Name: \_\_\_\_\_Adam Capdeville\_\_ In-Class

Database Systems September 25, 2019

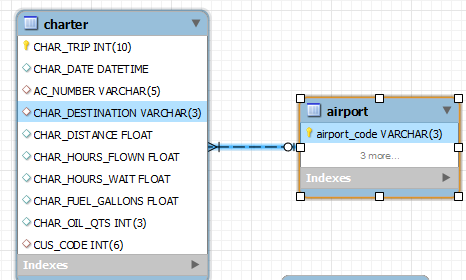
**Part I. (10pts)**

*Preprocessing*

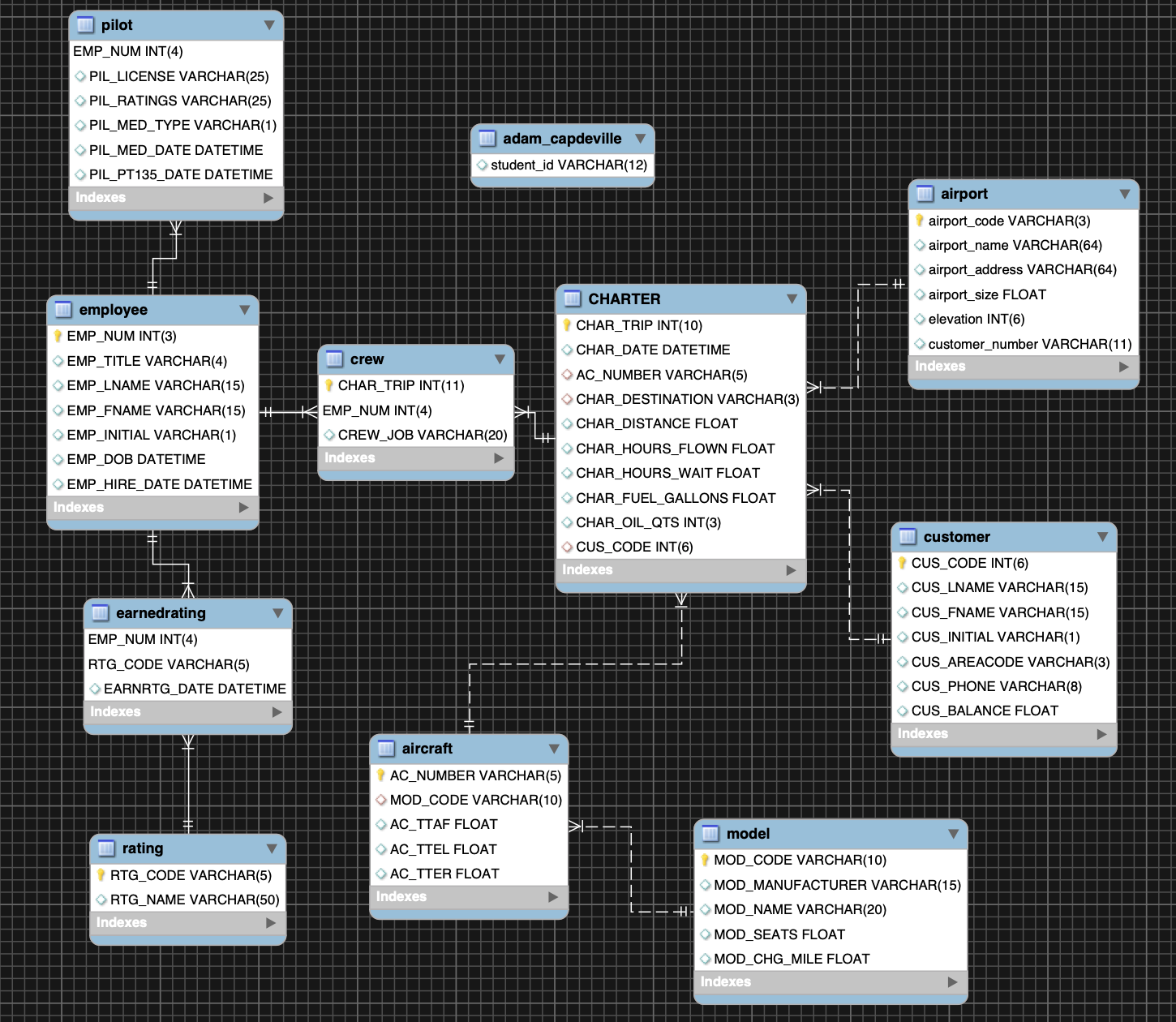
* Run AviationCo.sql, which is the DDL to create the database, AviationCo.
* Minimize redundancy where appropriate.
* Create the proper database constraints to enforce referential integrity for all tables.
* Insert a new customer with your name and information.

