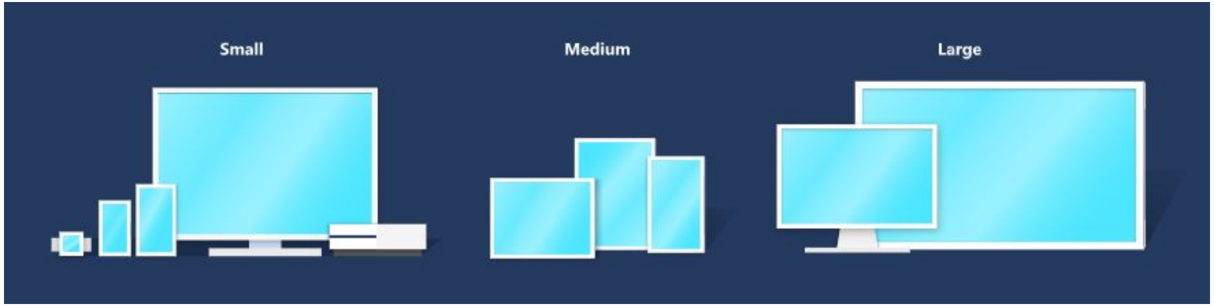
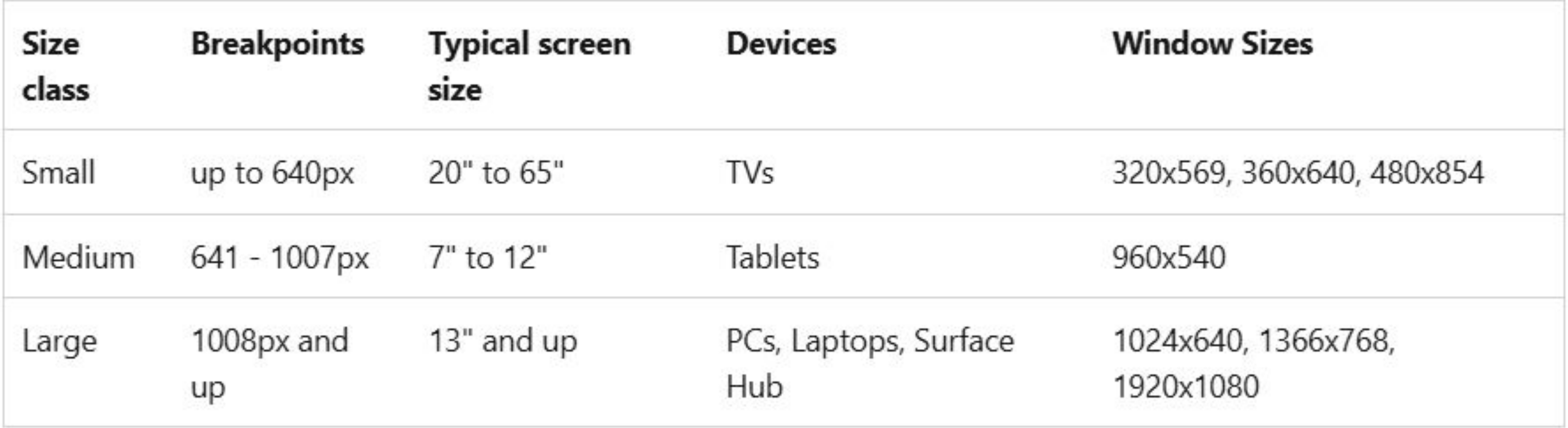
Layout basic

https://learn.microsoft.com/en-us/windows/apps/design/layout

Screen sizes & breakpoints

Breakpoints

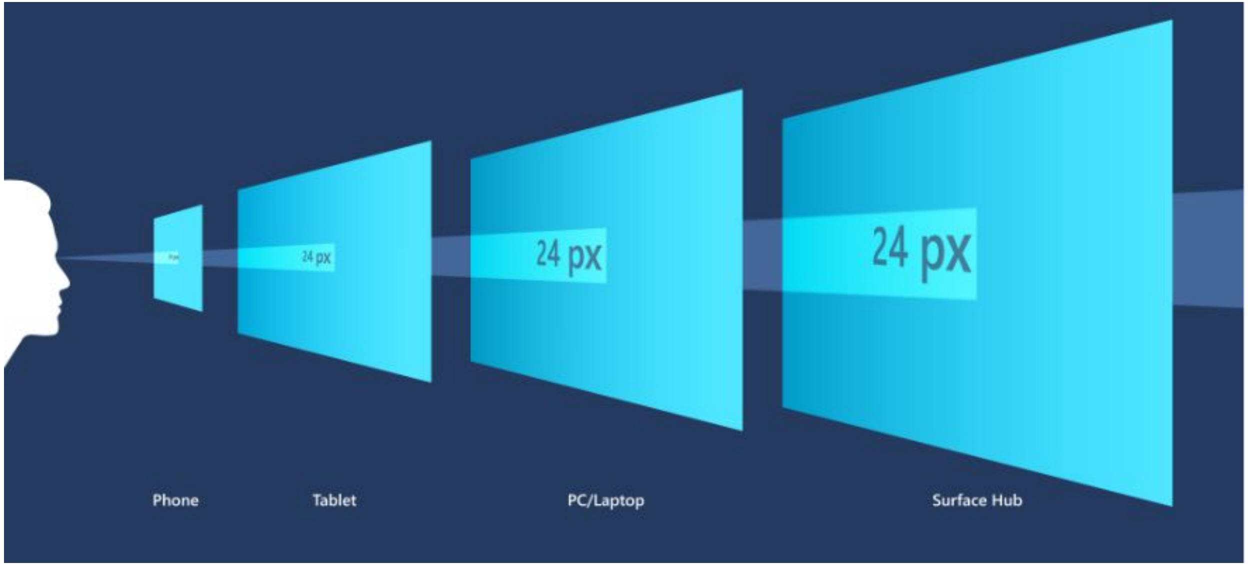




Size class	Breakpoints	Typical screen size	Devices	Window Sizes
Small	up to 640px	20" to 65"	TVs	320x569, 360x640, 480x854
Medium	641 - 1007px	7" to 12"	Tablets	960x540
Large	1008px and up	13" and up	PCs, Laptops, Surface Hub	1024x640, 1366x768, 1920x1080

Effective pixels and scale factor

Scaling algorithm takes into account **viewing distance** and **screen density** (pixels per inch) to optimize for perceived size



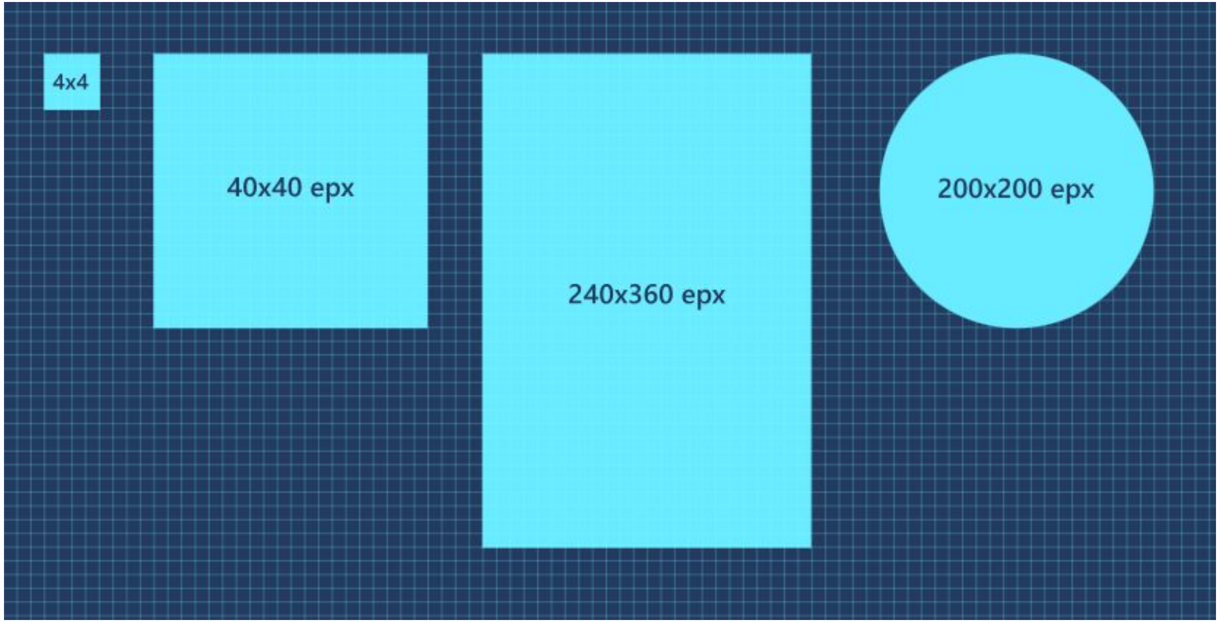
Why are TVs considered "small"?

