

ENLACE A DATOS – I

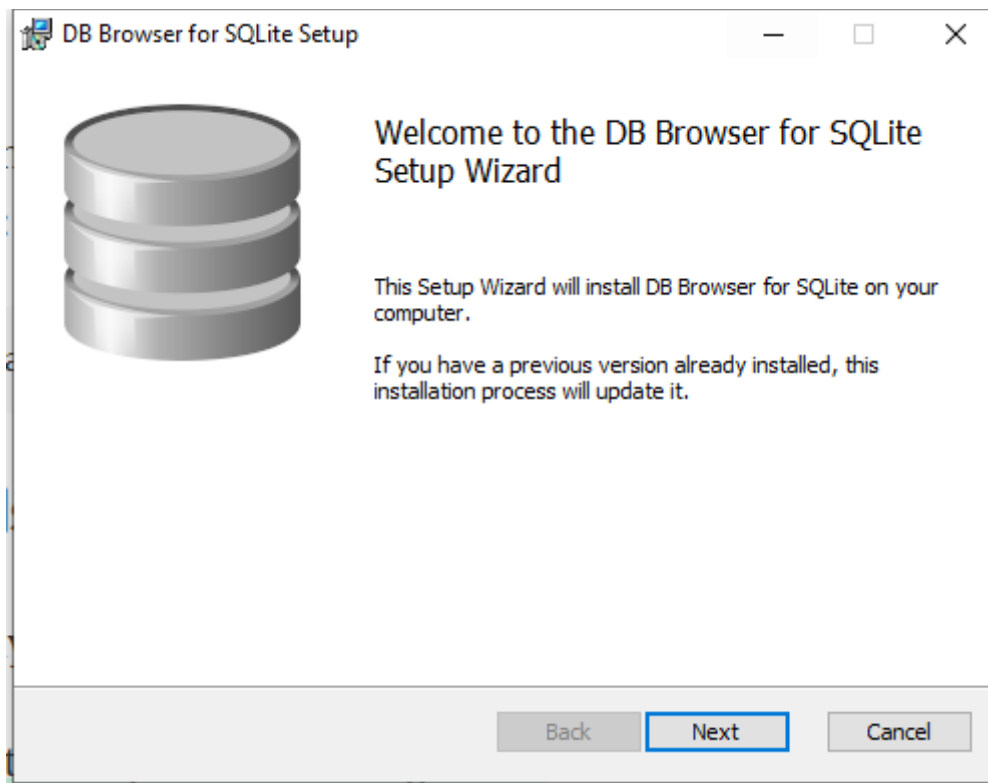
PyQT 5 soporta la conexión con varios servidores *SQL* como: **SQLite**, MySQL, ODBC, y PostgreSQL, así como servidores *noSQL* como **MongoDB**. Para conectarnos a cualquiera de estos servidores debemos instalar el controlador correspondiente. En nuestro caso usaremos **SQLite**.

