Database Management Report

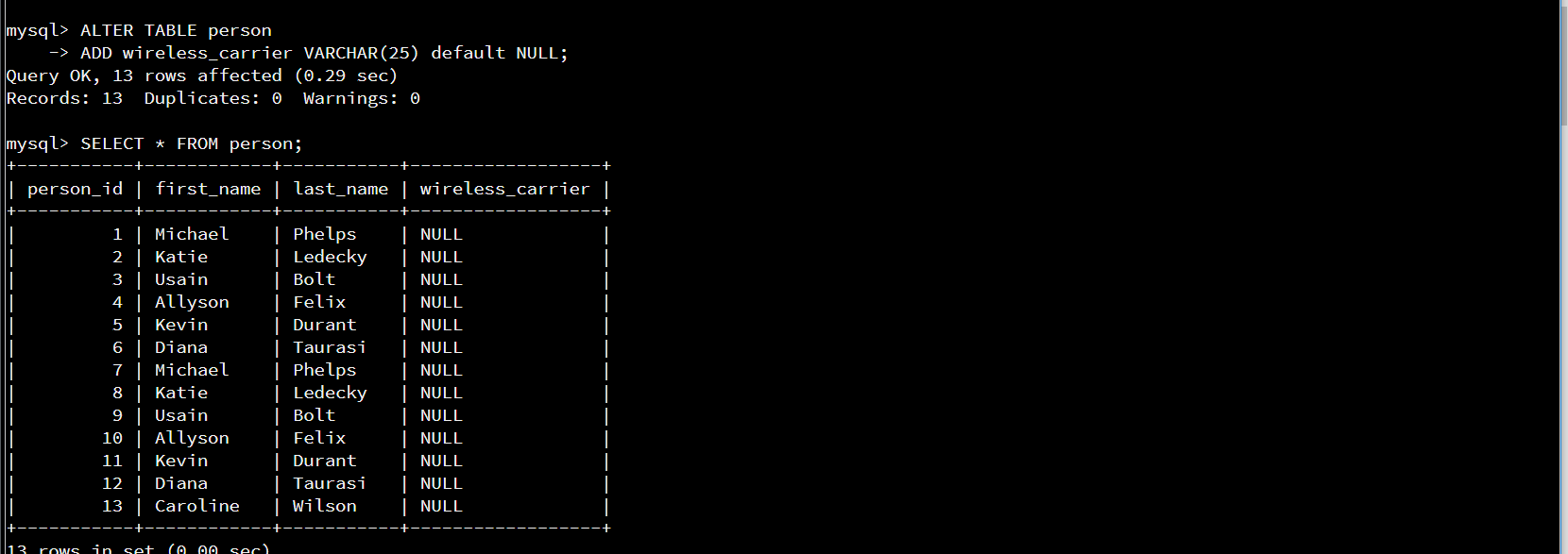
Caroline Wilson

For the purposes of this final, I was asked to imagine that I was a database engineer for a social media messaging start up. I was tasked with creating and altering tables and data within a given scripted database. The following is a representation on how I completed the tasks that were given within the given guidelines.

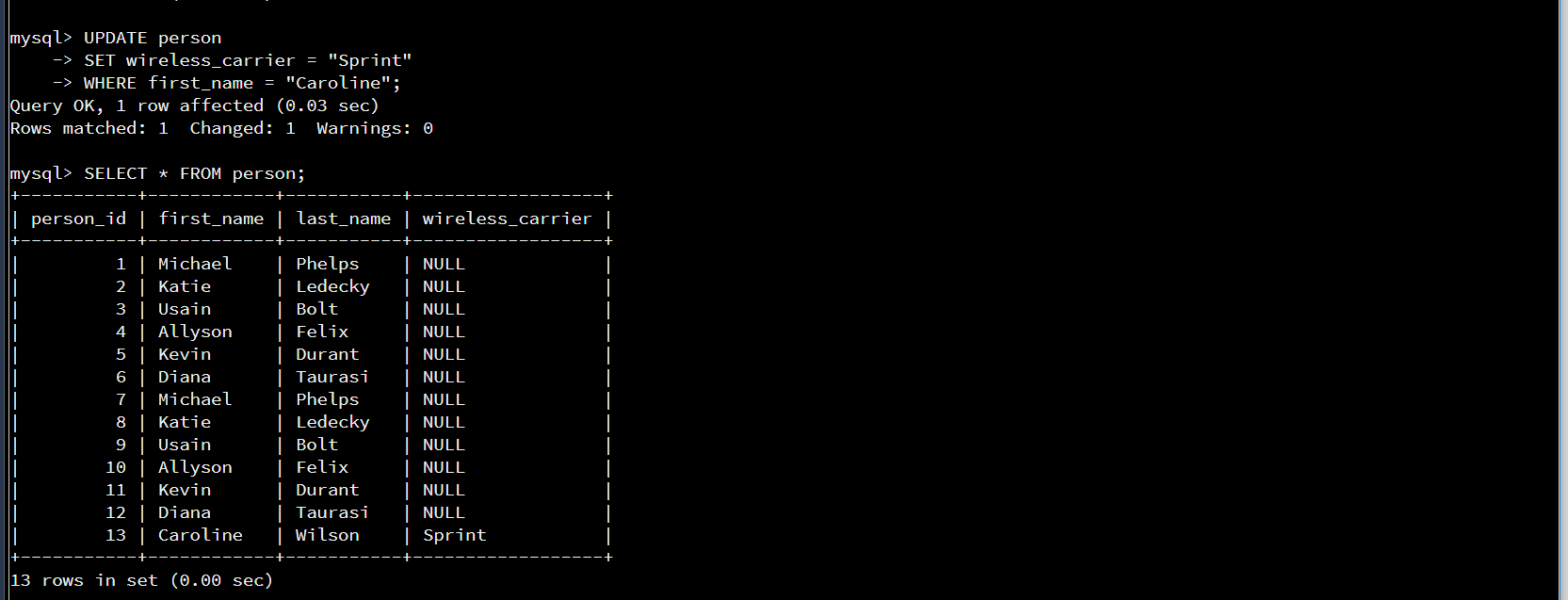
The first step was to use the database that was given to me. In order to do this, I used the command, “USE messaging.” From there I was to add myself into the list of people that were in the given database. The table I was to be entered into was the “person” table. I used the command “INSERT INTO”, followed by the parameters that I was going to input and the Values therein. The following screenshot represents these steps.



