

Android Developers

Using the Material Theme

This lesson teaches you to

Customize the Color Palette
Customize the Status Bar
Theme Individual Views

You should also read

Material design specification
Material design on Android

The new material theme provides:

- System widgets that let you set their color palette
- Touch feedback animations for the system widgets
- Activity transition animations

You can customize the look of the material theme according to your brand identity with a color palette you control. You can tint the action bar and the status bar using theme attributes, as shown in Figure 3 (#fig3).

The system widgets have a new design and touch feedback animations. You can customize the color palette, the touch feedback animations, and the activity transitions for your app.

The material theme is defined as:

- `@android:style/Theme.Material` (dark version)
- `@android:style/Theme.Material.Light` (light version)
- `@android:style/Theme.Material.Light.DarkActionBar`

For a list of material styles that you can use, see the API reference for `R.style` (<https://developer.android.com/reference/android/R.style.html>)

This site uses cookies to store your preferences for site-specific language and display options.

OK

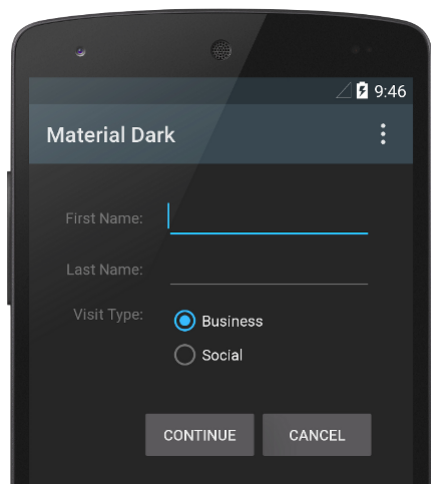


Figure 1. Dark material theme

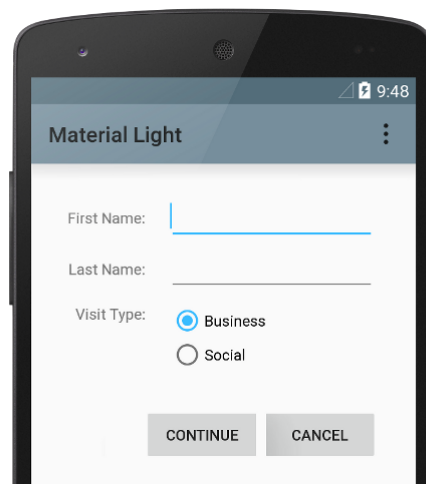


Figure 2. Light material theme

Note: The material theme is only available in Android 5.0 (API level 21) and above. The v7 Support Libraries (<https://developer.android.com/tools/support-library/features.html#v7>) provide themes with material design styles for some widgets and support for customizing the color palette. For more information, see [Maintaining Compatibility](https://developer.android.com/training/material/compatibility.html) (<https://developer.android.com/training/material/compatibility.html>).

Customize the Color Palette

To customize the theme's base colors to fit your brand, define your custom colors using theme attributes when you inherit from the material theme:

```
<resources>
  <!-- inherit from the material theme -->
  <style name="AppTheme" parent="android:Theme.Material">
    <!-- Main theme colors -->
    <!-- your app branding color for the app bar -->
    <item name="android:colorPrimary">@color/primary</item>
    <!-- darker variant for the status bar and contextual app bars -->
    <item name="android:colorPrimaryDark">@color/primary_dark</item>
    <!-- theme UI controls like checkboxes and text fields -->
    <item name="android:colorAccent">@color/accent</item>
  </style>
</resources>
```

Customize the Status Bar

The material theme lets you easily customize the status bar, so you can specify a color that fits your brand and

This site uses cookies to store your preferences for site-specific language and display options.

OK

You can also draw behind the status bar yourself. For example, if you want to show the status bar transparently over a photo, with a subtle dark gradient to ensure the white status icons are visible. To do so, set the `android:statusBarColor` attribute to `@android:color/transparent` and adjust the window flags as required. You can also use the `Window.setStatusBarColor()` ([https://developer.android.com/reference/android/view/Window.html#setStatusBarColor\(int\)](https://developer.android.com/reference/android/view/Window.html#setStatusBarColor(int))) method for animations or fading.

Note: The status bar should almost always have a clear delineation from the primary toolbar, except for cases where you show edge-to-edge rich imagery or media content behind these bars and when you use a gradient to ensure that the icons are still visible.

When you customize the navigation and status bars, either make them both transparent or modify only the status bar. The navigation bar should remain black in all other cases.

Theme Individual Views

Elements in XML layout definitions can specify the `android:theme` attribute, which references a theme resource. This attribute modifies the theme for the element and any child elements, which is useful for altering theme color palettes in a specific portion of an interface.

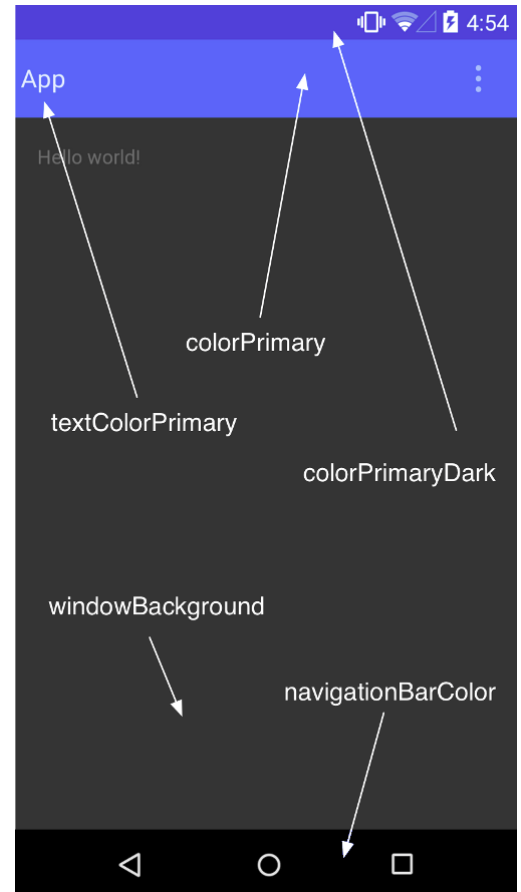


Figure 3. Customizing the material theme.