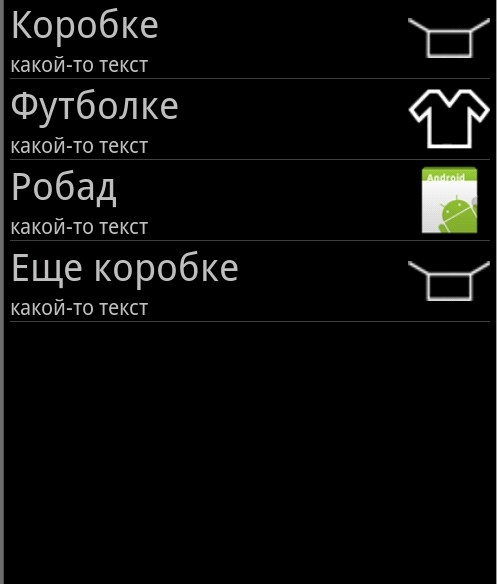
ListView - красивые списки на Android

Вчера появилась у меня задача сделать ListView с картинками с боку, полазил по интернетам немного очканул потому что такое количество кода для такой простой задачи я никогда не [видел](http://habrahabr.ru/blogs/android/78747/), по три четыре файла с кодом на 100 и более строк, нашел как создавать самый обычный ListView (без картинок) и начал экспериментировать, долго я занимался сексом с этим listview, и очень долго психовал и бился головой об стену, но потом мне пришло озарение, а зачем я делаю все в массивах когда можно сделать все намного проще, и сделал как можно проще. Вот об этом проще я сегодня Вам и расскажу. Кому интересно прошу подкат.  
  
  
  
Для тех кому не терпится вот то что у нас получится на выходе:  


И так приступаем к созданию нашего ListView. Для начала я расскажу про создание обычного ListView без картинок что бы было понятно что к чему, куда и за чем.  
  
Создаем проект в [Eclipse](http://www.eclipse.org/downloads/) — File — New — Progect — Android Project — ListviewActivity  
  
И заходим в res — layout — main.xml. Начинаем редактировать. Для того что бы у нас был красивый ListView нам нужно его описать.  
  
main.xml\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

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

    android:orientation="vertical"

    android:layout\_width="fill\_parent"

    android:layout\_height="fill\_parent"

    >

<ListView

    android:layout\_width="fill\_parent"

    android:layout\_height="fill\_parent"

    android:id="@+id/list"

    />

</LinearLayout>

**LinearLayout** — главное поле на котором находится все остальные формы.  
**ListView**— наш заветный ListView который мы сейчас делаем. Пока что это только формачка которая даже толком не отображается на форме.  
  
И создаем второй файл — list.xml, который будет отображать текст в listview. Код расположился ниже:  
  
list.xml\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout

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

  android:layout\_width="wrap\_content"

  android:orientation="horizontal"

  android:layout\_height="wrap\_content">

  <LinearLayout

  android:layout\_width="265dip"

  android:orientation="vertical"

  android:layout\_height="wrap\_content">

  <TextView android:layout\_width="fill\_parent"

  android:layout\_height="wrap\_content"

  android:id="@+id/text1"

  android:textSize="25dip"

  android:text="This is text1"/>

  <TextView android:layout\_width="fill\_parent"

  android:layout\_height="wrap\_content"

  android:id="@+id/text2"

  android:text="This is text2"/>

  </LinearLayout>

</LinearLayout>

**LinearLayout** — главное поле на котором находится все остальные формы.  
**TextView** — наше поле в которые мы будем выводить текст из программы.  
  
Теперь нам нужно написать код который будет запускаться и заполнять все наши пустые поля. Для этого открываем наш класс ListviewActivity.java и вставляем вот такой код:  
  
ListviewActivity.java\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

import java.util.ArrayList;

import java.util.HashMap;

import android.app.Activity;

import android.os.Bundle;

import android.widget.ListView;

import android.widget.SimpleAdapter;

public class ListviewActivity extends Activity {

        private ArrayList <HashMap<String, Object>> myBooks; //наш массив объектов, без него никак

        private static final String BOOKKEY = "bookname";    // Главное название, большими буквами

        private static final String PRICEKEY = "bookprice";  // Наименование, то что ниже главного

    @Override

    public void onCreate(Bundle savedInstanceState)          //вход в программу

    {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.main);                       //говорим что запускать форму main.xml

        ListView listView = (ListView)findViewById(R.id.list);  //определяем наш ListView в main.xml

        myBooks = new ArrayList<HashMap<String,Object>>();      //создаем массив списков

        HashMap<String, Object> hm;                             //список объектов

        ///С помощью ключевого HashMap добавляем название (то что большими буквами), и описание (маленькими)

        hm = new HashMap<String, Object>();

        hm.put(BOOKKEY, "Коробке");                 //Название

        hm.put(PRICEKEY, "какой-то текст");         //Описание

        myBooks.add(hm);                            //Добавляем на форму для отображения, без этой функции мы не видим сам вью

        hm = new HashMap<String, Object>();

        hm.put(BOOKKEY, "Футболке");

        hm.put(PRICEKEY, "какой-то текст");

        myBooks.add(hm);

        hm = new HashMap<String, Object>();

        hm.put(BOOKKEY, "Робад");

        hm.put(PRICEKEY, "какой-то текст");

        myBooks.add(hm);

        hm = new HashMap<String, Object>();

        hm.put(BOOKKEY, "Еще коробке");

        hm.put(PRICEKEY, "какой-то текст");

        myBooks.add(hm);

