**Unity3D教程：Unity3D与Sqlite数据库直连**

Posted on 2013年01月10日 by U3d / [Unity3D 基础教程](http://www.unitymanual.com/category/manual/unity3d-%e5%9f%ba%e7%a1%80%e6%95%99%e7%a8%8b)/被围观 475 次

**环境介绍：**

Windows7，Unity3D，SQLite Expert Personal 3

**开发语言：**

JavaScript

**需要的dll文件：**

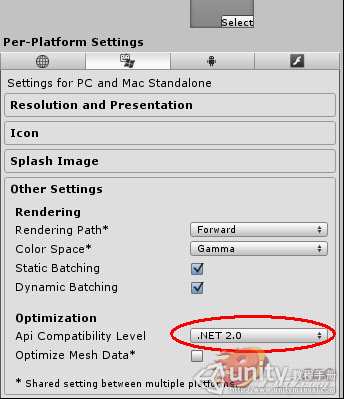
Mono.Data.Sqlite.dll和sqlite3.dll，dll文件位置，截图：



Unity3D教程：Unity3D与Sqlite数据库直连

一定要在这个目录下，请保持一致。

**如果需要将编译好的程序发布成功的话，需要改一些地方，具体见下面的截图：**



Unity3D教程：Unity3D与Sqlite数据库直连

要改动的地方已用红色标记，注意这个要改成.NET2.0，这样才能够发布的。系统默认的不是.NET2.0，这一点要注意！！！

**下面来看下代码吧，先看下如何创建数据库的代码，这一篇代码是不用挂到任何对象上面去的，你只用把它当成一个工具即可。如下所示：**

/\* Javascript class for accessing SQLite objects.

To use it, you need to make sure you COPY Mono.Data.SQLiteClient.dll from wherever it lives in your Unity directory

to your project's Assets folder

Originally created by dklompmaker in 2009

http://forum.unity3d.com/threads ... sier-Database-Stuff

Modified 2011 by Alan Chatham \*/

//#pragma strict

/\*

**代码描述**

\*本代码是为了在Windows环境下运行unity3d和Sqlite数据库而写的；实现的基本功能是unity3d能够与数据库之间进行基本的通信，比如说：在数据库中的数据被改变了以后，unity3d中得到的数据也会在刷新了之后跟着改变；这只是一个基本的核心的技术，为的是能够应用在大型的unity3d项目中，能够存储场景中的项目的属性，在需要改变对象的属性或增加、减少等对象时能够很方便的用得上。要实现本代码。首先需要一些dll文件，一个是Mono.Data.SQLiteClient.dll，另外一个是sqlite3.dll，这些文件都能够在unity3d的安装目录中找得到。除此之外，还需要把这两个文件放在你的项目的这个路径下面：\Assets\Plugins\，没有Plugins文件夹就必须创建这个文件夹，然后将这两个dll文件放在该文件夹写。当然，如果你想能够在PC上面发布成可执行文件，还需要改动一些地方。在unity3d中的Play Setting ->Other Setting 中将Api Compatibility的等级改为.NET 2.0;那么这些操作做完了以后，如果你的代码写得没有问题，那么你就可以成功了。

细解释代码：

\*

\*/

import System.Data; // we import our data class 我们先导入我们的数据集

import Mono.Data.Sqlite; // we import sqlite 我们导入sqlite数据集，也就是Plugins文件夹下的那个dll文件

class dbAccess {

// variables for basic query access

private var connection : String; //数据库的连接字符串，用于建立与特定数据源的连接

private var dbcon : IDbConnection; //IDbConnection的连接对象，其实就是一个类对象

private var dbcmd : IDbCommand; //IDbCommand类对象，用来实现操作数据库的命令:注解:我在网上资料看到的如何实现对数据库执行命令：

//首先创建一个IDbConnection连接对象，然后将一条数据库命令赋值给一个字符串，利用这个字符串和连接对象

//就可以创建(new)一个IDbCommand对象了，然后使用提供的方法就可以执行这个命令了。

private var reader : IDataReader; //reader的作用就是读取结果集的一个或多个只进结果流

function OpenDB(p : String){

connection = "URI=file:" + p; // we set the connection to our database

dbcon = new SqliteConnection(connection);

dbcon.Open(); //打开数据库连接操作

}

function BasicQuery(q : String, r : boolean){ // run a baic Sqlite query

dbcmd = dbcon.CreateCommand(); // create empty command

dbcmd.CommandText = q; // fill the command

reader = dbcmd.ExecuteReader(); // execute command which returns a reader 返回IDataReader的对象，创建IDataReader的对象

if(r){ // if we want to return the reader

return reader; // return the reader 返回读取的对象，就是读到了什么东西

}

}

// This returns a 2 dimensional ArrayList with all the

// data from the table requested

function ReadFullTable(tableName : String){

var query : String;