*End Preprocessing*

1. Construct a new table, *AIRPORT*. Include the primary key attribute, airport\_code. Include at least 5 additional attributes. Use the internet to identify relevant attributes.
2. Insert relevant airport data based on existing AviationCo charters. Use the Internet to obtain airport data for your additional attributes. This data will be used in Part II and III.
3. Establish entity and relational integrity as follows:



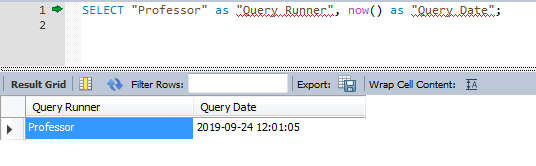
1. (4pts) Reverse engineer the ERD for AviationCo using MySQL workbench. Show all relationships clearly. Your ERD should consist of two additional entities, *YOURNAME and AIRPORT* along with their relationships*.*

******

**Part II. Get Set (20pts)**

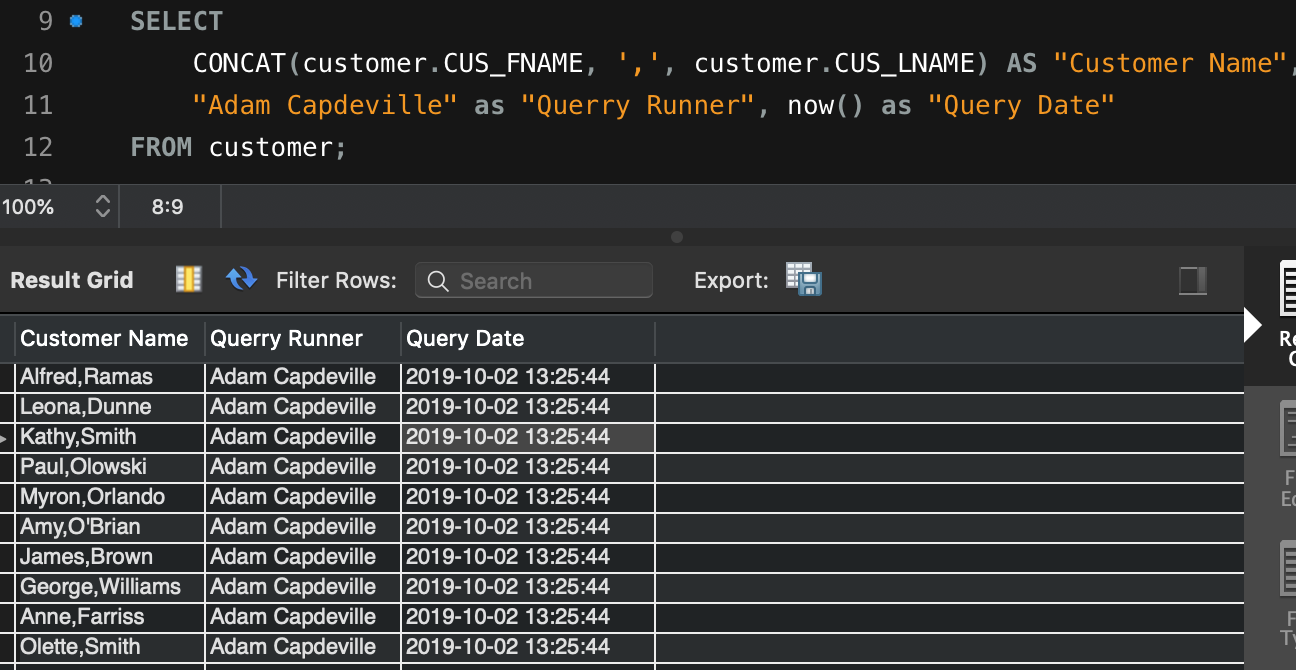
***IMPORTANT: For each query, be sure to include a column, Query Creator and Date to print your full name and current date alongside your query results.***

Example: Construct a query to return your name and the current date.

