widget 英[ˈwɪdʒɪt]

美[ˈwɪdʒɪt]

n. 小器具，装饰品，窗口小部件;

# TASK

修改SearchView的下划线颜色

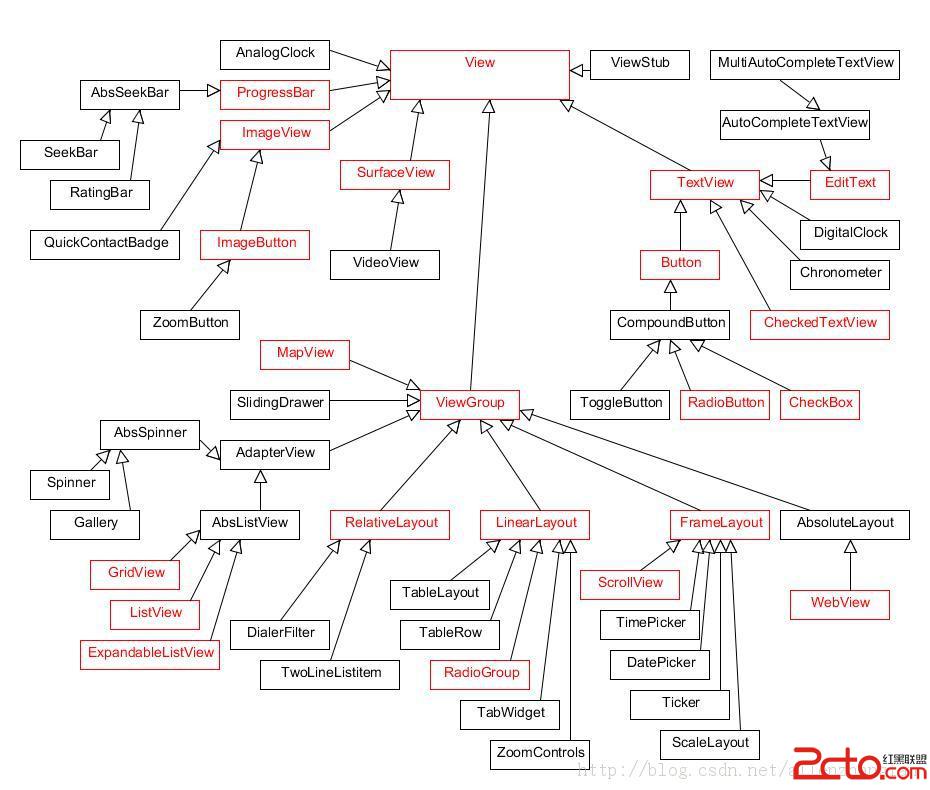
公共控件UI定制原理

<https://blog.csdn.net/weixin_36039900/article/details/79409800>

https://blog.csdn.net/ngl272/article/details/70199992

# View

## view及其子类的关系



# animated-selector

android5.0开始支持的动效了

android:state\_checked

android:state\_enabled=

<https://blog.csdn.net/qq_33689414/article/details/51712297>

# nine-patch

# tint

Tint的存在一定程度上减少了我们对图片的需求以及apk的大小，我们拿ImageView来说吧，假如它的背景图有两种，一种是默认情况下需要显示的是背景图片1，另一种是是在触摸模式下单击时需要显示的是背景图片2。一般情况下背景图片1和背景图片2之间除了颜色不一样，其他都一样的。我们之前的做法也许会找UI要这样两张颜色不一样的图片，但是如果我们使用Tint的话，一张矢量图是能适配所有的颜色

我们仅需要设置tint这个属性即可，android:tint=”@color”来改变颜色

## 改变EditText的背景色以及光标的颜色

https://blog.csdn.net/qq\_20785431/article/details/52106012

## tintMode

https://blog.csdn.net/zenglanjing/article/details/54603242

Android Tint的使用及源码分析

<https://blog.csdn.net/u013085697/article/details/70055621>

# EditText

**public** EditText(Context context, AttributeSet attrs) {  
 **this**(context, attrs, com.android.internal.R.attr.***editTextStyle***);  
}

# **水波纹效果**

**<ripple>是API 21才有的新Tag,正是实现水波纹效果的;**

 其中<ripple android:color="#FF21272B" .... />这个是指定水波纹的颜色.

