# 3. View, Styles, Support library основы.

## View, ViewGroup, Attributes

Activity - один экран с пользовательским интерфейсом. Активити имеет свой жизненный цикл, о котором будет в следующих уроках.

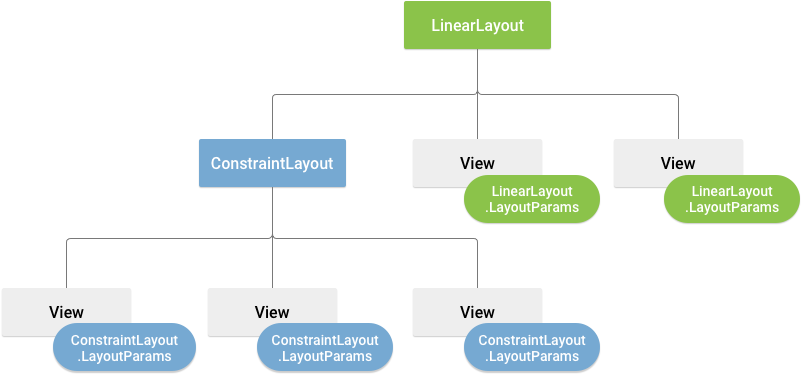
* View - базовый класс.
* ViewGroup - базовый для всех, которые могут иметь детей.
* setContentView(R.layout.main\_layout);
* Атрибуты (textSize)
* Идентификатор android:id="@+id/my\_button", android:id="@android:id/empty"

<Button android:id="@+id/my\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/my\_button\_text"/>



Рассказать про базовые вьюхи

### 

<https://o7planning.org/ru/10423/cache/images/i/1189616.png>

## Советы по производительности

* Избегай глубоких вложенностей. Меньше вложенность - тем быстрее рендерится экран, тем меньше тормозов. Но соблюдай баланс между читаемостью и маленькой вложенностью
* Грамотно используйте возможности стандартных лейаутов. Лучше использовать TextView c свойством drawableLeft, чем положить в LinearLayout TextView и ImageView
* Переиспользуй верстку

## Переиспользование view

Если элементы верстки встречаются более, чем в одном месте - имеет смысл вынести их в отдельный файл.

<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="@color/titlebar\_bg"

tools:showIn="@layout/activity\_main" >

<ImageView android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:src="@drawable/gafricalogo" />

</FrameLayout>

include tag:

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@color/app\_bg"

android:gravity="center\_horizontal">

**<include layout="@layout/titlebar"/>**

<TextView android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="@string/hello"

android:padding="10dp" />

...

</LinearLayout>

Переопределение параметров:

<include android:id="@+id/news\_title"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

layout="@layout/title"/>

merge tag:

**test.xml**

<merge xmlns:android="http://schemas.android.com/apk/res/android">

<Button

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:text="@string/add"/>

<Button

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content"

android:text="@string/delete"/>

</merge>

Теперь это можно вставить в layout:

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@color/app\_bg"

android:gravity="center\_horizontal">

**<include layout="@layout/**test**"/>**

<TextView android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="@string/hello"

android:padding="10dp" />

...

</LinearLayout>

Предложить изучить ViewStub самостоятельно

<https://developer.android.com/training/improving-layouts/loading-ondemand.html#ViewStub>

## Style and Themes

Стили - набор атрибутов

Темы - набор стилей

Для чего нужно?

<TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:textColor="#00FF00"  
 android:typeface="monospace"  
 android:textSize="18sp"  
 android:text="@string/hello" />

### res/values/styles.xml <?xml version="1.0" encoding="utf-8"?> <resources> <style name="MyTextStyle" parent="@style/Text"> <item name="android:textSize">18sp</item> <item name="android:textColor">#00FF00</item> <item name="android:typeface">monospace</item> </style> </resources>

Тогда в файле разметки теперь будет так:

<TextView  
 style="@style/MyTextStyle"  
 ...  
 android:text="@string/hello" />

**Refactor | Extract | Style**

## Наследование стилей

**Наследование от стилей, предоставляемых андроидом:**

**<style name="GreenText" parent="@android:style/TextAppearance">  
 <item name="android:textColor">#00FF00</item>  
</style>**

**Наследование от своих собственных стилей:**

**<style name="MyTextStyle.Red">  
 <item name="android:textColor">#FF0000</item>  
</style>**

## Использование стилей и тем платформы

**<EditText  
 id="@+id/edit"  
 android:textColor:="?android:textColorSecondary" />**

**Знак ? применяется для поиска значения стиля в текущей теме, а подстрока ?android означает поиск значения стиля в системной теме Android.**

## Menu

Рассказать про actionBar, про toolbar.

В общем понимании menu - всплывающая/выпадающая панель действий, которая появляется при клике/долгом нажатии.

Чтобы определить меню, создайте файл XML в папке res/menu/ вашего проекта и постройте меню со следующими элементами:

<menu>

Определяет класс [Menu](https://developer.android.com/reference/android/view/Menu.html?hl=ru), который является контейнером для пунктов меню. Элемент <menu> должен быть корневым узлом файла, в котором может находиться один или несколько элементов <item> и <group>.

<item>

Создает класс [MenuItem](https://developer.android.com/reference/android/view/MenuItem.html?hl=ru), который представляет один пункт меню. Этот элемент может содержать вложенный элемент <menu> для создания вложенных меню.

