

# Arlexovincy的专栏

☰ 目录视图

☰ 摘要视图

**RSS** 订阅

## 个人资料



Arlexovincy

访问： 470次  
积分： 15分  
排名： 千里之外

原创： 1篇    转载： 1篇  
译文： 0篇    评论： 0条

## 文章搜索

## 文章分类

数据库 (1)  
IOS (1)  
UIResponder (1)

## 文章存档

2013年07月 (1)  
2013年02月 (1)

## 阅读排行

FMDatabase的用法介绍 (330)  
UIResponder响应链 找到 (88)

## 评论排行

FMDatabase的用法介绍 (0)  
UIResponder响应链 找到 (0)

## 推荐文章

[有奖征资源，博文分享有内涵](#)    [5月推荐博文汇总](#)    [第二届战神杯编程挑战月赛](#)    [2014 CSDN博文大赛](#)    [技术牛人的蜕变之路-陆其明](#)

## FMDatabase的用法介绍

分类： 数据库    2013-02-11 20:03    330人阅读    评论(0)    收藏    举报

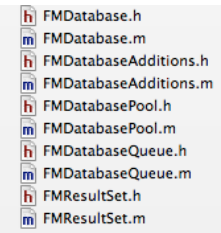
FMDatabase    FMDaBase    FMDatabase    ios    iOS    IOS    sqlite    SQLite    Sqlite    数据库

本人以后都会不定时的写一些关于IOS开发的教程，大家希望可提些意见，大家希望写些什么，对大家有帮助的，希望帮到大家：

FMDatabase其实是一个很轻级的sqlite数据库封装库，用Object – C封装C，它主要有2个类：FMDatabase和FMResultSet

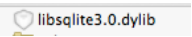
要使用它首先就要下载这个开源库，它的下载地址是：<https://github.com/ccgus/fmdb>

下载完成后，我们就可以开始使用它了：



它主要有这些文件

然后，就需要添加sqlite框架



这样，你的工程就能开始使用FMDatabase了。

下面就介绍一下它的用法：

1.首先我们需要创建一个FMDatabase实例：

```
+(FMDatabase*)DataBaseSigonInstance
{
    //数据库初始化
    NSString *homeDir = [NSString homeDirectory];
    //NSLog(@"%@",homeDir);
    NSString *dbPath = [homeDir stringByAppendingPathComponent:@"Library/Caches
```

```
/FMDatabaseDemo.db"];  
    FMDatabase *database = [FMDatabaseWithPath:dbPath];  
  
    return database;  
}
```

2.接下来就是需要创建一个数据表了:

```
//打开数据库  
FMDatabase *db = [FMDatabasePathDatabaseSigonInstance];  
  
if (!db.open)  
{  
    NSLog(@"fail to open database");  
  
    return;  
}  
db.logsErrors =YES;//开启错误日志  
  
[db executeUpdate:@"CREATE TABLE IF NOT EXISTS studentTable(name varchar,age  
integer,totalScore float)"];  
  
[dbclose];
```

3.添加数据:

```
FMDatabase *db = [FMDatabasePathDatabaseSigonInstance];  
if (!db.open)  
{  
    NSLog(@"fail to open database");  
  
    return;  
}  
  
//获得数据  
student.name = ((UITextField*)[alertViewviewWithTag:TEXTFIELD_TAG]).text;  
student.age = [((UITextField*)[alertViewviewWithTag:TEXTFIELD_TAG +  
1]).textintValue];  
student.totalScore = [((UITextField*)[alertViewviewWithTag:TEXTFIELD_TAG +  
2]).textfloatValue];  
  
db.logsErrors =YES;  
  
[db executeUpdate:@"insert into  
studentTable(name,age,totalScore)values(?,?,?)",student.name,  
[NSNumber numberWithInt:student.age],[NSNumber numberWithFloat:student.totalScore]];//  
注意参数必须是对象  
  
[dbclose];
```

4.删除数据:

```
FMDatabase *db = [FMDatabasePathDatabaseSigonInstance];
```

```
if (!db.open)
{
    NSLog(@"fail to open!");
    return;
}

db.logsErrors =YES;

NSString *delDataSqlStr=@"delete from studentTable";
[dbexecuteUpdate:delDataSqlStr];

[dbclose];
```

5.修改数据:

```
FMDatabase *db = [FMDataBasePathDataBaseSigonInstance];
if (!db.open)
{
    NSLog(@"fail to open!");
    return;
}

db.logsErrors =YES;

[db executeUpdate:@"update studentTable set totalScore = 100"];
[dbclose];
```

