Advanced Control in Android

Dac Hoang Nguyen

Introduction

- Your role
- Your background and experience in the subject:
  - Java.
  - Basic Android.
- What do you want from this course

Course Objectives

- At the end of the course, you will have acquired sufficient knowledge to:
  - Understand Advanced control in Android: Listview, GridView…
  - Be able to use Advanced control with dataset to develop Android application.

Agenda

I. **Advanced control in Android**

II. **Practice**

Set Up Environment

- To complete the course, your PC must install:
  - Eclipse with Android plugins
  - Android SDK

Course Administration

- In order to complete the course you must:
  - Sign in the Class Attendance List
  - Participate in the course
  - Provide your feedback in the End of Course Evaluation

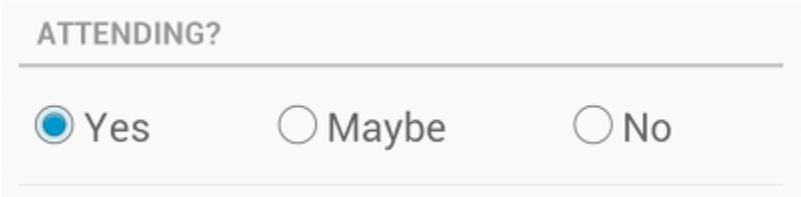
Assessment Disciplines

- Class Participation: 40%
- Assignment: 60%
- Final Exam: 0%
- Passing Scores: 70%

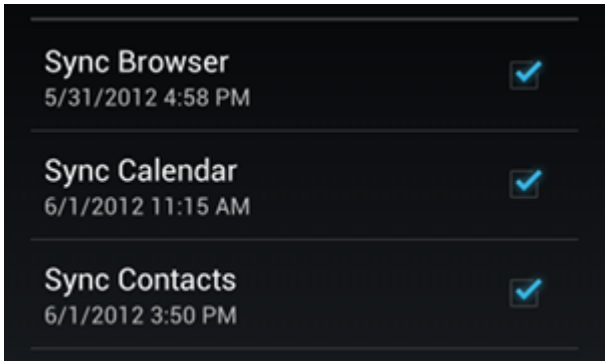
Advanced control in Android

Advanced control in Android

- RadioButton and CheckBox are suitable for a small set of options.
- RadioButton:



