

## Style và ControlTemplate

Giống như sử dụng Cascading Style Sheets (CSS) đối với HTML, việc định nghĩa các giá trị cho các thuộc tính của các đối tượng giao diện một lần, sau đó áp dụng nhiều lần cho các đối tượng khác cùng loại giúp người lập trình giảm công sức viết mã.

WPF cũng cung cấp tính năng tương tự bằng việc sử dụng thành phần **Style** của XAML.

Một Style có thể được dẫn xuất từ một Style khác, thừa kế hoặc chồng lên những thuộc tính đã thiết lập.

Mỗi style có thể định nghĩa các trigger cho phép tạo ra những hiệu ứng tương tác đặc biệt, chẳng hạn như khi lướt chuột qua nút bấm, nút bấm chuyển thành màu vàng.

WPF cũng hỗ trợ sử dụng **Template**.

Mỗi template tương tự như một style, và ở hai dạng:

- Template cho dữ liệu: sử dụng thành phần DataTemplate của XAML để thiết lập một nhóm thuộc tính hiển thị của dữ liệu như màu sắc, phương thức căn lề...
- Template cho control: sử dụng thành phần ControlTemplate của XAML để định ra diện mạo của một control.

Có thể coi **Style** là một cách thuận tiện để áp dụng một tập hợp các giá trị thuộc tính cho nhiều phần tử. Chúng ta có thể thay đổi giao diện mặc định bằng cách đặt các thuộc tính, chẳng hạn như FontSize và FontFamily, trên từng phần tử TextBlock trực tiếp.

Trong khi đó, **ControlTemplate** chỉ định cấu trúc trực quan và hành vi trực quan của một điều khiển. Chúng ta có thể tùy chỉnh giao diện của một điều khiển bằng cách tạo một ControlTemplate mới. Khi tạo ControlTemplate, chúng ta sẽ thay thế giao diện của một điều khiển hiện có mà không thay đổi chức năng của nó. Ví dụ: chúng ta có thể thiết lập các button là tròn thay vì hình vuông mặc định, nhưng nút sẽ vẫn có sự kiện Click.

-----

Trong một **style** chúng ta có thể **thiết lập** các thuộc tính cho một điều khiển:

```
<Style x:Key="MyButtonStyle" TargetType="Button">
    <Setter Property="Background" Value="Red"/>
</Style>
```

Và khi sử dụng:

```
<Button Style="{StaticResource MyButtonStyle}"/>
```

Trong một **template** chúng ta **định nghĩa** một UI (**cấu trúc**) của một điều khiển:

```
<ControlTemplate x:Key="MyButtonTemplate" TargetType="Button">
    <Grid>
        <Rectangle Fill="Green"/>
        <ContentPresenter/>
    </Grid>
</ControlTemplate>
```

```
<Button Template="{StaticResource MyButtonTemplate}"/>
```

Tất cả các button mà sử dụng template này sẽ có màu nền là green và nó không thể thay đổi.

Các giá trị được thiết lập trong một **template** có thể chỉ được thay thế bằng cách thay đổi toàn bộ template.

Các giá trị trong một **style** có thể được thay thế bằng cách đặt giá trị một cách rõ ràng khi sử dụng điều khiển.

Đó là lý do tại sao tốt hơn nên sử dụng các thuộc tính của điều khiển bằng cách sử dụng **TemplateBinding** thay vì giá trị cụ thể.

```
<ControlTemplate x:Key="MyButtonTemplate" TargetType="Button">
    <Grid>
        <Rectangle Fill="{TemplateBinding Background}"/>
        <ContentPresenter/>
    </Grid>
</ControlTemplate>
```

Và như vậy chúng ta có thể đặt một giá trị cụ thể cho Background:

```
<Button Template="{StaticResource MyButtonTemplate}" Background="Yellow"/>
```

Kết hợp Templates và Styles: Chúng ta có thể đặt thuộc tính Template trong một style:

```
<Style TargetType="Button">
    <Setter Property="Background" Value="Red"/>
    <Setter Property="Template">
        <Setter.Value>
            <ControlTemplate TargetType="Button">
                <Grid>
                    <Rectangle Fill="{TemplateBinding Background}"/>
                    <ContentPresenter/>
                </Grid>
            </ControlTemplate>
        </Setter.Value>
    </Setter>
</Style>
```