<group>

Необязательный, невидимый контейнер для элементов &lt;item&gt;. Он позволяет разделять пункты меню на категории и назначать им одинаковые свойства, такие как активное состояние и видимость. Подробные сведения изложены в разделе [Создание групп меню](https://developer.android.com/guide/topics/ui/menus.html?hl=ru#groups).

Вот пример меню с именем game\_menu.xml:

<?xml version="1.0" encoding="utf-8"?>

<menu xmlns:android="http://schemas.android.com/apk/res/android">

<item android:id="@+id/new\_game"

android:icon="@drawable/ic\_new\_game"

android:title="@string/new\_game"

android:showAsAction="ifRoom"/>

<item android:id="@+id/help"

android:icon="@drawable/ic\_help"

android:title="@string/help" />

</menu>

Мы будем использовать

toolbar.inflateMenu(...)

toolbar.getMenu().findItemById(..).setOnMenuItemClickListener(..)

## 

## Material Design theme, АppCompat

<!-- Base application theme. -->

<style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">

<!-- Customize your theme here. -->

<item name="colorPrimary">@color/colorPrimary</item>

<item name="colorPrimaryDark">@color/colorPrimaryDark</item>

<item name="colorAccent">@color/colorAccent</item>

<item name="colorControlNormal">@color/textColorPrimary</item>

<item name="android:textColorPrimary">@color/textColorPrimary</item>

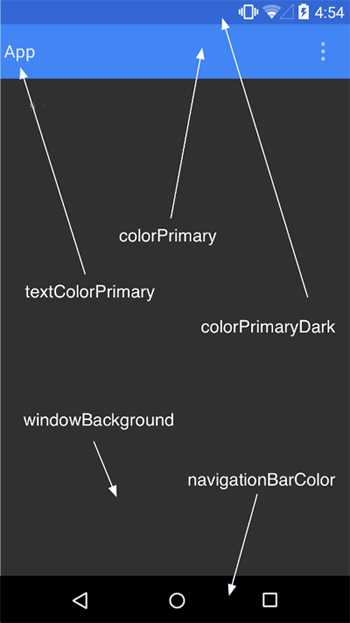
<item name="android:windowBackground">@color/white</item>

<item name="android:colorBackground">@color/white</item>

<item name="windowActionBar">false</item>

<item name="windowNoTitle">true</item>

</style>



<https://developer.android.com/guide/topics/ui/look-and-feel/themes.html>

#### Полезные ссылки:

1. введение про верстку - <https://developer.android.com/guide/topics/ui/declaring-layout.html>
2. базовые лейауты - <https://o7planning.org/ru/10423/android-ui-layouts-tutorial>
3. переиспользование лейаутов -<https://developer.android.com/training/improving-layouts/reusing-layouts.html>
4. view stub - <https://developer.android.com/training/improving-layouts/loading-ondemand.html#ViewStub>
5. <http://startandroid.ru/ru/uroki/vse-uroki-spiskom/15-urok-6-vidy-layouts-kljuchevye-otlichija-i-svojstva.html>
6. <http://startandroid.ru/ru/uroki/vse-uroki-spiskom/38-urok-7-layout-parametry-dlja-view-elementov.html>
7. Стили и темы <http://developer.alexanderklimov.ru/android/theme.php>
8. Стили и темы <https://developer.android.com/guide/topics/ui/look-and-feel/themes.html>
9. Обязательно озакомиться - <https://android-developers.googleblog.com/2014/10/appcompat-v21-material-design-for-pre.html> (на русском - <https://habrahabr.ru/post/241479/> )
10. <https://developer.android.com/training/constraint-layout/index.html>
11. <http://labs.udacity.com/images/Common-Android-Views-Cheat-Sheet.pdf>
12. Меню - <https://developer.android.com/guide/topics/ui/menus.html?hl=ru>
13. ConsraintLayout <https://developer.android.com/training/constraint-layout/>

#### Задание

1. Сконфигурировать тему с support compat theme

<!-- Base application theme. -->

<style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">

<!-- Customize your theme here. -->

<item name="colorPrimary">@color/colorPrimary</item>

<item name="colorPrimaryDark">@color/colorPrimaryDark</item>

<item name="colorAccent">@color/colorAccent</item>

<item name="windowActionBar">false</item>

<item name="windowNoTitle">true</item>

</style>

Добавить implementation 'com.android.support:design:27.1.1'

Пример базовой верстки:

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<android.support.v7.widget.Toolbar

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

app:navigationIcon="установить"

app:title="установить">

</android.support.v7.widget.Toolbar>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="16dp"

android:background="#ff0000"

></RelativeLayout>

</android.support.design.widget.AppBarLayout>

<android.support.v4.widget.NestedScrollView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<android.support.design.widget.TextInputLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_margin="16dp"

android:clickable="false"

android:focusable="false">

<android.support.design.widget.TextInputEditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:background="@null"

android:clickable="false"

android:focusable="false"

android:hint="установить"

android:text="установить"/>

</android.support.design.widget.TextInputLayout>

</android.support.v4.widget.NestedScrollView>

</LinearLayout>

3. Все кнопки должны выводить Toast. (карандаши, к клубу петровичей, выход)

4. Все данные о пользователе и баллах подставлять из джава модельки через java код, а не через xml

Продвинутый уровень - сделать схлопывающийся toolbar <https://antonioleiva.com/collapsing-toolbar-layout/>

и для блока с картой использовать constraint layout

Для фона “Выйти из аккаунта” и карандашика использовать android:background="?attr/selectableItemBackground"