

A decorative graphic on the left side of the slide consisting of two overlapping parallelograms. The front one is blue and the back one is a light green color. They are both tilted at an angle.

# MODELOS DE DATOS

*VISTAS*

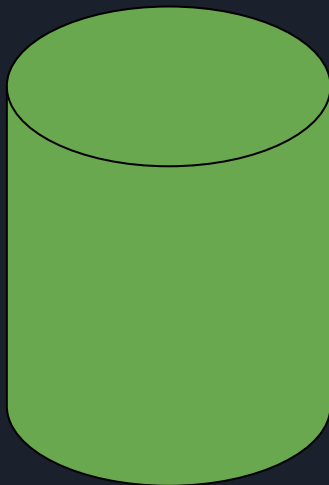


RECORDERIS...





# ¿QUÉ APRENDEREMOS HOY?



- Qué son las vistas
- Ventajas y desventajas
- Tipos de vistas que no existen en mysql ni mariadb, pero como emularlas en caso de ser necesario



# DEFINICIÓN: VISTA

**Código sql (ojo, no es pl/sql) que tenga cierto nivel de complejidad y que por su necesidad de ejecución repetitivamente, puede mostrarse como si fuera una tabla.**

**Algunos textos lo llaman "tabla virtual" porque a pesar que se muestra como una tabla, realmente no lo es.**

# VISTAS: Ejemplos

Recordando la tabla de  
"singers"...

ID, nombre, apellido,  
fecha de nacimiento...

```
MariaDB [singers]> select id, first_name, last_name, birth_date from singers;
```

id	first_name	last_name	birth_date
1	SELENA	GOMEZ	1992-07-22
2	BECKY	G	1997-03-02
3	KAROL	G	1991-02-14
4	TAYLOR	SWIFT	1989-12-13
5	ARIANA	GRANDE	1993-06-26
6	DUA	LIPA	1995-08-22
7	CAMILA	CABELLO	1997-03-03
8	SHAWN	MENDES	1998-08-08
9	BILLIE	EILISH	2001-12-18
10	ED	SHEERAN	1991-02-17
11	JUSTIN	BIEBER	1994-03-01
12	RIHANNA	FENTY	1988-02-20
13	KATY	PERRY	1984-10-25
14	HARRY	STYLES	1994-02-01
15	THE WEEKND	TESFAYE	1990-02-16
16	BEYONCÉ	KNOWLES	1981-09-04
17	ZAYN	MALIK	1993-01-12
18	CARDI	B	1992-10-11
19	POST	MALONE	1995-07-04
20	LIZZO	JEFFERSON	1988-04-27

20 rows in set (0,000 sec)



# VISTAS: Ejemplos

Situación: No se tiene la edad (porque es un dato calculado), sino que se tiene la fecha de nacimiento.

Necesidad: se debe retornar la edad en muchos módulos de ese sistema. Y se está volviendo un tema repetitivo...

```
select  id, first_name, last_name, birth_date,  
        floor(datediff(now(), birth_date)/365) as edad  
from    singers;
```

# VISTAS: Ejemplos

Ejemplo:

Tabla "singers"  
mostrando la edad,  
usando una consulta sql  
y funciones nativas para  
el cálculo

```
MariaDB [singers]> select id, first name, last name, birth_date,  
-> floor(datediff(now(), birth_date)/365) as edad from singers;
```

id	first_name	last_name	birth_date	edad
1	SELENA	GOMEZ	1992-07-22	31
2	BECKY	G	1997-03-02	27
3	KAROL	G	1991-02-14	33
4	TAYLOR	SWIFT	1989-12-13	34
5	ARIANA	GRANDE	1993-06-26	30
6	DUA	LIPA	1995-08-22	28
7	CAMILA	CABELLO	1997-03-03	27
8	SHAWN	MENDES	1998-08-08	25
9	BILLIE	ELISH	2001-12-18	22
10	ED	SHEERAN	1991-02-17	33
11	JUSTIN	BIEBER	1994-03-01	30
12	RIHANNA	FENTY	1988-02-20	36
13	KATY	PERRY	1984-10-25	39
14	HARRY	STYLES	1994-02-01	30
15	THE WEEKND	TESFAYE	1990-02-16	34
16	BEYONCÉ	KNOWLES	1981-09-04	42
17	ZAYN	MALIK	1993-01-12	31
18	CARDI	B	1992-10-11	31
19	POST	MALONE	1995-07-04	28
20	LIZZO	JEFFERSON	1988-04-27	35

20 rows in set (0,000 sec)

# Ejemplo: Vista sobre "singers"

```
create view singers_ext as
```

Nombre de la vista

Definición o cuerpo de la vista

```
select *,  
       floor(datediff(now(), birth_date)/365) as edad  
from   singers;
```

Acceder a la vista como si  
fuera una tabla

```
select id, first_name, last_name, birth_date, edad  
from   singers_ext;
```



# VISTAS: Ejemplos

El campo "edad" aparece como si fuera una tabla. Pero físicamente no existe.

