



VISTAS



1.- DEFINICIÓN: Es una consulta que se almacena como un objeto en la BD, la cual puede ser utilizada posteriormente. En la práctica, una vista guarda el código fuente de cualquier consulta como un archivo, el cual se puede ejecutar como una consulta tipo SELECT.

La tabla que se genera al ejecutar una vista se llama “TABLA VIRTUAL”, pues los resultados que se muestran no necesariamente existen en una tabla real en la BD.

Las vistas permiten:

- Ocultar información: permitiendo el acceso a algunos datos y manteniendo oculto el resto de la información que no se incluye en la vista. El usuario solo puede consultar la vista.
- Simplificar la administración de los permisos de usuario: se pueden dar al usuario permisos para que solamente pueda acceder a los datos a través de vistas, en lugar de concederle permisos para acceder a ciertos campos, así se protegen las tablas base de cambios en su estructura.
- Mejorar el rendimiento: se puede evitar tipear instrucciones repetidamente almacenando en una vista el resultado de una consulta compleja que incluya información de varias tablas.
- No se puede usar los comandos UPDATE, DELETE o INSERT

2.- SINTAXIS CREACIÓN :

```
CREATE [or replace ] VIEW nombre_vista AS  
SELECT .....;
```

3.- EJECUCIÓN DE LA VISTA

=#\d

=#\x [on / off] -----> Activa o desactiva el despliegue
del contenido de una función almacenada

#\dv -----> Muestra el listado de las vistas

=#\d+ nombre_vista ---> Muestra campos + código fuente

=#\sv+ nombre_vista ---> Muestra solo el código

=# SELECT * FROM nombre_vista WHERE <Condición> ;

4.- EJEMPLOS

Ejemplo 01:

```
CREATE OR REPLACE VIEW vempleados01 AS
```

```
select * from templeados ;
```

=#\d

=#\dv

=#\d+ vempleados01 ---> Muestra campos + código fuente

=#\sv+ vempleados01 ---> Muestra solo el código

=#select * from vempleados01 ;

Ejemplo 02:

CREATE OR REPLACE VIEW vempleados02 AS

```
SELECT  M.PKcedemple      AS Cedula ,
        M.nomemple       AS Nombre ,
        M.FKcodcar       AS Codigo ,
        T.dcargo         AS Descripcion

FROM    tmempleados      AS M , tmcargos AS T
WHERE   (M.FKcodcar = T.PKcodcar) and (M.FKcods = 1)

ORDER BY M.PKcedemple::BIGINT ;
```

=#\d

=#\dv

=#\d+ **vempleados02** ---> Muestra campos + código fuente

=#\sv+ **vempleados02** ---> Muestra solo el código

=#select * from **vempleados02** ;

=#select * from **vempleados02**

where cedula between '3000' and '6000' ;

=#select * from **vempleados02** where codigo IN (2,4,6) ;

Ejemplo 03:

CREATE OR REPLACE VIEW vretardos01 AS

```
SELECT B.FKcodexcu      AS Codigo ,
       B.fecha          AS fecha ,
       C.dexcusa        AS Excusa ,
       B.PKnr           AS No ,
       B.FKcedemple     AS Cedula ,
       A.nomemple       AS Nombre
FROM   tmempleados AS A , tdretardos AS B , tmexcusas AS C
WHERE  (B.FKcedemple = A.PKcedemple) and
       (B.FKcodexcu = C.PKcodexcu)

ORDER BY Codigo, fecha ;
```

=#\d

=#\dv

=#\d+ vretardados01 ---> Muestra campos + código fuente

=#\sv+ vretardados01 ---> Muestra solo el código

=#select * from vretardos01 ;

**=#select * from vretardos01
where fecha between '2022-11-01' and '2022-11-30';**

**=#select * from vretardos01
where (cedula = '7000') and
(fecha between '2022-11-01' and '2022-11-30');**

Ejemplo 04:

to_char(Campo_Numérico, '999G999G999D99') -----> Formato Numérico

CREATE OR REPLACE VIEW vcargos01 AS

```
SELECT PKcodcar AS "Código",  
       Dcargos::varchar(25) AS "Descripción",  
       to_char( sueldo, '999G999G999D99' ) AS "Sueldo" ,  
       to_char( (sueldo*0.02), '999G999G999D99' ) AS "Bono" ,  
       to_char( (sueldo*0.05), '999G999G999D99' ) AS "Deducción" ,  
       to_char( (sueldo + (sueldo*0.02) - (sueldo*0.05)),  
               '999G999G999D99' ) AS "SNeto"  
FROM tmcargos  
  
ORDER BY PKcodcar ;
```

=#\d

=#\dv

=#\d+ **vcargos01** ---> Muestra campos + código fuente

=#\sv+ **vcargos01** ---> Muestra solo el código

=#select * from **vcargos01** ;

Nota: Las vistas no reciben parámetros.

5.- BORRAR UNA VISTA **=# DROP VIEW nombre_vista ;**

Fuentes:

1.- <https://www.delftstack.com/es/howto/postgres/introduction-to-views-in-postgresql/>

