

# VISTAS EN MySQL

# Entonces las vistas son!!

## ➤ Como tablas virtuales

En donde podemos visualizar los datos que consultemos. Si actualizamos los datos de una vista, actualizaremos la tabla, y si actualizamos la tabla estos cambios serán visibles desde la vista.

# Por que creamos vistas!!

- ✓ Seguridad: visualizamos solo una parte de la información que existe
- ✓ Comodidad: el modelo relacional no es cómodo para visualizar , tener una vista nos simplifica esta tarea.



# ¿Qué es una Vista de BD?

Una vista de base de datos es un resultado de una consulta SQL de una o varias tablas; también se le puede considerar una tabla virtual.

- Tienen la misma estructura que una tabla: filas y columnas.
- Sólo almacenan la definición, no los datos.
- Los datos que se recuperan mediante una consulta a una vista se presentarán igual que los de una tabla.
- Se especifican a través de una expresión de consulta (una sentencia SELECT) que la calcula y que puede realizarse sobre una o más tablas.
- La mayoría de los DBMS soportan la creación y manipulación de vistas.

# Aplicaciones:

- En general: información derivada de consultas complejas a la que se accede con frecuencia.
- Obtención de información derivada de la relación entre varias tablas y estadísticas.
- Como mecanismo de seguridad: creación de vistas con, únicamente, los atributos de las tablas a los cuales se desea permitir acceder a determinados usuarios.
- Para la creación de esquemas externos

# Creación de vistas

Para crear una vista debemos utilizar la sentencia **CREATE VIEW**, debiendo proporcionar un nombre a la vista y una sentencia SQL **SELECT** válida.

```
CREATE VIEW <nombre_vista>  
AS  
(<sentencia_select>);
```

# Creación de vistas

Ejemplo: Crear una vista sobre nuestra tabla alquileres, en la que se nos muestre el nombre y apellidos del cliente en lugar de su código.

```
CREATE VIEW vAlquileres  
AS  
(  
SELECT          nombre,  
                apellidos,  
                matricula  
FROM tAlquileres,  
        tClientes  
WHERE ( tAlquileres.codigo_cliente = tClientes.codigo )
```

# Sintaxis de CREATE VIEW

**CREATE** [**OR REPLACE**] [ALGORITHM = {UNDEFINED | MERGE | TEMPTABLE}] **VIEW**  
**nombre\_vista** [(columnas)] **AS** sentencia\_select [**WITH** [CASCADED | LOCAL] CHECK  
OPTION]

Esta sentencia crea una vista nueva o reemplaza una existente si se incluye la cláusula **OR REPLACE**.

Las tablas y las vistas comparten el mismo espacio de nombres en la base de datos, por eso, una base de datos **no puede contener una tabla y una vista con el mismo nombre**.



# Modificación de vistas

Si queremos, modificar la definición de nuestra vista podemos utilizar la sentencia **ALTER VIEW**, de forma muy parecida a como lo hacíamos con las tablas. En este caso queremos añadir los campos fx\_alquiler y fx\_devolucion a la vista.

```
ALTER VIEW vAlquileres
AS
(
SELECT nombre,
        apellidos,
        matricula,
        fx_alquiler,
        fx_devolucion
FROM tAlquileres,
tClientes
WHERE ( tAlquileres.codigo_cliente =
tClientes.codigo )
)
```

# Sintaxis de ALTER VIEW

**ALTER** [ALGORITHM = {UNDEFINED | MERGE | TEMPTABLE}] **VIEW** nombre\_vista  
[(columnas)] **AS** sentencia\_select [**WITH** [CASCADED | LOCAL] CHECK OPTION]

Esta sentencia modifica la definición de una vista existente.  
La sintaxis es semejante a la empleada en CREATE VIEW.

# Eliminación de vistas

Por último podemos eliminar la vista a través de la sentencia **DROP VIEW**. Para eliminar la vista que hemos creado anteriormente se utilizaría:

```
DROP VIEW vAlquileres;
```

# Sintaxis de DROP VIEW

***DROP VIEW*** [*IF EXISTS*] ***nombre\_vista*** [, *nombre\_vista*] ... [*RESTRICT* | *CASCADE*]

DROP VIEW elimina una o más vistas de la base de datos.  
Se debe poseer el privilegio DROP en cada vista a eliminar.

# Sintaxis de SHOW CREATE VIEW

**SHOW CREATE VIEW** *nombre\_vista*

Muestra la sentencia CREATE VIEW que se utilizó para crear la vista

```
mysql> SHOW CREATE VIEW v;
```

```
+-----+-----+
| Table | Create Table                                |
+-----+-----+
| v     | CREATE VIEW `test`.`v` AS select 1 AS `a`,2 AS `b` |
+-----+-----+
```