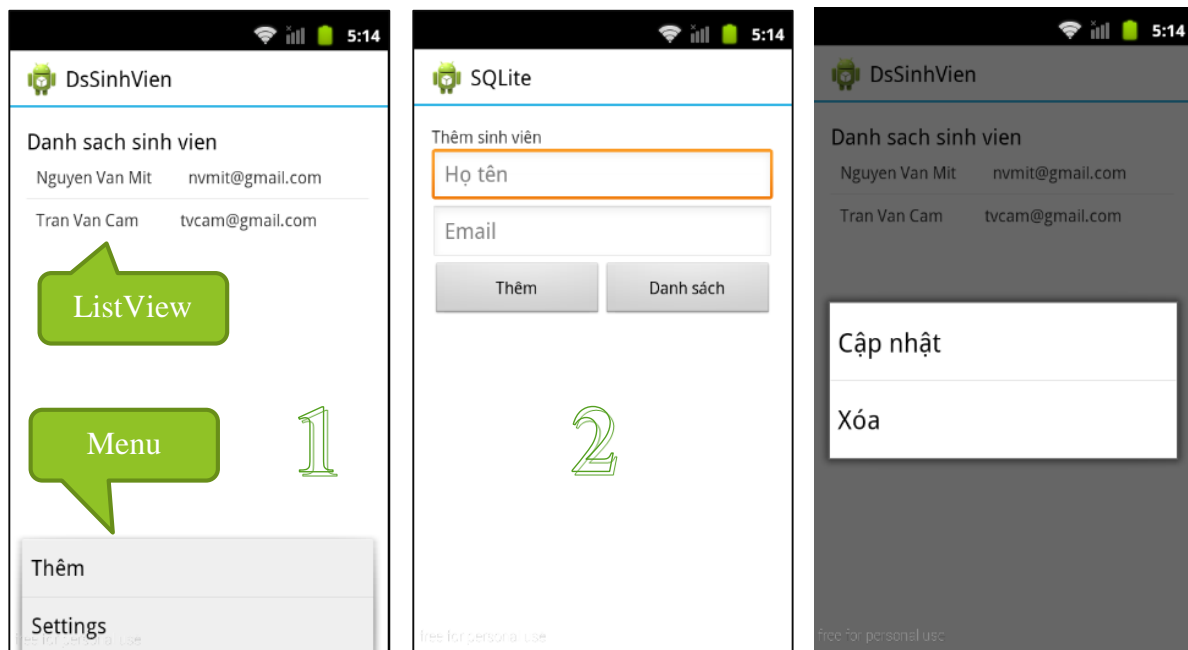


SQLite

Mục tiêu

1. Tạo được db SQLite
2. Đọc được dữ liệu từ bảng
3. Thay đổi nội dung trong một bảng
4. Sao lưu dữ liệu



Ứng dụng demo

1. Tạo ứng dụng tên android với tên SQLite
2. Tạo 2 Activity với tên “DsSinhVien” (1) và “ThemSinhVien” (2)
3. Tạo cơ sở dữ liệu “qlsv” với một bảng dữ liệu “sinhvien”: Tạo lớp **MySQLiteDB** thừa kế từ lớp **SQLiteOpenHelper**, hiệu chỉnh nội dung của lớp vừa tạo như sau:

```
public class MySQLiteDB extends SQLiteOpenHelper {
    static final String KEY_ROWID = "id";
    static final String KEY_NAME = "ten";
    static final String KEY_EMAIL = "email";
    static final String DATABASE_NAME = "qlsv";
    static final String DATABASE_TABLE = "sinhvien";
    static final String DATABASE_CREATE_TABLE = "" +
        "create table sinhvien(" +
        "id integer primary key autoincrement, " +
        "ten text not null, email text not null);";

    public MySQLiteDB(Context context) {
        super(context, DATABASE_NAME, null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(DATABASE_CREATE_TABLE);

        ContentValues initialValues = new ContentValues();
        initialValues.put(KEY_NAME, "Nguyen Van Mit");
        initialValues.put(KEY_EMAIL, "nvmit@gmail.com");
        db.insert(DATABASE_TABLE, null, initialValues);

        initialValues.clear();
        initialValues.put(KEY_NAME, "Tran Van Cam");
        initialValues.put(KEY_EMAIL, "tvcam@gmail.com");
        db.insert(DATABASE_TABLE, null, initialValues);

        initialValues.clear();
        initialValues.put(KEY_NAME, "Le Thi Luu");
        initialValues.put(KEY_EMAIL, "ltluu@gmail.com");
        db.insert(DATABASE_TABLE, null, initialValues);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("drop table if exists sinhvien " + DATABASE_TABLE);
        onCreate(db);
    }
}
```

