Android ListView example

By [mkyong](http://www.mkyong.com/author/mkyong/" \o "mkyong) | January 12, 2012 | Updated : August 29, 2012 | Viewed : 606,222 times +1,447 pv/w

In Android, [ListView](http://developer.android.com/reference/android/widget/ListView.html" \t "_blank) let you arranges components in a vertical scrollable list.

In this tutorial, we will show you 2 ListView examples :

1. Normal way to display components in ListView.
2. Custom array adapter to customize the item display in ListView.

*P.S This project is developed in Eclipse 3.7, and tested with Android 2.3.3.*

1. Normal ListView example

In this example, we show you how to display a list of fruit name via ListView, it should be easy and self-explanatory.

*1.1 Android Layout file*

File : res/layout/list\_fruit.xml

<?xml version="1.0" encoding="utf-8"?>

<TextView xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:padding="10dp"

android:textSize="20sp" >

</TextView>

*1.2 ListView*

package com.mkyong.android;

import android.app.ListActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.ListView;

import android.widget.TextView;

import android.widget.Toast;

import android.widget.AdapterView.OnItemClickListener;

public class ListFruitActivity extends ListActivity {

static final String[] FRUITS = new String[] { "Apple", "Avocado", "Banana",

"Blueberry", "Coconut", "Durian", "Guava", "Kiwifruit",

"Jackfruit", "Mango", "Olive", "Pear", "Sugar-apple" };

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

// no more this

// setContentView(R.layout.list\_fruit);

setListAdapter(new ArrayAdapter<String>(this, R.layout.list\_fruit,FRUITS));

ListView listView = getListView();

listView.setTextFilterEnabled(true);

listView.setOnItemClickListener(new OnItemClickListener() {

public void onItemClick(AdapterView<?> parent, View view,

int position, long id) {

// When clicked, show a toast with the TextView text

Toast.makeText(getApplicationContext(),

((TextView) view).getText(), Toast.LENGTH\_SHORT).show();

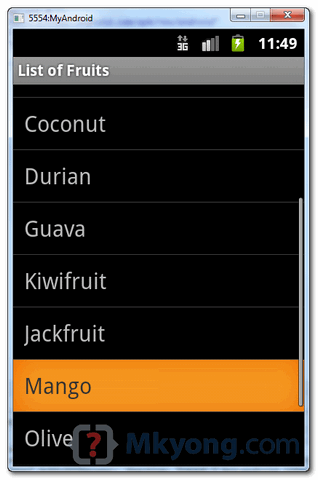
}

});

}

}

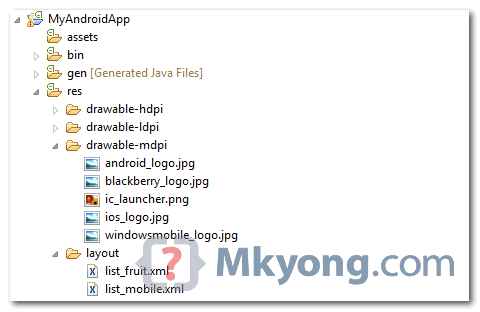
*1.3 Demo*



2. Custom ArrayAdapter example

In this example, we show you how to create 4 items in the ListView, and use a custom “ArrayAdapter” to display different images base on the “item name” in the list.

*2.1 Images*  
Get 4 images for demonstration.



*2.2 Android Layout file*  
File : res/layout/list\_mobile.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:padding="5dp" >

<ImageView

android:id="@+id/logo"

android:layout\_width="50px"

android:layout\_height="50px"

android:layout\_marginLeft="5px"

android:layout\_marginRight="20px"

android:layout\_marginTop="5px"

android:src="@drawable/windowsmobile\_logo" >

</ImageView>

<TextView

android:id="@+id/label"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@+id/label"

android:textSize="30px" >

</TextView>

</LinearLayout>

*2.3 Custom ArrayAdapter*  
Create a class extends ArrayAdapter and customize the item display in the getView() method.

package com.mkyong.android.adaptor;

import com.mkyong.android.R;

import android.content.Context;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.ArrayAdapter;

import android.widget.ImageView;

import android.widget.TextView;

public class MobileArrayAdapter extends ArrayAdapter<String> {

private final Context context;

private final String[] values;

public MobileArrayAdapter(Context context, String[] values) {

super(context, R.layout.list\_mobile, values);

this.context = context;

this.values = values;

}

@Override

public View getView(int position, View convertView, ViewGroup parent) {

LayoutInflater inflater = (LayoutInflater) context

.getSystemService(Context.LAYOUT\_INFLATER\_SERVICE);

View rowView = inflater.inflate(R.layout.list\_mobile, parent, false);

TextView textView = (TextView) rowView.findViewById(R.id.label);

ImageView imageView = (ImageView) rowView.findViewById(R.id.logo);

textView.setText(values[position]);

// Change icon based on name

String s = values[position];

System.out.println(s);

if (s.equals("WindowsMobile")) {

imageView.setImageResource(R.drawable.windowsmobile\_logo);

} else if (s.equals("iOS")) {

imageView.setImageResource(R.drawable.ios\_logo);

} else if (s.equals("Blackberry")) {

imageView.setImageResource(R.drawable.blackberry\_logo);

} else {

imageView.setImageResource(R.drawable.android\_logo);

}

return rowView;

}

}

*2.4 ListView*  
ListView, but use above custom adapter to display the list.

package com.mkyong.android;

import com.mkyong.android.adaptor.MobileArrayAdapter;

import android.app.ListActivity;

import android.os.Bundle;

import android.widget.ListView;

import android.widget.Toast;

import android.view.View;

public class ListMobileActivity extends ListActivity {

static final String[] MOBILE\_OS =

new String[] { "Android", "iOS", "WindowsMobile", "Blackberry"};

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setListAdapter(new MobileArrayAdapter(this, MOBILE\_OS));

}

@Override

protected void onListItemClick(ListView l, View v, int position, long id) {

//get selected items

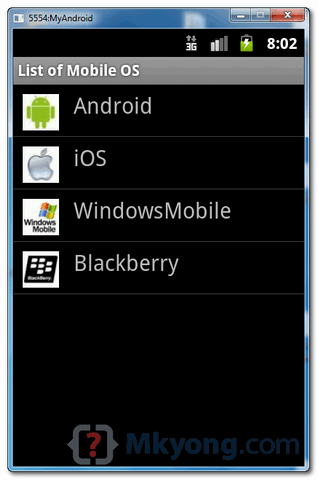
String selectedValue = (String) getListAdapter().getItem(position);

Toast.makeText(this, selectedValue, Toast.LENGTH\_SHORT).show();

}

}

*2.5 Demo*



Download Source Code

Download both examples – [Android-ListView-Example.zip](http://www.mkyong.com/wp-content/uploads/2012/01/Android-ListView-Example.zip) (21 KB)