**Руководство Android ListView**

1- Что такое ListView?

1.1- ListItem

1.2- Adapter (Адаптор)

1.3- ListView Selector

2- Основной ListView использует ArrayAdapter

2.1- ArrayAdapter

2.2- Пример ListView с ArrayAdapter

2.3- Готовые Layout для работы с ArrayAdapter

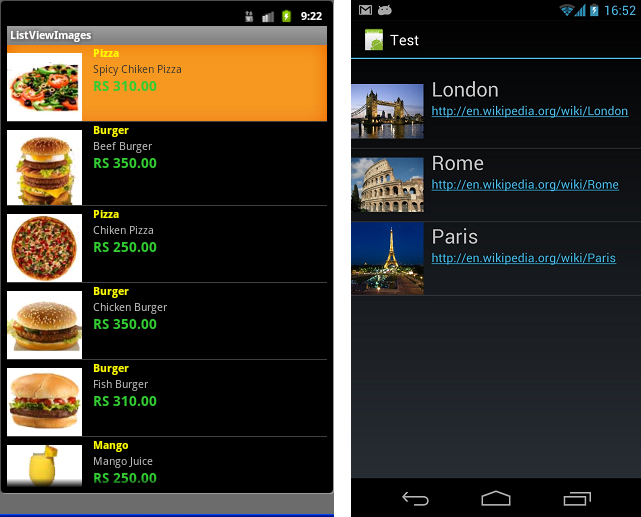
3- Кастомизировать ListView используя BaseAdapter

3.1- Пример кастомизированного ListView

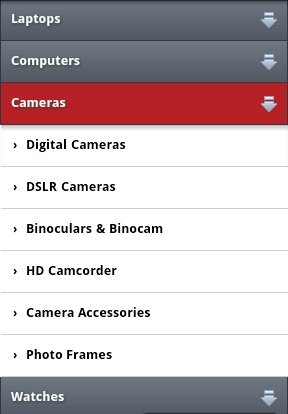
3.2- Пример кастомизации Selector

1- Что такое ListView?

**ListView** это  **view group**, отображающий элементы (elements) в виде списка, который можно прокручивать вертикально. **ListView** это важный view, который широко распространен в использовании для приложений **Android**. Простой пример **ListView** это ваша контактная книга, там где у вас есть список ваших контактов отображенных в одном  **ListView**.

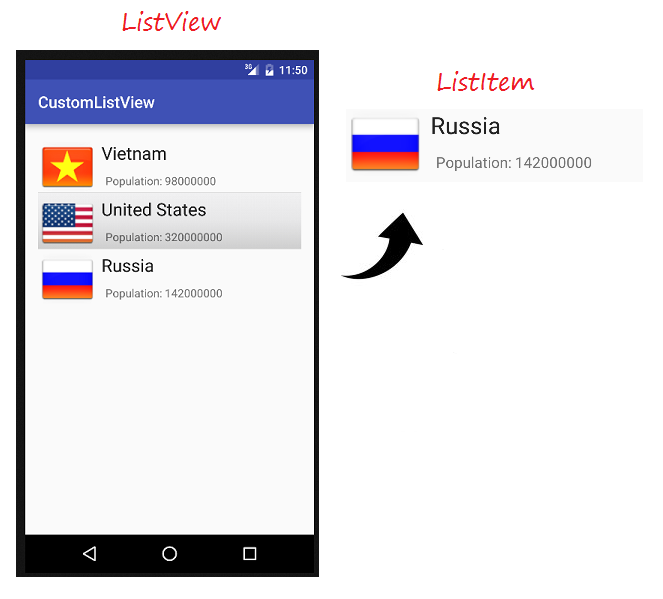


Помимо  **ListView**, **Android** так же предоставляет вам другой индентичный **view**это  **ExpandableListView**.

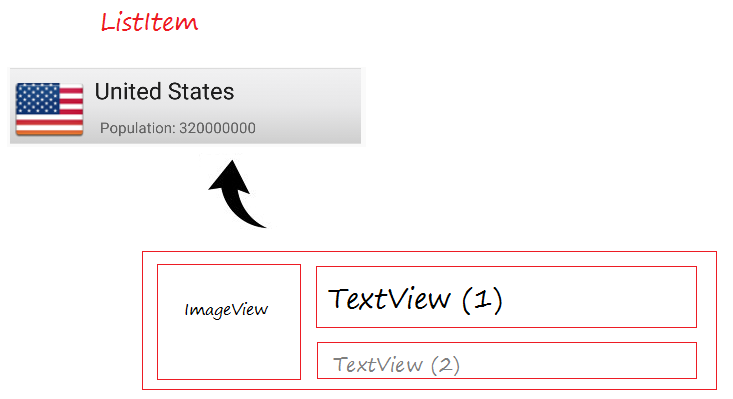


1.1- ListItem

**ListView**создан из списка  **ListItem**. **ListItem**это отдельная строка (row) в listview там где отображаются данные. Любые данные в listview только отображаются через listItem. Можно считать listview прокручиваемой группой **ListItem**.



**ListItem** это часть интерфейса, который может быть создан с помощью  **View**.

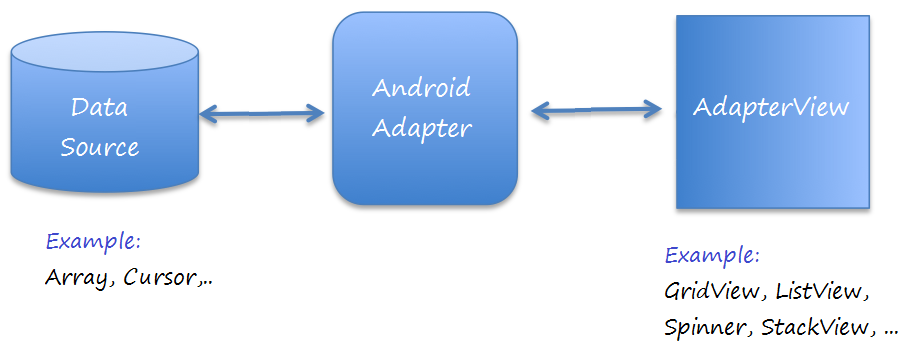


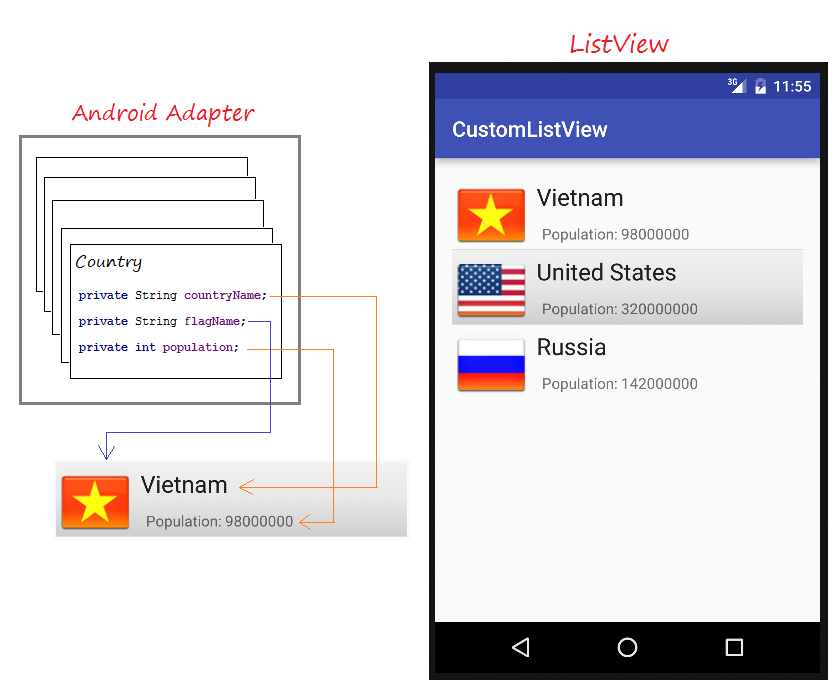
***Android****строит разные готорвые формы****ListItem****, называются предопределенные****Layout****, будут упомянуты мною в примерах данной статьи.*

