Homework #01

Complete By: **Thursday, January 19th @ 11:59pm**
(assignment will be accepted for full credit until Tuesday, 1/24/23 at 11:59pm)

Policy: **Individual work only**
Work *not* accepted after 1/24/23

Assignment: **SQL queries**

Submission: **submit electronically on Gradescope**

Overview

In HW #01 you're going to read about SQL, and then write SQL queries to retrieve data from the SQLite CTA L daily ridership database. You can work on replit.com, or work locally by installing SQLite. Either way, your queries must be saved in separate .sql files and submitted to Gradescope for grading.

Learning SQL: reading and sample exercises

Let's start by reading and learning some SQL basics. Here's a good site that features integrated SQL databases for practice:

http://sqlzoo.net/

Work through the following exercises on http://sqlzoo.net/. The associated quizzes are recommended but optional:

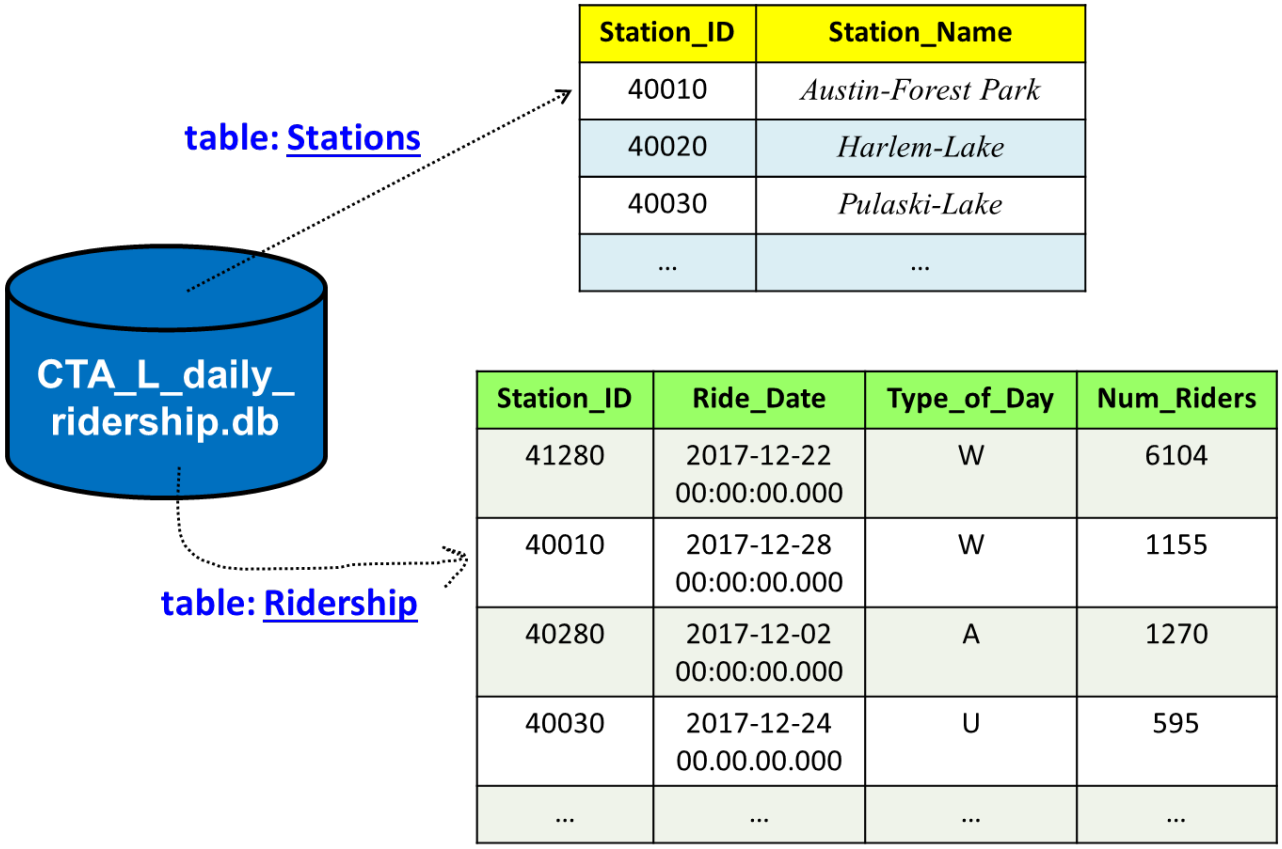
SELECT Basics:	exercises 1-3
SELECT from World:	exercises 1-7
SELECT from Nobel:	exercises 1-8
SELECT in SELECT:	exercises 1-2
SUM and COUNT:	exercises 1-5

In general, here's a good reference site for SQL: http://www.w3schools.com/sql/default.asp

Executing SQL using SQLite

SQLite is a standalone, easy to install database environment for Linux, Mac, and Windows. We'll be

working with the SQLite version of the CTA L daily ridership database; you are free to work on replit.com, or install SQLite and the database on your local computer. For local installations, we recommend SQLiteStudio, which is available from https://sqlitestudio.pl/. Here's a snapshot of the database:



Station_ID	Station_Name
40010	*Austin-Forest Park*
40020	*Harlem-Lake*
40030	*Pulaski-Lake*
...	...

Station_ID	Ride_Date	Type_of_Day	Num_Riders
41280	2017-12-22 00:00:00.000	W	6104
40010	2017-12-28 00:00:00.000	W	1155
40280	2017-12-02 00:00:00.000	A	1270
40030	2017-12-24 00.00.00.000	U	595
...	...	...	...

