

Hướng dẫn lập trình cơ bản với Android

Part 20

Các bước thực hiện với CSDL:

1. Tạo 1 CSDL (thông thường chỉ cần làm 1 lần)
2. Mở CSDL đó
3. Thêm giá trị vào trong table
4. Truy vấn.
5. Đóng CSDL

1. Tạo cơ sở dữ liệu.

Đầu tiên các bạn add 1 class DBAdapter để xử lý tất cả các thao tác liên quan đến CSDL.

Mã:

```
public static final String KEY_ID = "_id";
public static final String KEY_NAME = "name";

private DatabaseHelper mDbHelper;
private SQLiteDatabase mDB;

private static final String DATABASE_CREATE =
"create table users (_id integer primary key
autoincrement, "
    + "name text not null);";
private static final String DATABASE_NAME =
"Database_Demo";
private static final String DATABASE_TABLE =
"users";
private static final int DATABASE_VERSION = 2;

private final Context mContext;
```

Tạo 1 lớp bên trong DBAdapter được extend từ lớp SQLiteOpenHelper, override 2 phương thức onCreate() và onUpgrade() để quản lý việc tạo CSDL và version của CSDL đó.

Mã:

```
private static class DatabaseHelper extends
SQLiteOpenHelper{

    public DatabaseHelper(Context context,
String name,
CursorFactory factory, int
version) {
        super(context, name, factory, version);
        // TODO Auto-generated constructor stub
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        // TODO Auto-generated method stub
        db.execSQL(DATABASE_CREATE);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int
oldVersion, int newVersion) {
        // TODO Auto-generated method stub
        Log.i(TAG, "Upgrading DB");
        db.execSQL("DROP TABLE IF EXISTS
users");
        onCreate(db);
    }
}
```

2. Mở CSDL :

Mã:

```
public DBAdapter open()
{
    mDbHelper = new DatabaseHelper(mContext,
DATABASE_NAME, null, DATABASE_VERSION);
    mDB = mDbHelper.getWritableDatabase();
    return this;
}
```

3. Thêm giá trị vào CSDL

Mã:

```
public long createUser(String name){
    ContentValues inititalValues = new
ContentValues();
    inititalValues.put(KEY_NAME, name);
    return mDB.insert(DATABASE_TABLE, null,
inititalValues);
}
```

4. Truy vấn

Bạn có thể get toàn bộ data hoặc có thể get data theo ID (tiện cho việc chỉnh sửa hay cập nhật thông tin của từng bản ghi).

Mã:

```
public Cursor getAllUsers(){
    return mDB.query(DATABASE_TABLE, new
String[] {KEY_ID, KEY_NAME}, null, null, null, null,
null);
}
```

Còn rất nhiều các thao tác như sửa, xóa, update.... bản ghi, các bạn có thể tự phát triển.

Tất cả các chức năng đó đều được cung cấp bởi lớp SQLiteDatabase, các bạn chỉ cần cụ thể hóa bằng các câu truy vấn là được.

5. Đóng CSDL

Mã:

```
public void close(){
    mDbHelper.close();
}
```

6. Sử dụng CSDL

Để test CSDL mà bạn vừa tạo, các bạn có thể thêm 1 vài dòng code để thêm 1 user và hiển thị CSDL lên màn hình thông qua lớp Activity ban đầu:

Ở đây mình create 1 user thông qua câu lệnh mDB.createUser("Username"); sau đó

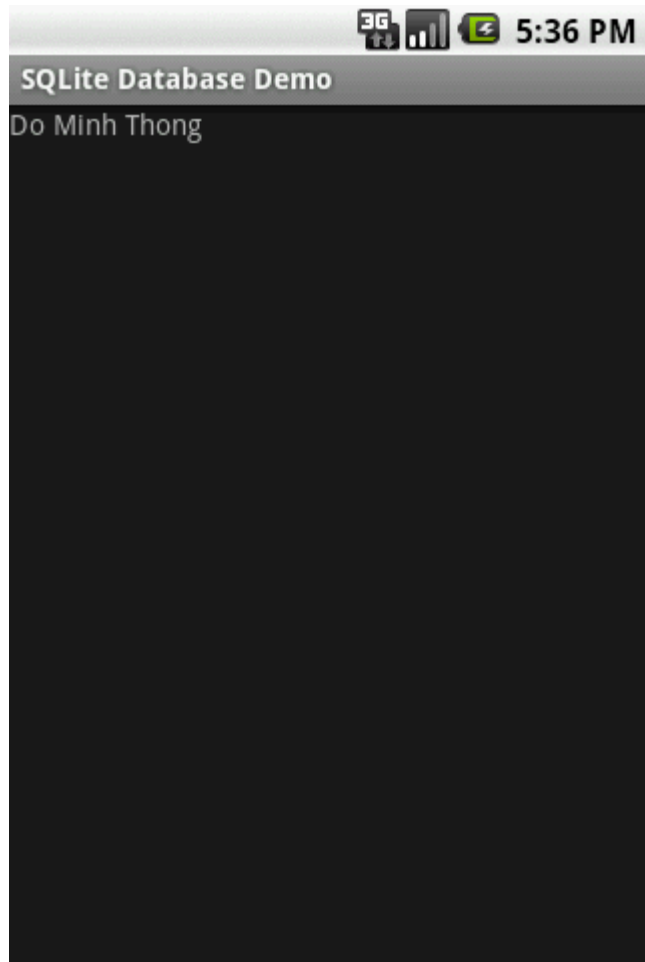
Mã:

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
}
```

```
        setContentView(R.layout.main);
        mDB = new DBAdapter(this);
        mDB.open();
        mDB.createUser("Do Minh Thong");
        getData();
    }

    private void getData(){
        mCursor = mDB.getAllUsers();
        startManagingCursor(mCursor);
        String[] from = new String[]{DBAdapter.KEY_NAME};
        int[] to = new int[] {R.id.text1};
        SimpleCursorAdapter users = new
SimpleCursorAdapter(this, R.layout.users_row, mCursor,
from, to);
        setListAdapter(users);
    }
}
```

Kết quả :



Sourcecode đầy đủ : <http://www.mediafire.com/?yzw2d1ijymo>

Bài này là bài cơ sở để mình viết tiếp Phần 7 : ContentProvider , các bạn chú ý theo dõi nhé