1.2- Adapter (Адаптор)

**Android Adapter** это соединяющий мост между **View**(например  **ListView**) и основные данные того  **View**.**Adapter** управляет данными и соединяет с отдельные строки (ListItem) в view.

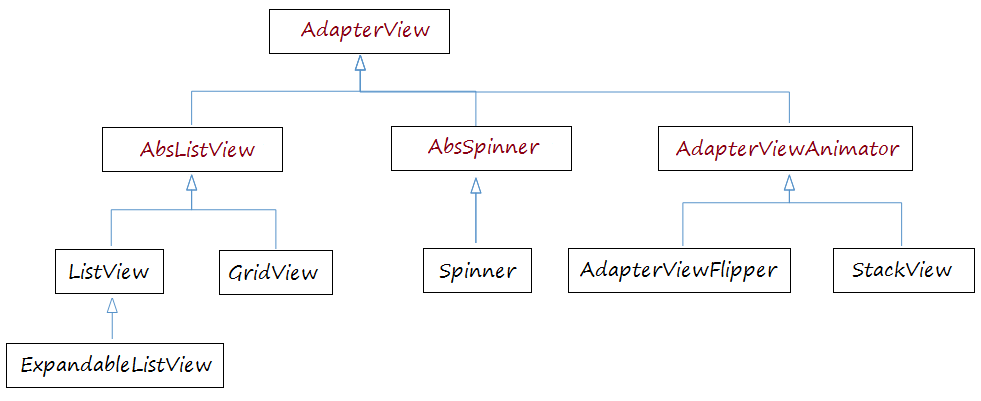
Вы можете прикрепить  **Adapter**с  **Android ListView**через метод  ***setAdapter***. Теперь, посмотрим как работает**Adapter**с помощью изображений ниже.



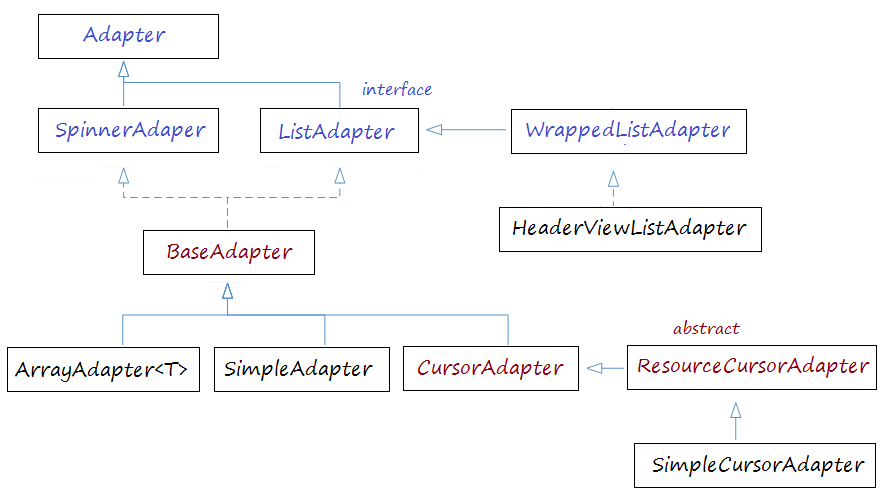


**AdapterView**

Многим  **View** нужен  **Android Adapter** для управления отображенных данных, эти  **View**являются подклассами **AdapterView**, вы можете посмотреть в изображении ниже:

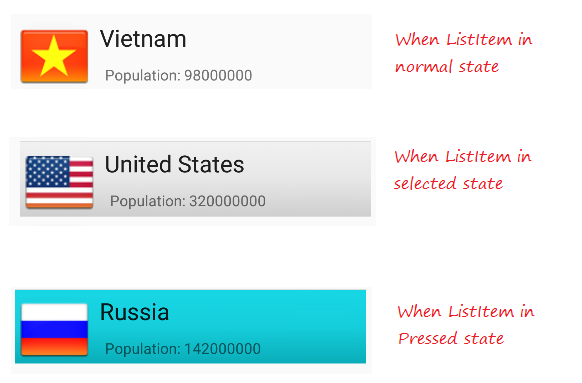


**Android Adapter**



1.3- ListView Selector

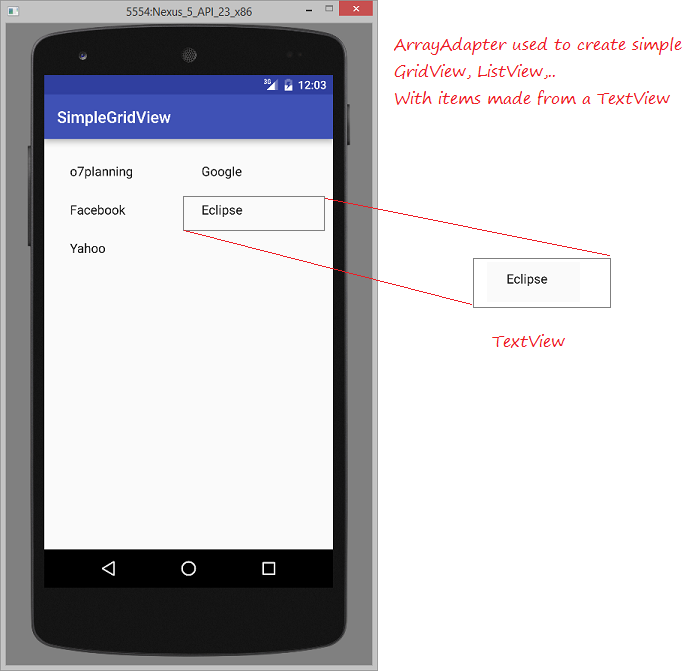
Чтобы  **ListView** отображался красивее, вам нужно кастомизировать эффекты, например при изменении цвета**ListItem**когда курсор перемещается на нем или когда меняется цвет фона когда выбран  **ListItem**. Вы можете посмотреть пример кастомизации **ListView Selector** в конце данной статьи.



