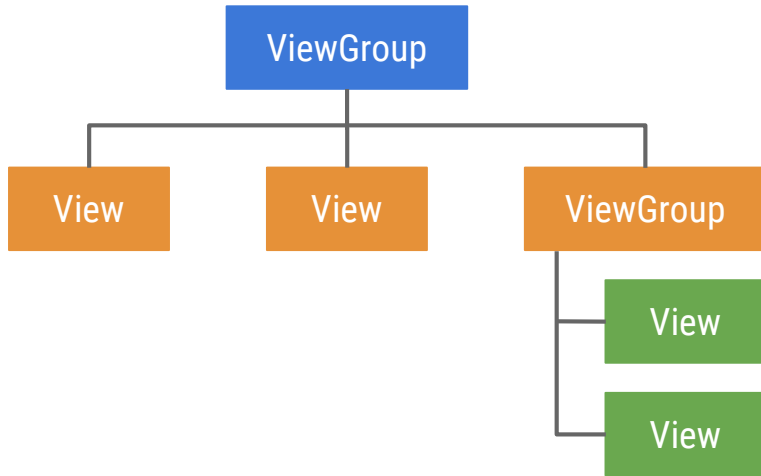




LinearLayout

Layout
Fundamentals

View Structure in Android



What is a View?

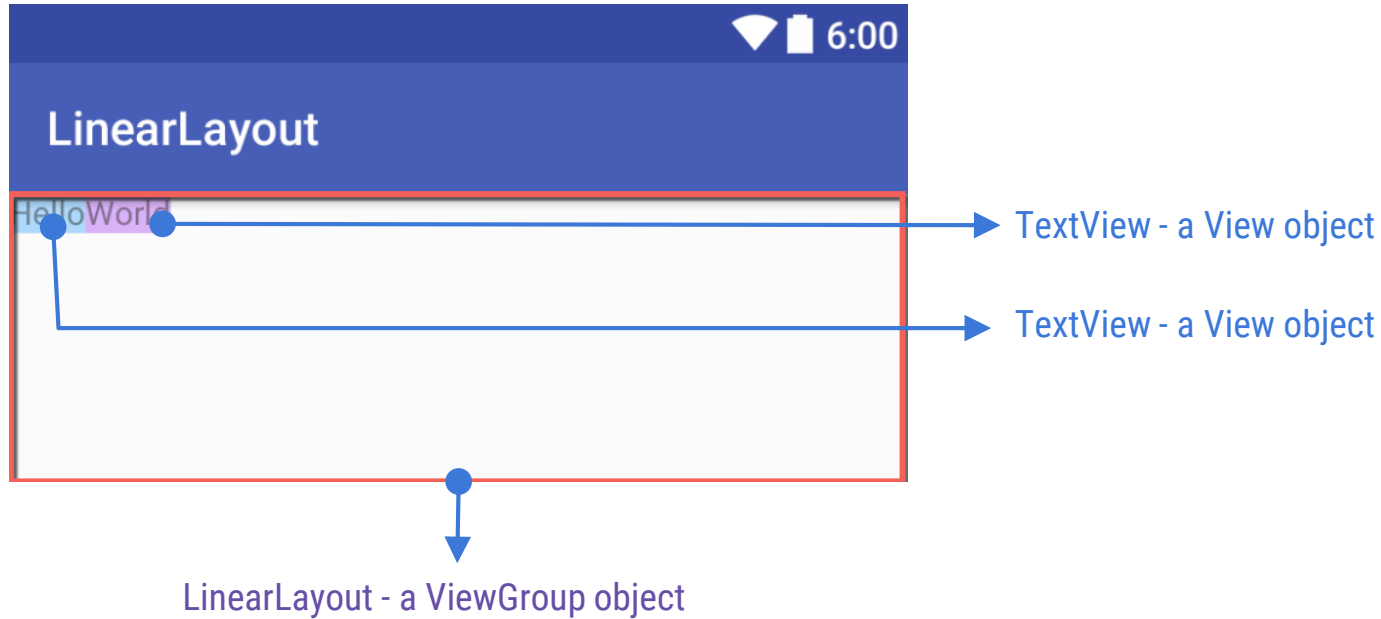
- An object that takes rectangular area on the screen and is responsible for handling events in that area.