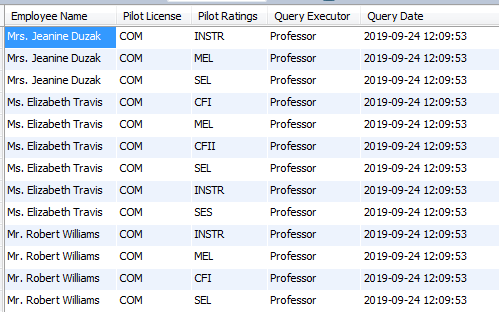


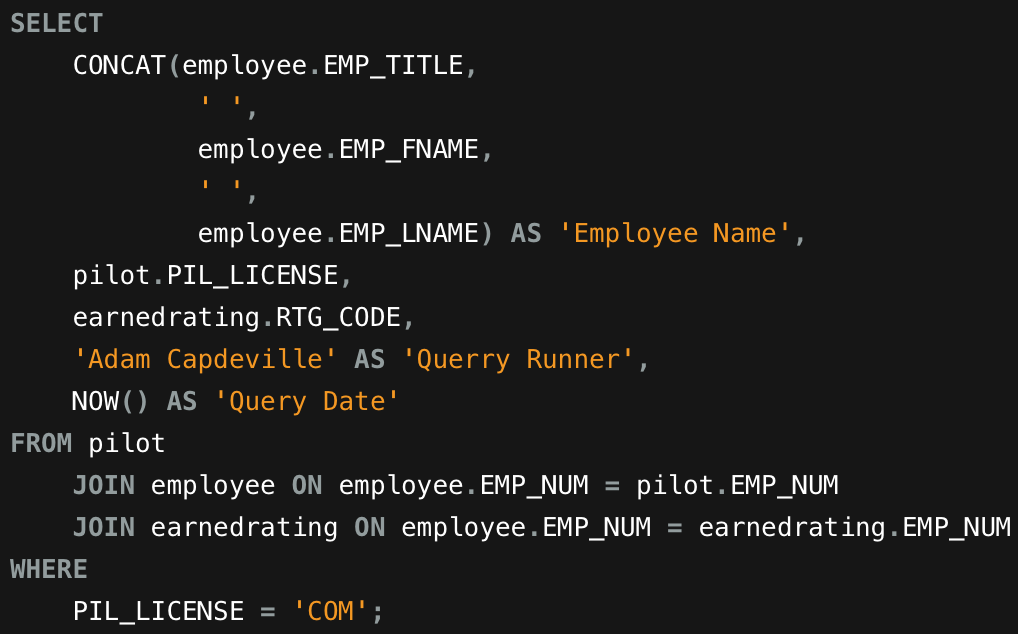
1. Construct a query to return the concatenated first and last names of customers with your last name.

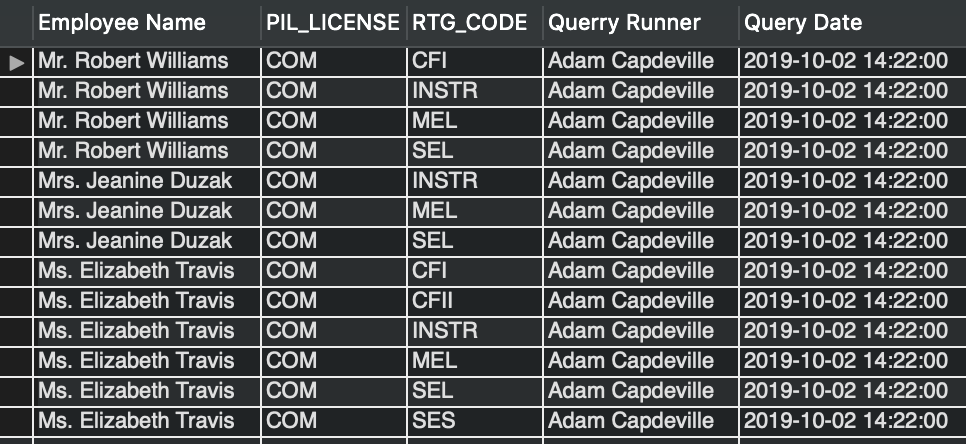




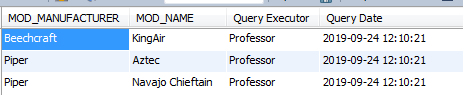
1. Construct a query to return the concatenated first and last names of pilots along with their license and ratings for all commercial pilots. Order by employee last name.

