

## Taller de Lenguajes II – 2025

Programador Universitario / Licenciatura en informática / Ingeniería en Informática  
TP Nro 6

### CONTENIDOS

- Base de Datos SQLite
- Sentencias SQL

### INTRODUCCIÓN

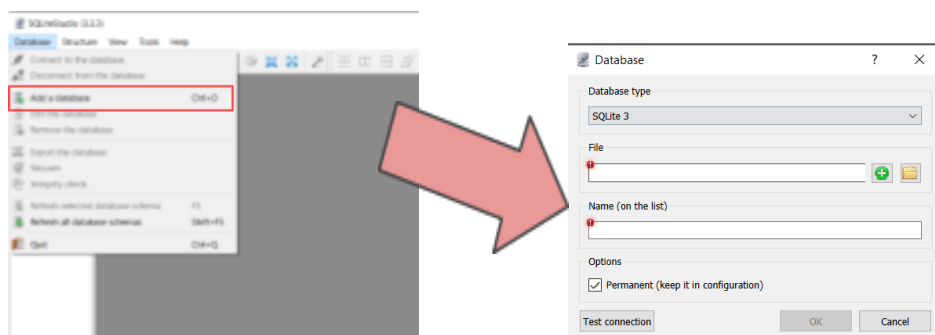
Copie el siguiente enlace en su navegador: <https://tinyurl.com/TL2-TP6-2025> esto creará el repositorio para poder subir el **Trabajo Práctico Nro. 6**.

Se nos solicita implementar un prototipo de generador de presupuestos de productos que será utilizado como prototipo para la página web de una Distribuidora de Insumos Informáticos.

- 1) Descargue la aplicación [SQLite Studio](https://sqlitestudio.pl) (<https://sqlitestudio.pl>),. Este software es una IDE que le permitirá administrar bases de datos SQLite de forma sencilla. Su interfaz gráfica facilita la navegación, visualización y modificación de datos dentro de las bases de datos SQLite.
- 2) Una vez que haya descargado e instalado SQLite Studio, cree una base de datos llamada **Tienda.db**, que contenga las siguientes tablas:
  - a) Productos
    - i) \*idProducto integer
    - ii) Descripcion string 100
    - iii) Precio numerico 10.2
  - b) Presupuestos
    - i) \*IdPresupuesto integer
    - ii) NombreDestinatario string 100
    - iii) FechaCreacion date
  - c) PresupuestosDetalle
    - i) \*idPresupuesto integer
    - ii) \*idProducto integer
    - iii) Cantidad integer

### Abrir y explorar la base de datos

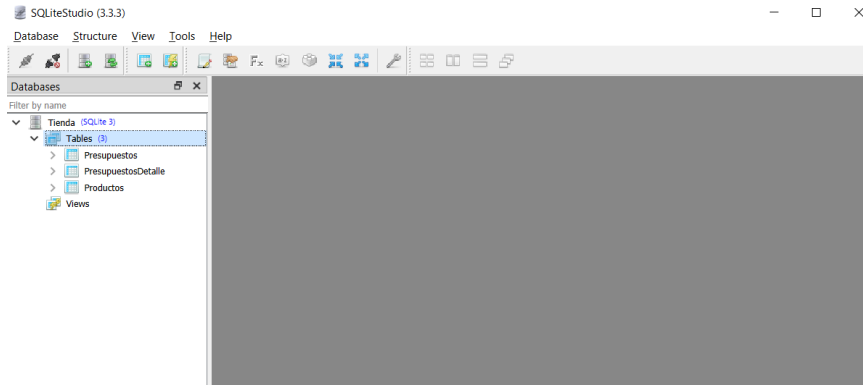
1. Abra **SQLite Studio** y en la barra de herramientas superior, seleccione la opción “Add database” o “Agregar Base de datos” (siga la **Imagenes** para orientación).



## Taller de Lenguajes II – 2025

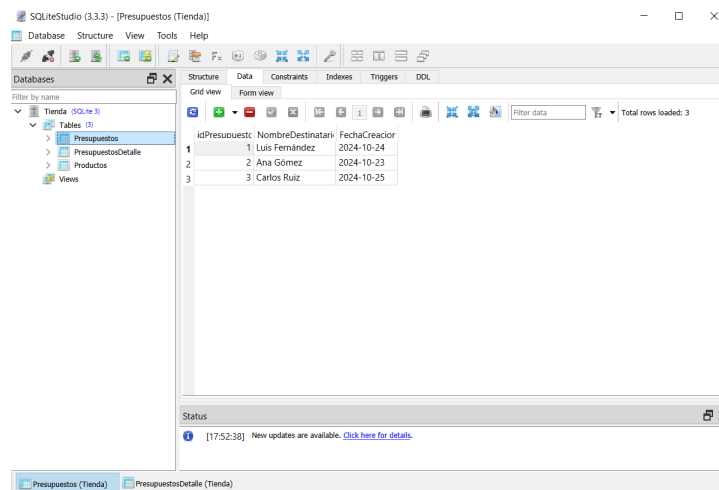
Programador Universitario / Licenciatura en informática / Ingeniería en Informática  
TP Nro 6

2. Navegue hasta la ubicación de **Tienda.db** y ábrala. Al cargar la base de datos, cree las tablas **Productos**, **Presupuestos** y **PresupuestosDetalle** en el panel izquierdo de la ventana (debería tener algo similar a lo que muestra la **Imagen 2**).



