# Building a User Interface



Thomas Claudius Huber
Software Developer

@thomasclaudiush www.thomasclaudiushuber.com



#### Module Outline



#### **Know the WPF layout panels**

- Build a user interface with the Grid

Use layout properties to position elements

Set attached properties in

- XAML
- C#

**StackPanel** 

Grid

```
<StackPanel>
    <StackPanel.Children>
        <Rectangle Fill="LightBlue"
            Height="20" Margin="2"/>
            <Rectangle Fill="LightBlue"
            Height="20" Margin="2"/>
            </StackPanel.Children>
</StackPanel>
```



**StackPanel** 

Grid



**StackPanel** 

Grid



**StackPanel** 

Grid

```
<StackPanel Orientation="Horizontal">
    <Rectangle Fill="LightBlue"
      Width="20" Margin="2"/>
      <Rectangle Fill="LightBlue"
      Width="20" Margin="2"/>
      </StackPanel>
```



**StackPanel** Grid Canvas <Grid> <Grid.RowDefinitions> <RowDefinition/> <RowDefinition/> </Grid.RowDefinitions> <Grid.ColumnDefinitions> <ColumnDefinition/> <ColumnDefinition/> </Grid.ColumnDefinitions>

</Grid>



StackPanel

Grid

```
<Grid ShowGridLines="True">
  <Grid.RowDefinitions>
    <RowDefinition/>
    <RowDefinition/>
  </Grid.RowDefinitions>
  <Grid.ColumnDefinitions>
    <ColumnDefinition/>
    <ColumnDefinition/>
  </Grid.ColumnDefinitions>
```



**StackPanel** Grid Canvas <Grid> <Grid.RowDefinitions> <RowDefinition/> <RowDefinition/> </Grid.RowDefinitions> <Grid.ColumnDefinitions> <ColumnDefinition\_Width="40"/> <ColumnDefinition/> </Grid.ColumnDefinitions>





**StackPanel** Grid Canvas <Grid> <Grid.RowDefinitions> <RowDefinition/> <RowDefinition/> </Grid.RowDefinitions> <Grid.ColumnDefinitions> <ColumnDefinition Width="40"/> <ColumnDefinition/> </Grid.ColumnDefinitions> <Rectangle Fill="LightBlue"/>





**StackPanel** Grid <Grid> <Grid.RowDefinitions> <RowDefinition/> <RowDefinition/> </Grid.RowDefinitions> <Grid.ColumnDefinitions> <ColumnDefinition Width="40"/> <ColumnDefinition/> </Grid.ColumnDefinitions> <Rectangle Fill="LightBlue"</pre> Grid.Column="1"/>

</Grid>



**StackPanel** Grid Canvas <Grid> <Grid.RowDefinitions> <RowDefinition/> <RowDefinition/> </Grid.RowDefinitions> <Grid.ColumnDefinitions> <ColumnDefinition Width="40"/> <ColumnDefinition/> </Grid.ColumnDefinitions> <Rectangle Fill="LightBlue"</pre> Grid.Column="1" Grid.Row="1"/>

</Grid>

**StackPanel** Grid Canvas <Grid> <Grid.RowDefinitions> <RowDefinition/> <RowDefinition/> </Grid.RowDefinitions> <Grid.ColumnDefinitions> <ColumnDefinition Width="40"/> <ColumnDefinition/> </Grid.ColumnDefinitions> <StackPanel Grid.Column="1"> </StackPanel>

</Grid>

**StackPanel** 

Grid

Canvas

```
<Canvas>
<Rectangle Fill="LightBlue"
Height="50" Width="50"/>
```

</Canvas>



**StackPanel** 

Grid

Canvas

```
<Canvas>
  <Rectangle Fill="LightBlue"
   Height="50" Width="50"
   Canvas.Left="50"/>
```

</Canvas>



**StackPanel** 

Grid

Canvas

```
<Canvas>
     <Rectangle Fill="LightBlue"
        Height="50" Width="50"
        Canvas.Left="50"
        Canvas.Top="100"/>
```

