Day 19

###Views

It is a logical table or virtual table .

Views are logical objects of database they will not have any

storage.

Table is used to store the data.

View is used to extract data from the table.

view is created over an sql query.

ex-->

create view view\_name as select \* from employ

main purpose of view:

security: By hiding the query used to generate the view.

ex-> If client want to see only expensive product

create view exp\_vw

as select \* from product where price>1000;

Types of views

1)simple views

It is created on a single table(base table) with one or more

columns

It is possible to perform DML operations on view.

>create view emp\_vw as select \* from emp1;

>select \* from emp\_vw

>drop view emp\_vw

points to remember

-If base table is drop , view becomes invalid

-Any chages occured on data of base table can be reflect in

view as well & vice versa.

-if base table column name is change--> view becomes invalid

(because you are trying to modify the table structure)

view does not store the data, but view will store the structure

of the query, thats why we cannot change column name ..

>alter table emp1 rename column salary to sal;

-If you alter the table structure it will not reflect in view.

we have to refresh the view to see the latest table structure

in view

Read only simple view

>create view emp\_vw1 as select \* from emp1 with read only;

note: cannot perform a DML operation on a read-only view

But changes in base table will reflect in view

2)Complex view

It is created on more than one table.(means involve joins)

DML operations are not allowed.

>create view dept1\_emp1\_vw as

select e.empid,e.ename,e.salary,d.deptno,d.dname from employ1 e

join dept1 d

on e.deptno=d.deptno;

>select \* from dept1\_emp1\_vw;

Note: any changes in base tables, reflect in view..

#Force view

A view can be created without having base table

Normally it is possible to create a view on an existing table.

But when force keyword is used then view can be created

even though there is no base table.

But view starts working after the creation of base table

>select \* from avdtable;

error-->table not found

>create force view avd\_table\_vw as select \* from avdtable;

3)materialized view

These view are also created on one or more tables of local

or remote database..

Materialised view maintains local copy of the remote table

generally.

Any changes occur on the remote database table we can used

refresh method to refresh it.

Time to time data can be refreshed using refresh method.

In general materialized view are read only, so no DML

operations are allowed on materialized view.

syntax-->

#on local table

>create materialized view emp1\_mv1 as select \* from emp1;

#on remote table

>create materialized view emp1\_mv1 as select \* from emp1@db\_link;

#Refresh method

In order to get latest data changes from remote database table

>exec dbms\_mview.refresh('EMP1\_MV1');

##### CTE (Common table expression)

it is nothing but temporary table

Repeated piece of code you can put into CTE so that no need

to write same piece of code again & again.

-it improves reusability

-it improves readability of code

-it is easy to debug the code(Finding the errors).

ex-

>with temp\_gen as (

select gender,marks,case when gender in ('male','men','boy','man') then 'M'

when gender in ('women','female','girl') then 'F'

end as new\_gender from gen)

select \* from temp\_gen;

ex-

with saleCTE as

(select productID,sum(quantity) as total\_sales from sale

group by productID)

select productnew.productname,saleCTE.total\_sales from productnew

left join saleCTE

on productnew.productID=saleCTE.productID