3. Una vez creada la tabla con sus campos correspondientes, podrá agregar y navegar los datos almacenados haciendo clic sobre la tabla y accediendo a la solapa **“Data”**. Desde allí podrá cargar nuevos registros en la tabla. Recuerde que los valores ingresados deben cumplir con las restricciones definidas anteriormente.

En el siguiente ejemplo se muestran los campos de los productos existentes de forma encolumnada, tal como se observa en la **Imagen 3**.



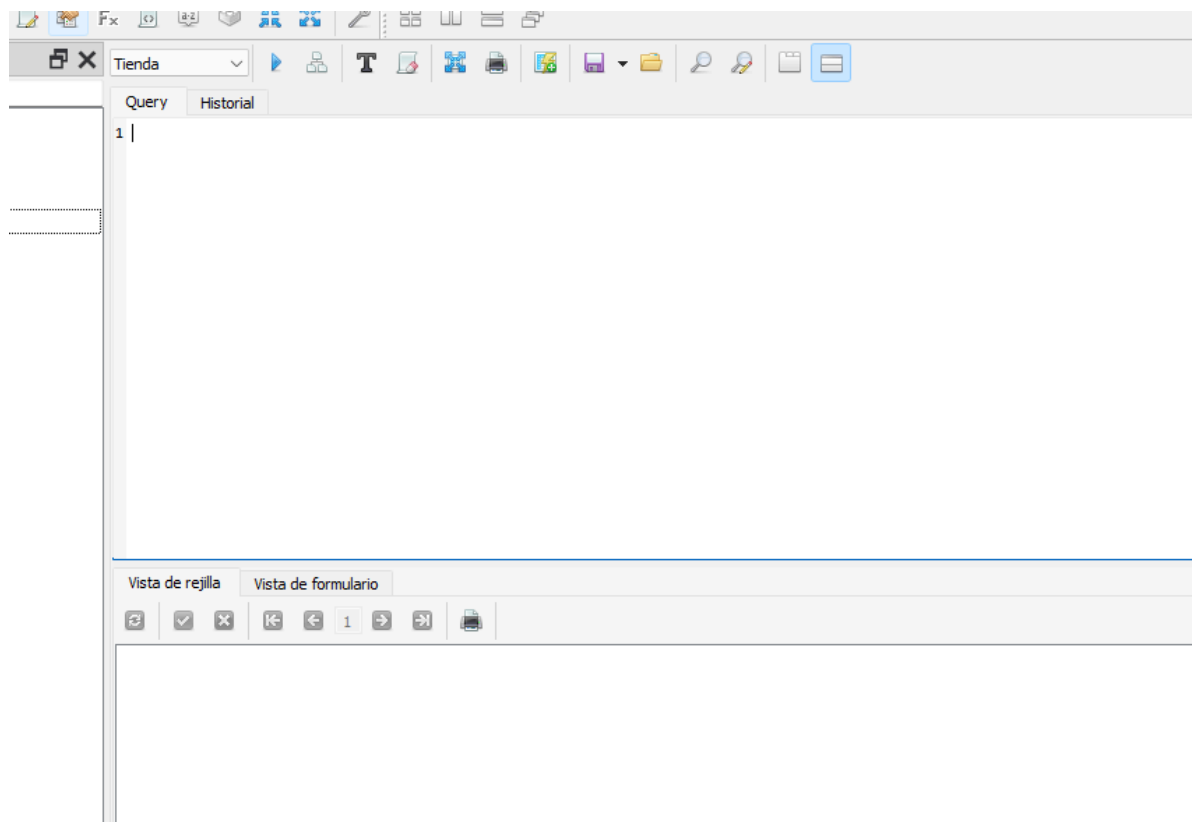
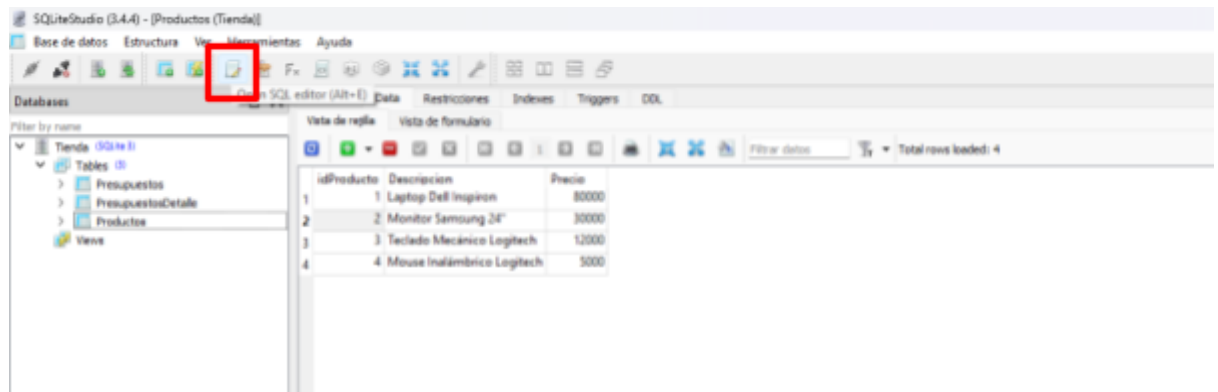
- 3) **En SQLite Studio pruebe las consultas SQL y vea los resultados.**