- Viewing distance
- Effective pixels
  - 1080p TV has 1080 physical pixels
  - 540 effective pixels

=> designing for a **TV** == designing for a small screen

Multiples of Four

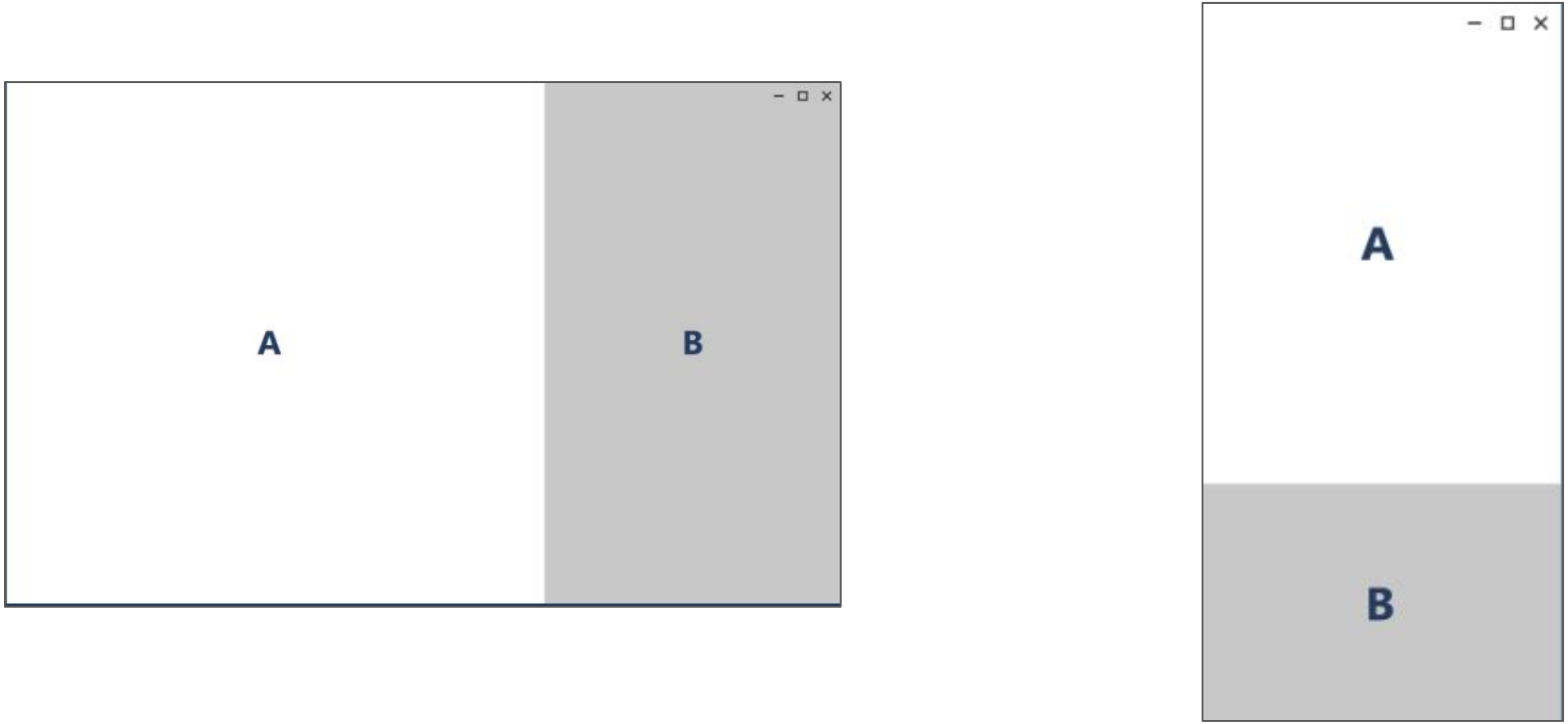
The sizes, margins, and positions of UI elements should always be in multiples of 4 epx



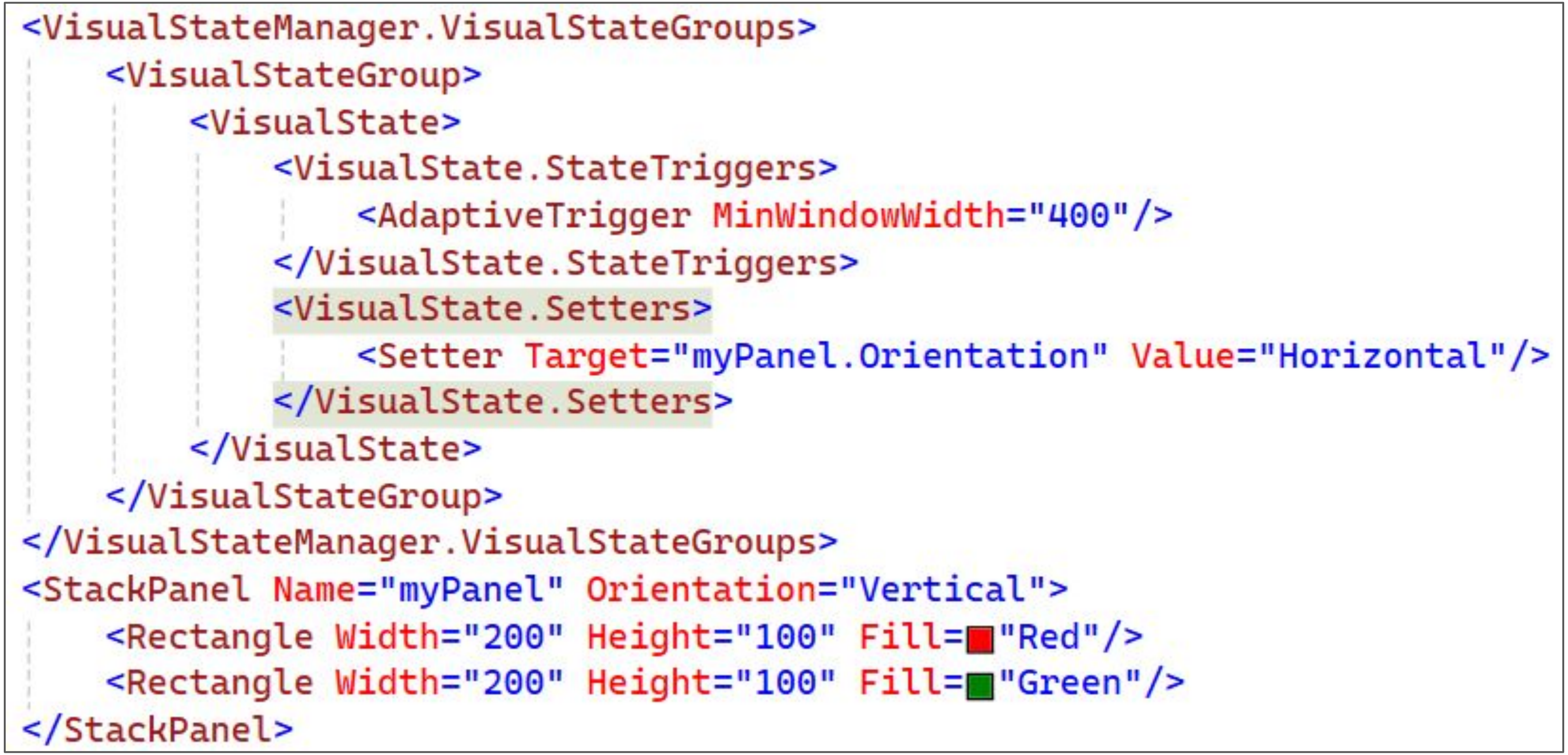
Responsive design techniques

[Responsive design techniques - Windows apps | Microsoft Learn]