query = "SELECT \* FROM " + tableName;

dbcmd = dbcon.CreateCommand();

dbcmd.CommandText = query;

reader = dbcmd.ExecuteReader();

var readArray = new ArrayList();

while(reader.Read()){

var lineArray = new ArrayList();

for (var i = 0; i < reader.FieldCount; i++)

lineArray.Add(reader.GetValue(i)); // This reads the entries in a row

readArray.Add(lineArray); // This makes an array of all the rows

}

return readArray; // return matches

}

// This function deletes all the data in the given table. Forever. WATCH OUT! Use sparingly, if at all

function DeleteTableContents(tableName : String){

var query : String;

query = "DELETE FROM " + tableName;

dbcmd = dbcon.CreateCommand();

dbcmd.CommandText = query;

reader = dbcmd.ExecuteReader();

}

function CreateTable(name : String, col : Array, colType : Array){ // Create a table, name, column array, column type array

var query : String;

query = "CREATE TABLE " + name + "(" + col[0] + " " + colType[0];

for(var i=1; i<col.length; i++){

query += ", " + col *+ " " + colType;*

*}*

*query += ")";*

*dbcmd = dbcon.CreateCommand(); // create empty command*

*dbcmd.CommandText = query; // fill the command*

*reader = dbcmd.ExecuteReader(); // execute command which returns a reader*

*}*

*function InsertIntoSingle(tableName : String, colName : String, value : String){ // single insert*

*var query : String;*

*query = "INSERT INTO " + tableName + "(" + colName + ") " + "VALUES (" + value + ")";*

*dbcmd = dbcon.CreateCommand(); // create empty command*

*dbcmd.CommandText = query; // fill the command*

*reader = dbcmd.ExecuteReader(); // execute command which returns a reader*

*}*

*function InsertIntoSpecific(tableName : String, col : Array, values : Array){ // Specific insert with col and values*

*var query : String;*

*query = "INSERT INTO " + tableName + "(" + col[0];*

*for(var i=1; i<col.length; i++){*

*query += ", " + col;*

*}*

*query += ") VALUES (" + values[0];*

*for(i=1; i<values.length; i++){*

*query += ", " + values;*

*}*

*query += ")";*

*dbcmd = dbcon.CreateCommand();*

*dbcmd.CommandText = query;*

*reader = dbcmd.ExecuteReader();*

*}*

*function InsertInto(tableName : String, values : Array){ // basic Insert with just values*

*var query : String;*

*query = "INSERT INTO " + tableName + " VALUES (" + values[0];*

*for(var i=1; i<values.length; i++){*

*query += ", " + values;*

*}*

*query += ")";*

*dbcmd = dbcon.CreateCommand();*

*dbcmd.CommandText = query;*

*reader = dbcmd.ExecuteReader();*

*}*

*// This function reads a single column*

*// wCol is the WHERE column, wPar is the operator you want to use to compare with,*

*// and wValue is the value you want to compare against.*

*// Ex. - SingleSelectWhere("puppies", "breed", "earType", "=", "floppy")*

*// returns an array of matches from the command: SELECT breed FROM puppies WHERE earType = floppy;*

*function SingleSelectWhere(tableName : String, itemToSelect : String, wCol : String, wPar : String, wValue : String){ // Selects a single Item*

*var query : String;*

*query = "SELECT " + itemToSelect + " FROM " + tableName + " WHERE " + wCol + wPar + wValue;*

*dbcmd = dbcon.CreateCommand();*

*dbcmd.CommandText = query;*

*reader = dbcmd.ExecuteReader();*

*var readArray = new Array();*

*while(reader.Read()){*

*readArray.Push(reader.GetString(0)); // Fill array with all matches*

*}*

*return readArray; // return matches*

*}*

*function CloseDB(){*

*reader.Close(); // clean everything up*

*reader = null;*

*dbcmd.Dispose();*

*dbcmd = null;*

*dbcon.Close();*

*dbcon = null;*

*}*

*}*

7、如何在Unity3D中使用这个数据库的代码：

*//#pragma strict*

*/\* Script for testing out SQLite in Javascript*

*2011 - Alan Chatham*

*Released into the public domain*

*This script is a GUI script - attach it to your main camera.*

*It creates/opens a SQLite database, and with the GUI you can read and write to it.*

