**Диалоговые окна в Android 4 и 5 ©**

В этой статье мы рассмотрим диалоги следующих типов:

r    AlertDialog — диалог с кнопками;

r    DatePickerDialog — диалог выбора даты;

r    TimePickerDialog — диалог выбора времени.

1. AlertDialog самый частый вариант применения диалога.

AlertDialog — это классический диалог вопроса с кнопками Да и Нет. Вот сейчас и займемся его разработкой. При запуске наше приложение отобразит диалог. При нажатии кнопки Нет не будет произведено никаких действий — будет вызван метод cancel(). А вот действие при нажатии кнопки Да запрограммируете вы сами. Взгляните, как будет выглядеть наш диалог на рис. 1. Код приложения, отображающего этот диалог, представлен в листинге 1.

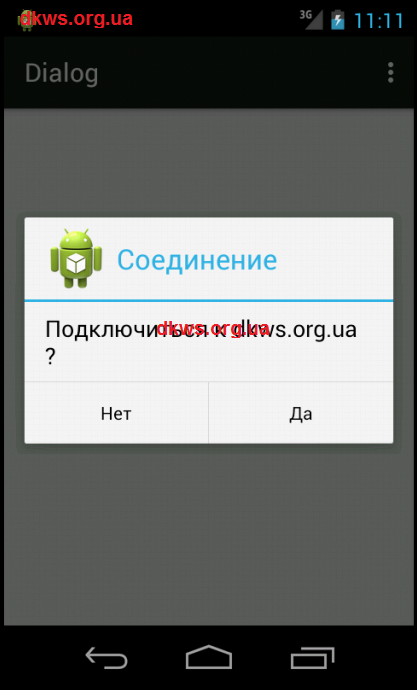


Рис. 1. Диалог типа AlertDialog

Листинг 1. Отображение диалога типа AlertDialog package com.sample.dialog;

import android.support.v7.app.ActionBarActivity;

import android.os.Bundle;

import android.view.Menu;

import android.view.MenuItem;

import android.content.DialogInterface;

import android.app.AlertDialog;

public class MainActivity extends ActionBarActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

AlertDialog.Builder alt\_bld = new AlertDialog.Builder(this);          // Сообщение диалога

alt\_bld.setMessage("Подключиться к dkws.org.ua ?")

.setCancelable(false)

.setPositiveButton("Да", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int id) {            // Действие для кнопки Да

}

})

.setNegativeButton("Нет", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int id) {            //  Действие для кнопки Нет

dialog.cancel();

}

});

AlertDialog alert = alt\_bld.create();

alert.setTitle("Соединение");            // Title for AlertDialog

alert.setIcon(R.drawable.ic\_launcher);  // Icon for AlertDialog

alert.show();

} }

Далее мы рассмотрим диалоги выбора даты и времени.

**2. DatePickerDialog**

Диалоги выбора даты и времени очень "многострадальные". С ними вечно все не так. Мало того, что код вызова такого диалога довольно непростой, так еще и меняется из версии в версию API. Так, для показа диалога ранее использовался метод ShowDialog(), который сейчас использовать не рекомендуется. Вместо него нужно использовать вот такую конструкцию, вовлекающую метод show():

DatePickerDialog dpg =  new DatePickerDialog(this, mDateSetListener, mYear, mMonth, mDay);

dpg.show();

А что, если нужно, чтобы программа была максимально совместима со всеми версиями? Тогда нужно использовать ShowDialog() и в то же время обеспечить, чтобы среда разрешила компиляцию. Для этого нужно добавить директиву @SuppressWarnings("deprecation").

Да, решение не очень хорошее, но зато работает от API 8 (2.x) до API 21 (5.0). Далее я старался максимально упростить код. Сравните приведенный ниже код с кодом (и решением) с руководства разработчика: http://developer.android.com/guide/topics/ui/controls/pickers.html Само приложение тоже очень простое. Оно содержит только надпись с призывом выбрать дату (рис. 2). При щелчке по надписи открывается диалог выбора даты (рис. 3), а после выбора даты и нажатия кнопки Done выбранная пользователем дата отображается в надписи (рис. 4).

