

# Mobile Software Development for Android - 1397

IT COLLEGE, ANDRES KÄVER, 2015-2016

EMAIL: [AKAVER@ITCOLLEGE.EE](mailto:AKAVER@ITCOLLEGE.EE)

WEB: [HTTP://ENOS.ITCOLLEGE.EE/~AKAVER/2015-2016/DISTANCE/ANDROID](http://enos.itcollege.ee/~akaver/2015-2016/distance/android)

SKYPE: AKAVER



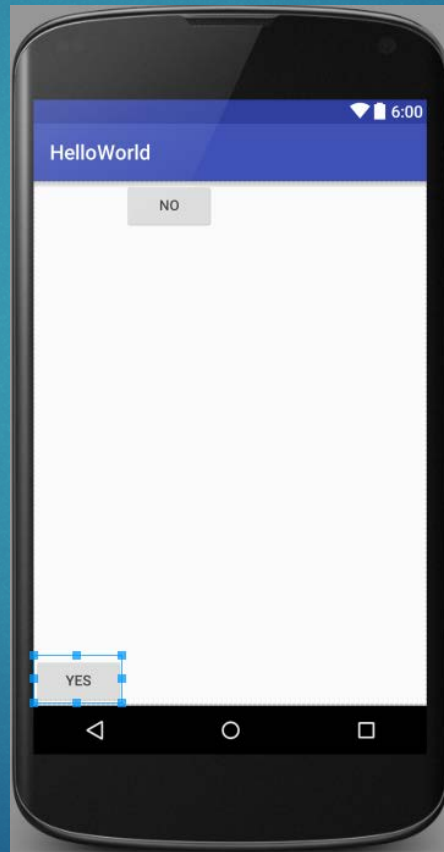
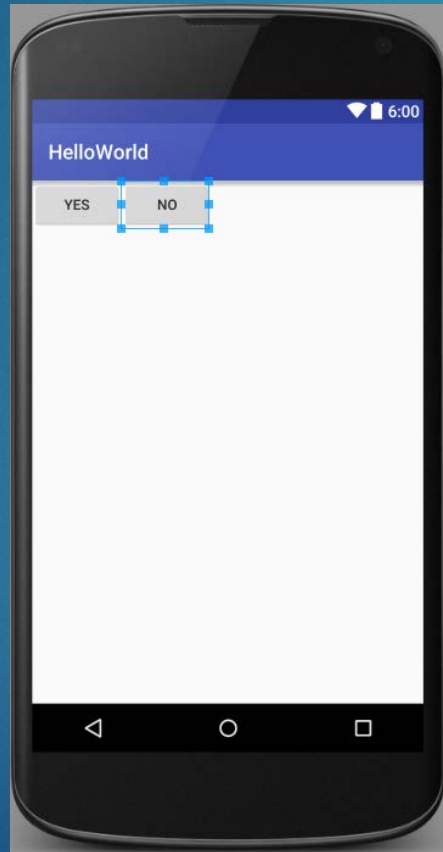
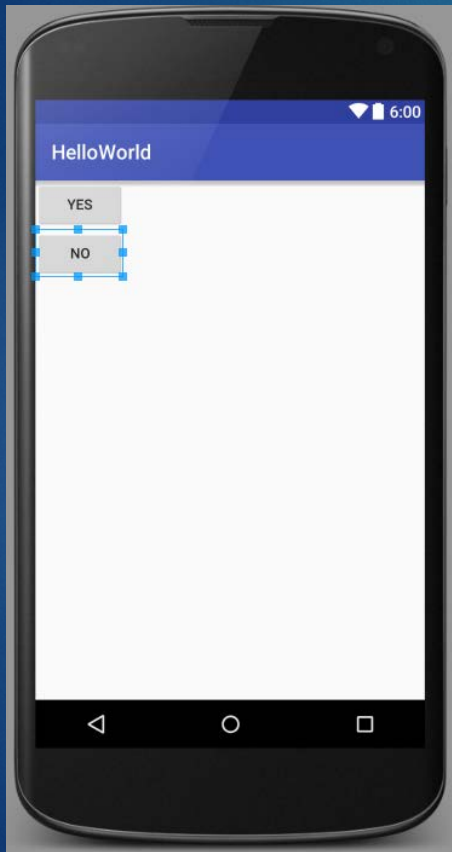
# Layout fundamentals

- ▶ Basic layout
  - ▶ Linear
  - ▶ Relative
  - ▶ Frame
  - ▶ Table
- ▶ Basic attributes
  - ▶ Size
  - ▶ Margin vs padding
  - ▶ Gravity

# Layout

- ▶ Defines declaratively visual structure for app
- ▶ Takes into consideration screen properties
  - ▶ size
  - ▶ pixel density
- ▶ System calculates sizes and position for all UI elements

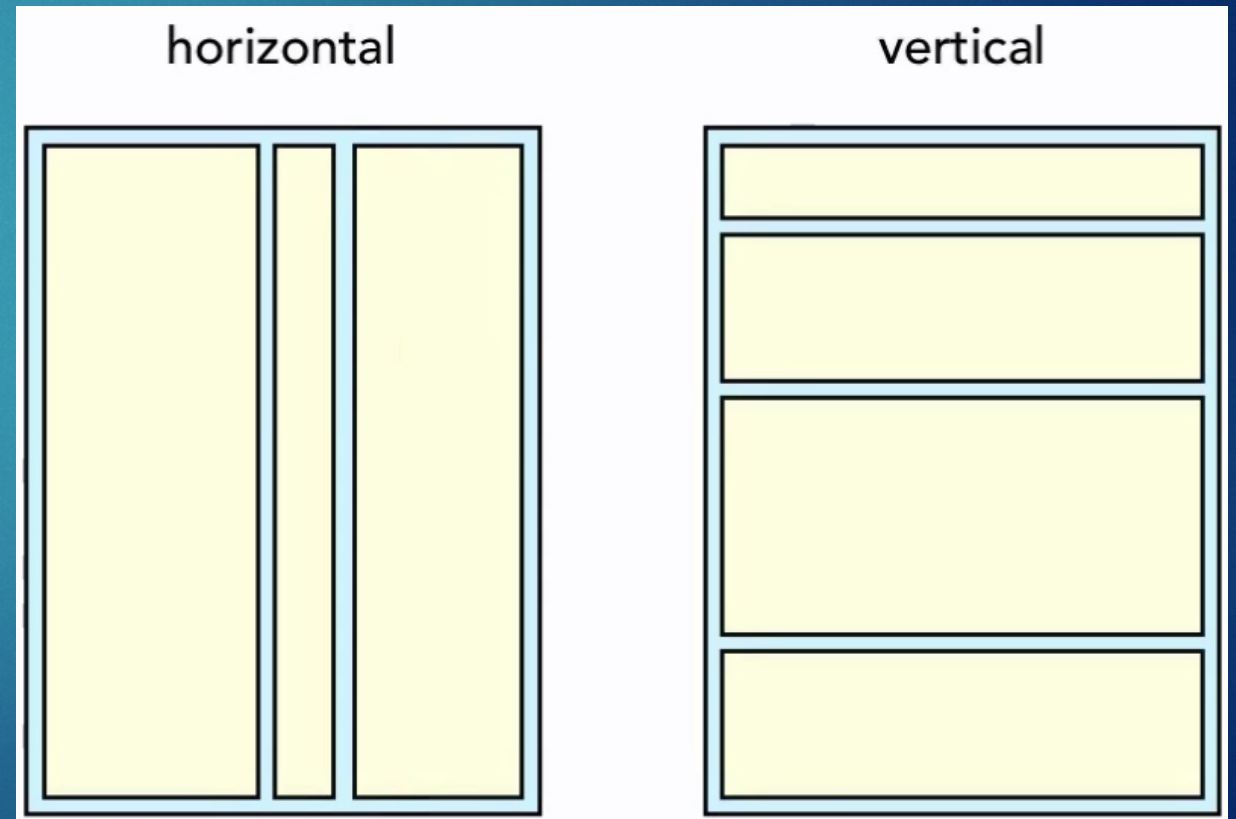
# Layout



# Layout - LinearLayout

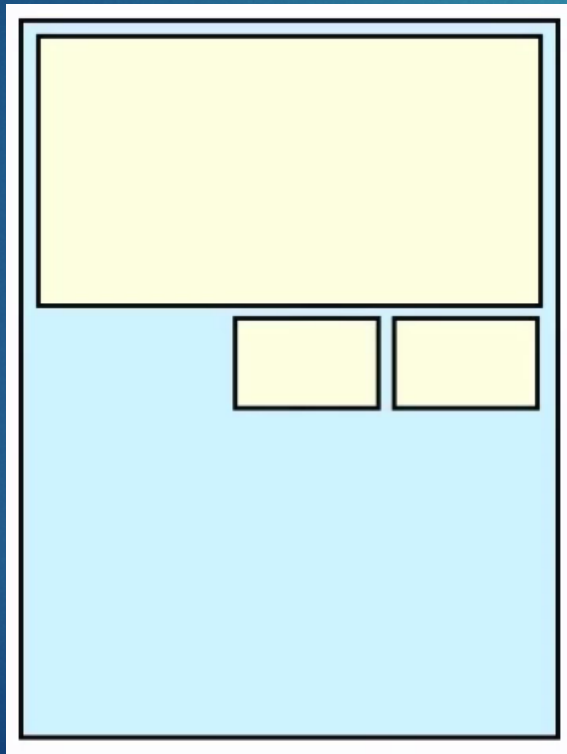
- ▶ Arranges its children in single direction
  - ▶ Orientation="horizontal"
  - ▶ Orientation="vertical"

```
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    android:orientation="horizontal"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent">  
  
    ...ui elements...  
</LinearLayout>
```



