SQL Cheat Sheet

Basics

Comments

- Single-line comment: -- This is a comment
- Multi-line comment: /* This is a multi-line comment */

Data Definition Language (DDL)

Create Table

CREATE TABLE table_name (

   column1 datatype [constraints],

   column2 datatype [constraints],

   ...

);

Note: Use PRIMARY KEY, FOREIGN KEY, NOT NULL, UNIQUE, etc., as constraints.

Alter Table

-- Add column

ALTER TABLE table_name ADD column_name datatype;

-- Modify column

ALTER TABLE table_name MODIFY column_name datatype;

-- Drop column

ALTER TABLE table_name DROP COLUMN column_name;

Drop Table

DROP TABLE table_name;

Warning: Deletes the table and all its data permanently.

Data Manipulation Language (DML)

Insert Data

INSERT INTO table_name (column1, column2, ...) VALUES (value1, value2, ...);

Update Data

UPDATE table_name

SET column1 = value1, column2 = value2

WHERE condition;

Warning: Always use WHERE to avoid updating all rows.

Delete Data

DELETE FROM table_name WHERE condition;

Caution: Missing WHERE deletes all rows.

Querying Data

Basic Query

SELECT column1, column2 FROM table_name;

Filtering Data

SELECT * FROM table_name

WHERE condition;

Common Operators

- Equals: =

- Not Equals: != or <>

- Greater Than / Less Than: > / <

- LIKE (pattern matching): column LIKE 'pattern%'

- IN (list matching): column IN (value1, value2, ...)

Sorting

SELECT * FROM table_name

ORDER BY column_name [ASC|DESC];

Aggregation

```sql
SELECT COUNT(column_name), AVG(column_name), SUM(column_name), MIN(column_name),
MAX(column_name)

FROM table_name;
```

Grouping

```sql
SELECT column_name, COUNT(*)

FROM table_name

GROUP BY column_name;
```

Joining Tables

```sql
-- Inner Join

SELECT * FROM table1

INNER JOIN table2

ON table1.column_name = table2.column_name;


-- Left Join

SELECT * FROM table1

LEFT JOIN table2

ON table1.column_name = table2.column_name;


-- Right Join

SELECT * FROM table1

RIGHT JOIN table2

ON table1.column_name = table2.column_name;


-- Full Join

SELECT * FROM table1

FULL JOIN table2

ON table1.column_name = table2.column_name;
```

Indexing

Create Index

CREATE INDEX index_name

ON table_name (column_name);

Drop Index

DROP INDEX index_name;

Transactions

Start, Commit, and Rollback

-- Start transaction

BEGIN TRANSACTION;

-- Commit changes

COMMIT;

-- Rollback changes

ROLLBACK;

Useful Functions

String Functions

- UPPER(column_name): Converts to uppercase

- LOWER(column_name): Converts to lowercase

- CONCAT(string1, string2, ...): Concatenates strings

Date Functions

- NOW(): Current date and time

- CURDATE(): Current date

- DATEDIFF(date1, date2): Difference between dates

Example Query

```sql
SELECT employees.name, departments.name AS department
FROM employees
INNER JOIN departments
ON employees.department_id = departments.id
WHERE employees.salary > 50000
ORDER BY employees.name ASC;
```