



Basic Commands & Syntax

Connecting & Creating Databases

<code>sqlite3 database_name.db</code>	- Creates a new SQLite database or opens an existing one.
<code>.help</code>	- Displays available SQLite commands.
<code>.databases</code>	- Lists the currently open databases.
<code>.exit</code> or <code>.quit</code>	- Closes the SQLite connection.
<code>.show</code>	- Show current settings.

Basic SQL Operations

<code>CREATE TABLE table_name (column1 datatype, column2 datatype, ...);</code>	Creates a new table in the database.
<code>INSERT INTO table_name (column1, column2, ...) VALUES (value1, value2, ...);</code>	Inserts a new row into the table.
<code>SELECT column1, column2, ... FROM table_name WHERE condition;</code>	Retrieves data from the table based on the specified condition.
<code>UPDATE table_name SET column1 = value1, column2 = value2, ... WHERE condition;</code>	Modifies existing data in the table based on the specified condition.
<code>DELETE FROM table_name WHERE condition;</code>	Deletes rows from the table based on the specified condition.

Data Types & Constraints

Common Data Types

<code>INTEGER</code>	: Stores integer values.
<code>TEXT</code>	: Stores text strings.
<code>REAL</code>	: Stores floating-point numbers.
<code>BLOB</code>	: Stores binary data.
<code>NUMERIC</code>	: Stores numeric values (can behave like <code>INTEGER</code> or <code>REAL</code>).

Constraints

<code>PRIMARY KEY</code>	Uniquely identifies each row in a table.
<code>NOT NULL</code>	Ensures that a column cannot have a <code>NULL</code> value.
<code>UNIQUE</code>	Ensures that all values in a column are distinct.
<code>CHECK (expression)</code>	Specifies a condition that must be true for any row in the table.
<code>DEFAULT value</code>	Specifies a default value for a column when no value is provided.
<code>FOREIGN KEY</code>	Establishes a link between tables.

Advanced Queries

WHERE Clause

<code>SELECT * FROM table_name WHERE column1 = 'value';</code>	- Filters rows where column1 equals 'value'.
<code>SELECT * FROM table_name WHERE column2 > 10;</code>	- Filters rows where column2 is greater than 10.
<code>SELECT * FROM table_name WHERE column3 LIKE 'pattern%';</code>	- Filters rows where column3 starts with 'pattern'.
<code>SELECT * FROM table_name WHERE column4 IS NULL;</code>	- Filters rows where column4 is <code>NULL</code> .
<code>SELECT * FROM table_name WHERE column5 BETWEEN 1 AND 10;</code>	- Filters rows where column5 is between 1 and 10 (inclusive).

JOIN Operations

<code>INNER JOIN</code>	Returns rows only when there is a match in both tables.
<code>LEFT JOIN</code>	Returns all rows from the left table and matching rows from the right table. If no match, the right side will contain nulls.
<code>RIGHT JOIN</code>	Returns all rows from the right table and matching rows from the left table. If no match, the left side will contain nulls.
<code>FULL OUTER JOIN</code>	Returns all rows when there is a match in one of the tables.
<code>CROSS JOIN</code>	Returns all possible combinations of rows from all tables.

GROUP BY and Aggregate Functions

<code>GROUP BY column_name</code>	- Groups rows that have the same values in a column into summary rows, like how many customers are in each country.
<code>COUNT(column_name)</code>	- Returns the number of items in a group.
<code>SUM(column_name)</code>	- Returns the sum of the values in a group.
<code>AVG(column_name)</code>	- Returns the average of the values in a group.
<code>MIN(column_name)</code>	- Returns the smallest value in a group.
<code>MAX(column_name)</code>	- Returns the largest value in a group.

Managing Tables

Altering Tables

<code>ALTER TABLE table_name ADD COLUMN column_name datatype;</code>	- Adds a new column to an existing table.
<code>ALTER TABLE table_name RENAME TO new_table_name;</code>	- Renames an existing table.

Dropping Tables

<code>DROP TABLE table_name;</code>	- Deletes a table and its data.
-------------------------------------	---------------------------------

Indexes

<code>CREATE INDEX index_name ON table_name (column_name);</code>	Creates an index on a column to speed up query performance.
<code>DROP INDEX index_name;</code>	Removes an index.