MOBILE PROGRAMMING
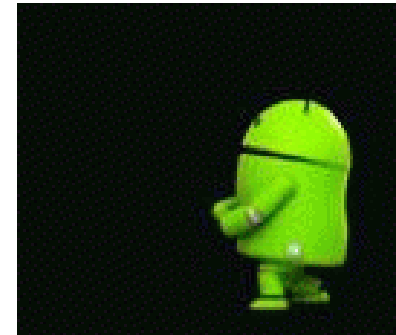
Chapter 5

ANDROID PROGRAMMING (ADVANCED)
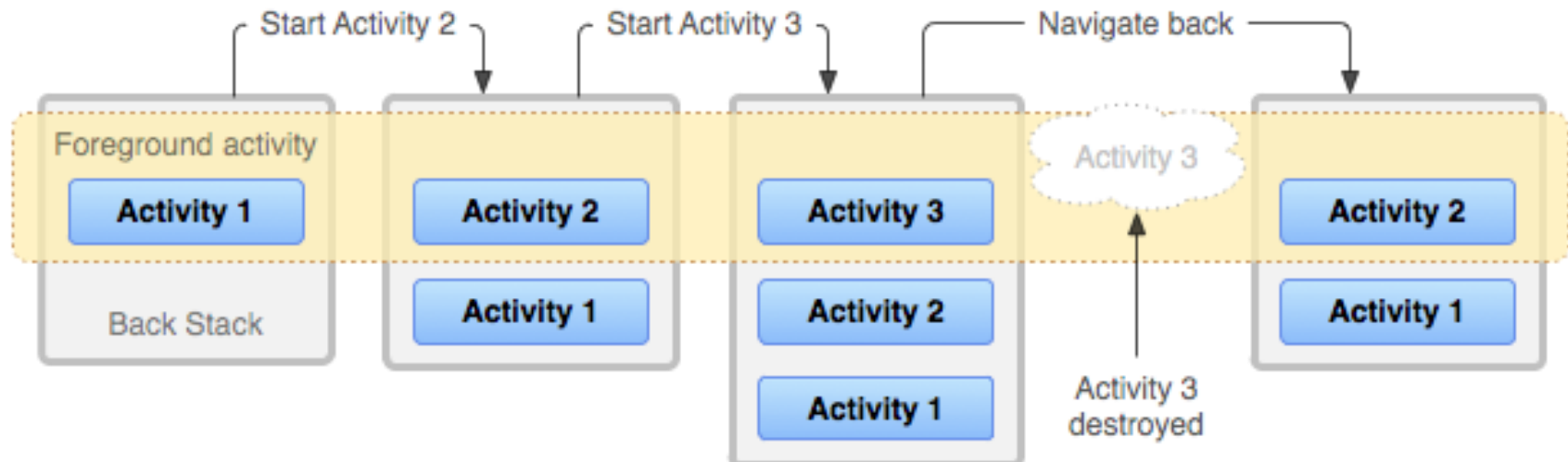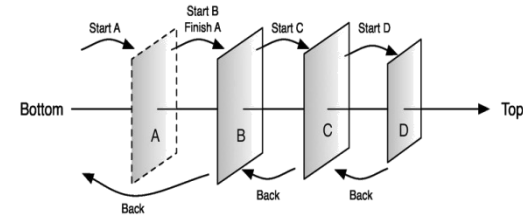
Contents

- **Details of activities, dialogs**
- **Adapter for list/grid view**
- **SharedReferences**
- **SQLite**
- **Network/Internet (Http)**
- **Maps,**
- **Sms & Telephony**
- **Multimedia,**
- **Notification,…**

5.1- DETAILS OF ACTIVITIES

- **Details of activity**
  - we can start an activity and back to
  - running activities are store in a stack
  - we can finish current activity & start another

5.1- DETAILS OF ACTIVITIES

- **Details of activity**

  – Step1) Create an Intent object

  application context

  class name of activity

  Intent Δ=new Intent(this, *TheClassName*.class);

  – Step2) Create Bundle (if needed $\rightarrow$ send data)

  Use "putExtra" instead of Bundle $\rightarrow$ getExtra

  Bundle ♦=new Bundle();
  ♦.put...(key, value);

  Declare the activity in Manifest.xml for running

  – Step3) Start activity with the Intent
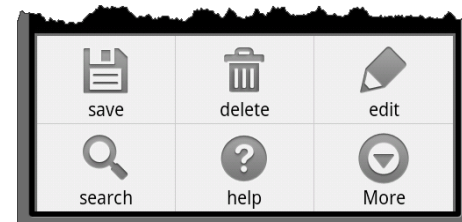
  startActivity(Δ, ♦);

  putInt, putString,... for sending data to new Activity

  call from an activity

5.1- DETAILS OF ACTIVITIES

- **Options menu**
  - Step1) Design a menu of items

```xml
Create_OptionsMenu.java | menu_options.xml | APK_VD Manifest
1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:android="http://schemas.android.com/apk/res/android" >
3      <item android:id="@+id/item1" android:title="Clear"></item>
4      <item android:id="@+id/item3" android:title="Exit"></item>
5      <group android:id="@+id/group1">
6          <item android:id="@+id/item4" android:title="Setting"></item>
7          <item android:id="@+id/item5" android:title="About"></item>
8      </group>
9  </menu>
10
```

  - Step2) Set the menu to menu-options

```java
26  public boolean onCreateOptionsMenu(Menu  m){
27      getMenuInflater().inflate(R.menu.menu_options, m);
28      return  true;
29  }
```

implement the "onCreateOptionsMenu" method

5.1- DETAILS OF ACTIVITIES

- ## **Options menu**
  - Step2') Dynamically creating options menu

```
20⊖    public boolean onCreateOptionsMenu(Menu  m){
21         m.add("Show me").setIcon(R.drawable.ic_launcher);
22         m.add("Clear").setIcon(android.R.drawable.btn_minus);
23         return  super.onCreateOptionsMenu(m);
24     }
```

Add items to Menu & set attributes

  - Step3) Implement item selections

boolean   onOptionsItemSelected(MenuItem   Δ){
   … DO SOMETHING HERE …
}

Δ.getItemId(),
Δ.getTitle(),…

```
27⊖    public boolean onOptionsItemSelected(MenuItem  item){
28         Toast.makeText(this, "Item="+item.getItemId(), Toast.LENGTH_LONG).show();
29         return  super.onOptionsItemSelected(item);
30     }
```

5.1- DETAILS OF ACTIVITIES

- **Dialogs**
  - Create dialog
    - Step1) Create dialog builder

      AlertDialog.Builder △=new AlertDialog.Builder(♦);

    - Step2) Change various charateristics

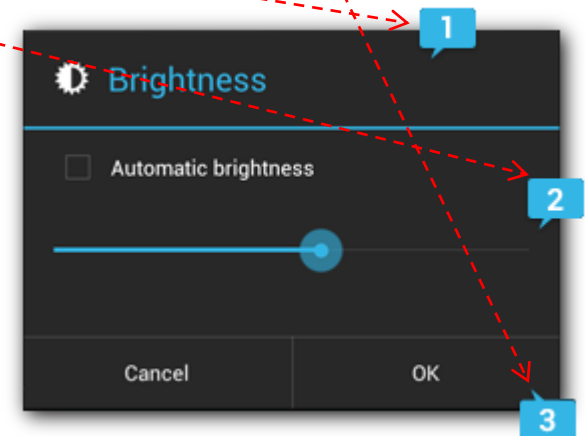   **△.setMessage(...).setTitle(...).setIcon(...).set*Xyz...*;**

    - Step3) Create dialog

      AlertDialog ♥=△.create();

    - Step4) Show dialog

      ♥.show();

setSingleChoiceItems(),
setNegativeButton(),
setPositiveButton(),...