Reposition

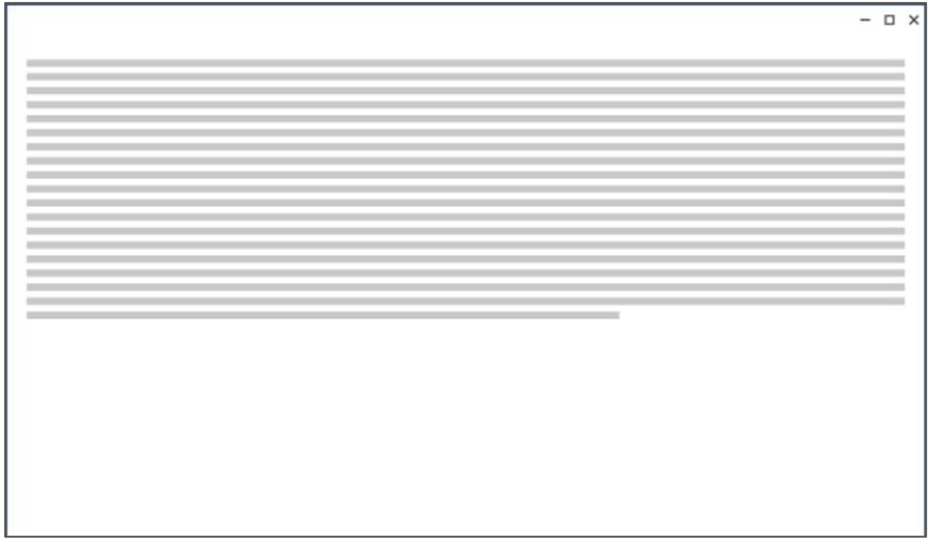


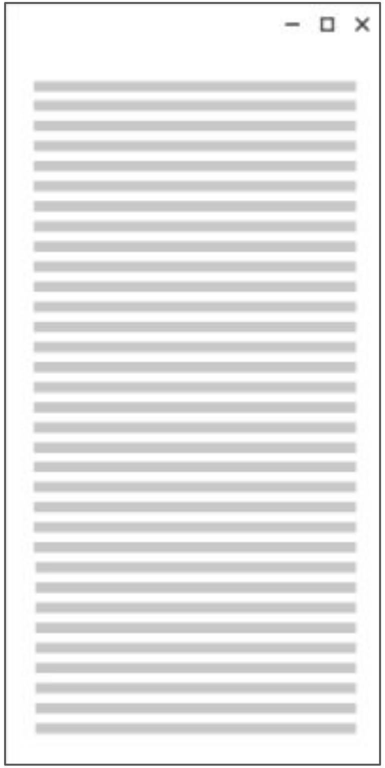
VisualState > StateTrigger (inside a Page)



```
<VisualStateManager.VisualStateGroups>
    <VisualStateGroup>
        <VisualState>
            <VisualState.StateTriggers>
                <AdaptiveTrigger MinWindowWidth="400"/>
            </VisualState.StateTriggers>
            <VisualState.Setters>
                <Setter Target="myPanel.Orientation" Value="Horizontal"/>
            </VisualState.Setters>
        </VisualState>
    </VisualStateGroup>
</VisualStateManager.VisualStateGroups>
<StackPanel Name="myPanel" Orientation="Vertical">
    <Rectangle Width="200" Height="100" Fill=🟥"Red"/>
    <Rectangle Width="200" Height="100" Fill=🟩"Green"/>
</StackPanel>
```

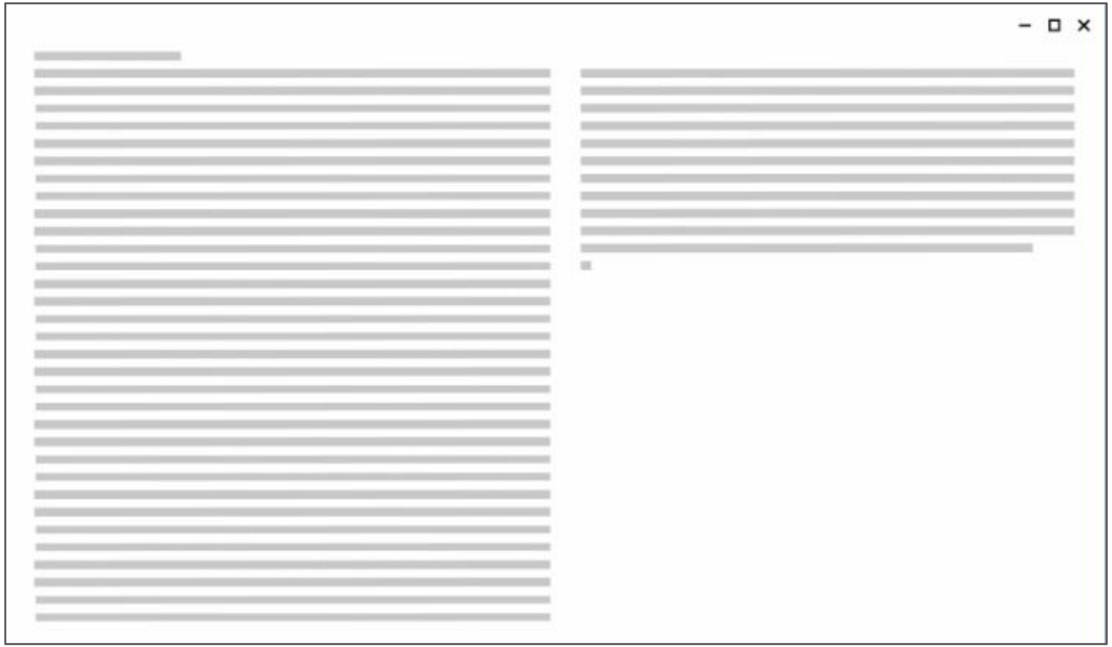
Resize

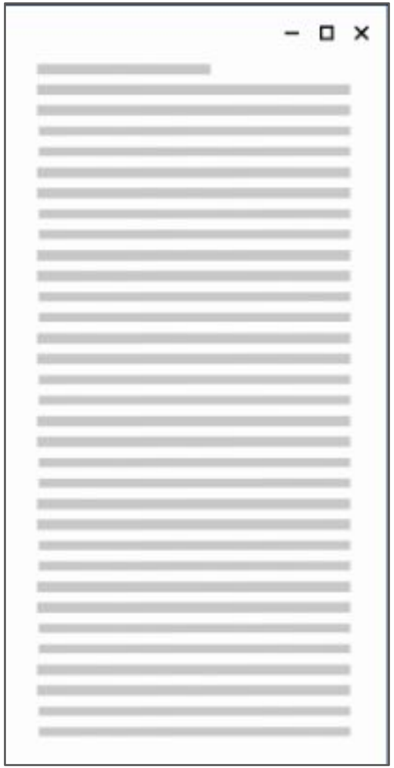




- ❑ Augment the reading experience on a larger screen by simply growing the content frame

