|  |  |  |
| --- | --- | --- |
| **Ex. No. 9** | **VIEWS** | **Date :** |

**VIEWS**

* An Imaginary table contains no data and the tables upon which a view is based are called base tables.
* Logically represents subsets of data from one or more tables

# Advantages of view

* To restrict database access
* To make complex queries easy
* To allow data independence
* To present different views of the same data

# Syntax

**CREATE [OR REPLACE] VIEW *view* [*col1 alias*, *col2 alias,*...]**

**AS *subquery***

# [ WITH CHECK OPTION [ CONSTRAINT *constraint* ] ] [ WITH READ ONLY ]

* You embed a subquery within the CREATE VIEW statement.
* The subquery can contain complex SELECT syntax.
* The subquery cannot contain an ORDER BY clause.

**Q1)** Create a view *empv10* that contains *empno, ename, job* of the employees who work in *dept* 10. Also describe the structure of the view.

**SQL> create view *empv10* as**

**select *empno, ename, job* from *emp***

**where *deptno=10;***

**SQL> desc *empv10;***

**Q2)** Create a view with column aliases *empv30* that contains *empno, ename, sal* of the employees who work in *dept* 30. Also display the contents of the view.

# SQL >

**SQL> select \* from *empv30;***

# Rules for Performing DML Operations on a View

* You can perform DML operations on simple views.
* You **cannot remove** a row/ **modify** data/ **add** data if the view contains the following
  + Group functions
  + A GROUP BY clause
  + The DISTINCT keyword
* You cannot modify data in a view if it contains columns defined by expressions or it contains ROWNUM pseudo column
* You cannot add data if any NOT NULL columns in the base tables that are not selected by the view

**Q3)** Update the view *empv10* by increasing 10% salary of the employees who work as ‘CLERK’. Also confirm the modifications in *emp table.*

**SQL >**

**SQL >**

**Q4)** Modify the view *empv10* which contains the data *empno, ename, job, sal*. Add an alias for each column name.

# SQL > create or replace view *empv10*

## (employee\_no, employee\_name, job, salary) as

**select *empno, ename, job,sal* from *emp* where *deptno=10;***

**Q5)** Using *emp* table, **c**reate a view *pay* which contains *ename, monthly\_sal, annual\_sal, deptno.*

# SQL>

**Q6)** Create a view *dept\_stat* which contains *department no., department name, minimum salary, maximum salary, total salary.*

# SQL>

**With check option**

* You can ensure that DML on the view stays within the domain of the view by using the WITH CHECK OPTION.

**Q7)** Execute the following query and then try to delete the row with dept no 20. Now write in words that you understand

# SQL> create or replace view *empv20*

**as select \* from *emp* where *deptno = 20***

**with check option constraint *empv20\_ck;***

# Denying DML Operations

* You can ensure that no DML operations occur by adding the WITH READ ONLY option to your view definition.

**Q8)** Create a view *empv10* with all the details of employees who work in dept no. 10.

Also ensure that no DML operations can be done with the view.

# SQL>

**Q9**) Statement1 : Update the view will update data in original table.

Statement 2: update in main table will affect the created view or not?

State whether the above statements is True or False with explanation.

# Deleting Views

# Syntax

# DROP VIEW view\_name;

# Q10) Delete the view empv20.

# SQL>