Project Documentation

Github: [thezack1212/431-Database-Management (github.com)]

Getting Started

To begin executing the python script, you must first open the file using a python integrated development environment (IDE) running python 3.10 or later. Once open, simply run the program in the IDE.

Running the Script

Once the script has been started, you will see a list of several different query options presented to you numbered 1 though 11 that will look like *Figure 1*. For numbers 1 through 9, select the query you would like to run by typing the matching number of the query you would like, and outputs will be printed to the terminal below as seen in *Figure 2*, each tuple represents a record in the table.

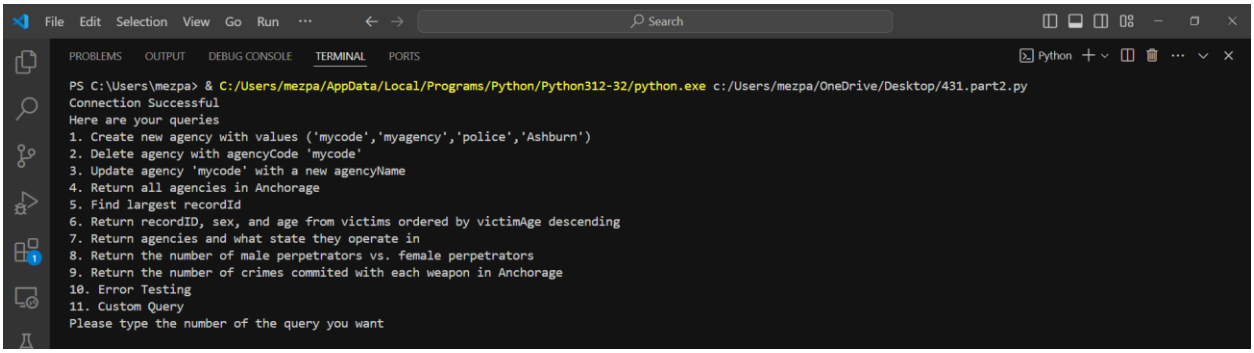

Figure 1: User being prompted for a query.

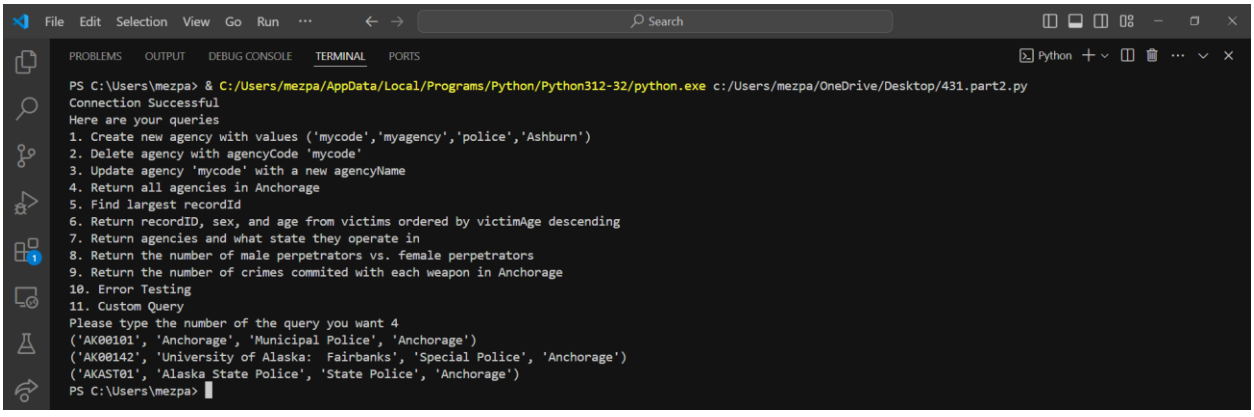

Figure 2: Executing query number 4.

