Introduction To Views ====>
==========

* A view is a virtual table that consists of columns and rows, but it is only the SELECT statement that is stored, not a physical table with data.

* A view's SELECT query may reference just one or multiple tables, called base tables.

* The base tables are typically actual tables or other views.

Advantage Of Views ===>
==========

* Views are beneficial in many ways to the programmer.

* The most popular advantages of Views are :
- Simplifying data retrieval.
- Maintaining logical data independence.
- Implementing data security.

Types Of Views ===>
==========

* Oracle provides us the support for two types of Views called as SINGLE VIEW and JOIN VIEW.

* A view created using a ONE BASE TABLE is called as a SINGLE VIEW while a view created using MULTIPLE BASE TABLE , for example a join query, is called a JOIN VIEW.

* Syntax Of Single View :-

CREATE VIEW view_name As (<select query>);
* Example: CREATE VIEW mystaff As (Select ename, deptno from emp);

Important Point: ==>
==========

* Creating a view is a special privilege which must be assigned to our user by the system user. So before executing the CREATE VIEW command we have to execute the following command from system user:-
# Grant Create View To OracleBatch;

1. Connect with system
- connect system/oracle

2. Grand permision foe view
- grant create view to prashant;

3. Connect with user
- connect prashant/prashant

4. Create a desire view
- create view mystaff as (select empno, ename from emp);

* The MYSTAFF view hides a number of columns that exist in the EMP table.

* We do not see the EMPNO and SAL columns and this is how views provide data security

* UPDATING data in the view also updates data in the base table.

* However only those columns can be updated which are a part of the view.

* We can INSERT data in the table through a view , but the columns which are not a part of view will receive NULL and if the table has a NOT NULL or PRIMARY KEY constraint enabled on that column then insertion will fail.

* DELETING data from a view actually deletes data from the base table.

* Changes made to the table are reflected in the view.

Clauses Used With Create View Command ===>
==========

OR REPLACE: ==>
==========

* To overwrite an existing view Oracle provides us OR REPLACE clause which instructs Oracle to replace an existing view .

* EXAMPLE: CREATE OR REPLACE VIEW mystaff As (Select * from emp where sal >15000);

WITH CHECK OPTION: ==>
==========

* The WITH CHECK OPTION clause protects the view from any changes to the underlying table that would produce rows which are not included in the defining query.

* EXAMPLE: CREATE OR REPLACE VIEW mystaff As (Select * from emp where sal >15000) WITH CHECK OPTION;

- Insert into mystaff values(504,'Gagan',20000,101); // OK
- Insert into mystaff values(505,'Kiran',10000,102); // ERROR

WITH READ ONLY: ==>
==========

* The WITH READ ONLY clause prevents the underlying tables from changes through the view.

* EXAMPLE: CREATE OR REPLACE VIEW mystaff As (Select * from emp where sal >15000) WITH READ ONLY;

- Insert into mystaff values(504,'Gagan',20000,101); // ERROR
- Update mystaff set sal=sal*0.1; // ERROR

FORCE: ==>
==========

* Usually, we create a new view based on existing tables. However, sometimes, we may want to create a view based on the tables that we will create later. In these cases, we can use the FORCE option.

* EXAMPLE: CREATE FORCEVIEW mybooks As (Select * from allbooks);

Restrictions On Single View ===>
==========

* If a NOT NULL column that does not have a DEFAULT clause is omitted from the view, then a row cannot be inserted into the base table using the view.

* If a view is defined with WITHCHECK OPTION, then a row cannot be inserted into, or updated in, the base table (using the view), if the view cannot select the row from the base table.

* If a view is defined with WITH READ ONLY, then no DML operations are allowed on the view.

* Views do not support DDL Commands.