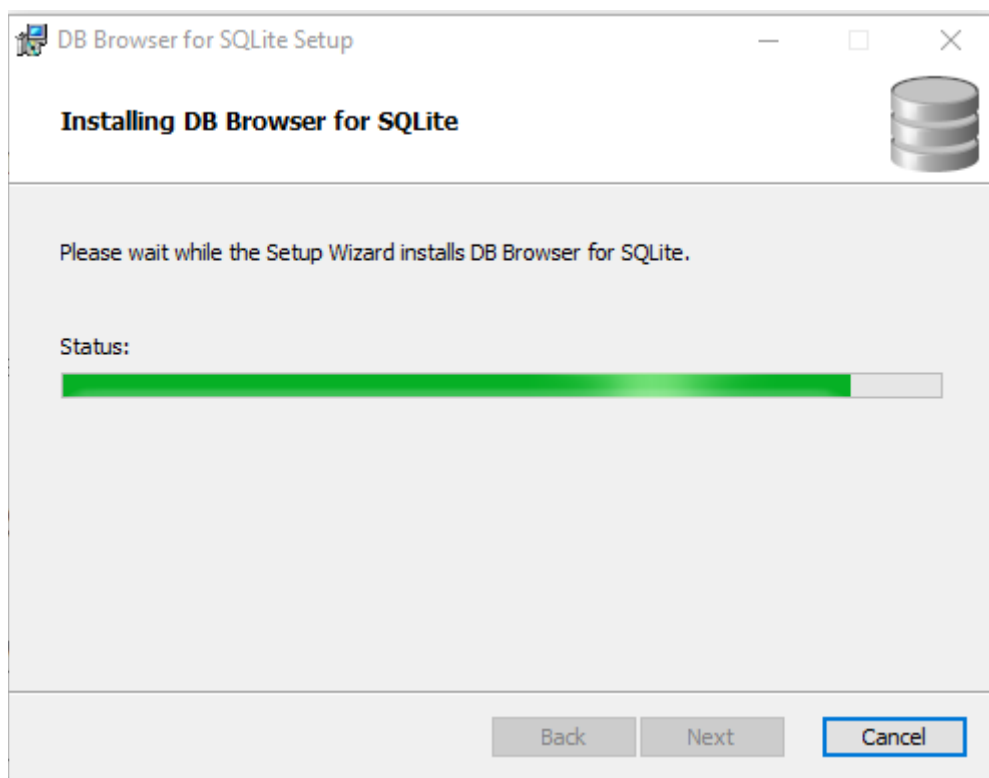
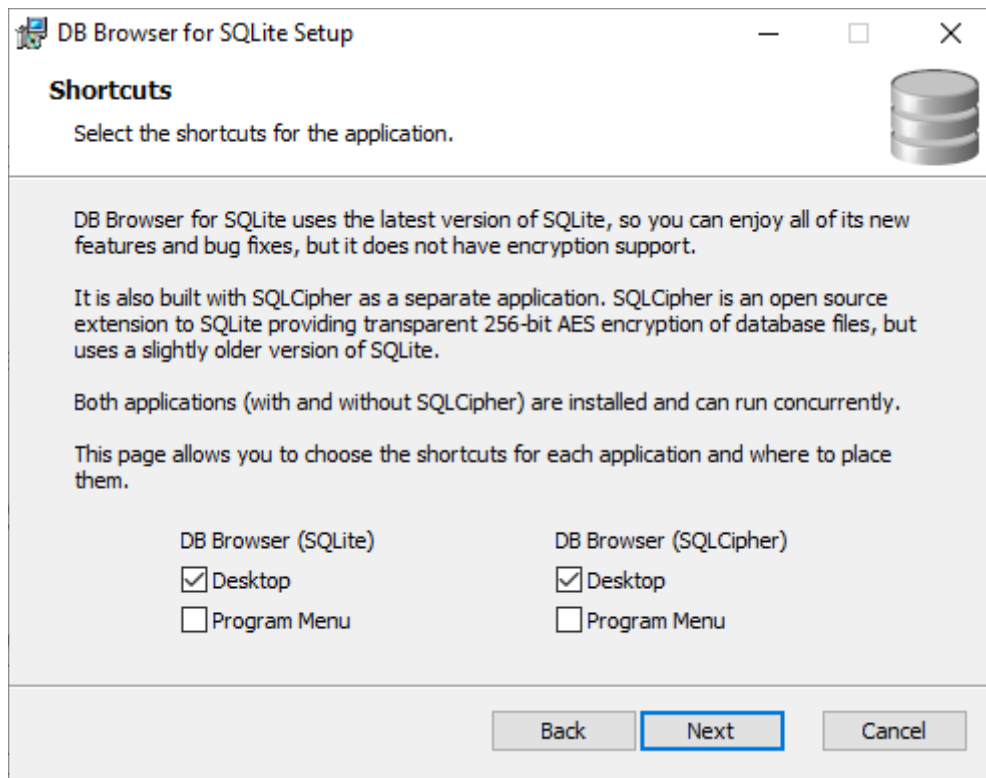
SQLite

Es un sistema de base de datos relacional contenido en una biblioteca desarrollada en C que se integra en la aplicación a diferencia del resto y que se caracteriza por ser libre y rápido. Dicho de otro modo, tiene integrado el motor de base de datos SQL mientras que en los otros gestores el motor se alberga en un servidor independiente.

SQLite lee y escribe directamente en archivos de disco ordinarios, el contenido de una base de datos se almacena en un solo archivo de disco, lo que le proporciona una gran versatilidad y rendimiento incluso en entornos de poca memoria. Es el sistema gestor de base de datos implementado en las aplicaciones Android.

