# TALLER: Vistas en SQL

# Integrantes:

# Maira Alejandra Arévalo Tobar

# Katherin Juliana Rosero Delgado

* ¿Qué es una vista en SQL y en qué se diferencia de una tabla?

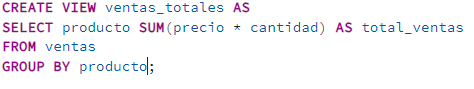
Una vista es una tabla virtual que se basa en una o más tablas base. Una tabla es un objeto físico que almacena datos en filas y columnas.

* Menciona **tres ventajas** de utilizar vistas en SQL y explica cada una con un ejemplo.

1. Permiten restringir el acceso a ciertas columnas o filas sin exponer la tabla completa. Ejemplo:



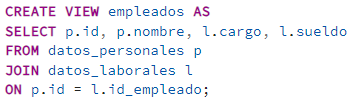
1. Se pueden almacenar consultas complejas en una vista para reutilizarlas sin repetir código.



Luego en vez de escribir la consulta completa, basta con:



1. Si la estructura de la base de datos cambia, las vistas pueden mantener la compatibilidad con aplicaciones que dependen de ellas. Ejemplo:



* Explica la diferencia entre CREATE VIEW y CREATE OR REPLACE VIEW.

CREATE VIEW crea una nueva vista, mientras que CREATE OR REPLACE VIEW actualiza una vista existente.

* Si una vista se basa en varias tablas con un JOIN, ¿qué sucede si eliminamos una de esas tablas? Justifica tu respuesta.

La vista queda invalidada ya que la vista depende de esa tabla para funcionar y si se intenta consultar la vista, la base de datos lanzará un error indicando que la tabla referenciada no existe, para esto será necesario modificar o recrear la vista sin incluir la tabla eliminada.

* ¿Por qué algunas vistas en SQL **no permiten realizar modificaciones** en los datos?

Algunas vistas en SQL no permiten realizar modificaciones en los datos por diferentes razones:

Algunas vistas pueden estar basadas en múltiples tablas o JOINS y esto lo hace complicado al momento de aplicar cambios, también podría ser porque algunas vistas tienen funciones agregadas lo cual no son actualizables, por otra parte, algunas vistas tienen restricciones que limitan las modificaciones para proteger la integridad de los datos.

* Explicar las diferencias entre vistas estáticas y dinámicas.

**Vistas Estáticas**

* **Almacenamiento:** Guardan físicamente los resultados de la consulta en la base de datos.
* **Rendimiento:** Mejoran el rendimiento en consultas frecuentes porque no se recalculan en cada acceso.
* **Actualización:** No reflejan automáticamente los cambios en las tablas base; requieren una actualización manual o programada.
* **Uso recomendado:** Cuando se trabaja con grandes volúmenes de datos y se necesita optimizar la velocidad de consulta.

**Vistas Dinámicas**

* **Almacenamiento:** No almacenan datos; solo guardan la consulta SQL.
* **Rendimiento:** Más lento en comparación con las vistas, ya que ejecutan la consulta cada vez que se accede a la vista.
* **Actualización:** Siempre muestran los datos más recientes porque se generan en tiempo real a partir de las tablas base.
* **Uso recomendado:** Para garantizar que los datos estén siempre actualizados sin necesidad de refrescar la vista manualmente.

**Objetivo:** Practicar la creación, modificación y consulta de vistas en bases de datos relacionales.

## Parte 1: Creación de la Base de Datos

1. Crea la base de datos BibliotecaOnline y conéctate a ella.
2. Crea las siguientes tablas:

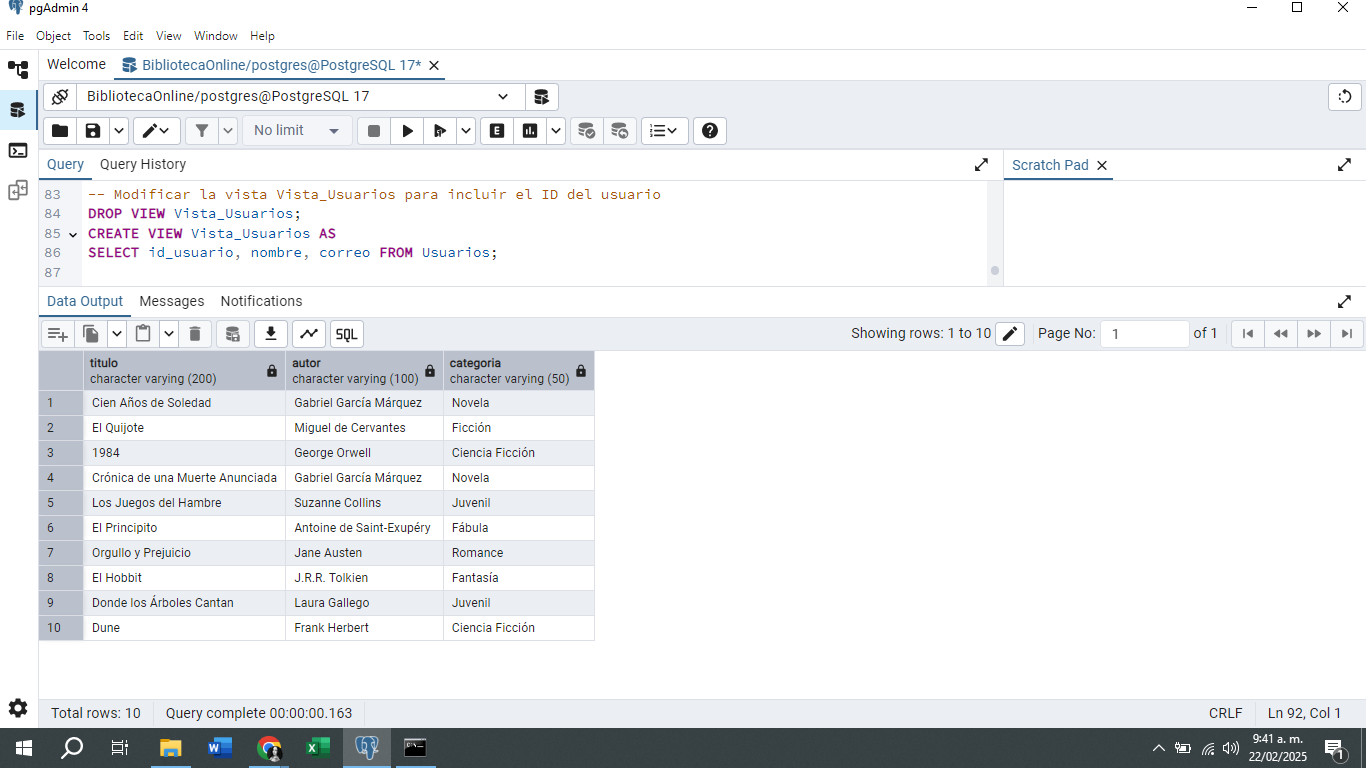
Usuarios con los atributos: id\_usuario, nombre,correo.

