

## VIEWS in SQL

In SQL, a view is a virtual table based on the result-set of an SQL statement. 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. A view is actually a composition of a table in the form of a predefined SQL query. A view can contain all rows of a table or select rows from a table. A view can be created from one or many tables which depends on the written SQL query to create a view.

### Advantages of Views:

1. **Security:** Each user can be given permission to access the database only through a small set of views that contain the specific data the user is authorized to see, thus restricting the user's access to stored data
2. **Query Simplicity:** A view can draw data from several different tables and present it as a single table, turning multi-table queries into single-table queries against the view.
3. **Structural simplicity:** Views can give a user a "personalized" view of the database structure, presenting the database as a set of virtual tables that make sense for that user.
4. **Consistency:** A view can present a consistent, unchanged image of the structure of the database, even if the underlying source tables are split, restructured, or renamed.
5. **Data Integrity:** If data is accessed and entered through a view, the DBMS can automatically check the data to ensure that it meets the specified integrity constraints.
6. **Logical data independence.:** View can make the application and database tables to a certain extent independent. If there is no view, the application must be based on a table. With the view, the program can be established in view of above, to view the program with a database table to be separated.

### Creating Views

Database views are created using CREATE VIEWS Statement. The basic syntax of CREATE VIEW statement as:

```
CREATE VIEW view_name as select column_1,column_2 from table_name
where condition;
```

Consider the example of "Demo" database.

#### Creating view

```
CREATE VIEW person_details as select id, name, address from person;
```

## **The WITH CHECK OPTION**

The WITH CHECK OPTION is a CREATE VIEW statement option. The purpose of the WITH CHECK OPTION is to ensure that all UPDATE and INSERTs satisfy the condition(s) in the view definition.

If they do not satisfy the condition(s), the UPDATE or INSERT returns an error.

```
create view person_view as select name, age, address from person where age is not null with check option;
```

## **Updating Views**

A view can be update under the certain condition which are as below:

- The SELECT clause may not contain the keyword DISTINCT.
- The SELECT clause may not contain summary functions.
- The SELECT clause may not contain set functions.
- The SELECT clause may not contain set operators.
- The SELECT clause may not contain an ORDER BY clause.
- The FROM clause may not contain multiple tables.
- The WHERE clause may not contain subqueries.
- The query may not contain GROUP BY or HAVING.
- Calculated columns may not be updated.
- All NOT NULL columns from the base table must be included in the view in order for the INSERT query to function.

```
UPDATE person_details set address='KTM' where id=10;
```

Note: Update in view is also update in your actual table.

## **Inserting Rows into views:**

We can insert rows in the view also and it automatically update in the actual table.

```
insert into person_details values (21, 'Deepa', 'Bhaktapur');
```

Note: values cannot insert in this views because we have not included not null column in the person\_detail view.

### **Deleting Rows from view:**

Rows of data can be deleted from a view. The same rules that apply to the UPDATE and INSERT commands apply to the DELETE command.

```
delete from person_details where id=20;
```

### **Dropping view:**

```
drop view person_view;
```