

# Style Resource

A style resource defines the format and look for a UI. A style can be applied to an individual [View](/reference/android/view/View.html) (</reference/android/view/View.html>) (from within a layout file) or to an entire [Activity](/reference/android/app/Activity.html) (</reference/android/app/Activity.html>) or application (from within the manifest file).

## SEE ALSO

[Styles and Themes](#)

For more information about creating and applying styles, please read [Styles and Themes](/guide/topics/ui/themes.html) (</guide/topics/ui/themes.html>).

**Note:** A style is a simple resource that is referenced using the value provided in the name attribute (not the name of the XML file). As such, you can combine style resources with other simple resources in the one XML file, under one `<resources>` element.

## FILE LOCATION:

`res/values/filename.xml`

The filename is arbitrary. The element's name will be used as the resource ID.

## RESOURCE REFERENCE:

In XML: `@[package:]style/style_name`

## SYNTAX:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <style
    name="style_name"
    parent="@[package:]style/style_to_inherit">
    <item
      name="[package:]style_property_name"
      >style_value</item>
    </style>
</resources>
```

## ELEMENTS:

`<resources>`

**Required.** This must be the root node.

No attributes.

`<style>`

Defines a single style. Contains `<item>` elements.

attributes:

name

*String. Required.* A name for the style, which is used as the resource ID to apply the style to a View, Activity, or application.

parent

*Style resource.* Reference to a style from which this style should inherit style properties.

`<item>`

Defines a single property for the style. Must be a child of a `<style>` element.

attributes:

name

*Attribute resource. Required.* The name of the style property to be defined, with a package prefix if necessary (for example `android:textColor`).

EXAMPLE:

XML file for the style (saved in `res/values/`):

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <style name="CustomText" parent="@style/Text">
    <item name="android:textSize">20sp</item>
    <item name="android:textColor">#008</item>
  </style>
</resources>
```

XML file that applies the style to a TextView (saved in `res/layout/`):

```
<?xml version="1.0" encoding="utf-8"?>
<EditText
  style="@style/CustomText"
  android:layout_width="fill_parent"
  android:layout_height="wrap_content"
  android:text="Hello, World!" />
```