



Invoice

Invoice_id
Customer_id
Order_id
Product_id
Date_time
Status
Total
Remark

UT9. OTROS OBJETOS DE LA BBDD. VISTAS

Módulo: BASES DE DATOS

Curso 2022/2023. 1º DAM

Ruth Lospitao Ruiz

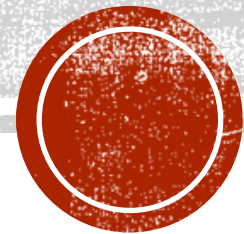


CONTENIDOS

- Introducción
- Crear o modificar una vista.
- Eliminar una vista.
- Renombrar una vista.
- Otras cuestiones de interés



INTRODUCCIÓN

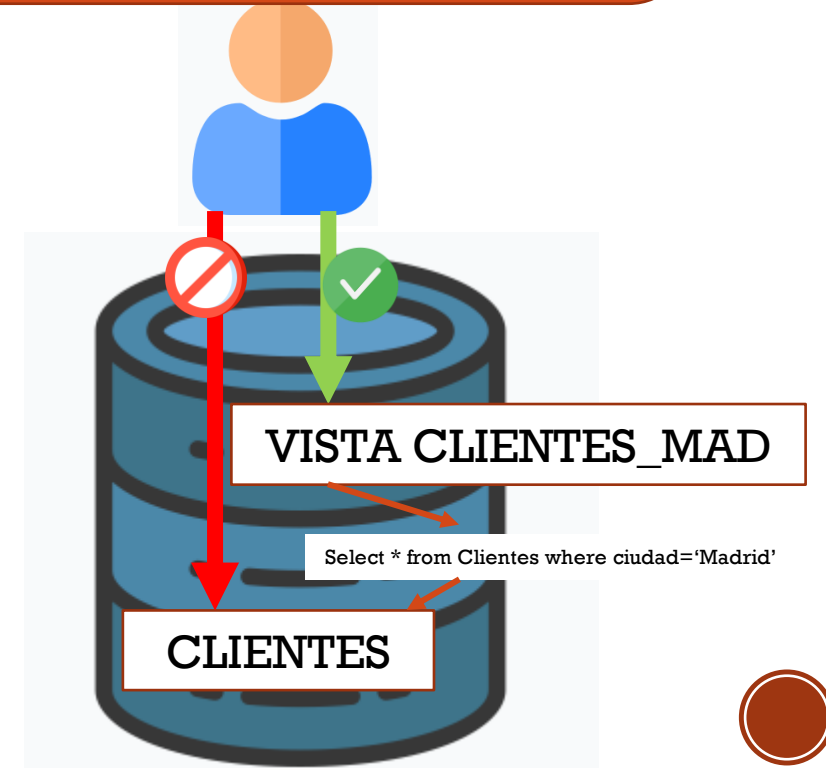


¿QUÉ SON LAS VISTAS?

Podemos definir una vista como una consulta almacenada en la base de datos que se utiliza como una tabla virtual.

Se define asociada a una o varias tablas y no almacena los datos sino que trabaja sobre los datos de las tablas sobre las que está definida, estando así en todo momento actualizada

- La diferencia de una vista con una consulta ejecutada directamente es que el tiempo de ejecución es menor.
- Mientras cada sentencia SQL enviada al SGDB tiene que pasar por un proceso de compilación, la vista es una consulta cuya definición ha sido almacenada previamente y que ya ha sido compilada
- También podría pasar que un usuario podría no tener acceso a la información de varias tablas y, sin embargo, sí a una vista que consulta dichas tablas



¿PARA QUÉ SIRVEN?

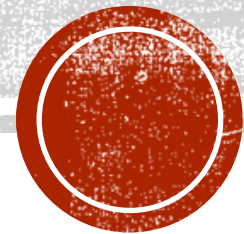
- Se trata de una perspectiva de la base de datos o ventana que permite a uno o varios usuarios ver solamente las filas y columnas necesarias para su trabajo

VENTAJAS

- **Seguridad y confidencialidad:** ya que la vista ocultará los datos confidenciales o aquellos para los que el usuario no tenga permiso.
- **Comodidad:** ya que solamente muestra los datos relevantes, permitiendo, incluso trabajar con agrupaciones de filas como si se tratase de una única fila o con composiciones de varias tablas como si se tratase de una única tabla.
- **Independencia** respecto a posibles cambios en los nombres de las columnas, de las tablas, etcétera.



