

# SQLite Database

SQLite is a small database engine that has become very popular. The database is stored as a local file, and you can embed the SQLite engine with your app using one of the libraries below.

Here are three different libraries you can use to access SQLite databases from .NET or Mono:

- Mono includes a [SQLite interface](#), but it does not yet support ADO.NET's DataAdapter class.
- The [SQLite.NET](#) wrapper works with Windows .NET, however it does not follow the standard [ADO.NET](#) api.
- (recommended) The [Finisar ADO.NET Provider for SQLite](#) complies with the [ADO.NET](#) API.

See [this tutorial](#) on using SQLite from a .NET app.

## Sample Code

Here is a sample using Mono's SQLite interface. It is a boo port of the Mono SQLite test case. Note, this code has not been tested, it may contain syntax errors. It is for illustration purposes only. Also, the DataAdapter section at the bottom will not work anyway.



```

//Sqlite example (DataAdapter part doesn't work)
//Requires Mono.Data.SqliteClient.dll which is included with Mono.

//On Windows, if you are compiling with .NET instead of Mono,
//put Mono.Data.SqliteClient in your build folder (if you are using booi)
//and first run "sn -Vr build\Mono.Data.SqliteClient.dll" from the
//command line to turn off strong name validation.

import System
import System.Data from System.Data

print 'If this test works, you should get:'
print 'Data 1: 5'
print 'Data 2: Mono'
print 'create SqliteConnection...'
dbcon as SqliteConnection = SqliteConnection()
connectionString as string = 'URI=file:SqliteTest.db'
print 'setting ConnectionString using: ' + connectionString
dbcon.ConnectionString = connectionString
print 'open the connection...'
dbcon.Open()
print 'create SqliteCommand to CREATE TABLE MONO_TEST'
dbcmd as SqliteCommand = SqliteCommand()
dbcmd.Connection = dbcon
dbcmd.CommandText = 'CREATE TABLE MONO_TEST ( ' + 'NID INT, ' + 'NDESC TEXT )'
print 'execute command...'
dbcmd.ExecuteNonQuery()
print 'set and execute command to INSERT INTO MONO_TEST'
dbcmd.CommandText = 'INSERT INTO MONO_TEST ' + '(NID, NDESC )' + 'VALUES(5,\Mono\)'
dbcmd.ExecuteNonQuery()
print 'set command to SELECT FROM MONO_TEST'
dbcmd.CommandText = 'SELECT * FROM MONO_TEST'
reader as SqliteDataReader
print 'execute reader...'
reader = dbcmd.ExecuteReader()
print 'read and display data...'
while reader.Read():
    print 'Data 1: ' + reader[0].ToString()
    print 'Data 2: ' + reader[1].ToString()

print 'read and display data using DataAdapter...'
adapter as SqliteDataAdapter = SqliteDataAdapter('SELECT * FROM MONO_TEST',
connectionString)
dataset as DataSet = DataSet()
adapter.Fill(dataset)
for myTable as DataTable in dataset.Tables:
    for myRow as DataRow in myTable.Rows:
        for myColumn as DataColumn in myTable.Columns:
            print myRow[myColumn])

print 'clean up...'
dataset.Dispose()
adapter.Dispose()
reader.Close()
dbcmd.Dispose()
dbcon.Close()
print 'Done.'

```