5.1- DETAILS OF ACTIVITIES
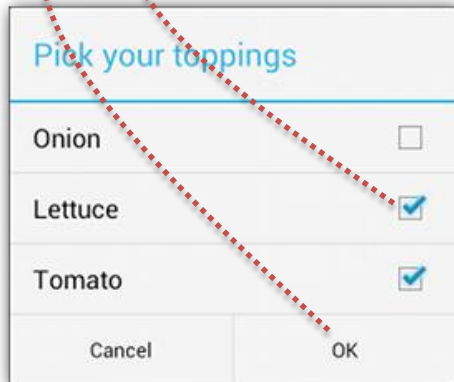
- **Dialogs**
  - Handling events on dialogs

    OnMultiChoiceClickListener

    - Step1) Create objects for handling events

      ♣ = new DialogInterface.**OnClickListener**(){
        void **onClick**(DialogInterface *d*, **int** *w*){ ... }
      };

    - Step2) Register to dialog with "set" methods

      Δ.**setItems**(String[] , ♣);
      Δ.**setMultiChoiceItems**(String[] , boolean[] , ♣);
      Δ.**setSingleChoiceItems**(String[] , int , ♣);
      Δ.**setNegativeButton**(String, ♣);
      Δ.**setPositiveButton**(String, ♣); ...

Pick your toppings

Onion ☐

Lettuce ☑

Tomato ☑

Cancel   OK

5.1- DETAILS OF ACTIVITIES

- **Dialogs**
  - Create custom dialog
    - Step1) Design a layout for dialog (♥.xml)
    - Step2) Create dialog object

      Δ = new **Dialog**(*theActivity*)**;**

    - Step3) Set the layout to dialog

      Δ.**setContentView**(*R.layout.♥*);

    - Step4) Handling some events on dialog

      findViewById $\rightarrow$ setOnClickListener,…

    - Step5) Show dialog

      Δ.**show**();

5.1- DETAILS OF ACTIVITIES

- **Example 5.1**
  - Create video welcome activity, it has options menu (Login $\rightarrow$ dialog, About $\rightarrow$ other activity).
  - Design welcome, login, about layout

```
2 <RelativeLayout xmlns:android="http://schem
3     android:layout_width="match_parent"
4     android:layout_height="match_parent" >
5     <VideoView
6         android:layout_height="fill_parent"
7         android:layout_width="fill_parent"
8         android:id="@+id/ex51_vv1"/>
9 </RelativeLayout>
```

```
2 <RelativeLayout xmlns:android="http://schem
3     android:layout_width="match_parent"
4     android:layout_height="match_parent" >
5     <ImageView
6         android:layout_height="fill_parent"
7         android:layout_width="fill_parent"
8         android:id="@+id/ex51_iv1"
9         android:src="@drawable/abc"/>
10 </RelativeLayout>
```
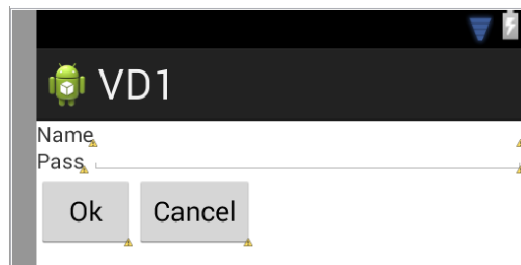
5.1- DETAILS OF ACTIVITIES

- ## Example 5.1

```
20  public class ex51_main extends Activity {
21      public void onCreate(Bundle ts){
22          super.onCreate(ts);
23          setContentView(R.layout.ex51_wellcome);
24          VideoView  vv=(VideoView)findViewById(R.id.ex51_vv1);
25          vv.setVideoURI(Uri.parse("android.resource://"+
26                  getPackageName()+"/"+R.raw.sample));
27          vv.start();
28      }
29      public boolean onCreateOptionsMenu(Menu  m){
30          m.add("Login").setIcon(android.R.drawable.btn_plus);
31          m.add("About").setIcon(android.R.drawable.btn_minus);
32          return  super.onCreateOptionsMenu(m);
33      }
```

5.1- DETAILS OF ACTIVITIES

- **Example 5.1**

```java
34  public boolean  onOptionsItemSelected(MenuItem  it){
35      if(it.getTitle().equals("Login")){
36          final  Dialog  dlg=new  Dialog(this);
37          dlg.setContentView(R.layout.ex51_login);
38          ((Button)dlg.findViewById(R.id.ex51_bt_ok)).setOnClickListener(
39              new android.view.View.OnClickListener() {
40                  public void onClick(View v) {
41                      Toast.makeText(ex51_main.this,
42                              "Your name & password are NOT correct!",
43                              Toast.LENGTH_SHORT).show();
44                  }
45              });
46          ((Button)dlg.findViewById(R.id.ex51_bt_cancel)).setOnClickListener(
47              new android.view.View.OnClickListener() {
48                  public void onClick(View v) {  dlg.dismiss();  }
49              });
50          dlg.show();
51      }else{
52          Intent  intent=new Intent(this, ex51_2nd_activity.class);
53          startActivity(intent);
54      }
55      return  true;
56  }
57 }
```