    而<item />里面的东西,我们都很熟悉,就是普通的定义一个带圆角的背景.

ripple\_bg.xml:

**[html]** [view plain](https://blog.csdn.net/cxc19890214/article/details/51605719) [copy](https://blog.csdn.net/cxc19890214/article/details/51605719)

1. **<?xml** version="1.0" encoding="utf-8"**?>**
2. **<ripple** xmlns:android="http://schemas.android.com/apk/res/android"
3. android:color="#FF21272B"**>**
4. **<item>**
5. **<shape** android:shape="rectangle"**>**
6. **<solid** android:color="#FFFFFF" **/>**
7. **<corners** android:radius="4dp" **/>**
8. **</shape>**
9. **</item>**
10. **<item** android:drawable="@drawable/rounded\_corners" **/>**
11. **</ripple>**

# Switch

## switchStyle

**public** Switch(Context context, AttributeSet attrs) {  
 **this**(context, attrs, com.android.internal.R.attr.***switchStyle***);  
}

## Switch\_track

**final** TypedArray a = context.obtainStyledAttributes(  
 attrs, com.android.internal.R.styleable.***Switch***, defStyleAttr, defStyleRes);

**mTrackDrawable** = a.getDrawable(com.android.internal.R.styleable.***Switch\_track***);

# Spinner

<**style name="Widget.Spinner"**>  
 <**item name="background"**>@drawable/btn\_dropdown</**item**>  
 <**item name="clickable"**>true</**item**>  
 <**item name="spinnerMode"**>dialog</**item**>  
  
 <**item name="dropDownSelector"**>@drawable/list\_selector\_background</**item**>  
 <**item name="popupBackground"**>@drawable/spinner\_dropdown\_background</**item**>  
 <**item name="dropDownVerticalOffset"**>-10dip</**item**>  
 <**item name="dropDownHorizontalOffset"**>0dip</**item**>  
 <**item name="dropDownWidth"**>wrap\_content</**item**>  
 <**item name="popupPromptView"**>@layout/simple\_dropdown\_hint</**item**>  
 <**item name="gravity"**>center</**item**>  
</**style**>

<**style name="Widget.Material.Spinner" parent="Widget.Spinner.DropDown"**>  
 <**item name="background"**>@drawable/spinner\_background\_material</**item**>  
 <**item name="dropDownSelector"**>?attr/listChoiceBackgroundIndicator</**item**>  
 <**item name="popupBackground"**>@drawable/popup\_background\_material</**item**>  
 <**item name="popupElevation"**>@dimen/floating\_window\_z</**item**>  
 <**item name="popupAnimationStyle"**>@empty</**item**>  
 <**item name="popupEnterTransition"**>@transition/popup\_window\_enter</**item**>  
 <**item name="popupExitTransition"**>@transition/popup\_window\_exit</**item**>  
 <**item name="dropDownVerticalOffset"**>0dip</**item**>  
 <**item name="dropDownHorizontalOffset"**>0dip</**item**>  
 <**item name="overlapAnchor"**>true</**item**>  
 <**item name="dropDownWidth"**>wrap\_content</**item**>  
 <**item name="popupPromptView"**>@layout/simple\_dropdown\_hint</**item**>  
 <**item name="gravity"**>start|center\_vertical</**item**>  
 <**item name="disableChildrenWhenDisabled"**>true</**item**>  
</**style**>

## popupPromptView

<**item name="popupPromptView"**>@layout/simple\_dropdown\_hint</**item**>

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

android:id="@android:id/text1"

android:textAppearance="?android:attr/dropDownHintAppearance"

android:singleLine="true"

android:layout\_marginStart="3dip"

android:layout\_marginTop="3dip"

android:layout\_marginEnd="3dip"

android:layout\_marginBottom="3dip"

android:layout\_width="match\_parent"

