Murach 3e Chapter 11 Coding Assignment Instructions

Exercises to be Completed
Please complete Exercises 1 – 5. You will find those exercises on the attached sheet(s).

General Instructions
My expectations for your work on coding assignment exercises will grow as we progress through the course. In addition to applying any new coding techniques that have been covered in the current chapter, I will be expecting you to follow all of the good practices that we have adopted in the preceding weeks. Here is a quick summary of good practices that we have covered so far:

- Begin each script file that accesses the database with a USE statement (e.g., USE my_guitar_shop;).
- Use the *beautify* feature of the MySQL Workbench to *pretty-print* your code.
- End each statement in your script with a semicolon.
- Use the SQL features requested in the exercise description and/or covered in the chapter.
- Always include an ORDER BY in SELECT statements unless directed otherwise. If the exercise instructions ask for a particular order, then use that. Otherwise, choose any reasonable order.
- In SELECT statements that use JOIN, always use the explicit (ANSI) JOIN syntax implemented in the FROM clause. Do NOT use the implicit JOIN syntax implemented using the WHERE clause.
- When testing SELECT statements that use summary functions, always test with the ONLY_FULL_GROUP_BY setting set to ON.
- Do NOT include extra or unnecessary code in the script.

Tools
Use MySQL Workbench to create and test all scripts.

Submission Method
Use the following process to submit your work for this assignment:

- Locate the properly named directory associated with your assignment in the file system (see *File and Directory Naming,* below).
- Compress that directory into a single .ZIP file using a utility program. NOTE: Only one file may be submitted. File types other than .ZIP will not be accepted and will receive a grade of zero.
- Submit the properly named zip file to the submission activity for this assignment.

File and Directory Naming
<mark>Please note that file and directory names must be in all lower case. Deductions will be made for submissions that do not follow this standard.</mark>

Please use the following naming scheme for the directory that holds your scripts:

`surname_givenname_mgs_chap_11`

If this were my own project, I would name my PyCharm project as follows:

`trainor_kevin_mgs_chap_11`

A separate solution script file must be submitted for each exercise. Solution scripts must be named using the following form: ex_xx_yy.sql (where xx is the two-digit chapter number [04] and yy is the two-digit exercise number [01]). So, an example of a properly formed solution script file example would be:

`ex_11_01.sql`

Use a zip utility to create one zip file that contain the PyCharm project directory. The zip file should be named according to the following scheme:

`surname_givenname_mgs_chap_11.zip`

If this were my own project, I would name the zip file as follows:

`trainor_kevin_mgs_chap_11.zip`

Due By
Please submit this assignment by the date and time shown in the Weekly Schedule.
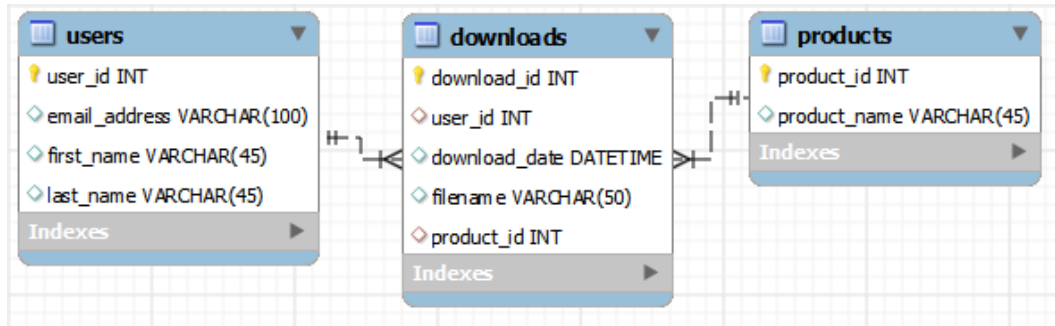
Last Revised
2021-03-21

Please see the exercises on the attached sheets

Chapter 11

How to create databases, tables, and indexes

Exercises

1. Write a script that adds an index to the my_guitar_shop database for the zip code field in the Addresses table.

2. Write a script that implements the following design in a database named my_web_db:

   In the Downloads table, the user_id and product_id columns are the foreign keys.

   Include a statement to drop the database if it already exists.

   Include statements to create and select the database.

   Include any indexes that you think are necessary.

   Specify the utf8mb4 character set for all tables.

   Specify the InnoDB storage engine for all tables.

3. Write a script that adds rows to the database that you created in exercise 2.

   Add two rows to the Users and Products tables.

   Add three rows to the Downloads table: one row for user 1 and product 2; one row for user 2 and product 1; and one row for user 2 and product 2. Use the NOW function to insert the current date and time into the download_date column.

   Write a SELECT statement that joins the three tables and retrieves the data from these tables like this:

email_address	first_name	last_name	download_date	filename	product_name
johnsmith@gmail.com	John	Smith	2019-03-14 11:41:17	pedals_are_falling.mp3	Local Music Vol 2
janedoe@yahoo.com	Jane	Doe	2019-03-14 11:41:17	turn_signal.mp3	Local Music Vol 1
janedoe@yahoo.com	Jane	Doe	2019-03-14 11:41:17	one_horse_town.mp3	Local Music Vol 2

   Sort the result set by the email address in descending sequence and the product name in ascending sequence.

4. Write an ALTER TABLE statement that adds two new columns to the Products table created in exercise 2.

    Add one column for product price that provides for three digits to the left of the decimal point and two to the right. This column should have a default value of 9.99.

    Add one column for the date and time that the product was added to the database.

5. Write an ALTER TABLE statement that modifies the Users table created in exercise 2 so the first_name column cannot store NULL values and can store a maximum of 20 characters.

    Code an UPDATE statement that attempts to insert a NULL value into this column. It should fail due to the NOT NULL constraint.

    Code another UPDATE statement that attempts to insert a first name that's longer than 20 characters. It should fail due to the length of the column.