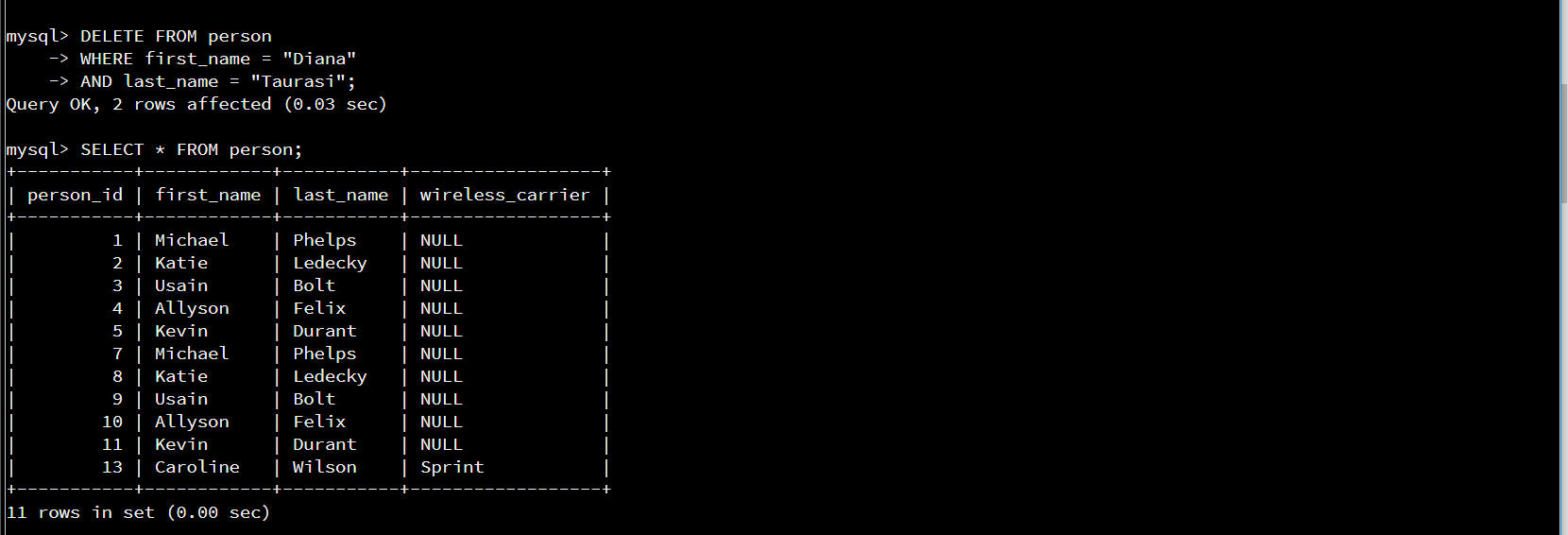
My next task was to add an additional column to the same table, person. In order to do this, I had to use the command “ALTER TABLE”, list the table to be altered and enter the name of the column and the parameters for the data that would be entered. For this, I chose to use “wireless carrier”. I made it a VARCHAR input, meaning that it could be alphanumeric and gave it the option to be left blank by using the “default NULL” option. The following screen shot depicts this step, as well as my check, using “SELECT \* FROM” to ensure that the command was completed correctly.



