1. Draw three different figures from the following problem: conceptual model, logical

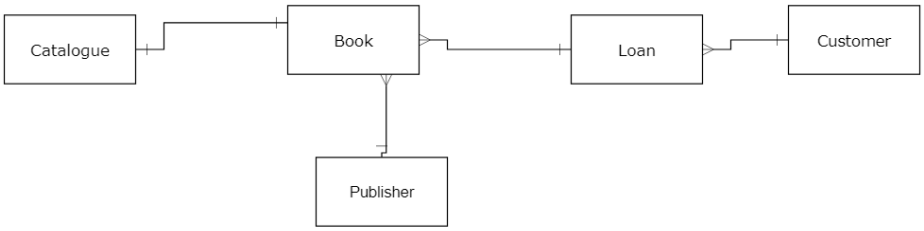
model and Physical Model.

Problem

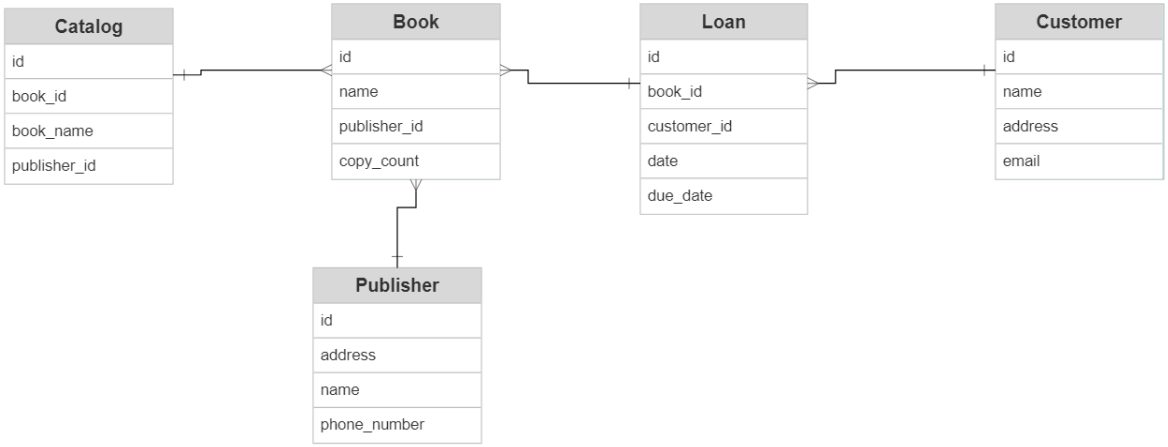
The library owns (physical) books that are stored on shelves and checked out by customers. Each of these books is represented by a catalog entry.

There is only one title card for each book in the catalog, but there can be many physical copies of that book on the shelves. Each book has a publisher and each publisher has some other information like phone number and address.

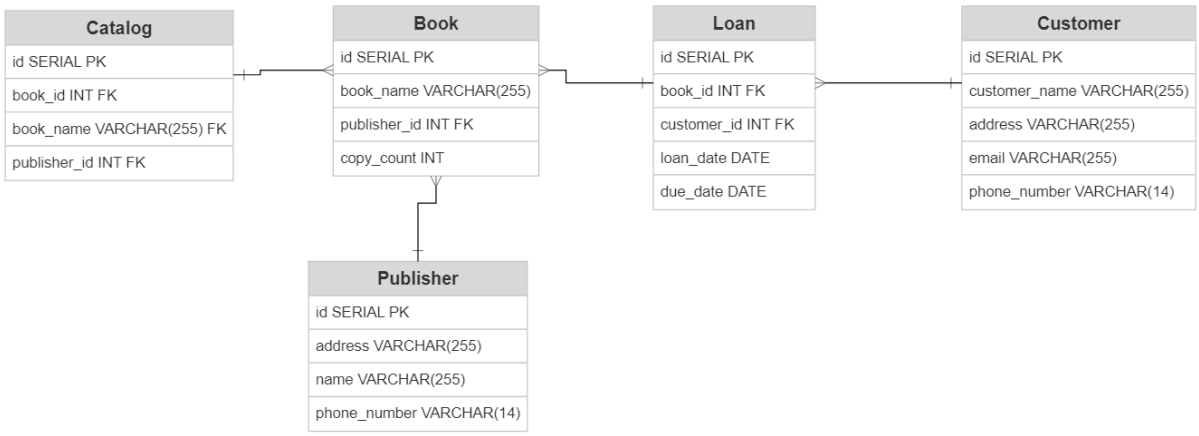
Conceptual model:



Logical model:



Physical model:

