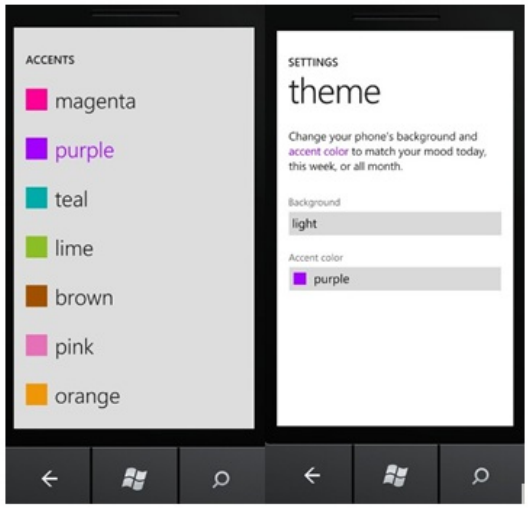
Theme aware Windows Phone application

This article explains how to get and set the current theme background and accent colour.

Introduction

Windows Phone themes are very basic, consisting only of whether the background is light or dark (default), and an accent color.

Windows Phone defines ten accent colors: magenta, purple, teal, lime, brown, pink, orange, blue, red, green (Lumia devices also include the accent color "Nokia Blue"). Note that Application Bar button icons should have a transparent back-ground and white foreground so that they are theme compatible.



Accent color and background settings dialogs

Supporting themes and accent color is important for the following reasons:

- Application look and feel is one of the important aspects along with its use and functionality.
- User wants to use your application more and more if it appears visually attractive and goes in the flow of the overall phone theme.
- Windows Phone allows the application look and feel adapt as per the theme and accent color chosen by the user for the phone.
- Theming allows you to convey aspects of your application branding

This article shows how to get the current theme and accent color, and use them in your app.

How to get current theme background

To get theme background, you can use `PhoneDarkThemeVisibility` resource. Its type is System.Windows.Visibility enum

```
using System.Windows

bool DarkThemeUsed ()
{
     return Visibility.Visible ==   (Visibility)Application.Current.Resources["PhoneDarkThemeVisibility"];
}

bool LightThemeUsed()
{
     return Visibility.Visible == (Visibility)Application.Current.Resources["PhoneLightThemeVisibility"];
}

...

 // Write the theme background value.
 if (DarkThemeUsed ())
{
    //Set resources according to dark theme
 }
else
{
     //Set resources according to light theme
 }
```

How to get current accent color

Accent color is given by `PhoneAccentColor` ressource. Its type is System.Windows.Media.Color.

```
 using System.Windows.Media;

Color AccentColor()
```


(C) Copyright Nokia 2013. All rights reserved.

```
{
    return (Color)Application.Current.Resources["PhoneAccentColor"];
}
```

How to use the accent color in XAML

Theme-aware applications work well with the theme and accent set by the user. To make XAML controls "accent-aware", developers must use the `staticResource` definition.

For example:

- To make a `TextBlock` theme-aware use the `Style` property

```
<TextBlock Style="{StaticResource PhoneTextAccentStyle}"/>
```

- To make a `Rectangle` theme-aware use the `Fill` property:

```
<Rectangle Fill="{StaticResource PhoneAccentBrush}" />
```

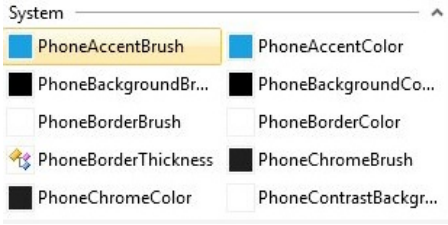
How to set the background in XAML

Similarly, we also use the `staticResource` definition to set the background. In the XAML below we create a button with different background based on `PhoneTheme`:

```
<StackPanel>
 <Button Height="70"  Visibility="{StaticResource PhoneDarkThemeVisibility}"  Width="70" BorderBrush="{x:Null}" Click="button_viewpr
    <Button.Background>
 <ImageBrush ImageSource="/Resources/Images/WhiteArrowImg.png" />
    </Button.Background>
</Button>

<Button Height="70" Margin="-10,0,0,0"  Visibility="{StaticResource PhoneLightThemeVisibility}"   Width="70"  BorderBrush="{x:Null}"
<Button.Background>
   <ImageBrush ImageSource="/Resources/Images/Black_ArrowImg.png" />
   </Button.Background>
</Button>
</StackPanel>
```

Below is the snapshot of Resource Dictionaries which can be accessed in Properties Box for the control.



Test code

A code example/test code can be downloaded from here: Media:ThemeBackgroundAccentColor.zip

The example app simply uses the previous c# function to display current theme background and accent color of the device. A rectangle is filled with accent color.

References

- How to apply theme resources for Windows Phone (Dev Center)
- Theme resources for Windows Phone (Dev Center)

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.