



3G 2:40 PM

## data Store ～帰ってきたSQL～

---

日本Androidの会 事務局長  
SIPropプロジェクト 代表  
グロースエクスパートナーズ

今村 謙之 (いまむら のりつな)





2:40 PM

## アジェンダ

- SQLiteにさわる
  - create table文
  - insert文
  - select文
  - delete文
- Androidから、生SQLを叩く
  - SQLiteOpenHelper





2:40 PM

宣伝にやってきました

## ● とある本の監訳やります

### ● 監訳

- 日本Androidの会
- 本日の講師も担当しています。

### ● 出版元

- 才●●●社

### ● 発売

- 5月目標



2:40 PM

## Androidの永続性

- SQLiteが使われている

- 軽い！
- 簡単！
- 無料！

- 基本的なSQL文に対応

- insert文
- select文
- delete文





2:40 PM

## SQLiteを試してみる

1. エミュレータを起動します。
2. コマンドラインから、下記を入力

```
> adb shell
```

※ android内のシェルに入ります。

```
# sqlite3
```

※ SQLite内のシェルに入ります

```
sqlite>
```

※ SQLiteコマンドを入力できます。



2:40 PM

DDL文(データ定義言語)

## ● データベースを操作するための文(コマンド)

```
sqlite> create table mytable (  
  _id integer primary key autoincrement,  
  name text,  
  phone text );
```

※ 1行で入力となります。

```
sqlite> drop table mytable
```

※ 削除されます。



2:40 PM

## 基本3兄弟

### ● テーブルを操作するための文(コマンド)

```
sqlite> insert into mytable values(null,  
    'noritsuna', '0611111111');
```

```
sqlite> select * from mytable where  
    name='noritsuna';
```

```
1|noritsuna|0611111111
```

※ こんな結果が返ってくる。

```
sqlite> delete from mytable where  
    name='noritsuna';
```





2:40 PM

Androidから扱うには？

## ● SQLiteOpenHelper を、利用する。

```
public class EventsData extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "events.db";
    private static final int DATABASE_VERSION = 1;

    /** Create a helper object for the Events database */
    public EventsData(Context ctx) {
        super(ctx, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE " + TABLE_NAME + " (" + _ID
            + " INTEGER PRIMARY KEY AUTOINCREMENT, " + TIME
            + " INTEGER, " + TITLE + " TEXT NOT NULL);");
    }
    (略)
}
```





2:40 PM

帰ってきた、3兄弟

## ● insert文

- SQLiteOpenHelper.insert(table, nullColumnHack, values);

## ● select文

- SQLiteOpenHelper.query(table, columns, selection, selectionArgs, groupBy, having, orderBy);

## ● delete文

- SQLiteOpenHelper.delete(table, whereClause, whereArgs);





2:40 PM

たまには、思い出してあげてください

## ● Queryと言ったら、Cursor ! ! !

```
Cursor cursor = db.query(TABLE_NAME, FROM, null, null, null,  
    null, ORDER_BY);
```

```
while (cursor.moveToNext()) {  
    // Could use getColumnIndexOrThrow() to get indexes  
    long id = cursor.getLong(0);  
    long time = cursor.getLong(1);  
    String title = cursor.getString(2);  
    builder.append(id).append(": ");  
    builder.append(time).append(": ");  
    builder.append(title).append("¥n");  
}
```



2:40 PM

Androidでデータベースをさわるための神髄

## ● データバインド

- Viewとデータベースを簡単に接続するための仕掛け

- アクティブレコードのような仕掛けで連携できる

- `<TextView android:id="@+id/name"/>`

## ● ContentProvider

- 他のアプリケーションから、自分のデータベースにアクセスさせるための仕掛け

- RESTfulのような仕掛けでアクセスさせる

- `content://myProgram/mytable/1`



2:40 PM

続きは、こちらで。

## ● とある本の監訳やります

### ● 監訳

- 日本Androidの会
- 本日の講師も担当しています。

### ● 出版元

- 才●●●社

### ● 発売

- 5月目標