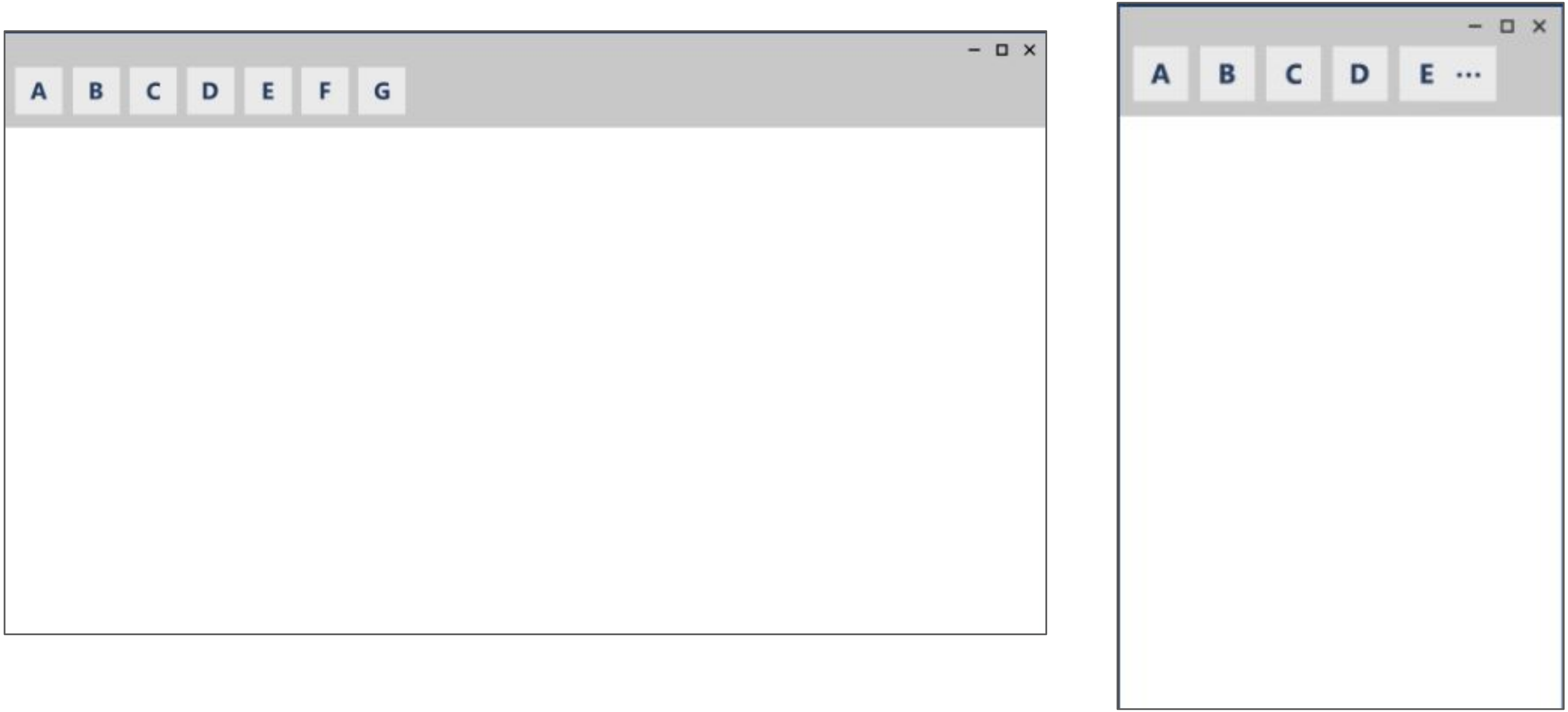
Reflow



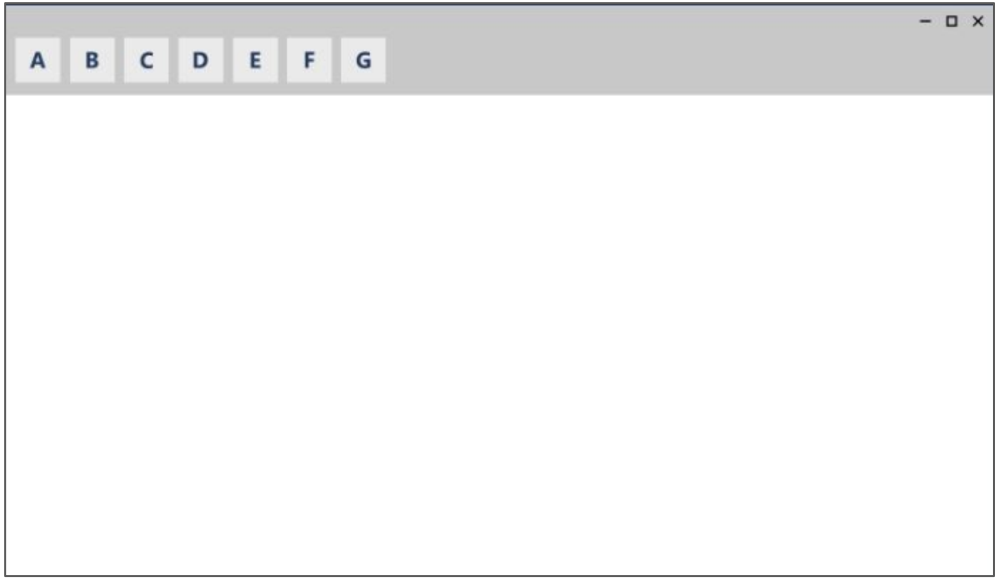


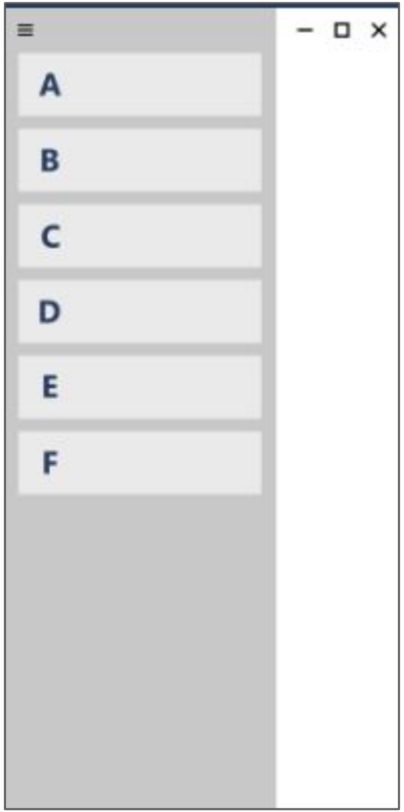
- ❑ Larger screen -> more columns, larger containers