       SimpleAdapter adapter = new SimpleAdapter(this,

                                                 myBooks,

             R.layout.list, new String[]{ // массив названий

             BOOKKEY,         //верхний текст

             PRICEKEY,        //нижний теккт

                                 }, new int[]{    //массив форм

             R.id.text1,      //наш id TextBox'a в list.xml

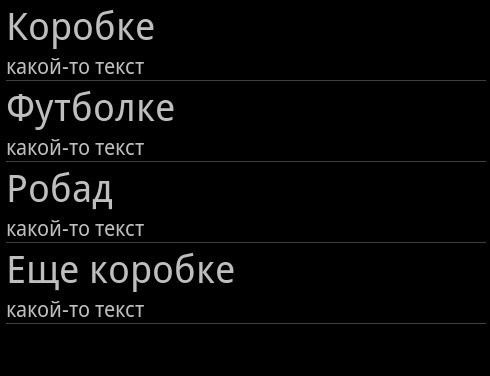
             R.id.text2});    //ссылка на объект отображающий текст

listView.setAdapter(adapter); //говорим программе что бы отображала все объекты

listView.setChoiceMode(ListView.CHOICE\_MODE\_SINGLE);  //Даем возможность выбора если есть желание сделать переход на другие формы

    }

}

И на выходе у нас получается вот такая красота:  


Если вы сделали все правильно то у Вас должно все работать как часы.  
  
Теперь давайте добавим картинки в наш проект, я свои картинки выкладывать не буду, предлагаю вставить обычный ic\_launcher ну или свои :) Как пожелаете.  
  
Открываем list.xml и перед последний LinearLayout вставляем  
  
list.xml\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

<ImageView android:layout\_width="55dip"

  android:layout\_height="fill\_parent"

  android:id="@+id/img"

  />

**ImageView** — картинка которая будет отображаться в нашем проекте.  
  
Вот так выглядит полностью весь list.xml  
  
list.xml\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

<LinearLayout

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

  android:layout\_width="wrap\_content"

  android:orientation="horizontal"

  android:layout\_height="wrap\_content">

  <LinearLayout

  android:layout\_width="265dip"

  android:orientation="vertical"

  android:layout\_height="wrap\_content">

  <TextView android:layout\_width="fill\_parent"

  android:layout\_height="wrap\_content"

  android:id="@+id/text1"

  android:textSize="25dip"

  android:text="This is text1"/>

  <TextView android:layout\_width="fill\_parent"

  android:layout\_height="wrap\_content"

  android:id="@+id/text2"

  android:text="This is text2"/>

  </LinearLayout>

  <ImageView android:layout\_width="55dip"

  android:layout\_height="fill\_parent"

  android:id="@+id/img"

  />

</LinearLayout>

Так как наш код для отображение изображений немного изменился но это маловидно, я напишу весь код сюда что бы не путать вас, что куда вставлять. Берем наш ListviewActivity.java и вставляем следующий код:  
  
ListviewActivity.java\_\_\_\_\_\_\_\_

import java.util.ArrayList;

import java.util.HashMap;

import android.app.Activity;

import android.os.Bundle;

import android.widget.ListView;

import android.widget.SimpleAdapter;

public class ListviewActivity extends Activity {

        private ArrayList <HashMap<String, Object>> myBooks;

        private static final String BOOKKEY = "bookname";

        private static final String PRICEKEY = "bookprice";

private static final String IMGKEY = "iconfromraw";  //Наша будущая картинка

    @Override

    public void onCreate(Bundle savedInstanceState)

    {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.main);

        ListView listView = (ListView)findViewById(R.id.list);

        myBooks = new ArrayList<HashMap<String,Object>>();

        HashMap<String, Object> hm;

        hm = new HashMap<String, Object>();

        hm.put(BOOKKEY, "Коробке");

        hm.put(PRICEKEY, "какой-то текст");

        hm.put(IMGKEY, R.drawable.ic\_launcher); //тут мы её добавляем для отображения

        myBooks.add(hm);

        hm = new HashMap<String, Object>();

        hm.put(BOOKKEY, "Футболке");

        hm.put(PRICEKEY, "какой-то текст");

        hm.put(IMGKEY,  R.drawable.ic\_launcher); //тут мы её добавляем для отображения

        myBooks.add(hm);

        hm = new HashMap<String, Object>();

        hm.put(BOOKKEY, "Робад");

        hm.put(PRICEKEY, "какой-то текст");

        hm.put(IMGKEY,  R.drawable.ic\_launcher); //тут мы её добавляем для отображения

        myBooks.add(hm);

        hm = new HashMap<String, Object>();

        hm.put(BOOKKEY, "Еще коробке");

        hm.put(PRICEKEY, "какой-то текст");

        hm.put(IMGKEY,  R.drawable.ic\_launcher); //тут мы её добавляем для отображения

        myBooks.add(hm);

       SimpleAdapter adapter = new SimpleAdapter(this,

                                                 myBooks,

             R.layout.list, new String[]{

             BOOKKEY,         //верхний текст

           PRICEKEY,        //нижний теккт

           IMGKEY          //наша картинка

                                                 }, new int[]{

         R.id.text1, //ссылка на объект отображающий текст

         R.id.text2, //ссылка на объект отображающий текст

         R.id.img}); //добавили ссылку в чем отображать картинки из list.xml

        listView.setAdapter(adapter);

        listView.setChoiceMode(ListView.CHOICE\_MODE\_SINGLE);

    }

}