# COMP 90018 Mobile Computing Systems Programming

## Tutorial on Android Development

Chu Luo, Eman Bin Khunayn

{chu.luo, eman.bin}@unimelb.edu.au

# Welcome

**Outcomes of this tutorial:**

1. **To store and read data from Content Provider and SQLite.**

# Question

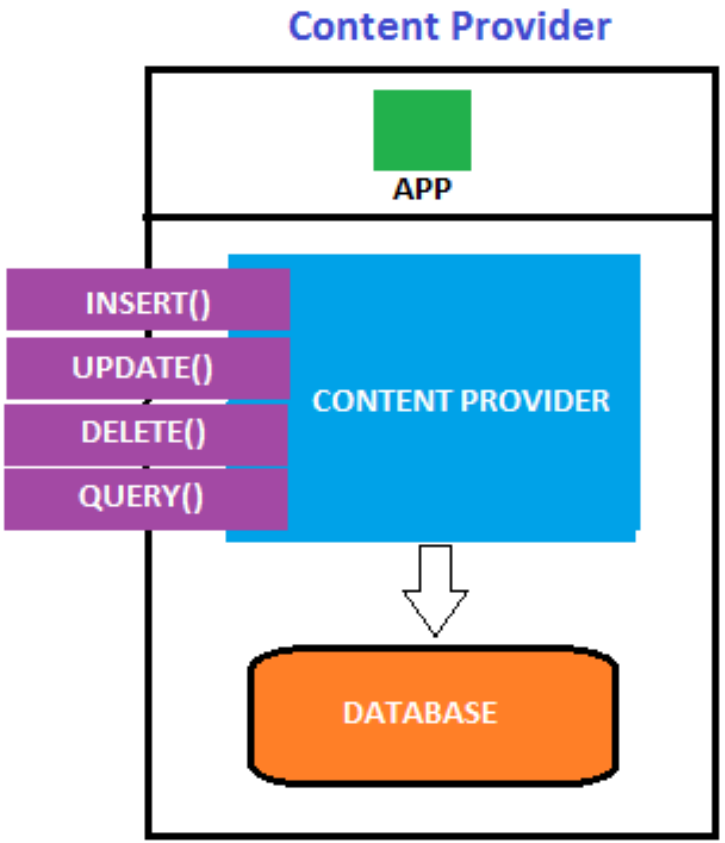
**What are the four components in Android Apps?**

# Four Components in Android Apps

1. Activity: foreground program
2. Service: background task
3. Broadcast Receiver: respond to events
4. Content Provider: read/write data on phone storage (or anywhere)

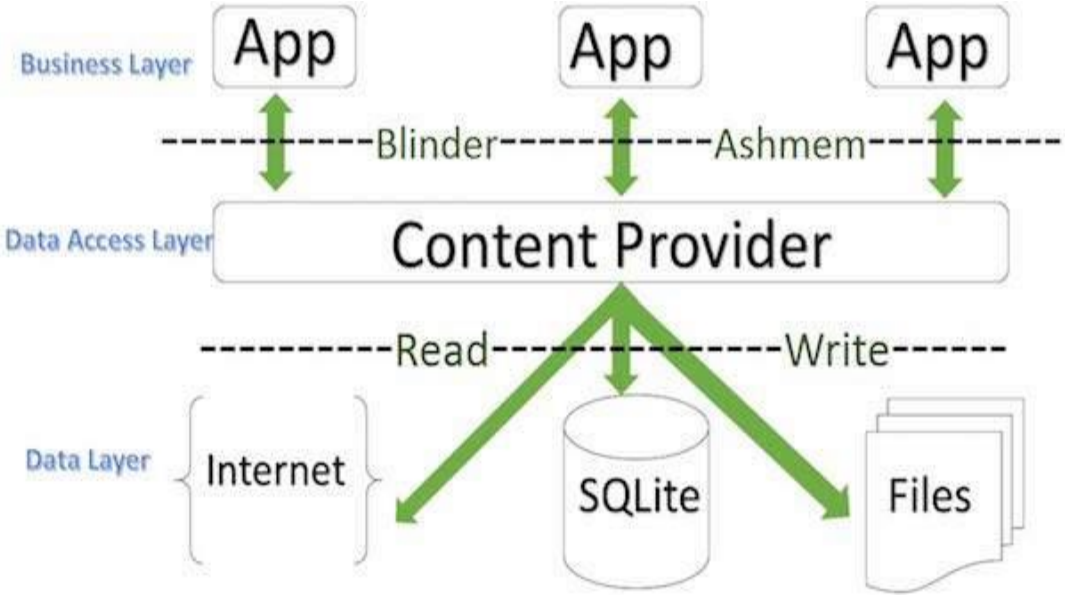
# Content Provider

- Mostly, for App to access Database



# Content Provider Component

- Supplies data from one application to others on request.
- Can use different ways to store its data and the data can be stored in a database, in files, or even over a network.