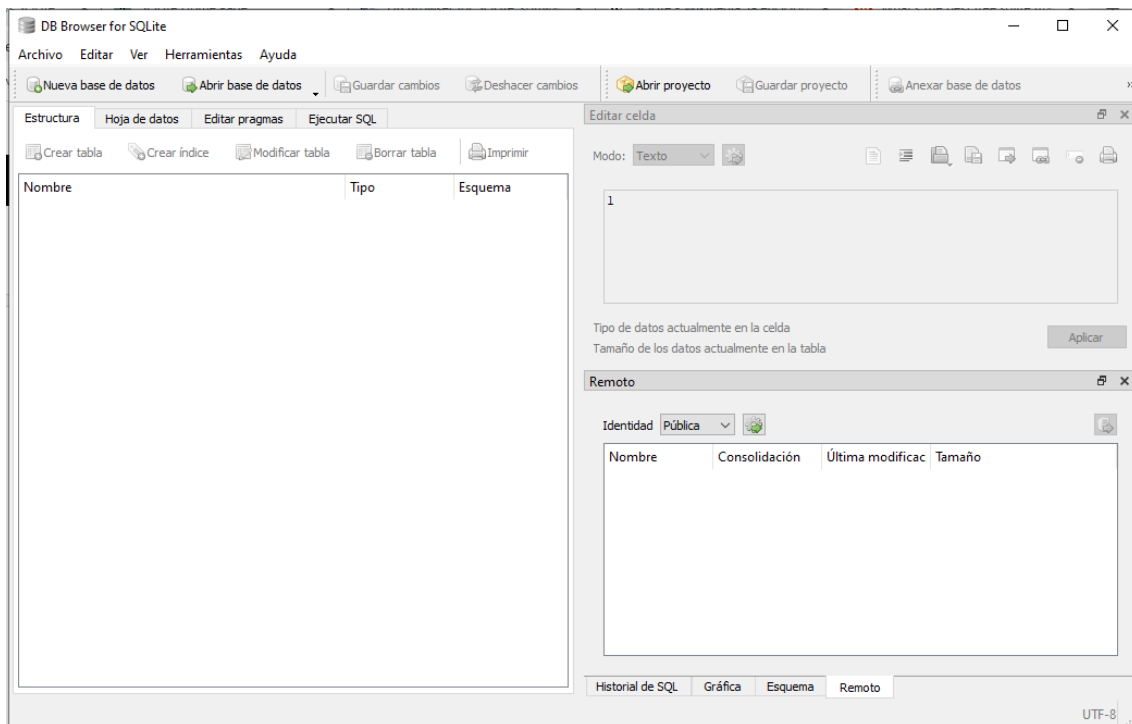
Su instalación en Windows es sencilla. Para ello vamos a la web <https://sqlitebrowser.org/dl/> y descargamos el instalador apropiado.