2- Основной ListView использует ArrayAdapter

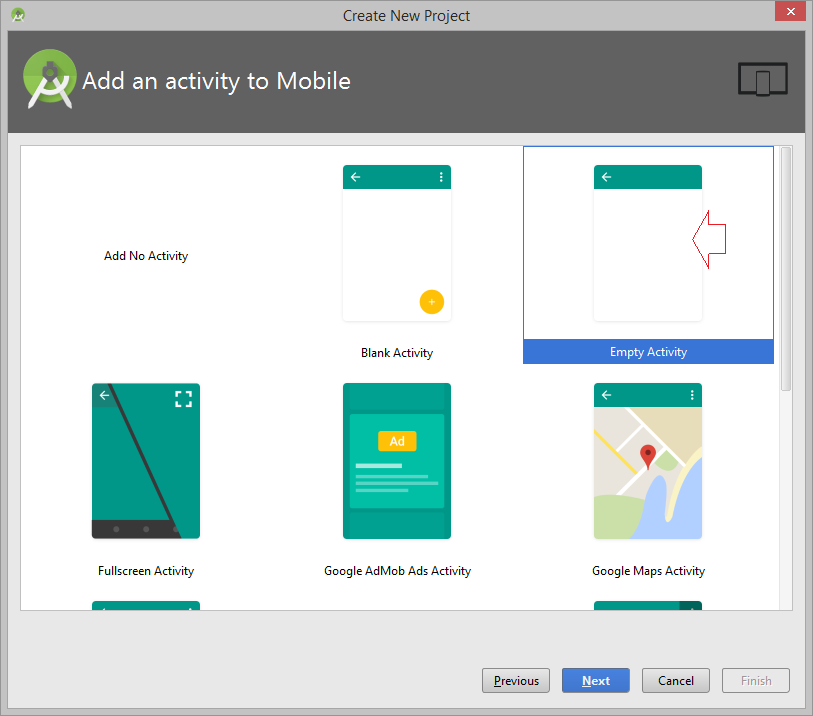
2.1- ArrayAdapter

**ArrayAdapter** используется для отображения  **ListView** с простыми  **ListItem**, **ListItem**можно создать из только одного**TextView**, **CheckedTextView**, **EditText**,...   
  
В случае если вы хотите иметь **ListView**с более сложным  **ListItem**, вы можете сами создать кастомизированный**Adapter**.

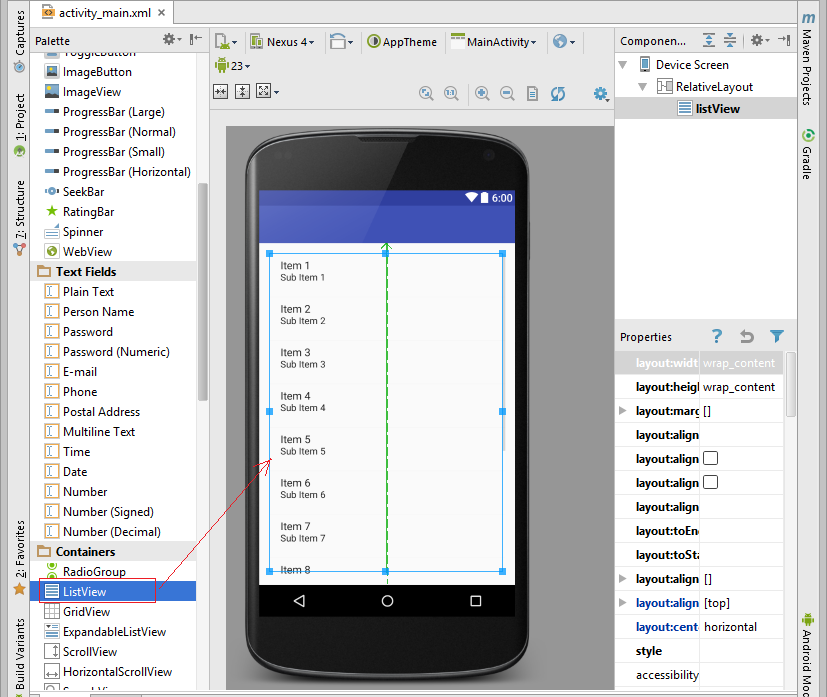


2.2- Пример ListView с ArrayAdapter

Создать новый Project с названием  **SimpleListView**.



* **activity\_main.xml**



**activity\_main.xml**

?

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

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

   xmlns:tools="<http://schemas.android.com/tools>"

   android:layout\_width="match\_parent"

   android:layout\_height="match\_parent"

   android:paddingBottom="@dimen/activity\_vertical\_margin"

   android:paddingLeft="@dimen/activity\_horizontal\_margin"

   android:paddingRight="@dimen/activity\_horizontal\_margin"

   android:paddingTop="@dimen/activity\_vertical\_margin"

   tools:context="org.o7planning.simplelistview.MainActivity">

   <ListView

       android:layout\_width="wrap\_content"

       android:layout\_height="wrap\_content"

       android:id="@+id/listView"

       android:layout\_alignParentTop="true"

       android:layout\_centerHorizontal="true" />

</RelativeLayout>

**UserAccount.java**

?

package org.o7planning.simplelistview;

import java.io.Serializable;

public class UserAccount implements Serializable {

   private String userName;

   private String userType;

   private boolean active;

   public UserAccount(String userName, String userType)  {

       this.userName= userName;

       this.userType = userType;

       this.active= true;

   }

   public UserAccount(String userName, String userType, boolean active)  {

       this.userName= userName;

       this.userType = userType;

       this.active= active;

   }

   public String getUserType() {

       return userType;

   }

   public void setUserType(String userType) {

       this.userType = userType;

   }

   public String getUserName() {

       return userName;

   }

   public void setUserName(String userName) {

       this.userName = userName;

   }

   public boolean isActive() {

       return active;

   }

   public void setActive(boolean active) {

       this.active = active;

   }

   @Override

   public String toString() {

       return this.userName +" ("+ this.userType+")";

   }

}

**MainActivity.java**

?

package org.o7planning.simplelistview;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.widget.ArrayAdapter;

import android.widget.ListView;

public class MainActivity extends AppCompatActivity {

   @Override

   protected void onCreate(Bundle savedInstanceState) {

       super.onCreate(savedInstanceState);

       setContentView(R.layout.activity\_main);

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

       //

       UserAccount tom = new UserAccount("Tom","admin");

  UserAccount jerry = new UserAccount("Jerry","user");

  UserAccount donald = new UserAccount("Donald","guest", false);

  UserAccount[] users = new UserAccount[]{tom,jerry, donald};

// android.R.layout.simple\_list\_item\_1 is a constant predefined layout of Android.

// used to create a ListView with simple ListItem (Only one TextView).