查询数据:

```
FMDatabase *db = [FMDataBasePathDataBaseSigonInstance];
if (!db.open)
{
    NSLog(@"fail to open!");
    return;
}

db.logsErrors =YES;

FMResultSet *rs = [dbexecuteQuery:@"select * from studentTable"];

NSMutableArray *studentArr = [[NSMutableArrayalloc]init];

while ([rsnext])
{
    NSMutableDictionary *dic = [[NSMutableDictionaryalloc]init];

    [dic setValue:[rsstringForColumn:@"name"] forKey:@"name"];
    [dic setValue:[NSNumber numberWithInt:[rsintForColumn:@"age"]] forKey:@"age"];
    [dic setValue:[NSNumber numberWithInt:[rsdoubleForColumn:@"totalScore"]] forKey:@"totalScore"];

    [studentArraddObject:dic];
}
```

```
[dicrelease];

}

for (int count = 0; count< [studentArrcount]; count++)
{
    NSLog(@"student:%@ age:%@ totalScore:%@",
[[studentArrobjectAtIndex:count]objectForKey:@"name"],
[[studentArrobjectAtIndex:count] objectForKey:@"age"],
[[studentArrobjectAtIndex:count]objectForKey:@"totalScore"]);
}

[studentArrrelease];

[dbclose];
```

上面的都是一些增删改查的用法

下面我们再看看FMDatabase 和 FMResultSet

FMDatabase: (主要用到这两个)

- (BOOL)executeUpdate:(NSString\*)sql, ...; 这个是用于执行无结果的sql语句的
- (FMResultSet \*)executeQuery:(NSString\*)sql, ...; 这个是用于执行有结果的sql语句的

FMResultSet:

- (BOOL)next; 结果集的下一个
- (int)columnCount; 结果集的字段个数
- (int)intForColumn:(NSString\*)columnName; 获取整形字段的信息 (下面的都是同理)
- (int)intForColumnIndex:(int)columnIdx;
- (long)longForColumn:(NSString\*)columnName;
- (long)longForColumnIndex:(int)columnIdx;
- (longlongint)longLongIntForColumn:(NSString\*)columnName;
- (longlongint)longLongIntForColumnIndex:(int)columnIdx;
- (unsignedlonglong int)unsignedLongLongIntForColumn:(NSString\*)columnName;
- (unsignedlonglong int)unsignedLongLongIntForColumnIndex:(int)columnIdx;
- (BOOL)boolForColumn:(NSString\*)columnName;
- (BOOL)boolForColumnIndex:(int)columnIdx;
- (double)doubleForColumn:(NSString\*)columnName;
- (double)doubleForColumnIndex:(int)columnIdx;
- (NSString\*)stringForColumn:(NSString\*)columnName;
- (NSString\*)stringForColumnIndex:(int)columnIdx;
- (NSDate\*)dateForColumn:(NSString\*)columnName;
- (NSDate\*)dateForColumnIndex:(int)columnIdx;

- (NSData\*)dataForColumn:(NSString\*)columnName; 获取二进制数据（图片可以使用这个）
- (NSData\*)dataForColumnIndex:(int)columnIdx;

更多 0

下一篇 [UIResponder响应链 找到需要的视图或controller](#)

主题推荐    [ios开发](#)    [NSMutableArray](#)    [数据库](#)    [二进制](#)    [NSDate](#)

猜你在找

- [河北软件职业技术学院怎么样？CSDN乐知学院](#)
- [iOS中创建自定义的圆角按钮](#)
- [UINavigationController 导航条上自定义界面](#)
- [Core Data数据持久性存储基础教程](#)
- [UIColor和UIImage转换](#)
- [ios中JSONKit对字符串里换行符的处理 ——NSData解析](#)
- [EGORefreshTableHeaderView 使用方法 自我助记](#)
- [iOS button auto resize](#)
- [cocos2d-x 模拟iOS时间选择器效果](#)
- [ios地图开发之-OpenStreetMap基础教程](#)



[查看评论](#)

暂无评论

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题

[Java](#) [VPN](#) [Android](#) [iOS](#) [ERP](#) [IE10](#) [Eclipse](#) [CRM](#) [JavaScript](#) [Ubuntu](#) [NFC](#)  
[WAP](#) [jQuery](#) [数据库](#) [BI](#) [HTML5](#) [Spring](#) [Apache](#) [Hadoop](#) [.NET](#) [API](#) [HTML](#) [SDK](#) [IIS](#)  
[Fedora](#) [XML](#) [LBS](#) [Unity](#) [Splashtop](#) [UML](#) [components](#) [Windows Mobile](#) [Rails](#) [QEMU](#) [KDE](#)  
[Cassandra](#) [CloudStack](#) [FTC](#) [coremail](#) [OPhone](#) [CouchBase](#) [云计算](#) [iOS6](#) [Rackspace](#) [Web App](#)  
[SpringSide](#) [Maemo](#) [Compuware](#) [大数据](#) [apttech](#) [Perl](#) [Tornado](#) [Ruby](#) [Hibernate](#) [ThinkPHP](#)  
[Spark](#) [HBase](#) [Pure](#) [Solr](#) [Angular](#) [Cloud Foundry](#) [Redis](#) [Scala](#) [Django](#) [Bootstrap](#)