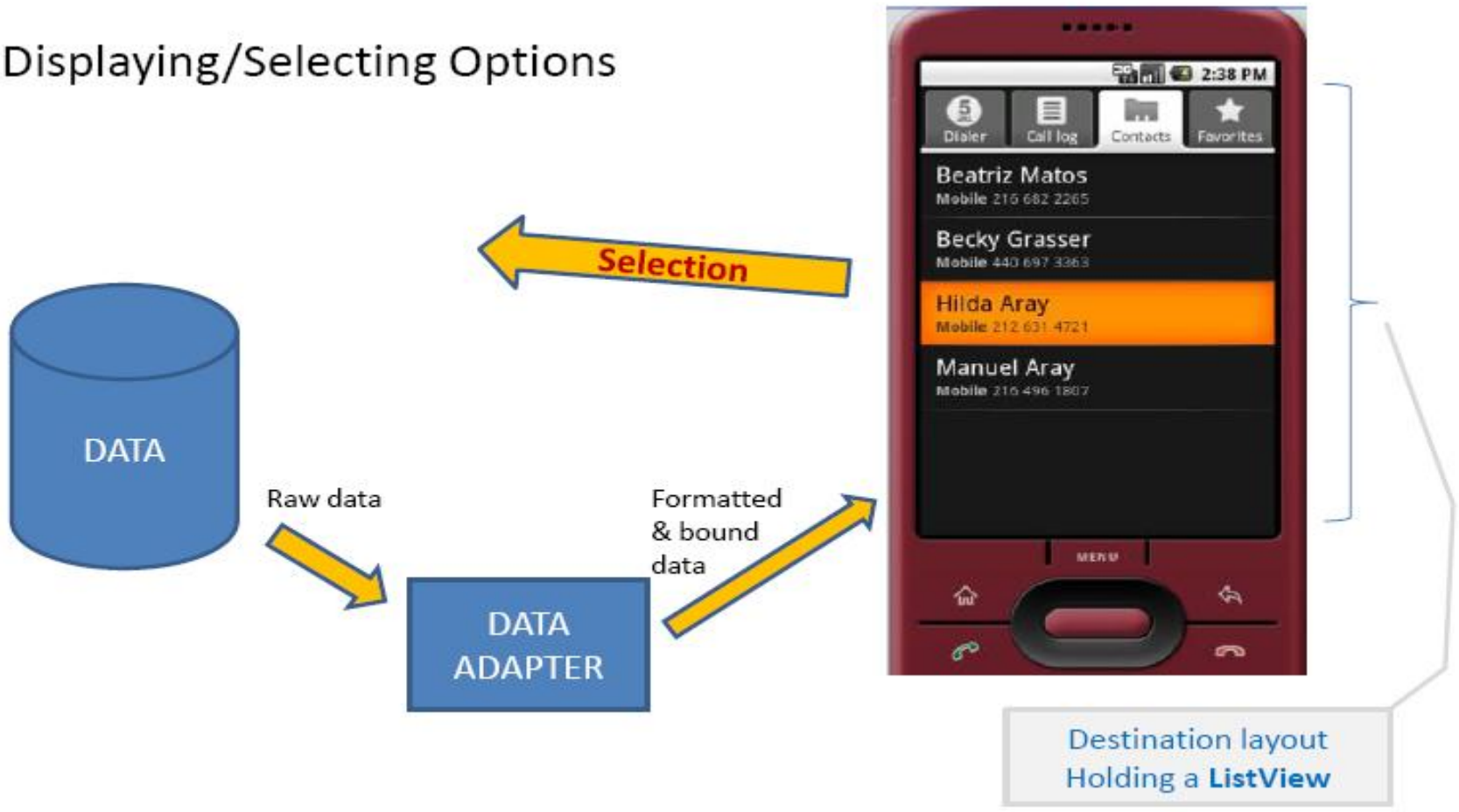
- CheckBox:



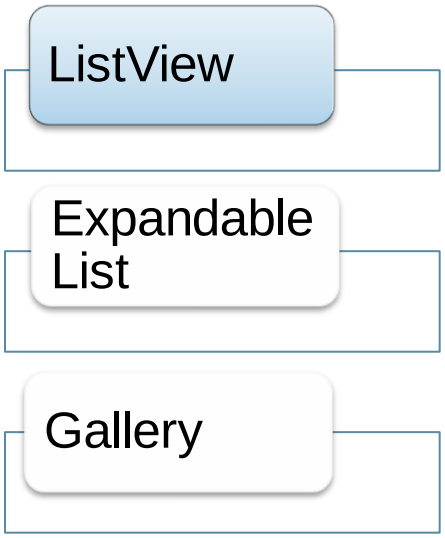
Advanced control in Android

- When we have a large, dynamic options, we use more appropriate controls: ListView, ExpandableList, Gallery….
- Android offers a framework of data adapters to get data and display in list.

Advanced control in Android



Android Layout: Common layout



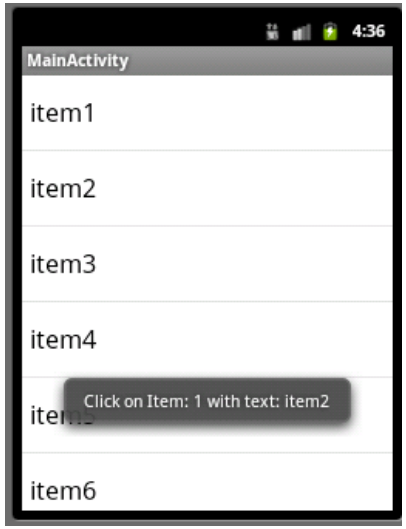
- ListView is a view group that displays a list of items.
- List items are get from list's adapter.
- We can use ListView in Layout or can have a separate ListView Activity.
- We can custom ListView by custom list items.

