CSCI4333-Project

Notown Records has decided to store information about musicians who perform on its albums (as well as other company data) in a database. The company has wisely chosen to hire you as a database designer. Here is the background information provided by the company:

1. Each musician that records at Notown has a name, address (number, street name, and street type), and an SSN.
2. Each instrument purchased at Notown has an id, a name (suppose only three names: guitar, synthesizer, flute), and a musical key (suppose only six types: C, B, E, C-flat, B-flat, E-flat).
3. Each album recorded on the Notown label has a title, an album identifier, a copyright date, and a format (CD or MC).
4. Each musician may participate in several albums and a given album may be designed by several musicians.
5. Each instrument can be used in any number of albums, and an album may use several instruments.

Originally, the data is stored in a huge CSV file. Luckily, somebody in the company helps you split the data into several tables(data_clean.zip). These are your tasks:
Task 1: (5 pt) Build the database via Python and SQLite. Enforcing all the key/participation constraints.
Task 2: (5 pt) Import the data from the data_clean.zip into the database.
Task 3: (10 pt) The company asks you to implement two Python files for the following functions:

- **summary.py**: Return the summary report of the company. The report includes:
    1. A total number of musicians and a list of musicians.
    2. A total number of albums and a list of albums recorded at Notown.
    3. A total number of instruments and a list of instruments at Notown.
    4. A table consists of the name of musicians and the total number of albums written by them.

- **update.py**: The company could use this Python file to insert or delete records in the database. If the operation is illegal, your Python code should print an error message. The file should be run in the following format:

  $python \quad update.py \quad --[add][delete] \quad --table \quad [table\_name] \quad --record \quad \text{“}[record]''$

  Examples of commands are:

  $python \quad update.py \quad --add \quad --table \quad album \quad --record \quad \text{“}Z, 26, 1990, CD''$

  Explanation: add the record “Z,26,1990,CD” into table “album”

  $python \quad update.py \quad --delete \quad --table \quad album \quad --record \quad \text{“}D, 5, 1991, CD''$

  Explanation: delete the record “D,5,1991,CD” from table “album”