# Layout - RelativeLayout

- ▶ Elements are arranged relative to
  - ▶ Parent UI element
  - ▶ Sibling UI element



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent" android:layout_height="match_parent">

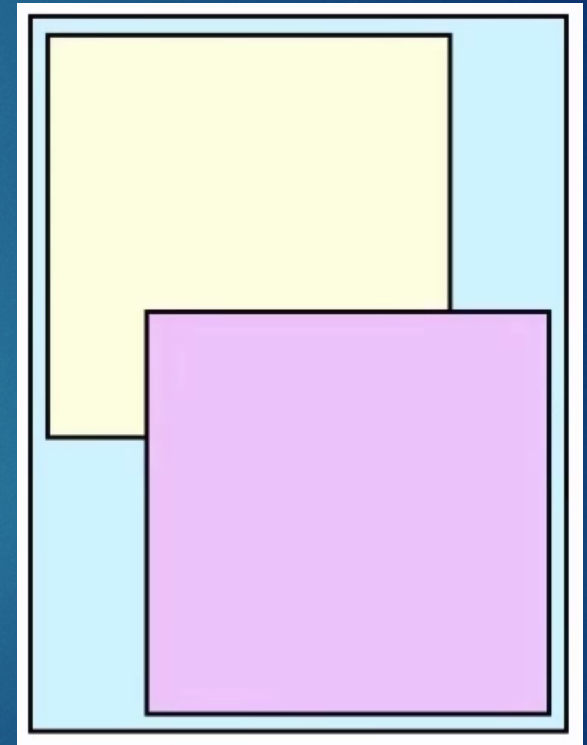
    <EditText
        android:layout_width="wrap_content"
        android:layout_height="200px"
        android:inputType="textMultiLine"
        android:ems="10"
        android:id="@+id/editText2"
        android:layout_alignParentTop="true"
        android:layout_alignParentStart="true"
        android:layout_alignParentEnd="true" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="New Button"
        android:id="@+id/button4"
        android:layout_below="@+id/editText2"
        android:layout_alignParentEnd="true" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="New Button"
        android:id="@+id/button5"
        android:layout_alignBottom="@+id/button4"
        android:layout_toStartOf="@+id/button4" />
</RelativeLayout>
```

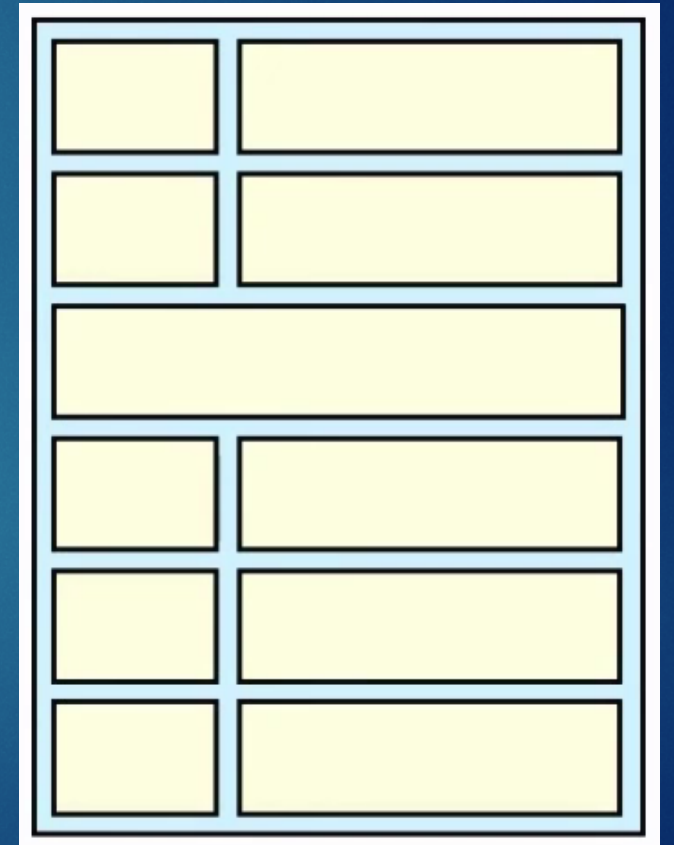
# Layout - FrameLayout

- ▶ Allows to put child views on top of each other
- ▶ Specify order
- ▶ Layout gravity for positioning



# Layout - TableLayout

- ▶ Rows and Columns
- ▶ Like Linear vertical nesting Linear horizontal
- ▶ Columns are aligned (by the widest in all the rows)





# Basic attributes

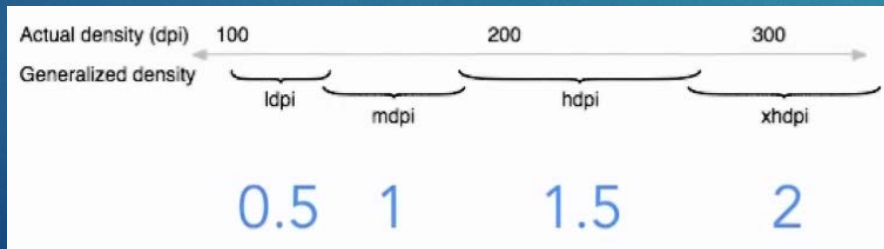
- ▶ Used across all layouts
  - ▶ Size
  - ▶ Margin vs padding
  - ▶ Gravity

# Size

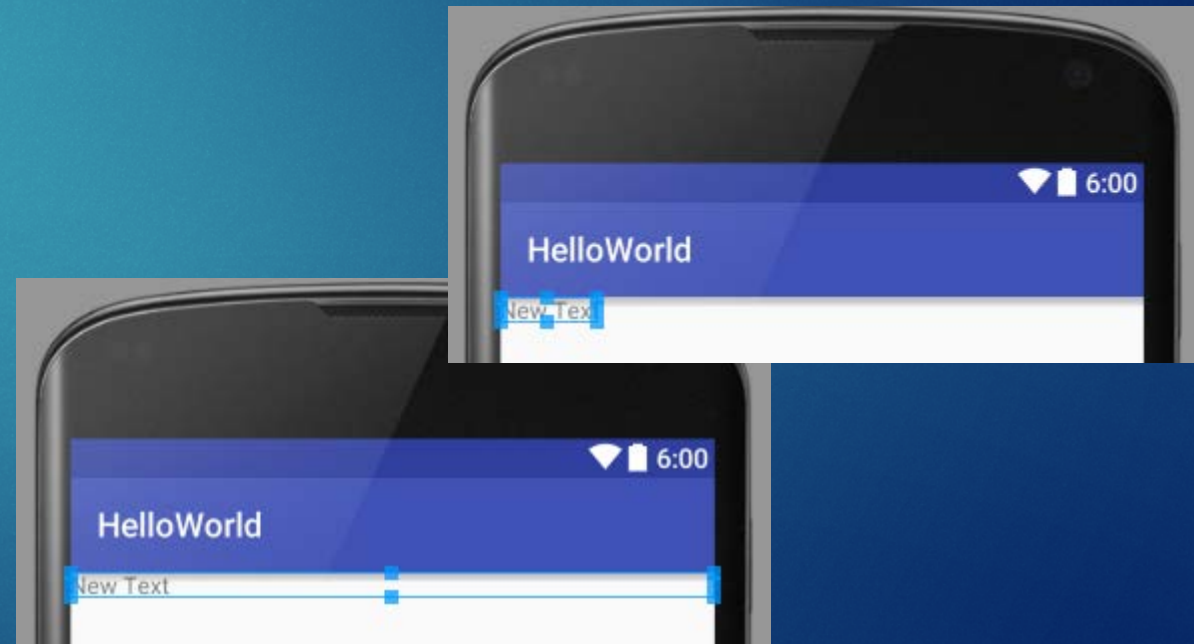
- ▶ Match\_parent
- ▶ Wrap\_content
- ▶ Use background tint for visual cues

```
android:background="#ff0000"
```

- ▶ Xxxdp
  - ▶ Density-independent pixel
  - ▶  $px = dp * density$
  - ▶  $density = bucket(dpi/160)$

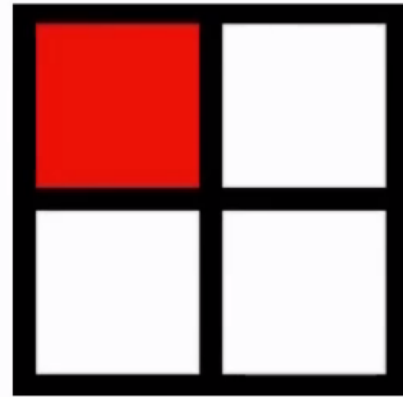


