Database Enhancement

Anthony Ochoa

Southern New Hampshire University

February 8, 2020

The third section of the final project entails enhancing a previous project that would display our expertise of databases. I chose to continue on my final project from the DAD-220 Introduction to SQL class. My original plan was to create an HTML interface to interact with the mysql database on the back end with the use of PHP scripts. Unfortunately, the time allotted for the enhancement would not provide a quality product. I then changed plans and decided to create a new database and import data from another program that I was working on in my other class DAT-220 Introduction to Data mining. This class has us using the JMP program to find hidden patterns and clusters in provided data in hope of using it to combat declining sales. While the overall project may sound simple, there were many speed bumps along the way.

**Creation of the Database in the MySQL DBMS**

A screenshot of a cell phone

Description automatically generated

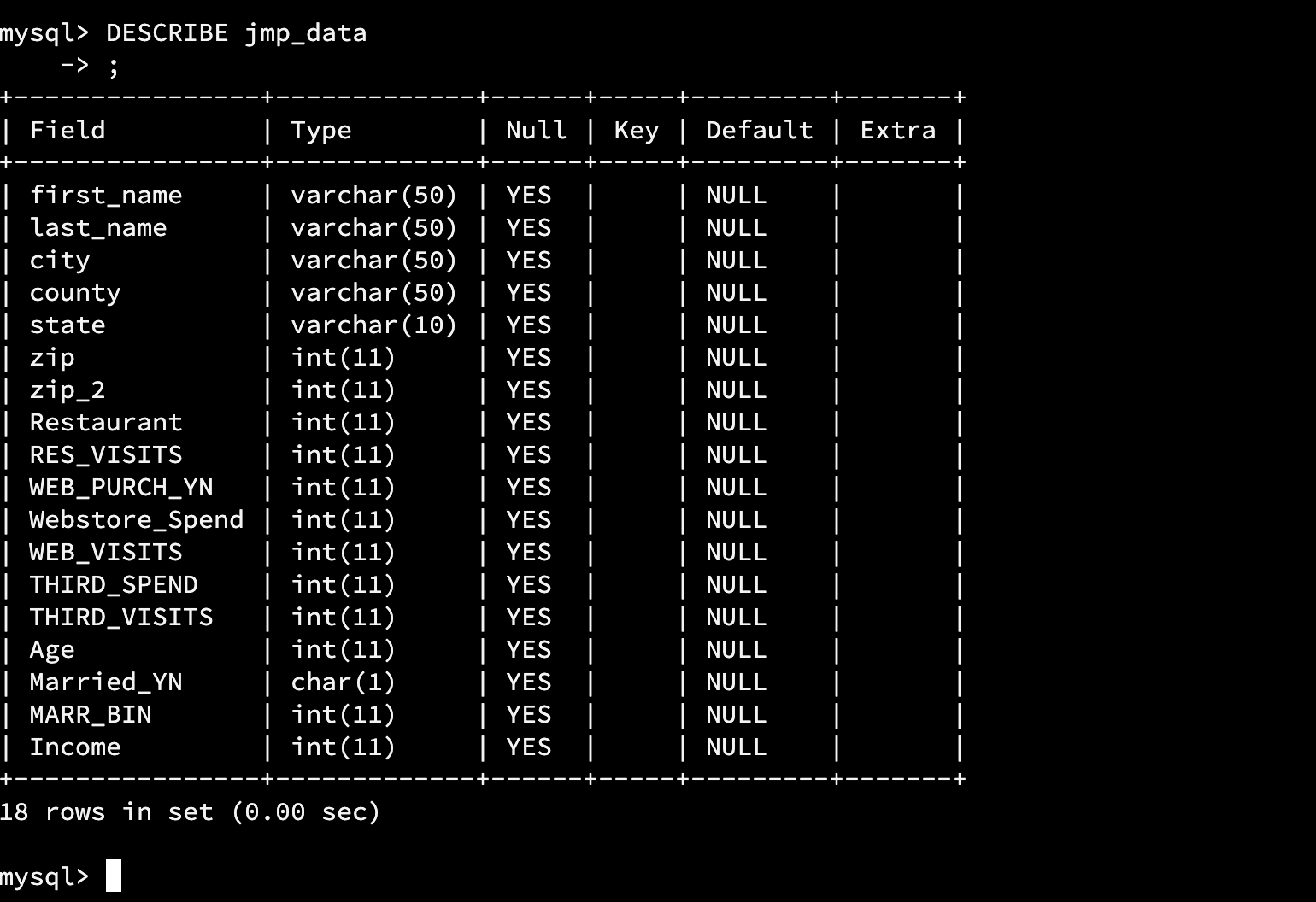
A screenshot of a cell phone

Description automatically generated

**Creation of Similar Table in New JMP Database**

**A screenshot of a cell phone

Description automatically generated**

****

**Exporting of Data from JMP**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**Issues Importing Data into New Database**

Once I had the data into the proper file folder, I needed to find a way to import this data in a way that would avoid entering the data one by one with INSERT commands. This is where I had the most trouble as it was tricky importing data into an existing table without any errors. My first attempt was using the LOAD DATA INFILE command with the corresponding file path for the file. This looked to be the best option but as I continued trying after errors I could not get the command to execute. My diagnosis of this problem was due to the fact that I was doing everything through the MySQL shell rather that the work bench. I also tried LOAD DATA LOCAL INFILE in hopes of parsing the text file and entering it into the corresponding jmp\_data table. This did not work and still errored out. My second attempt was to try using the Json import uitilty in mysql. This was also a dead end since I was using the MySQL shell inside of the codio playground. I did not find the issue to this problem, but I am thinking it is due to the codio site and certain restrictions to downloading new utilites. My third attempt was with mysqlimport command. This seemed to pretty straight forward but no matter what I entered it would error out. After some research I found that it needed to be run outside of the mysql server itself. Which in my opinion makes no sense? As you can see for the next screenshots the command was run in the original workplace with proper logins to the sql server.

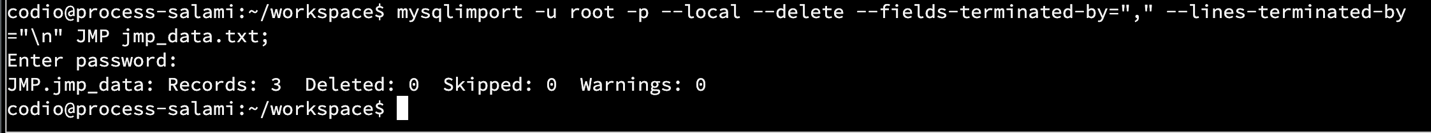
A close up of a screen

Description automatically generated

As you can see the amount of records provided was 10042 and this is incorrect as the amount of records was 500 in the original JMP program. When I queried this data, it proved a mess of data with many null values. I then began to research the options available to use with the import function that reads the file and then separates it accordingly. This again stopped me dead in my tracks as I wasn’t able to get the data to import correctly. I tried a combination of different options including –fields-terminated-by \*\*\* , –fields-enclosed-by\*\*\* and lines-terminated-by\*\*\*. After much frustration I decided to alter the data in the text to allow for an easier import. As you can see below the altered data and the altered import command to import that data.

A screenshot of a cell phone

Description automatically generated



**Simple Queries with Imported Data**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**