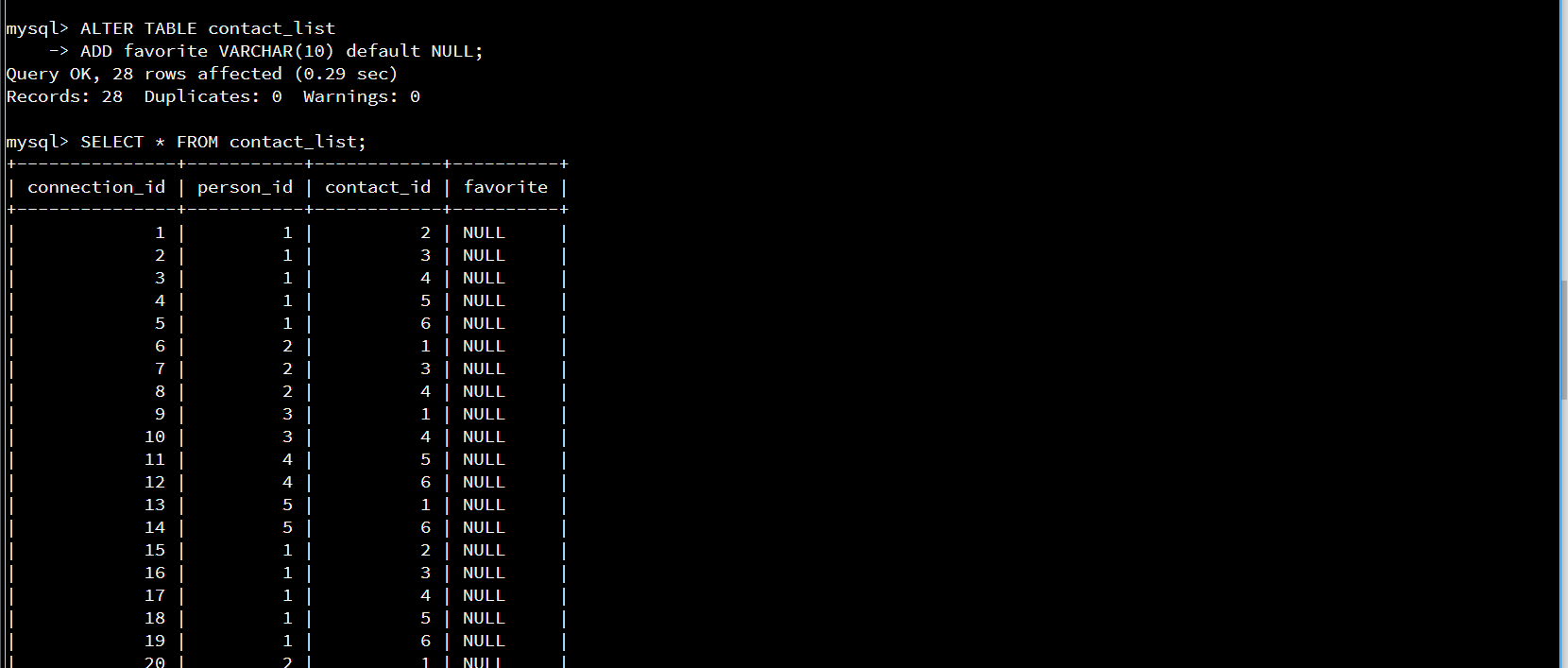
My next step was to utilize the column that I had just created by inputting a selection for myself. To do this, I used the “UPDATE”, “SET”, and “WHERE” commands. I chose the table to update, I set the new column to my carrier, and used the WHERE statement to have it only apply to myself by using it to find my first name. These steps are shown below.



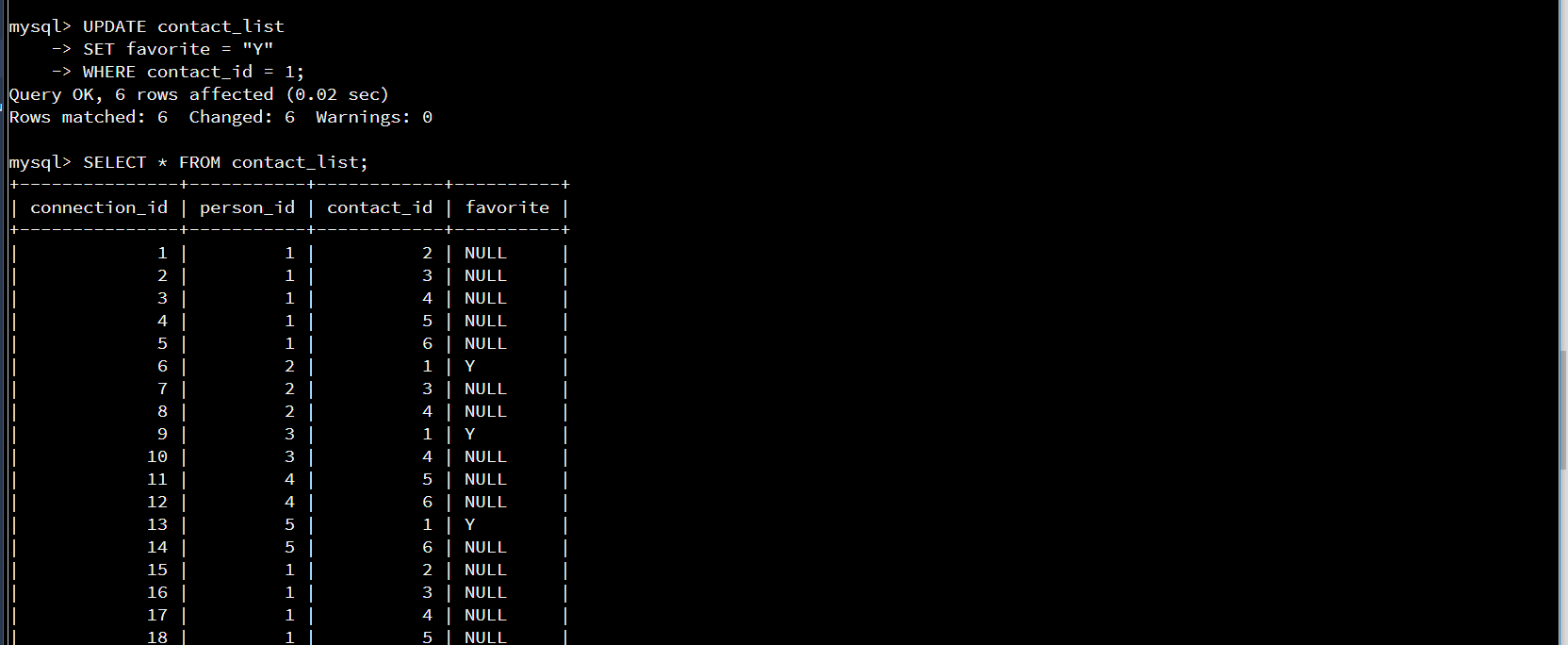
My next task was to delete a person from the table. In order to do this, I used the “DELETE”, “WHERE” and “AND” commands. This allowed me to delete from the table by setting guidelines. In this case, I used the where statement to find someone who had the first name, “Diana”, and the last name “Taurasi”. This step is shown in the following screenshot.

