Mastering Styles and Templates

Thomas Claudius Huber
SOFTWARE DEVELOPER

@thomasclaudiush   www.thomasclaudiushuber.com

Module Outline

Styles

- Define property values with a Style
- Inherit a Style from another Style
- Understand explicit and implicit Styles

Templates

- Define the look of a Control
- Create a ControlTemplate for the CheckBox

Demo

Define property values with a Style

Demo

Inherit a Style from another Style

Demo

Understand explicit and implicit Styles

Define the Look of a Custom Control

UserControl

Look can be changed by adjusting the UserControl's XAML file

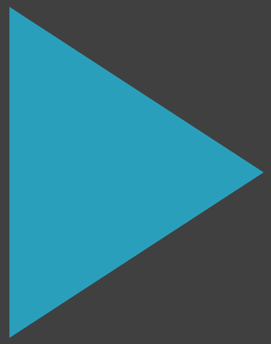
CustomerDetailControl

Custom Control / Templated Control

Look can be changed by assigning a new ControlTemplate to the Control's Template property

Define the Look of a Custom Control

```
<Button Background="LightBlue" Content="Pluralsight">
  <Button.Template>
    <ControlTemplate TargetType="Button">
      <Polygon Points="0,0 30,20 0,40"
               Fill="{TemplateBinding Background}"/>
    </ControlTemplate>
  </Button.Template>
</Button>
```



Define the Look of a Custom Control

```
<Setter Property="Template">
  <Setter.Value>
    <ControlTemplate TargetType="Button">
      <Polygon Points="0,0 30,20 0,40"
               Fill="{TemplateBinding Background}"/>
    </ControlTemplate>
  </Setter.Value>
</Setter>
```

Demo

Create a ControlTemplate for the CheckBox

Summary

Styles

- Define property values with a Style
- Inherit a Style from another Style
- Explicit and implicit Styles

Templates

- Define the look of a Control
- Create a ControlTemplate for the CheckBox