# Content Provider

- Behaves like a database where you can query it, edit its content, using (insert(), update(), delete(), and query()) methods.
- A content provider is implemented as a subclass of **ContentProvider** class.

```
public class My Application extends  ContentProvider {
}
```

# Question

**How do you identify a Content Provider?**

# URIs: For Identification

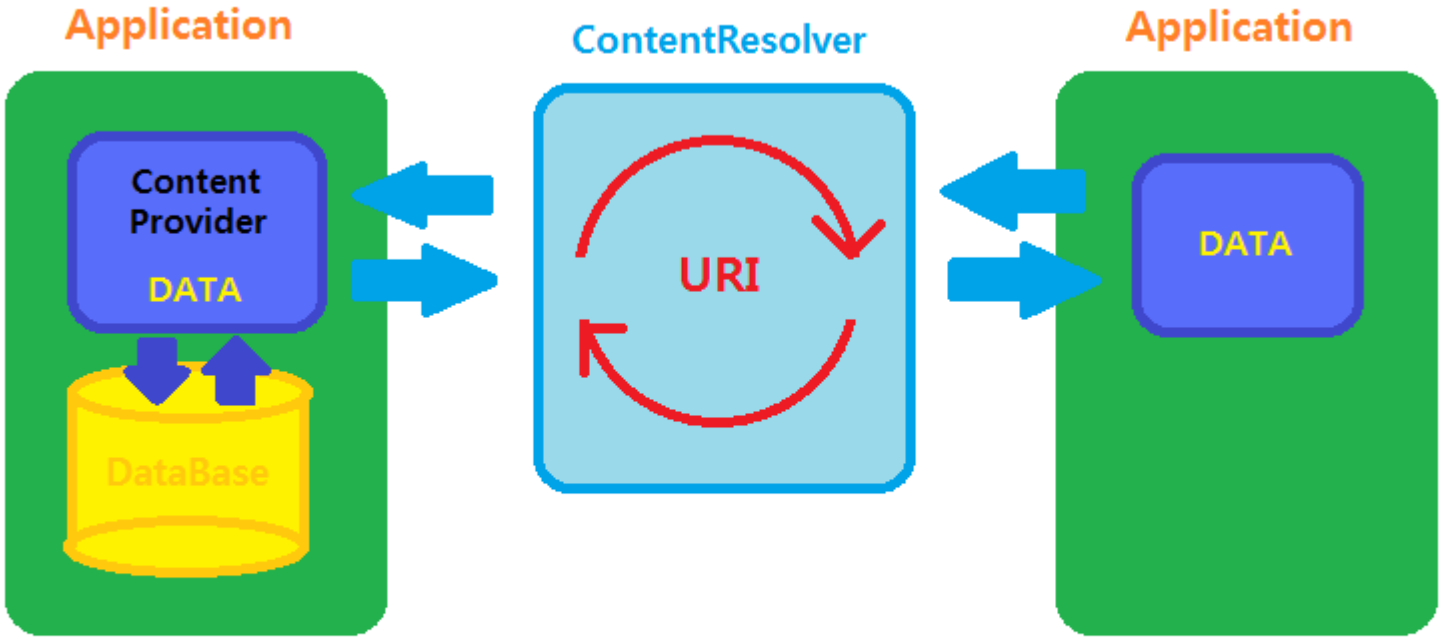



그림 출처 : arabiannight.tistory.com

# Content URIs

To query a content provider, you specify the query string in the form of a URI which has following format

content://<authority>/<data_type>/<id>

This specifies the name of the content provider, for example *contacts*,

type of data that this particular provider provides

specific record requested

# Create Content Provider

1. Create a Content Provider class that extends the *ContentProvider* baseclass.

2. Define your content provider URI address, which will be used to access the content.

3. Create your own database (*onCreate*()) to keep the content (e.g., SQLite, use *SQLite Open Helper* method to create/open the provider's database.).

