**Relational Databases:**

- Store data in rows and columns (like an Excel sheet) and can quickly and efficiently retrieve data from those tables (especially when there are multiple tables and the relationships between those tables are involved in the query)  
- *Database* – Contains many tables  
- *Relation/Table* – Contains tuples and attributes  
- *Tuple (Row)* – A set of fields that generally represent an object (like a person or artist name)  
- *Attribute (Column/Field)* – One of many possible elements of data corresponding to the object represented by the row (phone number, email address, album name, song name)  
- ***Database Schema*** – The outline for the database that determines rules for each filed (number of characters allowed, integer vs float)

**Structured Query Language (SQL):**

- The language used to issue commands to the database (Create table, retrieve table data, insert data, delete data)  
- Best practice is not to talk directly to database but to talk to an intermediary database application instead  
- Data needs to be cleaned first before it can be manipulated using SQL  
- Four basic functions (CRUD):   
1. **C**reate   
2. **R**ead   
3. **U**pdate   
4. **D**elete

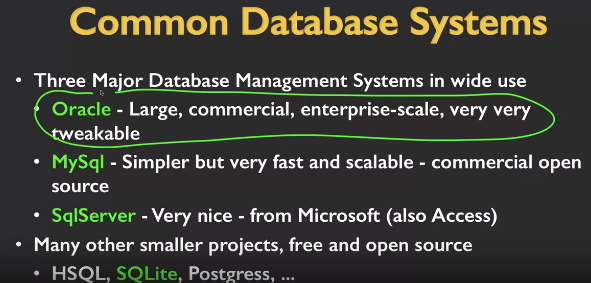
**Database Roles:**

- **Application Developer** – Builds the logic for the application, the UI, and monitors it for problems  
- **Database Administrator** – Monitors and adjusts the database as the program runs in production   
A diagram of software development

Description automatically generatedA diagram of a data analysis structure

Description automatically generated

Database Schema:

- The structure/format of the database. Provides rules and guidance on what can and cannot be contained in the database and how it is supposed to be stored  


- **Oracle** is for enterprise databases  
- **MySQL** is typically used for websites  
-**SQLite** is an embedded database that is built into software and not a standalone database that software must access  
- Already built into Python, just need to import it

**Single Table CRUD:**

- Need to be explicit about how large your data is going to be to help database be fast (variable characters up to 128 of them)  
A screenshot of a computer screen

Description automatically generated

**SQL Insert** – Inserts a new row into a table   
INSERT INTO Users (name, email) VALUES ('Kristin', 'kf@umich.edu')

**SQL Delete** – Deletes a row in a table  
DELETE FROM Users WHERE email='fred@umich.edu'

**SQL Update** – Allows for the updating of existing data in a row  
UPDATE Users SET name = 'Charles' WHERE email = 'csev@umich.edu'

**Retrieving Records** – Retrieves a group of records, either all of them or a subset if you use the WHERE clause  
SELECT \* FROM Users  
SELECT \*FROM Users WHERE email = 'csev@umich.edu'

**Ordering Records** – Used to get results sorted in ascending or descending order  
SELECT \* FROM Users ORDER BY email  
SELECT \* FROM Users ORDER BY name