5.2- ADAPTER FOR LIST

- **Adapter → bridge {AdapterViews and the underlying data}**

ListView

GridView
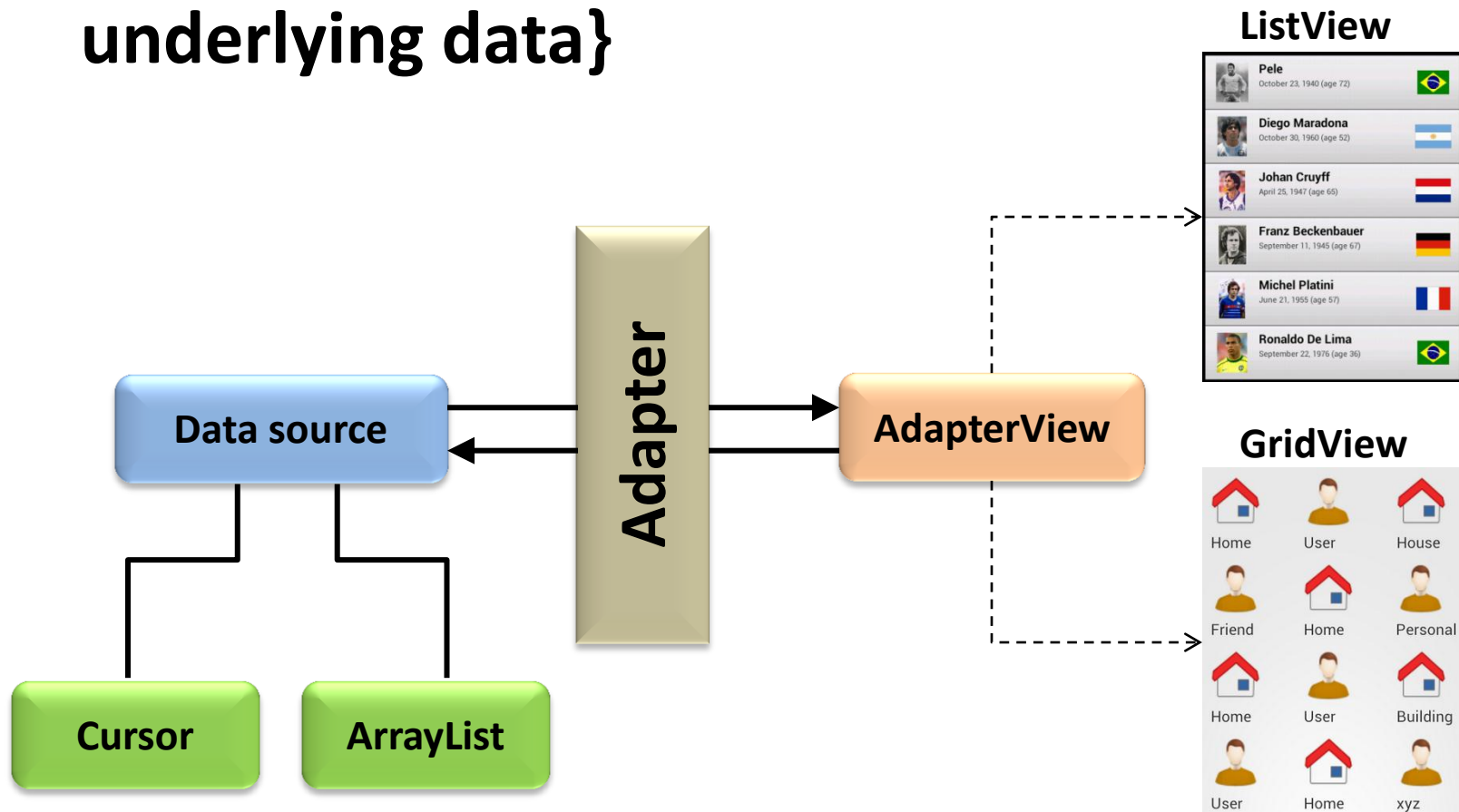
5.2- ADAPTER FOR LIST

- **Simple ListView**
  - Step1) Prepare data for list

```
String[] lst=new String[]{"One","Two","Three","Four","..."};
```

  - Step2) Create adapter (including data)

```
ArrayAdapter<String> ad=new ArrayAdapter<String>(this,
        android.R.layout.simple_list_item_1,lst);
```

  - Step4) Set adapter to list object

```
lv.setAdapter(ad);
```

type of list

list object

Example 1

One

Two

Three

Four

...

5.2- ADAPTER FOR LIST

- ## **Custom ListView**
  - – Step1) Design a list layout (♥.xml)
    - Use "**<ListView … /ListView>**" (id=⊗) tag in a layout
  - – Step2) Design a row layout for list (♣.xml)
    - Use any tag for rows such as **Image**, **TextView**,… (id=⊕)
  - – Step3) Extend ArrayAdapter (2 methods) → ◊
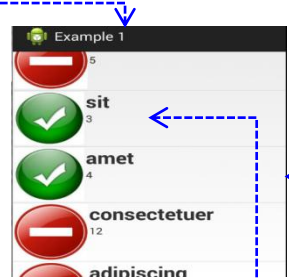   - constructor: call "super" for constructing an Adapter

      super(Context, layout_ID, txt_ID, txtList_array);

    - getView: generate a view for each item(row) of list

      getView(int position, View row, ViewGroup vg){…}

  - – Step4) Set adapter object to a list

      ⊗.**setAdapter**(◊);

findViewById(⊗)

set content to

an object

5.2- ADAPTER FOR LIST

- **Example 5.2: custom list view**
  - 1) Layout (♥.xml) has a ListView

```
2⊖ <LinearLayout xmlns:android="http://schemas
3      android:layout_width="fill_parent"
4      android:layout_height="fill_parent"
5      android:orientation="vertical" >
6      <ListView
7          android:layout_width="fill_parent"
8          android:layout_height="fill_parent"
9          android:id="@+id/clv_lv1">
10     </ListView>
11 </LinearLayout>
```

  - 2) Layout of row for list (♣.xml)

```
2  <LinearLayout xmlns:android="http://schemas.and
3      android:layout_width="fill_parent" android:
4      android:orientation="horizontal" >
5      <ImageView
6          android:id="@+id/clv_icon"
7          android:layout_width="wrap_content" andro
8          android:layout_gravity="center_vertical"
9          android:padding="2dip"
10         android:src="@drawable/ic_launcher"
11         android:contentDescription="Biểu tượng"/>
```

```
12     <LinearLayout
13         android:layout_width="fill_parent" a
14         android:orientation="vertical">
15         <TextView
16             android:id="@+id/label"
17             android:layout_width="wrap_content
18             android:textSize="25sp"
19             android:textStyle="bold"/>
20         <TextView
21             android:id="@+id/size"/>
22     </LinearLayout>
23 </LinearLayout>
```

5.2- ADAPTER FOR LIST

- ## Example 5.2: custom list view
  - 3) Extend ArrayAdapter (constructor, getView)

```java
28  class MyAdapter extends ArrayAdapter<String> {
29      MyAdapter() {
30          super(Create_Custom_ListView.this,
31                  R.layout.custom_listview_row, R.id.clv_label, items);
32      }
33      public View getView(int position, View convertView, ViewGroup parent) {
34          View row=super.getView(position, convertView, parent);
35          ImageView icon=(ImageView)row.findViewById(R.id.clv_icon);
            if (items[position].length()>4) {
                icon.setImageResource(R.drawable.delete);
            }else{
                icon.setImageResource(R.drawable.ok);
            }
            TextView size=(TextView)row.findViewById(R.id.clv_size);
            size.setText(""+items[position].length());
43          return(row);
44      }
45  }
```

convertView = a View for an item at position

super . getView → generate View as a row

  - 4) Set adapter object to list

```java
ListView  lv=(ListView)findViewById(R.id.clv_lv1);
lv.setAdapter(new  MyAdapter());
```

5.2- ADAPTER FOR LIST

- ## Custom ListView

  - Handling events

    - Create "AdapterView.**OnItemClickListener**" object (♥)

      ```
      void   onItemClick(  AdapterView<?>  parent,
                           View   view,  int   position, long  id){
          … DO SOMETHING HERE …
      }
      ```