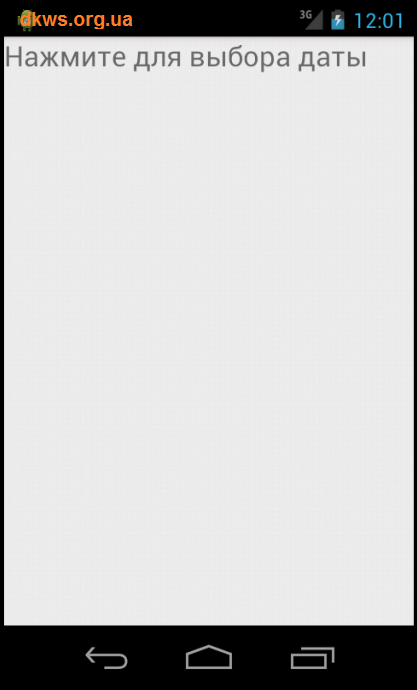
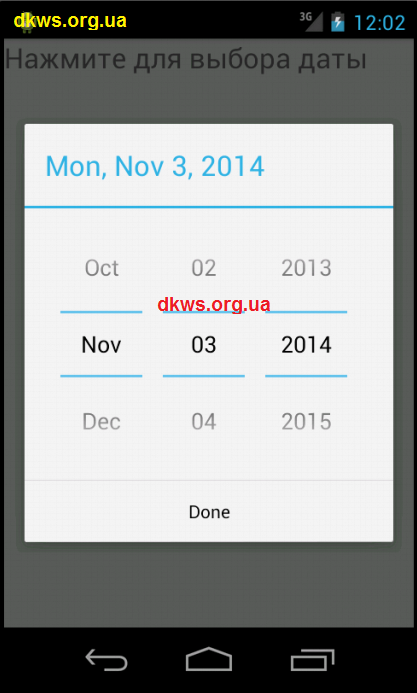
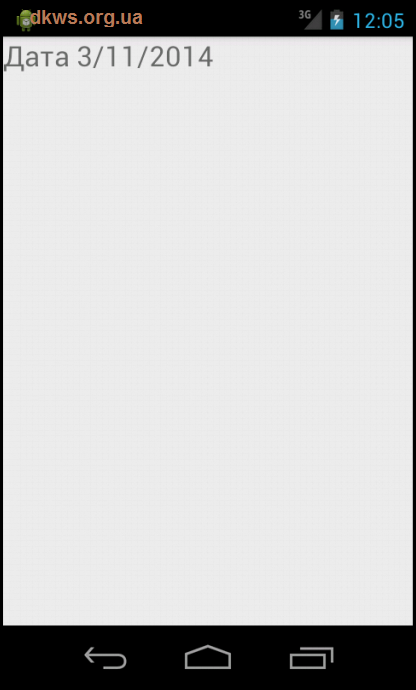
  

Рис. 2. Щелкните по надписи для Рис. 3. Диалог выбора даты Рис. 4. Выбранная дата

выбора даты отображена в текстовой

надписи

Разметка приложения приведена в листинге 2, а Java-код приложения — в листинге 3.

Листинг 2. Разметка приложения Листинг 3. Выбор даты

package com.sample.selectdatedlg;

import android.app.Activity;

import android.app.DatePickerDialog;

import android.app.DatePickerDialog.OnDateSetListener;

import android.app.Dialog;

import android.os.Bundle;

import android.view.View;

import android.widget.DatePicker;

import android.widget.TextView;

public class MainActivity extends Activity {

int DIALOG\_DATE = 1;   // Дата для инициализации (можно получить через Calendar текущую, но

// я старался сделать код как можно проще).

int myYear = 2014;

int myMonth = 10;   // ноябрь, нумерация месяцев начинается с 0!

int myDay = 03;

TextView tvDate;

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

tvDate = (TextView) findViewById(R.id.tvDate);     }

// Добавил директиву, чтобы среда не "ругалась" на старый код

@SuppressWarnings("deprecation")

public void onclick(View view) {       showDialog(DIALOG\_DATE);     }

@SuppressWarnings("deprecation")

protected Dialog onCreateDialog(int id) {

if (id == DIALOG\_DATE) {

DatePickerDialog tpd = new DatePickerDialog(this, myCallBack, myYear, myMonth, myDay);

return tpd;       }

return super.onCreateDialog(id);     }

OnDateSetListener myCallBack = new OnDateSetListener() {

public void onDateSet(DatePicker view, int year, int monthOfYear,         int dayOfMonth) {

myYear = year;       // увеличиваем полученный месяц для отображения

myMonth = monthOfYear + 1;

myDay = dayOfMonth;

tvDate.setText("Дата " + myDay + "/" + myMonth + "/" + myYear);