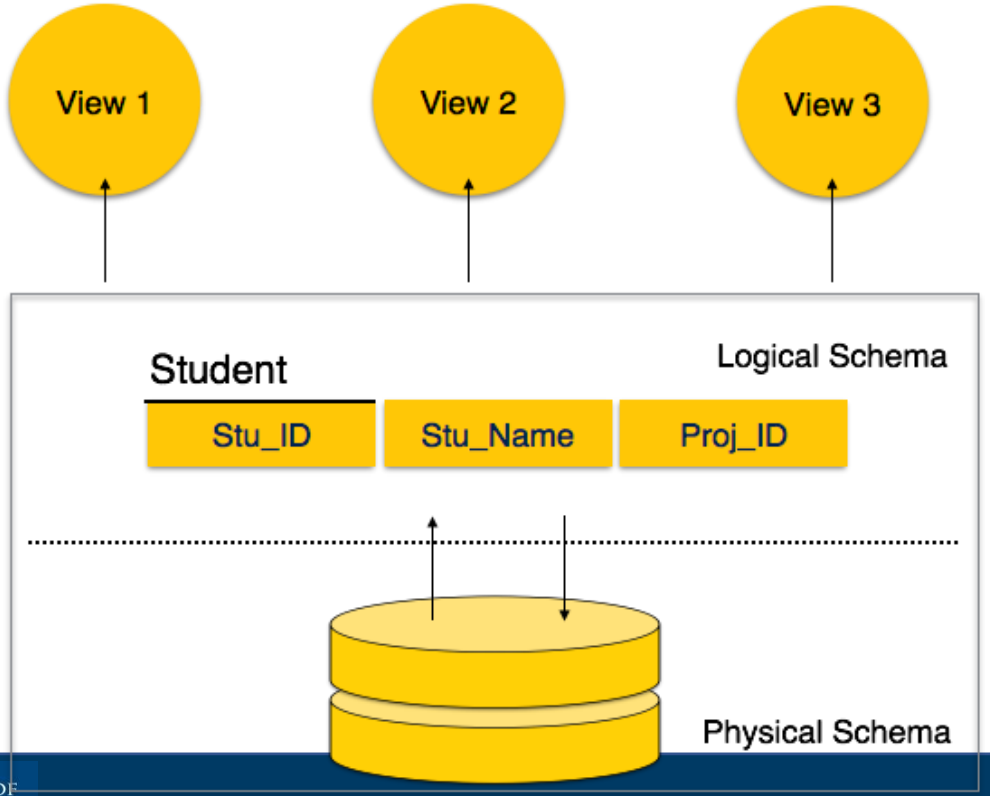
4. Implement Content Provider queries to perform database operations.

5. Register your Content Provider in your ***AndroidManifest.xml*** using <provider> tag.

# SQLite Database

- Open source SQL database via text file on a device.
- Built in SQLite database implementation.
- SQLite supports all the relational database features.
- No need to establish any connections, (like JDBC,ODBC )