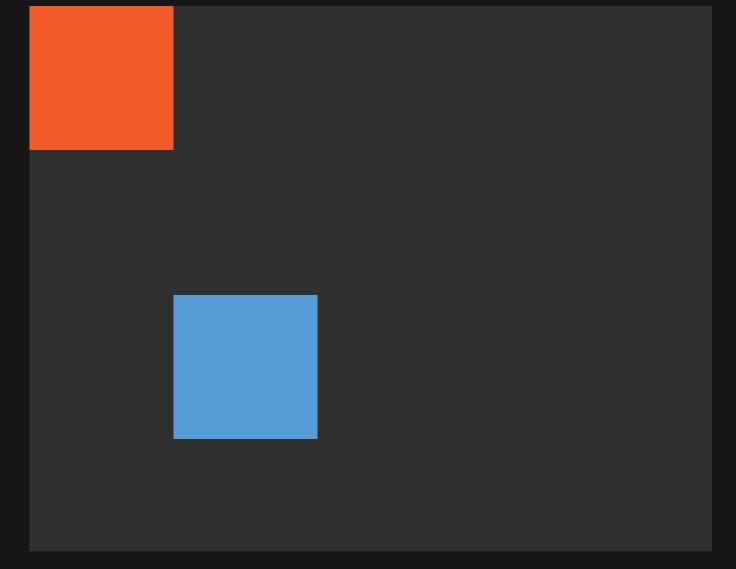
</Canvas>



**StackPanel** 

Grid

```
<Canvas>
  <Rectangle Fill="LightBlue"</pre>
    Height="50" Width="50"
    Canvas.Left="50"
    Canvas.Top="100"/>
  <Rectangle Fill="Orange"</pre>
    Height="50" Width="50"/>
</Canvas>
```





**StackPanel** 

Grid

```
<Canvas>
  <Rectangle Fill="LightBlue"</pre>
    Height="50" Width="50"
    Canvas.Left="50"
    Canvas.Top="100"/>
  <Rectangle Fill="Orange"</pre>
    Height="50" Width="50"
    Canvas.Left="75"/>
</Canvas>
```

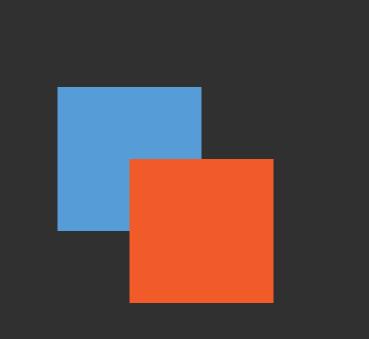




**StackPanel** 

Grid

```
<Canvas>
     <Rectangle Fill="LightBlue"
        Height="50" Width="50"
        Canvas.Left="50"
        Canvas.Top="100"/>
        <Rectangle Fill="Orange"
        Height="50" Width="50"
        Canvas.Left="75"
        Canvas.Top="125"/>
        </Canvas>
```

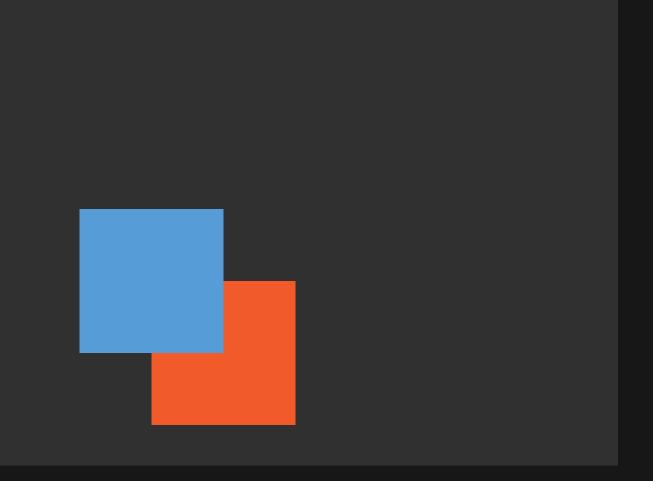




**StackPanel** 

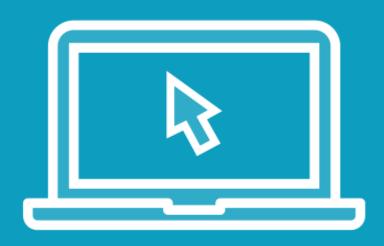
Grid

```
<Canvas>
  <Rectangle Fill="LightBlue"</pre>
    Height="50" Width="50"
    Canvas.Left="50" Panel.ZIndex="1"
    Canvas.Top="100"/>
  <Rectangle Fill="Orange"</pre>
    Height="50" Width="50"
    Canvas.Left="75"
    Canvas.Top="125"/>
</Canvas>
```





