Momentum Workshop – Database 101

The purpose of this lab is to give students hands-on experience working with a relational database. There are 5 parts to the data portion of lab:

1. Install SQLite and SQLiteStudio.
2. Create database Northwind.
3. Create database tables and load data.
4. Run SQL command to retrieve data from the database.
5. Power BI demo – examine the Northwind data model and visualize the data

# Technology

A logo of a software company

Description automatically generated

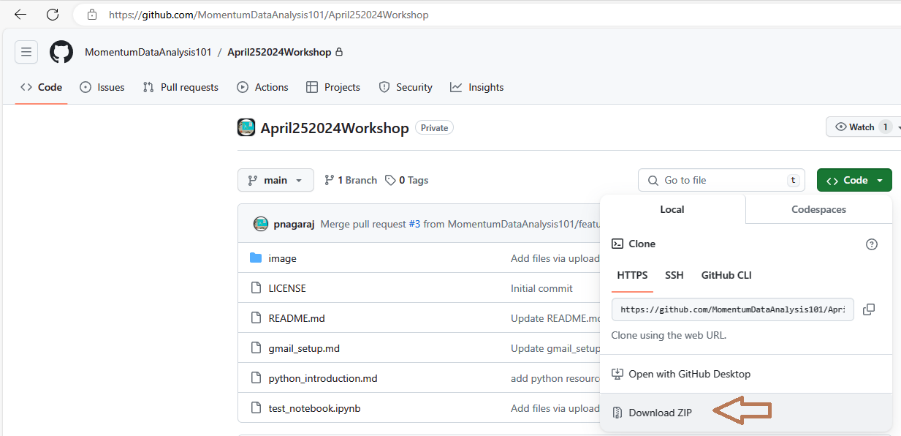
[**SQLite**](https://www.sqlite.org/index.html) is a lightweight, file based, relational database management system (RDBMS. It is a self-contained, serverless, zero-configuration, transactional SQL database; widely used in embedded systems, mobile applications, desktop software, and small to medium-scale websites.

[**SQLiteStudio**](https://sqlitestudio.pl/) is a cross-platform, open-source SQLite database manager. It provides a user-friendly interface for managing SQLite databases, executing SQL queries, designing database schemas, and performing various database-related tasks.

The **Northwind** database is a sample database used for demonstrating relational database concepts and practices. It was originally created by Microsoft for educational and demonstration purposes. The scripts provided by this lab have been adapted for SQLite. The Northwind database models a fictional company called "Northwind Traders," which sells various products to customers.

# Getting Started

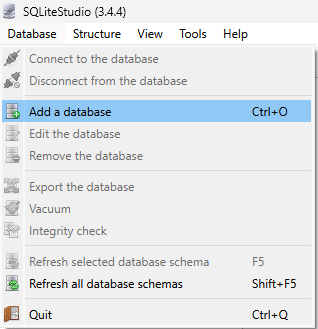
1. Install SQLite:
   1. Download precompiled binaries for window from [SQLite download page](https://www.sqlite.org/download.html).
   2. Save the zip file to the Downloads folder.
   3. Extract all content and save it to folder C:\Users\<userid>\Downloads\sqlite-tools-win-x64-3450100
2. Install SQLite Studio:
   1. Download the install program from the [SQLiteStudio web site](https://sqlitestudio.pl/).
   2. Run exe file SQLiteStudio-3.4.4-windows-x64-installer.exe. Note: You may need administrator right to install the program.
   3. Follow the prompts and accept all default values. The program will be installed in directory C:\Program Files\SQLiteStudio.
3. Download the lab materials from [GitHub](https://github.com/MomentumDataAnalysis101/April252024Workshop.git).



1. Unzipped the contents into a local folder.

# Create Database Northwind

1. Open SQLiteStudio.
2. Select option [Add a database] under the Database menu:



1. Fill in the form as follow:
   1. Database type: SQLite 3
   2. File: select a file location to store the database. Use the folder icon on the side to browse for the location.
   3. Name: enter Northwind as the database name.
   4. Click the OK button to create the database.

A screenshot of a computer

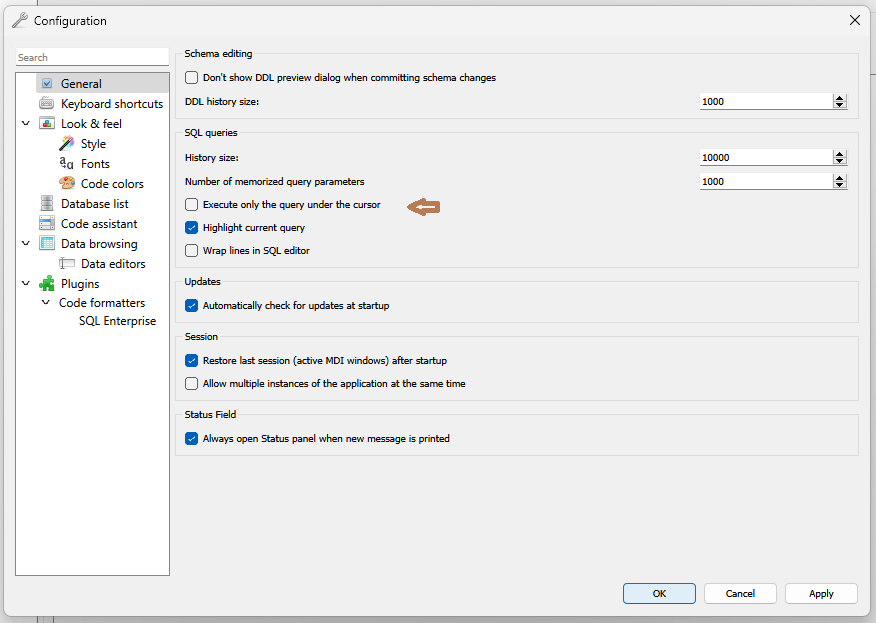
Description automatically generated

1. Database Northwind is now created and listed on the left panel.
2. To streamline the following steps, update the configuration by selecting option [Open configuration dialog] under the Tools menu or press F10.