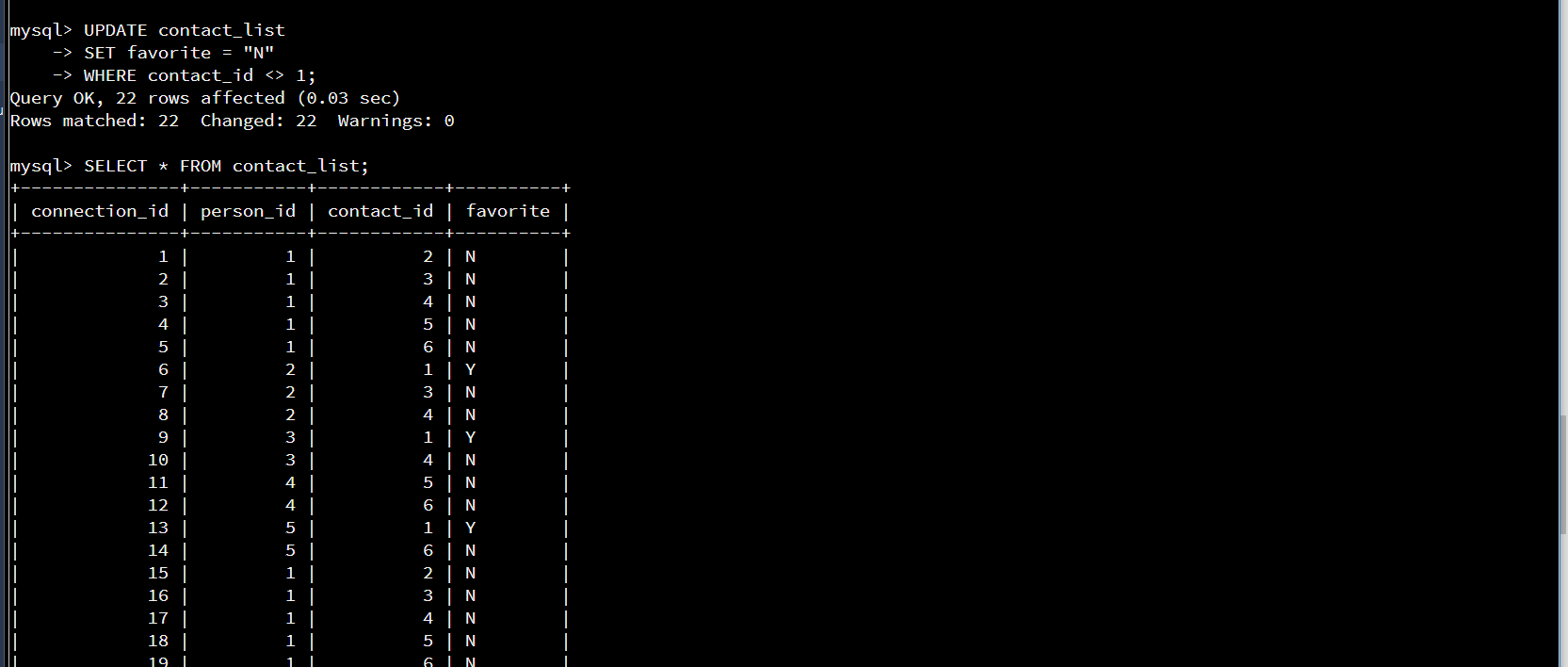


The next step was to add another column to the table, “contact\_list”. I executed the same steps as in the earlier step when I added the wireless carrier. I followed the guidelines for this column as given by the guidelines and rubric sheet. The entry and results are shown in the following screenshot.