    - Register to ListView

      ```
      Δ.setOnItemClickListener( ♥ );
      ```

      the ListView object

5.2- ADAPTER FOR LIST

- ## **Custom ListView**
  - ## – Set **SELECTOR** for list (if need)
    - Step6.a) Design a "shape" for drawing color (Δ)

```
2  <shape xmlns:android="http://schemas.android.com/apk/res/android" >
3      <gradient android:endColor="#ffc579" android:startColor="#fb9d23" android:angle="90"></gradient>
4  </shape>
```

    - Step6.b) Design a "selector" for selecting item (♦)

```
2  <selector xmlns:android="http://schemas.android.com/apk/res/android">
3      <item android:state_pressed="true" android:drawable="@drawable/list_selector_pressed" />
4  </selector>
```

    - Step6.c) Set selector to listview

```
6      <ListView
7          android:layout_width="fill_parent"
8          android:layout_height="fill_parent"
9          android:listSelector="@drawable/list_selector"
10         android:id="@+id/clv_lv1">
11     </ListView>
```

5.2- ADAPTER FOR LIST

- **Custom GridView: how to?**
  - Do the same as ListView
    - Step1) Design a layout including GridView control (Δ)
    - Step2) Design an **item layout** in GridView (♦)
    - Step3) Extend ArrayAdapter / BaseAdapter
      - Constructor: initalize an object by calling super(…)

        super(*theActivity* **,** Δ **,** *array_data* **);**

      - getView: generate an item for displaying in the grid

        LayoutInflater ♥ = *theActivity*.**getLayoutInflater**();
        convertView = ♥.inflate(♦ , parent, false);

        ViewGroup

    - Step4) Set adapter object to GridView

      GridView object

      ♣.**setAdapter**(*adapter_object*);

5.2- ADAPTER FOR LIST

- ## **Custom GridView: example**

```xml
2  <LinearLayout xmlns:android="http://schemas.
3      android:layout_width="fill_parent"
4      android:layout_height="fill_parent"
5      android:orientation="vertical" >
6      <GridView
7          android:layout_width="fill_parent"
8          android:layout_height="fill_parent"
9          android:numColumns="auto_fit"
10         android:id="@+id/cgv_gv1">
11     </GridView>
12 </LinearLayout>
```

```xml
2  <LinearLayout xmlns:android="http://schemas.andr
3      android:layout_width="fill_parent" android:la
4      android:orientation="vertical" >
5      <ImageView
6          android:id="@+id/cgv_icon"
7          android:layout_width="wrap_content"
8          android:layout_height="wrap_content"
9          android:layout_gravity="center_horizontal"
10         android:padding="2dip"/>
11     <TextView
12         android:id="@+id/cgv_label"
13         android:layout_width="wrap_content"
14         android:layout_height="wrap_content"
15         android:layout_gravity="center_horizontal"
16         android:textStyle="bold"/>
17 </LinearLayout>
```


Example 1

dolor   sit

amet   consectetuer

5.2- ADAPTER FOR LIST

- ## **Custom GridView: example**

```
30  class MyAdapter extends ArrayAdapter<String> {
31      MyAdapter() {
32          super(Create_Custom_GridView.this, R.layout.custom_gridview, items);
33      }
34      public View getView(int position, View convertView, ViewGroup parent) {
35          if(convertView==null){
36              LayoutInflater inflater = Create_Custom_GridView.this.getLayoutInflater();
37              convertView = inflater.inflate(R.layout.custom_gridview_item, parent, false);
38          }
39          ImageView icon=(ImageView)convertView.findViewById(R.id.cgv_icon);
40          TextView  tv=(TextView)convertView.findViewById(R.id.cgv_label);
41          if (items[position].length()>4) {
42              icon.setImageResource(R.drawable.delete);
43          }else{
44              icon.setImageResource(R.drawable.ok);
45          }
46          tv.setText(items[position]);
47          return (convertView);
48      }
49  }
```

5.3- SharedReferences

- **Introduction**
  - About the storage in android
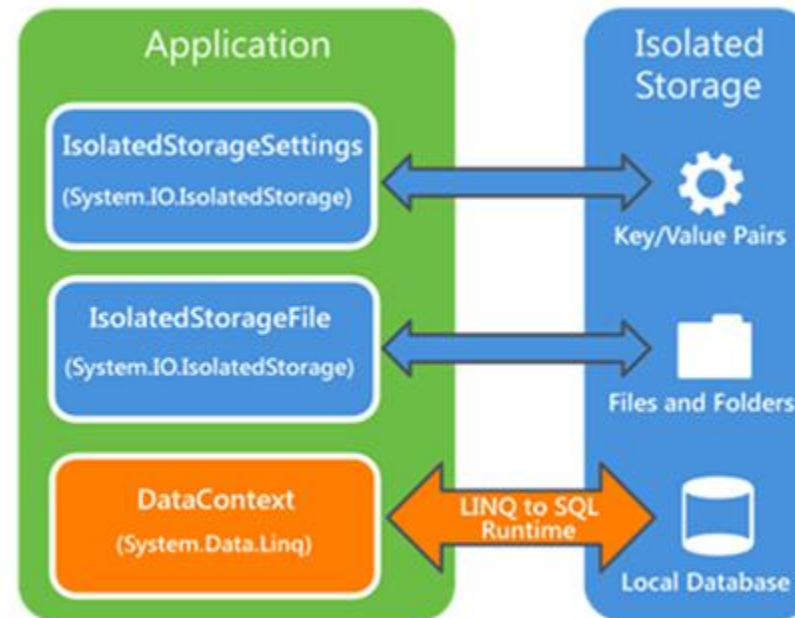
  - SharedReferences: **key/value** pairs for storing data

5.3- SharedReferences

- **How to?**
  - Step1) Get access object (♣) in Activity (◊)

    ♣ = ◊.**getSharedReferences**(*dataName*, *mode*);

    MODE_PRIVATE

  - Step2) Get edit object (♥)

    ♥ = ♣.**edit**();

    SharedPreferences.Editor

  - Step3) Edit data

    ♥.**putString**(*key* , *value*); … or putInt/Long/Float

  - Step4) Complete

    ♥.**commit**();

    for reading, we use alternately as:
       ◊.**getString**(*key*);
       ◊.**getInt**(*key*);
    …

