



VIEWS



VIEW

- A view is a virtual table
 - It is a query stored in the database
- A view is an object that is used to retrieve data from one or more tables or views
- A view can be used
 - To simplify, and customize the data seen by each user
 - To restrict the user from viewing certain column and rows
 - allowing users to access data through the view, without granting the users permissions to directly access the underlying base tables
 - Summarize data

CREATE VIEW

■ Syntax

```
CREATE [OR ALTER] VIEW viewName [ (column [ ,...n ] ) ]  
AS  
SELECT statement
```

- The SELECT statement can refer to one or more tables or other views
- Tables from different databases can be included
- The list of column names for the view is optional
 - If not specified, the column list derived from the SELECT statement will be used
- **OR ALTER**
 - Applies to SQL Server - starting with SQL Server 2016



CREATE VIEW - Examples

■ Example

```
CREATE VIEW DailySales
AS
SELECT  CustomerID, ProductID, Quantity * UnitPrice AS Amount
FROM    Sales
WHERE   SalesDate = GetDate()
```

■ Using the view

```
SELECT  CustomerID, ProductID, Amount
FROM    DailySales
```

■ Using the view

```
SELECT  *
FROM    DailySales
```



CREATE VIEW – Examples 2

■ Example

```
CREATE VIEW vwEmpNames (FirstName, LastName)  
AS  
SELECT    FName, LName  
FROM      Employee
```

■ Using the view

```
SELECT    FirstName, LastName  
FROM      vwEmpNames
```

■ Using the view

```
SELECT    *  
FROM      vwEmpNames
```



CREATE VIEW Statement

- The CREATE VIEW must be the first statement in a query batch
- The SELECT clauses in a view definition cannot include an ORDER BY clause, unless there is also a TOP clause in the select list of the SELECT statement
- The ORDER BY clause does not guarantee ordered results when the view is queried, unless ORDER BY is also specified in the query itself
- If a view depends on a table or view that was dropped
 - the Database Engine produces an error message when anyone tries to use the view
- If the new table or view structure changes
 - The view must be dropped and re-created



Updatable Views

- You can modify the data of an underlying base table through a view
- A view can be updated under certain conditions:
 - All updates must reference columns from only one base table
 - The SELECT clause may not contain the keyword DISTINCT, GROUP BY or HAVING, ORDER BY clause
 - The columns being modified are not affected by GROUP BY, HAVING, or DISTINCT clauses.
 - TOP is not used in the select statement of the view



Updatable Views - Example

■ Example

```
CREATE VIEW vwSalesEmployee
AS
SELECT [FName],[LName],[Salary], D.DepID
FROM Employee E JOIN Department D ON E.DepID = D.DepID
WHERE D.DepName = 'Sales'
GO
```

■ Using the view

```
select * from vwSalesEmployee
```

■ Updating by using the view

```
UPDATE vwSalesEmployee
SET      DepID = 3
WHERE  FName = 'Genet'
```



ALTER TABLE and Views

Question:

- What happens to your views when you add a new column to a table?
 - The NEW columns will not be accessible via the view
 - Even if your view is built using a **SELECT *** as its base statement
- Column names in views are resolved at the time the view is created
- You must either DROP and recreate the view or
 - Use an ALTER VIEW statement to rebuild it
 - Refresh the view's defn using **sp_refreshview**
EXEC sp_refreshview ***ViewName***