**EXPERIMENT 7 ( part a)**

# Views

**Objective**

* Lear how to create and Execute different views

## Views

A database view is a stored query that returns data from one or more database tables.  The stored query, or view, is a virtual table.   Once you have defined a view, you can reference it just as you would any other table in a database.  Since the view is the result of a stored query, it does not contain a copy of the data itself.  Instead, it references the data in the underlying base tables.

**Advantages of Views:**

A view can provide additional security.  By creating a view and creating the necessarily privileges, you can ensure that the users are only able to retrieve and modify data that is exposed by that view.  Users will not be able to see or access data in the underlying tables that is not exposed by the view.

Views can reduce query complexities.  By creating and storing complex queries and exposing them in the form of a view, the data from the view can be extracted using much simpler queries.

Since a database view is a stored query, not a copy of the actual data, views consume very little space.

**Examples:**

Some examples of the ways views are used are:

* To combine data from multiple tables into a single virtual table that can be queried using basic statements.
* To partition a complex table into multiple virtual tables that are simpler to query. For example, if a database table contains sales data from the past 10 years, views can be created and represented using tables names such as SalesData2000 or SalesData2001.
* To aggregate data and perform calculations.  The view (stored query) can request the database engine to sum or average data in underlying tables.  These sums or averages can then be queried more easily.

**Creating a View**

Below is the general syntax for creating a View:

CREATE VIEW [View\_Name]

AS

[SELECT Statement]

For example:

CREATE VIEW v\_employeeNames

As

SELECT fname, lnameFROM employee

which will create a View with the name customerData that will only contain customerNmae.

**Get result from a View:**

This is similar to a Select statement:

select \* from **v\_**employeeNames

**Modify an existing View**

Alter VIEW **v\_**employeeNames

As

SELECT ssn, fname, lname FROM employee

**Drop a View**

DROP VIEW **v\_**employeeNames

**Exercise Questions:**

Using your own database (created in Lab3), write an **SQL statement along with output in the space provided** to answer each of the following queries:

1. Create a view called v\_EmployeeInfo that displays the employees’ name, salary and number of dependents s/he has. Please note 0 should be displayed if the employee has no dependent. The output should be sorted on employee names
2. Remove from the database the view called v\_EmployeeInfo