**Пример использования виджета ImageView в Android**

[13/03/2016](http://prologistic.com.ua/primer-ispol-zovaniya-vidzheta-imageview-v-android.html)  [prologistic](http://prologistic.com.ua/author/prologistic)  [Android](http://prologistic.com.ua/category/android)

Виджет android.widget.ImageView используется в Android для отображения изображения. Именно этим виджетом мы сегодня и займемся: рассмотрим пример использования и основные возможности.

Также в Android есть виджет ImageButton, о котором можно почитать [здесь](http://prologistic.com.ua/primer-ispol-zovaniya-imagebutton-v-android.html).

**Пример использования виджета ImageView в Android**

ImageView является одним из виджетов пользовательского интерфейса, который используется для просмотра изображений. Чтобы использовать его в своем приложении, мы должны добавить одиночный тег <ImageView ... /> в XML макет:

Java

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | <?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:orientation="vertical">        <ImageView          android:id="@+id/imageView"          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_gravity="center\_horizontal"          android:src="@drawable/fbcom" />  </LinearLayout> |

Как видно из макета, мы использовали [LinearLayout](http://prologistic.com.ua/android-layouts-obzor-relativelayout-i-linearlayout.html" \t "_blank) в качестве корневого элемента компоновки (вы можете брать любой другой тип).

Обратите внимание на атрибут android:src="@drawable/fbcom". Здесь мы указываем, что виджет ImageView должен отображать картинку, которая находится в папке res/drawable под именем **fbcom** (расширение файла здесь не указывается). Также с помощью атрибута android:layout\_gravity мы указали, что виджет должен быть отцентрирован по горизонтали.

Вставьте любое небольшое изображение в папку res/drawable вашего проекта и замените значение атрибута **src** на @drawable/названиеВашегоИзображения.

Вот как это выглядит у меня в Android:

[](http://prologistic.com.ua/wp-content/uploads/2016/03/result-1.jpg)

Также изображение может задаваться и в коде программы:

Например, на примере ниже мы устанавливаем изображение в ImageView не в макете, а коде программы:

Java

|  |  |
| --- | --- |
| 1  2  3  4 | // инициализируем виджет  ImageView imageView = (ImageView) findViewById(R.id.imageView);  // устанавливаем изображение из папки res/drawable/ в виджет  imageView.setImageResource(R.drawable.fbcom); |

**ScaleType виджета ImageView в Android**

Виджет ImageView предлагает различные варианты конфигурации для поддержки различныхscaleTypes (способов масштабирования). Опция **ScaleType** используются для масштабирования изображения в пределах View. Давайте рассмотри наиболее полезные варианты scaleType:

* CENTER отображает изображение по центру View без какого-либо масштабирования.
* CENTER\_CROP центрирует и масштабирует изображения из сохранением соотношения сторон
* CENTER\_INSIDE — масштабирование изображения внутриView из сохранением соотношения сторон. Если изначально изображение меньше, чем View, то свойство делает то же самое, что и CENTER.
* FIT\_CENTER — масштабирование изображения внутриView из сохранением соотношения сторон. По крайней мере одна ось будет точно совпадать с View, а изображение будет выравнено по центру View.
* FIT\_START — то же, что и FIT\_CENTER, но выравнивается по левому верхнему углу View.
* FIT\_END — то же, что и FIT\_CENTER, но выравнивается по правому нижнему углу View.
* FIT\_XY — так масштабирует размеры изображения, чтобы оно соответствовать размеру View не сохраняя пропорции.
* MATRIX — масштабирование изображения с помощью дополнительного класса Matrix. Матрица может быть поставлена с помощью метода setImageMatrix. Класс Matrixможет быть использован, например, для вращения изображений.

**Работа с растровыми изображениями (Bitmaps)**

Работа с растровыми изображениями относится к классу android.graphics.Bitmap.

Чаще всего для масштабирования изображений с Drawable используется BitmapFactory. Если не использовать BitmapFactory, то это приводит к неэффективной работе с памятью.

Java

|  |  |
| --- | --- |
| 1  2  3  4  5 | ImageView imageView = (ImageView) findViewById(R.id.imageView);  // загружаем изображение из sd-карты в объект Bitmap  Bitmap bitMap = BitmapFactory.decodeFile("/sdcard/testImage.png");  // и устанавливаем его в виджет ImageView  imageView.setImageBitmap(bitMap); |

**ImageView для любознательных**

Взглянем еще раз на наш макет. Системная утилита Android Lint будет сообщать нам свое замечание:

**[Accessibility] missing contentDescription attribute on image**. Это означает, что нам желательно добавить недостающий атрибут contentDescription к разметке нашего виджетаImageView. На работу самого виджета это никак не повлияет и без этого атрибута все будет прекрасно работать, но давайте разберемся с этим замечанием:

Большинство людей никогда не увидят то, что вы впишете в значение атрибута, тем не менее он может быть полезен для людей с ограниченными возможностями (у которых проблемы со зрением). Этим атрибутом вы поможете им узнать где находится нужная им кнопка. Для этого просто добавьте следующий код в разметку виджета ImageView:

Java

|  |  |
| --- | --- |
| 1 | android:contentDescription="Кнопка Like фейсбука" |

В вашем случае описание измените под свою картинку.

Вот такая небольшая статья о виджете ImageView в Android. Следите за обновлениями и подписывайтесь на новые статьи!