***Представления элементов***

**Цель работы**

Познакомится с основными приемами визуального представления элементов приложения, управления их взаимным расположением, оформлением, работы с соответствующими атрибутами из XML и из Java кода.

### Контрольные вопросы

1. С помощью какого класса предоставляется доступ к ресурсам из кода Java?

Доступ к ресурсам из кода Java предоставляется с помощью класса R.

1. Какие основные квалификаторы ресурсов вы знаете?

* color — файлы, определяющие значения цветов.
* drawable — файлы с изображениями.
* layout — файлы с layout пользовательских интерфейсов.
* menu — файлы, которые определяют меню приложения.
* values — файлы, содержащие простые значения такие, как строки, числа и цвета.

1. Какие основные XML-атрибуты используются для задания расположения виджета на экране?

Для RelativeLayout существует множество способов для позиционирования:

1) слева, справа, сверху, снизу указанного элемента (layout\_toLeftOf, layout\_toRightOf, layout\_above, layout\_below)

2) выровненным по левому, правому, верхнему, нижнему краю указанного элемента (layout\_alignLeft, layout\_alignRight, layout\_alignTop, layout\_alignBottom)

3) выровненным по левому, правому, верхнему, нижнему краю родителя (layout\_alignParentLeft, layout\_alignParentRight, layout\_alignParentTop, layout\_alignParentBottom)

4) выровненным по центру вертикально, по центру горизонтально, по центру вертикально и горизонтально относительно родителя (layout\_centerVertical, layout\_centerHorizontal, layout\_centerInParent)

Для AbsoluteLayout используются атрибуты layout\_x и layout\_y, чтобы указать координаты левого верхнего угла компонента.

С помощью атрибута layout\_gravity можно задать расположение компонента относительно родителя.

Атрибуты layout\_margin, layout\_marginTop, layout\_marginLeft и др. предназначены для того, чтобы задать отступ от элемента.

1. Какие основные XML-атрибуты используются для задания отображения виджета на экране?

Для LinearLayout используется layout\_weight, чтобы указать, сколько свободного места будет занимать компонент относительно других.

Атрибуты layout\_width и layout\_height задают ширину и высоту компонента соответственно. Значение можно указать как абсолютное значение, так и константное.

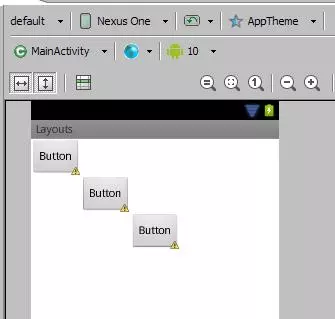
Часто используются атрибуты text и background для задания отображаемого текста компонента и фона соответственно.

### Дополнительные задания

1. Изучите все виды квалификаторов в окне создания нового ресурса. Изучите документацию и узнайте, для чего они служат.

|  |  |
| --- | --- |
| color | XML-файлы, определяющие «списки состояний» цветов. |
| drawable | Файлы растровых изображений и XML-файлы, скомпилированные в графические объекты. |
| mipmap | Иконки приложения для разных плотностей пикселей экрана. |
| layout | XML-файлы, определяющие пользовательский интерфейс. |
| menu | XML-файлы, определяющие меню приложений (например, меню параметров). |
| values | XML-файлы, содержащие простые значения, такие как строки, целые числа и цвета. |

1. Изучите основные виды расположений: LinearLayout, RelativeLayout, AbsoluteLayout, Table Layout. C помощью каждого из них постройте расположение элементов, как показано на рисунке:



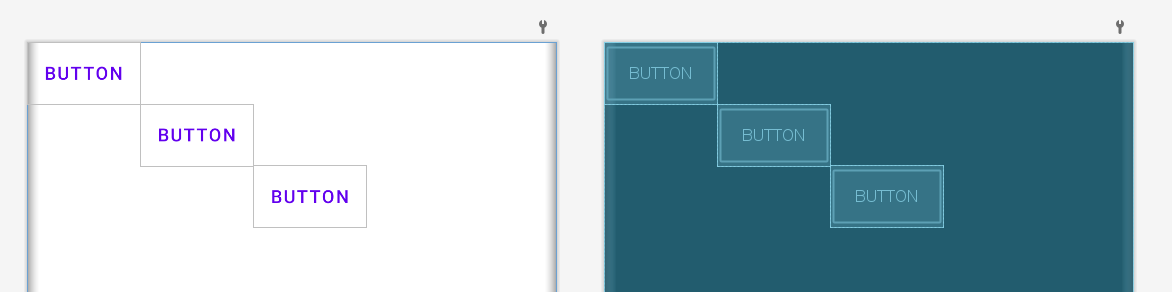
1. Проведите рефакторинг написанного приложения так, чтобы студия не показывала ни одного предупреждения.

strings.xml

<resources>  
 <string name="app\_name">Layouts</string>  
 <string name="button\_text">Button</string>  
</resources>

