

SQL In-class

Olympics

We will be using SQLite, which provides a standards-compliant SQL implementation. In reality, there are slight variations between the SQL dialects of different vendors (PostgreSQL, MySQL, SQLite, Oracle, Microsoft, etc.) – especially with respect to built-in functions. The SQL tutorial at <http://sqlzoo.net/>, provides a good introduction to the basic features of SQL. After following this tutorial, you should be able to answer most of the problems in this problem set. A more detailed tutorial is available at <https://www.sqlitetutorial.net/>.

The SQLite SELECT documentation at https://sqlite.org/lang_select.html will be helpful to you, and you can access all the other SQLite documentation on that site as well.

Download and install SQLiteStudio

<https://sqlitestudio.pl/>

Feel free to review the in-class notes as a refresher. Here is also a good resource

<https://www.w3schools.com/sql/>

Download the data

<https://public.gettysburg.edu/~jpuckett/ds325/data/olympics.db>

Dataset

For this assignment we use a dataset on modern Olympic Games adapted from <https://www.kaggle.com/heesoo37/120-years-of-olympic-history-athletes-and-results>. This dataset includes details about all athletes who competed in an Olympic event from Athens 1896 to Rio 2016. Everything you need to understand the dataset is contained in your SQLite database.

The database tables include:

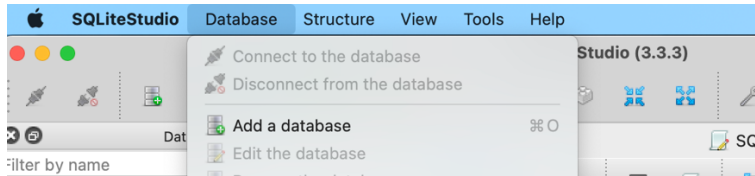
athletes: contains the id (unique to each athlete), name, sex, age when they competed in the Olympics for the first time, height (in centimeters), and weight (in kilograms) for each athlete who has participated in the Olympics.

host cities: contains the games (year and season, eg. “2016 Summer”), year, season (“Summer” or “Winter”), and city (host city) for each Olympic Games. **regions:** contains the noc (National Olympic Committee 3-letter code), region (country name), and notes for each country that has participated in the Olympics.

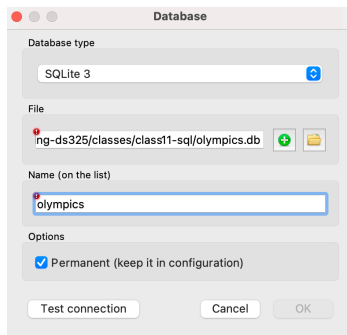
athlete events: contains id (athlete’s id), team (team name), noc (code of the country that the athlete represented), games (year and season of Olympics), sport, event, and medal (“Gold”, “Silver”, “Bronze”, or “NA”). Each row corresponds to an individual athlete competing in an Olympic event. Note that it is possible for any value in a non-primary key column to be NULL.

Exercise 1: Getting SQLiteStudio and Loading the Data

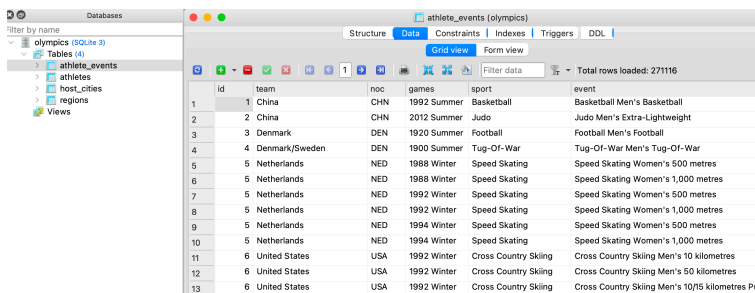
- Open SQLiteStudio and add the database



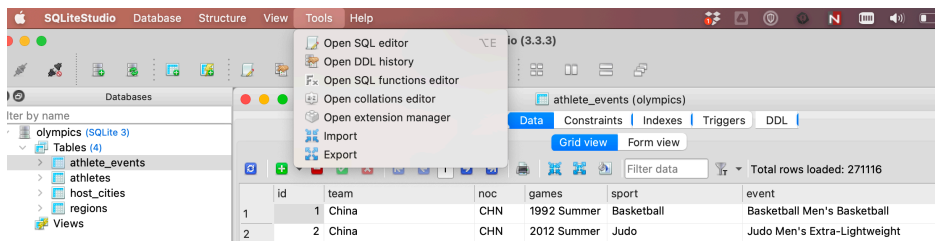
- Select the path to the **olympics.db** file you downloaded. Select OK.



- Spend a little time viewing the database. How is it structured? What are the fields?
- Query tables with a simple 'Select * from table_name limit 10' to see column names and data available
- Look for common column between two tables, ideally a unique identifier (id)



- Open a SQL editor.



Exercise: Queries

Now you're ready to do the really cool stuff!

Q1. How many sports are there?

Q2. How many teams are there?

Q3. Find all athletes who participated in an Olympic Games before they were 12. Print the id, name, and age of the athlete in the increasing order of id.

Q4. Who are the top 5 oldest athletes to ever compete in the Games?

Q5. Find the number of athletes who participated in each Olympic Games. Print the games, host city, number of athletes in the increasing order of games.

Q6. Find which women's basketball team won gold at the last 10 games.

Q7. Find all athletes who won gold medals in Olympic Games before they were 20. Print the id, name, age of the athlete and the event. Sort in the decreasing order of id.

Q8. Print the medals table for "2016 Summer" Olympic Games. Print rows for countries that participated in the **2010 Winter games** and display as team, number_of_gold, number_of_silver, number_of_bronze. Order by number_of_gold descending.

Q9. Find the athlete(s) with the most number of medals. Print the name, number of gold medals, and the team for each athlete.