```
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:orientation="vertical"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent">  
  
    <TextView  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:text="New Text"  
        android:id="@+id/textView2" />  
  
</LinearLayout>
```



# Size - dp

**Normal density**

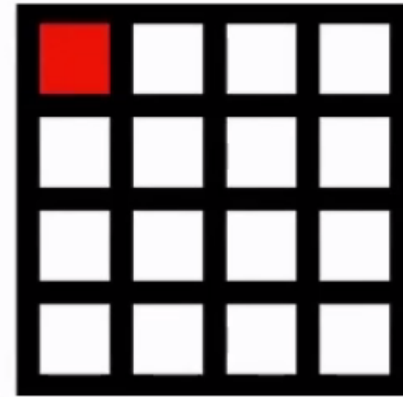


1px

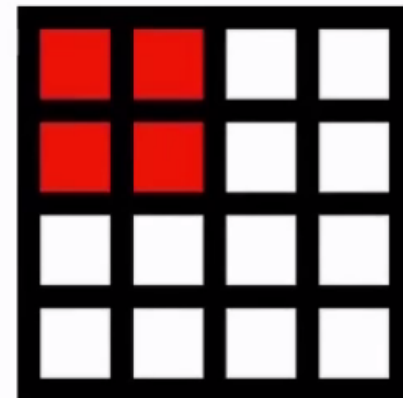


1dp

**Double density**

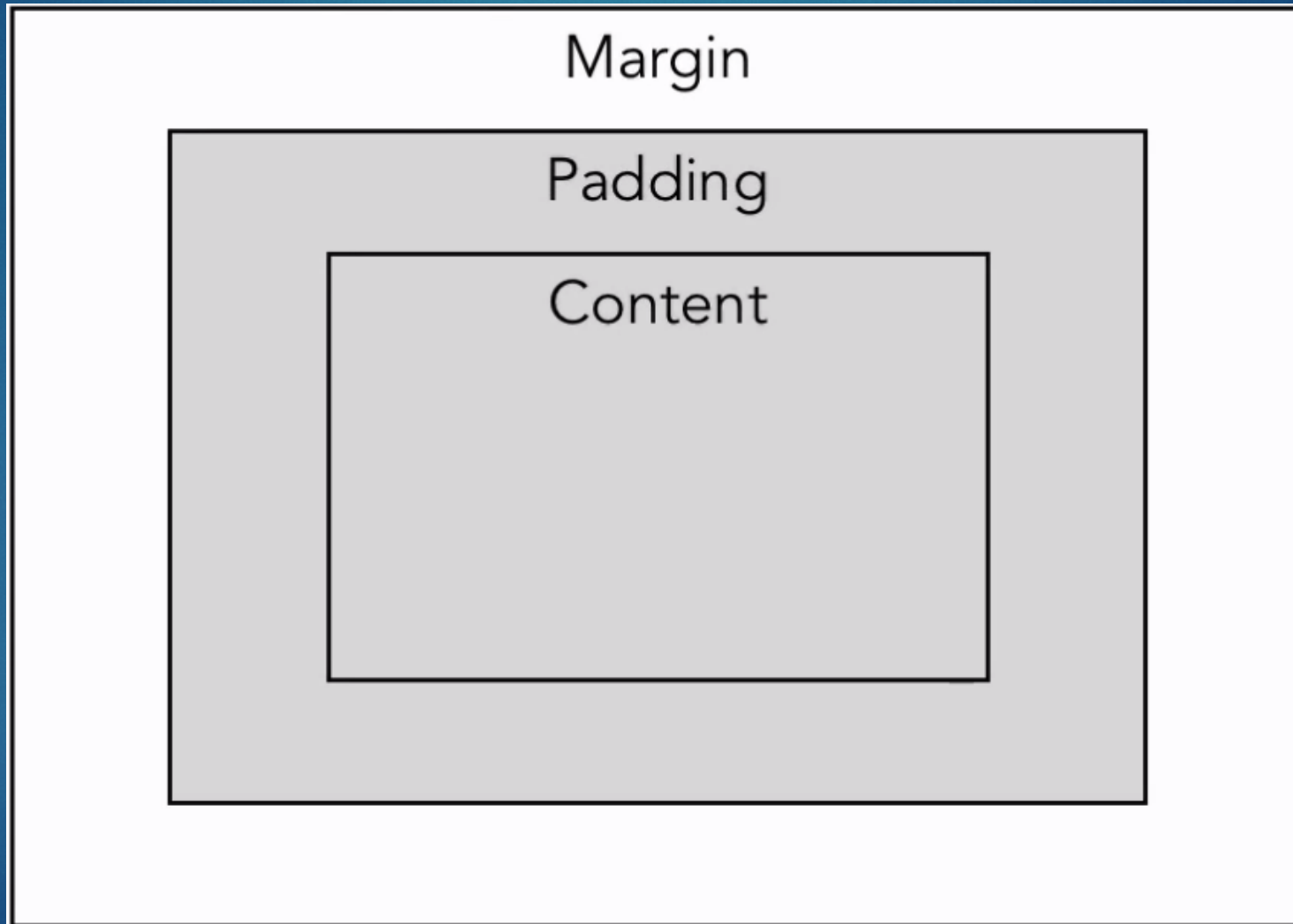


1px



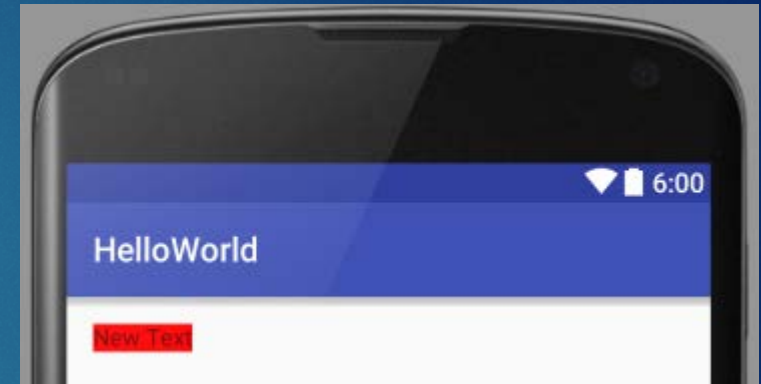
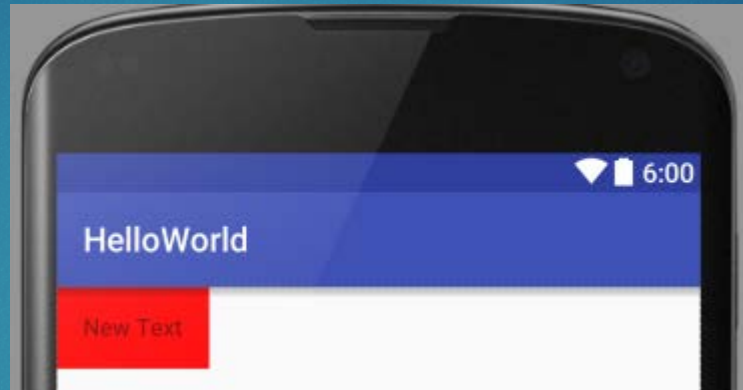
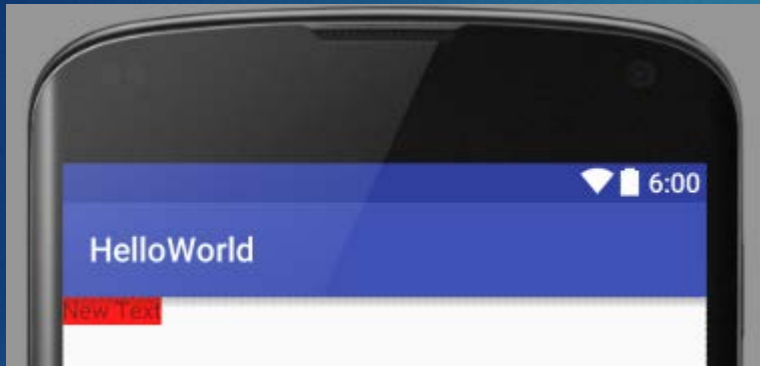
2px

# Margin vs padding



# Margin vs padding

13



- ▶ No margin
- ▶ android:padding
- ▶ android:layout\_margin
- ▶ layout\_XXX prefix – deals with data from outside the view
- ▶ padding – inside the view

# Gravity

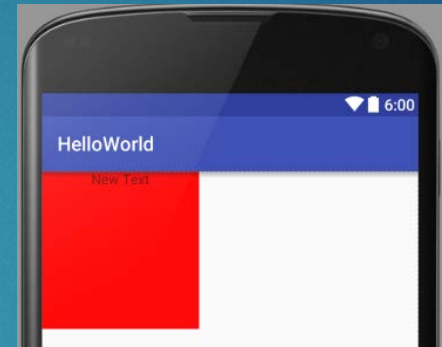
- ▶ `Android:layout_gravity`
  - ▶ Position of the view, regarding its parent
- ▶ `Android:gravity`
  - ▶ Position of the content inside the view

# Gravity

15

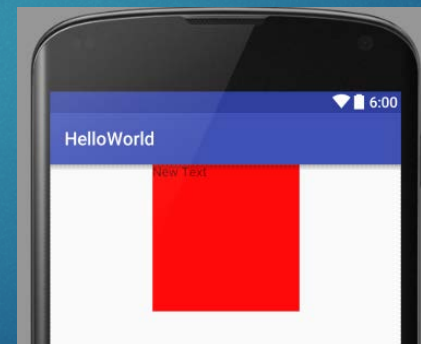
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="160dp"
        android:layout_height="160dp"
        android:gravity="center_horizontal"
        android:background="#ff0000"
        android:text="New Text"
        android:id="@+id/textView2" />
</LinearLayout>
```