En este caso, la edad se calcula *cada vez* que se haga select sobre la vista

```
MariaDB [singers]> select id, first_name, last_name, edad from singers_ext;
```

id	first_name	last_name	edad
1	SELENA	GOMEZ	31
2	BECKY	G	27
3	KAROL	G	33
4	TAYLOR	SWIFT	34
5	ARIANA	GRANDE	30
6	DUA	LIPA	28
7	CAMILA	CABELLO	27
8	SHAWN	MENDES	25
9	BILLIE	EILISH	22
10	ED	SHEERAN	33
11	JUSTIN	BIEBER	30
12	RIHANNA	FENTY	36
13	KATY	PERRY	39
14	HARRY	STYLES	30
15	THE WEEKND	TESFAYE	34
16	BEYONCÉ	KNOWLES	42
17	ZAYN	MALIK	31
18	CARDI	B	31
19	POST	MALONE	28
20	LIZZO	JEFFERSON	35

```
20 rows in set (0,001 sec)
```





# Más sobre las vistas...

La vista puede contener cualquier código sql, incluyendo condicionales, funciones de agregación, etc...

```
create view my_view as
    select    table1.field1,
             table1.field2,
             table2.field1,
             count(*) as total
    from      table1, table2
    where     table1.id = table2.id
    and       table1.field1 like '%TEST'
    and       table2.field2 in ('VAL1', 'VAL2')
    group by  table1.field1,
             table1.field2,
             table2.field1;
```

# Más sobre las vistas...

	
Simplifica consultas complejas. Ej: no hay que escribir cada vez funciones calculadas	Generalmente son de sólo lectura, por lo cual no siempre se puede hacer un insert directamente en ellas
Seguridad: Abstrae las tablas a las cuales se accede físicamente	Si cambia su estructura en el tiempo, implica una carga operativa adicional
No ocupan espacio de almacenamiento por su modalidad de ser "tablas virtuales"	Si usa funciones muy complejas sobre muchos datos, impacta el rendimiento



# EJERCICIO (1 / 5)

## Ejercicio 1:

Crear la vista "v\_countries\_01" con los siguientes campos de la tabla "country":

- id (corresponde al campo "Code" en la tabla)
- name (corresponde al campo "Name" en la tabla)
- relative\_size:
  - "SMALL" si el área es menor o igual a 1'000.000 m<sup>2</sup>
  - "MEDIUM" si el área es menor o igual a 5'000.000 m<sup>2</sup>
  - "BIG" si el área es mayor a 5'000.000 m<sup>2</sup>



# EJERCICIO (2 / 5)

## Ejercicio 2:

Crear la vista "v\_countries\_02" igual al ejercicio anterior pero sólo con los países de norte américa y sur américa.

Una vez revisados los datos, cambiar el área de algún país al azar y repetir la ejecución de la vista. Validar que los resultados reflejen los cambios en el tamaño del país.



# EJERCICIO (3 / 5)

## Ejercicio 3:

Crear la vista "v\_countries\_02" con los campos "Code", "Name", "Continent" e "IndepYear". Adicionar un campo calculado que indique hace cuantos años obtuvo cada país su independencia. Este debe ser un número entero.

Una vez revisados los datos, cambiar el año de independencia de algún país al azar y repetir la ejecución de la vista. Validar que los resultados reflejen los cambios.



# EJERCICIO (4 / 5)

## Ejercicio 4:

Crear la vista "v\_countries\_04" con los siguientes campos de las tablas "country" y "countrylanguage":

- id del país (corresponde al campo "Code" en la tabla "Country")
- language\_name (corresponde al campo "Language" en la tabla "countrylanguage")
- is\_official (corresponde al campo "IsOfficial" en la tabla "countrylanguage")
- Debe mostrar los idiomas hablados en todos y cada uno de los países



# EJERCICIO (5 / 5)

## Ejercicio 5:

Crear la vista "v\_cities\_countries\_01" con los siguientes campos de las tablas "country" y "city":

- id del país, id de la ciudad, nombre del país, nombre de la ciudad.
- El resultado debe estar ordenado alfabéticamente, ascendente, primero por nombre de país y luego por nombre de ciudad.

La idea es mostrar las ciudades en cada uno de los países





# ¿PREGUNTAS?