Show/Hide



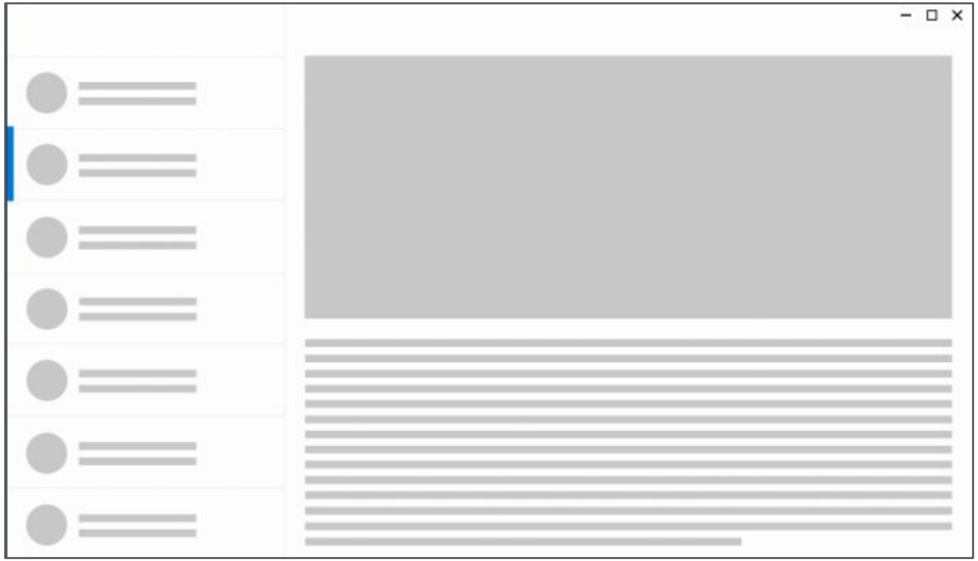
Replace

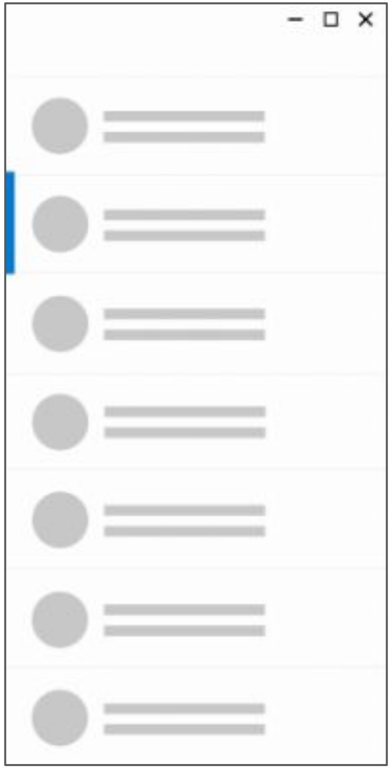




- ❑ **NavigationView**: set pane to top or left

Re-architect





- Expanding the window shows the entire list/details pattern

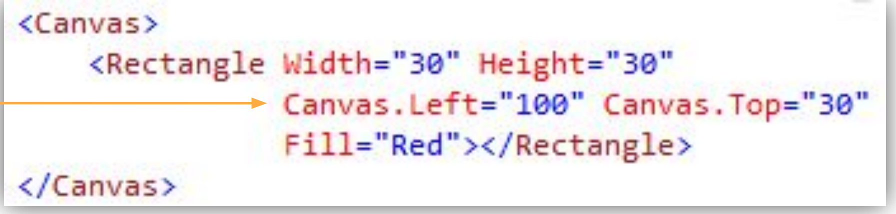
Layout panels

Layout panels for Windows apps - Windows apps | Microsoft Learn

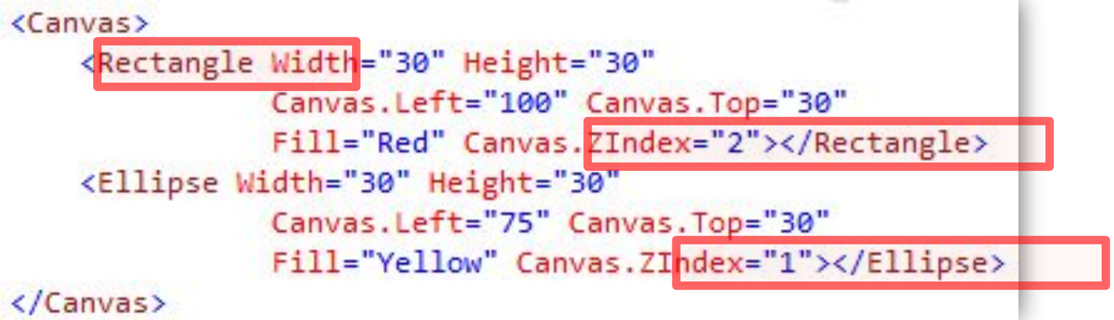
Canvas - Absolute positions

- Index: Canvas.ZIndex. Bigger number on top

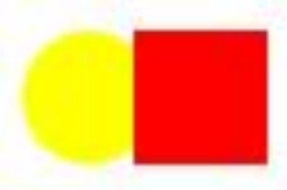
Attached properties →



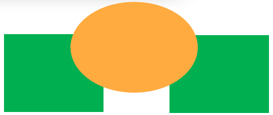
```
<Canvas>
    <Rectangle Width="30" Height="30"
               Canvas.Left="100" Canvas.Top="30"
               Fill="Red"></Rectangle>
</Canvas>
```



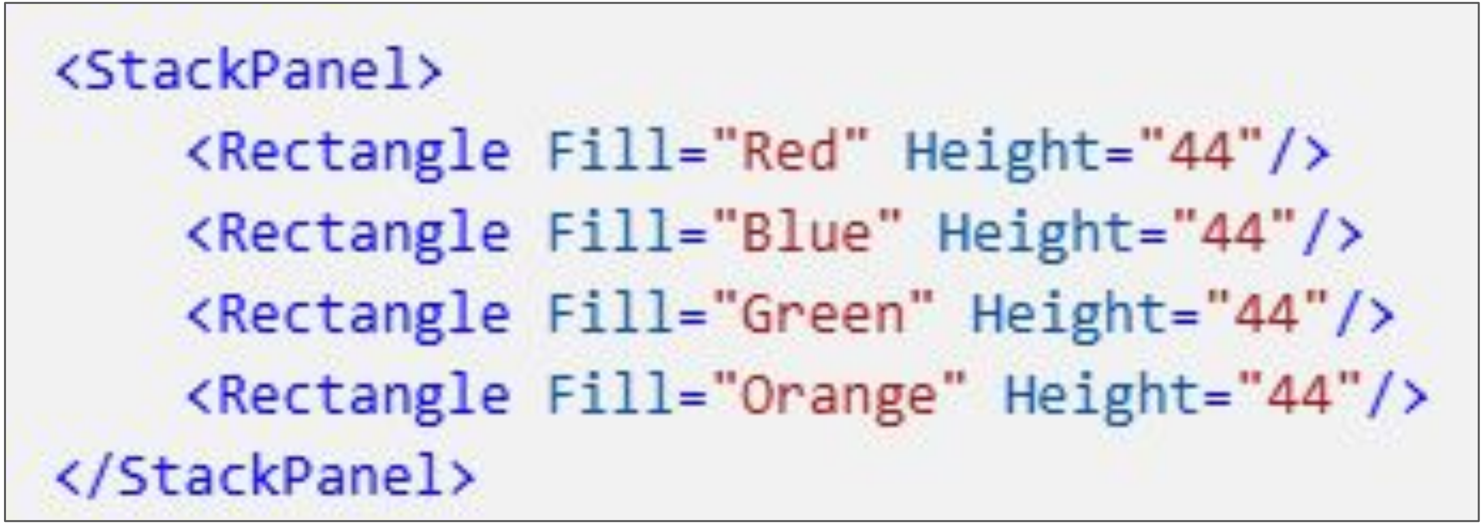
```
<Canvas>
    <Rectangle Width="30" Height="30"
               Canvas.Left="100" Canvas.Top="30"
               Fill="Red" Canvas.ZIndex="2"></Rectangle>
    <Ellipse Width="30" Height="30"
               Canvas.Left="75" Canvas.Top="30"
               Fill="Yellow" Canvas.ZIndex="1"></Ellipse>
</Canvas>
```



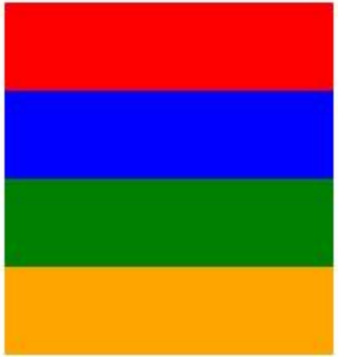
Quiz:



StackPanel

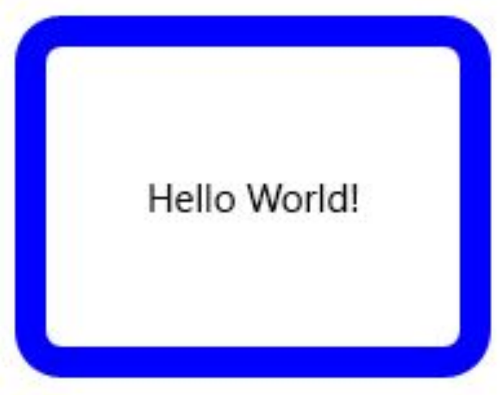


```
<StackPanel>
    <Rectangle Fill="Red" Height="44"/>
    <Rectangle Fill="Blue" Height="44"/>
    <Rectangle Fill="Green" Height="44"/>
    <Rectangle Fill="Orange" Height="44"/>
</StackPanel>
```



