# DAD 220 Database Documentation Template

Complete these steps as you work through the directions for Project One. Replace the bracketed text with your screenshots and brief explanations of the work they capture. Each screenshot and its explanation should be sized to approximately one quarter of the page, with the description written below the screenshot. Follow these rules for each of the prompts and questions below. Review the example document located in the Project One Supporting Materials for assistance.

## Step One: Create a Database

1. Navigate to your online integrated development environment (IDE). List and record the SQL commands that you used to complete this step here:

Text

Description automatically generated

Code: mysql

Typing mysql connects to the MySQL monitor.

1. Create a database schema called QuantigrationUpdates. List out the database name. Provide the SQL commands you ran against MySQL to successfully complete this in your answer:

Text

Description automatically generated

Code: create database QuantigrationUpdates;

Code: Show databases;

I used the create database command to create the QuantigrationUpdates database, and then used show databases in order to verify that it was correctly entered.

1. Using the entity relationship diagram (ERD) as a reference, create the following tables with the appropriate attributes and keys:
   1. A table named **Customers** in the QuantigrationUpdates database, as defined on the project ERD. Provide the SQL commands you ran against MySQL to complete this successfully in your answer:

A picture containing graphical user interface

Description automatically generated

Code: create table Customers (CustomerID int, FirstName varchar(25), LastName varchar(25), Street Varchar(50), City Varchar(50), State varchar(25), ZipCode int, Telephone varchar(15));

Code: Describe Customers;

Using create table, I created the table Customers with 8 different fields (CustomerID being the primary key). I then used describe to ensure that all fields were entered correctly with the correct data type.

* 1. A table named **Orders** in the QuantigrationUpdates database, as defined on the project ERD. Provide the SQL commands you ran against MySQL to complete this successfully in your answer:

Graphical user interface

Description automatically generated with medium confidence

Code: create table Orders (OrderID int, CustomerID int, SKU varchar(20), Description varchar(50));

Code: Describe Orders;

Using create table, I created the table Orders with 4 different fields (OrderID being the primary key and the CustomerID being the foreign key). I made sure everything was entered correctly by using the describe query.

* 1. A table named **RMA** in the QuantigrationUpdates database, as defined on the project ERD. Provide the SQL commands you ran against MySQL to complete this successfully in your answer:

A picture containing graphical user interface

Description automatically generated

Code: create table RMA (RMAID int, OrderID int, Step varchar(50), Status varchar(15), Reason varchar(15));

Code: Describe RMA;

Using create table, I created the table RMA with 5 different fields (RMAID being the primary key and the OrderID being the foreign key). I made sure everything was entered correctly by using the describe query.

## Step Two: Load and Query the Data

1. **Import the data from each file into tables.** 
   * Use the QuantigrationUpdates database, the three tables you created, and the three CSV files preloaded into Codio.
   * Use the import utility of your database program to load the data from each file into the table of the same name. You will perform this step three times, once for each table.

Text

Description automatically generated

Code: load data infile '/home/codio/workspace/rma.csv' into table RMA fields terminated by ',' lines terminated by '\r\n';

Code: load data infile '/home/codio/workspace/customers.csv' into table Customers fields terminated by ',' lines terminated by '\r\n';

Code:load data infile '/home/codio/workspace/orders.csv' into table Orders fields terminated by ',' lines terminated by '\r\n';

Via the infile query, I loaded the CSV files into their respective tables, and each field is separated by a comma, with each record separated by a new line.

1. **Write basic queries against imported tables to organize and analyze targeted data.** For each query, replace the bracketed text with a screenshot of the query and its output. You should also include a 1- to 3-sentence description of the output.
   * Write an SQL query that returns the **count** of orders for customers located only in the city of Framingham, Massachusetts.
     1. How many records were returned? -**505 total records**

Text

Description automatically generated with medium confidence

Code: select count(\*) from Customers inner join Orders on Customers.CustomerID = Orders.CustomerID where Customers.City = 'Framingham' and Customers.State = 'Massachusetts';

With a select count(\*) query and inner joining the Customers and Orders table on the CustomerID field, I added a where clause where only records where the City field in Customers was ‘Framingham’ and where the State field in Customers was ‘Massachusetts.’