### Demo



Build a layout with the Grid



```
<Grid>
     <Grid.RowDefinitions>
          <RowDefinition/>
           <RowDefinition/>
           </Grid.RowDefinitions>

</Grid>
</Grid>
```



```
<Grid>
     <Grid.RowDefinitions>
          <RowDefinition Height="*"/>
          <RowDefinition Height="*"/>
          </Grid.RowDefinitions>

</Grid>
```



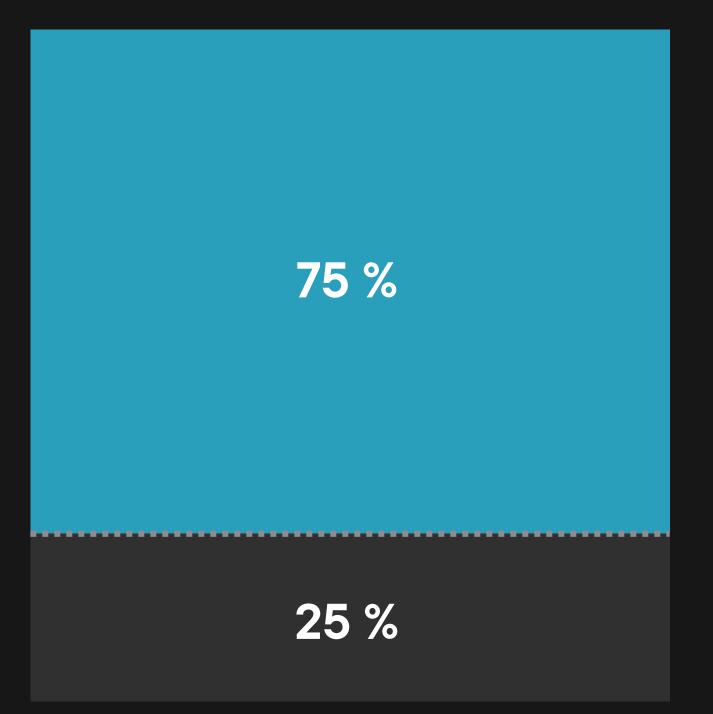
```
<Grid>
     <Grid.RowDefinitions>
          <RowDefinition Height="*"/>
          <RowDefinition Height="*"/>
          </Grid.RowDefinitions>

</Grid>
```



```
<Grid>
     <Grid.RowDefinitions>
          <RowDefinition Height="3*"/>
          <RowDefinition Height="*"/>
          </Grid.RowDefinitions>

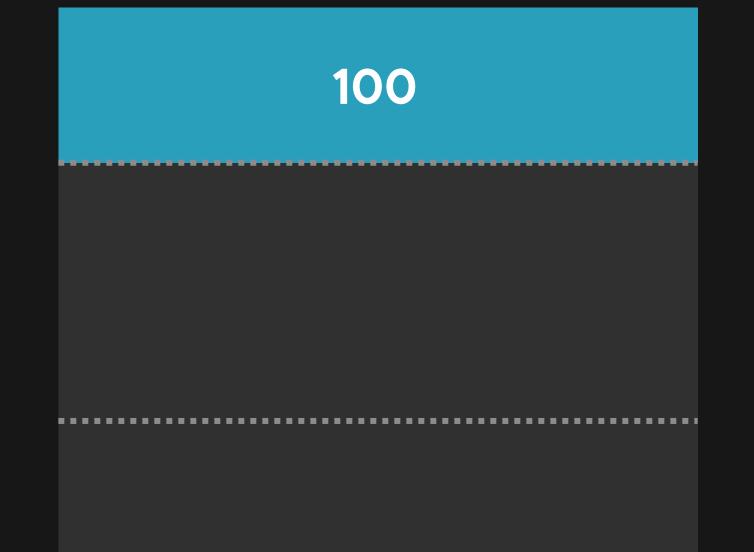
</Grid>
```





100







```
<Grid>
                                                     100
  <Grid.RowDefinitions>
    <RowDefinition Height="100"/>
    <RowDefinition Height="*"/>
    <RowDefinition Height="*"/>
  </Grid.RowDefinitions>
  <Rectangle Fill="LightBlue"/>
  <Rectangle Fill="Orange"</pre>
                                                     50
     Grid.Row="2" Height="50"/>
</Grid>
```



```
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition Height="100"/>
    <RowDefinition Height="*"/>
    <RowDefinition Height="Auto"/>
  </Grid.RowDefinitions>
  <Rectangle Fill="LightBlue"/>
  <Rectangle Fill="Orange"</pre>
     Grid.Row="2" Height="50"/>
</Grid>
```