5.3- SharedReferences

• Example 5.3: name/score pairs of users

```java
void        addUser(int k,String name,int  score){
    // Tao đối tương xử lý dữ liệu
    SharedPreferences mySP =
        getSharedPreferences("HS", MODE_PRIVATE);
    // Lấy đối tương chỉnh sửa dữ liệu.
    SharedPreferences.Editor editor=mySP.edit();
    // Thay đổi dữ liệu tên & mật khẩu.
    editor.putString("user"+k, name);
    editor.putInt("score"+k, score);
    // Hoàn tất việc thay đổi.
    editor.commit();
}
```

key/value pairs are **user$_k$ / name** & **score$_k$ / score**

```java
String       getUser(int k){
    // Tao đối tương xử lý dữ liệu.
    SharedPreferences mySP =
        getSharedPreferences("HS", MODE_PRIVATE);
    // Đọc dữ liệu tên & mật khẩu.
    String  u_name, kq=null; int  u_score;
    u_name = mySP.getString("user"+k,null);
    u_score = mySP.getInt("score"+k,0);
    if(u_name!=null) kq = u_name+"\n"+u_score;
    // Xử lý dữ liệu đọc được.
    return  kq;
}
```

k is index of user for distinct with each others

5.4- SQLite

- ## **SQLite introduction**
  - – Databases for Android
  - – Is relational model
  - – Use adapters for accessing
    - SQLiteOpenHelper
    - DatabaseHelper
    - SQLiteDatabase
    - Cursor
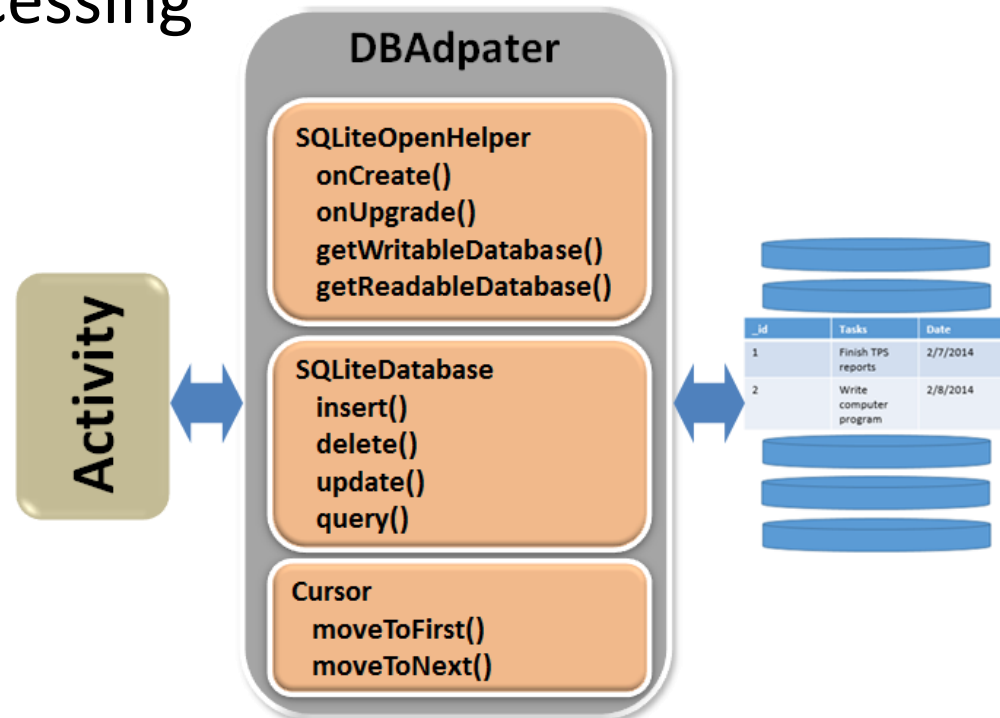
5.4- SQLite

- ## SQLite: simple way

  - Step1) Open/Create (if not exist) database

    Δ = **openOrCreateDatabase**(dbName, mode, null);

    Activity.**MODE_PRIVATE**

  - Step2) Execute any non_query SQL (insert, update, delete, create table…)

    Δ.**execSQL**(*nonQuerySQL_statement*);

  - Step3) Execute query SQL for retreive data

    Cursor   ♣ = Δ.**rawQuery**(*querySQL_statement*);

  - Step4) Get record data from Cursor

    ♣.**getString**(*col_idx*); … getInt/Long/Float

  - Step5) Close

    Δ.close();

    ♣.moveToFirst()
    ♣.moveToNext()

5.4- SQLite

- ## **SQLite: example in simple way**

  – Create database & add some records

```
SQLiteDatabase  myDb = openOrCreateDatabase("myDB.db", MODE_PRIVATE, null);
myDb.execSQL("create table sinhvien ( masv integer primary key autoincrement," +
            "hoten nvarchar(50), ngaysinh varchar(10), gioitinh boolean)");
myDb.execSQL("insert into sinhvien (hoten,ngaysinh,gioitinh) values "+
                                    "('Nguyễn Văn An','10/11/1995',1)");
myDb.execSQL("insert into sinhvien (hoten,ngaysinh,gioitinh) values "+
                                    "('Trần Thị Hoa','11/10/1959',0)");
myDb.close();
```

  – Open & read records

```
SQLiteDatabase  myDb = openOrCreateDatabase("myDB.db", MODE_PRIVATE, null);
Cursor  cs = myDb.rawQuery("select * from sinhvien", null);
cs.moveToFirst();  String  kq="";
do{
    kq = kq + "\n" + cs.getString(0) + "; " + cs.getString(1) + "; " +
                    cs.getString(2) + "; " + cs.getString(3);
}while(cs.moveToNext());
myDb.close();
tv.setText(kq);
```

5.4- SQLite

- **Create databases**

  – Extend "**SQLiteOpenHelper**" (override 3 methods)

    - constructors

      – for calling **super(Context,dbName,null,vrs)** to create DB
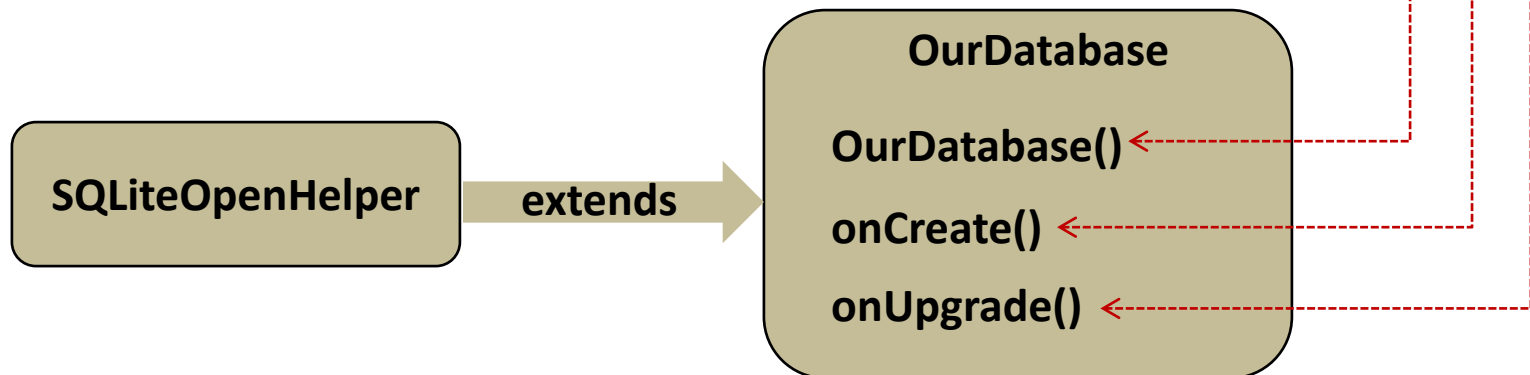
    - **onCreate**(SQLiteDatabase db)

```
// SQL statement to create book table
String CREATE_BOOK_TABLE = "CREATE TABLE books ( " +
        "id INTEGER PRIMARY KEY AUTOINCREMENT, " +
        "title TEXT, "+
        "author TEXT )";

// create books table
db.execSQL(CREATE_BOOK_TABLE);
```

