VIEW

# VIEW

- A view is like a virtual table. It takes the output of a query and treats it like a table. We can **create, update, and drop a VIEW**.

- A view can be based on one or more tables or other views. These tables/views are called base tables.

- A view contains no data. All the data it shows comes from the base tables.

- A view can provide an additional level of **table security** by restricting access to a set of rows or columns of a table.

- A view lets you **change the data** you can access, applying operators, aggregation functions, filters etc. on the base table.

# SAMPLE RELATION - CUSTOMER

| CUSTOMER_ID | NAME | CURRENT_BALANCE | PHONE_NO | CITY |
|---|---|---|---|---|
| 019523012 | George | 15000 | 9865984759 | Coimbatore |
| 019523013 | Antony | 18500 | 9659845789 | Chennai |
| 019523045 | Dharani | 32650 | 9658523214 | Chennai |
| 019523025 | Suresh | 62555 | 8965985478 | Salem |
| 019586489 | Vanitha | 26590 | 7854896589 | Trichy |

# VIEW

TO CREATE A VIEW:

## SYNTAX:

CREATE VIEW view_name AS SELECT column_name

FROM table_name [WHERE conditions];

## EXAMPLE 1:

```
SQL> CREATE VIEW VIEW_1 AS (Select *from Customer);

View created.

SQL> SELECT *FROM VIEW_1;

CUSTOMER_ID NAME          CURRENT_BALANCE    PHONE_NO CITY
----------- ----------    --------------- ----------- -----------
   19523012 George                  15000  9865984759 Coimbatore
   19523013 Antony                  18500  9659845789 Chennai
   19523045 Dharani                 32650  9658523214 Chennai
   19523025 Suresh                  62555  8965985478 Salem
   19586489 Vanitha                 26590  7854896589 Trichy
```

# VIEW

## EXAMPLE 2:

```
SQL> CREATE VIEW VIEW_2 AS (Select CUSTOMER_ID,NAME from Customer);

View created.

SQL> SELECT *FROM VIEW_2;

CUSTOMER_ID NAME
_____ _____
   19523012 George
   19523013 Antony
   19523045 Dharani
   19523025 Suresh
   19586489 Vanitha
```

## EXAMPLE 3:

```
SQL> CREATE VIEW VIEW_3 AS (Select *from Customer WHERE CURRENT_BALANCE>30000);

View created.

SQL> SELECT *FROM VIEW_3;

CUSTOMER_ID NAME          CURRENT_BALANCE    PHONE_NO CITY
_____ _____   _____  _____ _____
   19523045 Dharani                 32650 9658523214 Chennai
   19523025 Suresh                  62555 8965985478 Salem
```

# VIEW

TO INSERT INTO VIEW TABLE:

**SYNTAX:**

INSERT INTO VIEW_NAME VALUES('Value1',…, 'ValueN');

**EXAMPLE:**

```
SQL> INSERT INTO VIEW_1 VALUES(19523056,'Sita',56000,9658987412,'Tirunelveli');

1 row created.
```

# VIEW

TO DISPLAY DATA IN VIEW TABLE:

```
SQL> SELECT *FROM VIEW_1;

CUSTOMER_ID NAME         CURRENT_BALANCE    PHONE_NO CITY
----------- ----------   --------------- ----------- ----------------
   19523012 George                 15000  9865984759 Coimbatore
   19523013 Antony                 18500  9659845789 Chennai
   19523045 Dharani                32650  9658523214 Chennai
   19523025 Suresh                 62555  8965985478 Salem
   19586489 Vanitha                26590  7854896589 Trichy
   19523056 Sita                   56000  9658987412 Tirunelveli

6 rows selected.
```

TO DISPLAY DATA IN BASE TABLE:

```
SQL> SELECT *FROM CUSTOMER;

CUSTOMER_ID NAME         CURRENT_BALANCE    PHONE_NO CITY
----------- ----------   --------------- ----------- ----------------
   19523012 George                 15000  9865984759 Coimbatore
   19523013 Antony                 18500  9659845789 Chennai
   19523045 Dharani                32650  9658523214 Chennai
   19523025 Suresh                 62555  8965985478 Salem
   19586489 Vanitha                26590  7854896589 Trichy
   19523056 Sita                   56000  9658987412 Tirunelveli

6 rows selected.
```

**Note:** The record 'Sita' is displayed in both View and Base Tables. i.e., the **CHANGES** made in **VIEW** will reflect in **BASE TABLE** also.

# VIEW

TO DELETE FROM VIEW TABLE:

**SYNTAX:**

DELETE FROM view_name [WHERE conditions];

**EXAMPLE:**

```
SQL> DELETE FROM VIEW_1 where Name='George';

1 row deleted.
```

# VIEW

TO DISPLAY DATA IN VIEW TABLE:

```
SQL> SELECT *FROM VIEW_1;

CUSTOMER_ID NAME        CURRENT_BALANCE   PHONE_NO CITY
----------- ----------  --------------- ---------- ----------
   19523013 Antony                18500 9659845789 Chennai
   19523045 Dharani               32650 9658523214 Chennai
   19523025 Suresh                62555 8965985478 Salem
   19586489 Vanitha               26590 7854896589 Trichy
   19523056 Sita                  56000 9658987412 Tirunelveli
```

TO DISPLAY DATA IN BASE TABLE:

```
SQL> SELECT *FROM CUSTOMER;

CUSTOMER_ID NAME        CURRENT_BALANCE   PHONE_NO CITY
----------- ----------  --------------- ---------- ----------
   19523013 Antony                18500 9659845789 Chennai
   19523045 Dharani               32650 9658523214 Chennai
   19523025 Suresh                62555 8965985478 Salem
   19586489 Vanitha               26590 7854896589 Trichy
   19523056 Sita                  56000 9658987412 Tirunelveli
```

**Note:** The record 'George' is not displayed in both View and Base Tables. i.e., the **CHANGES** made in **VIEW** will reflect in **BASE TABLE** also.

# VIEW

The definition of an Oracle VIEW can be modified without dropping it by using the Oracle CREATE OR REPLACE VIEW Statement.

**Syntax:**

**CREATE OR REPLACE VIEW** view_name AS

SELECT columns FROM table [WHERE conditions];

**EXAMPLE:**

```
SQL> CREATE OR REPLACE VIEW VIEW_2 AS SELECT *from CUSTOMER where CITY='Salem';

View created.

SQL> SELECT *FROM VIEW_2;

CUSTOMER_ID NAME         CURRENT_BALANCE   PHONE_NO CITY
----------- ---------    ---------------   -------- ----------
   19523025 Suresh                 62555 8965985478 Salem
```

# VIEW

TO DROP A VIEW:

**SYNTAX:**

DROP VIEW view_name;

**EXAMPLE:**

```
SQL> DROP VIEW VIEW_2;

View dropped.

SQL> SELECT *FROM VIEW_2;
SELECT *FROM VIEW_2
              *
ERROR at line 1:
ORA-00942: table or view does not exist
```

**Note:** After dropping the view table, the data inside the base table remains the same.