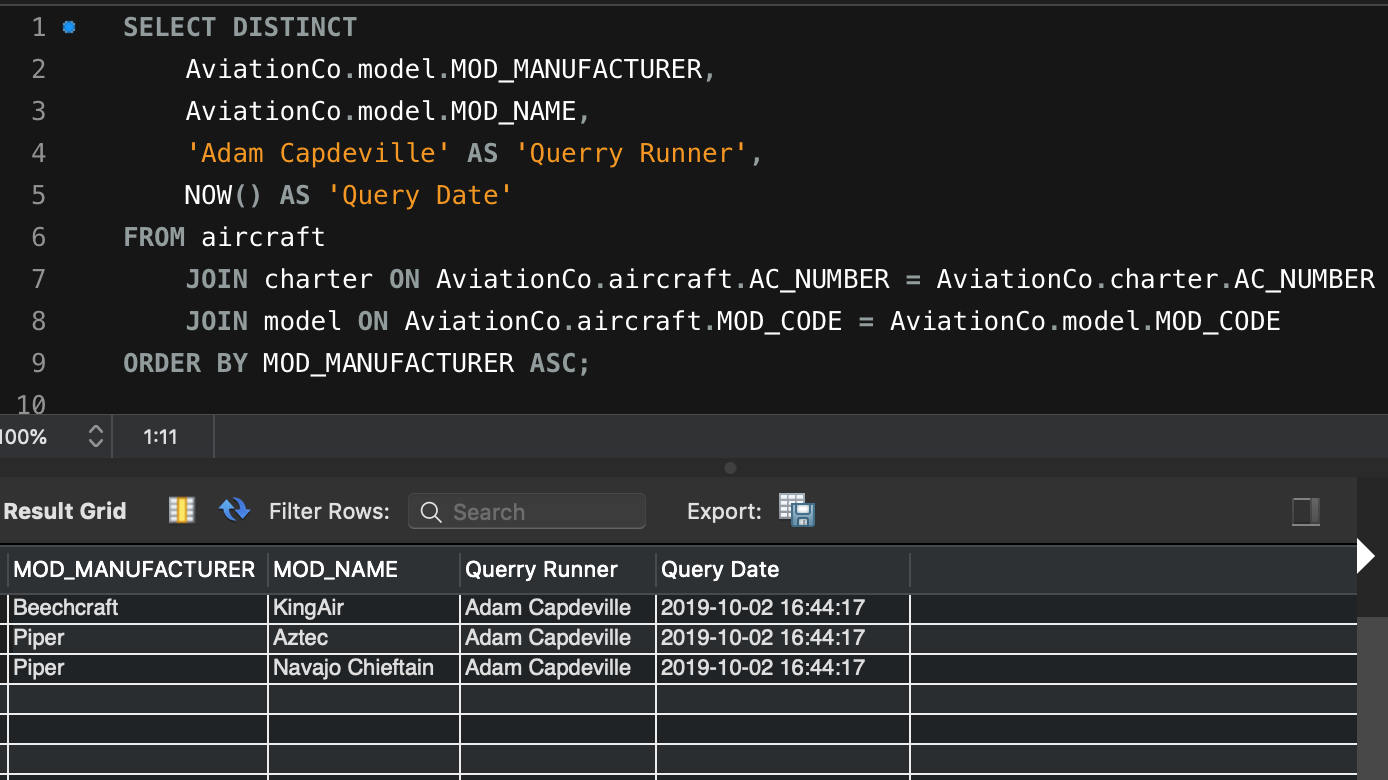




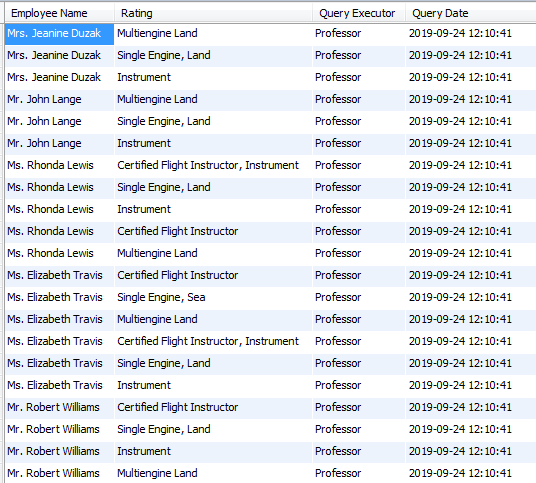


1. Construct a query to select distinct aircraft models for aircrafts with chartered flights. Order by manufacturer ascending.