      – for creating tables

    - **onUpgrade**(SQLiteDatabase db, int oVrs, int nVrs)

      – for modifying tables (delete → create new)

```
SQLiteOpenHelper  —extends→  OurDatabase

                              OurDatabase()

                              onCreate()

                              onUpgrade()
```

5.4- SQLite

- ## Access databases

SQLiteOpenHelper

SQLiteDatabase

insert/update/delete records

ContentValues

query records

query records

loop for
reading each values
of each record in turn

Cursor
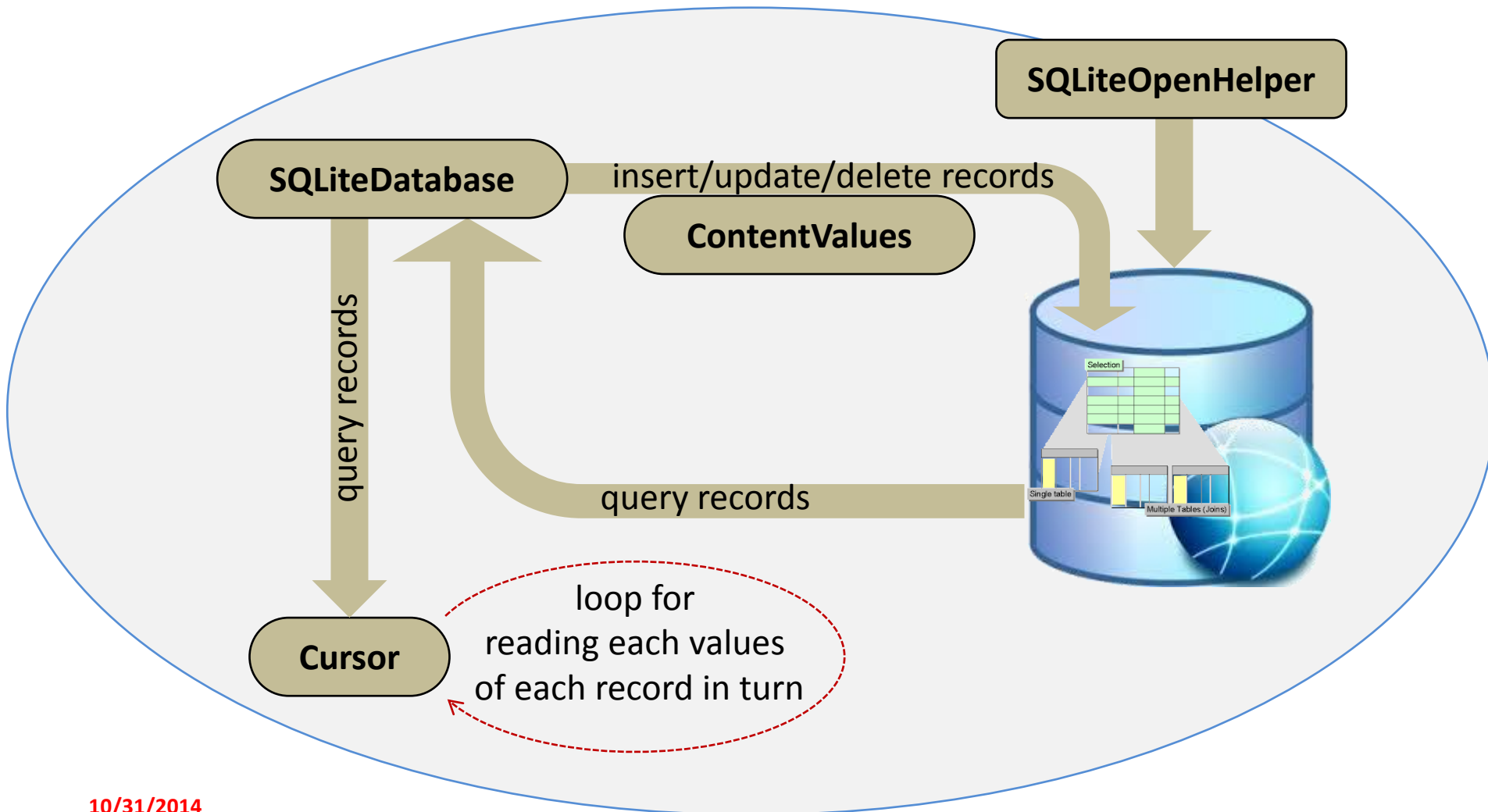
5.4- SQLite

- **Add records**
  - Step1) Get writable DB (Δ) from OurDatabase ($\blacklozenge$)

    > Δ = $\blacklozenge$.**getWritableDatabase**();

  - Step2) Create "**ContentValues**" of record ($\clubsuit$)

    > $\clubsuit$ = new **ContentValues**();
    > $\clubsuit$.**put**(*column1*, *value1*); ...

    ```
    // 2. create ContentValues to add key "colu
    ContentValues values = new ContentValues();
    values.put(KEY_TITLE, book.getTitle()); //
    values.put(KEY_AUTHOR, book.getAuthor()); //
    ```

  - Step3) Insert record

    > Δ.**insert**(*table_name*, **null**, $\clubsuit$);

    ```
    // 3. insert
    db.insert(TABLE_BOOKS, // table
            null, //nullColumnHack
            values); // key/value -> keys = col
    ```

  - Step4) Close

    > Δ.**close**();

5.4- SQLite

- **Delete records**
  - Step1) Get writable DB (Δ) from OurDatabase ($\blacklozenge$)

    $\Delta = \blacklozenge$.**getWritableDatabase**();

  - Step2) Delete record

    Δ.**delete**(*table_name* , *where* , *args*);

    - *where*: determine which records to be deleted ("column1=? AND column2=? ...")
    - *args*: array of values for replacing to ? in where
  - Step3) Close

    Δ.**close**();

```
// 2. delete
db.delete(TABLE_BOOKS, //table name
        KEY_ID+" = ?",  // selections
        new String[] { String.valueOf(book.getId()) });
```

5.4- SQLite

- **Modify records**
  - Step1) Get writable DB (Δ) from OurDatabase (♦)

    > Δ = ♦.**getWritableDatabase**();

  - Step2) Create "ContentValues" for updating

    > **ContentValues** ♣ = new **ContentValues**();
    > ♣.**put**(column1 , value1); …

  - Step3) Update new values to records

    > Δ.**update**(*table_name* , ♣ , *where* , *args*);

