



# Освой программирование игр

## Сайт Александра Климова



(<http://developer.alexanderklimov.ru/>)

*/\* Моя кошка замечательно разбирается в программировании. Стоит мне объяснить проблему ей - и все становится ясно. \*/*

*John Robbins, Debugging Applications, Microsoft Press, 2000*



(<http://feeds.feedburner.com/alexanderklimov/VJcl>)

Android (<http://developer.alexanderklimov.ru/android>)

C#/Visual Basic (<http://developer.alexanderklimov.ru/dotnet/>)

Windows Phone (<http://developer.alexanderklimov.ru/windowsphone/wp.php>)

WPF (<http://developer.alexanderklimov.ru/wpf/wpf.php>)

PHP (<http://developer.alexanderklimov.ru/php>)

Arduino (<http://developer.alexanderklimov.ru/arduino>)

Главная (<http://developer.alexanderklimov.ru/android/index.php>)

Теория (<http://developer.alexanderklimov.ru/android/theory/>)

Palette (<http://developer.alexanderklimov.ru/android/views.php>)

ListView (<http://developer.alexanderklimov.ru/android/listview/>)

Котошоп (<http://developer.alexanderklimov.ru/android/catshop/>)

Анимация (<http://developer.alexanderklimov.ru/android/animation/>)

SQLite (<http://developer.alexanderklimov.ru/android/sqlite/>)

OpenGL ES (<http://developer.alexanderklimov.ru/android/opengles/>)

Библиотеки (<http://developer.alexanderklimov.ru/android/library/>)

Игры (<http://developer.alexanderklimov.ru/android/games/>)

Wear (<http://developer.alexanderklimov.ru/android/wear/>)

Эмулятор (<http://developer.alexanderklimov.ru/android/emulator/>)

Советы (<http://developer.alexanderklimov.ru/android/tips-android.php>)

Статьи (<http://developer.alexanderklimov.ru/android/articles-android.php>)

Книги (<http://developer.alexanderklimov.ru/android/books.php>)

Java. Экспресс-курс (<http://developer.alexanderklimov.ru/android/java/java.php>)

Дизайн (<http://developer.alexanderklimov.ru/android/design/>)

Отладка (<http://developer.alexanderklimov.ru/android/debug/>)

Open Source (<http://developer.alexanderklimov.ru/android/opensource.php>)

Полезные ресурсы (<http://developer.alexanderklimov.ru/android/links.php>)

# ExpandableListView со значками

В стандартный **ExpandableListView** легко добавить значки к подгруппам. Достаточно настроить разметку для этой категории.

Подготовим разметку для групп.

## res/layout/exp\_list\_group\_item.xml

```
<CheckedTextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/textViewGroupName"
    android:layout_width="wrap_content"
    android:layout_height="60dp"
    android:layout_marginLeft="4dp"
    android:background="#009999"
    android:gravity="center_vertical"
    android:padding="10dp"
    android:textColor="#FFFFFF"
    android:textSelectHandleLeft="@string/hello_world"
    android:textSize="18sp"
    android:textStyle="bold" />
```

Разметка для дочерних элементов.

## res/layout/exp\_list\_child\_item.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="40dp"
    android:background="#99D6D6"
    android:clickable="true"
    android:orientation="vertical"
    android:paddingLeft="40dp" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="40dp"
        android:gravity="center_vertical" >

        <ImageView
            android:id="@+id/childImage"
            android:layout_width="28dp"
            android:layout_height="28dp"
            android:layout_margin="4dp"
            android:background="@drawable/ic_launcher"
            android:contentDescription="@string/hello_world" />

        <TextView
            android:id="@+id/textViewChild"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="4dp"
            android:textColor="#1919A3"
            android:textSize="16sp"
            android:textStyle="bold" />
    </LinearLayout>

    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="@android:color/white" />

</LinearLayout>

```

Класс для адаптера.

```

package ru.alexanderklimov.test;

import java.util.ArrayList;

import android.app.Activity;
import android.view.LayoutInflater;
import android.view.View;
import android.view.View.OnClickListener;

```

```
import android.view.ViewGroup;
import android.widget.BaseExpandableListAdapter;
import android.widget.CheckedTextView;
import android.widget.TextView;
import android.widget.Toast;

public class MyExpandableAdapter extends BaseExpandableListAdapter {

    private Activity activity;
    private ArrayList<Object> childItems;
    private LayoutInflater inflater;
    private ArrayList<String> parentItems, child;

    // constructor
    public MyExpandableAdapter(ArrayList<String> parents,
                               ArrayList<Object> children) {
        this.parentItems = parents;
        this.childItems = children;
    }

    public void setInflater(LayoutInflater inflater, Activity activity) {
        this.inflater = inflater;
        this.activity = activity;
    }

    @Override
    public int getGroupCount() {
        return parentItems.size();
    }

    @Override
    public int getChildrenCount(int groupPosition) {
        // TODO Auto-generated method stub
        return ((ArrayList<String>) childItems.get(groupPosition)).size();
    }

    @Override
    public Object getGroup(int groupPosition) {
        // TODO Auto-generated method stub
        return null;
    }

    @Override
    public Object getChild(int groupPosition, int childPosition) {
        // TODO Auto-generated method stub
        return null;
    }

    @Override
    public long getGroupId(int groupPosition) {
```

```

        // TODO Auto-generated method stub
        return 0;
    }

    @Override
    public long getChildId(int groupPosition, int childPosition) {
        // TODO Auto-generated method stub
        return 0;
    }

    @Override
    public boolean hasStableIds() {
        // TODO Auto-generated method stub
        return false;
    }