100

\* leftover space

50



#### Demo



Adjust the Grid in the application

```
<Grid Width="200" Height="200">
    <Button Content="OK"
        HorizontalAlignment="Left"
        VerticalAlignment="Top"/>
        </Grid>
```





```
<Grid Width="200" Height="200">
     <Button Content="OK"
        HorizontalAlignment="Right"
        VerticalAlignment="Top"/>
</Grid>
```



```
<Grid Width="200" Height="200">
     <Button Content="OK"
        HorizontalAlignment="Stretch"
        VerticalAlignment="Top"/>
</Grid>
```



```
<Grid Width="200" Height="200">
    <Button Content="OK"
    HorizontalAlignment="Stretch"
    VerticalAlignment="Center"/>
</Grid>
```



```
<Grid Width="200" Height="200">
     <Button Content="OK"
        HorizontalAlignment="Stretch"
        VerticalAlignment="Bottom"/>
     </Grid>
```



OK

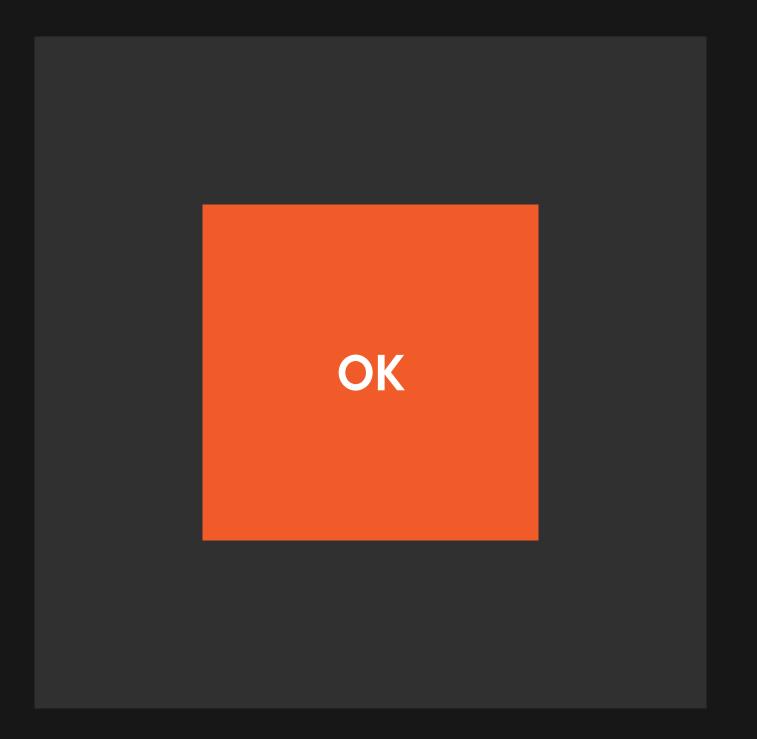




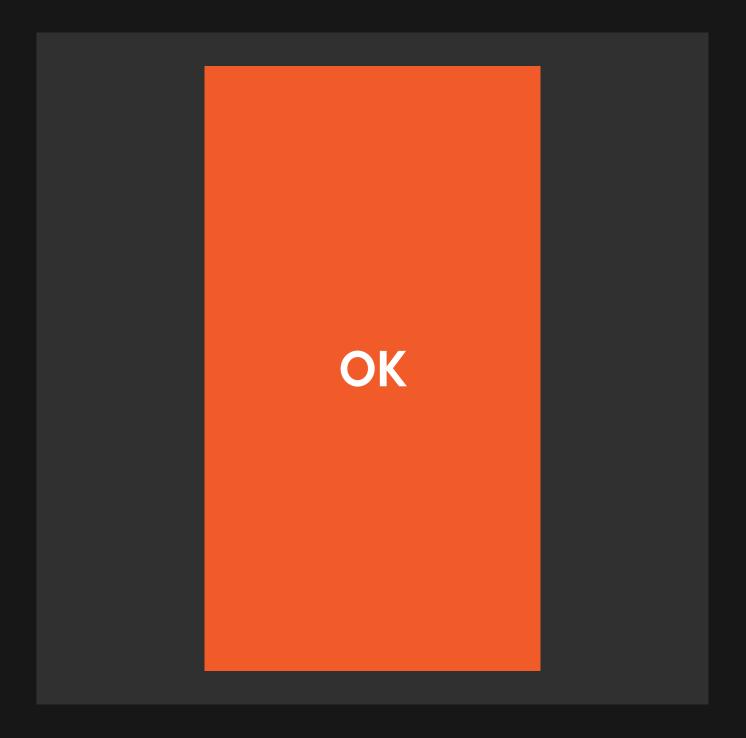




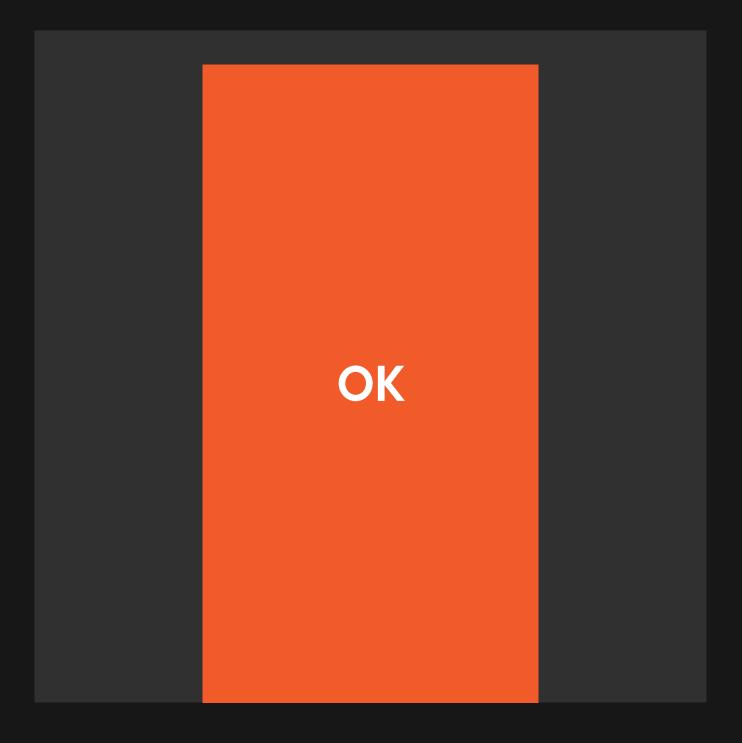




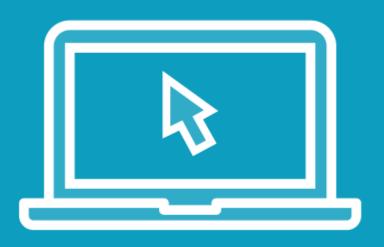




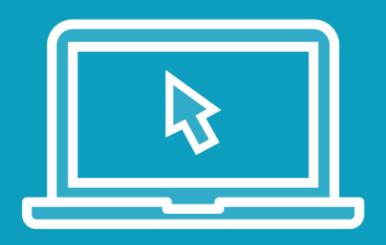




