

# FMDB

---

- 什么是FMDB

- FMDB是iOS平台的SQLite数据库框架
- FMDB以OC的方式封装了SQLite的C语言API

- FMDB的优点

- 使用起来更加面向对象，省去了很多麻烦、冗余的C语言代码
- 对比苹果自带的Core Data框架，更加轻量级和灵活
- 提供了多线程安全的数据库操作方法，有效地防止数据混乱

- FMDB的github地址

- <https://github.com/ccgus/fmdb>

# 核心类

---

- FMDB有三个主要的类

- FMDatabase

- ✓ 一个FMDatabase对象就代表一个单独的SQLite数据库
- ✓ 用来执行SQL语句

- FMResultSet

- ✓ 使用FMDatabase执行查询后的结果集

- FMDatabaseQueue

- ✓ 用于在多线程中执行多个查询或更新，它是线程安全的

# 打开数据库

---

- 通过指定SQLite数据库文件路径来创建FMDatabase对象

```
FMDatabase *db = [FMDatabase databaseWithPath:path];  
if (![db open]) {  
    NSLog(@"数据库打开失败! ");  
}
```

- 文件路径有三种情况

- p 具体文件路径

- ✓ 如果不存在会自动创建

- p 空字符串@""

- ✓ 会在临时目录创建一个空的数据库

- ✓ 当FMDatabase连接关闭时，数据库文件也被删除

- p nil

- ✓ 会创建一个内存中临时数据库，当FMDatabase连接关闭时，数据库会被销毁

# 执行更新

---

- 在FMDB中，除查询以外的所有操作，都称为“更新”

p create、drop、insert、update、delete等

- 使用executeUpdate:方法执行更新

p – (BOOL)executeUpdate:(NSString\*)sql, ...

p – (BOOL)executeUpdateWithFormat:(NSString\*)format, ...

p – (BOOL)executeUpdate:(NSString\*)sql  
withArgumentsInArray:(NSArray \*)arguments

- 示例

```
[db executeUpdate:@"UPDATE t_student SET age = ? WHERE name = ?;", @20, @"Jack"]
```

# 执行查询

---

- 查询方法

- p – (FMResultSet \*)executeQuery:(NSString\*)sql, ...
- p – (FMResultSet \*)executeQueryWithFormat:(NSString\*)format, ...
- p – (FMResultSet \*)executeQuery:(NSString \*)sql  
withArgumentsInArray:(NSArray \*)arguments

- 示例

// 查询数据

```
FMResultSet *rs = [db executeQuery:@"SELECT * FROM t_student"];
```

// 遍历结果集

```
while ([rs next]) {  
    NSString *name = [rs stringForColumn:@"name"];  
    int age = [rs intForColumn:@"age"];  
    double score = [rs doubleForColumn:@"score"];  
}
```

# FMDatabaseQueue

---

- FMDatabase这个类是线程不安全的，如果在多个线程中同时使用一个FMDatabase实例，会造成数据混乱等问题
- 为了保证线程安全，FMDB提供方便快捷的FMDatabaseQueue类
- FMDatabaseQueue的创建

```
FMDatabaseQueue *queue = [FMDatabaseQueue  
databaseQueueWithPath:path];
```

# FMDatabaseQueue

---

- 简单使用

```
[queue inDatabase:^(FMDatabase *db) {  
    [db executeUpdate:@"INSERT INTO t_student(name) VALUES (?)", @"Jack"];  
    [db executeUpdate:@"INSERT INTO t_student(name) VALUES (?)", @"Rose"];  
    [db executeUpdate:@"INSERT INTO t_student(name) VALUES (?)", @"Jim"];  
  
    FMResultSet *rs = [db executeQuery:@"select * from t_student"];  
    while ([rs next]) {  
        // ...  
    }  
}];
```

# FMDatabaseQueue

---

- 使用事务

```
[queue inTransaction:^(FMDatabase *db, BOOL *rollback) {  
    [db executeUpdate:@"INSERT INTO t_student(name) VALUES (?)", @"Jack"];  
    [db executeUpdate:@"INSERT INTO t_student(name) VALUES (?)", @"Rose"];  
    [db executeUpdate:@"INSERT INTO t_student(name) VALUES (?)", @"Jim"];  
  
    FMResultSet *rs = [db executeQuery:@"select * from t_student"];  
    while ([rs next]) {  
        // ...  
    }  
}];
```

- 事务回滚

```
*rollback = YES;
```