    @Override
    public View getView(int groupPosition, boolean isExpanded,
        View convertView, ViewGroup parent) {
        if (convertView == null) {
            convertView = inflater.inflate(R.layout.exp_list_group_item, null);
        }

        ((CheckedTextView) convertView).setText(parentItems.get(groupPosition));
        ((CheckedTextView) convertView).setChecked(isExpanded);

        return convertView;
    }

    @Override
    public View getChildView(int groupPosition, final int childPosition,
        boolean isLastChild, View convertView, ViewGroup parent) {
        // TODO Auto-generated method stub
        child = (ArrayList<String>) childItems.get(groupPosition);

        TextView textView = null;

        if (convertView == null) {
            convertView = inflater.inflate(R.layout.exp_list_child_item, null);
        }

        // get the textView reference and set the value
        textView = (TextView) convertView.findViewById(R.id.textViewChild);
        textView.setText(child.get(childPosition));

        // set the ClickListener to handle the click event on child item
        convertView.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View view) {

```

```

                Toast.makeText(activity, child.get(childPosition),
                                Toast.LENGTH_SHORT).show();
            }
        });
        return convertView;
    }

    @Override
    public boolean isChildSelectable(int groupPosition, int childPosition) {
        // TODO Auto-generated method stub
        return false;
    }
}

```

Код для активности.

```

package ru.alexanderklimov.test;

import java.util.ArrayList;

import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.widget.ExpandableListView;
import android.widget.ImageView;
import android.app.ExpandableListActivity;
import android.content.Context;

public class MainActivity extends ExpandableListActivity {

    // Create ArrayList to hold parent Items and Child Items
    private ArrayList<String> parentItems = new ArrayList<String>();
    private ArrayList<Object> childItems = new ArrayList<Object>();

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        // Create Expandable List and set it's properties
        ExpandableListView expandableList = getExpandableListView();
        expandableList.setDividerHeight(2);
        expandableList.setGroupIndicator(null);
        expandableList.setClickable(true);

        // Set the Items of Parent
        setGroupParents();
        // Set The Child Data
        setChildData();
    }
}

```

```

        // Create the Adapter
        MyExpandableAdapter adapter = new MyExpandableAdapter(parentItems,
                                                                childItems);

        adapter.setInflater(
            (LayoutInflater) getSystemService(Context.LAYOUT_INFLATER_SERVICE),

            this);

        // Set the Adapter to expandableList
        expandableList.setAdapter(adapter);
        expandableList.setOnChildClickListener(this);
    }

    // method to add parent Items
    public void setGroupParents() {
        parentItems.add("Fruits");
        parentItems.add("Flowers");
        parentItems.add("Animals");
        parentItems.add("Birds");
    }

    // method to set child data of each parent
    public void setChildData() {

        // Add Child Items for Fruits
        ArrayList<String> child = new ArrayList<String>();
        child.add("Apple");
        child.add("Mango");
        child.add("Banana");
        child.add("Orange");

        childItems.add(child);

        // Add Child Items for Flowers
        child = new ArrayList<String>();
        child.add("Rose");
        child.add("Lotus");
        child.add("Jasmine");
        child.add("Lily");

        childItems.add(child);

        // Add Child Items for Animals
        child = new ArrayList<String>();
        child.add("Lion");
        child.add("Tiger");
        child.add("Horse");
        child.add("Котик");
    }

```

```

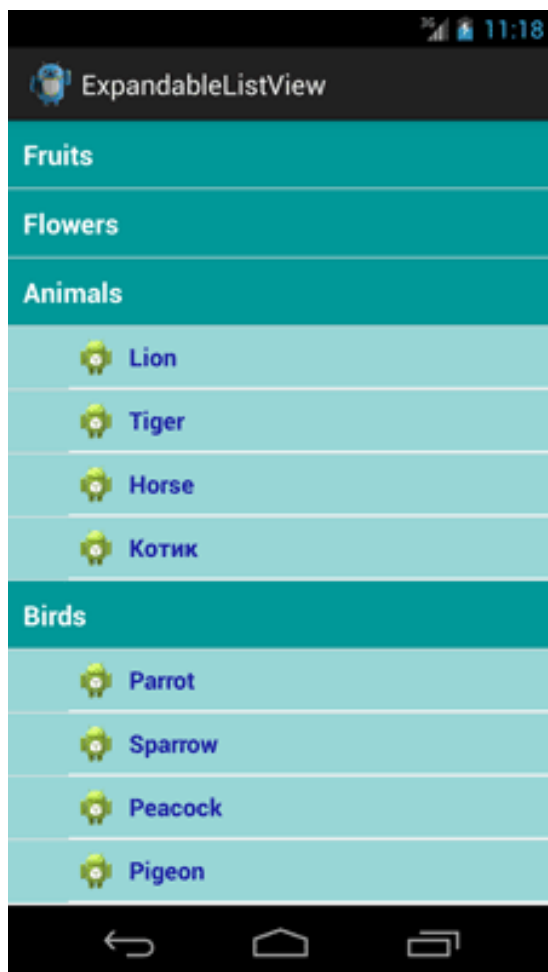
        childItems.add(child);

        // Add Child Items for Birds
        child = new ArrayList<String>();
        child.add("Parrot");
        child.add("Sparrow");
        child.add("Peacock");
        child.add("Pigeon");

        childItems.add(child);
    }
}

```

Результат.




Отсюда (<http://www.learn-android-easily.com/2013/07/android-expandablelistview-example.html>)

## Реклама

Реклама






© 2015 А.Климов (mailto:rusproject@mail.ru)  (<http://plus.google.com/109061106977829925124?prsrc=3>)



(<http://top.mail.ru/jump?from=228158>)



(<http://feeds.feedburner.com/alexanderklimov/VJcl>)

 Рекомендовать в Google