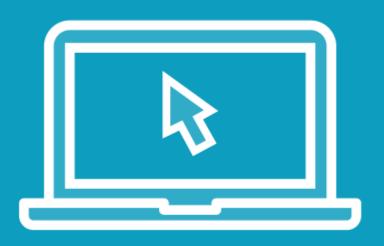




Position elements in the header of the Customers application



Center the header



Create a nested Grid for the navigation



#### Set Attached Properties in XAML

```
Attribute syntax <Button Content="OK"/>
```

```
Property element
syntax

Sutton.Content>
OK

</pre
```

<Button>

</Button>



#### Set Attached Properties in XAML

```
Attribute syntax <Button Grid.Row="1"/>
```



#### Set Attached Properties in C#

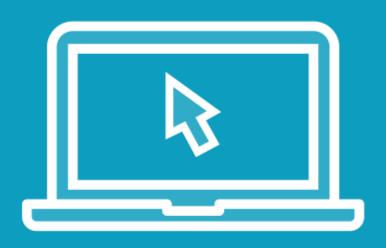
```
XAML <Button Grid.Row="1"/>
```

```
var btn = new Button();
btn.SetValue(Grid.RowProperty, 1);

C#

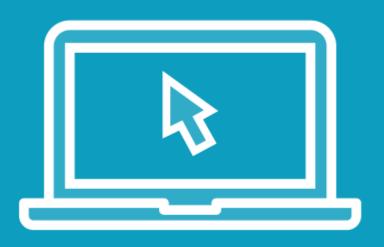
var row =
   (int)btn.GetValue(Grid.RowProperty);
```





Set attached properties in C#

Move the navigation to the other side



Access attached properties with static methods



#### Summary



#### Use the WPF layout panels

- Grid, StackPanel, Canvas

Position elements with layout properties

Set attached properties in

- XAML
- C#

# Up Next: Organizing Code with UserControls