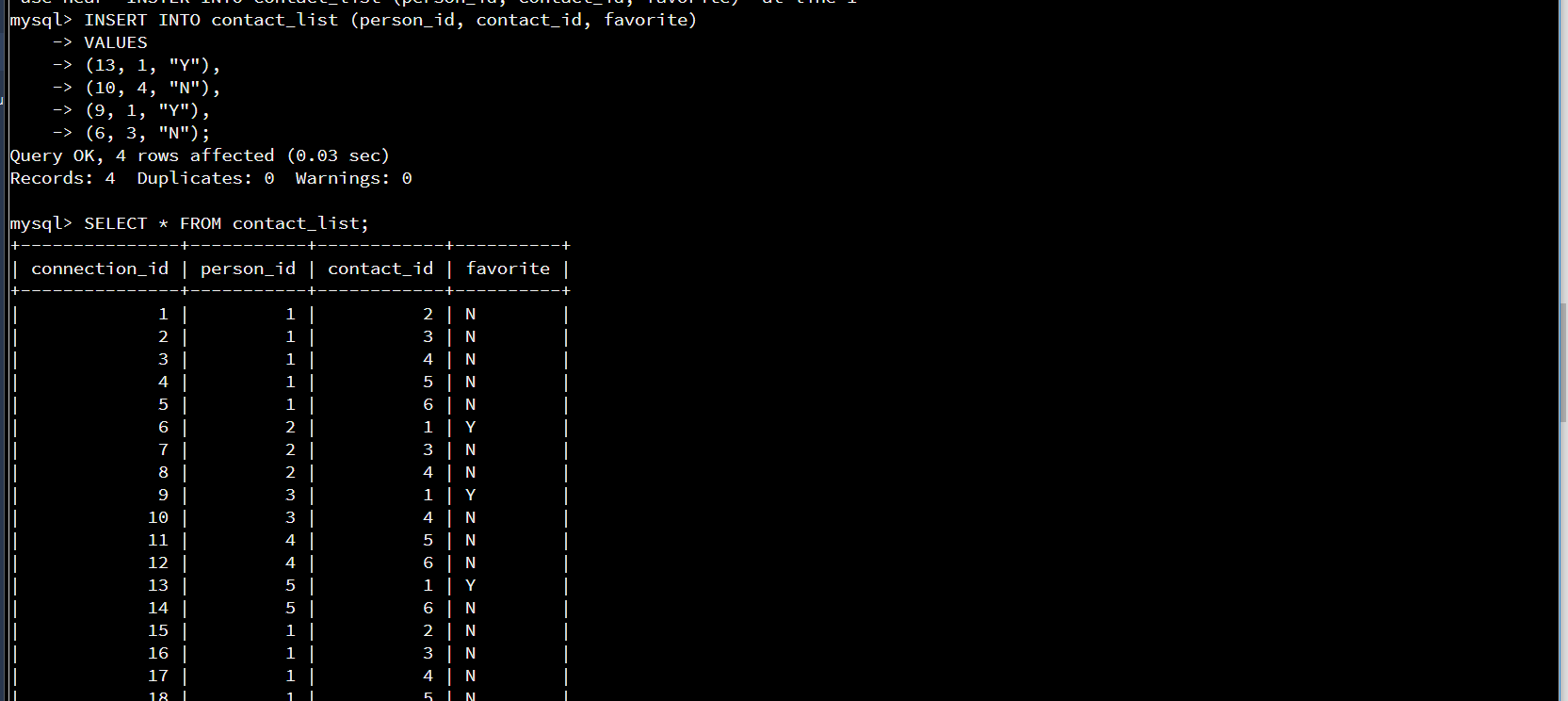


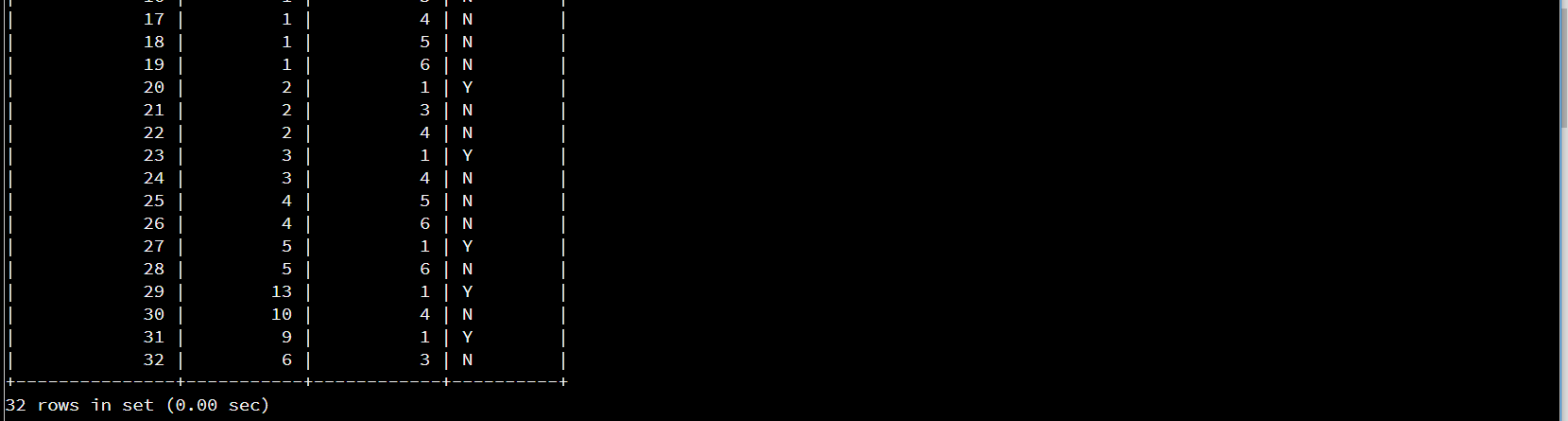
The next two steps were to populate the column that I had just created by marking “Michael Phelps” as the favorite contact. In order to do this, I had to find that his contact ID was 1. I updated the contact list using “UPDATE”, “SET” and “WHERE” to mark a “y” if the contact id was equal to one, and a “n” if it was out of range. The steps and table checks are shown in the next two screenshots.

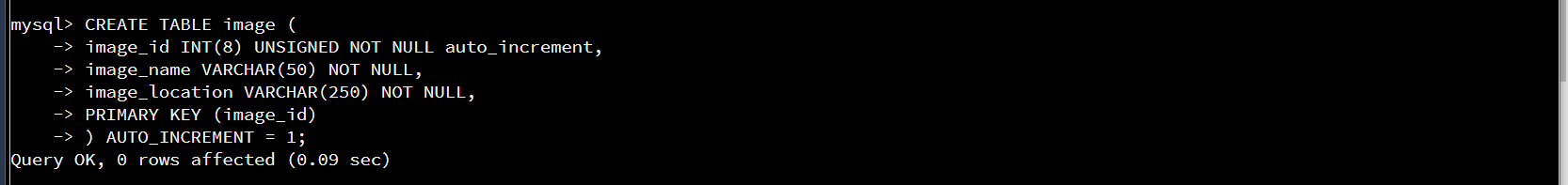




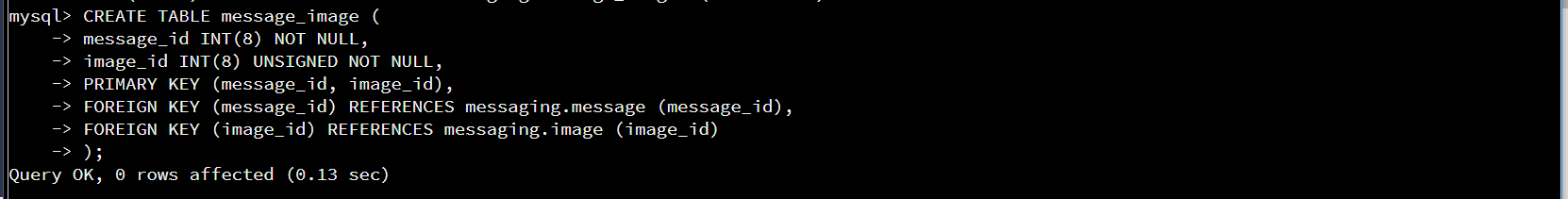
The next task to execute was to add additional people to the contact list, myself included. I know from the previous check that my person id was 13. In order to complete this, I used the   
“INSERT INTO” command. This was followed by the information that was to be inputted, including the column that I had just created, and the values of the data to be entered. The following screenshots show the means of accomplishing this task and the outcome of it.



