**1. Database Commands**

| **Command** | **Purpose** | **Example** |
| --- | --- | --- |
| CREATE DATABASE | Create a new database | CREATE DATABASE mydb; |
| SHOW DATABASES | List all databases | SHOW DATABASES; |
| USE | Switch to a database | USE mydb; |
| DESCRIBE DATABASE | Show database details | DESCRIBE DATABASE mydb; |
| DROP DATABASE | Delete a database | DROP DATABASE mydb CASCADE; |

**🔹 2. Table Commands**

| **Command** | **Purpose** | **Example** |
| --- | --- | --- |
| CREATE TABLE | Create table | CREATE TABLE student (id INT, name STRING, marks INT) ROW FORMAT DELIMITED FIELDS TERMINATED BY ','; |
| SHOW TABLES | List all tables | SHOW TABLES; |
| DESCRIBE | Show table structure | DESCRIBE student; |
| ALTER TABLE | Modify table (add/rename column, partition) | ALTER TABLE student ADD COLUMNS (age INT); |
| DROP TABLE | Delete a table | DROP TABLE student; |
| TRUNCATE TABLE | Delete all data but keep structure | TRUNCATE TABLE student; |

**🔹 3. DML (Data Manipulation Language)**

| **Command** | **Purpose** | **Example** |
| --- | --- | --- |
| LOAD DATA | Load file into table | LOAD DATA LOCAL INPATH '/home/user/students.csv' INTO TABLE student; |
| INSERT INTO | Insert new rows | INSERT INTO student VALUES (1, 'Alice', 85); |
| INSERT OVERWRITE | Overwrite table/partition | INSERT OVERWRITE TABLE student SELECT \* FROM temp; |
| EXPORT | Export table to HDFS path | EXPORT TABLE student TO '/user/hive/export/student'; |
| IMPORT | Import table data | IMPORT TABLE student FROM '/user/hive/export/student'; |

**🔹 4. DQL (Data Query Language)**

| **Command** | **Purpose** | **Example** |
| --- | --- | --- |
| SELECT | Query data | SELECT name, marks FROM student WHERE marks > 50; |
| ORDER BY | Global sort | SELECT \* FROM student ORDER BY marks DESC; |
| SORT BY | Sort within reducer | SELECT \* FROM student SORT BY marks; |
| DISTRIBUTE BY | Distribute rows across reducers | SELECT \* FROM student DISTRIBUTE BY id; |
| CLUSTER BY | Equivalent to DISTRIBUTE BY + SORT BY | SELECT \* FROM student CLUSTER BY id; |
| GROUP BY | Group data | SELECT subject, COUNT(\*) FROM student GROUP BY subject; |
| JOIN | Join tables | SELECT a.id, a.name, b.marks FROM student a JOIN marks b ON (a.id=b.id); |
| UNION ALL | Merge results | SELECT name FROM student1 UNION ALL SELECT name FROM student2; |

**🔹 5. Other Useful Commands**

| **Command** | **Purpose** | **Example** |
| --- | --- | --- |
| SHOW FUNCTIONS | List functions | SHOW FUNCTIONS; |
| DESCRIBE FUNCTION | Explain function | DESCRIBE FUNCTION upper; |
| SET | View/modify configuration | SET mapreduce.job.reduces=5; |
| ! | Run Linux command from Hive | ! ls -l; |

**✅ Example Workflow**

-- Create and use database

CREATE DATABASE college;

USE college;

-- Create table

CREATE TABLE student (id INT, name STRING, marks INT)

ROW FORMAT DELIMITED

FIELDS TERMINATED BY ',';

-- Load data

LOAD DATA LOCAL INPATH '/home/user/students.csv' INTO TABLE student;

-- Query

SELECT name, marks FROM student WHERE marks > 60 ORDER BY marks DESC;