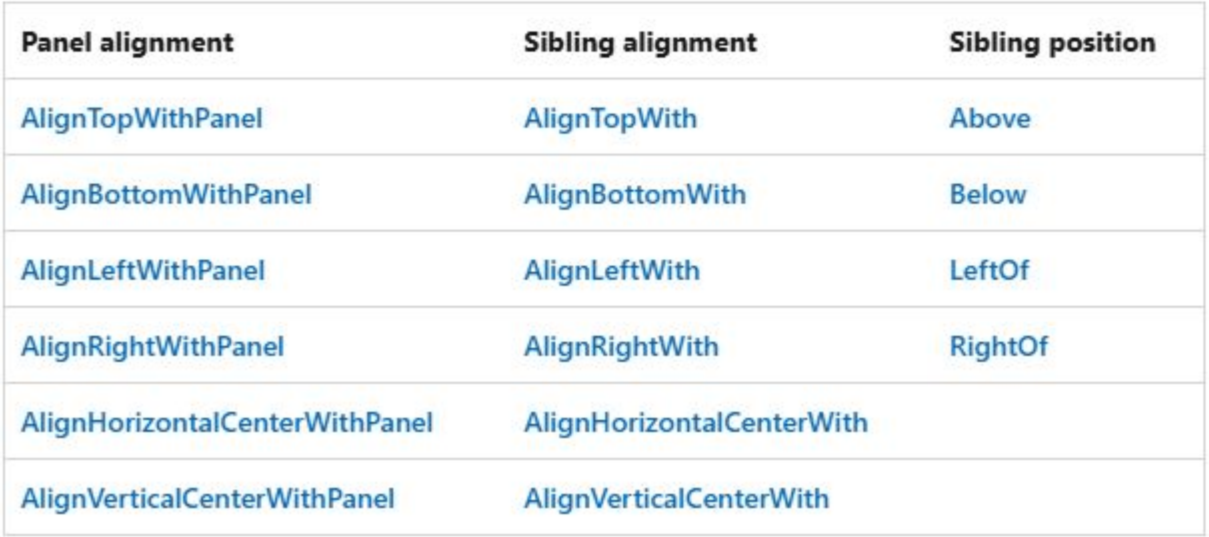
Panel border’s properties

- RelativePanel, StackPanel, and Grid:
  - BorderThickness
  - BorderBrush
  - CornerRadius
  - Padding



Hello World!

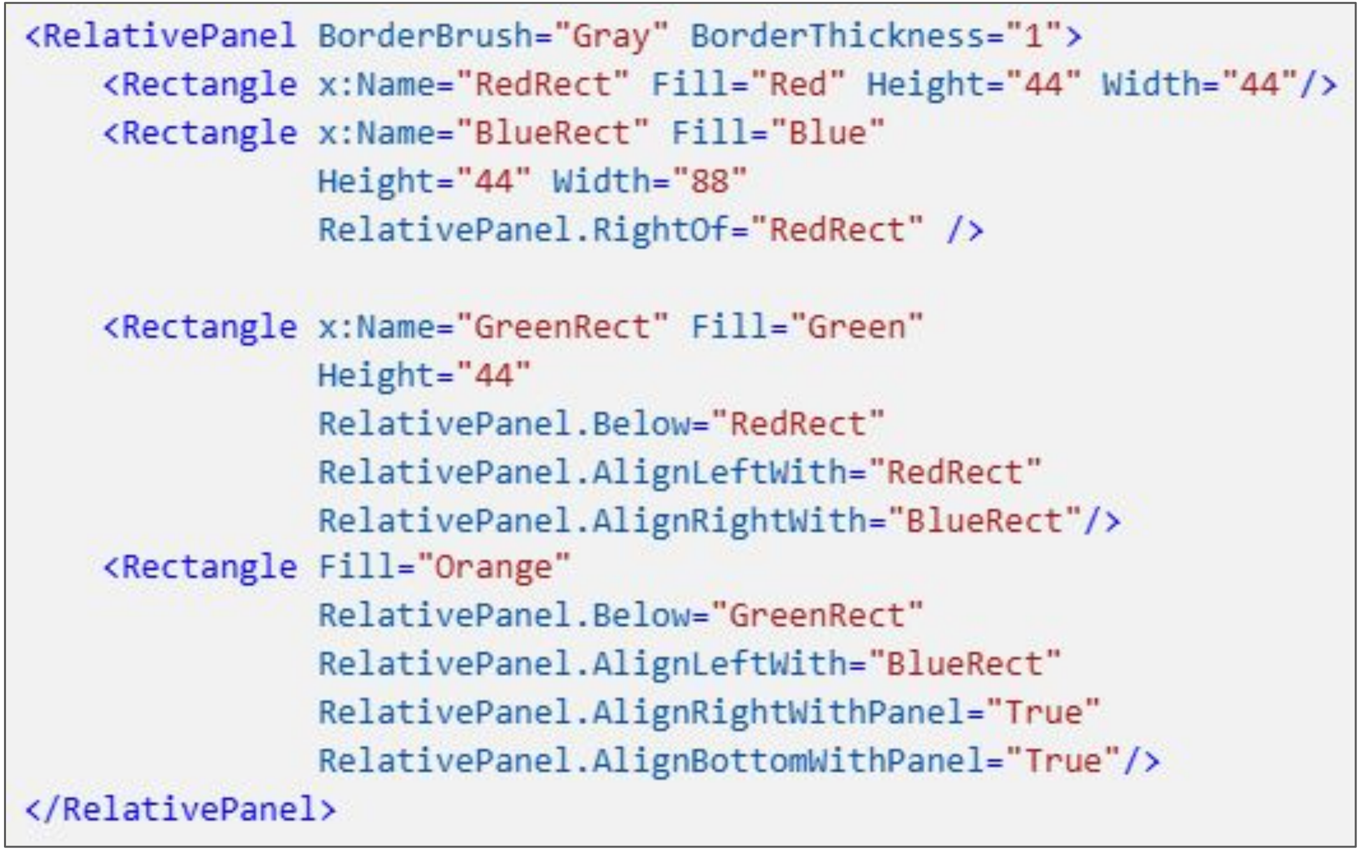
RelativePanel



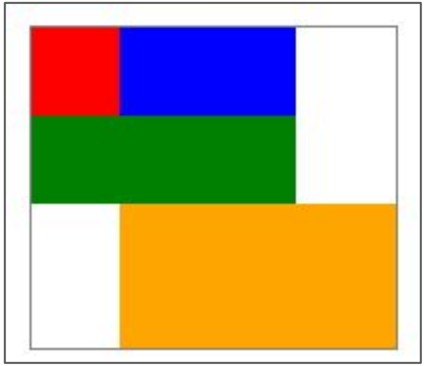
Panel alignment	Sibling alignment	Sibling position
AlignTopWithPanel	AlignTopWith	Above
AlignBottomWithPanel	AlignBottomWith	Below
AlignLeftWithPanel	AlignLeftWith	LeftOf
AlignRightWithPanel	AlignRightWith	RightOf
AlignHorizontalCenterWithPanel	AlignHorizontalCenterWith	
AlignVerticalCenterWithPanel	AlignVerticalCenterWith	

