MySQL Views

A **view** in MySQL is a **virtual table** based on the result of a `SELECT` query. It does not store data itself — it always reflects the **current data** in the base tables.

Views are useful when:

- You want to simplify complex queries
- You want to reuse logic
- You want to hide certain columns from users
- You want a "live snapshot" of filtered data

Creating a View

Suppose we want a view that lists all users earning more than ₹70,000.

```
CREATE VIEW high_salary_users AS
SELECT id, name, salary
FROM users
WHERE salary > 70000;
```

Querying the View

```
SELECT * FROM high_salary_users;
```

This will return all users from the `users` table where salary is above ₹70,000.

Demonstrating That a View is Always Up-To-Date

Let's see what happens when the underlying data changes.

Step 1: View before update

```
SELECT * FROM high_salary_users;
```

Output:

id	name	salary
2	Sneha	75000
5	Fatima	80000

Step 2: Update a user's salary

```
UPDATE users
SET salary = 72000
WHERE name = 'Raj';
```

Step 3: Query the view again

```
SELECT * FROM high_salary_users;
```

New Output:

id	name	salary
2	Sneha	75000
5	Fatima	80000
3	Raj	72000

> Notice how **Raj** is now included in the view — without updating the view itself. That's because views **always reflect live data** from the original table.

Dropping a View

To remove a view:

```
DROP VIEW high_salary_users;
```

Summary

- Views act like **saved SELECT queries**
- Views are **not duplicated data**
- Changes to base tables are **reflected automatically**
- Great for simplifying complex queries or creating filtered access