---

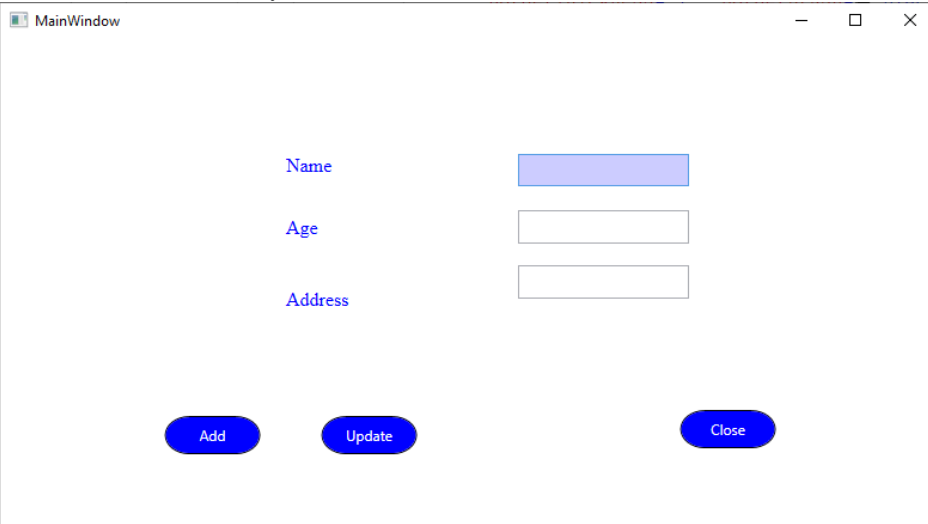
## Bài tập:

Thiết kế giao diện sau đây:



The screenshot shows a window titled "MainWindow" with a standard Windows title bar (minimize, maximize, close buttons). Inside the window, there are three labels: "Name", "Age", and "Address", each followed by a text input field. At the bottom of the window, there are three blue buttons: "Add", "Update", and "Close".

Khi các textbox được focus tới thì:



The screenshot shows the same "MainWindow" form. The "Name" text input field is now focused, indicated by a blue highlight around it. The "Age" and "Address" fields remain unfocused. The "Add", "Update", and "Close" buttons are still at the bottom.

Khi chuột over tới button thì:



The screenshot shows the same "MainWindow" form. The "Name" text input field now contains the text "Nga Đăng". The "Update" button is highlighted with a blue glow, indicating it is being hovered over by the mouse. The "Add" and "Close" buttons remain unchanged.

[illegible]

```

        <VisualBrush.Visual>
            <Border Width="80" Height="32" CornerRadius="20,20,20,20"
                BorderThickness="1" BorderBrush="Black"
                Background="{TemplateBinding Background}" />
            <!--Đặt background là màu cụ thể sẽ không áp dụng isMouseOver cho background-->
        </VisualBrush.Visual>
    </VisualBrush>
</Rectangle.Fill>

</Rectangle>
<ContentPresenter HorizontalAlignment="Center" VerticalAlignment="Center"/>
</Grid>
<ControlTemplate.Triggers>
    <Trigger Property="IsMouseOver" Value="True">
        <Setter Property="Foreground" Value="blue" />
        <Setter Property="FontWeight" Value="ExtraBold"/>
        <Setter Property="Background" Value="#ccf"/>
    </Trigger>
</ControlTemplate.Triggers>
</ControlTemplate>
</Setter.Value>
</Setter>
<Setter Property="Foreground" Value="White"/>
<Setter Property="Background" Value="Blue"/>
</Style>
</Grid.Resources>
<Label Content="Name" HorizontalAlignment="Left" Height="36" Margin="234,91,0,0" VerticalAlignment="Top" Width="151"/>
<Label Content="Age" HorizontalAlignment="Left" Height="36" Margin="234,143,0,0" VerticalAlignment="Top" Width="151"/>
<Label Content="Address" HorizontalAlignment="Left" Height="36" Margin="234,203,0,0" VerticalAlignment="Top" Width="151"/>
<TextBox HorizontalAlignment="Left" Margin="433,96,0,0" Text="" TextWrapping="Wrap" VerticalAlignment="Top" Width="143"
Height="27"/>
<TextBox HorizontalAlignment="Left" Margin="433,143,0,0" Text="" TextWrapping="Wrap" VerticalAlignment="Top" Width="143"
Height="28"/>
<TextBox HorizontalAlignment="Left" Margin="433,189,0,0" Text="" TextWrapping="Wrap" VerticalAlignment="Top" Width="143"
Height="28"/>
<!--Nếu có x:Key khi định nghĩa-->
<!--<Button Style="{StaticResource mybtn}" Content="Add" HorizontalAlignment="Left" Margin="112,313,0,0"
    VerticalAlignment="Top" Width="131" Height="36"/>-->
<Button Content="Add" HorizontalAlignment="Left" Margin="112,313,0,0" VerticalAlignment="Top" Width="131" Height="36"/>
<Button Content="Update" HorizontalAlignment="Left" Margin="243,313,0,0" VerticalAlignment="Top" Width="131" Height="36"/>
<Button Content="Close" HorizontalAlignment="Left" Margin="543,308,0,0" VerticalAlignment="Top" Width="131" Height="36"/>
</Grid>
</Window>

```