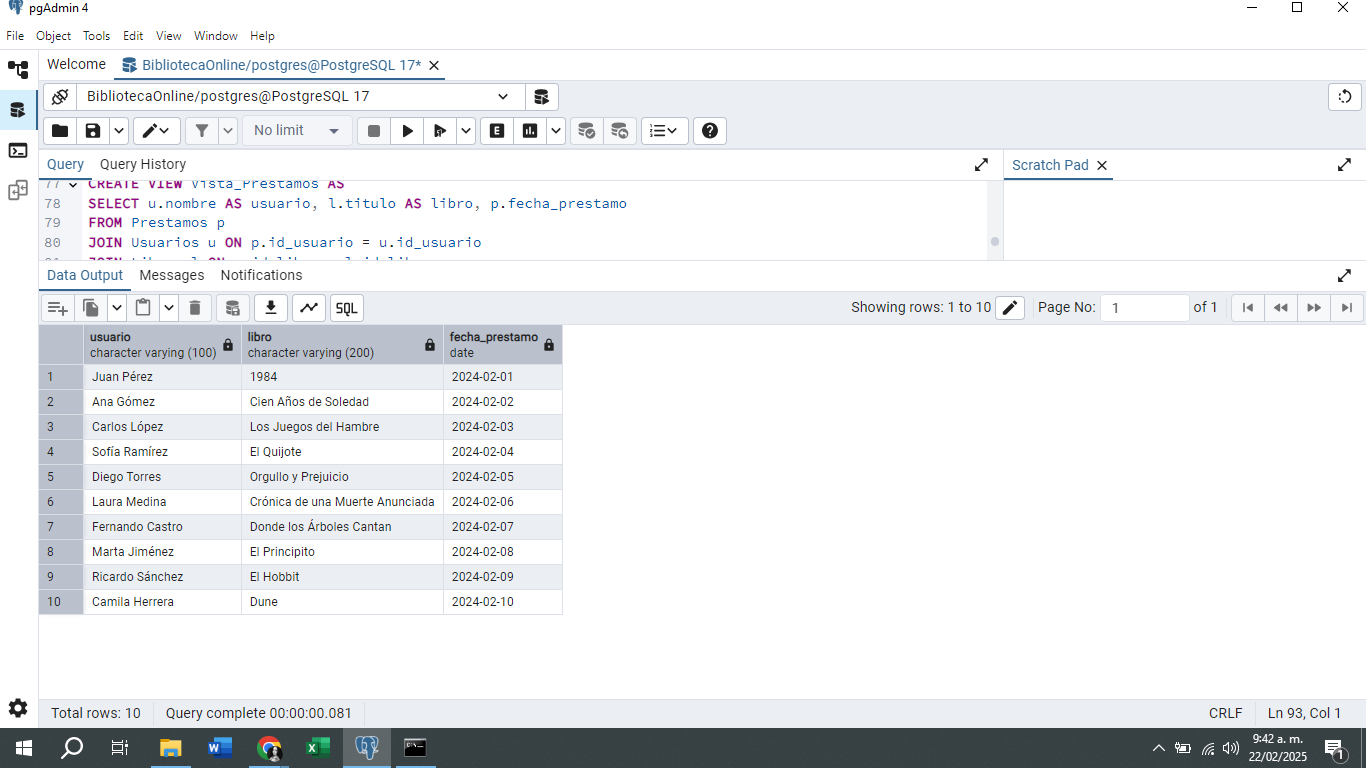
Libros con los atributos: id\_libro, titulo, autor, categoria , precio.

Prestamos con los atributos: id\_prestamo, id\_usuario, id\_libro, fecha\_prestamo

Inserta al menos 10 registros en cada tabla.

## Parte 2: Creación y Uso de Vistas

1. Crear una vista de usuarios y sus correos electrónicos
2. Crear una vista de título, autor y categoría.
3. Crear una vista de libros prestados con el nombre del usuario y el título del libro
4. Modificar la vista **Vista\_Usuarios** para incluir el ID del usuario
5. Eliminar la vista **Vista\_Usuarios**
6. Consultar las vistas creadas

****