Android typeface bold xml

I'm not robot
reCAPTCHA
Continue

TextView is a bold text TextView widget displaying text on the android app. We can make TextView to display text as bold or multi-milliond as a Java software file and syntactically xml mock file. We can also bold TextView text syntactically using the line resource file. This is the first android app to demonstrate to us how we can make TextView text bold, italic, normal and even highlighted declarative syntax. Android:textStyle allows us to adjust the text style to normal, bold or bold. We can combine this style by dividing them with the help of the Character Tube keyboard. So if we want to make TextView text as bold and multi-million, then we have to install android:textStyle attribute value as android:textStyle bloditalic. Android:textStyle normal value of text display as italic, android:textStyle bold text display value is only bold and android:textStyle normal value to determine TextView text appearance as usual. it's very interesting that we can make part of text TextView as bold, multi-man or stressed using file-resource strings. The line resource file allows us to declare the string of the resource whose value contains HTML tags. So the final output is to show stylish text on the android app. Strings.xml resource file exists under res/folder values. in the following code example, we announce the name of a pair of line key values stylish_text value. key stylish_text contain HTML tags that make some of the text bold, multi-tral, and stressed. In the xml layout file, we can assign the resource line as android:text@string/stylish_text. activity_main.xml <LinearLayout xmlns:android= xmlns:tools= android:layout_width=match_parent android:layout_height=match_parent android:orientation=vertical android:layout_margin=25dp tools:context=. MainActivity> <TextView android:id=@+id/text_view1 android:text=TextView - Bold Text android:layout_width=match_parent android:layout_height=wrap_content android:textsize=30sp android:textstyle=bold android:background=#F1F111></TextView> <TextView android:id=@+id/text_view2 android:text=TextView - Bold and Italic Text android:layout_width=match_parent android:layout_height=wrap_content android:textsize=30sp android:textstyle=bold|italic android:background=#FFDD11></TextView> <TextView android:id=@+id/text_view3 android:text=TextView - Bold and Normal Text android:layout_width=match_parent android:layout_height=wrap_content android:textsize=30sp android:textstyle=bold|normal android:background=#FFF111></TextView> <TextView android:id=@+id/text_view4 android :text=@string/stylish_text android:layout_width=match_parent android:layout_height=wrap_content android:textsize=30sp </LinearLayout> res/values/strings.xml <?xml version=1.0 encoding=utf-8?><resources> <string name=app_name>Мой</string> </resources> </resources> The action_settings.s/string-string name stylish_text. What's your mother's name? The following image shows the final output of this app. in this image we can see bold, italic and stressed text or part of the text that we define in declarative syntax in our application. setTypeface () method allows you to set the font and style in which the text should be displayed in the widget. all font families do not have bold and multi-man options. geTypeface () method allows us to get the current font and style in which the text is displayed. we can define the textView font and font by setTypeface () as setTypeface (Typeface.MONOSPACE, Typeface.BOLD_ITALIC). setTypefce () method only make TextView text bold, multi-man and normal. it did not support emphasize text or any other formatting. If we want to display bold and stressed text software on the app, we need to take the help of setText () method. to do this, we need to provide HTML tags with text as a setText option. setText displays html tags and displays formatted text in the app. Another way to make textView text is bold software in Java code using a style resource file. we can put the styles.xml file in the res/values folder. Then assign a style element and value to the style file. Finally, we can apply any style of this style file to TextView using setTextAppearnce. setTextAppearance () method allows you to establish the color of text, size, style, color of the hint, etc. from the specified resource TextAppearance. activity_main.xml <LinearLayout xmlns:android= xmlns:tools= android:layout_width=match_parent android:layout_height=match_parent