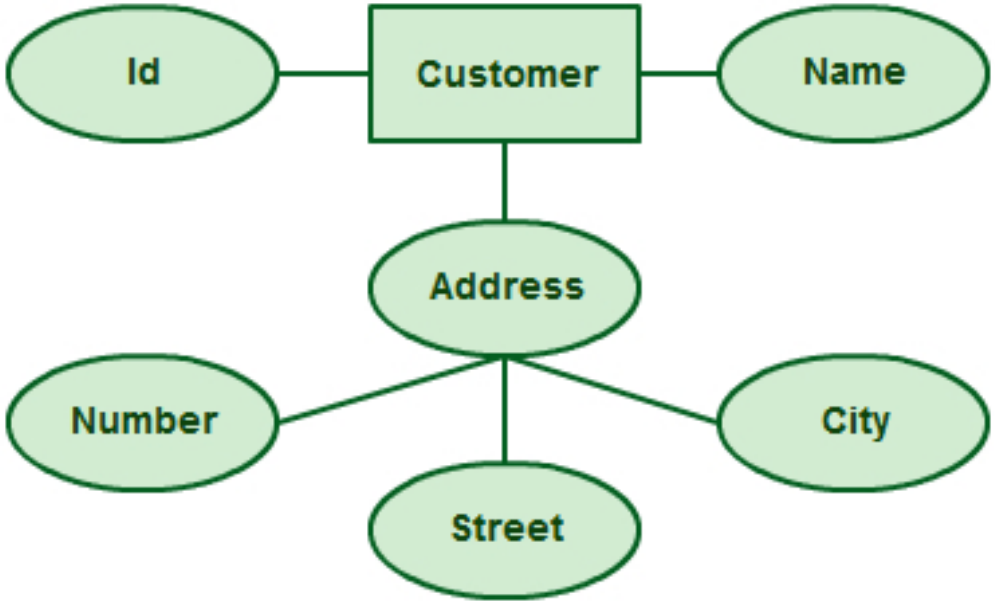
# Design a Schema First

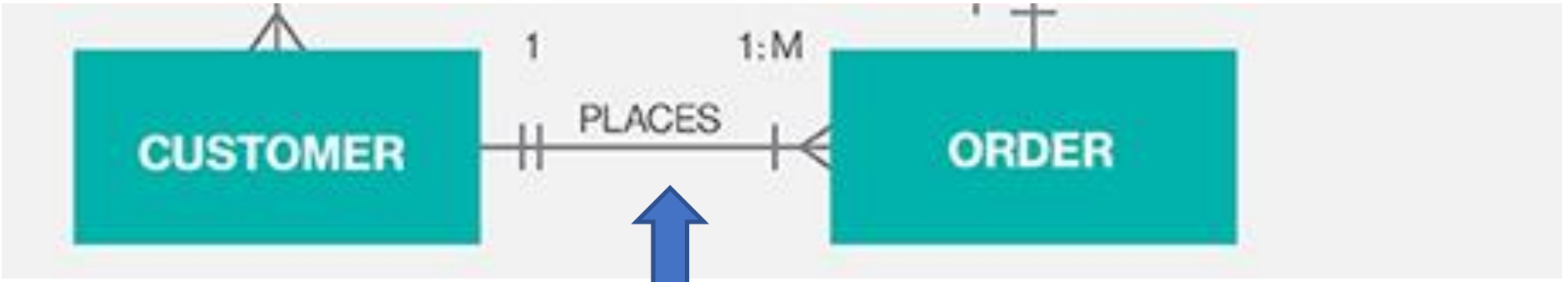


# Question

## How do you design a DB Schema?

# E.g., Using ER Diagram






Create table

Create table

Also create a table for relation

# To create DB:

```
SQLiteDatabase mydatabase = openOrCreateDatabase("your