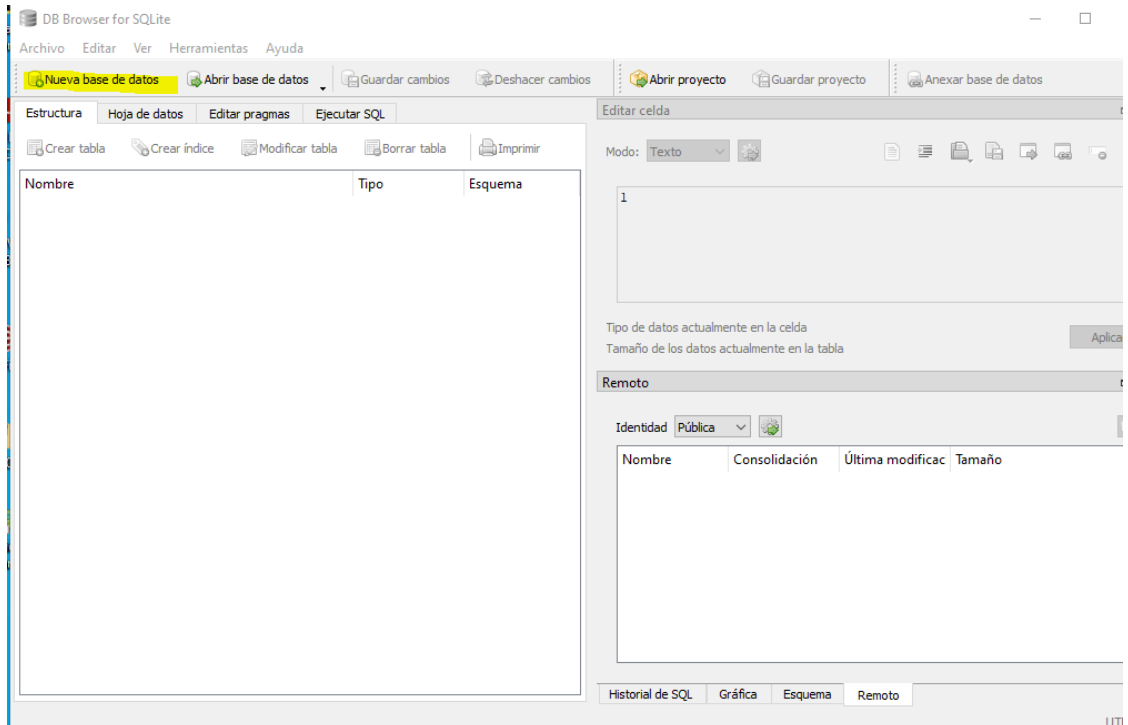


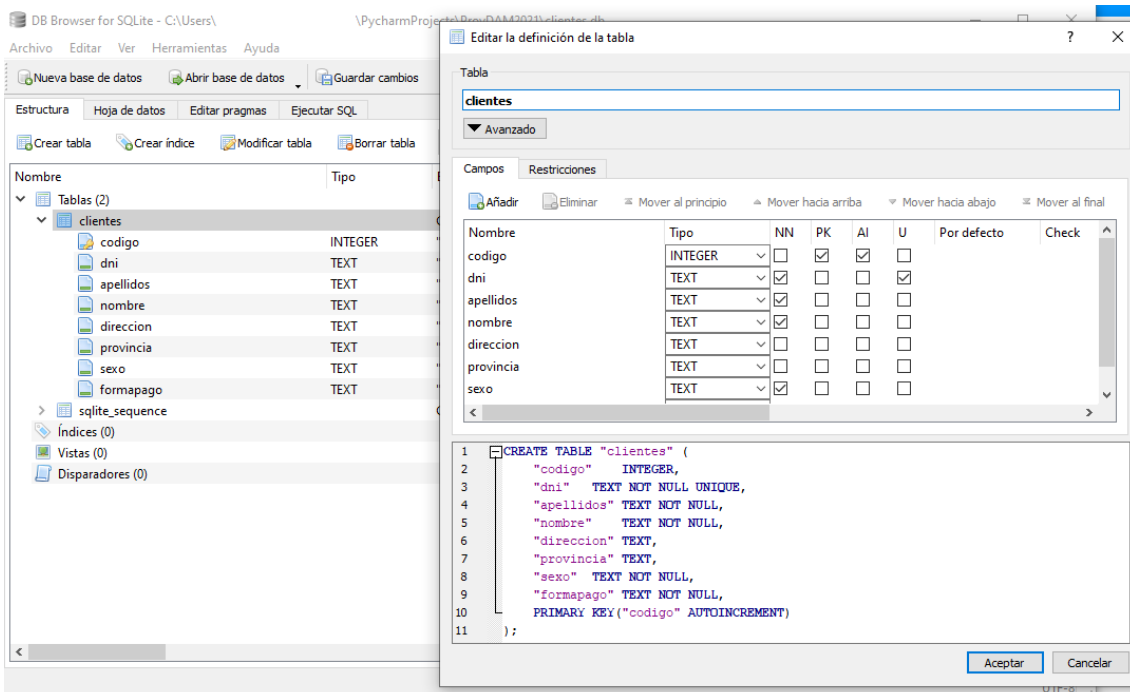
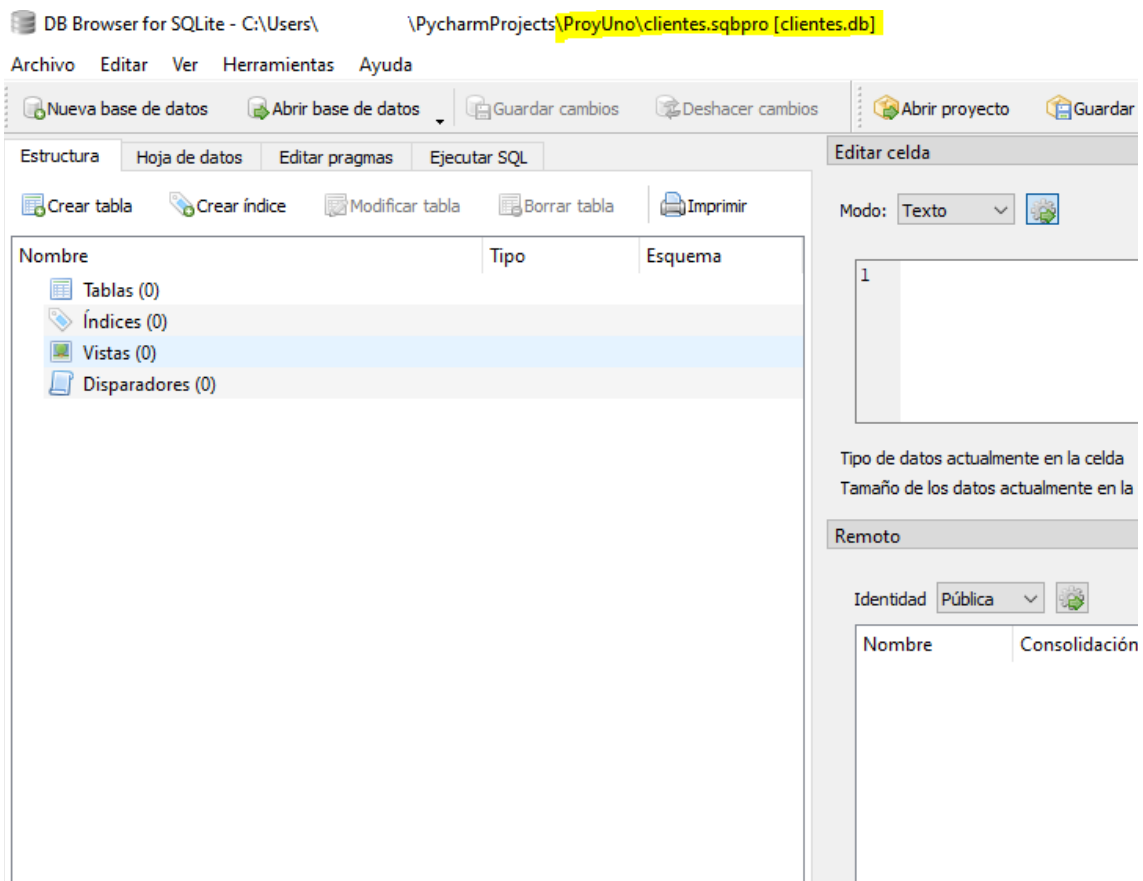
DBBrowser o SQLiteBrowser es una aplicación gratuita que facilita de forma sencilla e intuitiva la administración de bases de datos de forma gráfica. Existen otras opciones como SQLite Manager que es una extensión para Firefox o SQLiteStudio con alguna opción más.

