

# SQL Basics Expanded

Page 1: Introduction to SQL

SQL (Structured Query Language) is used to manage and manipulate relational databases.

Types of SQL Statements:

- Data Definition Language (DDL): CREATE, ALTER, DROP
- Data Manipulation Language (DML): SELECT, INSERT, UPDATE, DELETE
- Data Control Language (DCL): GRANT, REVOKE
- Transaction Control Language (TCL): COMMIT, ROLLBACK

Why use SQL?

- Widely used for data handling and analysis.
- Standardized language supported by most database systems.

## Page 2: Basic SQL Queries

SELECT:

```
SELECT * FROM employees;
```

INSERT:

```
INSERT INTO employees (name, age, department) VALUES ('Alice', 30, 'HR');
```

UPDATE:

```
UPDATE employees SET age = 31 WHERE name = 'Alice';
```

DELETE:

```
DELETE FROM employees WHERE name = 'Alice';
```

## Page 3: Data Constraints

Constraints ensure data accuracy and integrity.

- Primary Key: Ensures unique and non-null values.
- Foreign Key: Links two tables.
- Unique: Prevents duplicate values.
- Not Null: Ensures a column cannot have NULL values.

Example:

```
CREATE TABLE employees (  
    id INT PRIMARY KEY,  
    name VARCHAR(50) NOT NULL,  
    dept_id INT,  
    FOREIGN KEY (dept_id) REFERENCES departments(dept_id)  
);
```

## Page 4: Joins

Joins combine rows from two or more tables.

- INNER JOIN: Returns matching rows.
- LEFT JOIN: Returns all rows from the left table.
- RIGHT JOIN: Returns all rows from the right table.
- FULL JOIN: Combines results of LEFT and RIGHT JOIN.

Example:

```
SELECT employees.name, departments.dept_name
```

```
FROM employees
```

```
INNER JOIN departments ON employees.dept_id = departments.dept_id;
```

## Page 5: Functions and Aggregation

SQL provides functions for data analysis:

- COUNT: Counts rows.
- AVG: Calculates average.
- SUM: Totals values.
- MAX, MIN: Finds maximum and minimum values.

Example:

```
SELECT dept_id, AVG(salary) AS avg_salary  
FROM employees  
GROUP BY dept_id;
```

## Page 6: Transactions

Transactions ensure data integrity and consistency.

- COMMIT: Saves all changes.
- ROLLBACK: Reverts changes.

Example:

BEGIN;

UPDATE accounts SET balance = balance - 100 WHERE id = 1;

UPDATE accounts SET balance = balance + 100 WHERE id = 2;

COMMIT;

