# DAD 220 Database Documentation Project One

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 generatedOpen Codio from Brightspace in Google Chrome web browser.
* Click on the terminal icon at the top of the explorer menu at the right of the interface.

MySQL

SHOW DATABASES;

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 generatedCREATE DATABASE QuantigrationUpdates;

SHOW DATABASES LIKE ‘QuantigrationUpdates’;

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:

Text

Description automatically generatedUSE QuantigrationUpdates

CREATE TABLE Customers (

CustomerID INT Primary Key ,

FirstName VARCHAR(25),

LastName VARCHAR(25),

Street VARCHAR(50),

City VARCHAR(50),

State VARCHAR(25),

ZipCode INT,

Telephone VARCHAR(15));

DESCRIBE Customers;

* 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:

Text

Description automatically generatedCREATE TABLE Orders (

OrderID INT PRIMARY KEY,

CustomerID INT,

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID),

SKU VARCHAR(20),

Description VARCHAR(50));

DESCRIBE Orders;

* 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 text, scoreboard, plaque

Description automatically generatedCREATE TABLE RMA (

RMAID INT PRIMARY KEY,

OrderID INT,

FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),

Step VARCHAR(50),

Status VARCHAR(15),

Reason VARCHAR(15));

DESCRIBE Orders;

## 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.

LOAD DATA INFILE ‘/home/codio/workspace/customers.csv’

INTO TABLE Customers

FIELDS TERMINATED BY ‘,’;

LOAD DATA INFILE ‘/home/codio/workspace/orders.csv’

INTO TABLE Orders

FIELDS TERMINATED BY ‘,’;

LOAD DATA INFILE ‘/home/codio/workspace/rma.csv’

INTO TABLE

FIELDS TERMINATED BY ‘,’;

Text

Description automatically generated

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.

SELECT COUNT(\*) FROM Customers

INNER JOIN Orders ON Customers.CustomerID=Orders.CustomerID

WHERE UPPER(Customers.City)=’FRAMINGHAM’ AND UPPER(Customers.State)=’MASSACHUSETTS’;

* + 1. How many records were returned?

505 records were returned.

Text

Description automatically generated

* + 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.

SELECT COUNT(\*) FROM Customers

WHERE UPPER(Customers.State)=’MASSACHUSETTS’;

* + 1. Record an answer to the following question: How many records were returned?

982 records were returned.

Text

Description automatically generated

* + 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 |

INSERT INTO Customers VALUES ('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');

Text

Description automatically generated

**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 |

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'),

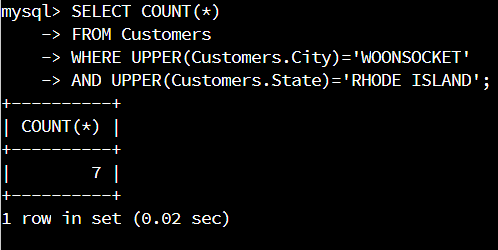
('1204307', '100006', 'ENT-24-10F', 'Enterprise Switch 10GigE SFP+ 24 Port'),

('1204308', '100007', 'ENT-48-10F', 'Enterprise Switch 10GigE SFP+ 48 port');

Graphical user interface, text

Description automatically generated

* + 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”?

SELECT COUNT(\*)

FROM Customers

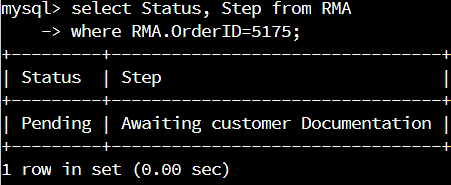
WHERE UPPER(Customers.City)=’WOONSOCKET’

AND UPPER(Customers.State)=’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?

Current Status is ‘Pending’

Current Step is ‘Awaiting customer Documentation’

SELECT Status, Step FROM RMA

WHERE RMA.OrderID=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?