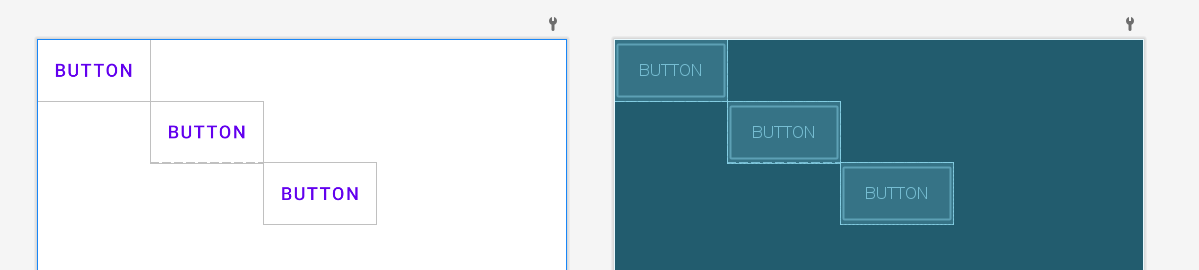
LinearLayout

*<?*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"  
 tools:context=".MainActivity">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="48dp"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="96dp"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
</LinearLayout>



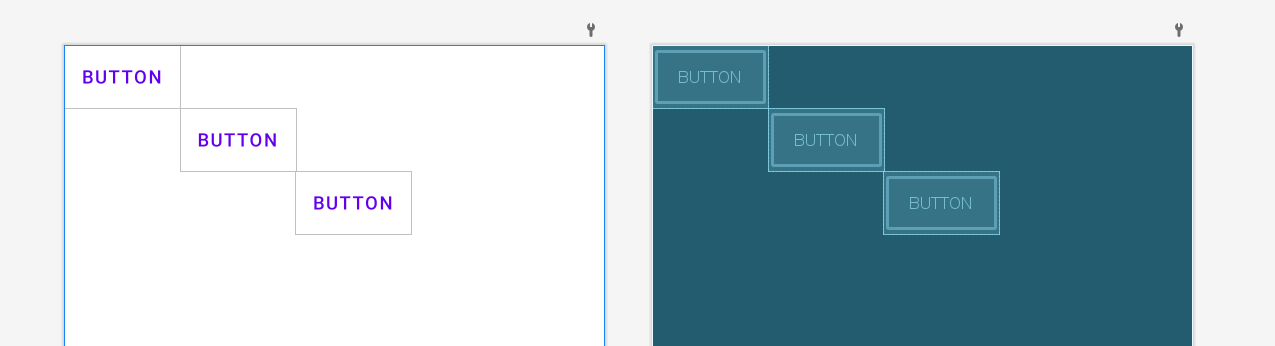
RelativeLayout

*<?*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"  
 tools:context=".MainActivity">  
  
  
 <Button  
 android:id="@+id/button1"  
 style="?android:attr/buttonBarButtonStyle"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text" />  
  
 <Button  
 android:id="@+id/button2"  
 style="?android:attr/buttonBarButtonStyle"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/button1"  
 android:layout\_toEndOf="@id/button1"  
 android:text="@string/button\_text" />  
  
 <Button  
 style="?android:attr/buttonBarButtonStyle"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/button2"  
 android:layout\_toEndOf="@id/button2"  
 android:text="@string/button\_text" />  
  
</RelativeLayout>



AbsoluteLayout

*<?*xml version="1.0" encoding="utf-8"*?>*<AbsoluteLayout  
 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"  
 tools:context=".MainActivity">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_x="88dp"  
 android:layout\_y="48dp"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_x="176dp"  
 android:layout\_y="96dp"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
</AbsoluteLayout>



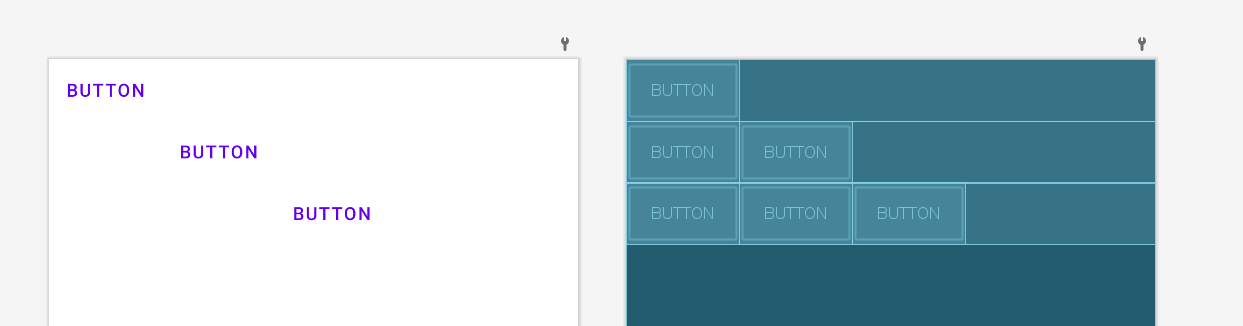
Изображение выглядит как текст

Автоматически созданное описание

Чтобы Android Studio не показывала предупреждения в случае использования AbsoluteLayout, от него нужно просто отказаться, так как он считается устаревшем, начиная с API level 3.

TableLayout

*<?*xml version="1.0" encoding="utf-8"*?>*<TableLayout  
 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"  
 tools:context=".MainActivity">  
  
 <TableRow  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 </TableRow>  
  
 <TableRow  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text"  
 android:visibility="invisible"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 </TableRow>  
  
 <TableRow  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text"  
 android:visibility="invisible"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text"  
 android:visibility="invisible"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button\_text"  
 style="?android:attr/buttonBarButtonStyle" />  
  
 </TableRow>  
  
</TableLayout>



Приведенные выше решения задания из пункта 2 уже содержат код, подвергнутый рефакторингу, поэтому Android Studio не показывает ни одного предупреждения, конечно, кроме случая с использованием AbsoluteLayout.