database name", MODE_PRIVATE,null);
```

# To create table & Insert:

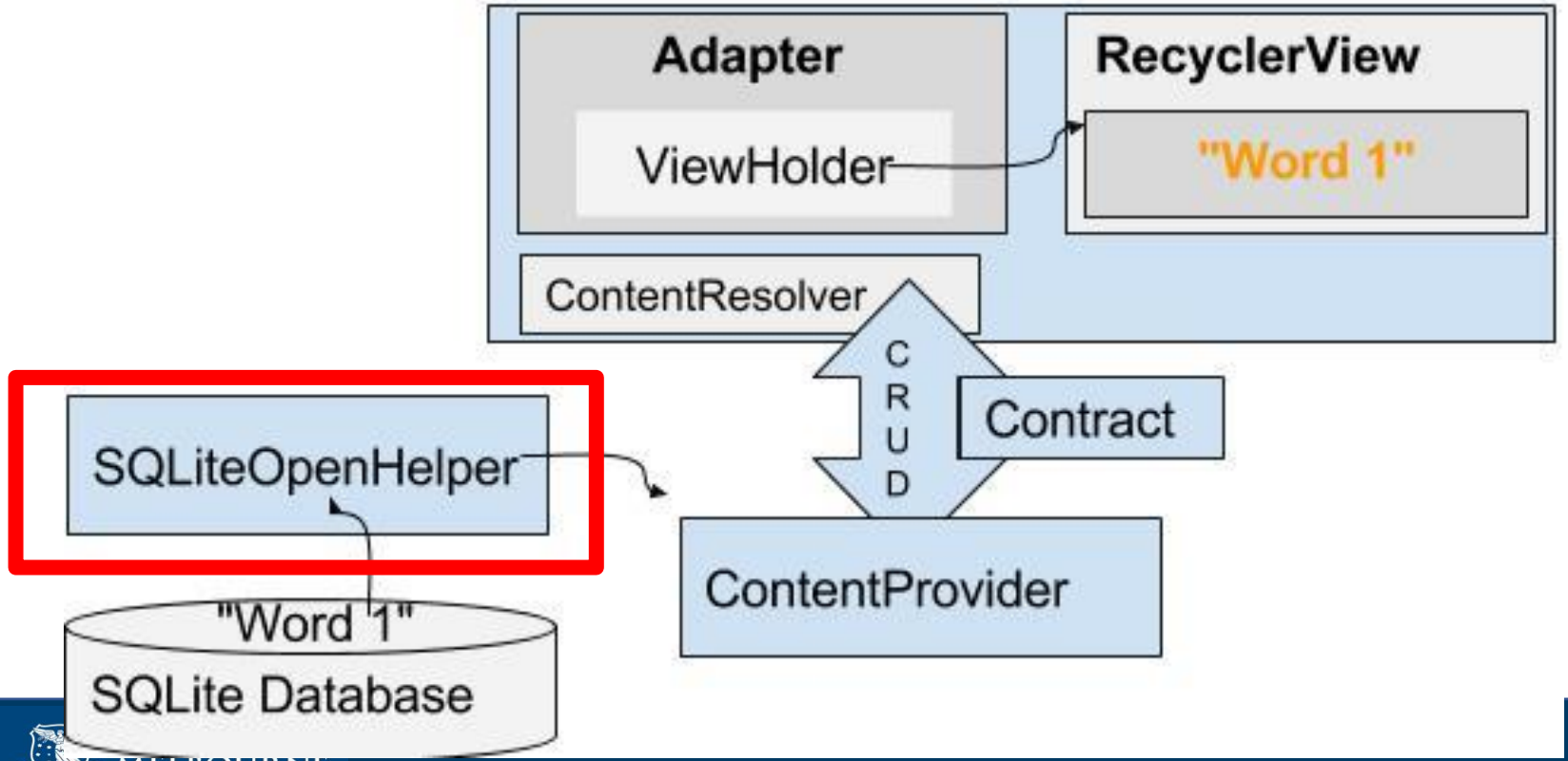
```
mydatabase.execSQL("CREATE TABLE IF NOT EXISTS
table_name(Username VARCHAR,Password VARCHAR);");

mydatabase.execSQL("INSERT INTO table_name
VALUES('admin','admin');");
```

