

DAY-23

VIEWS IN SQL



SHVETA MAINI





What are Views?

- A view in SQL is essentially a virtual table that represents the result of a query.
- A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.
- It is not a physical table in the database, but rather a stored SQL query that can be referenced like a table in subsequent queries.



SYNTAX TO CREATE VIEWS:-

```
CREATE VIEW view_name AS  
SELECT column1, column2, ...  
FROM table_name  
WHERE condition;
```



EXPLANATION OF SYNTAX:-

- **CREATE :-** This keyword tells the SQL database system that you are creating a new object (in this case, a view).
- **VIEW:-** The VIEW keyword indicates that you are defining a view, not a table or other database object.



EXPLANATION OF SYNTAX:-

- **view_name :-** This is the name of the view you are creating. The view will be stored in the database under this name. Once created, you can reference this view in SQL queries just like a regular table.



EXPLANATION OF SYNTAX:-

- **AS:-** This keyword separates the view definition (the SELECT query) from the CREATE VIEW statement. After AS, you specify the SQL query that defines the view.
- **SELECT:-** The SELECT statement inside the view defines what data will be returned when you query the view.



EXPLANATION OF SYNTAX:-

- **column1,column2,..:-** These are the columns that you want to display in the view.
- **FROM :-** This keyword specifies which table (or tables) the data will come from. A view can be based on a single table, multiple tables, or even subqueries.



EXPLANATION OF SYNTAX:-

- **table_name :-** This is the name of the table from which the view will fetch data.
- **WHERE condition :-** This optional clause allows you to filter the data retrieved by the view based on a specific condition.



SYNTAX FOR DROPPING A VIEW :-

If you no longer need a view, you can remove it using the **DROP VIEW** statement.

```
DROP VIEW view_name;
```



SYNTAX FOR UPDATING A VIEW :-

If you need to modify a view (such as changing the columns or query logic), you can use the **CREATE OR REPLACE VIEW** statement.

```
CREATE OR REPLACE VIEW view_name AS  
SELECT column1, column2, ...  
FROM table_name  
WHERE condition;
```



BENIFITS OF USING VIEWS:-

- 1.) **Simplicity:-** Views can simplify complex queries. Instead of writing complex queries repeatedly, you can encapsulate them in views and then query the views directly.



BENIFITS OF USING VIEWS:-

2.) Data Security & Access Control:-

- **Views can restrict access to specific columns or rows, which is particularly useful when sensitive information is stored in tables.**
- **You can create views to only show necessary data to specific users while hiding sensitive columns.**



BENIFITS OF USING VIEWS:-

3.) **Reusability:-** Once a view is defined, you can reuse it in multiple queries, which can make your code cleaner and easier to maintain.



BENIFITS OF USING VIEWS:-

4.) Logical Data Independence:- Views abstract the underlying database schema, so if the structure of the base tables changes (like column names or data types), you can update the view definition instead of rewriting all queries dependent on the original tables.



BENIFITS OF USING VIEWS:-

5.) Data Aggregation:- Views can aggregate data (e.g., SUM, AVG, COUNT) to provide summarized data without modifying the original data. This is often used in reports or dashboards to present high-level information

**I HOPE YOU FIND
THIS POST
HELPFUL!**

**DO HIT THE LIKE
BUTTON AND
SHARE IT**

THANK YOU !!



SHVETA MAINI

