1. **Discuss your solution's time complexity. What tradeoffs did you make?**

   R: It is linear (O(n)) since it will depend on the number of lines and the number of relation records on N:1 (for example number of votes per legislator) that are sent in the input files. I believe that it is enough to insert them into a simple DB as SQLite just to organize the data in a better way. Besides, it makes it possible to append data if multiple input files are needed, just a way to nest it. I also believe that with more time it would be possible to include new ways of doing it, such as:

   a. Using an endpoint where you could zip the CSVs and upload them with the endpoint;
   b. Use a storage service link and get it from there, like AWS S3 or Google Storage;
   c. Get the data by searching string names instead of sending the IDs, so that the user can easily pass the data on the CSVs;
   d. The columns could be adjusted based on the ones on the CSV (if a new column is added on the CSV input, it could be added on the SQLite table as well).

2. **How would you change your solution to account for future columns that might be requested, such as "Bill Voted On Date" or "Co-Sponsors"?**

   R: Adding new columns on the migration should solve it. Even new columns to normalize the data. So, for the first case, a new column would need to be added to the DB as a migration, added to the array that checks the columns on the import and then to the report at the end. The same for the Co-Sponsors, but also a new table would be needed to check out the sponsors of a bill being N:1, and the sponsor column should be out of the bill table. The new table would have the bill id, the sponsor id and a flag on 1 and 2 where 1 is primary sponsor and 2 is co-sponsor. Also, depending on how much it scales, maybe a more robust relational DB could handle it better.

3. **How would you change your solution if instead of receiving CSVs of data, you were given a list of legislators or bills that you should generate a CSV for?**

   R: In this case, it would be multiple input forms; it also could be structurally imported to the SQLite DB so that both input forms could work since the code accumulates the data provided from both. Doing it that way would make so that everything is standard after the import.

4. **How long did you spend working on the assignment?**

   Around 2:45 hours.