Para realizar consultas SQL manuales, seleccione la base de datos en el panel izquierdo y luego haga clic en el ícono “SQL Editor” (o vaya al menú Tools → Open SQL editor). Desde allí podrá escribir y ejecutar sus propias sentencias SQL, como por ejemplo consultas SELECT, inserciones o actualizaciones. Como puede ver en la Imagen 4 y 5

## Taller de Lenguajes II – 2025

Programador Universitario / Licenciatura en informática / Ingeniería en Informática

TP Nro 6



Una vez dentro del **SQL Editor** en la solapa Query, para probar el funcionamiento de la base de datos, copie y pegue las siguientes sentencias en el editor y haga clic en el botón **Execute SQL (F9)** para ejecutarlas.

De esta manera podrá visualizar los resultados de las consultas en la parte inferior de la ventana.

## Taller de Lenguajes II – 2025

Programador Universitario / Licenciatura en informática / Ingeniería en Informática

TP Nro 6

### a) Insertar un Producto

```
INSERT INTO Productos (Descripcion, Precio) VALUES ('[Descripcion]',  
[Precio]);
```

Ej:

```
INSERT INTO Productos (Descripcion, Precio) VALUES ('Mouse Inalámbrico  
Logitech', 5000.0);
```

### b) Insertar un Presupuesto

```
INSERT INTO Presupuestos (NombreDestinatario, FechaCreacion) VALUES  
('[NombreDestinatario]', '[FechaCreacion]');
```

Ej:

```
INSERT INTO Presupuestos (NombreDestinatario, FechaCreacion) VALUES  
('Carlos Ruiz', '2024-10-25');
```

### c) Insertar registros en PresupuestoDetalle

```
INSERT INTO PresupuestosDetalle (idPresupuesto, idProducto, Cantidad)  
VALUES ([idPresupuesto], [idProducto], [Cantidad]);
```

Ej:

```
INSERT INTO PresupuestosDetalle (idPresupuesto, idProducto, Cantidad)  
VALUES (1, 3, 2); -- 2 unidades del producto con id 3 en el presupuesto 1
```

### d) Modificar un producto

```
UPDATE Productos SET Descripcion = '[NuevaDescripcion]', Precio =  
[NuevoPrecio] WHERE idProducto = [idProducto];
```

Ej:

```
UPDATE Productos  
SET Descripcion = 'Teclado Mecánico Logitech',  
Precio = 12000  
WHERE idProducto = 3;
```

### e) Modificar NombreDestinatario de Presupuesto

```
UPDATE Presupuestos SET NombreDestinatario =  
'[NuevoNombreDestinatario]' WHERE idPresupuesto = [idPresupuesto];
```

Ej:

```
UPDATE Presupuestos SET NombreDestinatario = 'Luis Fernández' WHERE  
idPresupuesto = 1;
```

### f) Eliminar un registro de PresupuestoDetalle

```
DELETE FROM PresupuestosDetalle WHERE idPresupuesto =  
[idPresupuesto] AND idProducto = [idProducto];
```

Ej:

```
DELETE FROM PresupuestosDetalle WHERE idPresupuesto = 1 AND  
idProducto = 2;
```

## Taller de Lenguajes II – 2025

Programador Universitario / Licenciatura en informática / Ingeniería en Informática

TP Nro 6

### 4) Conexión con el Proyecto .NET

En el repositorio de GitHub que creaste, encontrarás un proyecto de consola en .Net

Este proyecto está preparado para conectarse a una base de datos `Base_Test.db` totalmente funcional.

Luego de ver como funciona, modifica el código para acceder a la base de datos que creaste, para eso hacer lo siguiente:

1. **Copia tu archivo `Tienda.db`** a la carpeta de raíz de tu repositorio.
2. **Explora el código en Visual Studio Code:** Abre el proyecto y observa cómo se utilizan librerías como `Microsoft.Data.Sqlite` para establecer una conexión y ejecutar las mismas sentencias SQL que probaste en SQLite Studio.

**Nota:** En caso que no te reconozca los comando relacionados a SQLite ejecuta para que te agregue la librería al proyecto

- `dotnet add package Microsoft.Data.Sqlite`