  - Step4) Close

    > Δ.**close**();

```
// 2. create ContentValues to add key "column"/value
ContentValues values = new ContentValues();
values.put("title", book.getTitle()); // get title
values.put("author", book.getAuthor()); // get author

// 3. updating row
int i = db.update(TABLE_BOOKS, //table
        values, // column/value
        KEY_ID+" = ?", // selections
        new String[] { String.valueOf(book.getId()) });
```

5.4- SQLite

- ## Get records
  - Step1) Get readable DB (△) from OurDatabase (♦)

    △ = ♦ .**getReadableDatabase**();

  - Step2) Get records from DB

    Cursor ♥ = △.**query**(*table* , *columns* , *where* , *args*);
    Cursor ♥ = △.**rawQuery**(*SQL* , *args*);

  - Step3) Get each record in turns

    ♥.**moveToFirst**();
    do{ ◇ = ♥.**getString**(*col_idx*); ... or getInt/Long...
    }**while**(♥.**moveToNext**());

  - Step4) Close

    △.**close**();

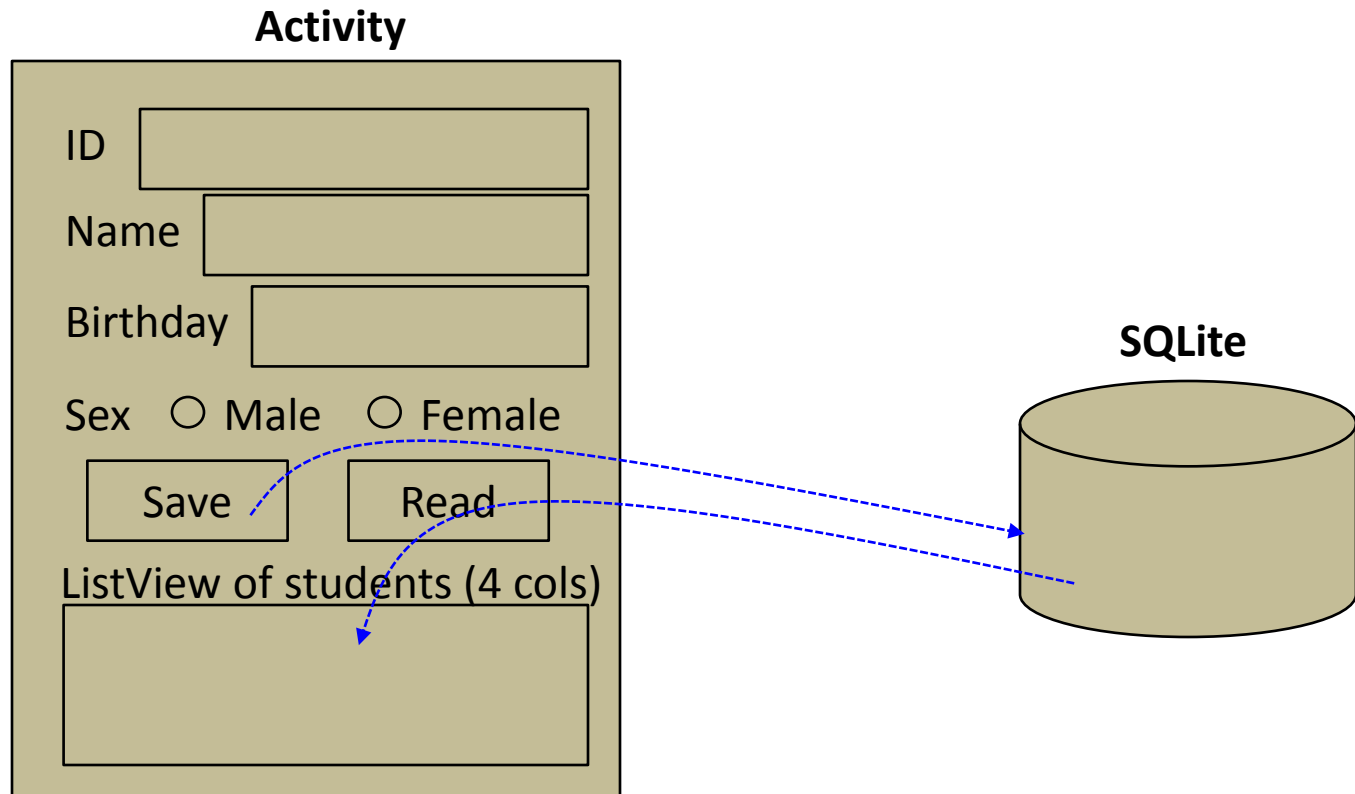   **process each value of
    each record**

5.4- SQLite

- **Example 5.3: access database by a class**

```java
class  MyDB  extends  SQLiteOpenHelper{
    static final String dbName="myDB1.db";
    MyDB(Context  ct){ super(ct,dbName,null,1); }
    public void onCreate(SQLiteDatabase  db){
        db.execSQL("create table student ( sid integer primary key," +
                    "name nvarchar(50), birthday varchar(10), sex boolean )");
    }
    public void onUpgrade(SQLiteDatabase  db, int oV, int nV){}
    public void addStudent(String  name, String  birthday, boolean  sex){
        SQLiteDatabase  db=getWritableDatabase();
        db.execSQL("insert into student(name,birthday,sex) values ('"+
                        name+"','"+birthday+"',"+(sex?0:1)+")");
        db.close();
    }
    String  getAllRecords(){
        SQLiteDatabase  db=getReadableDatabase();
        Cursor  cs=db.rawQuery("select * from student", null); String  kq="";
        if(cs!=null){  cs.moveToFirst();
            do{  kq+=cs.getString(0)+" *** "+cs.getString(1)+" *** "+
                        cs.getString(2)+" *** "+cs.getInt(3)+"\n";
            }while(cs.moveToNext());
        }
        db.close();
        return  kq;
    }
}
```

5.4- SQLite

- **Exercise 5.1: input & store student's info**

Activity

ID

Name

Birthday

Sex ○ Male ○ Female

Save   Read

ListView of students (4 cols)

SQLite

5.5- Internet

- **3W?**
  - Web applications
    - Need browsers
    - Need network for running
    - Only programming server site
  - Client/Server applications
    - Programming both client and server site
    - Cached data for offline using
    - Can use many services on client (android device)

5.5- Internet

- ## How?
  - – Declare permission in "AndroidManifest.xml"