ArrayAdapter<UserAccount> arrayAdapter

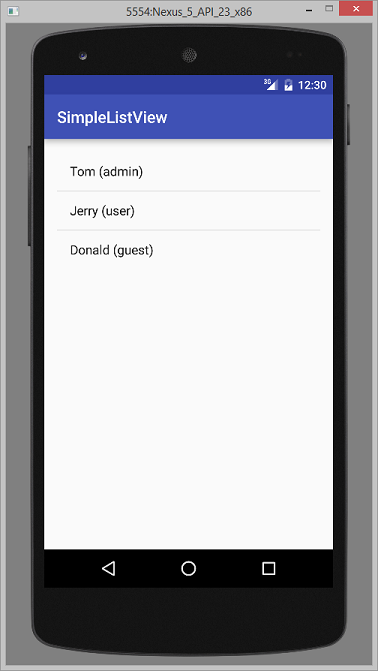
               = new ArrayAdapter<UserAccount>(this, android.R.layout.simple\_list\_item\_1 , users);

       listView.setAdapter(arrayAdapter);

   }

}

Запуск примера:



2.3- Готовые Layout для работы с ArrayAdapter

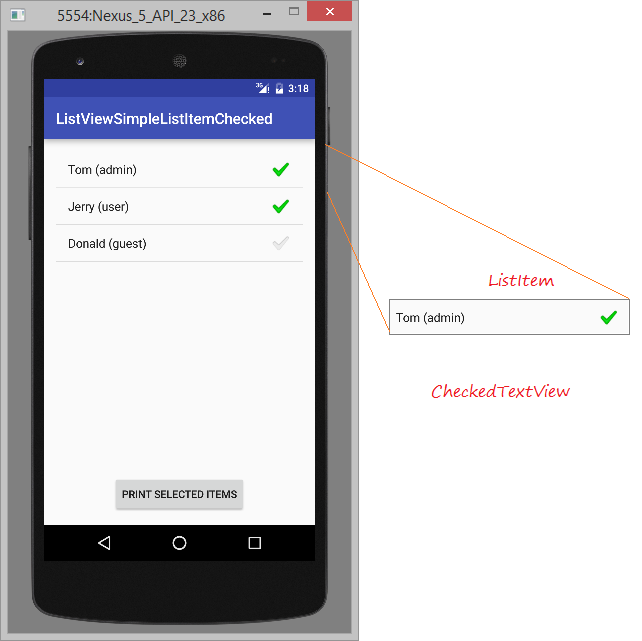
**Android**построил некоторые готовые  **Layout**(для ListItem) чтобы работать с  **ArrayAdapter**.

**android.R.layout.simple\_list\_item\_1**

* Это простой **Layout**в **ListItem**, созданный лишь одним **TextView** (Можете посмотреть пример выше).

**android.R.layout.simple\_list\_item\_checked  
   &  
android.R.layout.simple\_list\_item\_multiple\_choice**

* 2 **Layout**выше это простые layout для создания **ListView** с **ListItem** с одним **checkbox**.



Можете посмотреть пример по ссылке:

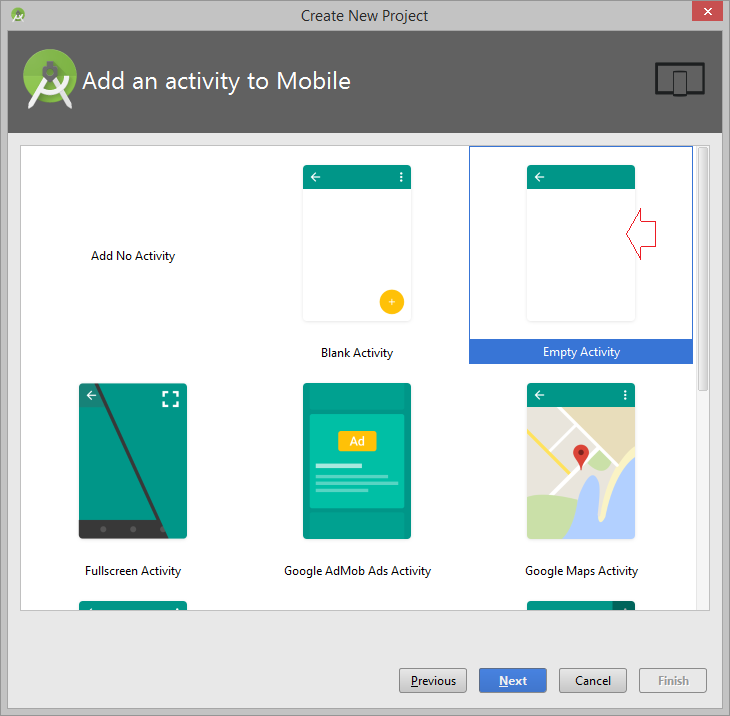
* [Android ListView с Checkbox с помощью ArrayAdapter](https://o7planning.org/ru/10923/android-listview-with-checkbox-using-arrayadapter)

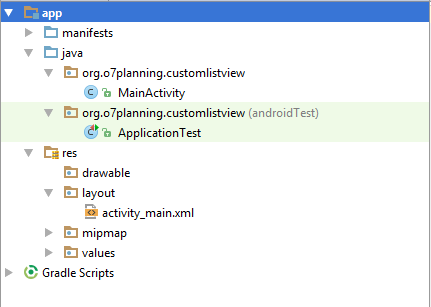
3- Кастомизировать ListView используя BaseAdapter

Вы можете кастомизировать  **ListViewer**. Вашему  **Adapter**стоит расширить из класса  **BaseAdapter**.

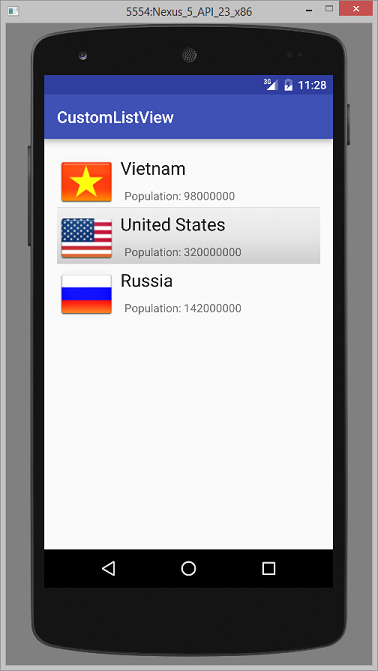
3.1- Пример кастомизированного ListView 

Создать новый  **"Empty Activity"** project с названием  **CustomListView**.





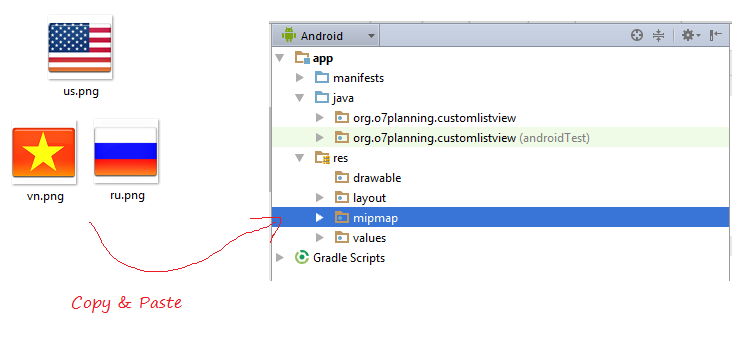
Это изображение приложения:

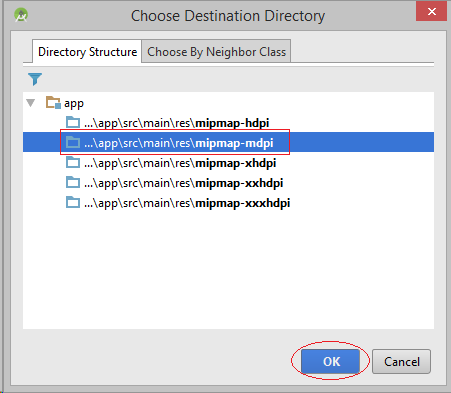


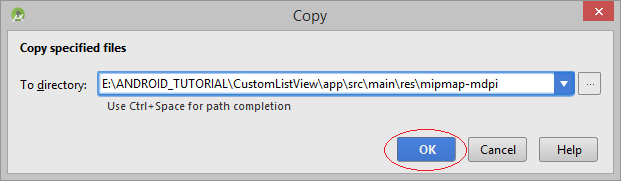
Для начала вам нужно приготовить несколько файлов изображения:

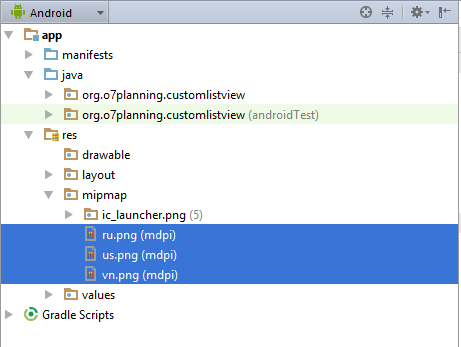
Копировать и вставить эти файлы изображения в папку **mipmap**:

1. vi.png
2. us.png
3. ru.png



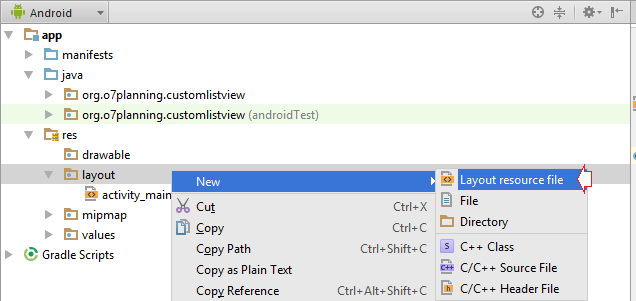






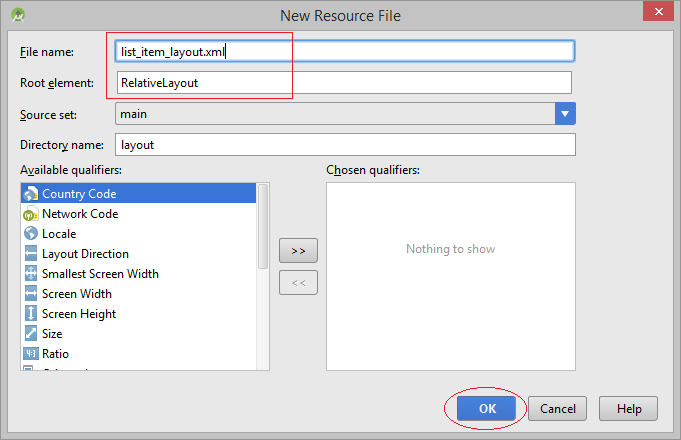
Вам нужно создать List item layouts. На  **Android Studio** нажмите на правую кнопку мыши на  **res/layout** выберите:

* **New/Layout resource file**

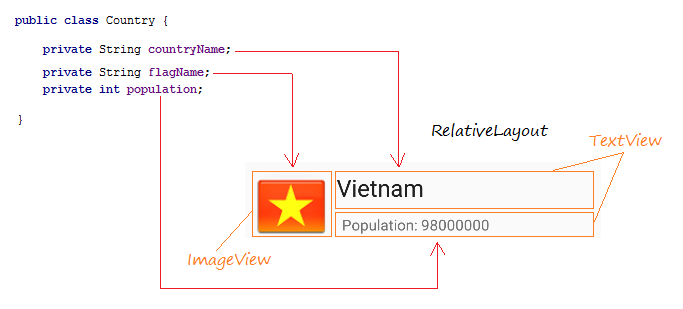


Ввести:

* **File name:** list\_item\_layout.xml
* **Root element:** RelativeLayout

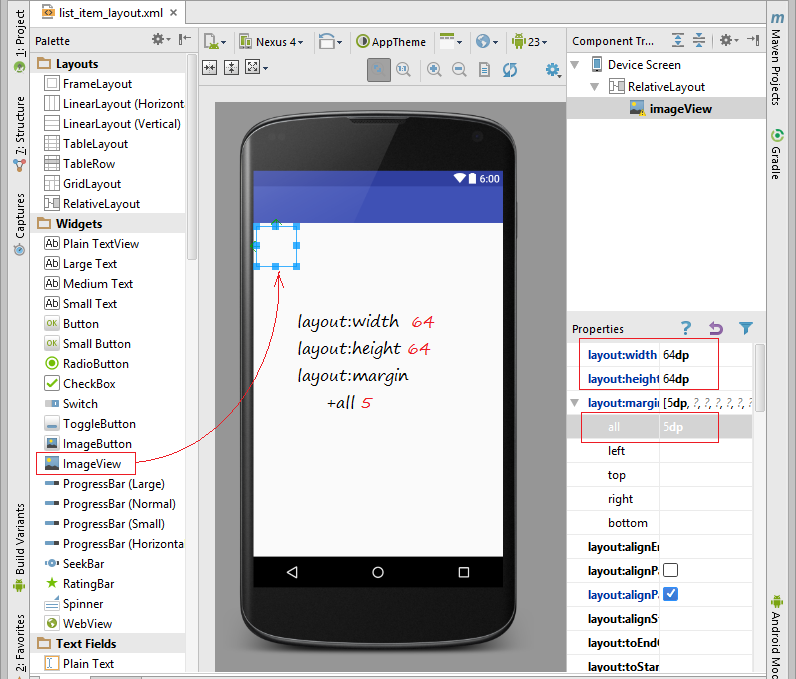


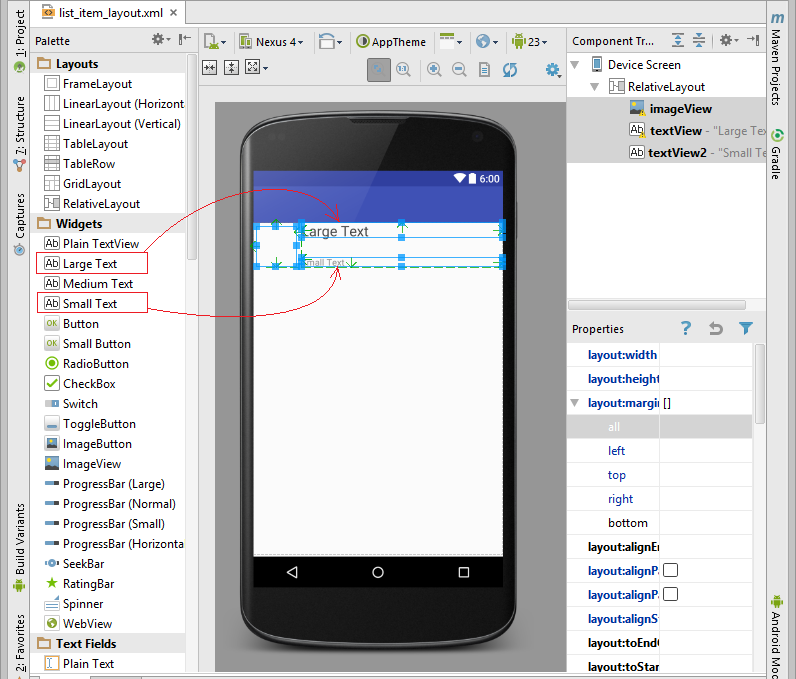
Сделать дизайн интерфейса для List Item.

