Intro to Database

Topics for Today

Intro to DB

Relational Models

Primary and Foreign Keys

ERD

Intro to Database

Used files to store data before (txt, json, csv)

What is Database?

- Definition: An **organized** collection of data
- Purpose: Store, manage, and retrieve data efficiently
- Examples: Student records, inventory, customer information

Types of Databases

- **Relational**: Tables with rows and columns
- Hierarchical: Tree-like structure
- Network: Complex relationships
- NoSQL: Flexible data models

Why Database?

- Data **Integrity**: Ensuring data accuracy and consistency
- Data **Security**: Protecting data from unauthorized access
- Data **Sharing**: Multiple users accessing data simultaneously
- Data **Backup** and **Recovery**: Protecting data from loss

Relational Database

- Based on the relational model
- Organized into tables
- Relationships between tables

Examples: School database has tables for teachers, students, courses, etc.

Advantages of Relational Database

Data independence

Flexible

Scalable

ACID Compliance

ACID

Atomicity

- All database operations are treated as a single unit
- Either all operations are completed or none are
- Example: Bank transaction (deposit or withdrawal)

Consistency

- Database transitions from one valid state to another
- Data integrity is maintained
- Example: Account balance remains correct after a transaction

Isolation

- Concurrent transactions do not interfere with each other
- Each transaction sees a consistent view of the database
- Example: Multiple users accessing the same data

Durability

- Committed transactions are permanently stored
- Data is not lost in case of system failure
- Example: Database backups and recovery

Disadvantages of Relational Database

- Performance issues with large datasets
- Complex schema design

Database Management System (DBMS)

Software that is used to control and manage data access.

Organised and a safer way to use Database.

Database Administrator is responsible for the system.

Examples: MySQL, Microsoft Access, Oracle, PostgreSQL, etc.

Structured Query Language (SQL)

Standard language for managing relational databases

Used for: data **definition**, **manipulation**, and **control**

Data Definition Language (DDL): Create, modify, and delete database structures

Data Manipulation Language (DML): Insert, update, delete, and retrieve data

Data Control Language (DCL): Grant and revoke user permissions

Relational Models

Logical Representation of a data model.

Tables: A collection of related data organized into rows and columns

- Rows: Records
- Columns: Fields or attributes
- Example: Student Table

Keys: Attributes with a purpose

- Primary Key: Uniquely identifies each row in a table (Student ID)
- Candidate Key: Any attribute or combination of attributes that can uniquely identify a row
- Foreign Key: A field in one table that refers to the primary key in another table

Normalisation: Organising data

- Purpose: Remove redundancy, Improve Data Integrity
- First Normal Form (1NF), Second Normal Form (2NF), Third Normal Form (3NF)

Primary and Foreign Keys

Primary Key:

- Uniquely identifies each row in a table
- Cannot be null
- Should be chosen carefully for performance

Foreign Key:

- A field in one table that references the primary key in another table
- Establishes relationships between tables
- Enforces data integrity

Examples

Student, Course, Subjects

In Excel

Normalisation

1NF (First Normal Form):

- Eliminates duplicate rows from a table.
- Ensures each column value is atomic (cannot be further divided).

2NF (Second Normal Form):

- Satisfies all the requirements of 1NF.
- Eliminates partial dependencies.
- A partial dependency exists when a non-key attribute depends only on a part of the primary key.
- Foreign keys are used here.

3NF (Third Normal Form):

- Satisfies all the requirements of 2NF.
- Eliminates transitive dependencies.
- A transitive dependency exists when a non-key attribute depends on another non-key attribute, which in turn depends on the primary key.

Example of Normalisation

Excel sheet

Entity-Relationship Diagram

Conceptual representation of the database

Entity and Attributes

Entities: Real-world objects or concepts (e.g., Student, Course, Subjects)

Attributes: Properties of entities (e.g., StudentID, Name, CourseCode, CourseName, SubjectCode, SubjectName)

https://app.diagrams.net/

Relationships

Associations between entities (e.g., Student Enrolls in Course)

Cardinality: Number of instances involved in a relationship (one-to-one, one-to-many, many-to-many)

Notation: Symbols, Crow's foot notation for cardinality.