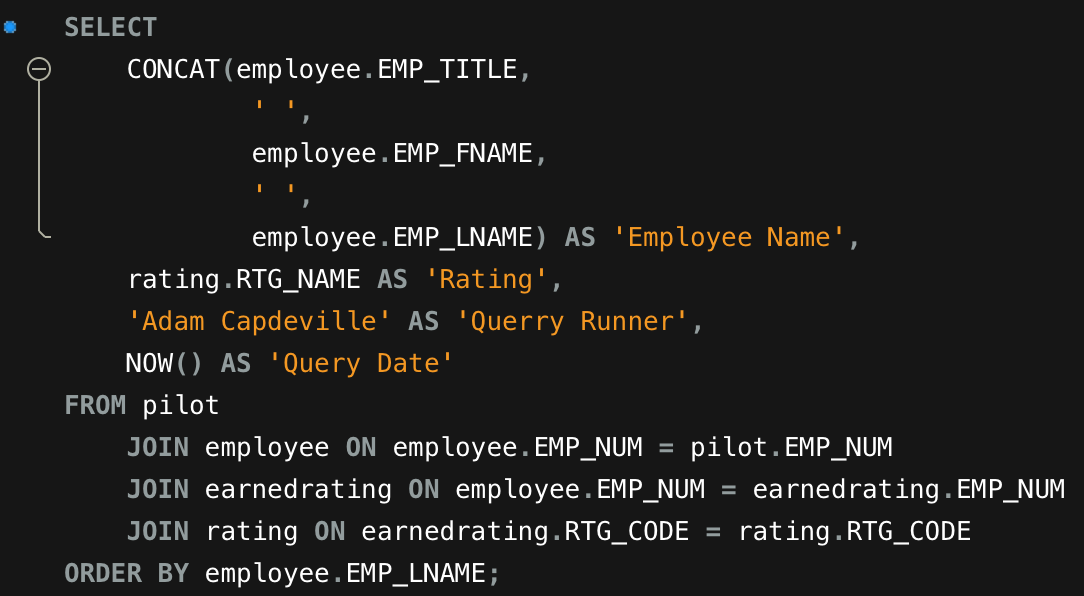


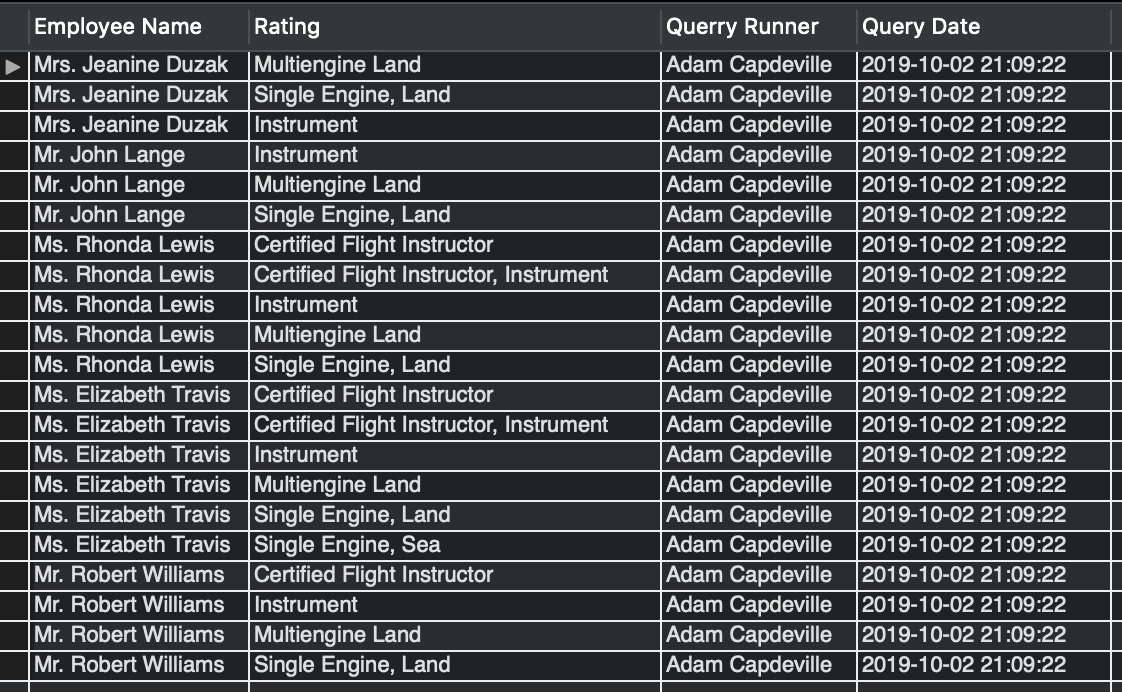


1. Construct a query to return the concatenated first and last name of employees along with their appropriate pilot ratings as shown below. Order by employee last name ascending.

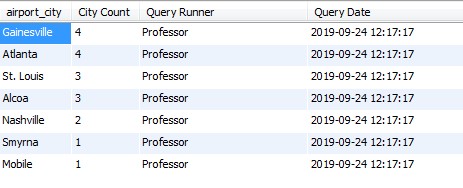


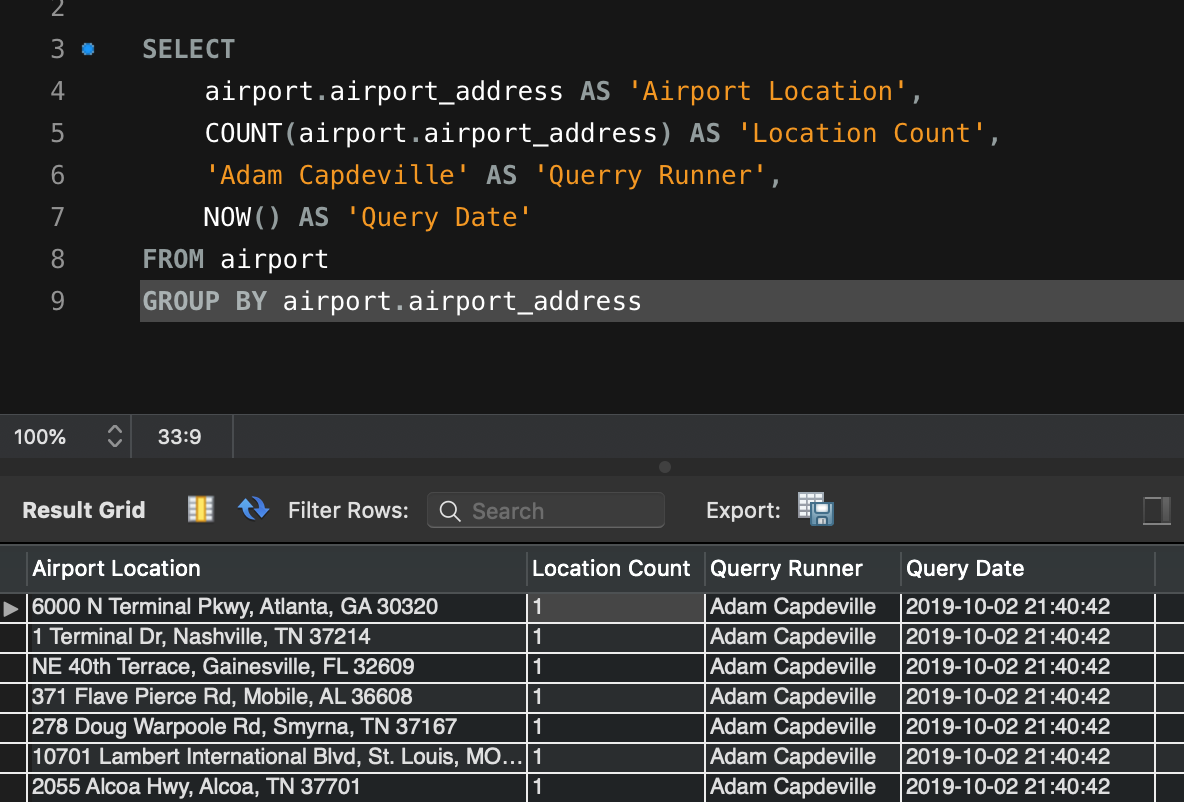
4:



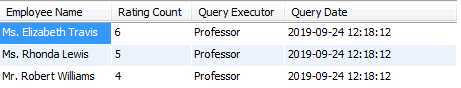


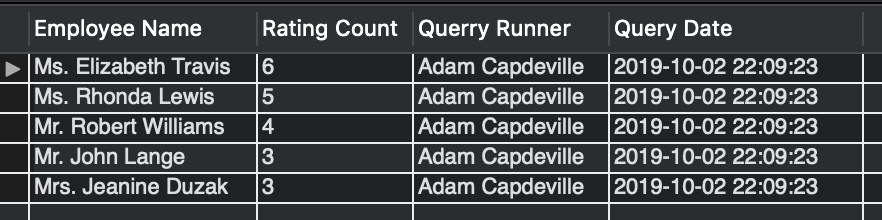
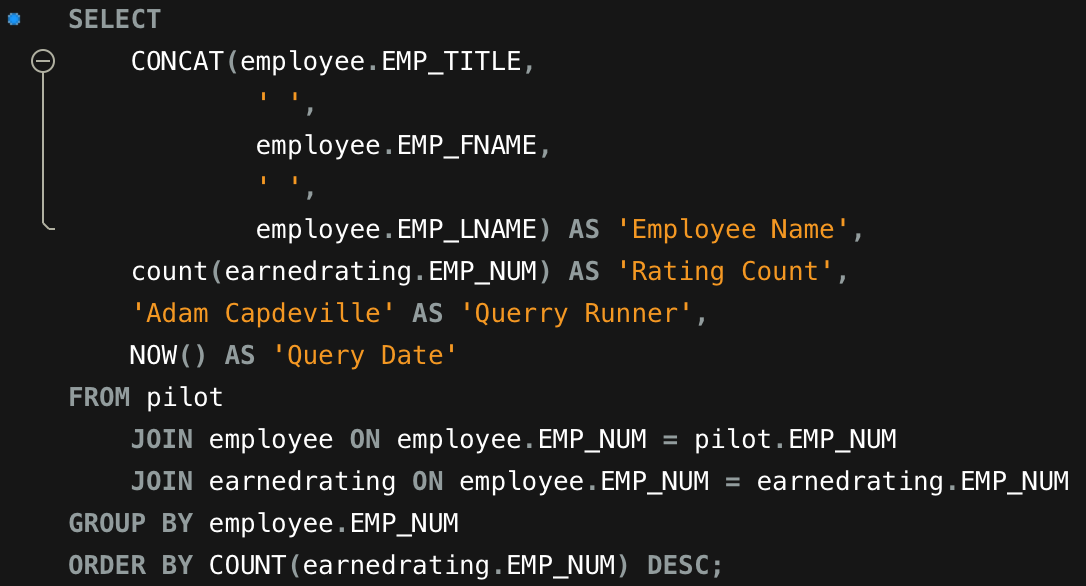
1. Construct a query to rank destinations.





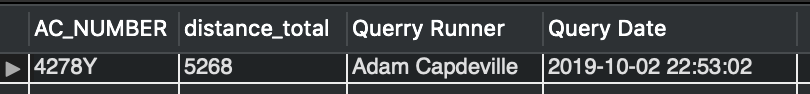
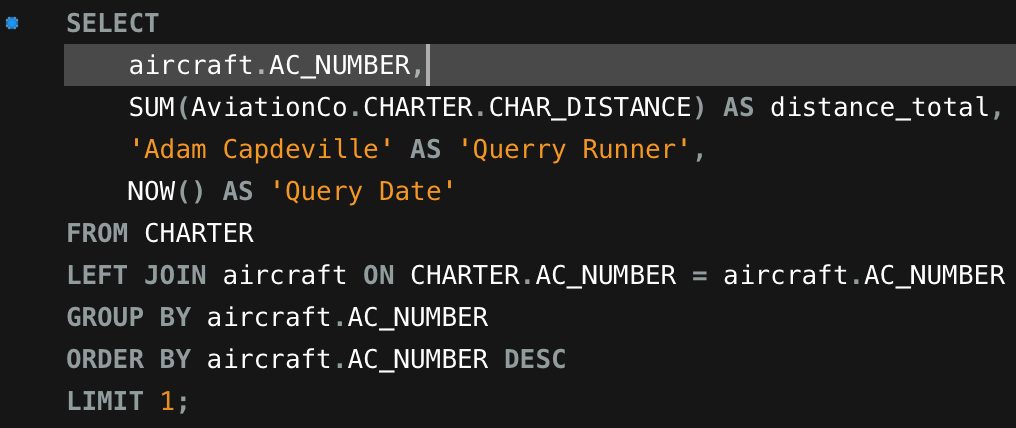
1. Construct a query to return the concatenated first and last name of employees along with the count of their rating codes for employees with more than 3 ratings. Order by their ratings count in descending order.