* + Write an SQL query to **select all** of the Customers located in the state of Massachusetts.
    1. Use a WHERE clause to limit the number of records in the Customers table to only those who are located in Massachusetts.
    2. Record an answer to the following question: How many records were returned? **– 982 total records**

Text

Description automatically generated with low confidence

Code: select\* from Customers where State = 'Massachusetts';

Using another select count(\*) query, I first wanted to find the total amount of records where the State field equaled ‘Massachusetts,’ and then I verified that information was entered correctly with a select\* query from Customers where the State field equaled ‘Massachusetts.’

* + Write a SQL query to insert four new records into the Orders and Customers tables using the following data:

**Customers Table**

| **CustomerID** | **FirstName** | **LastName** | **StreetAddress** | **City** | **State** | **ZipCode** | **Telephone** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 100004 | Luke | Skywalker | 15 Maiden Lane | New York | NY | 10222 | 212-555-1234 |
| 100005 | Winston | Smith | 123 Sycamore Street | Greensboro | NC | 27401 | 919-555-6623 |
| 100006 | MaryAnne | Jenkins | 1 Coconut Way | Jupiter | FL | 33458 | 321-555-8907 |
| 100007 | Janet | Williams | 55 Redondo Beach Blvd | Torrence | CA | 90501 | 310-555-5678 |

Graphical user interface

Description automatically generated with medium confidence

Code: insert into Customers Values (100004, 'Luke', 'Skywalker', '17 Maiden Lane', 'New York', 'NY', 10222, '212-555-1234'), (100005, 'Winston', 'Smith', '128 Sycamore Street', 'Greensboro', 'NC', 27401, '919-555-6623'), (100006, 'MaryAnne', 'Jenkins', '2 Coconut Way', 'Jupiter', 'FL', 33458, '321-555-8907'), (100007, 'Janet', 'Williams', '58 Redondo Beach Blvd', 'Torrence', 'CA', 90501, '310-555-5678');

Code: Select\* from Customers where CustomerID in (100004, 100005, 100006, 100007);

I used the insert into query to insert the 4 new records into the Customers table. I ran a select\* query from the Customers table with a where clause so that it only pulled up records where the CustomerID field equaled 100004, 100005, 100006, 100007 – the four new records.

**Orders Table**

| **OrderID** | **CustomerID** | **SKU** | **Description** |
| --- | --- | --- | --- |
| 1204305 | 100004 | ADV-24-10C | Advanced Switch 10GigE Copper 24 port |
| 1204306 | 100005 | ADV-48-10F | Advanced Switch 10 GigE Copper/Fiber 44 port copper 4 port fiber |
| 1204307 | 100006 | ENT-24-10F | Enterprise Switch 10GigE SFP+ 24 Port |
| 1204308 | 100007 | ENT-48-10F | Enterprise Switch 10GigE SFP+ 48 port |

A picture containing graphical user interface

Description automatically generated

Code: insert into Orders values (1204305, 100004, 'ADV-24-10C', 'Advanced Switch 10GigE Copper 24 port'), (1204306, 100005, 'ADV-48-10F', 'Advanced Switch 10 GigE Copper/Fiber 44 port copper 4 port fiber'), (1204307, 100006, 'ENT-24-10F', 'Enterprise Switch 10GigE SFP+ 24 port'), (1204308, 100007, 'ENT-48-10F', 'Enterprise Switch 10GigE SFP+ 48 Port');

Code: Select\* from Customers where CustomerID in (100004, 100005, 100006, 100007);

I used the insert into query to insert the 4 new records into the Orders table. I ran a select\* query from the Orders table with a where clause so that it only pulled up records where the CustomerID field equaled 100004, 100005, 100006, 100007 – these were the four new customer records, so I just used the CustomerID foreign key to verify.

