**Контекстное меню и подменю**

Кроме стандартного [меню](http://developer.alexanderklimov.ru/android/theory/menu.php) в Android используется также контекстное меню, вызываемое при нажатии на объект в течение двух-трёх секунд (событие long-press). Также на некоторых моделях устройств контекстное меню можно вызвать при нажатии трекбола или средней кнопки манипулятора D-pad. В отличие от обычного меню, в контекстном меню не поддерживаются значки и быстрые клавиши. Второе важно отличие - контекстное меню применимо к виду, а меню к активности. Поэтому в приложении может быть одно меню и несколько контекстных меню, например, у каждого элемента TextView.

**Программное создание контекстного меню**

Для создания контекстного меню используется метод обратного вызова **onCreateContextMenu()**. В данный метод можно добавлять пункты меню при помощи методов **add()**. C помощью метода **onContextItemSelected()** можно обрабатывать выбор пункта. Но сначала надо зарегистрировать контекстное меню для нужного объекта, например, для TextView при помоми метода **registerForContextMenu()**.

public static final int IDM\_OPEN = 101;

public static final int IDM\_SAVE = 102;

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

registerForContextMenu(textview);

@Override

public void onCreateContextMenu(ContextMenu menu, View v,

ContextMenuInfo menuInfo)

{

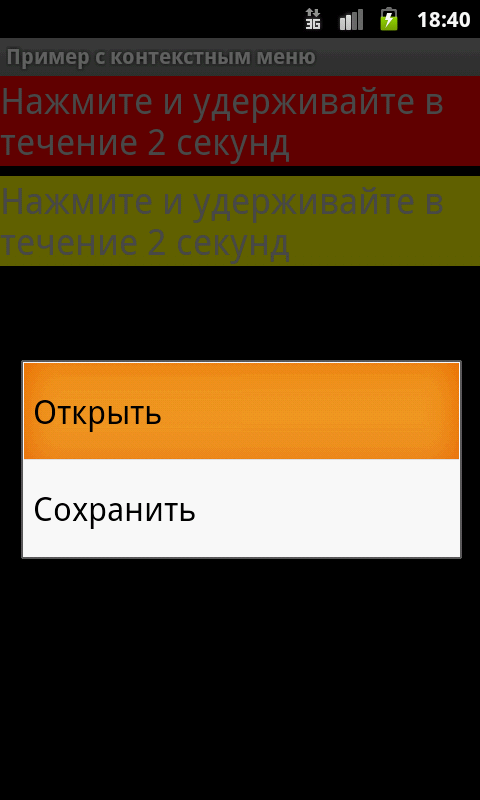
super.onCreateContextMenu(menu, v, menuInfo);

menu.add(Menu.NONE, IDM\_OPEN, Menu.NONE, "Открыть");

menu.add(Menu.NONE, IDM\_SAVE, Menu.NONE, "Сохранить");

}

Если запустить проект, то уже сейчас можно увидеть контекстное меню, которое пока не будет реагировать на выбранные пункты меню. Допишем код.



@Override

public boolean onContextItemSelected(MenuItem item)

{

CharSequence message;

switch (item.getItemId())

{

case IDM\_OPEN:

message = "Выбран пункт Открыть";

break;

case IDM\_SAVE:

message = "Выбран пункт Сохранить";

break;

default:

return super.onContextItemSelected(item);

}

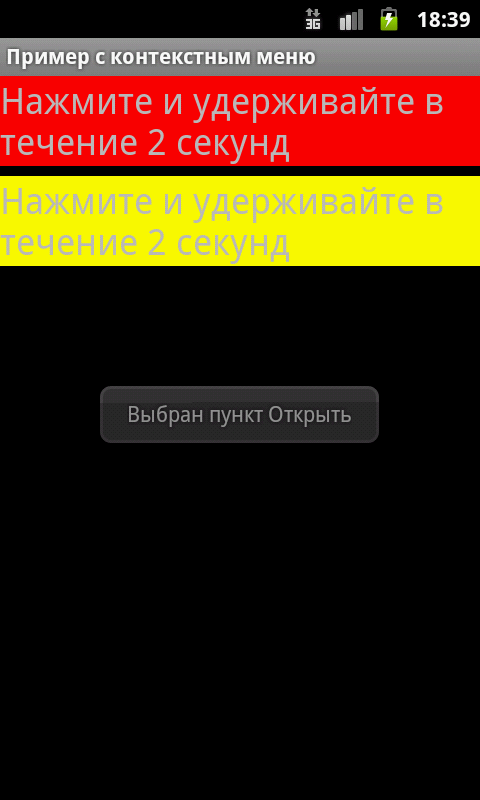
Toast toast = Toast.makeText(this, message, Toast.LENGTH\_SHORT);

toast.setGravity(Gravity.CENTER, 0, 0);

toast.show();

return true;

}



**Контекстное меню через ресурсы**

Если вам нужно создать локализованный продукт, то программное добавление пунктов контекстного меню не подойдет. В таких случаях используется традиционный способ через файл ресурсов, рекомендованный Google.

Создадим XML-файл для меню, а также добавим на форму еще один TextView:

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

<menu

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

<item android:id="@+id/change\_color"

android:title="@string/titleChangeColor" />

<item android:id="@+id/change\_textsize"

android:title="@string/titleChangeTextSize" />

</menu>

registerForContextMenu(textview2);

@Override

public void onCreateContextMenu(ContextMenu menu, View v,

ContextMenuInfo menuInfo)

{

super.onCreateContextMenu(menu, v, menuInfo);

MenuInflater inflater = getMenuInflater();

inflater.inflate(R.menu.context\_menu, menu);

}

@Override

public boolean onContextItemSelected(MenuItem item)

{

CharSequence message;

switch (item.getItemId())

{

case R.id.change\_color:

message = "Меняем цвет";

textview2.setBackgroundColor(Color.RED);

break;

case R.id.change\_textsize:

message = "Меняем размер шрифта";

textview2.setTextSize(18);

break;

default:

return super.onContextItemSelected(item);

}

Toast toast = Toast.makeText(this, message, Toast.LENGTH\_SHORT);

toast.setGravity(Gravity.CENTER, 0, 0);

toast.show();

return true;

}

Мы используем похожий код, как и при создании стандартного меню. В результате получим контекстное меню, созданное из ресурсов. При выборе нужного пункта изменится либо цвет, либо размер шрифта в TextView.