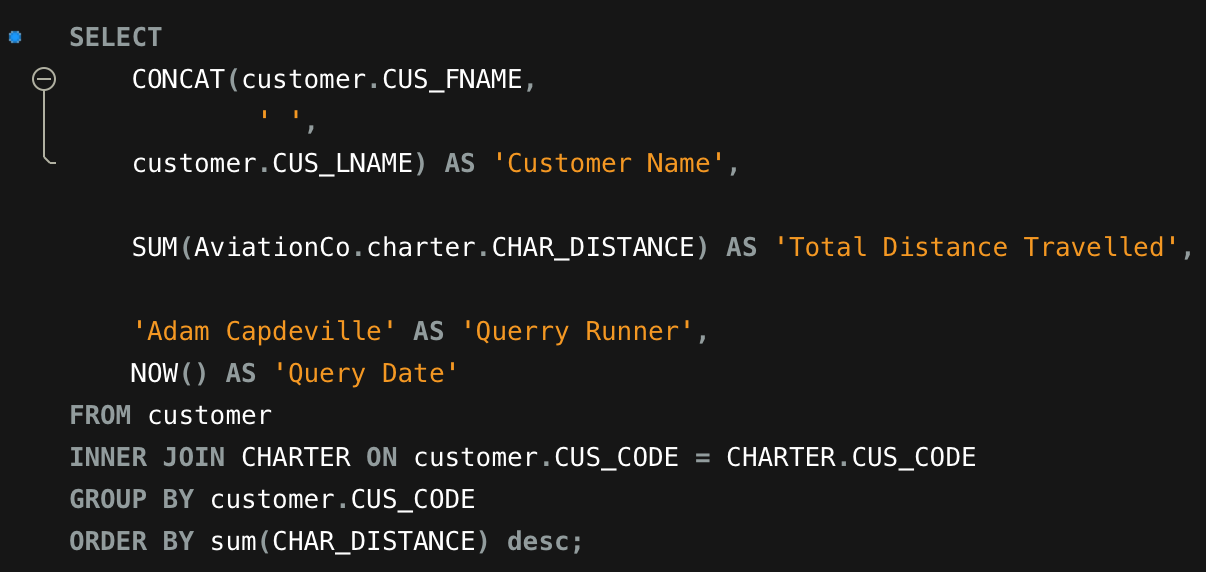
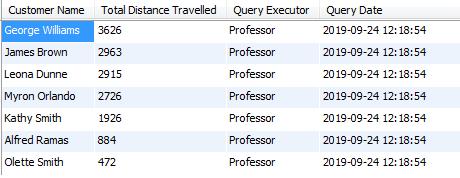


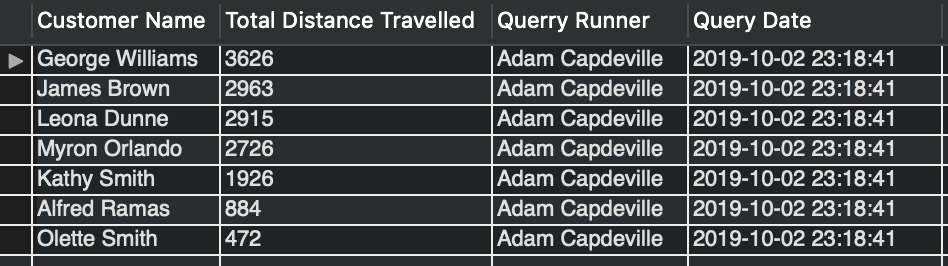
1. Construct a query to find the aircraft with the longest total distance chartered.



