **Each resident in the database can ONLY get tested in a specific testing center once.**
        - **NOTE: this requirement does not involve multiple instances between same entities in a M:N relationship**
    - **(BONUS: 5 pts)** Each resident in the database could get tested in a specific testing center multiple times on different dates. But on the same day, no resident gets tested in the same testing center more than once.
        - **HINTS: this is a situation of M:N relationship with multiple instances between the same entities.**

        - **Read these additional slides for what this means:**
            - **Ch2-MultipleInstancesBetweenSameEntities-ForFinalProject.pdf**
            - **Download Ch2-MultipleInstancesBetweenSameEntities-ForFinalProject.pdf**
            -
        - **If you don't understand these slides, you are welcome to set up individual meeting with me for further clarification.**
- Each resident in the database resides in one and only one zip code. For each zip code in the database, there can be many residents living there, or no resident at all.
- Each testing center in the database is located in exactly one zip code. For each zip code in the database, there can be at most one testing center, or no testing center at all.
- Each city in the database must have at least one zip code, and could have more than one zip code. Each zip code belongs to exactly one city.
- Each city in the database is incorporated into exactly one county. Each county in the database must include one city, and could include many cities.

The following are the requiremets about the ER Diagram

    - **Must show the cardinalities of all relationships:**
        - **Maximum cardinality**
        - **Minimum cardinality (Participation)**
        - **Exact cardinality if any**
    - **MUST be CONSISTENT WITH THE NOTATION used in Jukic's textbook. Otherwise, you don't get credits.**
    - **MUST follow the NAMING CONVENTION discussed in**
        - Slides 40, 41 in **Wk8-Lec15-Jukic-Ch2-ERD-P2.pdf**
        - Download Wk8-Lec15-Jukic-Ch2-ERD-P2.pdf
        -

2. **(25 pts)**

Map your ER diagram in Q1 into a relational schema.

The following are the requiremets about the Relational Schema

- **MUST be CONSISTENT WITH your ERD Diagram.**
- **MUST be CONSISTENT WITH THE NOTATION used in Jukic's textbook. If not, you don't get credits.**
- **MUST use the following NAMING CONVENTION**
    - **Use the same NAMING CONVENTION discussed in**
        - Slides 40, 41 in **Wk8-Lec15-Jukic-Ch2-ERD-P2.pdf**
        - Download Wk8-Lec15-Jukic-Ch2-ERD-P2.pdf
        -
    - **With the OPTIONAL EXCEPTION that**
        - **You can rename the relationship using the corresponding noun.**
    - **OPTIONAL means:**
        - **You can also name the relation generated from a relationship same as the relationship (a verb).**

3. **(25 pts)**

Create all the tables in your relational schema in Q2.

The following are the requirements about your SQL Code:

- **MUST be CONSISTENT WITH your Relational Schema.**
- **Use appropriate data types for columns**
- **Define Necessary Constraints**
    - **You must define at least the primary key and foreign key constraints.**
        - **Each table must have a primary key.**
        - **Each child table must have the necessary foreign key constraints.**
    - **You may add more constraints if necessary (optional).**
- **CREATE TABLE list shows foreign key constraint requirement**

        - **It means that in the list of your CREATE TABLE statements, parent table creation is shown BEFORE its child table creation.**
        - **It's recommended that you run those statements in SQL Developer to make sure that the order is correct.**
    - **NO Handwritten commands**
        - **All code must be entered and edited in a computer file.**
    - **CREATE TABLE statements must work in Oracle. You can test them in your CIS Oracle Database account.**

4. **(25 pts)**

Observe the following table AUTHERBOOKMIX with the sample data.

This table captures the data about authors, books, and publishers.

In addition, it shows the per-book royalty amount of an individual author.

Each author has a unique author id, last name, first name, and an institution to which he/she is affiliated.

Each book has a unique book id, **a title, and a price.**

Each publisher has a unique name and a city where it is headquartered. Each publisher has exactly one headquarter.

Each author can write at least one book and could write multiple books.

Each book is written by at least one author and could be written by multiple authors.

Each book is published by exactly one publisher.

Each publisher can publish at least one book, and could publish many books.

The following are the requirements about your work in b), c), d) in this question:

- **Mark the primary key in each table**
- **Name each table with a meaningful name.**
- **List the functional dependencies and mark their types (partial, full, transitive) in each table**
- **The column names in your tables must be the same as the original table in the question.**

a)

What are the types of the following functional dependencies (full, partial, and transitive) ?

AuthorID ---> AuthorLastName, AuthorFirstName, AuthorInstitution
BookID ---> BookTitle, BookPrice, BookPublisher, PublisherCity
BookPublisher ---> PublisherCity
AuthorID, BookID ---> AuthorBookRoyalty

b)

Is the table in 1NF? If not, normalize it to 1NF.

c)

Show the result of normalizing the table to 2NF.

d)

Show the result of normalizing the table to 3NF.