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="160dp"
        android:layout_height="160dp"
        android:layout_gravity="center_horizontal"
        android:background="#ff0000"
        android:text="New Text"
        android:id="@+id/textView2" />
</LinearLayout>
```



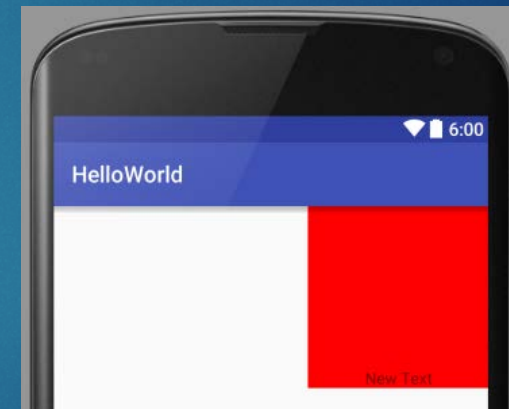
# Gravity

- ▶ Can be mixed and combined

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:layout_width="160dp"
        android:layout_height="160dp"
        android:gravity="center_horizontal|bottom"
        android:layout_gravity="right"
        android:background="#ff0000"
        android:text="New Text"
        android:id="@+id/textView2" />

</LinearLayout>
```

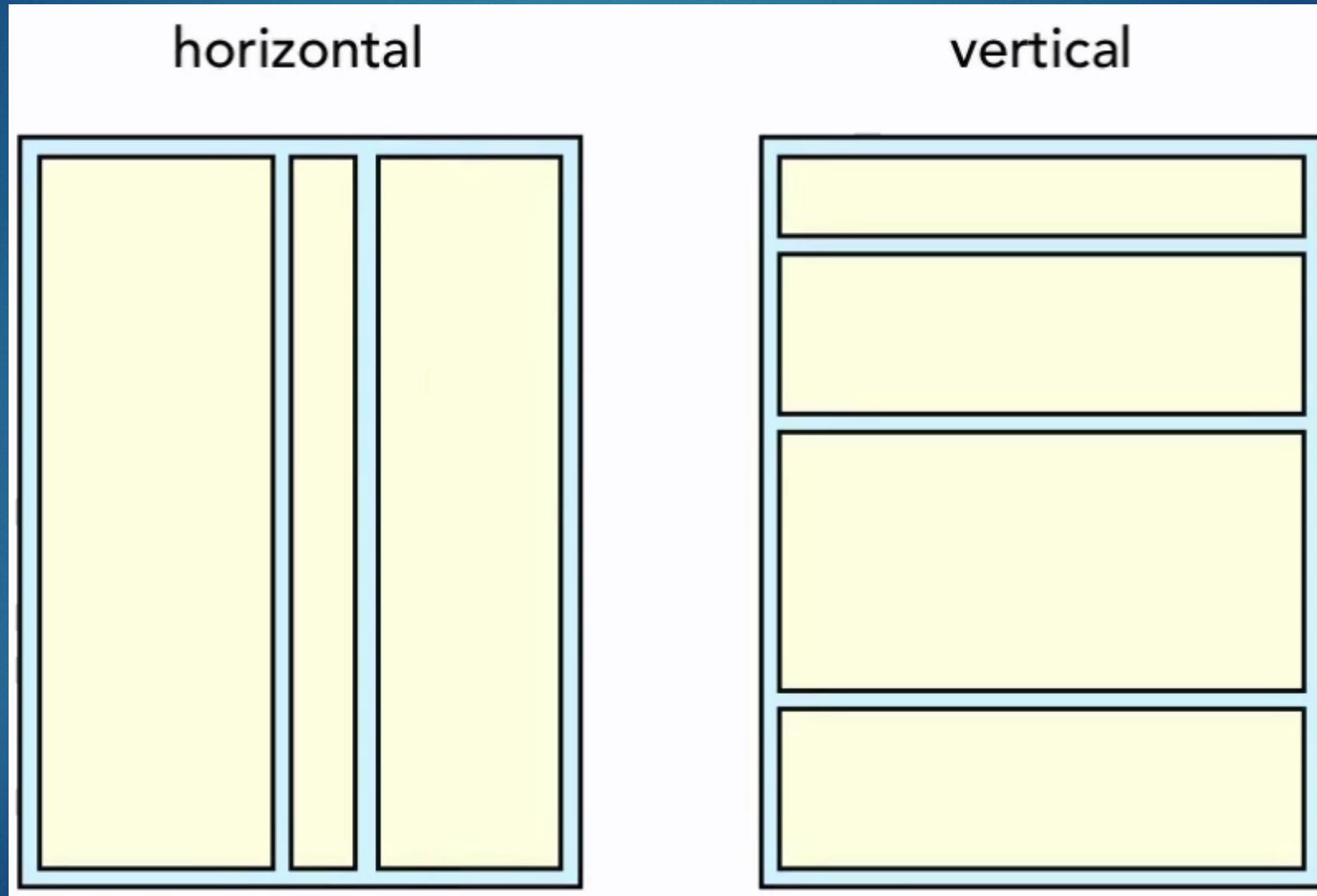




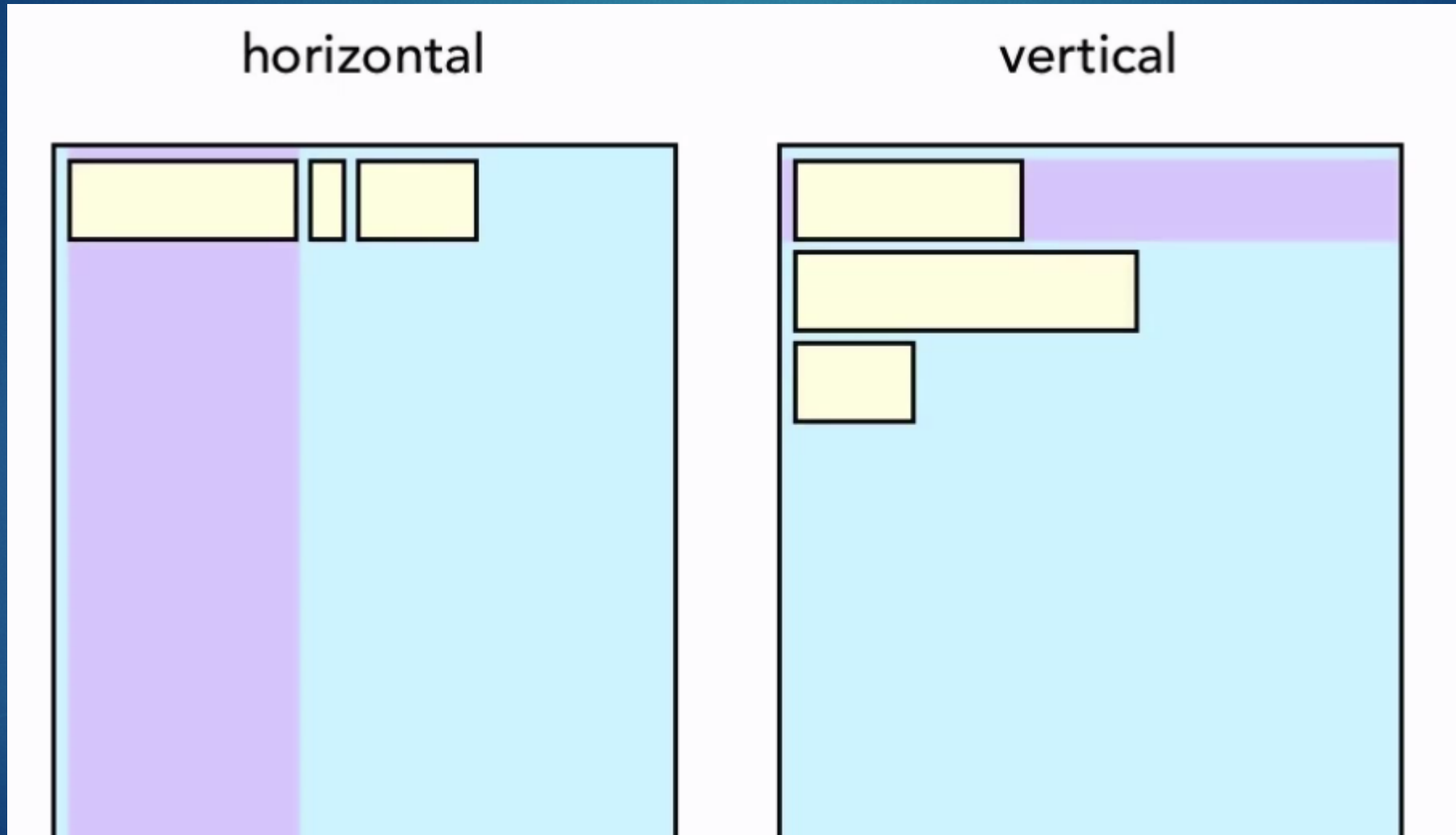
# LinearLayout

- ▶ What is LinearLayout?
- ▶ Gravity
- ▶ Weight
- ▶ Nested layout

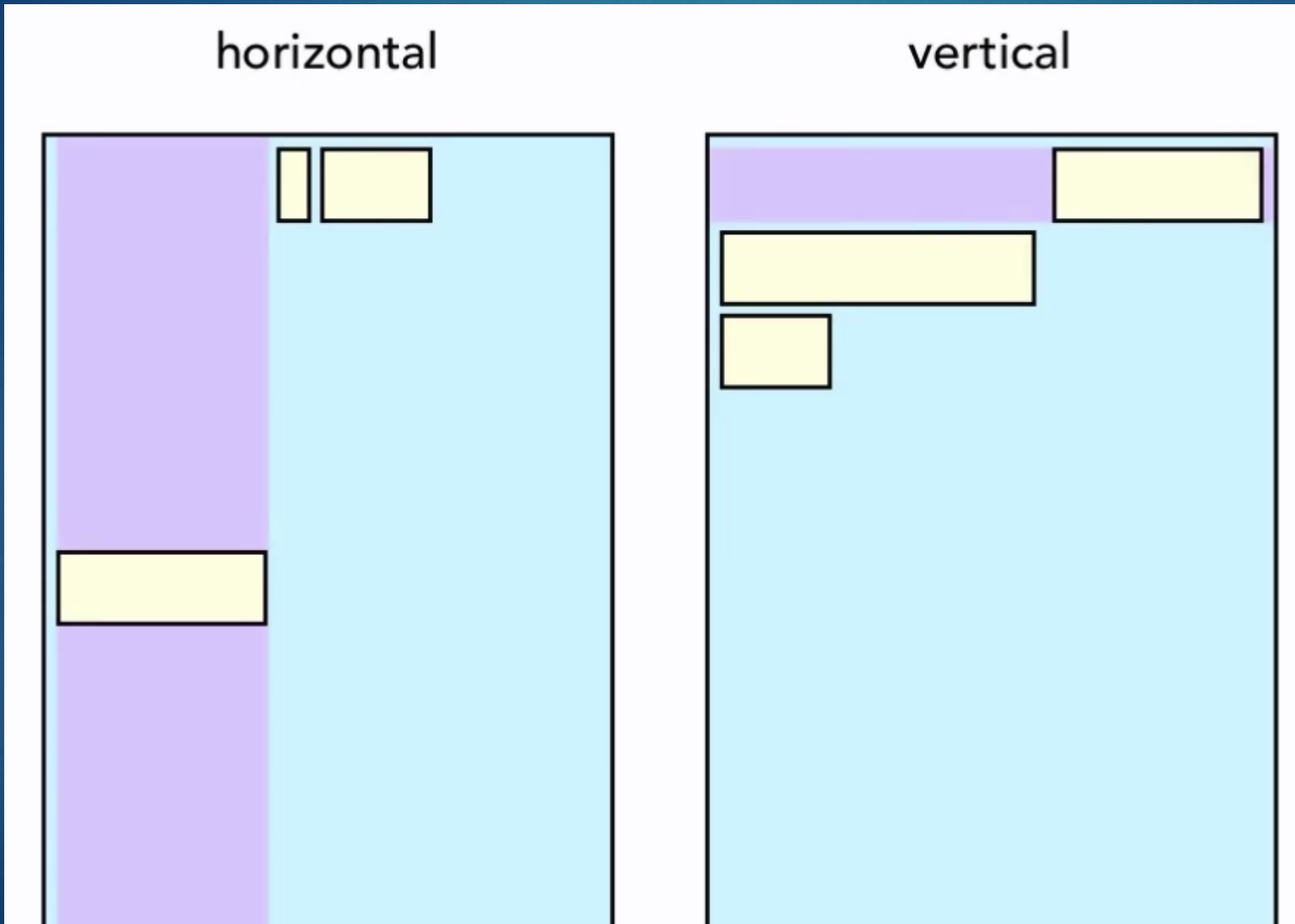
# LinearLayout



# LinearLayout - Gravity



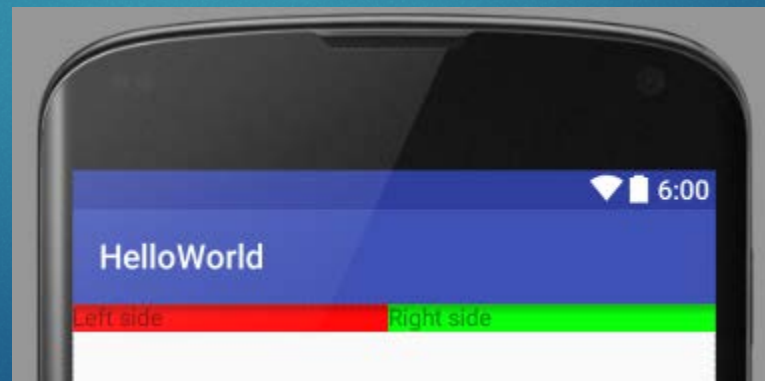
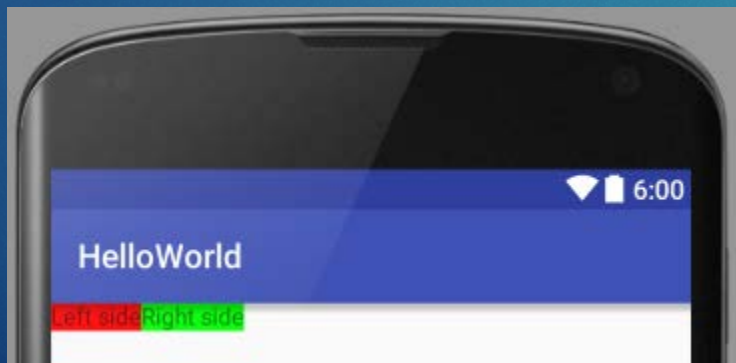
# LinearLayout - Gravity



# LinearLayout - Weight

21

- ▶ Additional way to specify dimensions (match\_parent, wrap\_content, xxxdp)
- ▶ Adds weights together
