INDIAN DATA CLUB

INDIAN DATA CLUB

Day 13 (17/11): INNER JOIN

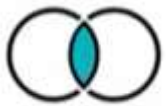
🎯 *Objective*

To understand how **INNER JOIN** works in SQL and learn how to combine related data from multiple tables using matching keys.

🔍 *Topics Covered*

- What is an **INNER JOIN** and when to use it
- Joining two tables using a common key
- Writing basic INNER JOIN queries
- Using table aliases for cleaner queries
- INNER JOIN with multiple conditions
- INNER JOIN across more than two tables
- Filtering joined data using WHERE

INDIAN DATA CLUB



Inner join

INNER JOIN returns **only the rows with matching values in both tables** based on the specified condition.

Syntax :-

```
SELECT table1.column1, table2.column2
FROM table1
INNER JOIN table2
ON table1.common_column = table2.common_column;
```

Use ON when:

- The **column names are different** in the two tables.
- You need more flexibility in the join condition.

Syntax :-

```
SELECT table1.column1, table2.column2
FROM table1
INNER JOIN table2
USING(common_column);
```

Use USING when:

- **The column name is the same in both tables.**
- **You want a cleaner, simpler join syntax.**

INDIAN DATA CLUB

❖ *Important Points About INNER JOIN*

Matching Happens Using ON Clause
The ON condition tells MySQL **how** the tables are related (usually via primary/foreign keys).
Example: table1.id = table2.id

Does Not Return Non-Matching Rows
If a row exists in one table but **not** the other, it's **excluded** from the result.

You Can Join Multiple Tables
You can chain multiple INNER JOINs to combine 3 or more related tables.

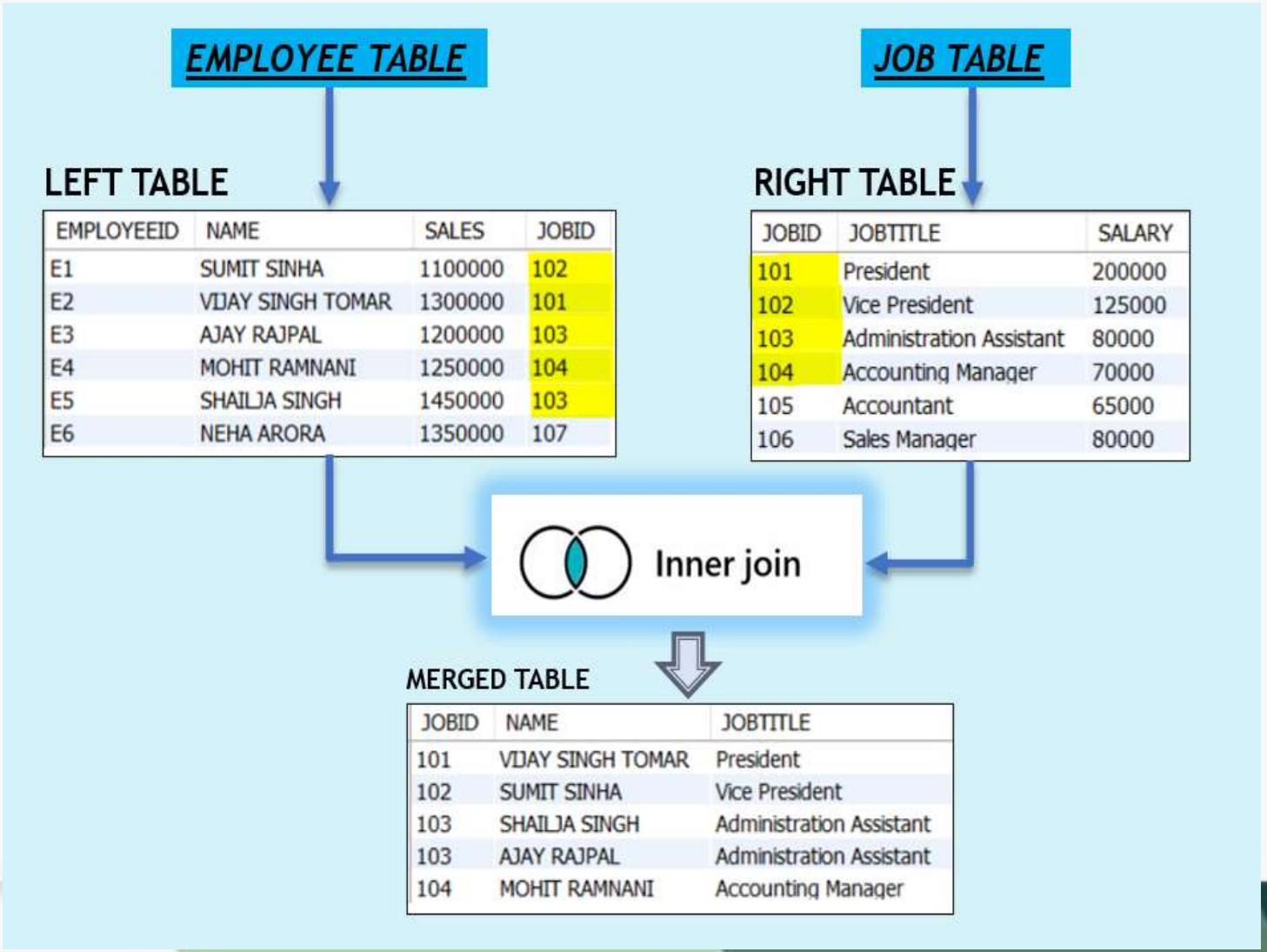
Can Include WHERE, ORDER BY, etc
You can add filtering and sorting