What is a ViewGroup?

- An object that surrounds the view and is invisible and is responsible for holding views in a certain fashion.



View Structure in Android



How to use a LinearLayout?

- When you define your UI appearance in XML, you always have a **ViewGroup** inside which all the other **Views** and **ViewGroups** are placed. This main **ViewGroup** is called the **root of the View hierarchy**.
- **match_parent** - means take all the available space
- **wrap_content** - means take only as much space as needed by my content
- Add other view contents and either arrange them in same row (horizontal) or 1 per row (vertical).



Margin vs Padding



layout_margin

```
<LinearLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:orientation="vertical">
```

```
    <TextView
```

```
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
```

```
        android:layout_margin="16dp"
```

```
        android:text="@string/hello"
```

```
        android:background="#ff6" />
```

```
</LinearLayout>
```



padding

```
<LinearLayout
```

```
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
```

```
    android:orientation="vertical">
```

```
    <TextView
```

```
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
```

```
        android:padding="16dp"
```

```
        android:text="@string/hello"
```

```
        android:background="#ff6" />
```

```
</LinearLayout>
```



Hello world!



Margin

`android:layout_margin`

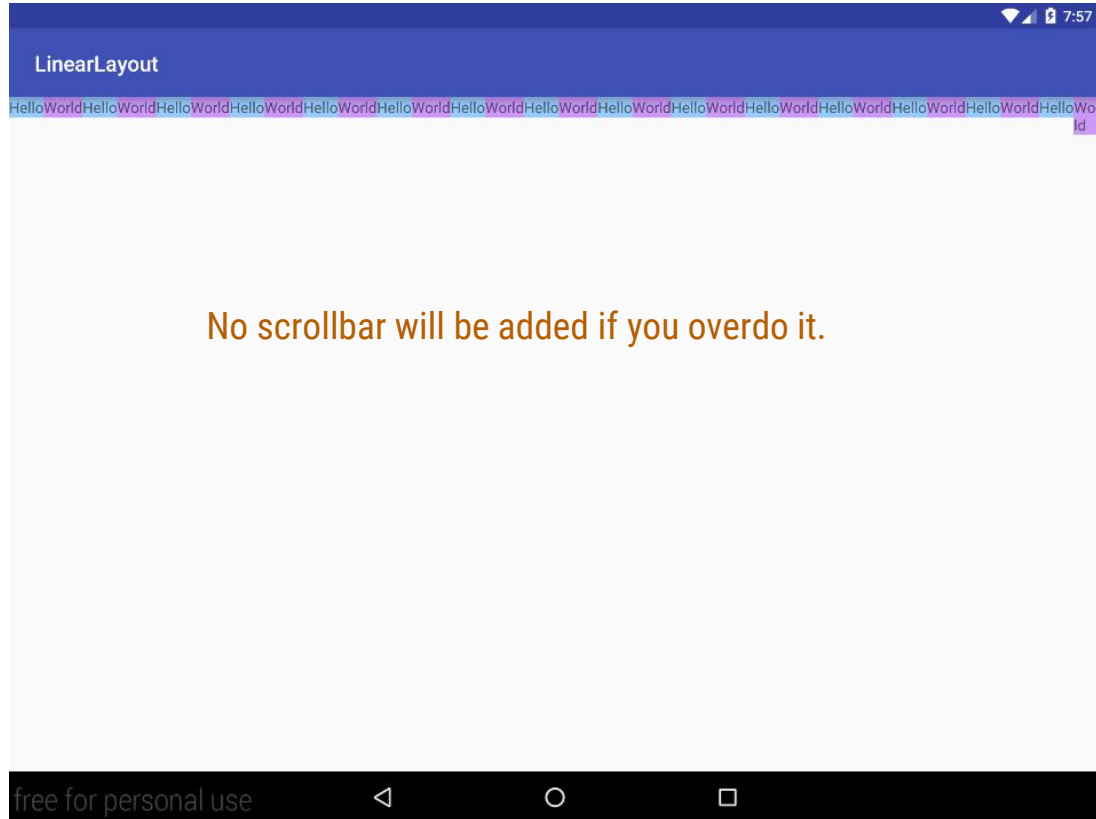
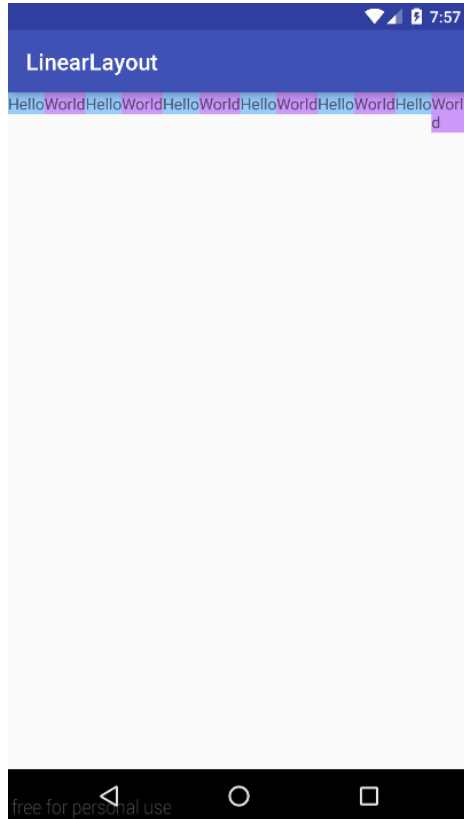
Padding