- ▶ Width is based on ratios ( $1+1=2 \Rightarrow \frac{1}{2}$  and  $\frac{1}{2}$ )
- ▶ Can be mixed every way



# LinearLayout - weightSum

22

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center_horizontal"
    android:weightSum="4">

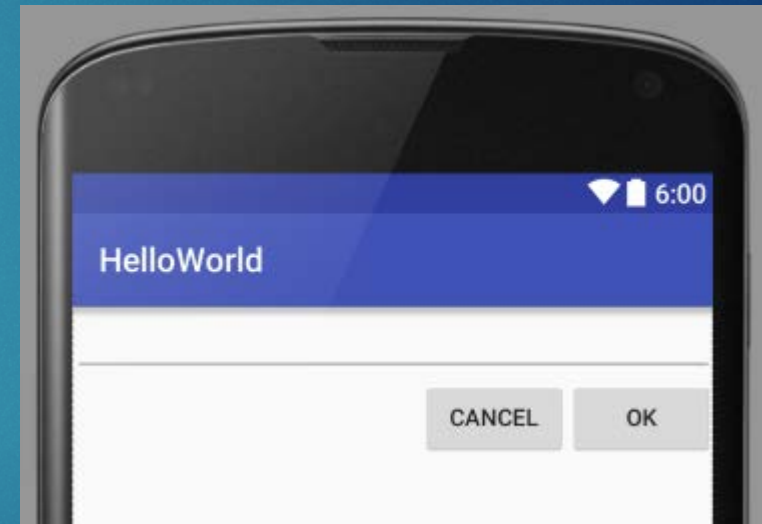
    <TextView
        android:layout_width="0dp"
        android:layout_weight="1"
        android:layout_height="wrap_content"
        android:background="#ff0000"
        android:text="Left side"
        android:id="@+id/textView2" />
    <TextView
        android:layout_width="0dp"
        android:layout_weight="1"
        android:layout_height="wrap_content"
        android:background="#00ff00"
        android:text="Right side"
        android:id="@+id/textView3" />
</LinearLayout>
```



# LinearLayout - nested

23

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/editText3" />
    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="right">
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Cancel"
            android:id="@+id/button6" />
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="OK"
            android:id="@+id/button7" />
    </LinearLayout>
</LinearLayout>
```

