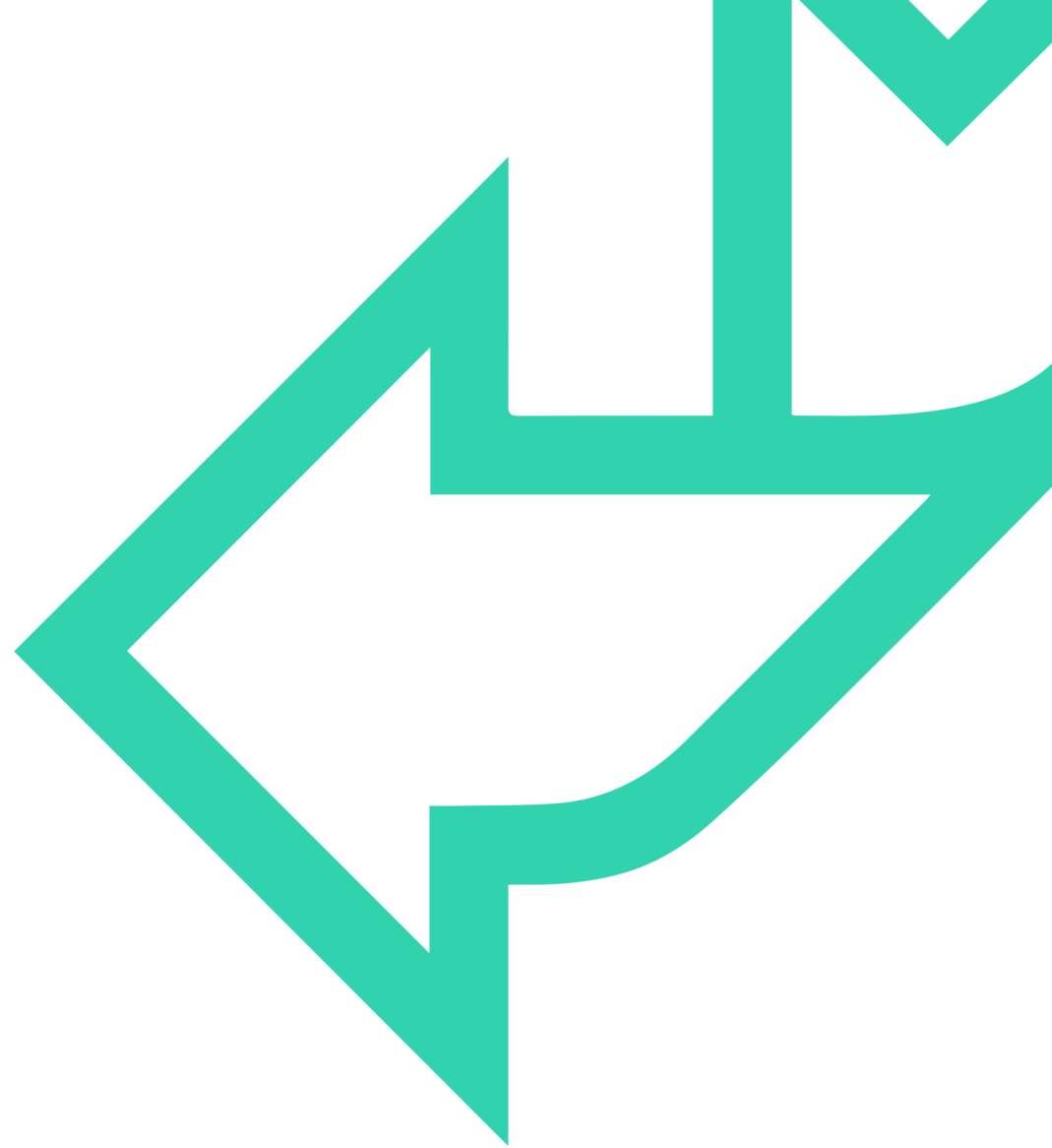




# SQL: Views





# VIEW

A **view** is a select query which has been pre-created and stored within the database.

- The view will not give any performance advantages over the original query.
- For data retrieval, a view can be used in the same way as an ordinary table.
- You can simplify the use of a difficult query for other users where they do not have to provide the full query again, but can refer to the view by name.
- As the views are called within select queries, by default views cannot include ORDER BY clause when they are created. This is to avoid an expensive sort operation. ORDER BY however can be applied to a view when it is used.



# CREATING AND USING VIEWS

```
CREATE VIEW company_sales
AS
SELECT      COMP.name AS 'Company_Name',
            CONT.name AS 'Contact_Name', S.*
FROM        company COMP JOIN contact CONT
            ON COMP.company_no =
            CONT.company_no
            JOIN sale S
            ON  S.company_no = CONT.company_no
            AND S.contact_code = CONT.contact_code
```

No embedded  
spaces

```
SELECT      *
FROM        company_sales
ORDER BY    emp_no
```



# CREATE / DROP VIEW

Column name list is optional, but any  
virtual columns must have a valid name

```
CREATE VIEW viewname(col1, alt_col2name, calc_col)
AS
SELECT  col1, col2, (col3 / col4 * 100) AS calc_col
FROM    table | view
```

Could define new  
column name here

```
DROP VIEW viewname
```

Also drops any privileges  
defined on it



# VIEWS AND DATA SECURITY

Name	ID	Dept	Ext
John	1	121	210
Sally	2	132	322
Peter	3	439	932
Kim	4	310	126
Bill	5	021	931

```
CREATE VIEW phonelist AS  
SELECT  name, id, dept, ext  
FROM    emp
```

**PhoneList**



Name	ID	Dept	Ext	Salary	DOB
John	1	121	210	12,000	01/02/56
Sally	2	132	322	12,500	27/03/50
Peter	3	439	932	20,000	12/09/60
Kim	4	310	126	20,000	28/02/60
Bill	5	021	931	15,000	11/04/55

**Emp**

In most systems there will be some data that should be generally available and some that should only be accessible by specific users. Views can be given their own access privileges, separate from those of the underlying tables.

In the above example, the privilege to select from the PhoneList view could be given to everyone and the access to the Emp table could be restricted to staff from the personnel department.



# VIEWS WITH CHECK OPTION

By adding the **WITH CHECK OPTION** to the view definition, the engine will ensure that data being added via the view is accessible via the same view.

```
CREATE VIEW dept3_staff AS  
SELECT      *  
FROM        salesperson  
WHERE       dept_no = 3  
WITH CHECK OPTION
```

- This will not allow entry of non-dept3 staff
- Any data inserted or updated via the view must satisfy the view's underlying **WHERE** clause



# VIEW RESTRICTIONS

**A view cannot be used for INSERT or UPDATE if it:**

- contains the keyword DISTINCT
- contains a sub-query, a GROUP BY or a HAVING clause

**A view cannot be used for INSERT if it:**

- omits any NOT NULL columns that do not have defaults
- contains calculated or aggregate columns

**Views based on joins typically only allow one underlying table at a time to be updated.**