Android Layout: Common layout

- ListView
- Expandable List
- Gallery

- Example:

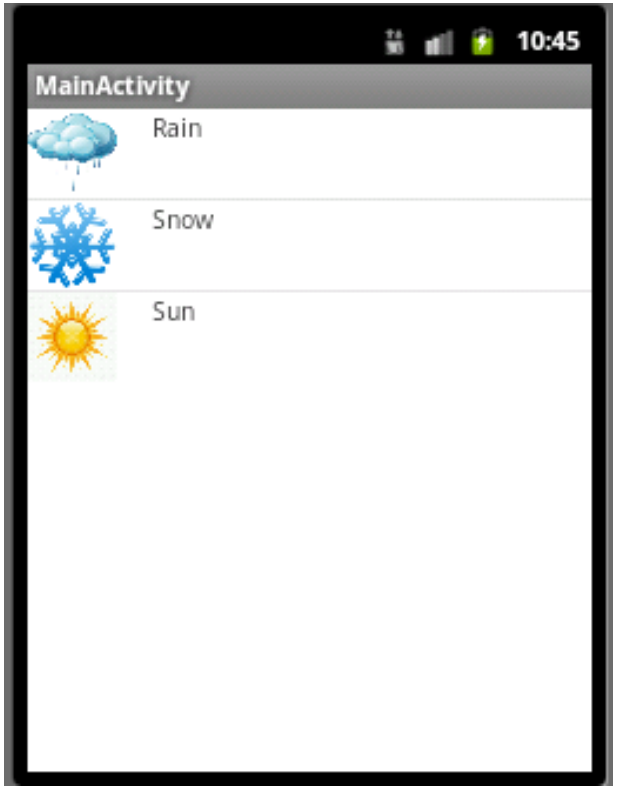
```
private String[] items;
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    ListView lv = (ListView)findViewById(R.id.listView1);
   items = new String[] {"item1","item2","item3","item4","item5","item6"};
    ArrayAdapter<String> arrAdap = new ArrayAdapter<String>(this,android.R.layout.simple_list_item_1, items);
    lv.setAdapter(arrAdap);
    lv.setOnItemClickListener(new OnItemClickListener() {

        @Override
        public void onItemClick(AdapterView<?> arrAdap, View listView, int position,
                long id) {
            // TODO Auto-generated method stub
            String text = items[position];
            Toast.makeText(getApplicationContext(),
                    "Click on Item: " + position + " with text: " + text,
                    Toast.LENGTH_SHORT).show();
        }
    });
}
```



Android Layout: Common layout

- ListView
- Expandable List
- Gallery

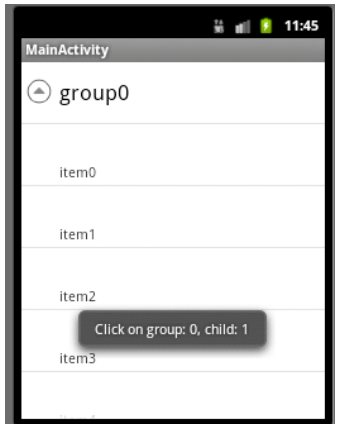
- Custom ListView:



Android Layout: Common layout

- ListView
- Expandable List
- Gallery

```
List<Map<String, String>> groups;
List<List<Map<String, String>>> items;
private static final String GROUPS = "GROUPS";
private static final String ITEMS = "ITEMS";
@Override
protected void onCreate(Bundle savedInstanceState) {
    // TODO Auto-generated method stub
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main_2);
    ExpandableListView expList = (ExpandableListView)findViewById(R.id.expandableListView1);
    groups = new ArrayList<Map<String,String>>();
    items = new ArrayList<List<Map<String,String>>>();
    for(int i = 0; i< 5; i++)
    {
        Map<String, String> map = new HashMap<String, String>();
        map.put(GROUPS, "group" + i);
        groups.add(map);
        List<Map<String, String>> children = new ArrayList<Map<String,String>>();
        for(int j=0;j<5;j++)
        {
            Map<String, String> mapChildren = new HashMap<String, String>();
            mapChildren.put("ITEMS", "item"+j);
            children.add(mapChildren);
        }
        items.add(children);
    }

    ExpandableListAdapter expAdap = new SimpleExpandableListAdapter(this, groups, android.R.layout.simple_expandable_list_item_1,
            new String[]{GROUPS,ITEMS}, new int[]{android.R.id.text1,android.R.id.text2},items,android.R.layout.simple_expandable_list_item_2,
            new String[]{GROUPS,ITEMS},new int[]{android.R.id.text1,android.R.id.text2});
    expList.setAdapter(expAdap);
    expList.setOnChildClickListener(new OnChildClickListener() {

        @Override
        public boolean onChildClick(ExpandableListView parent, View v,
                int groupPosition, int childPosition, long id) {
            // TODO Auto-generated method stub
            Toast.makeText(getApplicationContext(), "Click on group: " + groupPosition + ", child: " + childPosition, Toast.LENGTH_SHORT).show();
            return false;
        }
```



Android Layout: Common layout

ListView

Expandable List

Gallery

- Gallery control:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    // TODO Auto-generated method stub
    super.onCreate(savedInstanceState);
    setContentView(R.layout.gallery_view);
    Gallery gallery = (Gallery)findViewById(R.id.gallery1);
    gallery.setAdapter(new ImageAdapter(this));
    gallery.setOnItemClickListener(new OnItemClickListener() {

        @Override
        public void onItemClick(AdapterView<?> parent, View view, int pos,
                long id) {
            // TODO Auto-generated method stub
            Toast.makeText(getApplicationContext(),
            "Click on item: " + parent.getItemIdAtPosition(pos), Toast.LENGTH_SHORT).show();

        }
    });
}
```

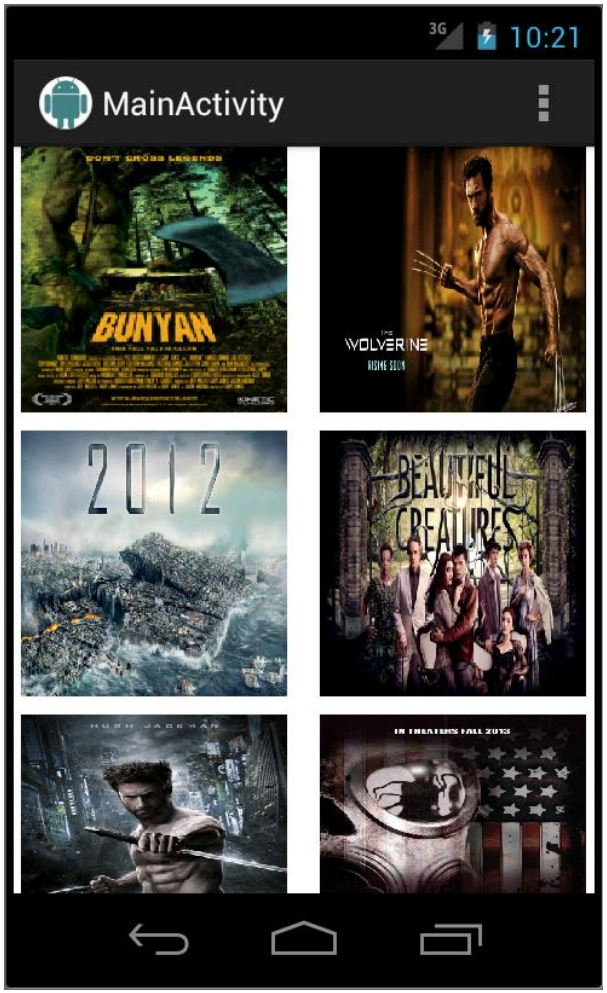
Android Layout: Common layout

ListView

Expandable List

Gallery

- Gallery using Gridview




Q&A

Thank You


CSC
BUSINESS SOLUTIONS
TECHNOLOGY
OUTSOURCING