`android:padding`

Content



What if you overdo adding contents?

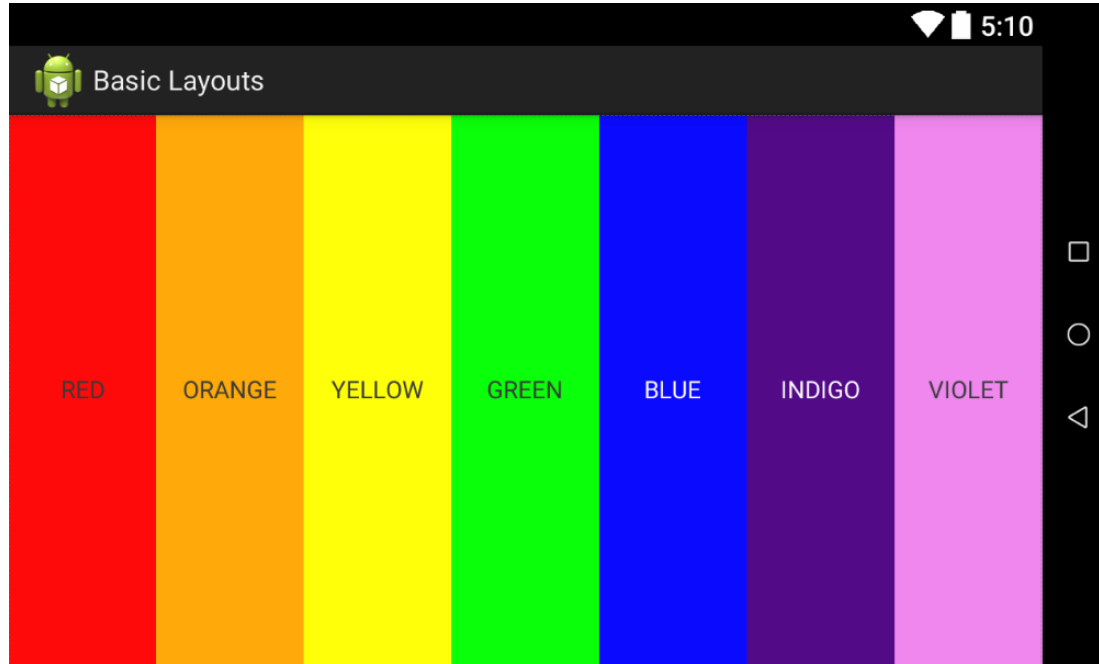


linear_exer1.xml (portrait)