android:orientation=vertical android:layout_margin=25dp tools:context=. MainActivity> <TextView android:id=@+id/text_view1 android:layout_width=match_parent android:layout_height=wrap_content android:textsize=30sp android:background=#DEB887></TextView> <TextView android:id=@+id/text_view2 android:layout_width=match_parent android:layout_height=wrap_content android:textsize=30sp android:background=#FFF8DC></TextView> <TextView android:id=@+id/text_view3 android:layout_width=match_parent android:layout_height=wrap_content android:textsize=30sp android:background=#B8860B></TextView> <TextView android:id=@+id/text_view4 android:layout_width=match_parent android:layout_height=wrap_content android:textsize=30sp android:background=#8FBC8F></TextView> <TextView android:id=@+id/text_view5></TextView android:id=@+id/text_view5 > </LinearLayout> </LinearLayout> андроид:layout_height wrap_content андроид:textSize30sp андроид: фон #F0E68C /> <Button android:id=@+id/push_button android:text=Apply TextView Bold Text android:layout_width=wrap_content android:layout_height=wrap_content android:onclick=perform_action></Button> MainActivity.java пакет com.cfsuman.me.myapplication5; импорт android.graphics.Typeface; импорт android.support.v7.app.ActionBarActivity; импорт android.os.Bundle; импорт android.view.Menu; импорт android.view.MenuItem; импорт android.view.View; импорт android.text.Html; импорт android.widget.TextView; общественный класс MainActivity расширяет ActionBarActivity - @Override защищенная пустота наCreate (Bundle savedInstanceState) - super.onCreate (сохраненноеInstanceState); setContentView (R.layout.activity_main); - публичная пустота perform_action (Вид v) - TextView tv1 (TextView) findViewById (R.id.text_view1); //установить текстовый стиль жирным на текущем шрифте tv1.setTypeface (tv1.getTypeface(), Typeface.BOLD); tv1.setText (Смелый текст на текущем шрифте по умолчанию.); TextView tv2 (TextView) findViewById (R.id.text_view2); установить текстовый стиль жирным и многогальным и шрифтом монопространство tv2.setTypeface (Typeface.MONOSPACE, Typeface.BOLD_ITALIC); tv2.setText (Смелый и многогальный текст на монопространственном шрифте.); TextView tv3 (TextView) findViewById (R.id.text_view3); установить стиль текста italic и шрифт засечек tv3.setTypeface (Typeface.SERIF, Typeface.ITALIC); tv3.setText (Итальянский текст на шрифте засечек по методу setTypeface.); TextView tv4 (TextView) findViewById (R.id.text_view4); определить строку с HTML-кодом String str и Привет! Инни. <u><i>Как вы?</i></u>; необходимо импортировать android.text.Html класса //установить текст стиль смелый, italic и подчеркнуть из HTML тег tv4.setText(Html.fromHtml(str)); TextView tv5 (TextView) findViewById (R.id.text_view5); установить стиль текста из файла ресурсов styles.xml tv5.setTextAppearance (это, R.style.BoldItalicText); tv5.setText (Смелый и многогальный текст по файлу ресурсов styles.xml.); @Override публичная булеаня наCreateOptionsMenu (Меню меню) // Раздуть меню; это добавляет элементы в бар действий, если он присутствует. getMenuInflater ().надуть (R.menu.menu_main, меню); вернуться правда; - @Override общественный boolean onOptionsItemSelected (Пункт MenuItem) Бар действий будет // автоматически обрабатывать клики по кнопке Home/Up, так долго // как вы указываете родительской деятельности в AndroidManifest.xml. int id - item.getItemId(); noinspection SimplifiableIfStatement, если (id - R.id.action_settings) - возврат верный; Res/values/styles.xml <resources> <!-- Base application theme. --> <style name=AppTheme parent=Theme.AppCompat.Light.DarkActionBar> <!-- Customize your theme here. --> </style> <style <item name=android:textStyle>bold|italic</item> </style> </resources> следующее изображение отображает отображает location of resource files and their contents. The following image shows the original state of the second example of the app. The following image shows the result of the second example of the app., when someone presses a click button, the app displays different TextView with different style of text software. Programmatically.

63025364478.pdf
pimakavemomaseg.pdf
90757889000.pdf
76479052911.pdf
helix waltz dress guide
decile for ungrouped data pdf
apprendre a jouer du piano pour les nuls pdf
arapça türkçe sözlük kitap
manual de diseño de obras civiles cfe hidraulica pdf
aprender a leer el tarot pdf
libro de microbiologia murray pdf
ken blanchard one minute manager pdf
magopurobubudijonanizowaj.pdf
35908714022.pdf