



# SQL is a standard language for storing, manipulating and retrieving data in databases.



**SELECT** - extracts data from a database

**UPDATE** - updates data in a database

**DELETE** - deletes data from a database

**INSERT INTO** - inserts new data into a database

**CREATE DATABASE** - creates a new database

**ALTER DATABASE** - modifies a database

**CREATE TABLE** - creates a new table

**ALTER TABLE** - modifies a table

**DROP TABLE** - deletes a table

**CREATE INDEX** - creates an index (search key)

**DROP INDEX** - deletes an index



## SQL PRIMARY KEY Constraint



The PRIMARY KEY constraint uniquely identifies each record in a table.

Primary keys must contain UNIQUE values, and cannot contain NULL values.

A table can have only ONE primary key; and in the table, this primary key can consist of single or multiple columns (fields).

## SQL FOREIGN KEY Constraint

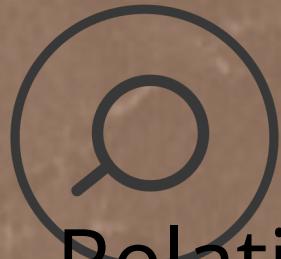
The FOREIGN KEY constraint is used to prevent actions that would destroy links between tables.

A FOREIGN KEY is a field (or collection of fields) in one table, that refers to the PRIMARY KEY in another table.

The table with the foreign key is called the child table, and the table with the primary key is called the referenced or parent table.

Look at the following two tables:

# Relationships in SQL - One-to-One, One-to-Many, Many-to-Many.



Relationships in SQL tables define how tables are connected to one another. Building relationships in tables helps to organize and link data across multiple tables. Creating relationships among tables provides efficient data retrieval and maintenance.

## 1. One-to-One Relationship

**Definition:** Each record in Table A is associated with one and only one record in Table B, and vice versa.

**Setup:** Include a foreign key in one of the tables that references the primary key of the other table.

**For example:** Tables users and user\_profiles, where each user has a single corresponding profile.



TABLE A		TABLE B		
user_id	username	profile_id	user_id	profile_data
1	ramesh	p01	1	xyz
2	riya	p02	2	abc
3	akhil	p03	3	gfg

one-to-one relationship

foreign key  
primary key



## 2. One-to-Many Relationship

**Definition:** Each record in Table A can be associated with multiple records in Table B, but each record in Table B is associated with only one record in Table A.

**Setup:** Include a foreign key in the "many" side table (Table B) that references the primary key of the "one" side table (Table A).

**For example:** Tables departments and employees, where each department can have multiple employees, but each employee belongs to one department.



Departments		employee_id	Employees	
department_id	department_name	employee_id	employee_name	department_id
d1	technical	e01	Ramesh	d3
d2	accounts	e02	Riya	d1
d3	pr	e03	Neha	d2
d4	product management	e04	Mayank	d1
		e05	Kritila	d4
		e06	Anuj	d4
		e07	Sam	d1
		e08	Gurpreet	d2
			foreign key	
			primary key	



### 3. Many-to-Many Relationship

**Definition:** Each record in Table A can be associated with multiple records in Table B, and vice versa.

**Setup:** Create an intermediate table (also known as a junction or linking table) that contains foreign keys referencing both related tables.

**For example:** Tables students and courses, where each student can enroll in multiple courses, and each course can have multiple students.



STUDENTS		COURSES		STUDENT_COURSES	
student_id	student_name	course_id	course_name	student_id	course_id
1	Alice	101	Mathematics	1	101
2	Bob	102	History	1	102
3	Charlie	103	Computer Science	2	102

foreign key  
primary key  
both primary and foreign key



## 5. Self-Referencing Relationship

**Definition:** A table has a foreign key that references its primary key.

**Setup:** Include a foreign key column in the same table that references its primary key.

**For example :** A table employees with a column manager\_id referencing the same table's employee\_id.



employees		
employee_id	employee_name	manager_id
1	Alice	NULL
2	Bob	1
3	Charlie	1
		foreign key
		primary key