4. Bổ sung thêm chức năng CRUD cho bảng dữ liệu **SinhVien** bằng cách bổ sung 5 hàm sau vào lớp trên.

- Định nghĩa lớp “SinhVien” bên trong lớp

```
public class SinhVien {
    public int id;
    public String ten;
    public String email;
}
```

- Bổ sung các phương thức

```
public Boolean ThemSinhVien(String ten, String email) {
    ContentValues initialValues = new ContentValues();
    initialValues.put(KEY_NAME, ten);
```

```

        initialValues.put(KEY_EMAIL, email);
        return this.getWritableDatabase().insert(DATABASE_TABLE, null,
            initialValues) > 0;
    }

    public Boolean CapNhatSinhVien(int id, String ten, String email) {
        ContentValues args = new ContentValues();
        args.put(KEY_NAME, ten);
        args.put(KEY_EMAIL, email);
        args.put(KEY_ROWID, id);
        return this.getWritableDatabase().update(DATABASE_TABLE, args,
            KEY_ROWID + "=" + id, null) > 0;
    }

    public Boolean XoaSinhVien(int id) {
        return this.getWritableDatabase().delete(DATABASE_TABLE,
            KEY_ROWID + "=" + id, null) > 0;
    }

    public SinhVien TimSinhVien(int id) {
        Cursor mCursor = this.getReadableDatabase().rawQuery(
            "Select id, ten, email from sinhvien wwwhere id="+id, null);
        if (mCursor == null) {
            return null;
        }
        // di chuyen den phan tu dau tien
        mCursor.moveToFirst();
        SinhVien sv = new SinhVien();
        sv.id = mCursor.getInt(0);
        sv.ten = mCursor.getString(1);
        sv.email = mCursor.getString(2);
        return sv;
    }

    // đọc hết dữ liệu
    public ArrayList<SinhVien> DanhSachSinhVien() {
        ArrayList<SinhVien> list = new ArrayList<SinhVien>();
        // mở dữ liệu để đọc
        Cursor cur = this.getReadableDatabase().rawQuery(
            "Select id, ten, email from sinhvien", null);
        // di chuyển về dòng đầu, ds rỗng sẽ trả về false
        if (!cur.moveToFirst())
            return null;

        do {
            SinhVien sv = new SinhVien();
            sv.id = cur.getInt(0);
            sv.ten = cur.getString(1);
            sv.email = cur.getString(2);
            list.add(sv);
        } while (cur.moveToNext());

        return list;
    }
}

```

5. Tạo lớp **MyAdapter** thừa kế từ **BaseAdapter** để tùy biến ds như sau

```

public class MyAdapter extends BaseAdapter {

    Context context;
    ArrayList<SinhVien> listAll;

    public MyAdapter(Context c, ArrayList<SinhVien> list) {
        context = c;
        listAll = list;
    }
}

```

```

    }

    @Override
    public int getCount() {
        return listAll.size();
    }

    @Override
    public Object getItem(int position) {
        return listAll.get(position);
    }

    @Override
    public long getItemId(int position) {
        return listAll.get(position).id;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        Mot_SinhVien sv;

        if (convertView == null) {
            LayoutInflater inflater = (LayoutInflater) context
                .getSystemService(Activity.LAYOUT_INFLATER_SERVICE);
            convertView = inflater.inflate(R.layout.mot_sinhvien, null);

            sv = new Mot_SinhVien();
            sv.ten = (TextView) convertView.findViewById(R.id.tvTen);
            sv.email = (TextView) convertView.findViewById(R.id.tvEmail);

            convertView.setTag(sv);
        } else {
            sv = (Mot_SinhVien) convertView.getTag();
        }

        sv.ten.setText(listAll.get(position).ten);
        sv.email.setText(listAll.get(position).email);

        return convertView;
    }

    class Mot_SinhVien {
        TextView ten;
        TextView email;
    }
}

```

6. Trong activity “DsSinhVien” hiệu chỉnh như sau

```

public class DsSinhVien extends ActionBarActivity {
    ListView lv;
    ArrayList<SinhVien> dsSinhVine = new ArrayList<SinhVien>();

    int idSinhVienChon = -1;

    MySQLiteDB db;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_ds_sinh_vien);

        lv = (ListView) findViewById(R.id.listView1);

        // lay du lieu
        db = new MySQLiteDB(this);
        dsSinhVine = db.DanhSachSinhVien();
    }
}