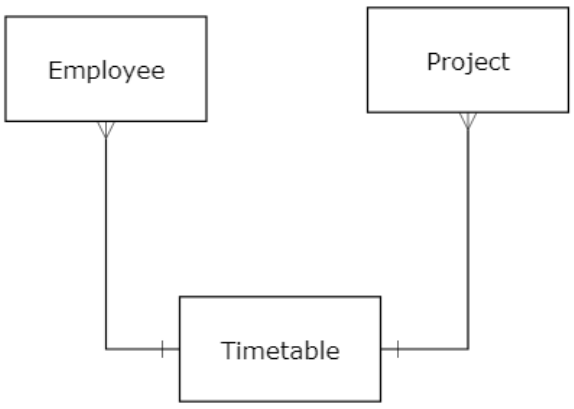


2. Draw three different figures from the following problem: conceptual model, logical model and Physical Model.

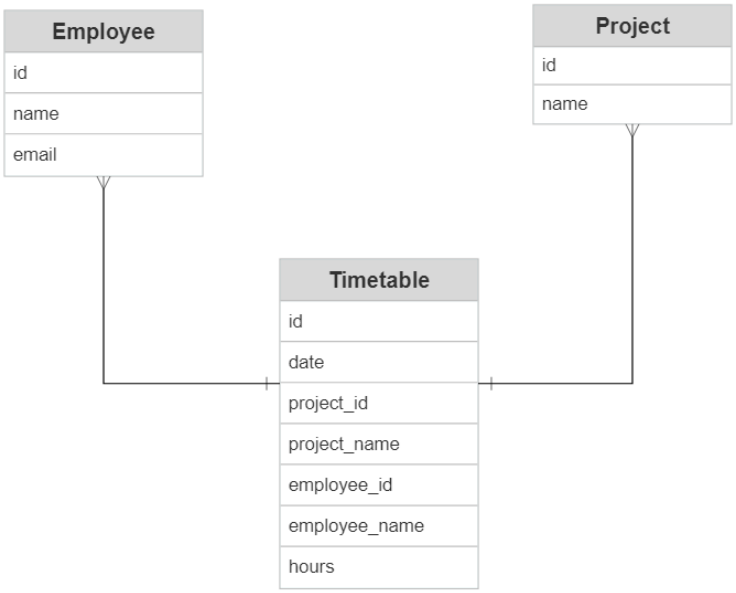
Background information: You work for a startup IT company that performs web development for outside companies. In any given day, you may be working on more than one project. Projects are typically group projects so others may be working on the same project on the same day that you are.

Problem: To keep track of what projects you and your co-workers are working on every day, each employee must create entries into the time system every night before they leave for that day. That time entry will list one project and the number of hours the employee worked on that project that day. There will be entry every day for each project that each person worked on that day.

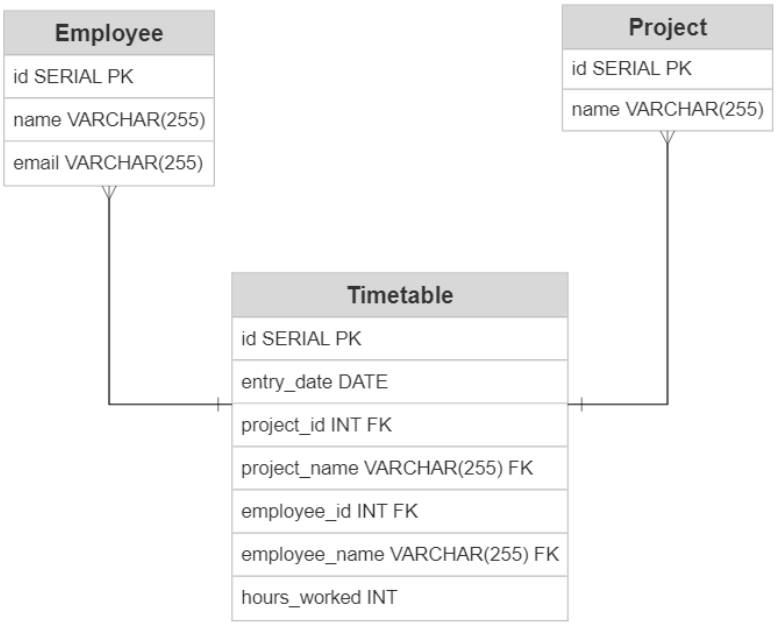
Conceptual model:



Logical model:



Physical model:



Problem

A database is to be made to store information about a catalogue of CDs. Information to be