* + In the Customers table, perform a query to count all records where the city is Woonsocket, Rhode Island.
    1. How many records are in the Customers table where the field “city” equals “Woonsocket”? – **7 total records**

A computer screen capture

Description automatically generated with low confidence

Code: select count(\*) from Customers where City = 'Woonsocket' and State = 'Rhode Island';

I used a select count(\*) query to pull the total count of records from the Customers table where the City field was ‘Woonsocket’ and State field was ‘Rhode Island.’

* + In the RMA database, update a customer’s records.
    1. Write an SQL statement to select the current fields of **status** and **step** for the record in the **RMA** table with an **orderid** value of “5175.”
       1. What are the current status and step? **The Status field is ‘Pending’ and the Step field is ‘Awaiting customer Documentation’**

Text

Description automatically generated

Code: select Status, Step from RMA where OrderID = 5175;

Wrote a select query of the Status and Step fields from the RMA table where the OrderID equaled 5175.

* + 1. Write an SQL statement to update the **status** and **step** for the **OrderID**, 5175 to **status** = “Complete” and **step** = “Credit Customer Account.”
       1. What are the updated **status** and **step** values for this record? – **The Status field is now ‘Complete’ and the Step field is now ‘Credit Customer Account’**

Graphical user interface, text

Description automatically generated

Code: update RMA set Status = 'Complete', Step = 'Credit Customer Account' where OrderID = 5175;

Using an update query, I set the Status field to ‘Complete’ and the Step field to ‘Credit Customer Account’ to records where the OrderID field equaled 5175.

* + Delete RMA records.
    1. Write an SQL statement to delete all records with a reason of “Rejected.”
       1. How many records were deleted? – **596 total records**

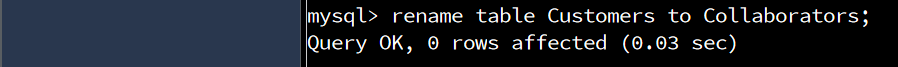
Text

Description automatically generated

Code: delete from RMA where Reason = 'Rejected';

Ran a delete query, I deleted any record from RMA where the Reason field was equal to ‘Rejected.’

1. **Update your existing tables** from “Customer” to “Collaborator” using SQL based on this change in requirements. Provide the SQL commands you ran against MySQL to complete this successfully in your answer:
   1. Rename all instances of “Customer” to “Collaborator.”

 Graphical user interface, text

Description automatically generated with medium confidence

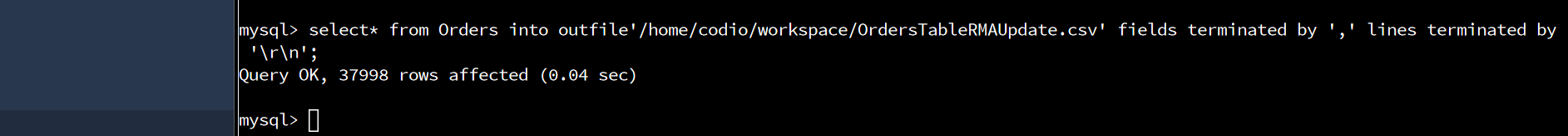
Code: rename table Customers to Collaborators;

Code: alter table Collaborators change CustomerID CollaboratorID int;

Code: alter table Orders change CustomerID CollaboratorID int;

In order to change all instances of ‘Customer’ to ‘Collaborator,’ a couple queries were run. First thing I did was run a rename query to change the name of the Customers table to Collaborators. I then ran an alter query on the newly renamed Collaborators table to change the CustomerID field into the CollaboratorID field (data type stayed as int). I then described the Collaborators table to ensure everything was entered correctly, and then I altered the Orders table to change the CustomerID field into the CollaboratorID field. Once again, I described the Orders table to ensure that it was entered correctly.

1. **Create an output file of the required query results.** Write an SQL statement to list the contents of the **Orders** table and send the output to a file that has a .csv extension.

 Graphical user interface, application

Description automatically generated

Code: select\* from Orders into outfile'/home/codio/workspace/OrdersTableRMAUpdate.csv' fields terminated by ',' lines terminated by '\r\n';

I ran a select\* query from the entire Orders table and sent into a CSV file via the outfile command. Since the previous activity had me sending it into a file called OrdersTable, I called this one OrdersTableRMAUpdate.