Update Status is ‘Complete’

Updated Step is ‘Credit Customer Account’

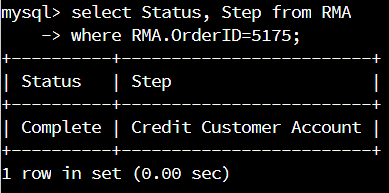
UPDATE RMA

SET Status=’Complete’,

Step=’Credit Customer Account’

WHERE RMA.OrderID=5175;

Text

Description automatically generated 

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

596 Records with reason of "Rejected” were deleted.

UPDATE RMA SET Reason = REPLACE(Reason,CHAR(10),’’);

UPDATE RMA SET Reason = REPLACE(Reason,CHAR(13),’’);

DELETE FROM RMA WHERE Reason=’Rejected’;

Text

Description automatically generated

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.”
2. A screenshot of a computer

   Description automatically generated with low confidenceRENAME TABLE Customers

TO Collaborators;

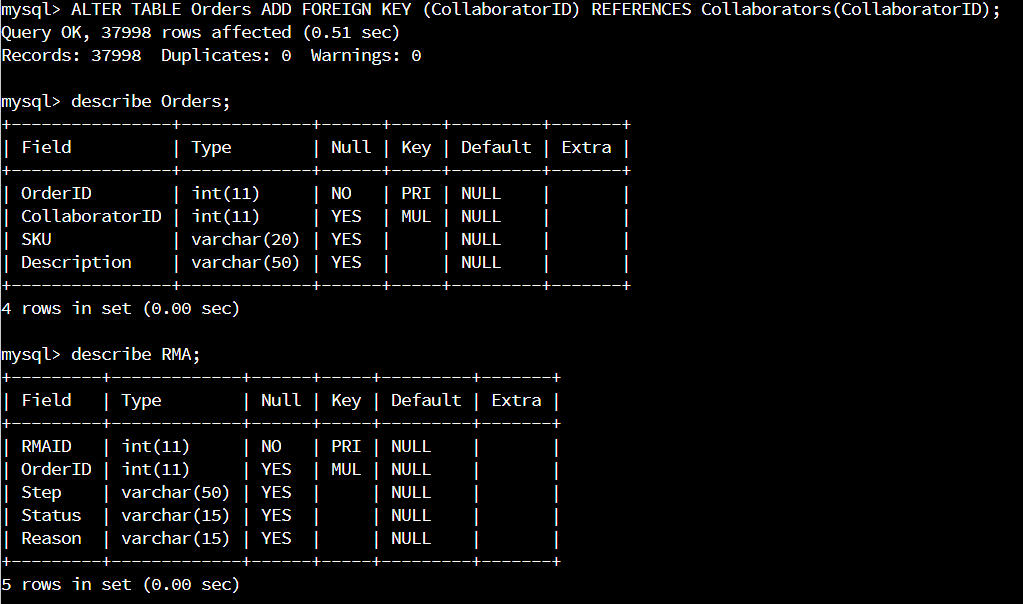
1. ALTER TABLE Orders DROP FOREIGN KEY Orders\_ibfk\_1; A picture containing text, orange, dark, screenshot

   Description automatically generated
2. ALTER TABLE Orders DROP CustomerID; A screenshot of a computer

   Description automatically generated with low confidence
3. ALTER TABLE Collaborators DROP CustomerID; A screenshot of a computer

   Description automatically generated with low confidence
4. ALTER TABLE Collaborators ADD COLUMN ‘CollaboratorID’ INT PRIMARY KEY AUTO\_INCREMENT FIRST; Graphical user interface, text

   Description automatically generated
5. ALTER TABLE Orders ADD COLUMN CollaboratorID INT AFTER OrderID;A picture containing text, orange, dark

   Description automatically generated
6. ALTER TABLE Orders ADD FOREIGN KEY (CollaboratorID) REFERENCES Collaborators(CollaboratorID);

This task required a lot of trial and error, but I was able to do what was described here. I believe the intent was to have students use CREATE VIEW, as we did in module 3 but the instructions indicated that we should update the existing table, not create a new one. Either way, After I completed the updates above, I used CREATE VIEW to create a Customers table with CustomerID as the primary key to demonstrate that the Foreign Keys do not update when a new view is created.

A screenshot of a computer

Description automatically generated with low confidenceCREATE VIEW Customers AS

SELECT CollaboratorID AS CustomerID,

FirstName,

LastName,

Street,

City,

State,

ZipCode,

Telephone

FROM Collaborators;

Text

Description automatically generated with low confidence

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.



Text

Description automatically generatedSELECT \* FROM Orders

INTO OUTFILE ‘/home/codio/workspace/OrdersProject1.csv’

FIELDS TERMINATED BY ‘,’;