Views – Create, Alter, Drop views

- A **view** is a virtual or logical table that allows viewing or manipulating the parts of the tables.
- A view is derived from one or more tables known as **base tables**.
- A view looks like and works similarly to normal tables. But, unlike tables, a view does not have **storage space** to store data.
- A view is created by a query, i.e. a **SELECT** statement which uses base tables.
- Data for views are extracted from these base tables based on specified query.
- A view is **dynamic** and always **reflects the current data of the base tables**.
- Only definition of view is stored in the database.
- When a view is referenced in SQL statement following steps will be followed:
    - Its definition is retrieved from database.
    - The base tables are opened.
    - A query, specified in definition is executed. o A view is created on top of the base tables.
- When any operation is performed on view, it is actually performed on the base table.

- For example, any **SELECT** operation on view displays data from the base table. In a similar way, **INSERT, UPDATE, DELETE** operations modify the contents of the base table

Types of Views

- View can be classified into two categories based on which type of operations they allow:

1) **Read-only View:**

- Allows only SELECT operation, this means user can only view data.
- No INSERT, UPDATE or DELETE operations are allowed. This means contents of base table cannot be modified.

2) **Updateable View:**

- Allows **SELECT** as well as **INSERT, UPDATE** and **DELETE** operations. This means contents of the base tables can be displayed as well as modified.

❖ Creating a View

- A view can be created using syntax as given below:

Syntax:

CREATE [OR REPLACE] VIEW viewName
As SELECT … ….
[WITH READ ONLY];

- This statement creates a view based on query specified in **SELECT** statement.

- **OR REPLACE** option re-creates the view if it is already existing maintaining the privileges granted to view that is given by view Name.
- **WITH READ ONLY** option creates **read-only views**. If this option is not provided then **by default updatable views** are created.
- The **SELECT** statement can include **WHERE, ORDER BY, GROUP BY** clauses if required.
- A view can be created using single base table as well as multiple base tables using joins.
- The following examples explain how to create views and how to use them in SQL statements. Consider tables – Account and Branch as given in below figure:

Account

Ano	Balance	B_Name
A01	1000	Rjt
A02	4000	Ahmd
A03	3000	Srt

Branch

B_Name	B_Address
Rjt	Kalawad Road, Rajkot
Ahmd	Elisbridge,Ahmedabad
Srt	Mota Bazaar, Surat

Example:

```
CREATE VIEW Acc_Rjt
AS SELECT * FROM Account
WHERE B_Name = 'Rjt';
```

Output:

View created.

❖ Advantages of View

- View the data without storing the data into the object.
- Restricts the view of a table. i.e. can hide some of columns in the tables.
- Join two or more tables and show it as one object to user.
- Restricts the access of a table so that nobody can insert the rows into the table.
- There are two major advantages of views:
  - Flexible enforcement of security
  - Simplification of complex query
  -

❖ Disadvantages of Views

- Cannot use DML operations on view.
- When table is dropped view becomes inactive.
- View is an object, so it occupies space.

❖ Destroying a View

- The DROP VIEW command drops the specified view.
- The base table will not be affected if a view is destroyed.
- If a base table is dropped or column included in view are altered then view will not be valid further.
- Oracle issues an error message while using such in-valid views.

Syntax:

DROP VIEW viewName;

Example:

DROP VIEW Acc_Branch;

Output: **View Dropped.**