

Structured Query Language (SQL)

CS 272 Software Development

Introduction

- **Structured Query Language (SQL)**
 - Often pronounced "sequel" or spelled out "S-Q-L"
 - Standard language for relational databases
- **Relational Database Management System (RDBMS)**
 - MySQL or MariaDB, PostgreSQL, SQL Server, Oracle, IBM DB2, SQLite, ...



SQL Statements

- **Keywords**
 - Traditionally UPPERCASE (optional)
 - Includes operators (e.g. =, >, etc.)
- **Identifiers**
 - Traditionally lowercase
- **Literal values**

<https://dev.mysql.com/doc/refman/8.0/en/language-structure.html>



Example SQL Statement

```
SELECT name FROM teams WHERE id=9;
```



Example Keywords

```
SELECT name FROM teams WHERE id=9;
```



Example Identifiers

```
SELECT name FROM teams WHERE id=9;
```



Example Literal

```
SELECT name FROM teams WHERE id=9;
```



Example Clause

```
SELECT name FROM teams WHERE id=9;
```

FROM Clause of a SELECT Statement



Data Definition Language (DDL)



Data Definition Language (DDL)

- Create and Manipulate Structure
 - e.g. tables, databases, schemas, etc.
- Example Statements
 - **CREATE:** Add new database object
 - **ALTER:** Manipulate existing database object
 - **DROP:** Remove database object

<https://mariadb.com/kb/en/data-definition/>



Data Definition Language (DDL)

```
CREATE TABLE [tab_name] (  
    [col_name] [DATA_TYPE] [CONSTRAINTS], ...  
);
```

```
CREATE TABLE students (  
    id INTEGER NOT NULL PRIMARY KEY,  
    name VARCHAR(50) NOT NULL,  
    Program VARCHAR(50));
```

<https://mariadb.com/kb/en/create-table/>



Column Data Types

- Numeric Data Types
 - e.g. TINYINT, INTEGER, DECIMAL, FLOAT, etc.
- Character Data Types
 - e.g. CHAR, VARCHAR, BLOB, ENUM, etc.
- Temporal Data Types
 - e.g. DATE, TIME, DATETIME, TIMESTAMP, etc.

<https://mariadb.com/kb/en/data-types/>



Numeric Data Types

| Type | Bytes | Minimum | Maximum |
|-----------|-------|--------------------------------|---------------|
| TINYINT | 1 | -128 or 0 | 127 or 255 |
| SMALLINT | 2 | -32,768 | 32,767 |
| MEDIUMINT | 3 | -8,388,608 | 8,388,607 |
| INTEGER | 4 | -2,147,483,648 | 2,147,483,647 |
| BIGINT | 8 | -9,223,372,036,854,775,808 ... | |

<https://mariadb.com/kb/en/numeric-data-type-overview/>



Numeric Data Types

- DECIMAL(precision[, scale]) for "exact" floating types
 - precision: total number of digits
 - scale: digits after decimal point
- FLOAT or DOUBLE for "approximate" floating types for really large or really small numbers

<https://mariadb.com/kb/en/data-types-numeric-data-types/>



Character Data Types

- CHAR(width) for fixed width text
- VARCHAR(maxwidth) for variable width text
- BLOB (binary large object) for images, files, etc.
- TEXT for large variable-sized text (e.g. articles)
- ENUM for fixed text values (e.g. types of fruit)

<https://mariadb.com/kb/en/string-data-types/>



Temporal Data Types

- DATE, TIME, DATETIME, TIMESTAMP, etc.
- Different input, storage, and display formats
- Input format depends on database system
 - e.g. usually YYYY-MM-DD date input format
- Display format controlled by query
 - e.g. MM/DD/YYYY

<https://mariadb.com/kb/en/date-and-time-data-types/>



Column Constraints

- **PRIMARY KEY**

- Indicates column values can be used to uniquely identify each row

- **AUTO_INCREMENT**

- Automatically increments value for each row (useful for primary keys)

<https://mariadb.com/kb/en/create-table/#column-definitions>



Column Constraints

- **UNIQUE**
 - Indicates column values must be unique
- **NULL** or **NOT NULL**
 - Indicates column values may (or may not) be null
- **DEFAULT**
 - Uses a default value if no value provided

<https://mariadb.com/kb/en/create-table/#column-definitions>



Data Manipulation Language (DML)



Data Manipulation Language (DML)

