# **[html 5 本地数据库（Web Sql Database）核心方法openDatabase、transaction、executeSql 详解](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/3003698.html)**

Web SQL数据库API实际上不是HTML5规范的组成部分，而是单独的规范。它通过一套API来操纵客户端的数据库。Safari、Chrome、Firefox、Opera等主流浏览器都已经支持Web SQL Database。HTML5的Web SQL Databases的确很诱惑人，当你发现可以用与mysql查询一样的查询语句来操作本地数据库时，你会发现这东西挺有趣的。今天，我们一起来了解HTML 5的Web SQL Database API。  
  
下面将一一将介绍怎样创建打开数据库，创建表，添加数据，更新数据，删除数据，删除表 。  
  
先介绍三个核心方法  
  
1、openDatabase：这个方法使用现有数据库或创建新数据库创建数据库对象。  
  
2、transaction：这个方法允许我们根据情况控制事务提交或回滚。  
  
3、executeSql：这个方法用于执行真实的SQL查询。

第一步：打开连接并创建数据库

var dataBase = openDatabase("student", "1.0", "学生表", 1024 \* 1024, function () { });if (!dataBase) {

alert("数据库创建失败！");

} else {

alert("数据库创建成功！");

}

解释一下openDatabase方法打开一个已经存在的数据库，如果数据库不存在，它还可以创建数据库。几个参数意义分别是：  
1，数据库名称。  
2，版本号 目前为1.0,不管他，写死就OK。  
3，对数据库的描述。  
4，设置数据的大小。  
5，回调函数(可省略)。  
初次调用时创建数据库，以后就是建立连接了。  
创建的数据库就存在本地，路径如下：  
C:\Users\Administrator\AppData\Local\Google\Chrome\User Data\Default\databases\http\_localhost\_\* 。  
创建的是一个sqllite数据库，可以用SQLiteSpy打开文件，可以看到里面的数据。SQLiteSpy是一个绿色软件，可以百度一下下载地址或SQLiteSpy官方下载：[SQLiteSpy](http://wikitaxi.org/delphi/lib/exe/fetch.php?hash=046523&media=http://www.yunqa.de/delphi/downloads/SQLiteSpy_1.9.1.zip" \t "http://www.cnblogs.com/wshiqtb/archive/2013/04/07/_blank)。

第二步：创建数据表

[IMG_256](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

this.createTable=function() {

dataBase.transaction( function(tx) {

tx.executeSql("create table if not exists stu (id REAL UNIQUE, name TEXT)",

[],

function(tx,result){ alert('创建stu表成功'); },

function(tx, error){ alert('创建stu表失败:' + error.message);

});

});

}

[IMG_257](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

解释一下，  
executeSql函数有四个参数，其意义分别是：  
1）表示查询的字符串，使用的SQL语言是SQLite 3.6.19。（必选）  
2）插入到查询中问号所在处的字符串数据。（可选）  
3）成功时执行的回调函数。返回两个参数：tx和执行的结果。（可选）  
4）一个失败时执行的回调函数。返回两个参数：tx和失败的错误信息。（可选）

第三步：执行增删改查

1）添加数据：

[IMG_258](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

this.insert = function () {

dataBase.transaction(function (tx) {

tx.executeSql("insert into stu (id, name) values(?, ?)",

[id, '徐明祥'],

function () { alert('添加数据成功'); },

function (tx, error) { alert('添加数据失败: ' + error.message);

} );

});

[IMG_259](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

 2）查询数据

[IMG_260](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

this.query = function () {

dataBase.transaction(function (tx) {

tx.executeSql("select \* from stu", [],

function (tx, result) { //执行成功的回调函数//在这里对result 做你想要做的事情吧...........},

function (tx, error) {

alert('查询失败: ' + error.message);

} );

});

}

[IMG_261](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

特别提醒  
上面代码中执行成功的回调函数有一参数result。   
  
result：查询出来的数据集。其数据类型为 SQLResultSet ，就如同C#中的DataTable。   
SQLResultSet 的定义为：

interface SQLResultSet {readonly attribute long insertId;readonly attribute long rowsAffected;readonly attribute SQLResultSetRowList rows;

};

其中最重要的属性—SQLResultSetRowList 类型的 rows 是数据集的“行” 。   
rows 有两个属性：length、item 。  
故，获取查询结果的第一行列名为name的值 ：result.rows.item(0).name  。

3）更新数据

[IMG_262](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

this.update = function (id, name) {

dataBase.transaction(function (tx) {

tx.executeSql("update stu set name = ? where id= ?",

[name, id],

function (tx, result) {

},

function (tx, error) {

alert('更新失败: ' + error.message);

});

});

}

[IMG_263](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

4）删除数据

[IMG_264](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

this.del = function (id) {

dataBase.transaction(function (tx) {

tx.executeSql("delete from stu where id= ?",

[id],

function (tx, result) {

},

function (tx, error) {

alert('删除失败: ' + error.message);

});

});

}

[IMG_265](http://www.cnblogs.com/wshiqtb/archive/2013/04/07/javascript:void(0);)

5）删除数据表

this.dropTable = function () {

dataBase.transaction(function (tx) {

tx.executeSql('drop table stu');

});

}