Color	Value
red	#f00
orange	#ffa500
yellow	#ffff00
green	#0f0
blue	#00f
indigo	#4b0082
violet	#ee82ee



linear_exer1.xml (landscape)



android:layout_weight

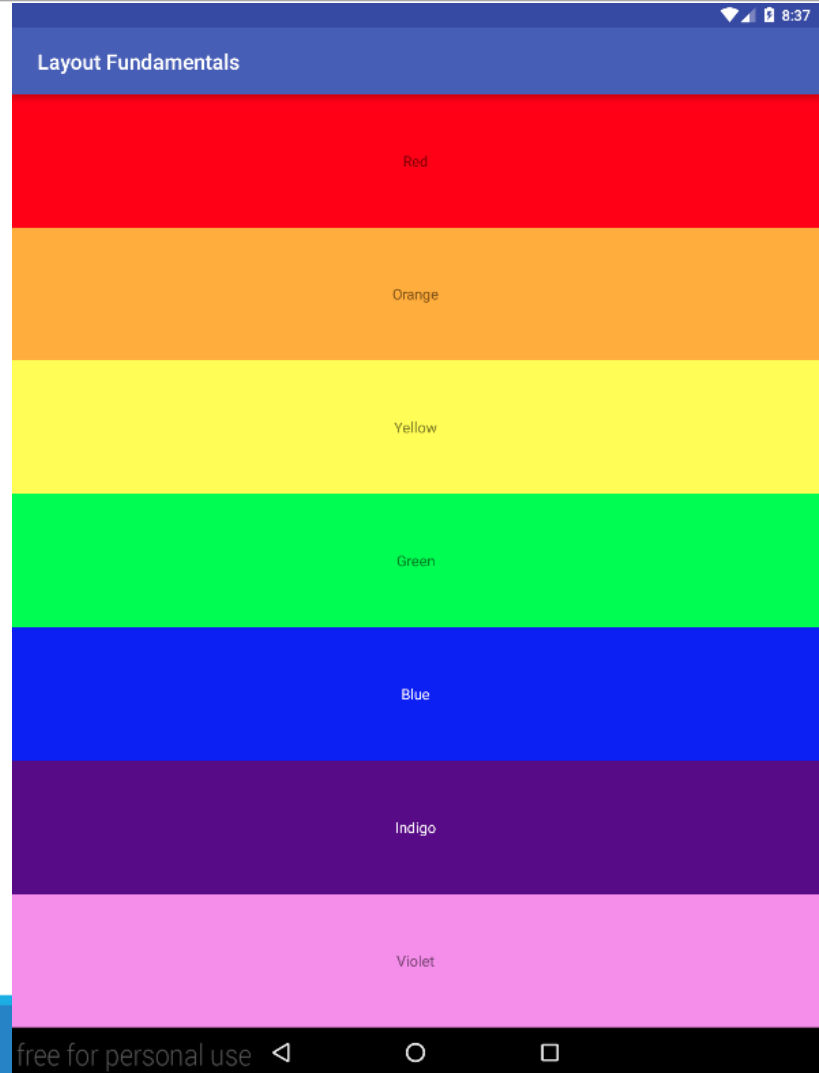


Layout Weight

- Assign importance on how much screen a View can relatively occupy compared to others.
- With 1 view, `layout_weight` acts like a priority, when more than 1 view is declared `layout_weight`, Android calculates how much space each View should occupy.
- By default, `android:layout_weight="0"` is set for all controls.