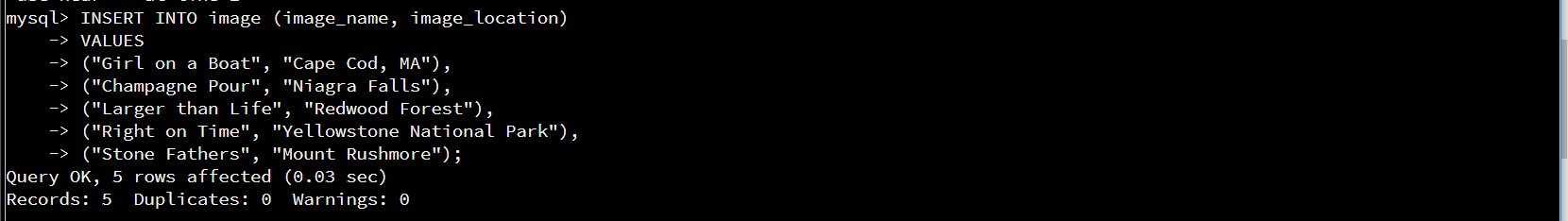
 My next step was to create a new table that would allow for information about images that were to be sent via the app. In order to do this, I had to know that values would eventually be entered, and how the data should be entered. This information was given to me in the guidelines. In order to create the table, I used the command “CREATE TABLE” and the following lines would set the parameters for each of the columns therein. The following picture represents that creation.

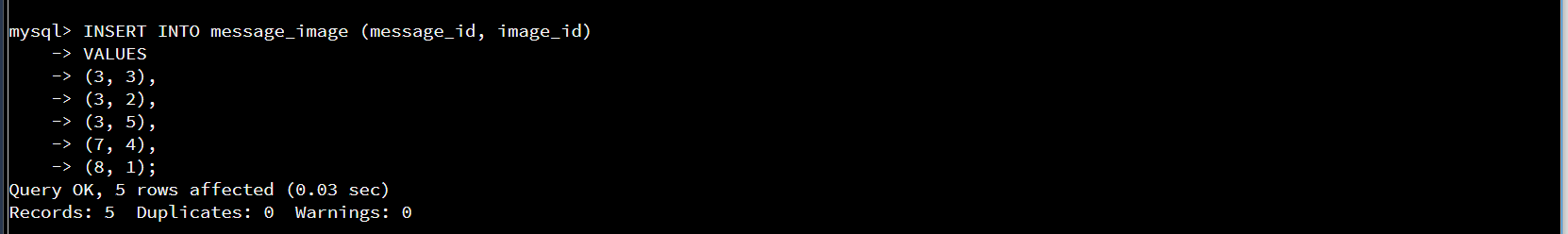


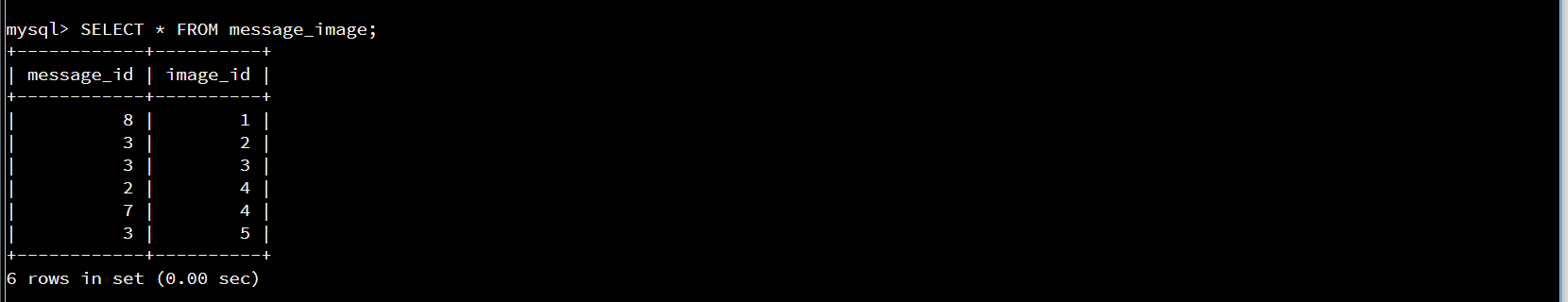
After the image table was created, I had to create an intersection table that would represent the connection between the message and image tables. In order to do this, I had to use the “CREATE TABLE” command followed by the parameters of the information that would eventually be inputted. This table differed from the last in the fact that I had to use “FOREIGN KEY” and “REFERENCES” to represent the connection between the two tables and how they were to interact with each other. The following screenshot shows the creation of this table.