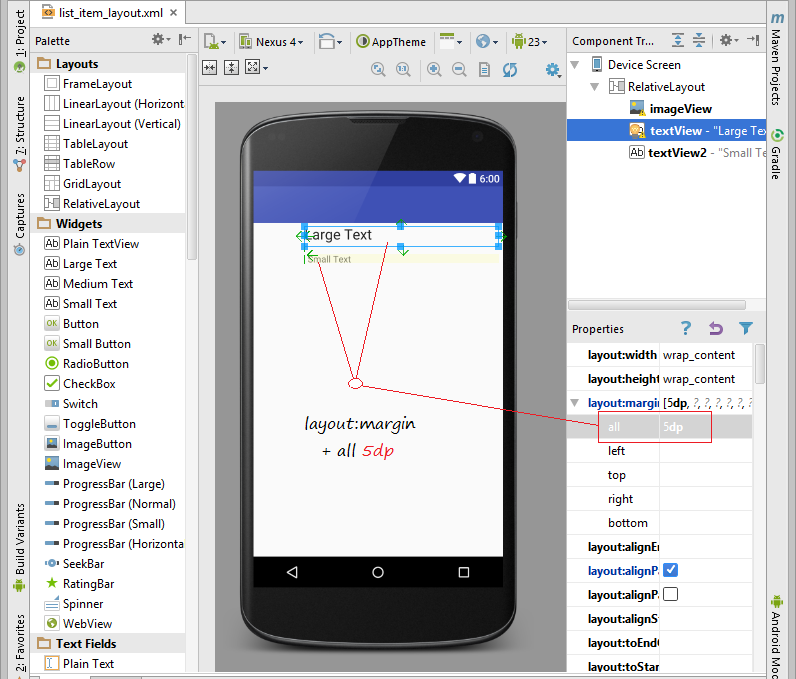


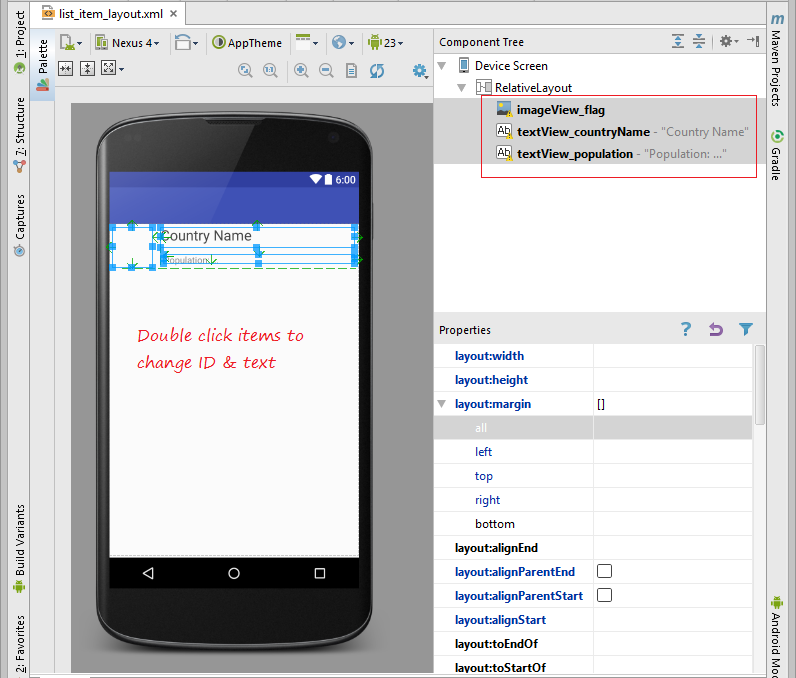
**Slider**шаги дизайна интерфейса **List Item**:

* **SLIDER**









ImageView

* **ID:** imageView\_flag

TextView 1:

* **ID:** textView\_countryName
* **Text:** Country Name

TextView 2:

* **ID:** textView\_population
* **Text:** Population ....

**list\_item\_layout.xml**

?

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

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

    android:layout\_width="match\_parent" android:layout\_height="match\_parent">

    <ImageView

        android:layout\_width="64dp"

        android:layout\_height="64dp"

        android:id="@+id/imageView\_flag"

        android:layout\_alignParentTop="true"

        android:layout\_alignParentLeft="true"

        android:layout\_alignParentStart="true"

        android:layout\_marginLeft="5dp"

        android:layout\_marginRight="5dp"

        android:layout\_marginTop="5dp" />

    <TextView

        android:layout\_width="wrap\_content"

        android:layout\_height="wrap\_content"

     android:textAppearance="?android:attr/textAppearanceLarge"

        android:text="Country Name"

        android:id="@+id/textView\_countryName"

        android:layout\_alignParentTop="true"

        android:layout\_toRightOf="@+id/imageView\_flag"

        android:layout\_alignParentRight="true"

        android:layout\_alignParentEnd="true"

        android:layout\_above="@+id/textView\_population"

        android:layout\_margin="5dp" />

    <TextView

        android:layout\_width="wrap\_content"

        android:layout\_height="wrap\_content"

android:textAppearance="?android:attr/textAppearanceSmall"

        android:text="Population: ..."

        android:id="@+id/textView\_population"

        android:layout\_alignBottom="@+id/imageView\_flag"

        android:layout\_alignLeft="@+id/textView\_countryName"

        android:layout\_alignStart="@+id/textView\_countryName"

        android:layout\_alignParentRight="true"

        android:layout\_alignParentEnd="true"

        android:layout\_margin="5dp" />

</RelativeLayout>

**Country.java**

?

package org.o7planning.countrieslistview;

package org.o7planning.customlistview;

public class Country {

   private String countryName;

   // Image name (Without extension)

   private String flagName;

   private int population;

   public Country(String countryName, String flagName, int population) {

       this.countryName= countryName;

       this.flagName= flagName;

       this.population= population;

   }

   public int getPopulation() {

       return population;

   }

   public void setPopulation(int population) {

       this.population = population;

   }

   public String getCountryName() {

       return countryName;

   }

   public void setCountryName(String countryName) {

       this.countryName = countryName;

   }

   public String getFlagName() {

       return flagName;

   }

   public void setFlagName(String flagName) {

       this.flagName = flagName;