Query 10 and Error handling

Query 10 is a simple case to demonstrate how the scripts handles errors in the code that may be triggered by query 11 or by a connection error. All query 10 does is trigger our scripts error handling process as seen in *Figure 3*. That alone may seem unimpressive since we are not passing any legitimate errors. Our error handler fortunately does much more when is triggered at any point during a query execution, will print out the error type we had experienced as well as issue a rollback to prevent any data corruption.

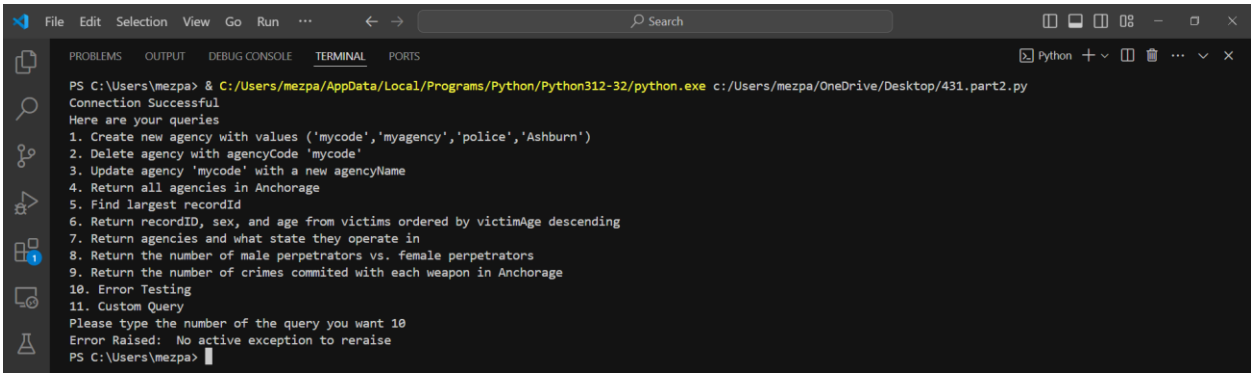

Figure 3: Query 10 when everything is normal.

Query 11

Query 11 is an additional and special case, presented to the users for those trained in creating SQL queries. This case opens room for users to input their own queries that are still protected by our error handling and rollback protection. Any improperly written queries will execute the error handling block and be prevented from creating any possible data hazards, such as what may be seen in *Figure 4*.

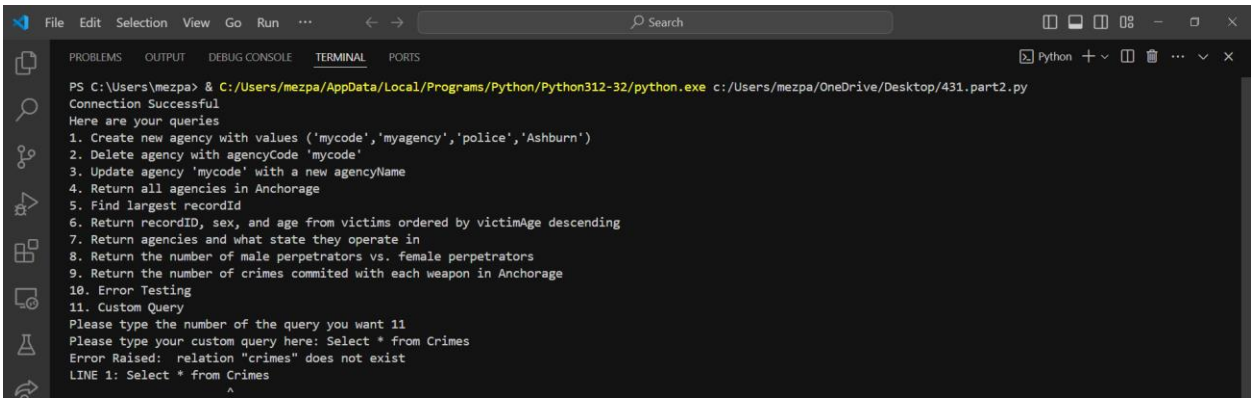

Figure 4: Error raised from naming an unknown table.