*\*/*

*// This is the file path of the database file we want to use*

*// Right now, it'll load TestDB.sqdb in the project's root folder.*

*// If one doesn't exist, it will be automatically created.*

*public var DatabaseName : String = "TestDB.sqdb";*

*// This is the name of the table we want to use*

*public var TableName : String = "TestTable";*

*var db : dbAccess;*

*function Start(){*

*// Give ourselves a dbAccess object to work with, and open it*

*db = new dbAccess();*

*db.OpenDB(DatabaseName);*

*// Let's make sure we've got a table to work with as well!*

*var tableName = TableName;*

*var columnNames = new Array("firstName","lastName");*

*var columnValues = new Array("text","text");*

*try {db.CreateTable(tableName,columnNames,columnValues);*

*}*

*catch(e){// Do nothing - our table was already created判断表是否被创建了*

*//- we don't care about the error, we just don't want to see it*

*}*

*}*

*// These variables just hold info to display in our GUI*

*var firstName : String = "First Name";*

*var lastName : String = "Last Name";*

*var DatabaseEntryStringWidth = 100;*

*var scrollPosition : Vector2;*

*var databaseData : ArrayList = new ArrayList();*

*// This GUI provides us with a way to enter data into our database*

*// as well as a way to view it*

*function OnGUI(){*

*GUI.Box(Rect (25,25,Screen.width - 50, Screen.height - 50),"Data");*

*GUILayout.BeginArea(Rect(50, 50, Screen.width - 100, Screen.height - 100));*

*// This first block allows us to enter new entries into our table*

*GUILayout.BeginHorizontal();*

*firstName = GUILayout.TextField(firstName, GUILayout.Width (DatabaseEntryStringWidth));*

*lastName = GUILayout.TextField(lastName, GUILayout.Width (DatabaseEntryStringWidth));*

*//lastName = GUILayout.TextField();*

*GUILayout.EndHorizontal();*

*if (GUILayout.Button("Add to database")){*

*// Insert the data*

*InsertRow(firstName,lastName);*

*// And update the readout of the database*

*databaseData = ReadFullTable();*

*}*

*// This second block gives us a button that will display/refresh the contents of our database*

*GUILayout.BeginHorizontal();*

*if (GUILayout.Button ("Read Database"))*

*databaseData = ReadFullTable();*

*if (GUILayout.Button("Clear"))*

*databaseData.Clear();*

*GUILayout.EndHorizontal();*

*GUILayout.Label("Database Contents");*

*scrollPosition = GUILayout.BeginScrollView(scrollPosition, GUILayout.Height(100));*

*for (var line : ArrayList in databaseData){*

*GUILayout.BeginHorizontal();*

*for (var s in line){*

*GUILayout.Label(s.ToString(), GUILayout.Width(DatabaseEntryStringWidth));*

*}*

*GUILayout.EndHorizontal();*

*}*

*GUILayout.EndScrollView();*

*if (GUILayout.Button("Delete All Data")){*

*DeleteTableContents();*

*databaseData = ReadFullTable();*

*}*

*GUILayout.EndArea();*

*}*

*// Wrapper function for inserting our specific entries into our specific database and table for this file*

*function InsertRow(firstName, lastName){*

*var values = new Array(("'"+firstName+"'"),("'"+lastName+"'"));*

*db.InsertInto(TableName, values);*

*}*

*// Wrapper function, so we only mess with our table.*

*function ReadFullTable(){*

*return db.ReadFullTable(TableName);*

*}*

*// Another wrapper function...*

*function DeleteTableContents(){*

*db.DeleteTableContents(TableName);*

*}*

**运行结果：**



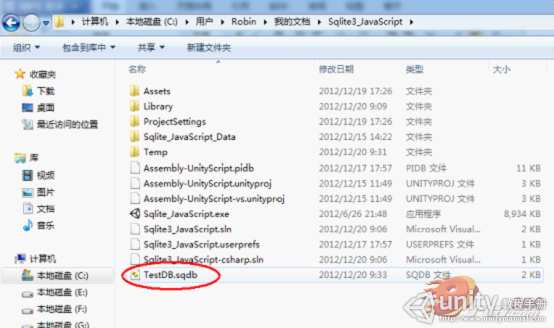
Unity3D教程：Unity3D与Sqlite数据库直连

这是在Unity3D中运行的结果，数据的操作结果如下：

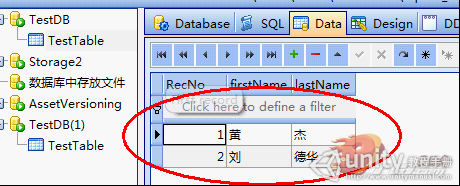


Unity3D教程：Unity3D与Sqlite数据库直连

我们看见了数据的操作能够成功，经过测试，其他的Button也都能出现相对应的效果，那我们再看看这个到底有没有生成我们想要的数据库文件：



Unity3D教程：Unity3D与Sqlite数据库直连



Unity3D教程：Unity3D与Sqlite数据库直连

文件当中数据：经测试，我们在对数据库中的数据进行操作的时候，我们的Unity3D中的数据也会发生相应的改变了!