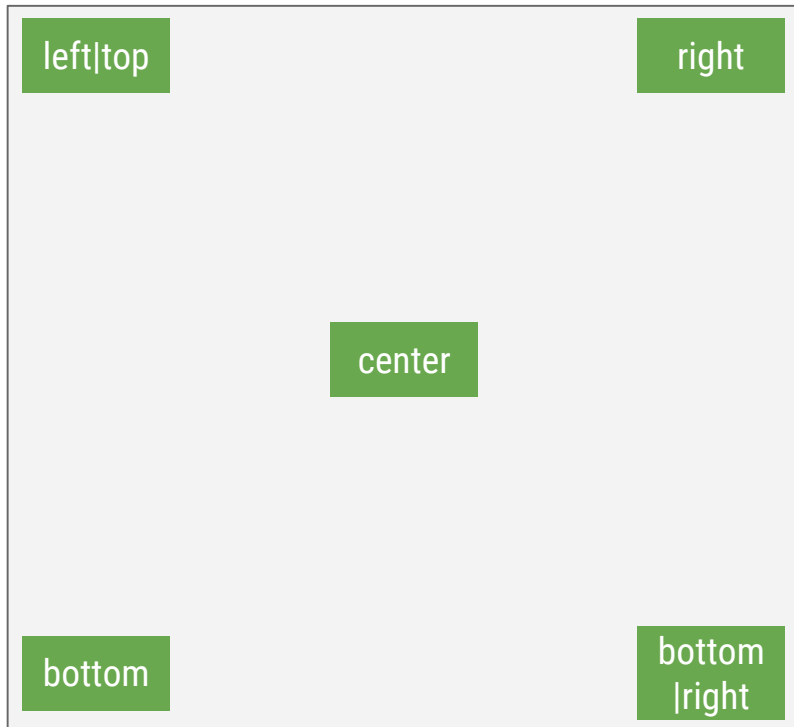
Consistency



android:layout_gravity



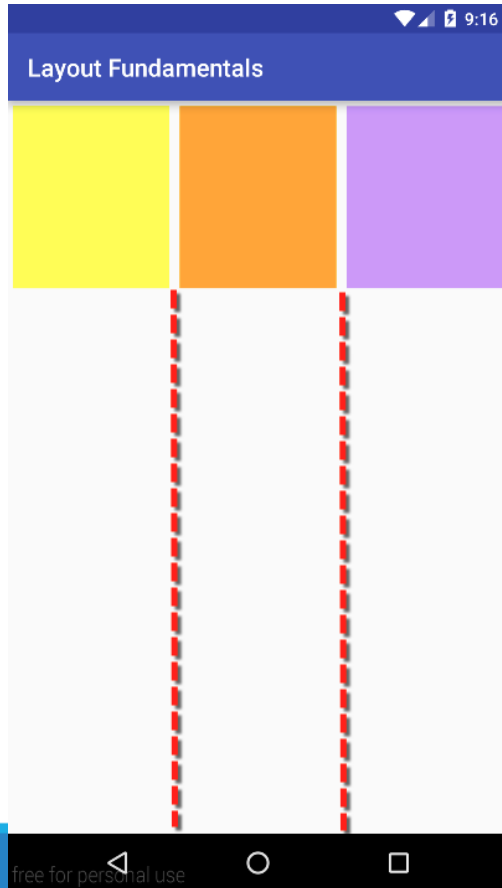
What is layout_gravity?



- Controls where a View should lie inside its parent.
- The parent should have a size bigger than its view so that layout_gravity takes effect.



What is layout_gravity (horizontal)?



- For a horizontal linear layout, layout_gravity values do not change columns.
- Only when the parent is bigger than the View, you will see the effect of layout_gravity.



What is layout_gravity (vertical)?



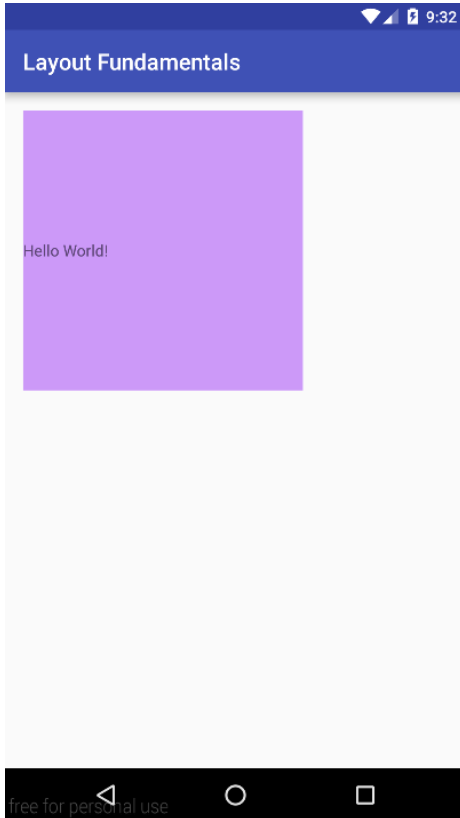
- For a vertical linear layout, layout_gravity values do not change rows.



android:gravity



What is android:gravity?



- Control where the contents of the View should appear inside the View or a ViewGroup.



linear_calc.xml (landscape)