To rearrange, use: VisualStateManager & AdaptiveTrigger

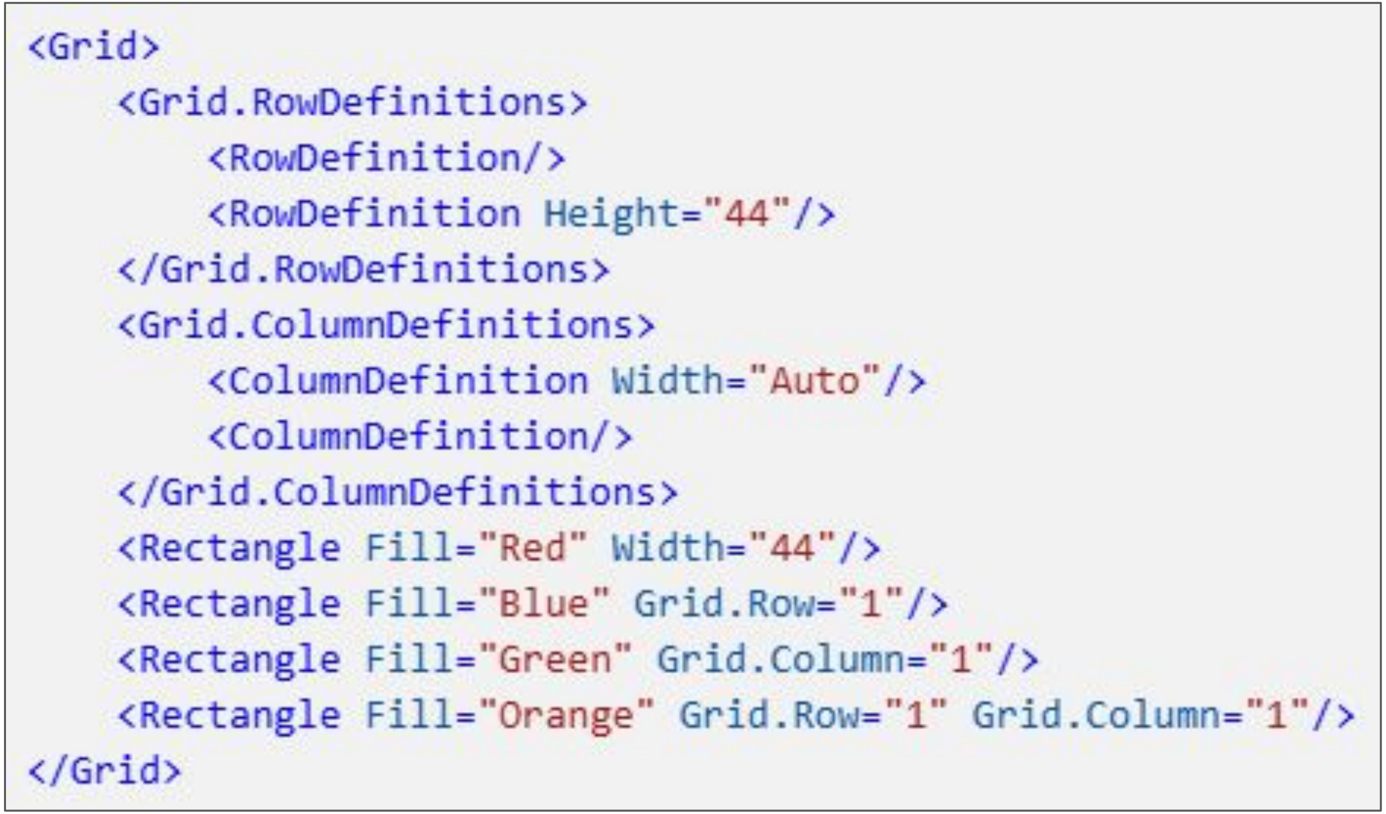
RelativePanel Example



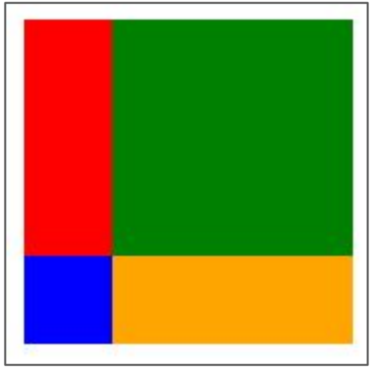
```
<RelativePanel BorderBrush="Gray" BorderThickness="1">
    <Rectangle x:Name="RedRect" Fill="Red" Height="44" Width="44"/>
    <Rectangle x:Name="BlueRect" Fill="Blue"
               Height="44" Width="88"
               RelativePanel.RightOf="RedRect" />

    <Rectangle x:Name="GreenRect" Fill="Green"
               Height="44"
               RelativePanel.Below="RedRect"
               RelativePanel.AlignLeftWith="RedRect"
               RelativePanel.AlignRightWith="BlueRect"/>
    <Rectangle Fill="Orange"
               RelativePanel.Below="GreenRect"
               RelativePanel.AlignLeftWith="BlueRect"
               RelativePanel.AlignRightWithPanel="True"
               RelativePanel.AlignBottomWithPanel="True"/>
</RelativePanel>
```



Grid



```
<Grid>
    <Grid.RowDefinitions>
        <RowDefinition/>
        <RowDefinition Height="44"/>
    </Grid.RowDefinitions>
    <Grid.ColumnDefinitions>
        <ColumnDefinition Width="Auto"/>
        <ColumnDefinition/>
    </Grid.ColumnDefinitions>
    <Rectangle Fill="Red" Width="44"/>
    <Rectangle Fill="Blue" Grid.Row="1"/>
    <Rectangle Fill="Green" Grid.Column="1"/>
    <Rectangle Fill="Orange" Grid.Row="1" Grid.Column="1"/>
</Grid>
```



Grid Sizing

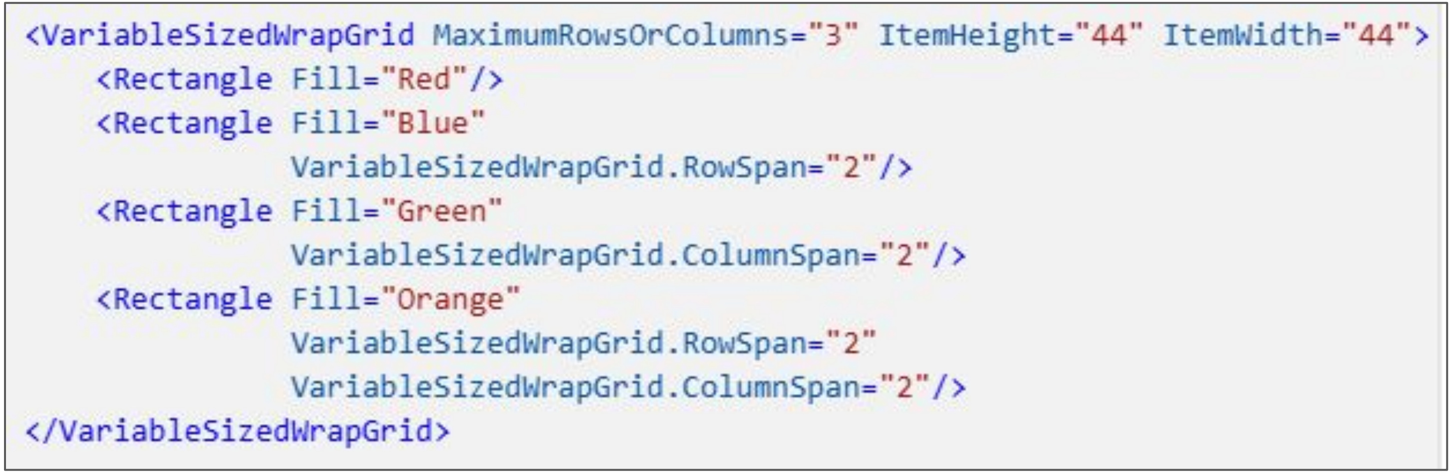
❑ **Auto**: UI elements resize to fit their content or parent container

❑ *: proportional sizing

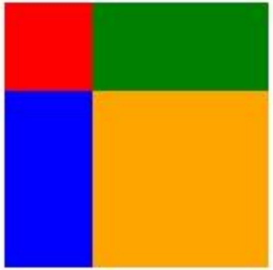
VariableSizedWrapGrid

- Rows or columns automatically **wrap to a new row** or column when the **MaximumRowsOrColumns** value is reached.

