Introduction to Structured Query Language (SQL).

Overview : Relational Database

- What is Relational Database?
  - A relational database is a type of database that stores and provides access to data points that are related to one another.
- Relational databases are based on the relational model
  - an intuitive, straightforward way of representing data in tables.
- In a relational database, each row in the table is a record
  - A record has unique ID called the key.
- The columns of the table hold attributes of the data
  - Each record usually has a value for each attribute, making it easy to establish the relationships among data points.

Overview : Relational Database

- How Relational Databases are structured?

- The relational model is logical data structure
  - the data tables, views, and indexes are separate from the physical storage structures.
  - This separation means that database administrators can manage physical data storage without affecting access to that data as a logical structure.
  - For example, renaming a database file does not rename the tables stored within it.

Overview : Relational Database

- The distinction between logical and physical also applies to database operations.
  - Database operations are defined actions that enable applications to manipulate the data and structures of the database.
  - Logical operations allow an application to specify the content it needs
  - Physical operations determine how that data should be accessed and then carries out the task.
- Relational databases ensures that data is always accurate and accessible
  - Relational databases follow certain integrity rules.
  - For example, an integrity rule can specify that duplicate rows are not allowed in a table in order to eliminate the potential for erroneous information entering the database.

Overview : Benefits of Relational Database

- Data Consistency across applications and database copies (called *instances*).
  - For example, when a customer deposits money at an ATM and then looks at the account balance on a mobile phone, the customer expects to see that deposit reflected immediately in an updated account balance.
  - Relational databases excel at this kind of data consistency.
  - It ensures that multiple instances of a database have the same data all the time.

Overview : Benefits of Relational Database

- It is difficult for other types of databases to maintain this level of timely consistency with large amounts of data.
  - Some recent databases, such as NoSQL, can supply only “eventual consistency.”
  - Under this principle, when the database is scaled or when multiple users access the same data at the same time, the data needs some time to “catch up.”
  - Eventual consistency is acceptable for some uses, such as to maintain listings in a product catalog
  - For critical business operations such as shopping cart transactions, the relational database is still the gold standard.

Overview : Benefits of Relational Database

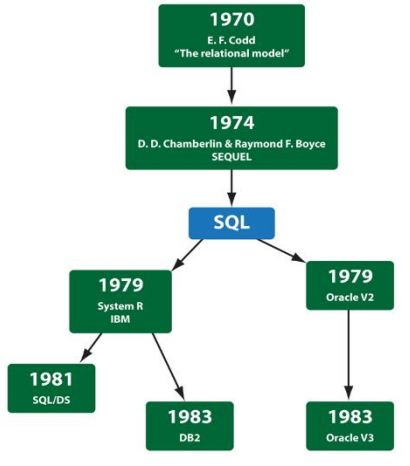
- Commitment and Atomicity
  - Relational databases handle business rules and policies at a very granular level
  - They implement strict policies about *commitment* (that is, making a change to the database permanent).
  - For example, consider an inventory database that tracks three parts that are always used together. When one part is pulled from inventory, the other two must also be pulled. If one of the three parts isn’t available, none of the parts should be pulled—all three parts must be available before the database makes any commitment.

Overview : Benefits of Relational Database

- A relational database won’t commit for one part until it knows it can commit for all three. This multifaceted commitment capability is called atomicity.
  - Atomicity is the key to keeping data accurate in the database and ensuring that it is compliant with the rules, regulations, and policies of the business.

History of the SQL

- SQL stands for “Structured Query Language”.
- Initially, it was called SEQUEL (Structured English Query Language)
- Raymond Boyce and Donald Chamberlin developed SQL at IBM in the early 1970s.
- SQL has turned into an official standard for the ANSI (American National Standards Institute) and ISO (International Organization for Standardization).



History : The Relational Model

- In the early years of databases, every application stored data in its own unique structure.
  - When developers wanted to build applications to use that data, they had to know a lot about the particular data structure to find the data they needed.
  - These data structures were inefficient, hard to maintain, and hard to optimize for delivering good application performance.
  - The relational database model was designed to solve the problem of multiple arbitrary data structures.
- The relational model provided a standard way of representing and querying data
  - Chief strength of the relational database model was in its use of tables.
  - Tables are an intuitive, efficient, and flexible way to store and access structured information.

History : The Relational Model

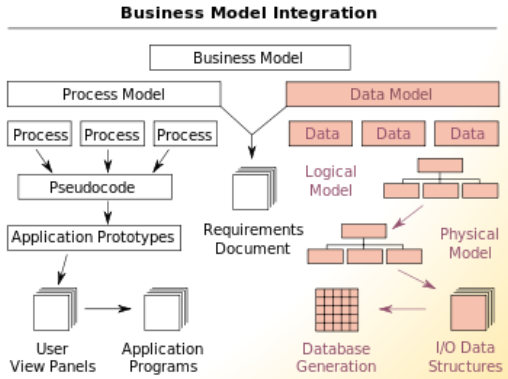
- Over time, another strength of the relational model emerged
  - As developers began to use structured query language (SQL) to write and query data in a database.
  - For many years, SQL has been widely used as the language for database queries.
  - Based on relational algebra, SQL provides an internally consistent mathematical language that makes it easier to improve the performance of all database queries.
  - In comparison, other approaches must define individual queries.

Structured Query Language

- Declarative language common to most database management systems.
- Used by the database system internally and by users to manipulate and query the data.
- Uses set theory (as opposed to row processing) to process requests.
- Limited but flexible set of commands.
- Free-form, but helpful to adopt a consistent style.

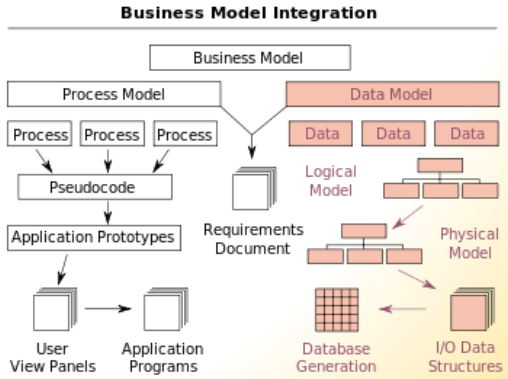
Database Model - Structured Query Language

- Database model organizes data into one or more tables (or "relations") of columns and rows.
- A unique key identifies each row.
- Rows are also called records or tuples.
- Columns are also called attributes.



Database Model - Structured Query Language

- Each table/relation represents one "entity type" (such as customer or product).
- The rows represent instances of that type of entity (such as “bookcase" or "chair")
- The columns representing values attributed to that instance (such as address or price).



Relational Database

Questions