   }

   @Override

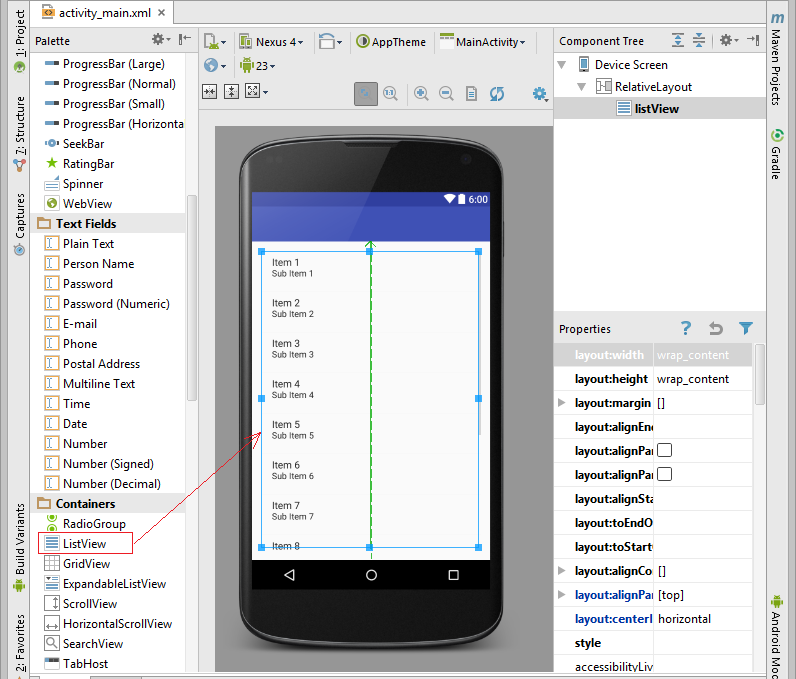
   public String toString()  {

       return this.countryName+" (Population: "+ this.population+")";

   }

}

* **activity\_main.xml**



**activity\_main.xml**

?

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

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

   xmlns:tools="<http://schemas.android.com/tools>"

   android:layout\_width="match\_parent"

   android:layout\_height="match\_parent"

   android:paddingBottom="@dimen/activity\_vertical\_margin"

   android:paddingLeft="@dimen/activity\_horizontal\_margin"

   android:paddingRight="@dimen/activity\_horizontal\_margin"

   android:paddingTop="@dimen/activity\_vertical\_margin"

   tools:context="org.o7planning.customlistview.MainActivity">

   <ListView

       android:layout\_width="wrap\_content"

       android:layout\_height="wrap\_content"

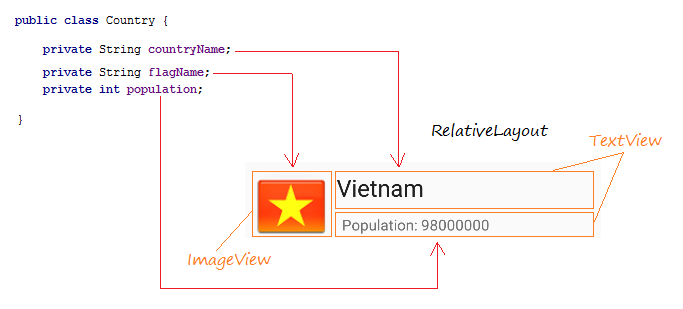
       android:id="@+id/listView"

       android:layout\_alignParentTop="true"

       android:layout\_centerHorizontal="true" />

</RelativeLayout>

**CustomListAdapter** это расширенный класс из  **BaseAdapter**, который выполняет функцию отображения данных на**List Item**.



**CustomListAdapter.java**

?

package org.o7planning.customlistview;

import android.content.Context;

import android.util.Log;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.BaseAdapter;

import android.widget.ImageView;

import android.widget.TextView;

import java.util.List;

public class CustomListAdapter  extends BaseAdapter {

   private List<Country> listData;

   private LayoutInflater layoutInflater;

   private Context context;

   public CustomListAdapter(Context aContext,  List<Country> listData) {

       this.context = aContext;

       this.listData = listData;

       layoutInflater = LayoutInflater.from(aContext);

   }

   @Override

   public int getCount() {

       return listData.size();

   }

   @Override

   public Object getItem(int position) {

       return listData.get(position);

   }

   @Override

   public long getItemId(int position) {

       return position;

   }

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

       ViewHolder holder;

       if (convertView == null) {

convertView = layoutInflater.inflate(R.layout.list\_item\_layout, null);

           holder = new ViewHolder();

holder.flagView = (ImageView) convertView.findViewById(R.id.imageView\_flag);

holder.countryNameView = (TextView) convertView.findViewById(R.id.textView\_countryName);

holder.populationView = (TextView) convertView.findViewById(R.id.textView\_population);

           convertView.setTag(holder);

       } else {

           holder = (ViewHolder) convertView.getTag();

       }

       Country country = this.listData.get(position);

       holder.countryNameView.setText(country.getCountryName());

       holder.populationView.setText("Population: " + country.getPopulation());

       int imageId = this.getMipmapResIdByName(country.getFlagName());

       holder.flagView.setImageResource(imageId);

       return convertView;

   }

   // Find Image ID corresponding to the name of the image (in the directory mipmap).

   public int getMipmapResIdByName(String resName)  {

       String pkgName = context.getPackageName();

       // Return 0 if not found.

       int resID = context.getResources().getIdentifier(resName , "mipmap", pkgName);

       Log.i("CustomListView", "Res Name: "+ resName+"==> Res ID = "+ resID);

       return resID;

   }

   static class ViewHolder {

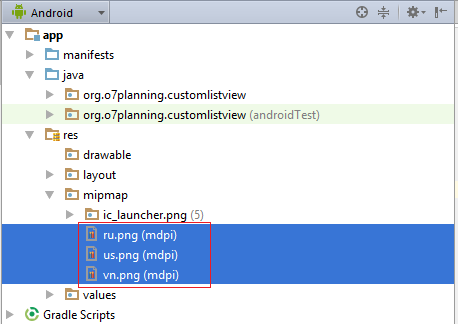
       ImageView flagView;

       TextView countryNameView;

       TextView populationView;

   }

}



**ActivityMain.java**

?

package org.o7planning.customlistview;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ListView;

import android.widget.Toast;

import java.util.ArrayList;

import java.util.List;

public class MainActivity extends AppCompatActivity {

   @Override

   protected void onCreate(Bundle savedInstanceState) {

       super.onCreate(savedInstanceState);

       setContentView(R.layout.activity\_main);

       List<Country> image\_details = getListData();

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

       listView.setAdapter(new CustomListAdapter(this, image\_details));

       // When the user clicks on the ListItem

       listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {

           @Override

           public void onItemClick(AdapterView<?> a, View v, int position, long id) {

               Object o = listView.getItemAtPosition(position);

               Country country = (Country) o;

               Toast.makeText(MainActivity.this, "Selected :" + " " + country, Toast.LENGTH\_LONG).show();

           }

       });

   }

   private  List<Country> getListData() {

       List<Country> list = new ArrayList<Country>();

       Country vietnam = new Country("Vietnam", "vn", 98000000);

       Country usa = new Country("United States", "us", 320000000);

       Country russia = new Country("Russia", "ru", 142000000);

       list.add(vietnam);

       list.add(usa);

