Experiment No 10

Objective: creation of views

The SQL VIEW is, in essence, a virtual table. It does not physically exist. Rather, it is created by a SQL query joining one or more tables.

Views can provide advantages over tables:

- Views can represent a subset of the data contained in a table; consequently, a view can limit the degree of exposure of the underlying tables to the outer world: a given user may have permission to query the view, while denied access to the rest of the base table.
- Views can join and simplify multiple tables into a single virtual table
- Views can act as aggregated tables, where the database engine aggregates data (sum, average etc.) and presents the calculated results as part of the data.
- Views can hide the complexity of data; for example a view could appear as Sales2000 or Sales2001, transparently partitioning the actual underlying table
- Views take very little space to store; the database contains only the definition of a view, not a copy of all the data which it presents
- Depending on the SQL engine used, views can provide extra security

CREATE VIEW

Use the CREATE VIEW statement to define a view, which is a logical table based on one or more tables or views. A view contains no data itself. The tables upon which a view is based are called base tables.

Create a view which lists out the bill_no, bill_date, cust_id, item_id, price, qty_sold, amount

SQL>create view cust as (select s.billno, s.billdate, c.custid, i. iitemid, i.price, s.qty from customer c,sale s item I where c.custid=s.custid and i.iemid=s.itemid);

view created.

SQL>select * from cust;

BILLNO	BILLDATE	CUSTID	ITEMID	PRICE	QTY
3432	12-JAN-06	3	3244	120	2
4424	20-FEB-06	1	3456	20	3
332	13-MAR-06	1	1234	5	2
2343	10-MAR	5	5001	10	1
1331	11-MAR-06	4	76776	350	4