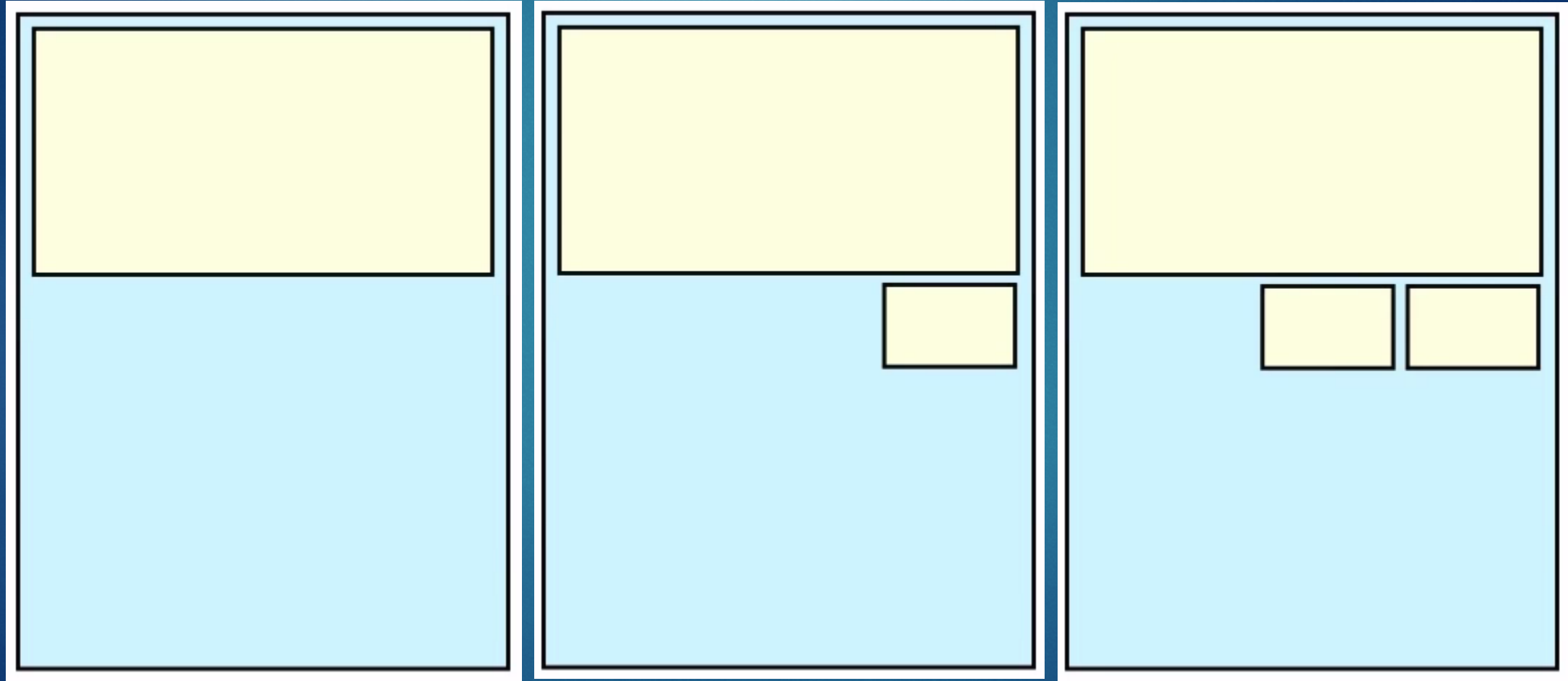


# RelativeLayout

- ▶ What it is?
- ▶ Relative Positioning
- ▶ Relative Alignement
- ▶ Missing Views



# RelativeLayout



# RelativeLayout

- ▶ Relative
  - ▶ Position
  - ▶ Alignment
- ▶ To
  - ▶ Parent
  - ▶ Sibling

# RelativeLayout - attributes

XML Attributes		
Attribute Name	Related Method	Description
<a href="#">android:layout_above</a>		Positions the bottom edge of this view above the given anchor view ID.
<a href="#">android:layout_alignBaseline</a>		Positions the baseline of this view on the baseline of the given anchor view ID.
<a href="#">android:layout_alignBottom</a>		Makes the bottom edge of this view match the bottom edge of the given anchor view ID.
<a href="#">android:layout_alignEnd</a>		Makes the end edge of this view match the end edge of the given anchor view ID.
<a href="#">android:layout_alignLeft</a>		Makes the left edge of this view match the left edge of the given anchor view ID.
<a href="#">android:layout_alignParentBottom</a>		If true, makes the bottom edge of this view match the bottom edge of the parent.
<a href="#">android:layout_alignParentEnd</a>		If true, makes the end edge of this view match the end edge of the parent.
<a href="#">android:layout_alignParentLeft</a>		If true, makes the left edge of this view match the left edge of the parent.
<a href="#">android:layout_alignParentRight</a>		If true, makes the right edge of this view match the right edge of the parent.
<a href="#">android:layout_alignParentStart</a>		If true, makes the start edge of this view match the start edge of the parent.
<a href="#">android:layout_alignParentTop</a>		If true, makes the top edge of this view match the top edge of the parent.
<a href="#">android:layout_alignRight</a>		Makes the right edge of this view match the right edge of the given anchor view ID.
<a href="#">android:layout_alignStart</a>		Makes the start edge of this view match the start edge of the given anchor view ID.
<a href="#">android:layout_alignTop</a>		Makes the top edge of this view match the top edge of the given anchor view ID.
<a href="#">android:layout_alignWithParentIfMissing</a>		If set to true, the parent will be used as the anchor when the anchor cannot be found for <code>layout_toLeftOf</code> , <code>layout_toRightOf</code> , etc.
<a href="#">android:layout_below</a>		Positions the top edge of this view below the given anchor view ID.
<a href="#">android:layout_centerHorizontal</a>		If true, centers this child horizontally within its parent.
<a href="#">android:layout_centerInParent</a>		If true, centers this child horizontally and vertically within its parent.
<a href="#">android:layout_centerVertical</a>		If true, centers this child vertically within its parent.
<a href="#">android:layout_toEndOf</a>		Positions the start edge of this view to the end of the given anchor view ID.
<a href="#">android:layout_toLeftOf</a>		Positions the right edge of this view to the left of the given anchor view ID.
<a href="#">android:layout_toRightOf</a>		Positions the left edge of this view to the right of the given anchor view ID.
<a href="#">android:layout_toStartOf</a>		Positions the end edge of this view to the start of the given anchor view ID.

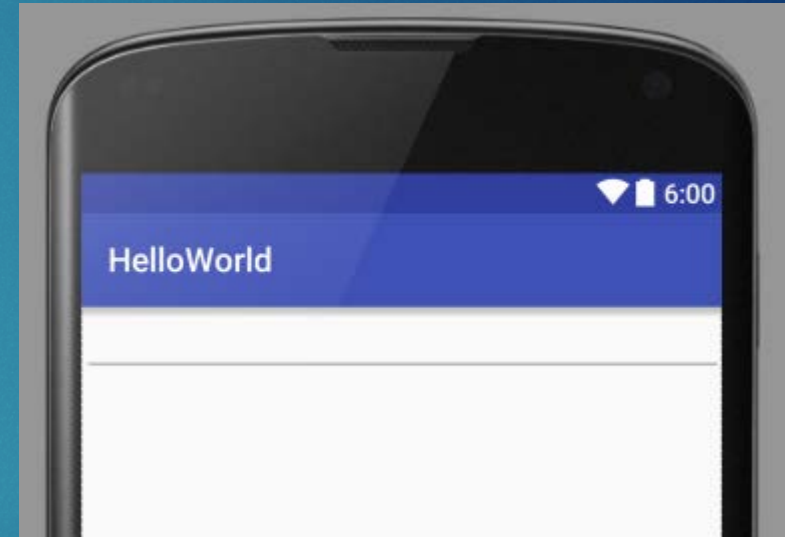
# RelativeLayout

28

- ▶ Uses default parameters
  - ▶ Top
  - ▶ Left

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/editText2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textMultiLine" />
</RelativeLayout>
```



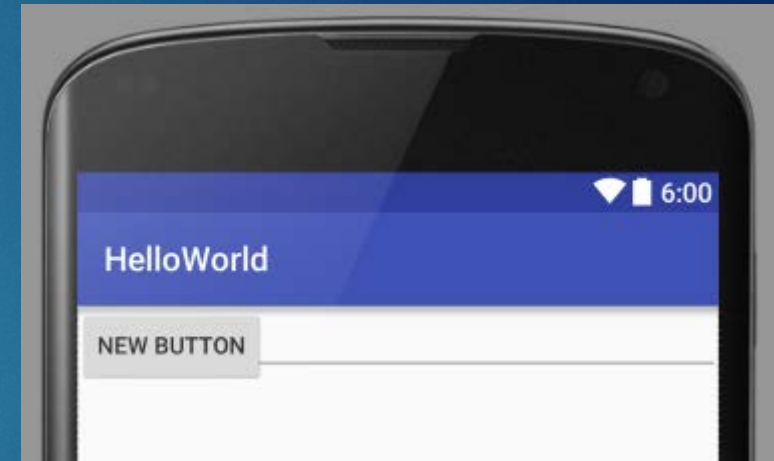
# RelativeLayout

- ▶ No positions given
  - ▶ Views are on top of each other

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/editText2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textMultiLine" />

    <Button
        android:id="@+id/button4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="New Button" />
</RelativeLayout>
```



