

Summarize

- CMU Lecture 1
- Database Design (first 2 hours)
- SQL - Full Course for Beginners (1 hour)

[1] Introduction & Relational Model to Database Systems

- A database is an organized collection of inter-related data that models some aspect of the real-world (e.g., modeling the students in a class or a digital music store). People often confuse “databases” with “database management systems” (e.g., MySQL, Oracle, MongoDB). A database management system (DBMS) is the software that manages a database.
- A data model is a collection of concepts for describing the data in a database. The relational model is an example of a data model.
- A schema is a description of a particular collection of data, using a given data model.
- A schema is a description of a particular collection of data, using a given data model.

The relational data model defines three concepts:

- Keys and unique values
 - **unique value** is a value that doesn't show up in any other row in a given column. So there's one and only one of any particular value for that particular field.
 - unique values can also be used as keys
 - A table doesn't require a **primary key**, but having one helps to access specific records easily.

- **synthetic key or a surrogate key** is when making the value in the new column a number and telling the database to increment the number for every new number that's added >> ex: ID.
 - In some situations, we might not be able to modify the schema of the table, and we might need to `use combination of (two or more fields) in the data to act as a key`. This is called a **composite key**.
 - **Foreign key** this is what a primary key from one table is called when it's referenced in another table.
- Relationship rules and referential integrity
 - **Referential integrity**
 - means the database will be aware of the relationship and will not let you or another user modify data in a way that violates that relationship
 - it prevents the user from entering a record that refers to `nonexistent` data
 - This helps us to maintain the consistency of the database.

[2] Database Design

- A database is a collection of organized data stored and accessed electronically.
 - There are several types of databases, each with their own specific use cases and features. Some of the most common types include:
 1. Relational databases: These are the most widely used type of database, and they store data in tables with rows and columns. Relationships between tables are established using primary and foreign keys. Examples of relational databases include MySQL, Oracle, and Microsoft SQL Server.
 2. NoSQL databases: These databases are designed to handle large amounts of unstructured or semi-structured data. They do not use a fixed schema and do not rely on tables, rows, and columns to store data. Examples of NoSQL databases include MongoDB, Cassandra, and Redis.

3.

Object-oriented databases: These databases are designed to store and manipulate objects, which are a collection of data and methods. They are typically used in object-oriented programming languages such as Java and C++. Examples of object-oriented databases include Gemstone and Zope.

4.

Graph databases: These databases are used to store and query data in the form of a graph, with nodes and edges representing entities and relationships. They are particularly well-suited for use cases such as social networks and recommendation systems. Examples of graph databases include Neo4j and ArangoDB.

5.

Time-series databases: These databases are optimized for storing and querying time-stamped data, and they are commonly used for monitoring and tracking systems. Examples of time-series databases include InfluxDB and OpenTSDB.

6.

In-memory databases: These databases store data in RAM, which allows for faster data access and manipulation. They are often used for real-time analytics, high-performance transaction systems, and other use cases that require low latency. Examples of in-memory databases include SAP HANA and MemSQL.

- A relational database is a type of database that stores data in separate tables instead of putting all the data in one big store, this allows for better organization, easier management, and data integrity.
- RDBMS (Relational Database Management System) refers to the software used to manage and manipulate relational databases.
- Database design is the process of creating a detailed model of a database, including its data and the relationships between the data. Data integrity refers to maintaining and ensuring the accuracy, completeness, and consistency of data over its lifetime. There are many terms used in databases, including atomic values, relationships, one-to-one relationships, one-to-many relationships, and many-to-many relationships.

- 1. One-to-One Relationships: A one-to-one relationship occurs when two entities have exactly one instance of a related entity. For example, a person can have only one passport, and a passport belongs to only one person.
- 1. One-to-Many Relationships: A one-to-many relationship occurs when one entity has multiple related entities, but each related entity only has one corresponding entity. For example, a department has many employees, but each employee belongs to only one department.
- 1. Many-to-Many Relationships: A many-to-many relationship occurs when multiple entities have multiple related entities. For example, a student can enroll in many courses, and a course can have many students. This relationship is often resolved by creating a bridge or junction table.

[3] SQL Tutorial

- SQL
 - SQL stands for Structured Query Language and it is the syntax that allows us to interact with a SQL type database.
 - SQL allows us to write statements which the DBMS interprets, and that's how we interact with the data in the database, from apps, or even within the DBMS itself. In this role of interacting with data, SQL is called a `data manipulation language`, or **DML**.
 - **RDBMS** : relational database management system.
 - when we're asking for information from the database or asking for the database to do something, these statements are called **queries**
 - [create, read, update, delete] are called **crud**, and they're the basis of interacting with data.
- CRUD
 - CREATE --> `INSERT INTO (name) VALUES ('Asgard');`
 - READ --> `SELECT * FROM (name);`
 - UPDATE --> `UPDATE (name) SET sighting_date = '2023-4-2' WHERE id='1';`

- DELETE --> `DELETE FROM (name) WHERE id='1';` -> delete row where `id=1`
- UNION
 - The UNION operator is used to combine the result sets of two or more SELECT statements into a single result set. It returns only unique rows, meaning it removes any duplicates.
- JOIN
 - A join operation combines rows from two or more tables into a single result set based on a related column between them. There are several types of joins, including inner join, left join, right join, and full outer join.
- DELETE
 - DELETE clause specifies the action to be taken when a referenced row is deleted in a foreign key constraint. This can be set to either "CASCADE," meaning that all related rows in the referencing table will also be deleted, or "RESTRICT," meaning that the deletion will not be allowed if there are related rows in the referencing table.
- Triggers in SQL
 - A trigger is a special type of stored procedure that automatically executes in response to specific events, such as the insertion, update, or deletion of data. Triggers can be used to enforce business rules, validate data, and maintain audit trails.
- Nested Queries
 - A nested query is a query within another query. It is used to retrieve data from one table based on data from another table.