If you plan to work on replit.com

If you haven't already, join the team for this class: "cs-341-spring-2023". Instructions are posted on BB. Under teams, look for the project named "Homework 01" and click "Start Project" or "Continue Working". You will need to create a sequence of .sql files, one per exercise: hw01-01.sql, hw01-02.sql, etc. You'll submit these files to Gradescope for grading. [Note: it appears that the "Run" button in replit.com only runs the contents of the "main.sql" file. One option is to develop in "main.sql", and then paste your code into the appropriate hw .sql file. Option two is to develop in the appropriate hw .sql file, such as "hw01-01.sql", and then in the console window execute via ".read hw01-01.sql".]

If you plan to work locally with SQLiteStudio

First you need to install SQLiteStudio, and then you need to download a copy of the CTA Ridership database. You will need to create a sequence of .sql files, one per exercise: hw01-01.sql, hw01-02.sql, etc. You'll submit these files to Gradescope for grading.

Exercises

First, check your email for an invitation from ***Gradescope***. This is the mechanism we'll be using for collecting and grading HW and project submissions. If you did not receive an invite, post privately to Piazza and we'll resend the invite.

What follows are a series of short SQL programming exercises, which you can submit at any time to Gradescope. If you are new to Gradescope, keep in mind that we grade your **LAST SUBMISSION** --- this implies your last submission should contain answers to all exercises. You can submit a .zip of your .sql files, or submit them as a set, whichever is easier; you have unlimited submissions. Program comments are not necessary.

Exercise #01: Total number of L stations?

In the file "hw01-01.sql", write a single SQL select query to retrieve the total # of L stations in the Stations table.

[Hint: use count() function. Answer: 147]

Exercise #02: L stations in alphabetical order

In the file "hw01-02.sql", write a single SQL select query to retrieve the name of every L station, in alphabetical order.

Exercise #03: Which L station has the station ID 41240?

In the file "hw01-03.sql", write a single SQL select query to retrieve the name of the L station with station ID 41240.

[Answer: Addison-O'Hare]

Exercise #04: Total # of riders that passed through station 41300 in the year 2015?

In the file "hw01-04.sql", write a single SQL select query to retrieve the total # of riders that passed through the turnstiles of station 41300 during the year 2015.

[Hint: use sum() function, the date() function, and where clause with date range. Answer: 1821379]

Exercise #05: Midway Airport ridership 2019 vs. 2020

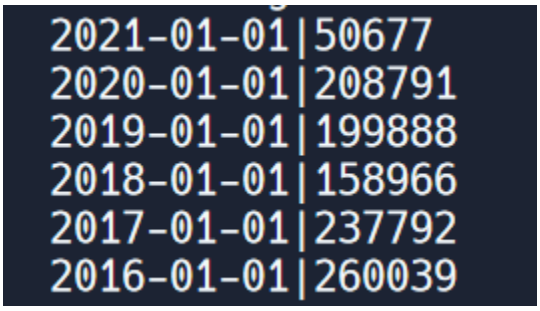
In the file "hw01-05.sql", write two SQL select queries, one to retrieve the total # of riders through Midway Airport in 2019 and another to retrieve the total # of riders through Midway Airport in 2020. Instead of hard-coding the station ID for Midway Airport into the queries, each select query must use the "*Select in Select*" approach discussed in the reading.

[Answer: 2477340 followed by 861649]

Exercise #06: Number of riders on January 1st for years 2016 through 2021, inclusive

In the file "hw01-06.sql", write a single SQL select query to retrieve the date and total number of riders for each January 1st in years 2016 through 2021, inclusive. The results should be in descending order by the date.

[Hint: use sum() function, and you'll need to **group by Ride_Date** to compute the correct results, which are



Grading and Electronic Submission

A few days before the assignment is due, login to Gradescope.com and look for the assignment "Homework 01". Submit your individual query files "hw01-01.sql", "hw01-02.sql", etc., under "HW01". You can also zip the six query files together and submit the single zip file as your submission. You have unlimited submissions; keep in mind we grade your LAST SUBMISSION unless you select an earlier submission for grading.

On replit.com you can click the vertical "..." to download your files as a .zip, but note this will include the database file as well. This file is large (86MB), which will make uploading to gradescope slow. To speed things up, download as .zip, open the .zip, remove the .db file from the .zip, and then upload the much smaller .zip to gradescope.

Academic Conduct Policy

Late work is not accepted for this assignment. All work is to be done individually — group work is not allowed. While we encourage you to talk to your peers and learn from them, this interaction must be superficial with regards to all work submitted for grading. This means you *cannot* work in teams, you cannot work side-by-side, you cannot submit someone else's work (partial or complete) as your own. The University's policy is available here:

https://dos.uic.edu/conductforstudents.shtml

In particular, note that you are guilty of academic dishonesty if you **extend or receive any kind of unauthorized assistance**. Absolutely no transfer of program code between students is permitted (paper or electronic), and you may not solicit code from family, friends, or online forums (e.g. you cannot download answers from Chegg). Other examples of academic dishonesty include emailing your program to another student, sharing your screen so that another student may copy your work, copying-pasting code from the internet, working together in a group, and allowing a tutor, TA, or another individual to write an answer for you. Academic dishonesty is unacceptable, and penalties range from a letter grade drop to expulsion from the university; cases are handled via the official student conduct process described at https://dos.uic.edu/conductforstudents.shtml .