}     }; }

**3. TimePickerDialog**

Сейчас напишем приложение выбора времени, но оно будет немного отличаться от предыдущего (dkws.org.ua). Вместо надписи у нас будет кнопка, вызывающая диалог выбора времени (рис. 5), а выбранное время будет отображено в качестве уведомления (рис. 6). Разметка приложения приведена в листинге 4а, а полный код приложения - в листинге 4б. Листинг 4а. Разметка приложения        Листинг 4б. Выбор времени (код приложения)

package com.sample.timeselect;

import android.app.Activity;

import android.os.Bundle;

import android.app.Dialog;

import android.app.TimePickerDialog;

import android.widget.Button;

import android.widget.TimePicker;

import android.widget.Toast;

import android.view.View;

public class MainActivity extends Activity {         private Button b1;

        static final int TIME\_DIALOG\_ID = 0;

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

                setContentView(R.layout.activity\_main);

                // Кнопка

b1 = (Button) findViewById(R.id.button1);

                b1.setOnClickListener(new View.OnClickListener() {

// Обработчик нажатия кнопки — вызываем диалог

@SuppressWarnings("deprecation")

public void onClick(View v) {

showDialog(TIME\_DIALOG\_ID);

                        }          });

         }

private TimePickerDialog.OnTimeSetListener mTimeSetListener =          new TimePickerDialog.OnTimeSetListener() {

 // Обработчик нажатия кнопки Set диалога выбора времени (dkws.org.ua)

public void onTimeSet(TimePicker view, int hourOfDay,   int minute) {

// Отображаем уведомление с выбранным пользователем временем

Toast.makeText(MainActivity.this,   "Timeis="+hourOfDay+":"+minute,    Toast.LENGTH\_LONG).show();

         }          };

         @Override

 protected Dialog onCreateDialog(int id) {

switch (id) {

case TIME\_DIALOG\_ID: return new TimePickerDialog(this, mTimeSetListener, 0, 0, false);

                 }          return null;

         } }

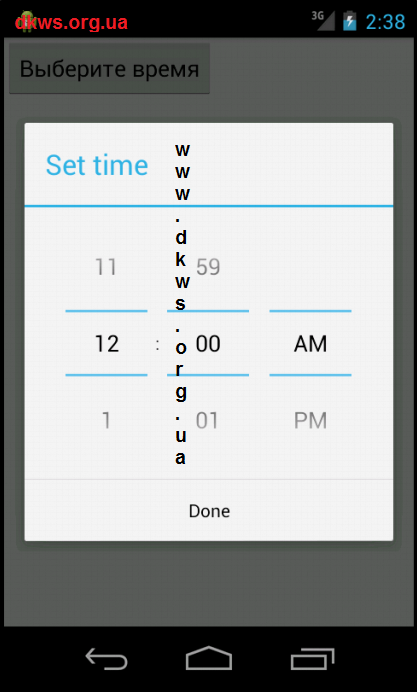


Рис. 5. Диалог выбора времени

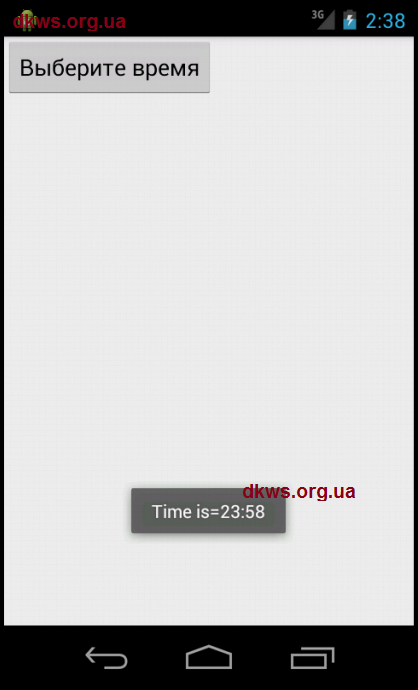


Рис. 6. Выбранное пользователем время