Creating an SQLite database in Qt

This example shows you how to create an SQLite database in Qt.

Preconditions

For Maemo SQLite development, the following packages must be installed:

- libqt4-sql
- libqt4-sql-sqlite
- libsqlite3-0
- libsqlite3-dev

Project file (.pro)

Add the following line to your .pro file

```
QT += sql
```

Header

```
#include <QObject>
#include <QSqlDatabase>
#include <QSqlError>
#include <QFile>

class DatabaseManager : public QObject
    {
    public:
        DatabaseManager(QObject *parent = 0);
        ~DatabaseManager();

    public:
        bool openDB();
        bool deleteDB();
        QSqlError lastError();

    private:
        QSqlDatabase db;
    };
```

Source

```
bool DatabaseManager::openDB()
    {
    // Find QSLite driver
    db = QSqlDatabase::addDatabase("QSQLITE");

    #ifdef Q_OS_LINUX
    // NOTE: We have to store database file into user home folder in Linux
    QString path(QDir::home().path());
    path.append(QDir::separator()).append("my.db.sqlite");
```


(C) Copyright Nokia 2014. All rights reserved.

```
    path = QDir::toNativeSeparators(path);
    db.setDatabaseName(path);
    #else
    // NOTE: File exists in the application private folder, in Symbian Qt implementation
    db.setDatabaseName("my.db.sqlite");
    #endif

    // Open databasee
    return db.open();
    }

QSqlError DatabaseManager::lastError()
    {
    // If opening database has failed user can ask
    // error description by QSqlError::text()
    return db.lastError();
    }

bool DatabaseManager::deleteDB()
    {
    // Close database
    db.close();

    #ifdef Q_OS_LINUX
    // NOTE: We have to store database file into user home folder in Linux
    QString path(QDir::home().path());
    path.append(QDir::separator()).append("my.db.sqlite");
    path = QDir::toNativeSeparators(path);
    return QFile::remove(path);
    #else

    // Remove created database binary file
    return QFile::remove("my.db.sqlite");
    #endif
    }
```

Postconditions

The database binary file is created on the device disk in Symbian & Windows and in the device memory in Maemo.

See also

- Creating a database table in Qt
- Inserting a row into a database in Qt
- Searching for data in a database in Qt
- Deleting data from a database in Qt
- Selecting data from a database without using SQL statements in Qt
- Using QDataWidgetMapper to show data from a database in Qt
- QtSql module
- Qt QSQlite driver
- SQLite

(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.