- Create and Manipulate Data
 - e.g. create or modify row in table
- Example Statements
 - **INSERT**: Creates new row(s) in a table
 - **UPDATE**: Updates value(s) in a table
 - **DELETE**: Removes rows from a table

<https://mariadb.com/kb/en/data-manipulation/>



Data Manipulation Language (DML)

```
INSERT INTO [tab_name]
([column_order])
VALUES ([column_values]);
```

```
INSERT INTO students
(name, degree)
VALUES ('Yasmin', 'CS');
```

<https://mariadb.com/kb/en/insert/>



SQL SELECT Statements

- Retrieves information from database
- Common clauses
 - FROM [tab_name]
 - WHERE [condition], HAVING [condition]
 - GROUP BY [col_name], ORDER BY [col_name]

<https://mariadb.com/kb/en/selecting-data/>



SQL SELECT JOIN Clauses

- Combines results together from multiple tables
- Multiple types of JOINS
 - e.g. INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN, NATURAL JOIN, etc.
- Critical component of SELECT statements

<https://mariadb.com/kb/en/joins/>



INNER JOIN

upper

| | |
|---|---|
| 1 | A |
| 2 | B |
| 3 | C |

lower

| | |
|---|---|
| b | 2 |
| c | 3 |
| d | 4 |

LEFT OUTER JOIN

upper

| | |
|---|---|
| 1 | A |
| 2 | B |
| 3 | C |

lower

| | |
|---|---|
| b | 2 |
| c | 3 |
| d | 4 |

LEFT OUTER JOIN

upper

| | |
|---|---|
| 1 | A |
| 2 | B |
| 3 | C |

lower

| | |
|---|---|
| b | 2 |
| c | 3 |
| d | 4 |

RIGHT OUTER JOIN

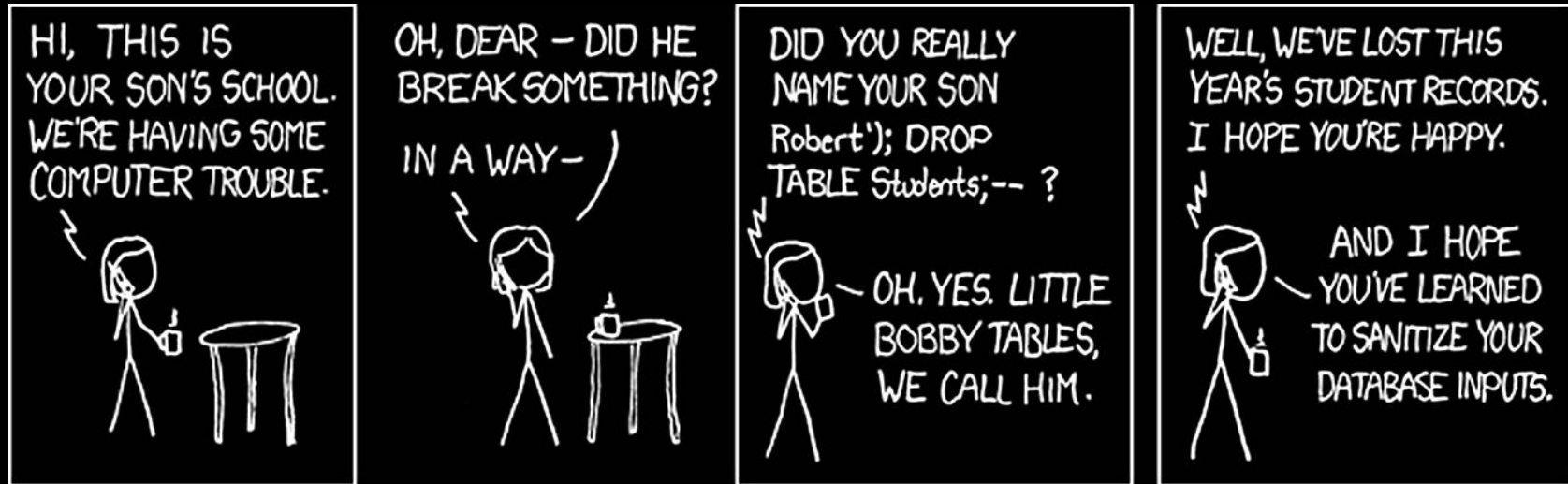
upper

| | |
|---|---|
| 1 | A |
| 2 | B |
| 3 | C |

lower

| | |
|---|---|
| b | 2 |
| c | 3 |
| d | 4 |

Bobby Tables



<https://xkcd.com/327/>

Questions?

