

4.3: Structured Query Language

Once you have a database designed and loaded with data, how will you do something useful with it? The primary way to work with a relational database is to use Structured Query Language, SQL (pronounced “sequel,” or stated as S-Q-L).

Definition: Structured Query Language (SQL)

SQL (structured query language) is a specialized programming language designed for managing and manipulating relational databases. It allows you to access and work with database data in a standard way.

Almost all applications that work with databases (such as database management systems, discussed below) use SQL to analyze and manipulate relational data. SQL allows users to interact and manage tables in a relational database, such as define the structure of the database, insert new data, update existing data, delete data, and, most importantly, query data (i.e., retrieve specific information).

Some common SQL commands are:

SELECT - Retrieve specific columns and rows from a database table. This might be used to look up a customer's purchase history.

INSERT - Add new data, like a new customer record, into a database.

UPDATE - Modify existing data in a database, like changing a product price.

DELETE - Remove data from a database table, like removing discontinued products.

✓ Example 4.3.1

Using the Student Clubs database, form a query to retrieve a list of the first and last names of the club presidents.

Solution

```
SELECT "First Name," "Last Name" FROM "Students" WHERE "Students.ID" =
```

✓ Example 4.3.2

Using the Student Clubs database, form a query will create a list of the number of students in each club, listing the club name and then the number of members.

Solution

```
SELECT "Clubs.Club Name", COUNT("Memberships.Student ID") FROM "Clubs"
```

SQL is the primary way database administrators and business analysts interact with many databases and a flexible approach to managing company data. Almost every major database system uses SQL. SQL skills allow analysts to enjoy the full potential of databases for key objectives like customer intelligence, streamlined operations, and data-driven strategy.

While SQL has a steeper learning curve than spreadsheets, it is a high-demand skill in the IT job market.

An in-depth description of how SQL works is beyond this introductory text's scope. Still, these examples should give you an idea of the power of using SQL to manipulate relational data. Many database packages, such as Microsoft Access, allow you to visually create the query you want to construct and then generate the SQL query for you. For more learning resources, you can check out free courses from [Codecademy](https://www.codecademy.com/), or watch this series of [videos SQL Basics](#) from Khan Academy.

4.3.1: Rows and Columns in a Table

In a relational database, all the tables are related by one or more fields, so it is possible to connect all the tables in the database through the field(s) they have in common. For each table, one of the fields is identified as a primary key. This key is the unique identifier for each record in the table. To help you understand these terms further, let's walk through the process of designing the following database.

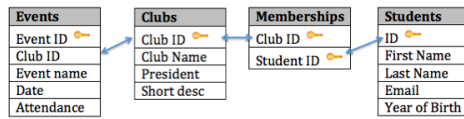


Figure 4.3.1: Data design flow. Image by David Bourgeois, Ph.D. is licensed under

CC BY 4.0

This page titled [4.3: Structured Query Language](#) is shared under a [CC BY 4.0](#) license and was authored, remixed, and/or curated by [Ly-Huong T. Pham and Tejal Desai-Naik](#) (Evergreen Valley College) .

- [4.3: Structured Query Language](#) by Ly-Huong T. Pham, Tejal Desai-Naik, Laurie Hammond, & Wael Abdeljabbar is licensed [CC BY 3.0](#).