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

[android](https://www.pocketdigi.com/tag/android).R.layout.simple\_spinner\_item

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

android:id="@android:id/text1"

style="?android:attr/spinnerItemStyle"

android:singleLine="true"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:ellipsize="marquee"

android:textAlignment="inherit"/>

spinnerItemStyle

# checkbox

<item name="listChoiceIndicatorMultiple">@drawable/btn\_check\_material\_anim</item>

btn\_check\_material\_anim.xml

btn\_checkbox\_checked\_mtrl.xml

# Alert

## QA 如何改为最大呢

**Theme.DeviceDefault.Settings.Dialog.Alert**

<**style name="Theme.DeviceDefault.Settings.Dialog.Alert" parent="Theme.Material.Settings.Dialog.Alert"**>  
 *<!-- Color palette -->* <**item name="colorPrimary"**>@color/primary\_device\_default\_settings</**item**>  
 <**item name="colorPrimaryDark"**>@color/primary\_dark\_device\_default\_settings</**item**>  
 <**item name="colorSecondary"**>@color/secondary\_device\_default\_settings</**item**>  
 <**item name="colorAccent"**>@color/accent\_device\_default\_light</**item**>  
</**style**>

<**style name="Theme.Material.Settings.Dialog.Alert" parent="Theme.Material.Settings.Dialog.BaseAlert"** />

**style name="Theme.Material.Settings.Dialog.BaseAlert" parent="Theme.Material.Light.Dialog.BaseAlert"**>  
 <**item name="colorPrimary"**>@color/primary\_material\_settings</**item**>  
 <**item name="colorPrimaryDark"**>@color/primary\_dark\_material\_settings</**item**>  
 <**item name="colorSecondary"**>@color/secondary\_material\_settings</**item**>  
</**style**>

<**style name="Theme.Material.Light.Dialog" parent="Theme.Material.Light.BaseDialog"**/>

<**style name="Theme.Material.Light.BaseDialog"**>  
 <**item name="colorBackground"**>?attr/colorBackgroundFloating</**item**>  
  
 <**item name="windowFrame"**>@null</**item**>

<**item name="windowTitleStyle"**>@style/DialogWindowTitle.Material.Light</**item**>  
 <**item name="windowTitleBackgroundStyle"**>@style/DialogWindowTitleBackground.Material.Light</**item**>  
 <**item name="windowBackground"**>@drawable/dialog\_background\_material</**item**>  
 <**item name="windowElevation"**>@dimen/floating\_window\_z</**item**>  
 <**item name="windowIsFloating"**>true</**item**>  
 <**item name="windowContentOverlay"**>@null</**item**>  
 <**item name="windowAnimationStyle"**>@style/Animation.Material.Dialog</**item**>  
 <**item name="windowSoftInputMode"**>stateUnspecified|adjustPan</**item**>  
 <**item name="windowActionBar"**>false</**item**>  
 <**item name="windowActionModeOverlay"**>true</**item**>  
 <**item name="windowCloseOnTouchOutside"**>@bool/config\_closeDialogWhenTouchOutside</**item**>  
  
 <**item name="colorBackgroundCacheHint"**>@null</**item**>  
  
 <**item name="listPreferredItemPaddingLeft"**>?attr/dialogPreferredPadding</**item**>  
 <**item name="listPreferredItemPaddingRight"**>?attr/dialogPreferredPadding</**item**>  
 <**item name="listPreferredItemPaddingStart"**>?attr/dialogPreferredPadding</**item**>  
 <**item name="listPreferredItemPaddingEnd"**>?attr/dialogPreferredPadding</**item**>  
  
 <**item name="listDivider"**>@null</**item**>  
  
 <**item name="preferencePanelStyle"**>@style/PreferencePanel.Dialog</**item**>  
</**style**>

http://blog.supenta.com/2014/07/02/how-to-style-alertdialogs-like-a-pro/