

先建一個自己的資料庫 CODE:

```
public void onCreate(SQLiteDatabase db) {
    final String SQL = "CREATE TABLE IF NOT EXISTS " + _TableName + "( " +
        ID_FIELD + " integer PRIMARY KEY AUTOINCREMENT, " +
        NAME_FIELD + " text, " +
        SEXUAL_FIELD + " text, " +
        ADDRESS_FIELD + " text, " +
        IMAGE_FIELD + " binary )";
    db.execSQL(SQL);
}
```

資料庫增加一筆資料 CODE:

```
long add(String name, String sex, String address, byte[] image) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(NAME_FIELD, name);
    values.put(SEXUAL_FIELD, sex);
    values.put(ADDRESS_FIELD, address);
    values.put(IMAGE_FIELD, image);
    long result = db.insert(_TableName, nullColumnHack: null, values);

    return result;
}
```

資料庫修改某筆資料 CODE:

```
long update(String name, String sexual, String address, byte[] image, String whereClause) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    if (name != null) values.put(NAME_FIELD, name);
    if (sexual != null) values.put(SEXUAL_FIELD, sexual);
    if (address != null) values.put(ADDRESS_FIELD, address);
    if (image != null) {
        values.put(IMAGE_FIELD, image);
        System.out.println("AFTER" + values);
    }

    long result = db.update(_TableName, values, whereClause, whereArgs: null);
    db.close();
    return result;
}
```

資料庫刪除某筆資料 CODE:

```
int delete(String _id) {
    SQLiteDatabase db = this.getWritableDatabase();
    int result = db.delete(_TableName, whereClause: ID_FIELD + " =" + _id, whereArgs: null);
    db.close();
    return result;
}
```

資料庫查詢所有資料 CODE:

```
Cursor selectAll() {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor c = null;
    c = db.rawQuery( sql: "SELECT * FROM " + _TableName, selectionArgs: null);
    return c;
}
```

資料庫查詢某筆資料 CODE:

```
Cursor selectData(String id, String name, String sexual, String address) {
    SQLiteDatabase db = this.getReadableDatabase();
    Cursor c = null;

    if (!id.equals("")){
        c = db.rawQuery( sql: "SELECT * FROM " + _TableName + " WHERE " + "_id=" + id + "'", selectionArgs: null);
    }
    if (!name.equals("")) {
        c = db.rawQuery( sql: "SELECT * FROM " + _TableName + " WHERE " + "name=" + name + "'", selectionArgs: null);
    } else if (!sexual.equals("")) {
        c = db.rawQuery( sql: "SELECT * FROM " + _TableName + " WHERE " + "sex = " + sexual + "'", selectionArgs: null);
    } else if (!address.equals("")) {
        c = db.rawQuery( sql: "SELECT * FROM " + _TableName + " WHERE " + "address = " + address + "'", selectionArgs: null);
    }
    return c;
}
```

選取圖片 CODE:

```
image.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        Intent gallery = new Intent(Intent.ACTION_PICK, MediaStore.Images.Media.INTERNAL_CONTENT_URI);
        startActivityForResult(gallery, PICK_IMAGE);
    }
});
```

選完圖片回傳圖片 CODE:

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if(requestCode == PICK_IMAGE && resultCode == RESULT_OK){
        Uri imageUri = data.getData();
        image.setImageURI(imageUri);
    }
}
```

因為我的資料庫是可以上傳圖片的，但要轉換型態 CODE:

```
Bitmap bitmap = ((BitmapDrawable)image.getDrawable()).getBitmap();
ByteArrayOutputStream stream = new ByteArrayOutputStream();
bitmap.compress(Bitmap.CompressFormat.PNG, quality: 100, stream);
byte[] byteArray = stream.toByteArray();
```

在每次回覆主畫面時，更新資料庫 ListView:

```
protected void onResume() {
    Cursor c = access.selectAll();
    if (adapter == null) {
        adapter = new ImageCursorAdapter(context: this, c, flags: 0);
        list.setAdapter(adapter);
    } else {
        adapter.changeCursor(c);
    }
    super.onResume();
}
```

那這裡我使用自定義的 ImageCursorAdapter 繼承自 CursorAdapter CODE:

```
public void bindView(View view, Context context, Cursor cursor) {
    TextView name = (TextView) view.findViewById(R.id.name);
    name.setText(cursor.getString(cursor.getColumnIndex(DBOpenHelper.NAME_FIELD)));
    TextView sexual = (TextView) view.findViewById(R.id.sex);
    sexual.setText(cursor.getString(cursor.getColumnIndex(DBOpenHelper.SEXUAL_FIELD)));
    TextView address = (TextView) view.findViewById(R.id.address);
    address.setText(cursor.getString(cursor.getColumnIndex(DBOpenHelper.ADDRESS_FIELD)));
    byte[] imageByte = cursor.getBlob(cursor.getColumnIndex(DBOpenHelper.IMAGE_FIELD));
    ImageView image = (ImageView) view.findViewById(R.id.image);
    if (imageByte != null) {
        image.setImageBitmap(BitmapFactory.decodeByteArray(imageByte, offset: 0, imageByte.length));
    }
}
```

會使用 CursorAdapter 而不是 SimpleCursorAdpater 是因為要顯示圖片，用 SimpleCursorAdpater 時，每次只要跳轉回來，在 onResume()裡的 adapter.changeCursor(c)都會報錯，我不太了解原因，最後就換成 CursorAdapter。

修改某個資料 CODE:

要先從資料庫裡查詢資料

```
access=new DBOpenHelper(context: this, name: "friend", factory: null, version: 1);
Bundle bundle = getIntent().getExtras();
id = bundle.getString(key: "id", defaultValue: "0");
Cursor c = access.selectData(id, name: "", sexual: "", address: "");
c.moveToFirst();
```

修改完再用資料庫裡的修改資料 CODE:

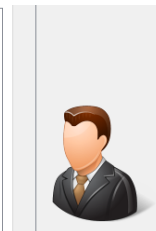
```
access.update(
    name: editName.getText() + "",
    sexual: editSex.getText() + "",
    address: editAddress.getText() + "",
    byteArray,
    whereClause: DBOpenHelper.ID_FIELD + " = " + id
);
finish();
```

刪除也是用資料庫裡的刪除資料 CODE:

```
access.delete(id);  
finish();
```

那我另外把 DB 直接從手機上拉出來用 DB Browser 看
確實都有上傳到資料庫裡

	_id	NAME	SEX	ADDRESS	IMAGE
	過濾	過濾	過濾	過濾	過濾
1	6	葉人豪	報導	三重	BLOB
2	12	Ren	報導	三重	BLOB



後面有 Layout

起始畫面

5:24

Hw6

輸入姓名

輸入性別

輸入地址

加入

查詢

列表

輸入資料

5:25

Hw6

葉人豪

男

三重

加入

查詢

列表

上傳圖片

5:25

← 選取相片

裝置資料夾

Download

4 個項目

選取圖片

5:25

× 選取相片

今天

昨天

選取圖片

加入資料庫

5:25

Hw6

葉人豪

男

三重

加入

查詢

列表

葉人豪

男

三重

5:28

Hw6

輸入姓名


輸入性別

輸入地址

加入

查詢


列表



葉人豪

男


三重



Hao

男女


蘆洲



Ren

女

新莊



Ren

男

三重

5:29

Hw6

輸入姓名


男

輸入地址

加入

查詢


列表



葉人豪

男

三重



Ren

男

三重

5:29

Hw6

輸入姓名


男

輸入地址

加入

查詢


列表



葉人豪

男


三重



Hao

男女


蘆洲



Ren

女

新莊



Ren

男

三重

刪除某筆資料

修改某筆資料

性別改為女

5:29

Hw6

輸入姓名


男

輸入地址

加入

查詢


列表



葉人豪

男


三重



Hao

男女

蘆洲



Ren

男

三重

5:35

姓名

Hao

性別

男女

地址

蘆洲

修改

刪除

5:35

姓名

Hao

性別

女

地址

蘆洲

修改

刪除

資料庫更改正確

5:35

Hw6

輸入姓名


男

輸入地址

加入

查詢


列表



葉人豪

男


三重



Hao

女

蘆洲



Ren

男

三重