Works with Aggregates
You can use GROUP BY, along with aggregation functions like COUNT(), AVG(), SUM(), etc., **after using a JOIN** to summarize the joined data.

INDIAN DATA CLUB

❖ These are our EMPLOYEE and JOB tables, which we are going to use perform the INNER JOIN

RESULT :-



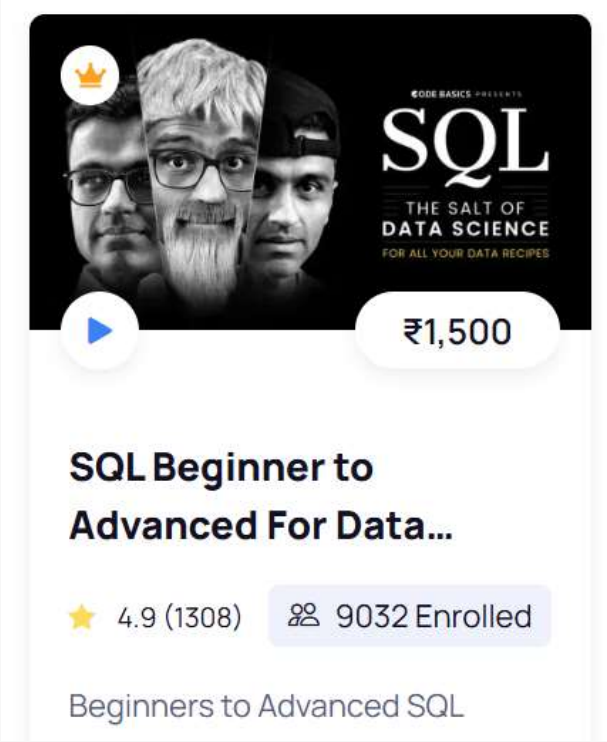
EMPLOYEEID	NAME	SALES	JOBID
E1	SUMIT SINHA	1100000	102
E2	VIJAY SINGH TOMAR	1300000	101
E3	AJAY RAJPAL	1200000	103
E4	MOHIT RAMNANI	1250000	104
E5	SHAILJA SINGH	1450000	103
E6	NEHA ARORA	1350000	107

JOBID	JOBTITLE	SALARY
101	President	200000
102	Vice President	125000
103	Administration Assistant	80000
104	Accounting Manager	70000
105	Accountant	65000
106	Sales Manager	80000

JOBID	NAME	JOBTITLE
101	VIJAY SINGH TOMAR	President
102	SUMIT SINHA	Vice President
103	SHAILJA SINGH	Administration Assistant
103	AJAY RAJPAL	Administration Assistant
104	MOHIT RAMNANI	Accounting Manager

INDIAN DATA CLUB

Resources:



Data with Baraa **SQL INNER JOIN - SQL Tutorial #22**

Amigoscode **PostgreSQL: Inner Joins | Course | 2019**

INDIAN DATA CLUB

Practice Questions:

```sql
-- Join patients and staff based on their common service field
-- (show patient and staff who work in same service).

SELECT *
FROM patients p
JOIN staff s
USING(service)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

service	patient_id	name	age	arrival_date	departure_date	satisfaction	staff_id	staff_name	role
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-fdc1ae8e	Cameron Parker	nurse
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-e983db1d	Kenneth Scott	nursing_assistant
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-e573be71	Shannon Walker	nurse
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-e328da72	Anna Henderson	nurse
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-ddd99f9e	Victor Baker	nursing_assistant
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-dd7b8861	Kelly Donovan	doctor
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-d77b1013	Christopher Bass	nurse
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-d7371c06	Shirley Suarez	nurse
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-cf9225db	Crystal Johnson	nurse
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-cb0c3c50	Lisa Barnes	doctor
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-cacb7290	Matthew Mcmillan	nurse
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-c479af96	Rebecca Valencia	nursing_assistant
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-b00ed3e2	Larry Dixon	nursing_assistant
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-abfdc900	Garrett Lin	nurse
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	STF-ab59db32	Paul Jones	doctor

- **SELECT *** → retrieves all columns from the joined results.
- **FROM patients p** → selects data from the patients table and assigns it the alias **p**.
- **JOIN staff s** → joins the staff table with the patients table and assigns it the alias **s**.
- **USING(service)** → matches rows where both tables have the **same service** value

INDIAN DATA CLUB

Practice Questions:

```sql
-- Join services_weekly with staff to show weekly service data with staff information.
SELECT *
FROM services_weekly sw
JOIN staff s
USING(service)
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

service	week	month	available_beds	patients_request	patients_admitted	patients_refused	patient_satisfaction	staff_morale	event	st
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST
emergency	1	1	32	76	32	44	67	70	none	ST

Result Grid | Form Editor | Field Types | Query Stats

- **SELECT *** → returns all columns from both tables.
- **FROM services_weekly sw** → selects weekly service data
- **JOIN staff s** → connects the staff table to this data.
- **USING(service)** → matches rows where **both tables have the same service**

Practice Questions:

```sql
-- Create a report showing patient information along with staff assigned to their service
SELECT *
FROM patients p
JOIN services_weekly sw
USING(service)
JOIN staff s
USING(service);
```

service	patient_id	name	age	arrival_date	departure_date	satisfaction	week	month	available_beds	patients_request	patients_a
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10
ICU	PAT-003ce690	Larry Dixon	29	2025-01-19	2025-01-21	60	20	5	10	14	10

- **SELECT *** → gets all columns from all tables.
- **FROM patients p** → starts with patient details.
- **JOIN services_weekly sw USING(service)** → adds weekly service data for the same service.
- **JOIN staff s USING(service)** → adds staff working in that same service.

INDIAN DATA CLUB

Daily Challenge:

```sql
-- Create a comprehensive report  showing patient_id, patient name, age, service,
-- and the total number of staff members available in their service.
-- Only include patients from services that have more than 5 staff members.
-- Order by number of staff descending, then by patient name.
SELECT
    p.patient_id,
    p.name AS patient_name,
    p.age,
    p.service,
    COUNT(s.staff_id) AS no_of_staff
FROM patients p
JOIN staff s USING(service)
GROUP BY p.patient_id, p.name, p.age, p.service
HAVING COUNT(s.staff_id) > 5
ORDER BY no_of_staff DESC, patient_name;
```

patient_id	patient_name	age	service	no_of_staff
PAT-1cc7c1b5	Adam Vaughan	2	ICU	48
PAT-bf1b13b6	Alejandro Deleon	14	ICU	48
PAT-d78122d6	Alexa Buck	73	ICU	48
PAT-f316dd98	Alexander Collins	17	ICU	48
PAT-615c986b	Alexandra Dominquez	80	ICU	48
PAT-ddd92df7	Alison Brown	50	ICU	48
PAT-c012faab	Allison Hickman	6	ICU	48
PAT-d4fc50d8	Allison Spencer	84	ICU	48
PAT-4275a12b	Alyssa Day	69	ICU	48
PAT-c1717ebc	Alyssa Haynes	82	ICU	48
PAT-12523d2c	Amanda Sullivan MD	65	ICU	48
PAT-91fbb614	Amy Kelley	79	ICU	48
PAT-20043fb2	Amy Miller	30	ICU	48
PAT-994866af	Andrea Hubbard	41	ICU	48
PAT-333f8368	Andrew Reynolds	23	ICU	48
PAT-301d1f28	Angel Gordon	85	ICU	48
PAT-5044053e	Angel Perry	85	ICU	48
PAT-7aeac4f3	Angelica Parker	27	ICU	48
PAT-8718ccfc	Anna Estes	42	ICU	48
PAT-5cc7a8f3	Anthony Armstrong	47	ICU	48

- **SELECT** → picks patient details and counts how many staff work in the same service.
- **JOIN staff USING(service)** → connects each patient with all staff from their service.
- **GROUP BY patient columns** → produces one row per patient.
- **COUNT(s.staff_id)** → gives total staff in that service.
- **HAVING COUNT > 5** → keeps only patients from services that have more than 5 staff members.
- **ORDER BY no_of_staff DESC, patient_name** → sorts by staff count first, then patient name.