```

```

        if (dsSinhVien != null) {
            // khoi tao giao dien
            MyAdapter adp = new MyAdapter(this, dsSinhVien);
            lv.setAdapter(adp);

            registerForContextMenu(lv);
        } else {
            Toast.makeText(this, "Khong tim thay sinh vien nao",
                Toast.LENGTH_SHORT).show();
        }

        lv.setOnItemClickListener(new OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view,
                int position, long id) {
                idSinhVienChon = position;
            }
        });
    }
}

```

- Chạy ứng dụng chúng ta sẽ có như sau



vn.cusc.sqlite	2014-10-06
databases	2014-10-06
qlsv	5120 2014-10-06
lib	2014-10-06

7. Khi chọn menu thêm chuyển sang chức năng thêm mới. Hiệu chỉnh phương thức onOptionsItemSelected như sau

```

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    if (id == R.id.action_settings) {
        return true;
    }

    if (id == R.id.mnuThem) {
        Intent intent = new Intent(this, ThemSinhVien.class);
        startActivity(intent);
        finish();
        return true;
    }
}

```

```

        return super.onOptionsItemSelected(item);
    }

```

8. Hiệu chỉnh lại activity thêm mới như sau

```

public class ThemSinhVien extends ActionBarActivity implements OnClickListener {

    MySQLiteDB db;

    Button btnThem, btnDanhSach;
    EditText etTen, etEmail;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_them_sinh_vien);
        // lay du lieu
        db = new MySQLiteDB(this);

        btnThem = (Button) findViewById(R.id.btnThem);
        btnThem.setOnClickListener(this);
        btnDanhSach = (Button) findViewById(R.id.btnDanhSach);
        btnDanhSach.setOnClickListener(this);

        etTen = (EditText) findViewById(R.id.etTen);
        etEmail = (EditText) findViewById(R.id.etEmail);
    }

    @Override
    public void onClick(View v) {
        if(v.getId() == R.id.btnThem)
        {
            db.ThemSinhVien(etTen.getText().toString(),
            etEmail.getText().toString());
            Toast.makeText(this, "Sinh viên đã được lưu",
            Toast.LENGTH_SHORT).show();
        }
        if(v.getId() == R.id.btnDanhSach)
        {
            Intent intent = new Intent(this, DsSinhVien.class);
            startActivity(intent);
            finish();
        }
    }
}

```

- Chạy ứng dụng để kiểm tra kết quả

Xây dựng chức năng chọn một phần tử từ danh sách để xóa hay cập nhật

1. Tạo file context menu gồm hai mục xóa và sửa

```

<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools" >

    <item
        android:id="@+id/mnuXoa"
        android:orderInCategory="100"
        android:title="Xóa"
        app:showAsAction="ifRoom"/>
    <item
        android:id="@+id/mnuCapNhat"
        android:orderInCategory="1"
        android:showAsAction="ifRoom"

```

```
        android:title="Cập nhật"/>
    </menu>
```

2. Đăng ký hiển thị context menu khi người dùng chọn từ listview

```
registerForContextMenu(lv);
```

3. Định nghĩa chồng lại sự kiện chọn phần tử trên context menu

```
// khi chọn từ context menu
@Override
public boolean onContextItemSelected(MenuItem item) {
    // nếu chọn xóa
    if (item.getItemId() == R.id.mnuCapNhat) {
        AdapterContextMenuInfo info = (AdapterContextMenuInfo) item
            .getMenuInfo();
        View view = info.targetView.findViewById(R.id.tvTen);
        return true;
    } else {
        db = new MySQLiteDB(this);
        db.XoaSinhVien(idSinhVienChon);
        dsSinhVine = db.DanhSachSinhVien();
        if (dsSinhVine != null) {
            // khởi tạo giao diện
            MyAdapter adp = new MyAdapter(this, dsSinhVine);
            lv.setAdapter(adp);
        } else {
            Toast.makeText(this, "Không tìm thấy sinh viên nào",
                Toast.LENGTH_SHORT).show();

            lv.setAdapter(null);
        }
        return true;
    }
}
```

- Chạy ứng dụng, chọn giữ lâu trên sinh viên và chọn xóa.

Bài tập tự làm

1. Xây dựng chức năng cập nhật.
2. Bổ sung chức năng sao lưu dữ liệu cho ứng dụng

Hints:

- Tham khảo chức năng thêm và xóa
- Sao lưu chỉ là chép tập tin chứa nội dung dữ liệu SQLite