SQL Basics and Database Operations

Date: Jan 20, 2026

Overview

This class covered the fundamentals of databases and SQL. The focus was on understanding what a database is, how SQL works, different SQL command types, table operations, constraints, keys, and how data is created, modified, and deleted safely.

Key Topics Covered

Database & DBMS

- **Database**: A collection of structured data stored in tables.
- **DBMS (Database Management System)**: Software used to manage databases.
- Example mentioned: MySQL Server with MySQL Workbench as the interface.

SQL (Structured Query Language)

- SQL is used to create, read, update, and delete data in relational databases.
- Works on tables that contain rows (records) and columns (fields).

Database Structure

- Server

  → Database

  → Schema

→ Tables

→ Columns & Rows

SQL Command Categories

DDL – Data Definition Language

Used to define or change table structure.

- CREATE
- ALTER
- DROP
- TRUNCATE
- RENAME

DML – Data Manipulation Language

Used to modify data inside tables.

- INSERT
- UPDATE
- DELETE

DQL – Data Query Language

Used to fetch data.

- SELECT

DCL – Data Control Language

Used for permissions.

- GRANT
- REVOKE

TCL – Transaction Control Language

Used to manage transactions.

- COMMIT
- ROLLBACK
- SAVEPOINT

CRUD Operations

- **Create** → INSERT
- **Read** → SELECT
- **Update** → UPDATE
- **Delete** → DELETE

Constraints & Keys

Primary Key

- Uniquely identifies each record.
- Cannot be NULL.
- Ensures data uniqueness.

Foreign Key

- Refers to a primary key in another table.
- Creates a relationship between tables (parent–child).

Other Constraints

- **NOT NULL** – value must be provided
- **UNIQUE** – no duplicate values
- **CHECK** – condition-based validation (example: age restriction)
- **DEFAULT** – assigns a value if none is given (example: city = New York)

Referential Actions

ON DELETE

- **RESTRICT** – cannot delete parent if child exists
- Rule mentioned: *Delete child first, then parent*

ON UPDATE

- **CASCADE** – updates child records automatically when parent key changes
- Example: phone number update propagates automatically

IF EXISTS

- Used to avoid errors while dropping objects.
- Example use case:

  Drop table only if it already exists.

DROP vs TRUNCATE vs DELETE

DELETE

- Deletes selected rows using WHERE condition.
- Table structure remains.

TRUNCATE

- Deletes all rows.
- Table structure remains.
- Faster than DELETE.
- Data cannot be recovered easily.

DROP

- Deletes table completely.
- Both structure and data are removed.

Relationships in Database

- One-to-One
- One-to-Many
- Many-to-Many

Important Concepts / Definitions

- **Schema**: Structure of a table (columns and data types)
- **Metadata**: Information about table structure, not actual data
- **Common Table Expression (CTE)**: Mentioned conceptually (used for readable queries)

Key Takeaways

- Always use proper constraints to maintain data integrity.
- Understand differences between DELETE, TRUNCATE, and DROP.
- Use IF EXISTS to prevent SQL errors.

- Be careful with parent–child relationships when updating or deleting data.
- SQL commands are grouped based on purpose (DDL, DML, DQL, DCL, TCL).

Action Items / Things to Remember

- Primary key must always be unique and not null.
- Foreign keys control relationships between tables.
- Use CASCADE carefully—it automatically affects related data.
- Always think before running DROP or TRUNCATE commands.
- Use WHERE clause with DELETE to avoid removing unintended data.