

ACTIVIDAD 7. GUI's y orígenes de datos.

En esta actividad utilizaremos el codificado de la anterior y añadiremos una conexión a una base de datos para almacenar los registros de forma permanente.

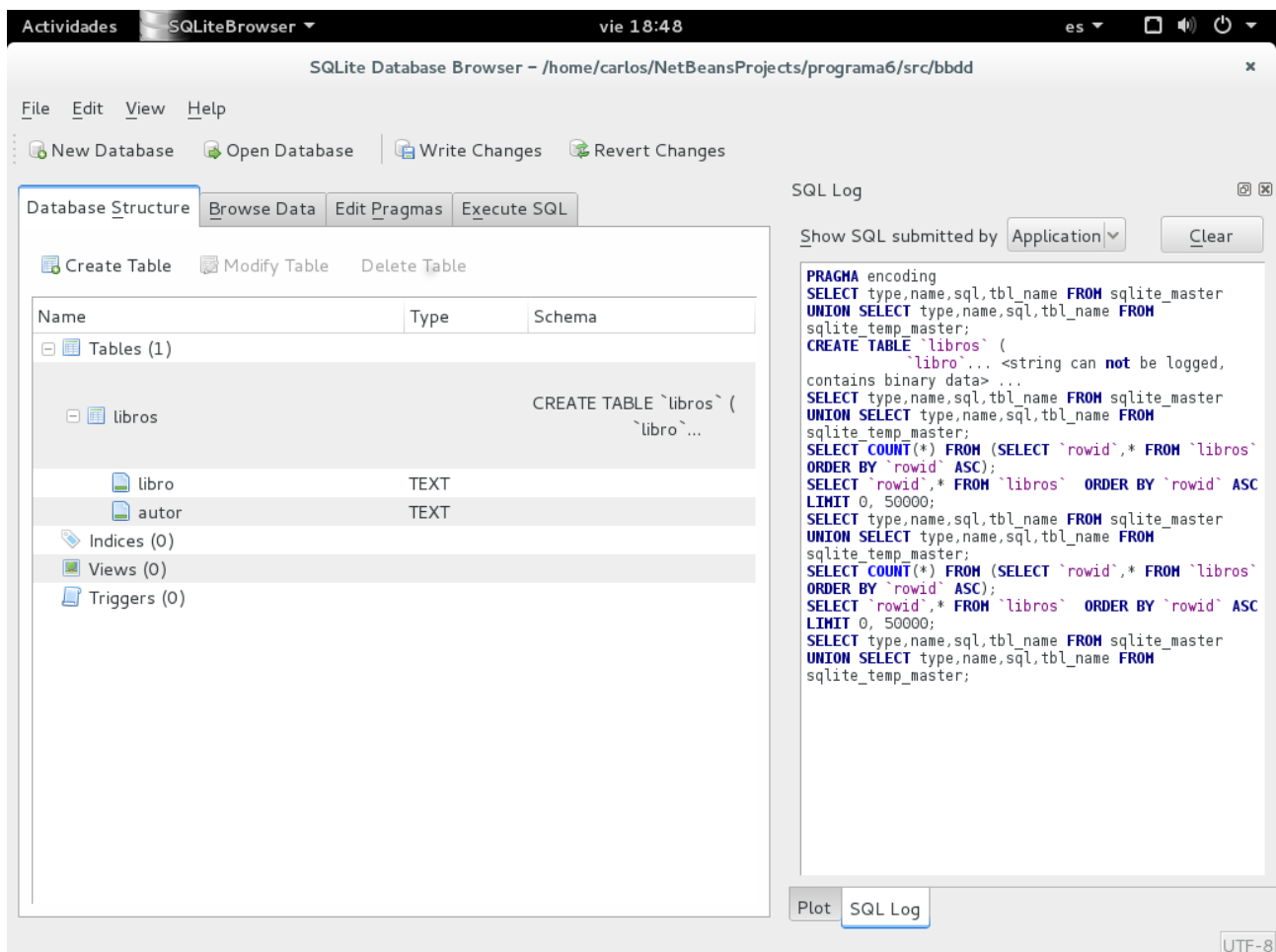
Para ello usaremos SQLite. SQLite es un gestor de base de datos de tamaño pequeño-mediano. Frente a MySQL o SQLServer tiene bastantes [limitaciones](#) como por ejemplo el carecer de un tipado de datos más desarrollado y no permitir conexiones simultáneas al estilo de sus competidores. Pero a diferencia de los otros es software libre y muy ligero.

Primero vamos a instalarlo.

```
#apt-get install sqlite //no olvidemos de actualizar previamente el sistema
```

Hay varias formas de gestionar la base de datos. Firefox tiene un complemento para ello, *sqlitemanager*. En Debian tenemos herramientas como *sqliteman* y *sqlitebrowser* y que será la que usemos. Pero cada uno que use la que prefiera.

En primer lugar creamos la base de datos.



Luego nos vamos al código de la aplicación e incluimos lo siguiente:

```
.....  
import sqlite3 as dbapi  
bd = 'bdd'  
# realizamos la conexión  
conex = sqlite3.connect(bdd)  
c = conex.cursor()
```

```
class main:
```

```
.....
```

Dentro del botón que iba insertando los valores vamos a poner el código necesario para ingresar un registro. En nuestro caso sería algo así:

```
registro = (u,c, ...,)  
try  
    c.execute('INSERT INTO tLibros (campo1,campo2....) VALUES  
              (?,?,?...), (registro))  
    conex.commit() #muy importante  
except sqlite3.Error e:  
    mensaje = "Se ha producido un error..."  
    # aquí mostraríamos una ventana no-modal advirtiéndolo  
    ...  
    conex.rollback() #por si acaso
```

ACTIVIDADES

1. Modifica el ejercicio de forma que cuando marquemos un elemento de la lista en dos **label** (también vale dos **entry**) se carguen los datos de la fila señalada.

model = modelo de lista (2 valores en este caso)

iter = apunta al registro seleccionado

```
.....  
model, iter = self.tview.get_selection().get_selected()  
libro = model[iter][0]  
autor = model[iter][1]  
.....
```