

# Gravity

Последнее обновление:



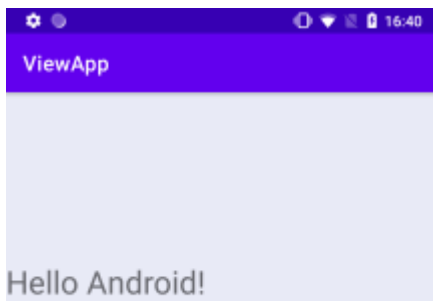
Атрибут **gravity** задает позиционирование содержимого внутри визуального элемента. Он может принимать следующие значения:

- `top`: элементы размещаются вверху
- `bottom`: элементы размещаются внизу
- `left`: элементы размещаются в левой стороне
- `right`: элементы размещаются в правой стороне контейнера
- `center_vertical`: выравнивает элементы по центру по вертикали
- `center_horizontal`: выравнивает элементы по центру по горизонтали
- `center`: элементы размещаются по центру
- `fill_vertical`: элемент растягивается по вертикали
- `fill_horizontal`: элемент растягивается по горизонтали
- `fill`: элемент заполняет все пространство контейнера

- `clip_vertical`: обрезает верхнюю и нижнюю границу элементов
- `clip_horizontal`: обрезает правую и левую границу элементов
- `start`: элемент позиционируется в начале (в верхнем левом углу) контейнера
- `end`: элемент позиционируется в конце контейнера(в верхнем правом углу)

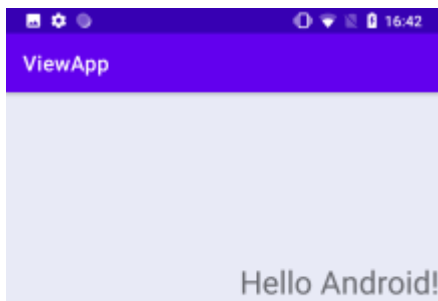
Например, поместим текст в самый низ в элементе `TextView`:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     tools:context=".MainActivity">
9
10    <TextView
11        android:gravity="bottom"
12
13        android:layout_width="0dp"
14        android:layout_height="200dp"
15        android:text="Hello Android!"
16        android:textSize="30sp"
17        android:background="#e8eaf6"
18        app:layout_constraintLeft_toLeftOf="parent"
19        app:layout_constraintRight_toRightOf="parent"
20        app:layout_constraintTop_toTopOf="parent" />
21
22 </androidx.constraintlayout.widget.ConstraintLayout>
```



При необходимости мы можем комбинировать значения, разделяя их вертикальной чертой:

```
1 <TextView
2     android:gravity="bottom|right"
3
4     android:layout_width="0dp"
5     android:layout_height="200dp"
6     android:text="Hello Android!"
7     android:textSize="30sp"
8     android:background="#e8eaf6"
9     app:layout_constraintLeft_toLeftOf="parent"
10    app:layout_constraintRight_toRightOf="parent"
11    app:layout_constraintTop_toTopOf="parent" />
```



## Программная установка gravity

Чтобы установить параметр `gravity` у элемента надо вызвать метод **`setGravity()`**. В качестве параметра в метод передается одна из констант класса **`Gravity`**, которые аналогичны значениям атрибута (за тем исключением, что названия в верхнем регистре):

```
1 package com.example.viewapp;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import androidx.constraintlayout.widget.ConstraintLayout;
5
6 import android.os.Bundle;
7 import android.view.Gravity;
8 import android.widget.TextView;
9
10 public class MainActivity extends AppCompatActivity {
11
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15
16         ConstraintLayout constraintLayout = new ConstraintLayout(this);
17         TextView textView = new TextView(this);
18         textView.setText("Hello Android!");
19         textView.setTextSize(30);
20         textView.setBackgroundColor(0xffe8eaf6);
21
22         // установка gravity
23         textView.setGravity(Gravity.CENTER);
```

```

24
25 // установка высоты и ширины
26 ConstraintLayout.LayoutParams layoutParams = new ConstraintLayout.La
27     (ConstraintLayout.LayoutParams.MATCH_CONSTRAINT, 200);
28 layoutParams.leftToLeft = ConstraintLayout.LayoutParams.PARENT_ID;
29 layoutParams.rightToRight = ConstraintLayout.LayoutParams.PARENT_ID;
30 layoutParams.topToTop = ConstraintLayout.LayoutParams.PARENT_ID;
31 layoutParams.bottomToBottom = ConstraintLayout.LayoutParams.PARENT_
32 textView.setLayoutParams(layoutParams);
33
34 constraintLayout.addView(textView);
35 setContentView(constraintLayout);
36 }
37 }

```



Hello Android!



Для сочетания нескольких значений также можно использовать вертикальную черту:

```

1 textView.setGravity(Gravity.BOTTOM | Gravity.CENTER);

```