Si lanzamos DB Browser nos aparece la pantalla siguiente:



Aprovechamos y creamos la primera tabla de **clientes**. En primer lugar, creamos la base de datos:





No nos olvidemos guardar los cambios.

Para establecer la conexión al servidor de bases de datos usaremos el método estático `QSqlDatabase.addDatabase('QSQLITE')` indicando el servidor que utilizaremos. Según el tipo de servidor tenemos: QSQLITE, QMYSQL, QMYSQL3, QODBC, QODBC3, QPSQL, y QPSQL7. El **método** `open()` para establecer la conexión y verificar la misma.

En nuestro ejemplo crearemos un fichero llamado *conexión.py*. Para evitar rutas lo que haremos es guardar el fichero de la base de datos en el mismo directorio que la aplicación.

En *main.py* hacemos la llamada al módulo de conexión. No nos olvidemos importar el módulo en las llamadas a las librerías.

```
var.ui.Cltitable.setSelectionBehavior(QtWidgets.QTableWidget.SelectRows)
conexion.Conexion.db_connect(var.filebd)
```

La variable *var.filebd* hace referencia a la base de datos que hemos creado y que guardamos en el módulo *var.py*.

```
dialogo = None
filebd = 'clientes.db'
```

Por último, queda el código principal, que es el módulo que se conecta a la BBDD.

```
from PyQt5 import QtWidgets, QSql

class Conexion():

    def db_connect(filename):

        db = QSql.QSqlDatabase.addDatabase('QSQLITE')
        db.setDatabaseName(filename)
        if not db.open():
            QtWidgets.QMessageBox.critical(None, 'No se puede abrir la base de datos',
                                           'No se puede establecer conexion.\n' 'Haz Click para Cancelar.',
                                           QtWidgets.QMessageBox.Cancel)
            return False
        else:
            print('Conexión Establecida')
            return True
```

Resultado:

```
Conexión establecida

Process finished with exit code 0
```