Querying your SQL database

ETHAN YOO, DATA SCIENCE GRADUATE SPECIALIST

FEBRUARY 21, 2023

ETHAN.YOO@RUTGERS.EDU

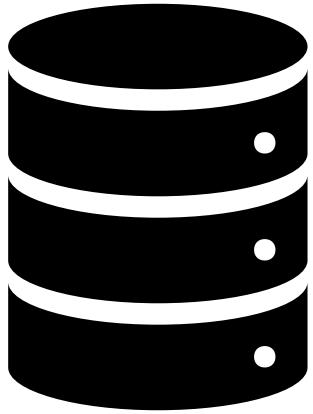
Outline

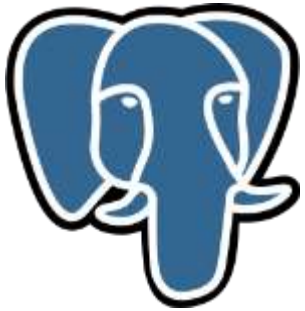
- Introduction/review of relational databases and SQLite
- Filtering for specific fields (columns) and records (rows)
- Merging related tables
- Demonstration/walkthrough
- Additional resources

Terminology

- "**SQL** (Structured Query Language) is a descriptive computer language designed for updating, retrieving, and calculating data in table-based databases" (MDN Web Docs)
- Table
- Database engine
- Create, Read, Update, and Delete (CRUD)
- Database management system (DBMS)

Introduction to relational databases





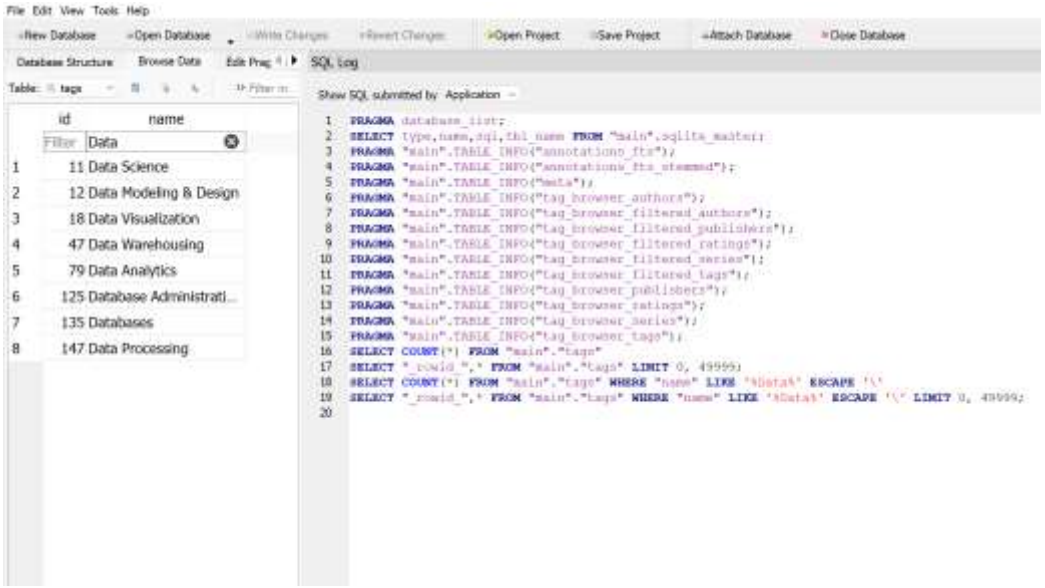
SQLite

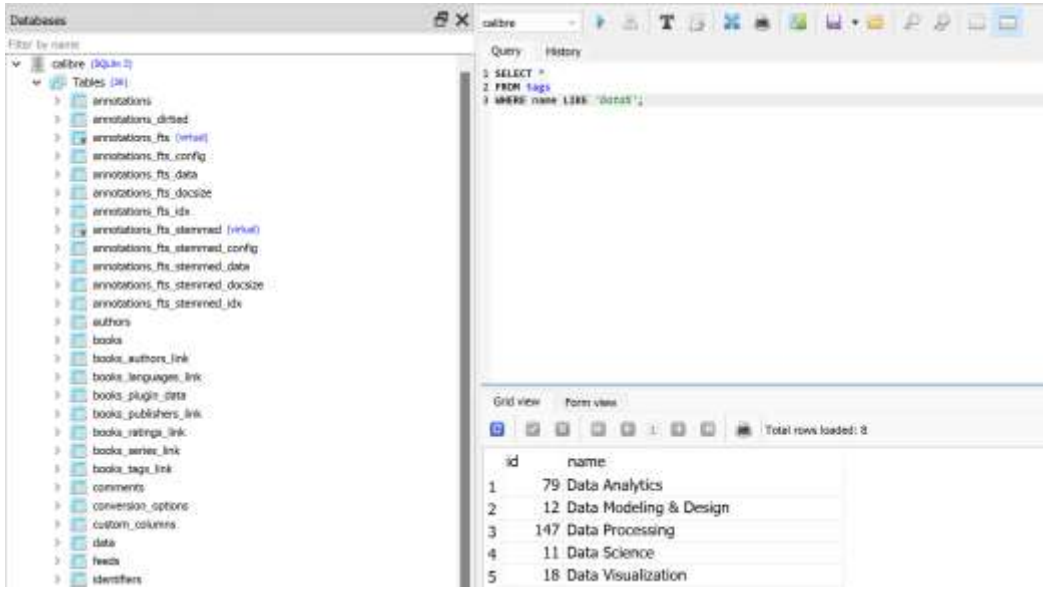
- Embedded database and database engine
- Public domain
- Digital format recommended by the Library of Congress for datasets
- Major known users from among millions of applications
  - **Apple**: Many native applications across devices
  - **Google**: The Android operating system and Chrome web browser
  - **Microsoft** Windows 10 and other Microsoft products
  - **Mozilla** Firefox (web browser) and Thunderbird (email client)
  - All **Python** distributions since Python 2.5

Database management tools

DB Browser for SQLite

SQLiteStudio





Specifying columns

SELECT columnName

FROM table_name;

You can specify multiple columns by separating each column's name with a comma

Filtering rows

```
SELECT *
FROM table_name
WHERE condition;
```

Boolean or logical operators

- **TRUE or FALSE**
- **AND**
- **OR**
- **NOT**

Patterns

- % (percentage sign)
- _ (underscore)
- ESCAPE '%'

Built-in aggregate functions

- AVG()
- COUNT()
- MIN()
- MAX()
- SUM()

Filtering groups

```
SELECT *
FROM table_name
GROUP BY columnName;
```

Filtering groups

```
SELECT *
FROM table_name
GROUP BY columnName
HAVING condition;
```

Merging tables with JOIN operators

- INNER JOIN or JOIN
- LEFT JOIN
- RIGHT JOIN
- FULL JOIN

```
SELECT table_1.column1, table_1.column2, table_2.column2
FROM table_1
INNER JOIN table2
ON table_1.column1 = table_2.column2;
```

Additional Resources

- **Tutorials**
  - **Data Carpentry (uses DB Browser for SQLite)**
  - **Khan Academy**
  - **SQLBolt (browser-based)**
  - **SQL Zoo (browser-based)**
- **Documentation**
  - **MySQL 8.0**
  - **PostgreSQL 15.2**
  - **SQLite (e.g., SELECT)**