

DBMS Lab - Getting Started with SQLite

Welcome to the DBMS Lab! This guide will help you set up your environment and start working with SQLite databases using the Chinook sample database.

Table of Contents

- [Prerequisites](#)
- [Setup Instructions](#)
- [Working with Chinook Database](#)
- [Basic SQLite Commands](#)
- [Sample Queries](#)
- [Additional Resources](#)

Prerequisites

1. **VS Code** - Download from code.visualstudio.com
2. **SQLite3** - Usually pre-installed on macOS/Linux. For Windows, download from [sqlite.org](https://www.sqlite.org)
3. **DB Browser for SQLite** - Download from sqlitebrowser.org/dl

Setup Instructions

1. Install SQLite Viewer Extension in VS Code

1. Open VS Code
2. Go to Extensions (⌘+Shift+X on Mac, Ctrl+Shift+X on Windows/Linux)
3. Search for "**SQLite Viewer**" or "**SQLite**"
4. Install the extension by alexcvzz or qwtel
5. Restart VS Code if needed

2. Install DB Browser for SQLite

1. Visit sqlitebrowser.org/dl
2. Download the installer for your operating system:
 - **macOS:** Download the .dmg file
 - **Windows:** Download the .msi installer
 - **Linux:** Use package manager or download AppImage
3. Install and launch the application

3. Set Up Chinook Database

The **Chinook database** is a sample database representing a digital media store with tables for artists, albums, tracks, customers, invoices, and more.

Database already included: `chinook.db` is in this repository

To download fresh copy:

- Visit sqlitetutorial.net/sqlite-sample-database
- Download the chinook.db file
- Place it in your project directory

Working with Chinook Database

Method 1: Using SQLite Viewer in VS Code

1. Open the `chinook.db` file in VS Code
2. The SQLite Viewer extension will display the database structure
3. Click on tables to view their contents
4. Right-click on a table to run queries

Method 2: Using DB Browser for SQLite

1. Open DB Browser for SQLite application
2. Click "**Open Database**"
3. Navigate to and select `chinook.db`
4. Explore tabs:
 - **Database Structure:** View all tables and schemas
 - **Browse Data:** See table contents
 - **Execute SQL:** Run custom queries

Method 3: Using Terminal/Command Line

Open terminal in VS Code and run:

```
sqlite3 chinook.db
```

Basic SQLite Commands

Inside SQLite Shell

```
-- List all tables
.tables

-- Show table schema
.schema table_name

-- Show all table schemas
.schema

-- Enable better formatting
```

```
.mode column
.headers on

-- Execute a query
SELECT * FROM artists LIMIT 10;

-- Exit SQLite
.quit
```

Run Queries from Terminal

```
# Run a single query
sqlite3 chinook.db "SELECT * FROM artists LIMIT 5;"

# Execute SQL file
sqlite3 chinook.db < query.sql

# Save output to file
sqlite3 chinook.db -column -header < query.sql > results.txt

# Better formatted output
sqlite3 chinook.db -column -header "SELECT * FROM albums LIMIT 5;"
```

Sample Queries

Explore Database Structure

```
-- List all tables
SELECT name FROM sqlite_master WHERE type='table';

-- Count rows in each table
SELECT 'artists' as table_name, COUNT(*) as row_count FROM artists
UNION ALL
SELECT 'albums', COUNT(*) FROM albums
UNION ALL
SELECT 'tracks', COUNT(*) FROM tracks;
```

Basic SELECT Queries

```
-- View all artists
SELECT * FROM artists LIMIT 10;

-- Find customers from USA
SELECT FirstName, LastName, Email FROM customers
WHERE Country = 'USA';

-- Get unique countries
SELECT DISTINCT Country FROM customers ORDER BY Country;
```

JOIN Queries

```
-- Artists with their albums
SELECT artists.Name as Artist, albums.Title as Album
FROM albums
JOIN artists ON albums.ArtistId = artists.ArtistId
ORDER BY artists.Name;

-- Tracks with album and artist info
SELECT
    tracks.Name as Track,
    albums.Title as Album,
    artists.Name as Artist,
    tracks.Milliseconds / 60000 as Minutes
FROM tracks
JOIN albums ON tracks.AlbumId = albums.AlbumId
JOIN artists ON albums.ArtistId = artists.ArtistId
LIMIT 10;
```

Aggregate Queries

```
-- Count tracks by genre
SELECT genres.Name as Genre, COUNT(*) as TrackCount
FROM tracks
JOIN genres ON tracks.GenreId = genres.GenreId
GROUP BY genres.Name
ORDER BY TrackCount DESC;

-- Total sales by country
SELECT
    customers.Country,
    COUNT(DISTINCT customers.CustomerId) as Customers,
    ROUND(SUM(invoices.Total), 2) as TotalRevenue
FROM customers
JOIN invoices ON customers.CustomerId = invoices.CustomerId
GROUP BY customers.Country
ORDER BY TotalRevenue DESC;
```

Database Schema Overview

The Chinook database contains the following tables:

- **albums** - Album information with links to artists
- **artists** - Artist names and IDs
- **customers** - Customer information
- **employees** - Employee records
- **genres** - Music genres
- **invoices** - Purchase invoices

- **invoice_items** - Line items for each invoice
- **media_types** - Types of media formats
- **playlists** - Playlist information
- **playlist_track** - Tracks in each playlist
- **tracks** - Individual track details

Pre-made SQL Query Files

This repository includes several SQL query files you can execute:

1. **query_artists_albums.sql** - Artists ranked by album count
2. **sales_report.sql** - Sales analysis by country
3. **popular_tracks.sql** - Most purchased tracks

Run them with:

```
sqlite3 chinook.db -column -header < query_artists_albums.sql
```

Tips for Students

1. **Start Simple:** Begin with basic SELECT queries before moving to JOINs
2. **Use LIMIT:** Always use LIMIT when exploring large tables
3. **Format Your Code:** Write readable SQL with proper indentation
4. **Experiment:** Try modifying existing queries to learn
5. **Save Your Queries:** Keep useful queries in .sql files for reuse
6. **Check Data Types:** Use `.schema table_name` to understand table structure
7. **Use Comments:** Add `--` comments to explain your SQL logic

Common Mistakes to Avoid

- Forgetting to use quotes around string values: `WHERE Country = 'USA'`
- Missing GROUP BY when using aggregate functions
- Not using table aliases in complex JOINs
- Forgetting semicolons at the end of statements (in files)

Additional Resources

- **Chinook Database:** sqlitetutorial.net/sqlite-sample-database
- **DB Browser:** sqlitebrowser.org/dl
- **SQLite Tutorial:** sqlitetutorial.net
- **SQLite Documentation:** sqlite.org/docs.html
- **SQL Practice:** sqlzoo.net

Quick Reference Card

Command	Description
<code>.tables</code>	List all tables
<code>.schema TABLE</code>	Show table structure
<code>.mode column</code>	Format output in columns
<code>.headers on</code>	Show column headers
<code>.quit</code>	Exit SQLite shell
<code>LIMIT n</code>	Restrict results to n rows
<code>ORDER BY col</code>	Sort results
<code>WHERE condition</code>	Filter rows
<code>GROUP BY col</code>	Aggregate by column
<code>JOIN</code>	Combine tables

Happy querying! 

For questions or issues, please refer to the course materials or ask your instructor.