DatePicker

Элемент **DatePicker** позволяет пользователю выбрать дату (число, месяц, год). Данный виджет используют как правило на отдельном экране. Также существует похожее по функционалу диалоговое окно **DatePickerDialog**.

**DatePicker** находится в разделе **Date & Time**

Нужно помнить, что внутренняя нумерация месяцев в Android начинается с 0, поэтому январь = 0, февраль = 1 и т.д. Поэтому как правило, в коде следует прибавлять единицу к значению месяца при работе с датой или наоборот - отнимать единицу, если данные поступают снаружи (например, из текстового поля).

Добавим на экран текстовую метку, кнопку и виджет даты:

<?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="ru.alexanderklimov.datepicker.MainActivity">

<TextView

android:id="@+id/textView3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentTop="true"

android:layout\_centerHorizontal="true"

android:text="Large Text"

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

<Button

android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/textView3"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="47dp"

android:text="New Button"/>

<DatePicker

android:id="@+id/datePicker"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/button"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="48dp"/>

</RelativeLayout>

Пользователь может выбрать любую дату на виджете, а обычный щелчок на кнопке позволит отобразить выбранную дату в текстовой метке. Длительное нажатие на кнопку устанавливает текущую дату на виджете и в текстовой метке.

Настройка даты на самом виджете происходит через метод **init()**. Изменение даты можно прослушивать через **OnDateChangedListener()**:

package ru.alexanderklimov.datepicker;

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.Button;

import android.widget.DatePicker;

import android.widget.TextView;

import android.widget.Toast;

import java.util.Calendar;

public class MainActivity extends AppCompatActivity {

private DatePicker mDatePicker;

private TextView mInfoTextView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

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

mDatePicker = (DatePicker) findViewById(R.id.datePicker);

Calendar today = Calendar.getInstance();

mDatePicker.init(today.get(Calendar.YEAR), today.get(Calendar.MONTH),

today.get(Calendar.DAY\_OF\_MONTH), new DatePicker.OnDateChangedListener() {

@Override

public void onDateChanged(DatePicker view, int year,

int monthOfYear, int dayOfMonth) {

Toast.makeText(getApplicationContext(),

"onDateChanged", Toast.LENGTH\_SHORT).show();

mInfoTextView.setText("Год: " + year + "\n" + "Месяц: "

+ (monthOfYear + 1) + "\n" + "День: " + dayOfMonth);

}

});

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

changingDateButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

mInfoTextView.setText(new StringBuilder()

// Месяц отсчитывается с 0, поэтому добавляем 1

.append(mDatePicker.getDayOfMonth()).append(".")

.append(mDatePicker.getMonth() + 1).append(".")

.append(mDatePicker.getYear()));

}

});

changingDateButton.setOnLongClickListener(new View.OnLongClickListener() {

@Override

public boolean onLongClick(View v) {

setCurrentDateOnView();

return true;

}

});

}

// устанавливаем текущую дату

public void setCurrentDateOnView() {

final Calendar calendar = Calendar.getInstance();

int year = calendar.get(Calendar.YEAR);

int month = calendar.get(Calendar.MONTH);

int day = calendar.get(Calendar.DAY\_OF\_MONTH);

// set current date into textview

mInfoTextView.setText(new StringBuilder()

// Month is 0 based, just add 1

.append(day).append(".").append(month + 1).append(".")

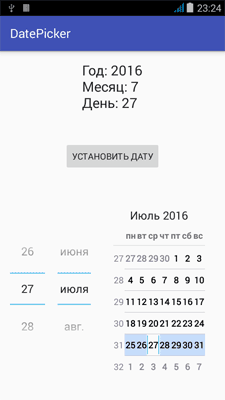
.append(year));

// Устанавливаем текущую дату для DatePicker

mDatePicker.init(year, month, day, null);

}

}



Настройка компонента

По умолчанию **DatePicker** состоит из двух элементов - **Spinner** и **CalendarView**. Вы можете настроить внешний вид компонента, если укажете через XML-атрибуты**android:calendarViewShown** и **android:spinnersShown**, какой элемент не нужно выводить. Также можно это сделать программно через методы **setCalendarViewShown()** и**setSpinnersShown()** (API level 11).

Напишем пример программного управления внешним видом **DatePicker**:

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

<LinearLayout 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:orientation="vertical"

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

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

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

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

<RadioGroup

android:id="@+id/optGroup"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<RadioButton

android:id="@+id/optCalendar"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:checked="true"

android:text="CalendarView"/>

<RadioButton

android:id="@+id/optSpinners"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Spinners"/>

<RadioButton

android:id="@+id/optBoth"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Both"/>

</RadioGroup>

<DatePicker

android:id="@+id/datePicker"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:calendarViewShown="true"

android:spinnersShown="false"/>

</LinearLayout>

Код для активности:

package ru.alexanderklimov.datepicker;

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.widget.DatePicker;

import android.widget.RadioButton;

import android.widget.RadioGroup;

public class MainActivity extends AppCompatActivity {

private DatePicker mDatePicker;

private RadioButton mCalendarRadionButton;

private RadioButton mSpinnerRadioButton;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

mDatePicker = (DatePicker) findViewById(R.id.datePicker);

RadioGroup radioGroup = (RadioGroup) findViewById(R.id.optGroup);

mCalendarRadionButton = (RadioButton) findViewById(R.id.optCalendar);

mSpinnerRadioButton = (RadioButton) findViewById(R.id.optSpinners);

mDatePicker = (DatePicker) findViewById(R.id.datePicker);

radioGroup.setOnCheckedChangeListener(new RadioGroup.OnCheckedChangeListener() {

@Override

public void onCheckedChanged(RadioGroup group, int checkedId) {

if (mCalendarRadionButton.isChecked()) {

mDatePicker.setCalendarViewShown(true);

mDatePicker.setSpinnersShown(false);

} else if (mSpinnerRadioButton.isChecked()) {

mDatePicker.setCalendarViewShown(false);

mDatePicker.setSpinnersShown(true);

} else {

mDatePicker.setCalendarViewShown(true);

mDatePicker.setSpinnersShown(true);

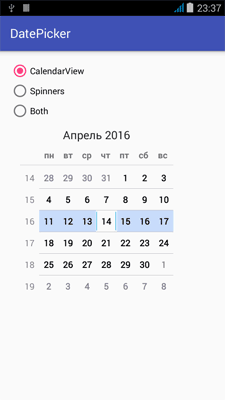
}

}

});

}

}



Новый внешний вид в Android 5.0 (API 21) Lollipop

В Android 5.0 (API 21) внешний вид компонента изменился.



Скриншот взят из статьи [Криминальное чтиво](http://developer.alexanderklimov.ru/android/criminal.php).