| Data Definition Language (DDL) Commands | | |
| --- | --- | --- |
| Command | Explanation | Syntax |
| CREATE TABLE: | Creates a new table in the database with specified columns and constraints. | CREATE TABLE table_name ( column1 datatype [constraint], column2 datatype [constraint], ... ); |
| ALTER TABLE: | Modifies an existing table structure by adding, modifying, or dropping columns or constraints. | Syntax given below: |
| ALTER TABLE table_name ADD column_name datatype [constraint]; | ALTER TABLE table_name MODIFY column_name datatype [constraint]; | ALTER TABLE table_name DROP COLUMN column_name; |
| DROP TABLE: | Deletes a table from the database. | DROP TABLE table_name; |
| TRUNCATE TABLE: | Removes all records from a table, but keeps the table structure intact. | TRUNCATE TABLE table_name; |
| CREATE INDEX: | Creates an index on one or more columns of a table, which speeds up data retrieval operations. | CREATE INDEX index_name ON table_name (column1, column2, ...); |
| DROP INDEX: | Removes an index from the database. | DROP INDEX index_name; |
| CREATE VIEW: | Creates a virtual table based on the result set of a SELECT query, which can simplify complex queries and provide data security. | CREATE VIEW view_name AS SELECT column1, column2, ... FROM table_name WHERE condition; |
| DROP VIEW: | Deletes a view from the database. | DROP VIEW view_name; |
| CREATE SCHEMA: | Creates a new schema in the database, which is a logical container for database objects. | CREATE SCHEMA schema_name; |
| DROP SCHEMA: | Deletes a schema from the database, along with all its contained objects. | DROP SCHEMA schema_name; |

## Data Manipulation Language (DML) Commands

| | | |
|---|---|---|
| INSERT INTO: | Adds new records into a table. | INSERT INTO table_name (column1, column2, ...) VALUES (value1, value2, ...); |
| UPDATE: | Modifies existing records in a table. | UPDATE table_name SET column1 = value1, column2 = value2, ... WHERE condition; |
| DELETE FROM: | Removes records from a table. | DELETE FROM table_name WHERE condition; |

## Data Query Language (DQL) Commands

| | | |
|---|---|---|
| SELECT: | Retrieves data from one or more tables in a database. | SELECT column1, column2, ... FROM table_name WHERE condition; |
| FROM: | Specifies the tables from which data is retrieved in a SELECT statement. | SELECT column1, column2, ... FROM table_name WHERE condition; |
| WHERE: | Filters rows based on specified conditions in a SELECT statement. | SELECT column1, column2, ... FROM table_name WHERE condition; |
| GROUP BY: | Explanation: Groups rows that have the same values into summary rows, typically used with aggregate functions like COUNT, SUM, AVG, etc. | SELECT column1, column2, ... FROM table_name GROUP BY column1, column2, ...; |
| HAVING: | Filters groups based on specified conditions in a GROUP BY query. | SELECT column1, column2, ... FROM table_name GROUP BY column1 HAVING condition; |
| ORDER BY: | Sorts the result set in ascending or descending order based on one or more columns. | SELECT column1, column2, ... FROM table_name ORDER BY column1 [ASC\|DESC]; |
| DISTINCT: | Retrieves unique values from a specific column or combination of columns in a SELECT query. | SELECT DISTINCT column1, column2, ... FROM table_name; |
| LIMIT: | Limits the number of rows returned by a SELECT query. | SELECT column1, column2, ... FROM table_name LIMIT n; |

## Data Control Language (DCL) Commands

| | | |
|---|---|---|
| GRANT: | The GRANT command is used to assign specific privileges to database users or roles, allowing them to perform certain actions on database objects. | GRANT privileges ON object TO user; |
| REVOKE: | The REVOKE command is used to withdraw previously granted privileges from database users or roles, restricting their access to specific database objects. | REVOKE privileges ON object FROM user; |

| Transaction Control Language (TCL) Commands | | |
|---|---|---|
| COMMIT: | Saves the changes made during the current transaction permanently to the database. | COMMIT; |
| ROLLBACK: | Rolls back the changes made during the current transaction to the last savepoint or the beginning of the transaction. | ROLLBACK; |
| SAVEPOINT: | Sets a named savepoint within the current transaction, allowing partial rollback to that point. | SAVEPOINT savepoint_name; |
| RELEASE SAVEPOINT: | Removes a previously defined savepoint, releasing the transaction's resources. | RELEASE SAVEPOINT savepoint_name; |
| SET TRANSACTION: | Sets characteristics for the transaction, such as isolation level or read/write mode. | SET TRANSACTION [transaction_characteristics]; |
| BEGIN TRANSACTION: | Begins a new transaction explicitly. | BEGIN TRANSACTION; |
| ROLLBACK TO SAVEPOINT: | Rolls back the transaction to a specific savepoint. | ROLLBACK TO SAVEPOINT savepoint_name; |