    <uses-permission android:name = "android.permission.INTERNET"/>

  - – Connect by HTTP: GET / POST method
    - Parameters is sent with URL request
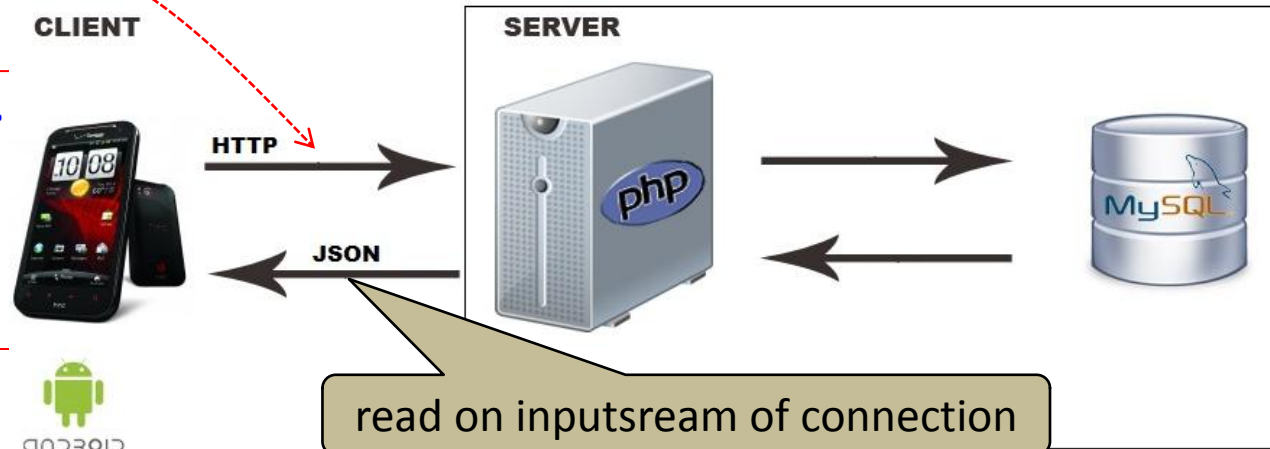   - Parameters is sent on independent stream

http://xyz?par1=val1&par2=...
or
http://xyz
then send parameters on
outputstream

CLIENT

SERVER

HTTP

JSON

read on inputsream of connection

5.5- Internet

- **How? GET method**
  - Step1) Make a connection to server

    ♥ = new **URL**("url_address_with_parameters");
    △ = (**HttpURLConnection**)♥.openConnection();
    if(△.**getResponseCode**() == HTTP_OK) _next step_...

  - Step2) Get the response stream from server

    ♣ = △.**getInputStream**();
    while((α = ♣.**read**()) != -1){
        ..._process the read byte in α_...
    }

    > we can decode to Bitmap as:
    > ♠ = BitmapFactory.decodeStream(♣);

  - Step3) Disconnect

    ♣.**close**();
    △.**disconnect**();

    > convert to String as:
    > ⊕=ByteArrayOutputStream{ write(α) }
    > → new String(⊕.toByteArray(), "utf-8");

5.5- Internet

- **Example**

```
22⊖    Runnable  r = new  Runnable() {
23⊖        public  void  run(){
24            try{
25                URL   url = new URL("http://hou.edu.vn/index.php");
26                HttpURLConnection   hc = (HttpURLConnection)url.openConnection();
27                if(hc.getResponseCode()==HttpURLConnection.HTTP_OK){
28                    InputStream   is = hc.getInputStream();
29                    ByteArrayOutputStream   bs = new ByteArrayOutputStream();
30                    int   aByte;
31                    while((aByte=is.read())!=-1){  bs.write(aByte);   }
32                    kq = new String(bs.toByteArray(), "utf-8");
33                    hc.disconnect();
34                }else{
35                    kq="Lỗi:\n"+hc.getResponseMessage();
36                }
37            }catch(Exception  e){
38                kq="Lỗi:\n"+e.toString();
39            }
40⊖            handler.post(new  Runnable() {
41⊖                public void run() {
42                    tv.setText(kq);
43                }
44            });
45        }
46    };
```

run on new Thread

5.5- Internet

- **JSON (JavaScript Object Notation) parse**
  - JSONArray [] & JSONObject {}

    [{"attr1":"val1","attr2":"val2",...}, {...}, ...]

  - Step1) Convert the response in byte[] to String

    ♥ = new String(byte[] , "utf-8");

    > **JSONArray, JSONObject can be nested in each other**

  - Step2) Create new JSONArray

    △ = new JSONArray(♥);

  - Step3) Get JSONObject idxth from JSONArray

    ◊ = △.optJSONObject(idx);

    > getInt/Long/Float/...

  - Step4) Get value of attributes in JSONObject

    α = ◊.getString("attribute_name_of_object");

5.5- Internet

- ## JSON example

  - ## PHP for responding JSON format (get_json.php)

```php
<?php
  header('Content-type: application/json');
  $data1 = array();
  $data1[0] = array();
  $data1[0]["name"] = "Churchill";     $data1[0]["nation"] = "UK";
  $data1[1] = array();
  $data1[1]["name"] = "Obama";         $data1[1]["nation"] = "USA";
  echo json_encode( $data1 );
?>
```

lhlk.esy.es/teaching/get_json.php
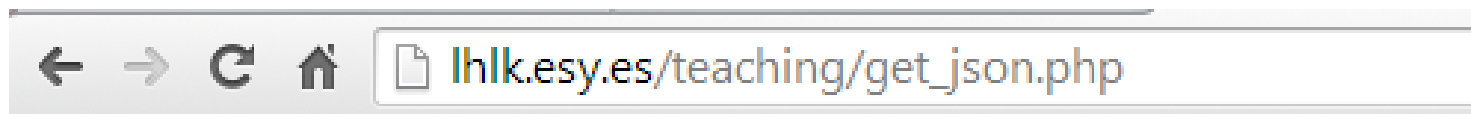
```json
[{"name":"Churchill","nation":"UK"},{"name":"Obama","nation":"USA"}]
```

5.5- Internet

- **JSON example**
  - Get connection & read the results → String

```
29    Thread  t1=new Thread(){
30      public void run(){
31        try{
32          URL url=new URL("http://lhlk.esy.es/teaching/get_json.php");
33          HttpURLConnection  hc=(HttpURLConnection)url.openConnection();
34          if(hc.getResponseCode()==hc.HTTP_OK){
35            InputStream  is=hc.getInputStream();
36            ByteArrayOutputStream os=new ByteArrayOutputStream();
37            int  b;
38            while((b=is.read())!=-1) os.write(b);
39            is.close();
40            kq = new String(os.toByteArray(),"utf-8");
41            handler.post(r1);
42            os.close();
43          }
44        }catch(Exception e){ e.printStackTrace(); }
45      }
46    };
```

5.5- Internet

- **JSON example**
  - Parsing JSON from String

```
47   Runnable r1=new Runnable() {
48      public void run() {
49         String  kv=kq+"\n\n"+"RESULTS\n\n";
50         try{
51            JSONArray js=new JSONArray(kq);
52            int  len = js.length();
53            for(int i=0;i<len;i++){
54               JSONObject jsi = js.optJSONObject(i);
55               kv = kv + "idx["+i+"]    ";
56               kv += jsi.getString("name")+"    "+
57                     jsi.getString("nation")+"\n";
58            }
59         }catch(Exception e){ e.printStackTrace(); }
60         et.setText(kv);
61      }
62   };
```

MyApplication

[{"name":"Churchill","nation":"UK"},{"name":"Obama","nation":"USA"}]

RESULTS

idx[0] Churchill UK
idx[1] Obama USA

5.5- Internet

- **Exercise 5.2: input name & score send to server for adding new user into mysql**