# To retrieve:

```
Cursor resultSet = mydatbase.rawQuery("Select * from
table_name",null);
resultSet.moveToFirst();
String username = resultSet.getString(0);
String password = resultSet.getString(1);
```

Other functions like: getCount(), getColumnCount(), getColumnIndex(String columnName), getColumnName(int columnIndex), getPosition(), ..

# DatabaseHelper: Connect CP and DB



# Database - Helper class

To automatically manages the creation/update of the database.

```
public class DBHelper extends SQLiteOpenHelper {
   public DBHelper(){
            super(context,DATABASE_NAME,null,1);
                }
   public void onCreate(SQLiteDatabase db) {}
   public void onUpgrade(SQLiteDatabase database, int oldVersion,
int newVersion) {}
}
```

# Querying Data in Activity or Service:

# ContentResolver.query()

https://developer.android.com/guide/topics/providers/content-provider-basics.html

# Inserting Data in Activity or Service:

# ContentValues

# ContentResolver

https://developer.android.com/guide/topics/providers/content-provider-basics.html

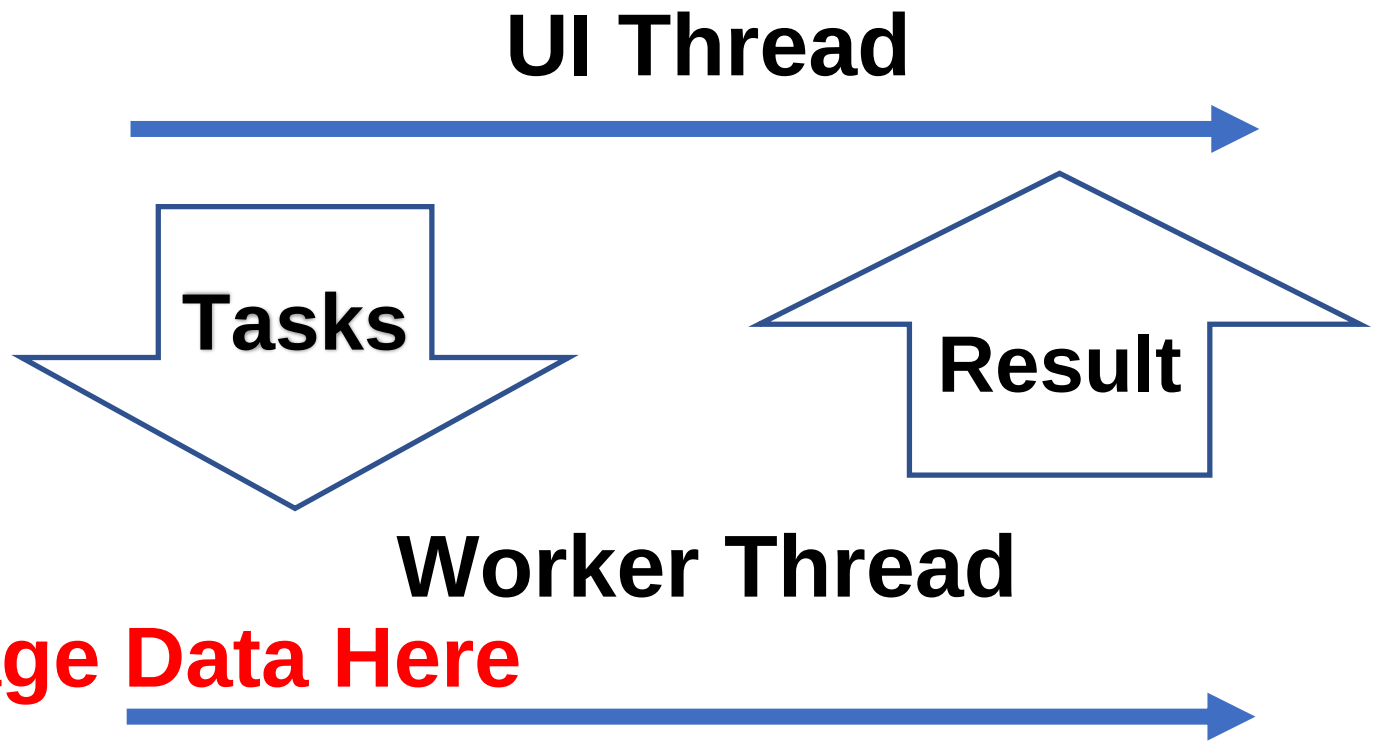
# Updating and Deleting :

# ContentResolver.update()/ delete()

https://developer.android.com/guide/topics/providers/content-provider-basics.html

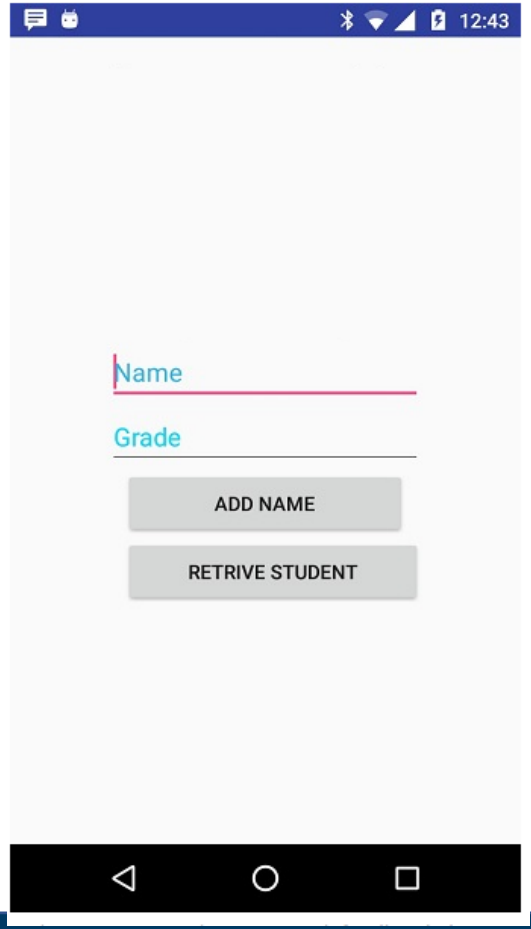
# Important: Do Not Manage Data on UI Thread

# A better design:



# Exercise:

Creates a basic Students Provider application that allows insertion, deletion and modification of student info using SQLite.



# Code Demo

# An easy implementation

# Code Example: A data collection middleware - AWARE

# Very good DBHelper and CP code

https://github.com/denzilferreira/aware-client/tree/master/aware-core/src/main

https://github.com/denzilferreira/aware-client/blob/master/aware-core/src/main/java/com/aware/providers/Accelerometer_Provider.java

https://github.com/denzilferreira/aware-client/blob/master/aware-core/src/main/java/com/aware/utils/DatabaseHelper.java

# More learning directions:

# 1. To use Azure (e.g., SQL database)

# See you next week

# COMP 90018

## Tutorial on Android Development

Chu Luo, Eman Bin Khunayn

{chu.luo, eman.bin}@unimelb.edu.au