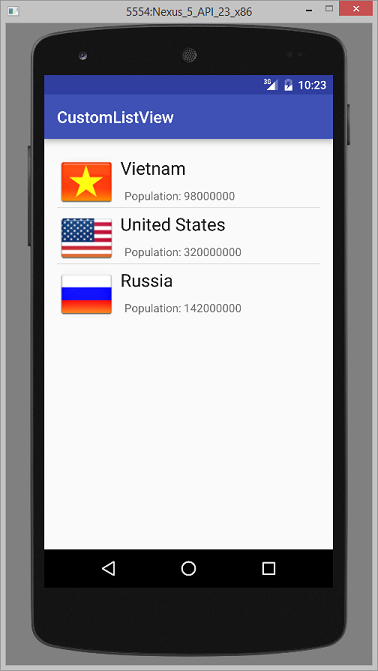
       list.add(russia);

       return list;

   }

}

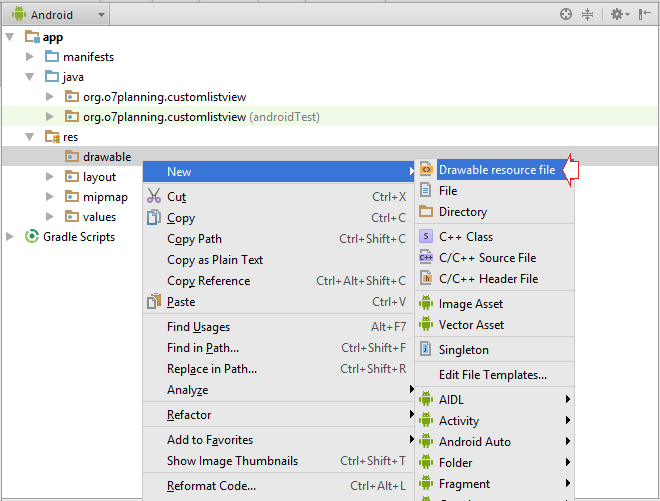
Запуск примера:



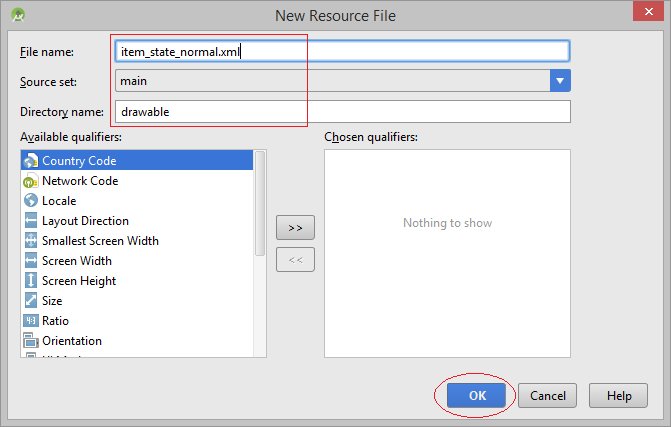
3.2- Пример кастомизации Selector

Чтобы  **ListView** отображался красивее, вам нужно кастомизировать эффекты, например когда меняется цвет фона   **ListItem**при перемещении курсора на нем или менятся цвет фона когда выбран тот **ListItem**. Мы продолжаем с примером выше.

Создать файлы конфигурации:

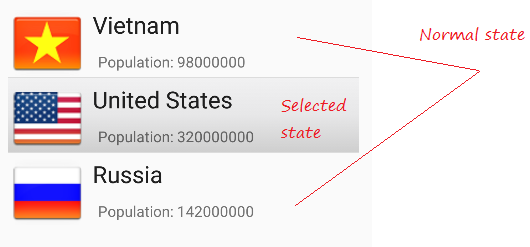


* **File name:** list\_item\_normal.xml
* **Directory:** drawable



Индентично создать 3 других файла:

* item\_state\_pressed.xml
* item\_state\_selected.xml
* list\_selector.xml



Когда  **List Item** в нормальном состоянии стили настроенные в ***item\_state\_normal.xml*** будут применены к  **ListItem**.

**item\_state\_normal.xml**

?

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

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

  android:shape="rectangle">

  <gradient

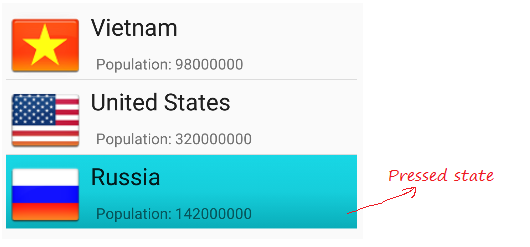
      android:startColor="#f1f1f2"

      android:centerColor="#e7e7e8"

      android:endColor="#cfcfcf"

      android:angle="270" />

</shape>



Когда нажимается  **List Item** стили настроенные в  ***item\_state\_pressed.xml*** будут применены к  **ListItem**

**item\_state\_pressed.xml**

?

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

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

  android:shape="rectangle">

  <gradient

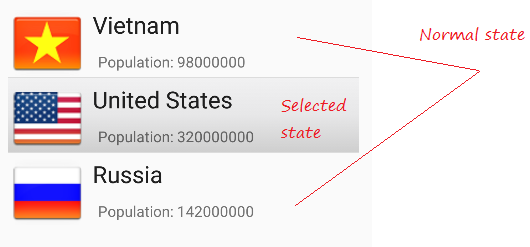
      android:startColor="#18d7e5"

      android:centerColor="#16cedb"

      android:endColor="#09adb9"

      android:angle="270" />

</shape>



Когда нажимается  **List Item** стили настроенные в ***item\_state\_selected.xml*** будут применены к  **ListItem**

**item\_state\_selected.xml**

?

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

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

  android:shape="rectangle">

  <gradient

      android:startColor="#18d7e5"

      android:centerColor="#16cedb"

      android:endColor="#09adb9"

      android:angle="270" />

</shape>

Прикрепить определенные состояния **List Item** с файлами  **xml**.

**list\_selector.xml**

?

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

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

       <item

           android:state\_selected="false"

           android:state\_pressed="false"

           android:drawable="@drawable/item\_state\_normal" />

       <item android:state\_pressed="true"

           android:drawable="@drawable/item\_state\_pressed" />

       <item android:state\_selected="true"

           android:state\_pressed="false"

           android:drawable="@drawable/item\_state\_selected" />

</selector>

Настроить  **ListSelector** для  **ListView**:

**activity\_main.xml**

?

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

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

   xmlns:tools="<http://schemas.android.com/tools>"

   android:layout\_width="match\_parent"

   android:layout\_height="match\_parent"

   android:paddingBottom="@dimen/activity\_vertical\_margin"

   android:paddingLeft="@dimen/activity\_horizontal\_margin"

   android:paddingRight="@dimen/activity\_horizontal\_margin"

   android:paddingTop="@dimen/activity\_vertical\_margin"

   tools:context="org.o7planning.customlistview.MainActivity">

   <ListView

       android:layout\_width="wrap\_content"

       android:layout\_height="wrap\_content"

       android:id="@+id/listView"

       android:layout\_alignParentTop="true"

       android:layout\_centerHorizontal="true"

       android:listSelector="@drawable/list\_selector" />

</RelativeLayout>

Перезапустить ваше приложение.