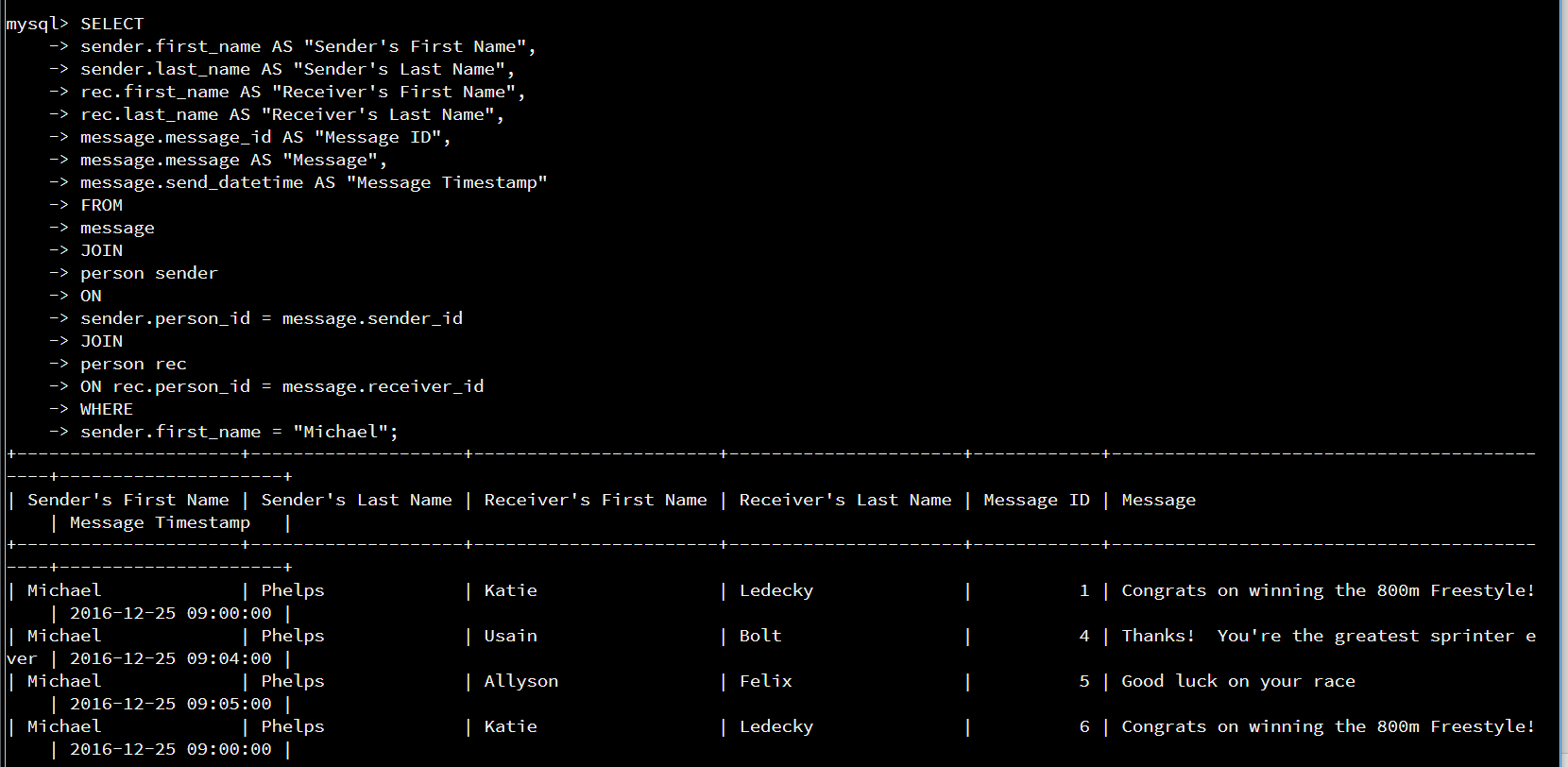
After the tables were created, I had to again use the “INSERT INTO” command along with parameters and data to populate the tables with the given data. The following pictures show how this data was entered and the outcome of the input.