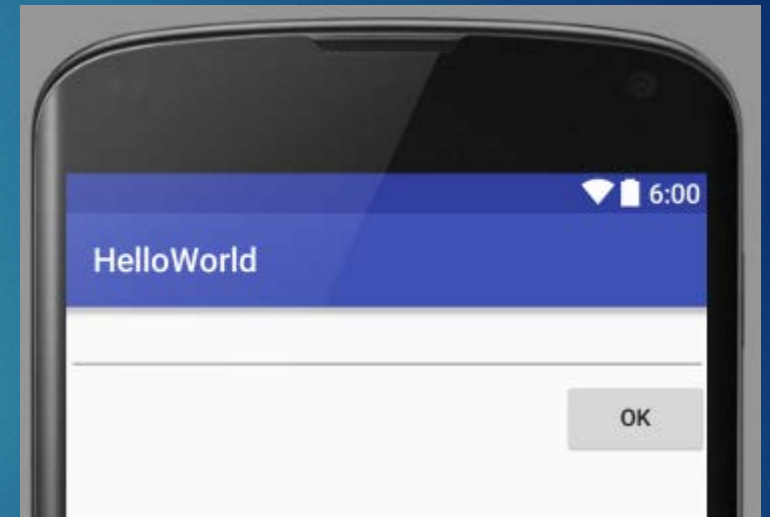
# RelativeLayout

30

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/editTextGreeting"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textMultiLine" />

    <Button
        android:id="@+id/buttonOk"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextGreeting"
        android:layout_alignParentRight="true"
        android:text="OK" />
</RelativeLayout>
```



# RelativeLayout

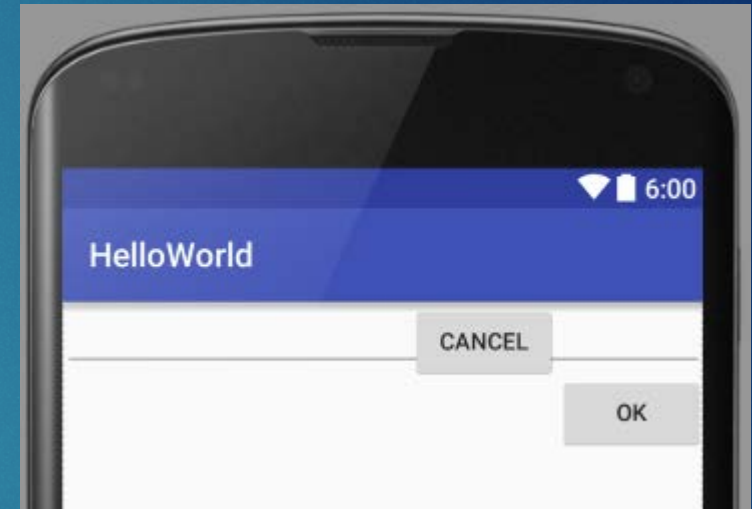
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/editTextGreeting"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textMultiLine" />

    <Button
        android:id="@+id/buttonOk"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextGreeting"
        android:layout_alignParentRight="true"
        android:text="OK" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toLeftOf="@id/buttonOk"
        android:text="Cancel"
        android:id="@+id/buttonCancel" />

</RelativeLayout>
```



# RelativeLayout

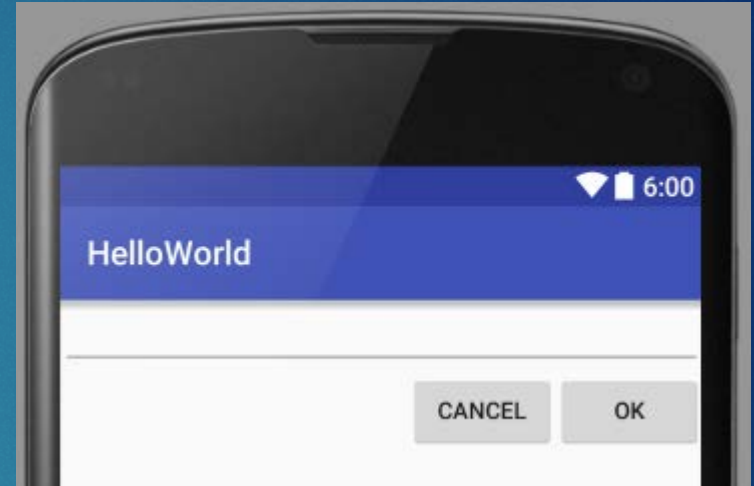
32

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/editTextGreeting"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textMultiLine" />

    <Button
        android:id="@+id/buttonOk"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextGreeting"
        android:layout_alignParentRight="true"
        android:text="OK" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toLeftOf="@id/buttonOk"
        android:layout_below="@id/editTextGreeting"
        android:text="Cancel"
        android:id="@+id/buttonCancel" />
</RelativeLayout>
```





# RelativeLayout

- ▶ Relative to sibling

- ▶ Layout\_above
- ▶ Layout\_below
- ▶ Layout\_toLeftOf
- ▶ Layout\_toRightOf

- ▶ Relative to sibling

- ▶ Layout\_alignTop
- ▶ Layout\_alignBottom
- ▶ Layout\_alignLeft
- ▶ Layout\_alignRight
- ▶ Layout\_alignBaseline

# RelativeLayout

- ▶ Relative to parent
  - ▶ Layout\_alignParentTop
  - ▶ Layout\_alignParentBottom
  - ▶ Layout\_alignParentLeft
  - ▶ Layout\_alignParentRight
  - ▶ Layout\_centerHorizontal
  - ▶ Layout\_centerVertical
  - ▶ Layout\_centerInParent

# RelativeLayout – Missing views

- ▶ `Visibility="invisible"`
  - ▶ View is still there, just invisible
  - ▶ Placement of element is not changed
- ▶ `Visibility="gone"`
  - ▶ UI layout changes, view is really not there
  - ▶ Defaults apply (top left)
- ▶ When changing visibility programmatically
  - ▶ `Layout_alignWithParentIfMissing="true"`

# FrameLayout

- ▶ Allows to put views on top of each other
  - ▶ Position child views with
    - ▶ `layout_gravity`
    - ▶ `Layout_margin`
- ▶ Usually used for placing images on top of other images
  - ▶ Think of `PictureFrame`
- ▶ You can save one view by using on `FrameLayout` definition
  - ▶ `Android:foreground="@drawable/somepicture"`
  - ▶ Control scaling of foreground (default is fill)
    - ▶ `Android:foregroundGravity="top | left"`

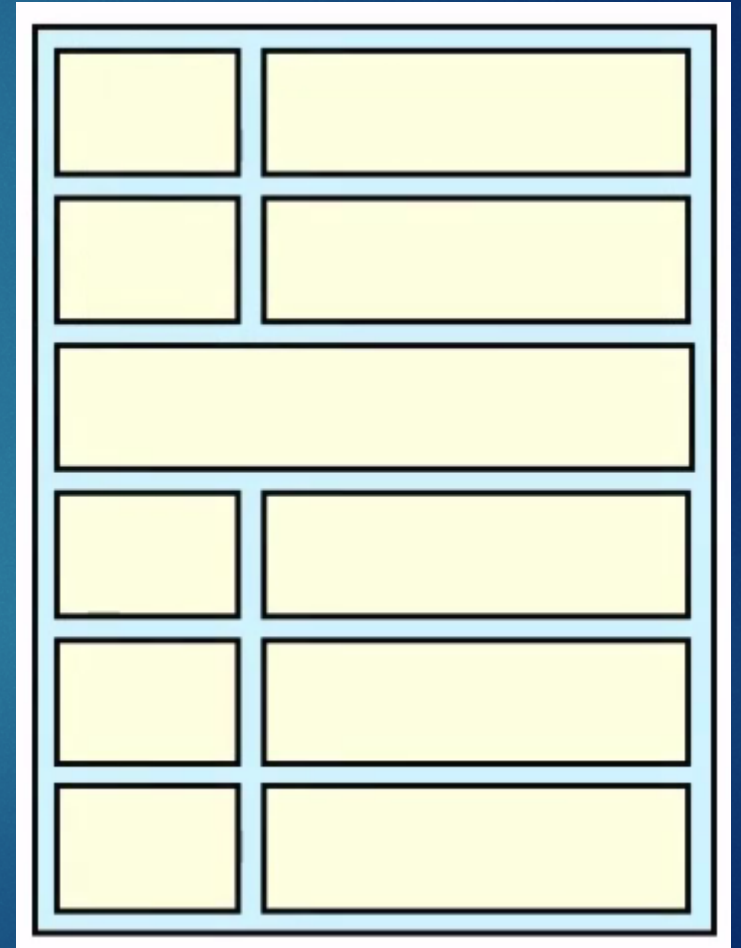
# TableLayout

- ▶ What is TableLayout
- ▶ Spanning and skipping
- ▶ Shrinking, stretching and collapsing

# TableLayout

38

- ▶ Specialized linear layout
  - ▶ Rows
  - ▶ Columns
  - ▶ TableRow is always
    - ▶ `Layout_width="match_parent"`
    - ▶ `Layout_height="wrap_content"`



# TableLayout

39

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TableRow>
        <TextView
            android:background="#ff0000"
            android:text="First" />
        <TextView
            android:background="#00ff00"
            android:text="Second long" />
        <TextView
            android:background="#0000ff"
            android:text="Third0" />
    </TableRow>
    <TableRow>
        <TextView
            android:background="#ff0000"
            android:text="First long" />
        <TextView
            android:background="#00ff00"
            android:text="Second" />
        <TextView
            android:background="#0000ff"
            android:text="Third" />
    </TableRow>
</TableLayout>
```