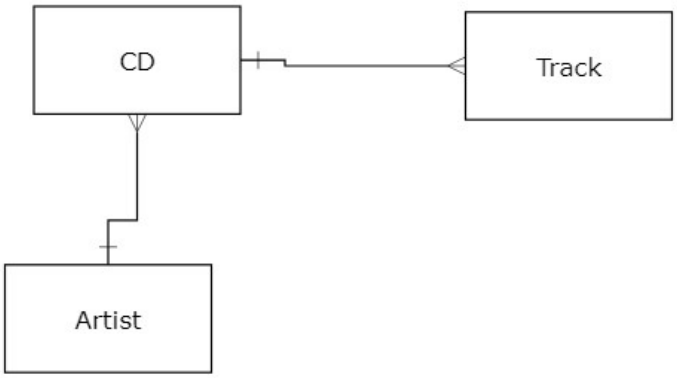
stored about each CD includes their title, price, genre, and a list of tracks. Each CD will also

have an artist, and each artist may produce several CDs. Tracks will have a title and a running

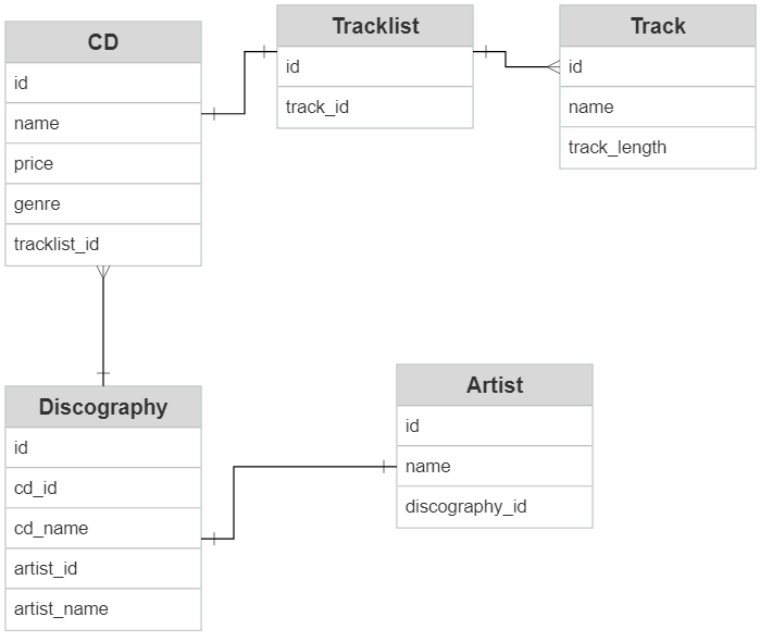
time (in seconds). Artists have names associated with them and it should be possible to

search the database by artist names.

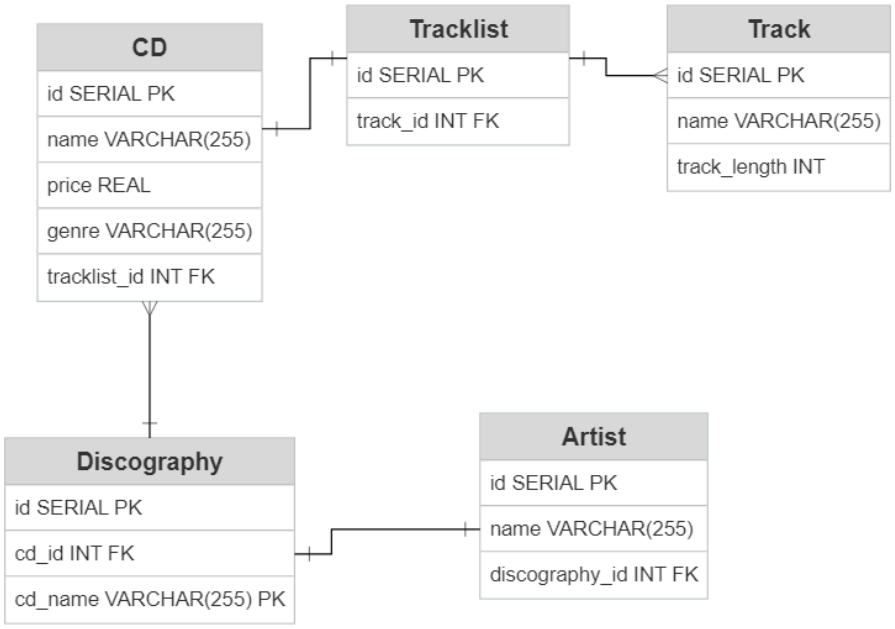
Conceptual model:



Logical model:



Physical model:



4. Make the python scripts to make it easier to establish the database structure for the

previous exercise:

a. Implement a python script create_database_tables.py in which you must create the

database and table structure according to your physical model

b. Create another script populate_tables.py in which you Insert some test data (at least 3

rows) to each table.

See each folder for the implementations. To run the code, make sure to configure the database_config_template.py, give it the proper port, host address and user credentials. Run create_database_tables.py, as it fetches test data from the populate_tables.py file, comment out the execute_commands(blaa blaa, testdata) line from the bottom of the file if you don’t wish for test data to be inserted.