CREACIÓN O MODIFICACIÓN



INTRODUCCIÓN

- Como hemos dicho son tablas virtuales resultado de una consulta realizadas sobre tablas ya existentes.
- Las vistas no ocupan espacio en la base de datos ya que lo único que se almacena es la definición de la vista. El gestor de la base de datos se encargará de comprobar los comandos SQL que hagan referencia a la vista, transformándolos en los comandos correspondientes referidos a las tablas originales, todo ello de forma transparente para el usuario



SINTAXIS

Nombre que tendrá la vista. Se lo damos nosotros

CREATE VIEW NombreVista

[(DefiniciónColumna [,DefiniciónColumna...])]

AS Código_SQL_Consulta;

- Permite especificar un nombre y su correspondiente tipo para cada columna de la vista.
- Si no se especifica, cada columna quedará con el nombre o el alias correspondiente y tipo asignado por la consulta

Código SQL con la consulta, cuyo resultado formará la vista

Documentación oficial: <https://dev.mysql.com/doc/refman/8.0/en/create-view.html>



EJEMPLO DE CREACIÓN Y USO

```
CREATE VIEW juanes AS
```

```
SELECT nombre, apellido1, apellido2, edad
```

```
FROM personas
```

```
WHERE nombre = 'JUAN';
```

Consultamos el contenido de la vista:

```
Select * from juanes;
```



Nombre	Apellido1	Apellido2	Edad
JUAN	PÉREZ	RUIZ	23
JUAN	MARTÍNEZ	SÁNCHEZ	56
JUAN	RAMÍREZ	MORENO	41



EJEMPLO

- Se pueden crear vistas para que los usuarios no expertos puedan acceder de forma fácil a la información, proporcionándoles, a través de una vista, información obtenida a través de una sentencia SQL compleja

```
CREATE VIEW VistaPedidos (CodigoPedido, Cliente, Total) AS  
    SELECT CodigoPedido, NombreCliente, SUM(Cantidad*PrecioUnidad)  
    FROM Clientes NATURAL JOIN Pedidos NATURAL JOIN DetallePedidos  
    GROUP BY CodigoPedido;
```

Recuerda: Natural Join es como el Inner Join y se emplea cuando el nombre de los campos son iguales
Consulta [aquí](#) para repasar



RESTRICCIONES

- No se puede usar la cláusula ORDER BY en la creación de una vista ya que las filas de una tabla no están ordenadas (la vista es una tabla virtual). No obstante, si se puede utilizar dicha cláusula a la hora de recuperar datos de la vista
- Es obligatorio especificar la lista de nombres de columnas de la vista o un alias cuando la consulta devuelve funciones de agrupamiento como SUM, COUNT, etcétera y posteriormente quiere hacerse referencia a ellas.
- Pueden utilizarse funciones de agrupación sobre columnas de vistas que se basan a su vez en funciones de agrupación lo que permite resolver los casos en los que un doble agrupamiento que no está permitido por el estándar. Así creamos una vista con una primera función de agrupación y sobre ella aplicamos la segunda función de agrupación, obteniendo el resultado deseado.



RENOMBRAR UNA VISTA

```
RENAME TABLE nombre_vista TO nuevo_nombre_vista;
```

Ejemplo:

```
RENAME TABLE VistaPedidos TO viewPedidos;
```

<https://dev.mysql.com/doc/refman/8.0/en/rename-table.html>



ELIMINACIÓN DE VISTAS



SINTAXIS

DROP VIEW [IF EXISTS] NombreVista;

EJEMPLO:

DROP VIEW IF EXISTS juanes;

<https://dev.mysql.com/doc/refman/8.0/en/drop-view.html>

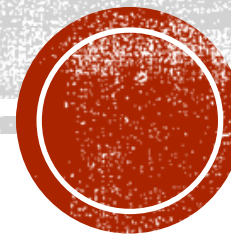


BORRADO DE TABLAS ASOCIADAS A UNA VISTA

- Si se borran las tablas a las vistas sobre las que están definidas las vistas la vista se queda invalida (no se borra su definición pero no está utilizable).



OTRAS CUESTIONES DE INTERÉS



CONSULTAR EL LISTADO DE VISTAS DISPONIBLES

SHOW FULL TABLES;

SHOW FULL TABLES WHERE table_type = 'VIEW';

CONSULTAR LA SENTENCIA QUE SE UTILIZÓ PARA CREAR UNA VISTA

SHOW CREATE VIEW view_name





Invoice

Invoice_id
Customer_id
Order_id
Product_id
Date_time
Status
Total
Remark

UT9. OTROS OBJETOS DE LA BBDD. VISTAS

Módulo: BASES DE DATOS

Curso 2022/2023. 1º DAM

Ruth Lospitao Ruiz