# TableLayout – skipping

40

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TableRow>
        <TextView
            android:background="#ffff00"
            android:text="Very long title"
            android:layout_column="1"/>
    </TableRow>
    <TableRow>
        <TextView
            android:background="#ff0000"
            android:text="First" />
        <TextView
            android:background="#00ff00"
            android:text="Second long" />
        <TextView
            android:background="#0000ff"
            android:text="Third0" />
    </TableRow>
    <TableRow>
        <TextView
            android:background="#ff0000"
            android:text="First long" />
        <TextView
            android:background="#00ff00"
            android:text="Second" />
        <TextView
            android:background="#0000ff"
            android:text="Third" />
    </TableRow>
</TableLayout>
```





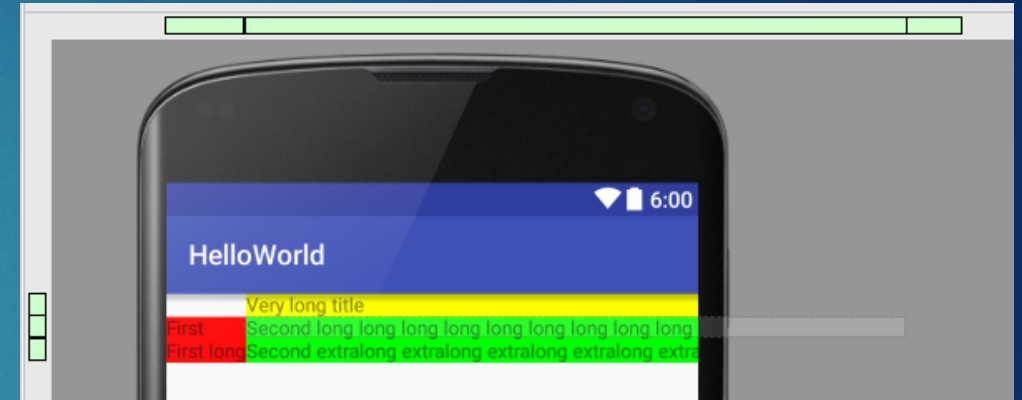
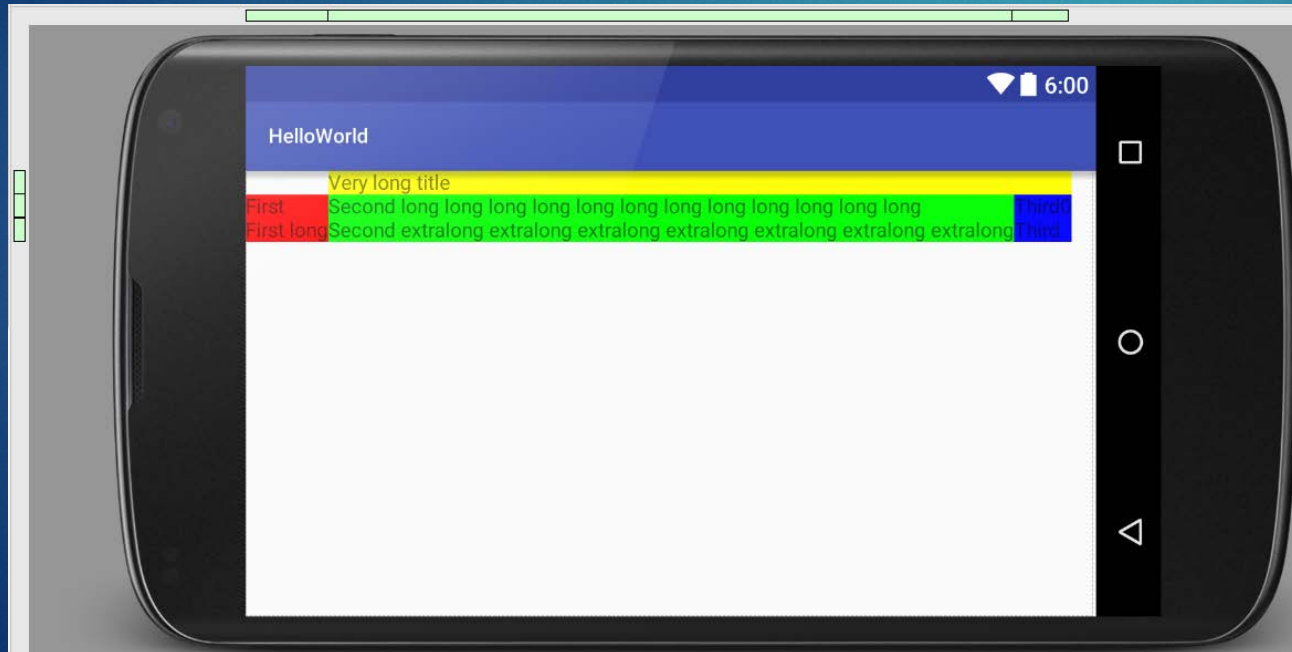
# TableLayout – spanning

41

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TableRow>
        <TextView
            android:background="#ffff00"
            android:text="Very long title"
            android:layout_column="1"
            android:layout_span="2"/>
    </TableRow>
    <TableRow>
        <TextView
            android:background="#ff0000"
            android:text="First" />
        <TextView
            android:background="#00ff00"
            android:text="Second long" />
        <TextView
            android:background="#0000ff"
            android:text="Third0" />
    </TableRow>
    <TableRow>
        <TextView
            android:background="#ff0000"
            android:text="First long" />
        <TextView
            android:background="#00ff00"
            android:text="Second" />
        <TextView
            android:background="#0000ff"
            android:text="Third" />
    </TableRow>
</TableLayout>
```



# TableLayout – shrinking, stretching, collapsing





# TableLayout – stretching

44

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:shrinkColumns="1"
    android:stretchColumns="*">
    <TableRow>
        <TextView
            android:layout_column="0"
            android:layout_marginLeft="4dp" android:layout_marginRight="4dp"
            android:layout_span="3"
            android:background="#ffff00"
            android:text="Very long title" />
    </TableRow>
    <TableRow>
        <TextView
            android:layout_height="match_parent"
            android:layout_marginLeft="4dp" android:layout_marginRight="4dp"
            android:background="#ff0000"
            android:text="First" />
        <TextView
            android:layout_height="match_parent"
            android:layout_marginRight="4dp"
            android:background="#00ff00"
            android:singleLine="false"
            android:text="Second long long long long long long long long long long long long long long end" />
        <TextView
            android:layout_height="match_parent"
            android:layout_marginRight="4dp"
            android:background="#0000ff"
            android:text="Third0" />
    </TableRow>
</TableLayout>
```



# TableLayout – collapsing

45

- ▶ `Android:collapseColumns=" *"`
  - ▶ `*`
  - ▶ `0`
  - ▶ `1,2,4`
- ▶ Same as `visibility="gone"`

## How many passes?

Layout	Normal	Extra
LinearLayout	1	2 with weight
RelativeLayout	2	
FrameLayout	1	
TableLayout	2	3 with shrinking and stretching

# Varia – Hierarchy Viewer

47

- ▶ ViewServer
  - ▶ Download ViewServer.java from <https://github.com/romainguy/ViewServer>
  - ▶ Add INTERNET permission
  - ▶ Enable ViewServer in your activity

# Varia – Hierarchy Viewer

48

- ▶ Launch Android Device Monitor
- ▶ Window -> Open Perspective... -> Hierarchy View
- ▶ Select your running app from tree
- ▶ Click the icon "Load the view hierarchy into the tree view"