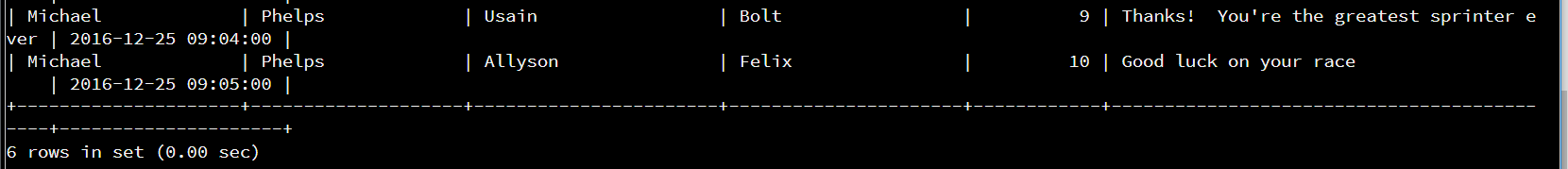




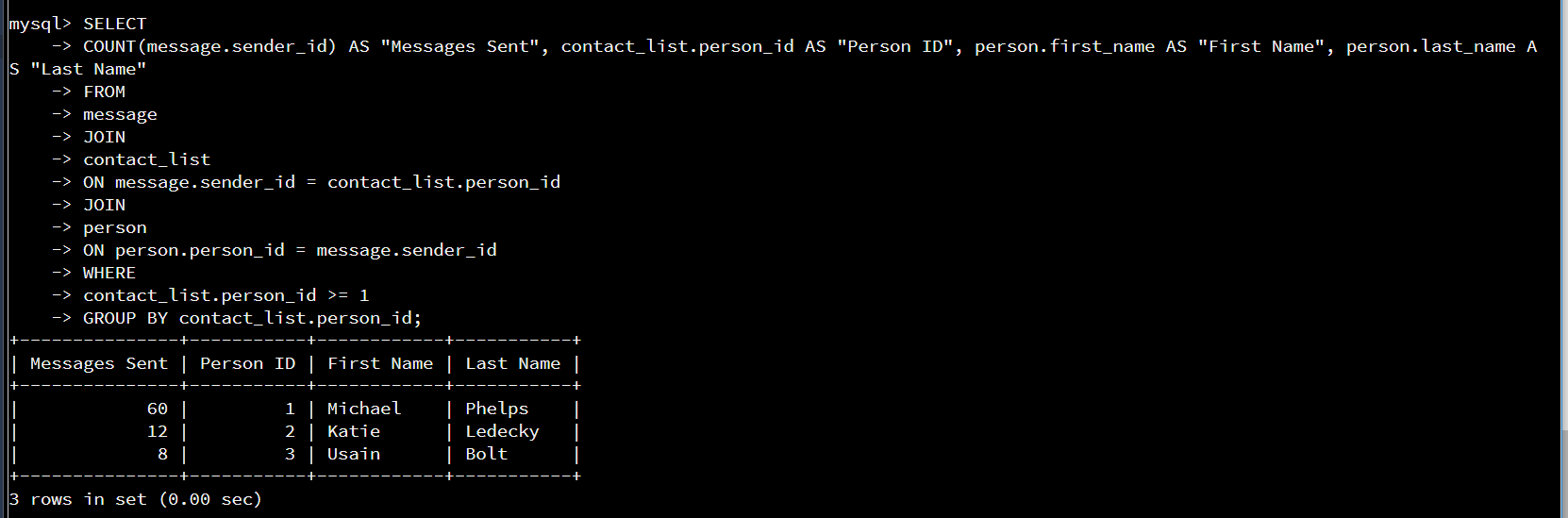


Next, I was asked to access information within the database. My first step in this was to find all of the messages that Michael Phelps had sent using the WHERE clause. I found that it was best in this case to create a sort of temporary file to join the tables together to make it easier to draw from more than one table. I first had to list the information that I wanted to show using the “SELECT” command and made it easier to read using the “AS” addition. I then had to depict the tables that I was drawing from and JOIN them so that they would be able to bring information from multiple tables and have them work with one another in the output. The following are screenshots of both my input and output for this step.

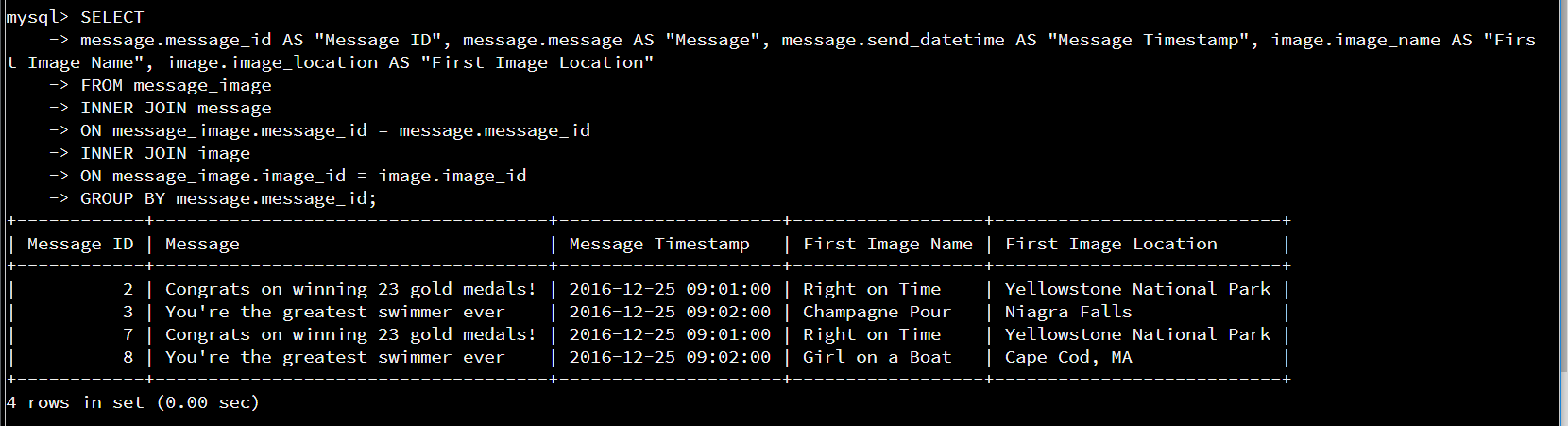




My next task was to use the “COUNT” command to see how many messages were sent by each of the senders. In order to do this, I used the “SELECT”, and “COUNT” commands. I again used the “AS” addition to make the table more easily read. I again had to use “JOIN” so that information could be drawn logically from more than one table. I used the “WHERE” command to narrow the results, and the “GROUP BY” line so that the table would correctly represent in the information that I was trying to gather in a more logical output. Both my input and the output are shown in the following screenshot.



Finally, I was tasked to find all of the messages that had been sent with a picture attached to them. I was asked to use “INNER JOIN” to allow for the tables to represent shared data correctly. I again used “SELECT” and “AS” to pull and represent the data more easily. I then inputted the tables that I wished to draw information from, used “INNER JOIN” and “ON” to represent the tables and data that should be outputted and how they correlate to one another. I then used “GROUP BY” to correctly show the information in a comprehensive table. This also allowed for only the first instance of a message to be represented. The following screenshot shows the commands and output for this final step.



For the purposes of this final I was tasked with imagining myself in the place of a database engineer. I was tasked with creating and altering tables and to show the ability to draw information from these tables based on given guidelines and parameters. I then explained each step and the commands that were used and accompanied them with screenshots of my results. The formentioned explanations and pictures were the representation of those steps taken.