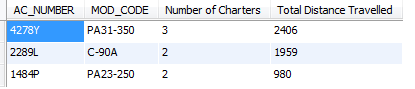


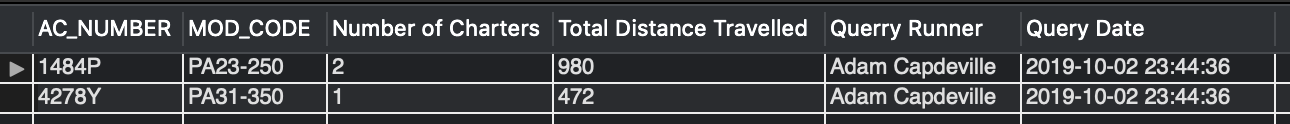
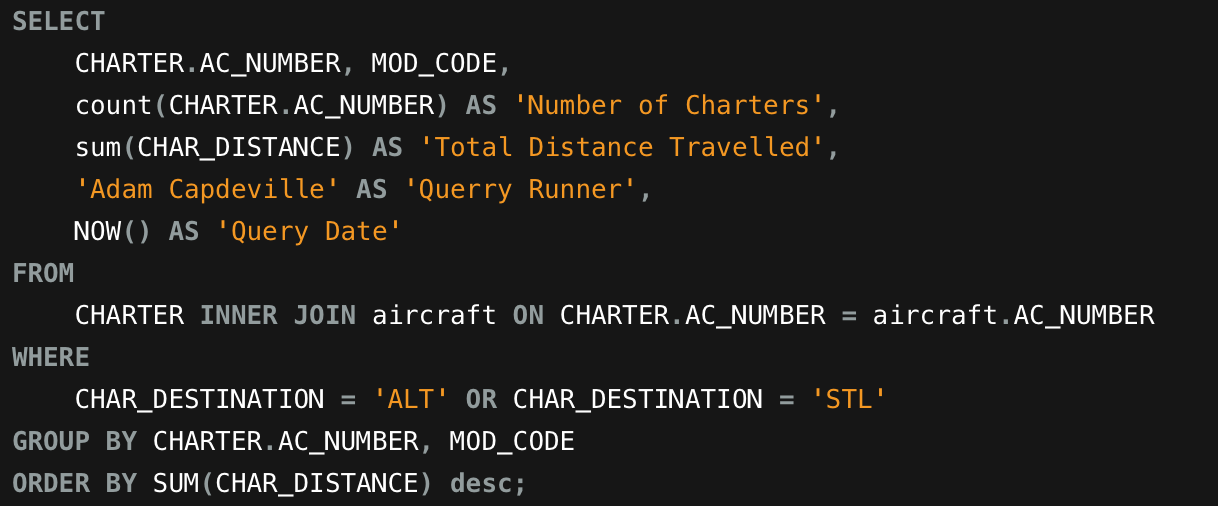
1. Construct a query to return all customers and their distance travelled.



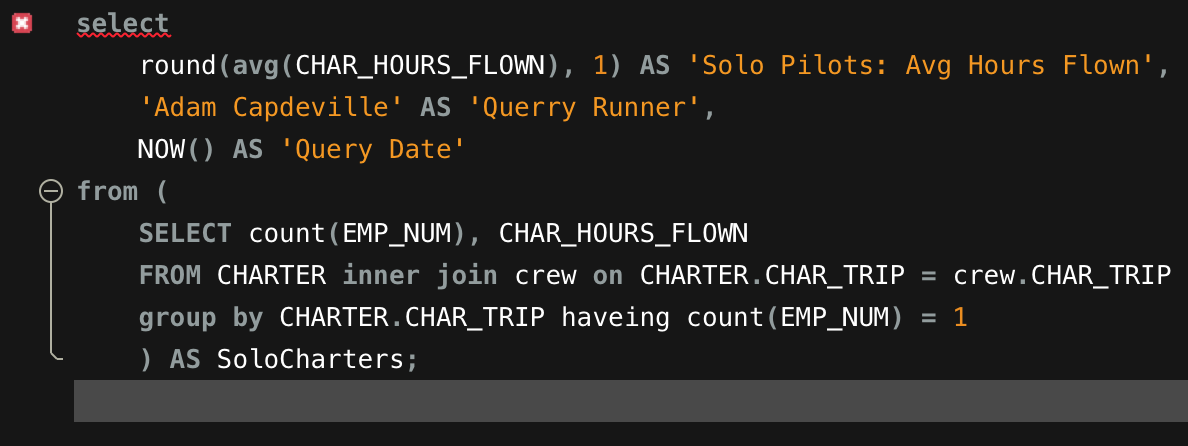


1. Construct a query to return aircrafts, their model codes and number of charters for charters to St. Louis or Atlanta. Order your results by total distance these aircrafts have flown.





1. Construct a query to return the average flight time for chartered flights with only 1 crew member (aka, solo pilots). Your query will use a group by to return charters with only 1 crew member and a virtual table to calculate the average.



Couldn’t get this to work...

**Part III. Go (10pts)**

Using similar syntax above, construct five additional queries based on the criteria below. Each query should make use of your newly constructed AIRPORT entity. Also include a short description of the information each query returns.

1. Incorporate a compound WHERE statement. Describe query objective and data returned.
2. Incorporate a statistical function. Describe query objective and data returned.
3. Incorporate a three table join. Describe query objective and data returned.
4. Incorporate GROUP BY / HAVING. Describe query objective and data returned.
5. Incorporate a sub-query in the WHERE clause. Describe query objective and data returned.