

**Android Basic**

**(Part 7)**

Họ và tên : Trần Đức Lâm

Trường : Học viện công nghệ Bưu chính Viễn thông

Staff-code: INS046

Khóa: Open Education 12.

Đơn vị : Framgia

Hà Nội – 2018

1. Stype & Theme

Stype & Theme trên Android cho phép bạn tách riêng chi tiết về thiết kế ứng dụng của bạn khỏi cấu trúc và hành vi giao diện người dùng, tương tự như biểu định kiểu trong thiết kế web.

Một Stype là một tập hợp các thuộc tính xác định sự xuất hiện cho một single [View](https://developer.android.com/reference/android/view/View.html). Kiểu có thể chỉ định các thuộc tính như màu phông chữ, kích thước phông chữ, màu nền và nhiều thứ khác.

Một Theme là một loại phong cách được áp dụng với toàn bộ phân cấp ứng dụng, hoạt động, hoặc xem, không chỉ là một cái nhìn cá nhân. Khi bạn áp dụng kiểu của mình làm chủ đề, mỗi chế độ xem trong ứng dụng hoặc hoạt động sẽ áp dụng từng thuộc tính kiểu mà nó hỗ trợ. Chủ đề cũng có thể áp dụng kiểu cho các phần tử không xem, chẳng hạn như thanh trạng thái và nền cửa sổ.

Stype & Theme được khai báo trong một [tập tin tài nguyên phong cách](https://developer.android.com/guide/topics/resources/style-resource.html) trong res/values/, thường được đặt tên styles.xml.

1. Tạo và áp dụng Stype

Để tạo Stype hoặc Theme mới, hãy mở tệp dự án của bạn res/values/styles.xml . Đối với mỗi kiểu bạn muốn tạo, hãy làm theo các bước sau:

* Thêm <style>phần tử có tên xác định duy nhất kiểu.
* Thêm một <item>phần tử cho mỗi thuộc tính kiểu mà bạn muốn xác định.

Trong namemỗi mục xác định một thuộc tính bạn sẽ sử dụng như một thuộc tính XML trong layout của bạn. Giá trị trong <item>phần tử là giá trị cho thuộc tính đó.

Ví dụ :

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

<resources>

<style name="CustomFontStyle">

<item name="android:layout\_width">fill\_parent</item>

<item name="android:layout\_height">wrap\_content</item>

<item name="android:capitalize">characters</item>

<item name="android:typeface">monospace</item>

<item name="android:textSize">12pt</item>

<item name="android:textColor">#00FF00</item>/>

</style>

</resources>

* Cách dùng

Khi Stype của bạn được xác định, bạn có thể sử dụng nó trong tệp XML Layout của bạn bằng **cách** sử dụng thuộc tính **style** như sau:

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

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

android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent"

android:orientation="vertical" >

<TextView

android:id="@+id/text\_id"

style="@style/CustomFontStyle"

android:text="@string/hello\_world" />

</LinearLayout>

1. Áp dụng kiểu làm Theme

Bạn có thể tạo Theme giống như cách bạn tạo kiểu. Sự khác biệt là cách bạn áp dụng: thay vì áp dụng kiểu với style thuộc tính trên chế độ xem, bạn áp dụng chủ đề với android:theme thuộc tính trên <application> thẻ hoặc<activity> thẻ trong AndroidManifest.xmltệp.

Ví dụ :

<manifest ... > <ứng dụng android: theme = "@ style / Theme.AppCompat" ... > </ application> </ manifest>

1. Áp dụng màu sắc cho Theme

Sau đó, tài nguyên màu của bạn có thể được áp dụng cho một số thuộc tính chủ đề, chẳng hạn như nền cửa sổ và màu văn bản chính, bằng cách thêm các phần tử <item> vào chủ đề tùy chỉnh của bạn. Các thuộc tính này được định nghĩa trong tệp styles.xml của bạn. Ví dụ: để áp dụng màu tùy chỉnh cho nền cửa sổ, hãy thêm hai phần tử <item> sau vào chủ đề tùy chỉnh của bạn, được xác định trong tệp MyAndroidApp / res / values ​​/ styles.xml

Ví dụ :

<resources>

...

<style name="MyCustomTheme" ...>

<item name="android:windowBackground">@color/my\_custom\_color</item>

<item name="android:colorBackgroundCacheHint">@color/my\_custom\_color</item>

</style>

...

</resources>

1. Tùy chỉnh các nút

Một bản vá chín bản vá là một loại hình ảnh đặc biệt có thể được chia tỷ lệ theo chiều rộng và chiều cao trong khi duy trì tính toàn vẹn trực quan của nó. Chín bản vá là cách phổ biến nhất để chỉ định giao diện của các nút Android, mặc dù bất kỳ loại có thể kéo nào đều có thể được sử dụng.

Nút vá chín

**MỘT MẪU CỦA NÚT NINE-PATCH**

Các bước để tạo các nút Nine-Patch

* Lưu bitmap này dưới dạng /res/drawable/my\_nine\_patch.9.png
* Xác định kiểu mới
* Áp dụng kiểu nút mới cho thuộc tính buttonStyle của chủ đề tùy chỉnh của bạn

Ví dụ :

<resources>

...

<style name="MyCustomButton" parent="android:Widget.Button">

<item name="android:background">@drawable/my\_nine\_patch</item>

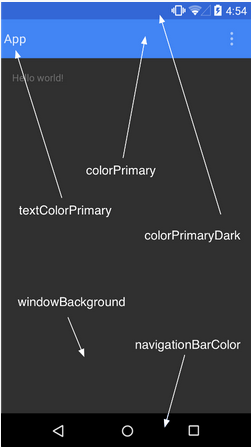
</style>

...

</resources>

1. Tạo kiểu bảng màu

Thiết kế bố cục có thể triển khai dựa trên màu dựa trên chúng, ví dụ như thiết kế sau được thiết kế dựa trên màu sắc (màu xanh)

1. 

Bố cục trên đã được thiết kế dựa trên tệp style.xml, Đã đặt ở vị trí **res / values ​​/**

<resource>

<style name="AppTheme" parent="android:Theme.Material">

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

<item name ="android:color/primaryDark">@color/primary\_dark</item>

<item name ="android:colorAccent/primary">@color/accent</item>

</style>

<resource>