VariableSizedWrapGrid Example



```
<VariableSizedWrapGrid MaximumRowsOrColumns="3" ItemHeight="44" ItemWidth="44">
    <Rectangle Fill="Red"/>
    <Rectangle Fill="Blue"
               VariableSizedWrapGrid.RowSpan="2"/>
    <Rectangle Fill="Green"
               VariableSizedWrapGrid.ColumnSpan="2"/>
    <Rectangle Fill="Orange"
               VariableSizedWrapGrid.RowSpan="2"
               VariableSizedWrapGrid.ColumnSpan="2"/>
</VariableSizedWrapGrid>
```



Special-purpose panels

Only as an **ItemsPanel** to display items in an **ItemsControl**

❑ ItemsStackPanel

❑ ItemsWrapGrid

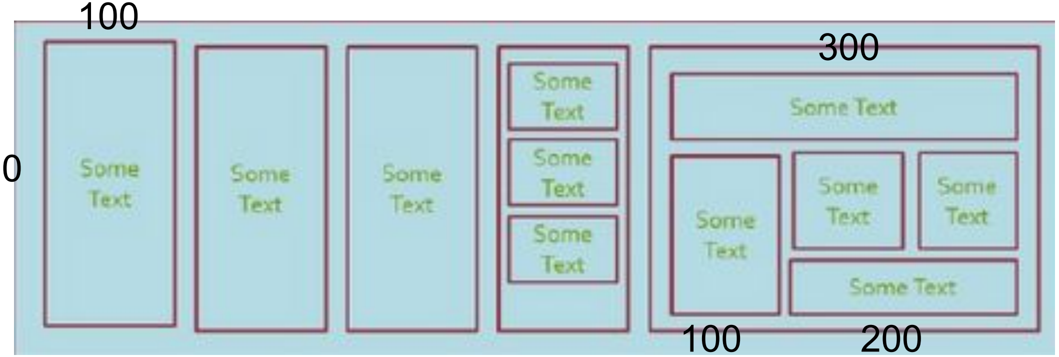
❑ VirtualizingStackPanel

❑ WrapGrid

Inclass excersise

- Write xaml code to create UI nearly like the image (80%)
- Hint: there are a lot of solutions

1



100

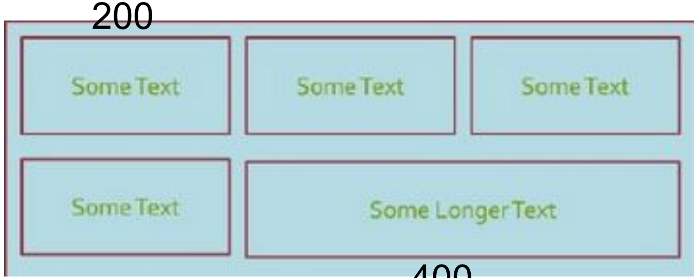
400

300

100

200

2

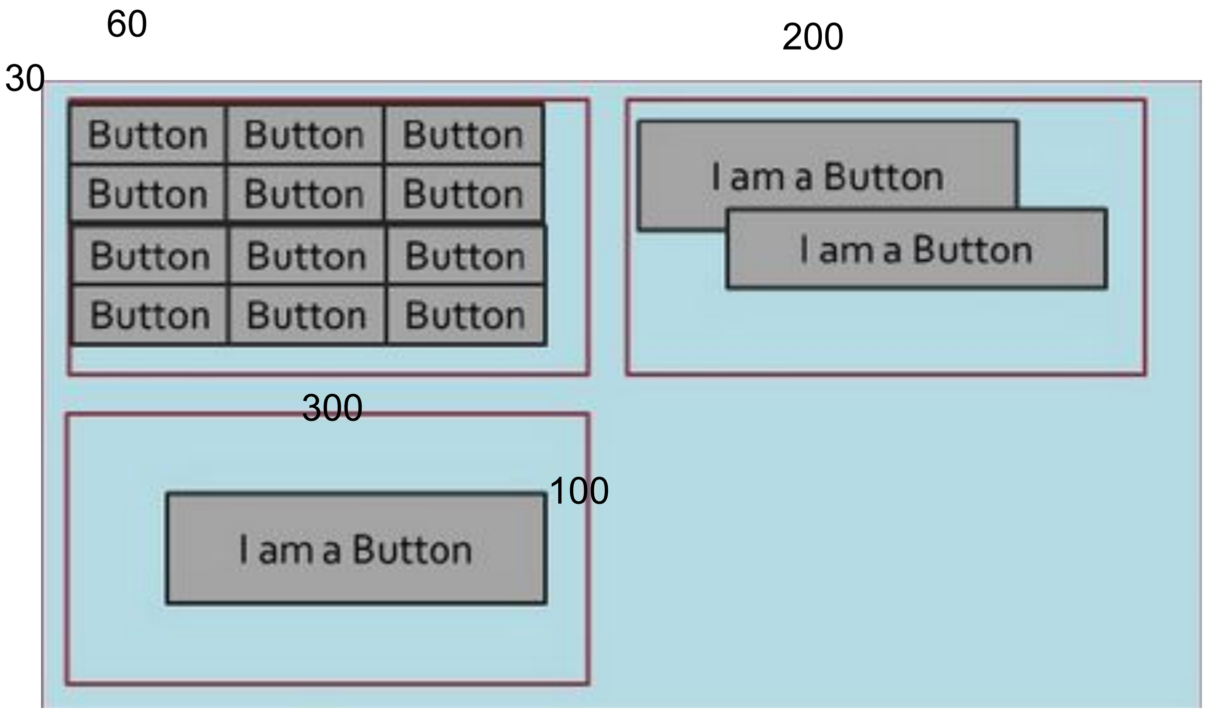


200

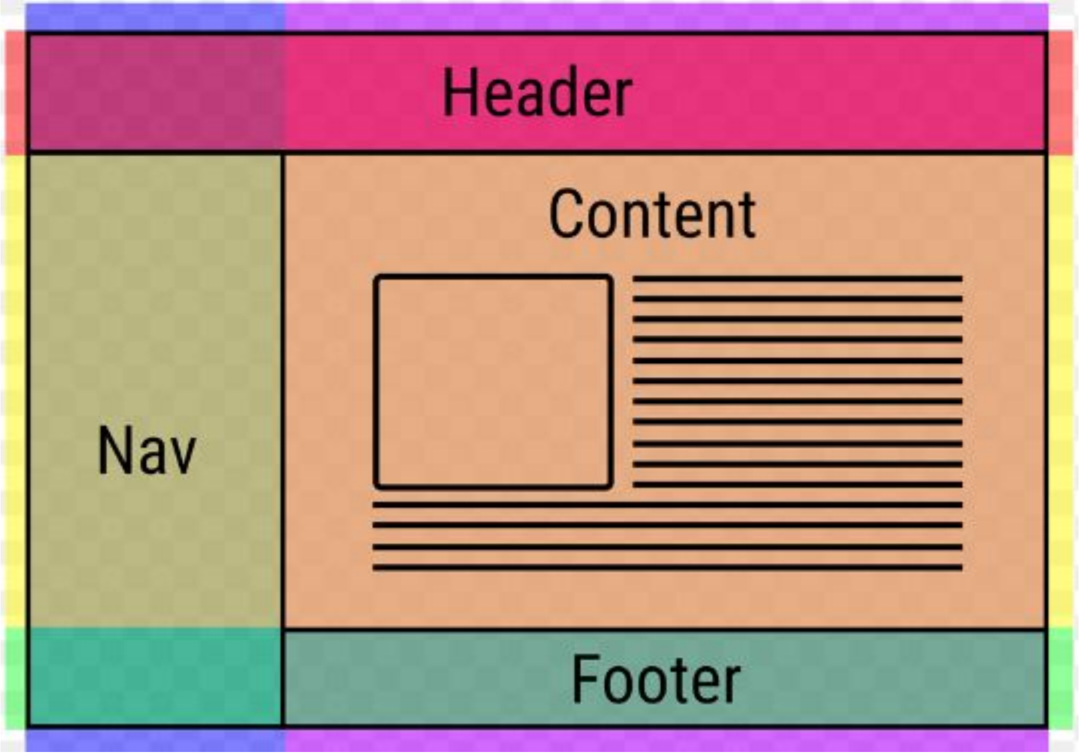
400

Exercises (cont)

3



Inclass Exercise 4 - Classic layouts



Exercise - Responsive design

Settings