A screenshot of a computer

Description automatically generated

1. Uncheck option [Execute only the query under the cursor].



1. Click the Apply button at the bottom right to save the change.
2. Click the OK button to exit the dialog box.

# Create Tables and Load Sample Data

1. Open a SQL editor by selecting option [Open SQL editor] under the Tools menu.

A screenshot of a computer

Description automatically generated

1. Right click on the query pane and select option [Load SQL from file]

A screenshot of a computer

Description automatically generated

1. Select and execute all files under folder Northwind Database in ascending order based on their name.
2. The scripts will create 8 tables and load the sample data.
3. The Northwind database is ready for query once all the scripts are executed.

# Explore

You should now see all the eight tables in the Northwind data base as shown in the diagram below:

A screenshot of a computer

Description automatically generated

In this section, we will explore the attributes of a database table.

1. Expand the Customers table. You should see the customer table has 11 columns and 4 indexes.

A screenshot of a computer

Description automatically generated

1. Double click the Customers table to edit the table.

A screenshot of a computer

Description automatically generated

1. The edit window has 6 tabs:
   1. Structure – list the columns and their attributes.
   2. Data – allows you to browse the contents of the table.
   3. Constraints – data validation rules and limitations
   4. Indexes – data structure used to improve data retrieval performance.
   5. Trigger – actions to perform when data is modified: insert, update, delete.
   6. DDL – SQL constructs used to create the table.
2. Check out other tables in the

# Query using SQL

SELECT statement is the most used SQL statement. It retrieves data from one or more tables or views. The full syntax is complex, but the main clauses can be summarized as follows:

SELECT select\_list [ INTO new\_table ]

[ FROM table\_source ] [ WHERE search\_condition ]

[ GROUP BY group\_by\_expression ]

[ HAVING search\_condition ]

[ ORDER BY order\_expression [ ASC | DESC ]]

1. Open a SQL editor in SQLite Studio by pressing Alt+E.
2. Copy the following statements into the SQL edition and pressing F9 to execute the statement:

* List all orders:

SELECT OrderID,

CustomerID,

EmployeeID,

OrderDate,

RequiredDate,

ShippedDate,

ShipVia,

Freight,

ShipName,

ShipAddress,

ShipCity,

ShipRegion,

ShipPostalCode,

ShipCountry

FROM Orders;

* Count numbers of orders by year and month:

SELECT strftime('%Y%m',OrderDate) as "Period"

, count(1) as "# of Orders"

FROM Orders

GROUP BY strftime('%Y%m',OrderDate);

* Count numbers of orders by year and month and customers:

SELECT c.CompanyName

, strftime('%Y%m',OrderDate) as "Period"

, count(1) as "# of Orders"

FROM Orders o

JOIN Customers c

ON o.CustomerID = c.CustomerID

GROUP BY c.CompanyName

, strftime('%Y%m',OrderDate);

* Return customers with at least 3 orders per month:

SELECT c.CompanyName

, strftime('%Y%m',OrderDate) as "Period"

, count(1) as "# of Orders"

FROM Orders o

JOIN Customers c

ON o.CustomerID = c.CustomerID

GROUP BY c.CompanyName

, strftime('%Y%m',OrderDate)

HAVING count(1) >= 3;

1. Now try to create SQL to answer to following questions:
   1. List the active customers?
   2. What are the total orders by customers?

See appendix for the answer.

# Northwind Data Model

A data model is an abstract representation of the data that reflects the business domain or part of it. It is usually a graphical representation as shown in the diagram below.

A screenshot of a computer

Description automatically generated

This simplified Northwind database contains eight tables which can be categorized into three different types of data:

1. Master data: Products, Customers, Suppliers, Shippers

Represents core business objects.

1. Reference data: Categories

Provides additional context or codes that are used to categorize or classify transaction or master data. It is used to standardize and organize information across the business domain or the organization.

1. Transaction data: Orders, Order Details

Captures business transactions or events. It represents the activities of the organization. Data is dynamic and changes frequently. It provides crucial business information.

# Data Visualization

While you can capture the relationships between tables using primary and secondary key, not all RDBMS have the capability to visualize the data model. Power BI is a free tool that allows you to build and visualize data models.

A screenshot of a computer

Description automatically generated

The ability to visualize data. Share your observations based on the diagram below:

A screenshot of a computer

Description automatically generated

# 

# Appendix – Answer to the query questions

1. List of active customers – 89 customers

SELECT DISTINCT c.CompanyName

FROM Orders o

JOIN Customers c

ON o.CustomerID = c.CustomerID

;

1. Total orders by customers

SELECT c.CompanyName

, Count(OrderID)

FROM Orders o

JOIN Customers c

ON o.CustomerID = c.CustomerID

GROUP BY c.CompanyName

ORDER by Count(OrderID) DESC

;