

Тема 10. PopupMenu. Всплывающее меню¹

Начиная с Android 3.0, в системе появилась возможность создавать всплывающее меню, привязанное к элементу View. Меню реализовано в виде модального окна, которое отображается снизу от родителя меню или в другом месте, если места снизу недостаточно.

PopupMenu не нужно путать с контекстным меню. У них разные задачи, хотя поведение схоже.

В новых версиях Android использование всплывающих меню предпочтительнее контекстных, которые можно считать устаревшим интерфейсом.

В Android 4.0 добавили новую функциональность, чтобы работать было проще. В частности, всплывающее меню можно получить из XML-файла, используя метод `inflate(int)`, которому следует передать идентификатор ресурса меню. А до этого приходилось использовать отдельный класс `MenuInflater` с избыточным кодом.

Также появился слушатель `PopupMenu.OnDismissListener` для работы с закрытием меню. Он срабатывает либо, когда пользователь щёлкает на пункте меню и меню закрывается, либо пользователь щёлкает в другом месте экрана, и меню также закрывается.

Создать всплывающее меню очень просто. Мы повторяем шаги по созданию обычного меню. Сначала в ресурсах меню создадим нужный файл:

- `res/menu/popupmenu.xml`

```

<?xml version="1.0" encoding="utf-8"?>
<menu
xmlns:android="http://schemas.android.com/apk/res/android" >

    <group android:id="@+id/menugroup1" >
        <item
            android:id="@+id/menu1"
            android:icon="@drawable/ic_launcher"
            android:title="Popup menu item 1"/>
        <item
            android:id="@+id/menu2"
            android:title="Popup menu item 2"/>
        <item
            android:id="@+id/menu3"
            android:title="Popup menu item 3">
            <menu>
                <item
                    android:id="@+id/submenu"
                    android:title="Подменю"/>
            </menu>
        </item>
    </group>
    <group android:id="@+id/menugroup2" >
        <item
            android:id="@+id/menu4"
            android:checkable="true"
            android:checked="true"
            android:icon="@drawable/ic_launcher"
            android:title="Popup menu item 4"/>
        <item
            android:id="@+id/menu5"
            android:title="Popup menu item 5"
            android:enabled="false"/>
        <item

```

Мы постарались использовать всевозможные варианты использования атрибутов, которые отвечают за вывод флажков, доступность и т.д. Есть и другие атрибуты, о которых можно прочитать в соответствующей документации.

Далее добавим на экран активности текстовую метку, кнопку и ImageView. При щелчке на каждом из этих компонентов мы будем выводить одинаковое всплывающее меню:

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:gravity="center_horizontal"
android:orientation="vertical">
<button
android:id="@+id/button"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Показать меню" />
<TextView
android:id="@+id/textView"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="A TextView" />
<ImageView
android:id="@+id/imageView"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:src="@drawable/ic_launcher_cat" />
</LinearLayout>

Осталось написать код. Обратите внимание на закомментированный код, который работал в Android 3.0. В Android 4.0 этот код можно не использовать.

```

package ru.schoolbreaker.testapplication;

import android.support.v7.app.ActionBarActivity;

import android.os.Bundle;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.widget.Button;

import android.widget.ImageView;

import android.widget.PopupMenu;

import android.widget.TextView;

import android.widget.Toast;

public class MainActivity extends ActionBarActivity {

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        Button button = (Button) findViewById(R.id.button);

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

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

        button.setOnClickListener(viewClickListener);

        textView.setOnClickListener(viewClickListener);

        imageView.setOnClickListener(viewClickListener);}

    View.OnClickListener viewClickListener = new View.OnClickListener() {

        @Override

```



```

public void onClick(View v) {
    showPopupMenu(v);} };

private void showPopupMenu(View v) {
    PopupMenu popupMenu = new PopupMenu(this, v);
    popupMenu.inflate(R.menu.popupmenu); // Для Android 4.0
    // для версии Android 3.0 нужно использовать длинный вариант
    // popupMenu.getMenuInflater().inflate(R.menu.popupmenu,
    // popupMenu.getMenu());
    popupMenu
        .setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {
            @Override
            public boolean onMenuItemClick(MenuItem item) {
                // Toast.makeText(PopupMenuDemoActivity.this,
                // item.toString(), Toast.LENGTH_LONG).show();
                // return true;
                switch (item.getItemId()) {
                    case R.id.menu1:
                        Toast.makeText(getApplicationContext(),
                            "Вы выбрали PopupMenu 1",
                            Toast.LENGTH_SHORT).show();
                        return true;
                    case R.id.menu2:
                        Toast.makeText(getApplicationContext(),
                            "Вы выбрали PopupMenu 2",

```



<code>Toast.LENGTH_SHORT).show();</code>
<code>return true;</code>
<code>case R.id.menu3:</code>
<code>Toast.makeText(getApplicationContext(),</code>
<code>"Вы выбрали PopupMenu 3",</code>
<code>Toast.LENGTH_SHORT).show();</code>
<code>return true;</code>
<code>default:</code>
<code>return false;} } });</code>
<code>popupMenu.setOnDismissListener(new PopupMenu.OnDismissListener() {</code>
<code>@Override</code>
<code>public void onDismiss(PopupMenu menu) {</code>
<code>Toast.makeText(getApplicationContext(), "onDismiss",</code>
<code>Toast.LENGTH_SHORT).show();} } });</code>
<code>popupMenu.show();} }</code>

Вам надо создать новый экземпляр `PopupMenu`, указав контекст активности и компонент, к которому будет привязано это меню. Далее загружаете меню из ресурсов и добавляете методы для обработки щелчков. Для отображения на экране вызывается метод `show()`.

Запустив проект, вы можете щёлкать по любому элементу на форме и увидеть всплывающее меню, рис. 1.



Рис. 1. Всплывающее меню