tutorial

Windows App SDK

Custom CheckBox

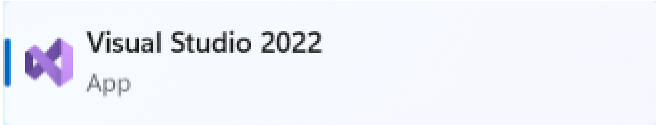
Custom CheckBox

Custom CheckBox shows how to create a customised **Style** for a **CheckBox** using **Windows App SDK**

Step 1

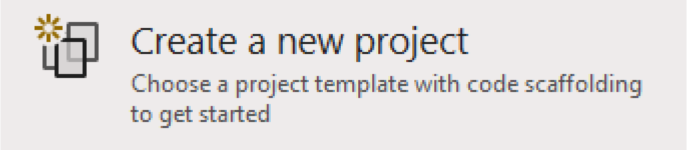
Follow **Setup and Start** on how to get **Setup** and **Install** what you need for **Visual Studio 2022** and **Windows App SDK**.

In **Windows 11** choose **Start** and then find or search for **Visual Studio 2022** and then select it.



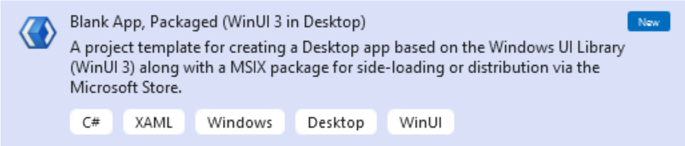
Visual Studio 2022
App

Once **Visual Studio 2022** has started select **Create a new project**.



Create a new project
Choose a project template with code scaffolding to get started

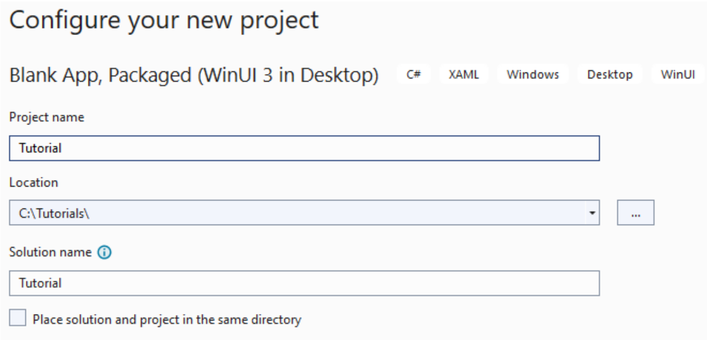
Then choose the **Blank App, Packages (WinUI in Desktop)** and then select **Next**.



Blank App, Packaged (WinUI 3 in Desktop)

A project template for creating a Desktop app based on the Windows UI Library (WinUI 3) along with a MSIX package for side-loading or distribution via the Microsoft Store.

After that in **Configure your new project** type in the **Project name** as *CustomCheckBox*, then select a Location and then select **Create** to start a new **Solution**.



Configure your new project

Blank App, Packaged (WinUI 3 in Desktop)

Project name: Tutorial

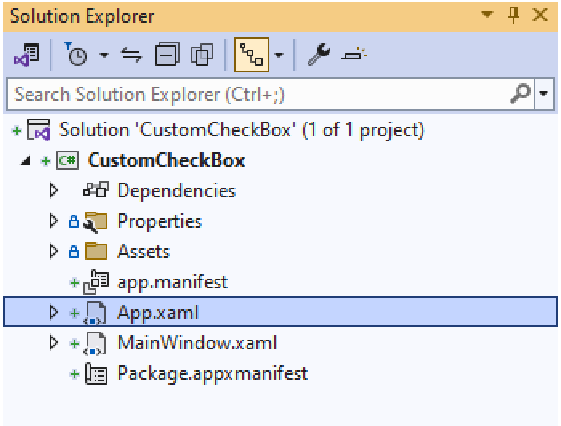
Location: C:\Tutorials\

Solution name: Tutorial

□ Place solution and project in the same directory

Step 2

Then in **Visual Studio** within **Solution Explorer** for the **Solution** double-click on **App.xaml** to see the **XAML** for the **Project**.



Solution Explorer

Search Solution Explorer (Ctrl+;)

- Solution 'CustomCheckBox' (1 of 1 project)
  - CustomCheckBox
    - Dependencies
    - Properties
    - Assets
    - app.manifest
    - App.xaml
    - MainWindow.xaml
    - Package.appxmanifest

Step 3

In the **XAML** for **App.xaml** below the **Comment** of **`<!-- Other app resources here -->`** type in the following **XAML**:

```
<Style x:Key="CustomCheckBox" TargetType="CheckBox">
    <Setter Property="Background" Value="Transparent"/>
    <Setter Property="Foreground"
    Value="{ThemeResource SystemControlForegroundBaseHighBrush}"/>
    <Setter Property="Padding" Value="5,5,0,0"/>
    <Setter Property="HorizontalAlignment" Value="Left"/>
    <Setter Property="VerticalAlignment" Value="Center"/>
    <Setter Property="HorizontalContentAlignment" Value="Left"/>
    <Setter Property="VerticalContentAlignment" Value="Top"/>
    <Setter Property="FontFamily"
    Value="{ThemeResource ContentControlThemeFontFamily}"/>
    <Setter Property="FontSize"
    Value="{ThemeResource ControlContentThemeFontSize}"/>
    <Setter Property="UseSystemFocusVisuals" Value="True"/>
    <Setter Property="Template">
        <Setter.Value>
            <ControlTemplate TargetType="CheckBox">
                <Grid BorderBrush="{TemplateBinding BorderBrush}"
                    BorderThickness="{TemplateBinding BorderThickness}"
                    Background="{TemplateBinding Background}">
                    <Grid.ColumnDefinitions>
                        <ColumnDefinition Width="30"/>
                        <ColumnDefinition Width="*"/>
                    </Grid.ColumnDefinitions>
                    <!-- Visual State Groups -->

                    <!-- Content -->

                </Grid>
            </ControlTemplate>
        </Setter.Value>
    </Setter>
</Style>
```

This **XAML** is part of a **`Style`** of **`CustomCheckBox`** that will be used to target a **`CheckBox`** which will contain some **Visual State Groups** and **Content** for the **Custom CheckBox**.

Step 4

While still in the **XAML** for **App.xaml** below the **Comment** of **`<!-- Visual State Groups -->`** type the following **XAML**:

```
<VisualStateManager.VisualStateGroups>
    <VisualStateGroup x:Name="CombinedStates">
        <VisualState x:Name="Unchecked">
            <Storyboard>
                <ObjectAnimationUsingKeyFrames
                    Storyboard.TargetProperty="Opacity"
                    Storyboard.TargetName="InnerRectangle">
                    <DiscreteObjectKeyFrame KeyTime="0" Value="0"/>
                </ObjectAnimationUsingKeyFrames>
            </Storyboard>
        </VisualState>
        <VisualState x:Name="Checked">
            <Storyboard>
                <ObjectAnimationUsingKeyFrames
                    Storyboard.TargetProperty="Opacity"
                    Storyboard.TargetName="InnerRectangle">
                    <DiscreteObjectKeyFrame KeyTime="0" Value="1"/>
                </ObjectAnimationUsingKeyFrames>
            </Storyboard>
        </VisualState>
    </VisualStateGroup>
</VisualStateManager.VisualStateGroups>
```

This **XAML** is for the **Visual State Groups** that will represent the **States** for the **`CheckBox`** including how it will behave when it is **`Unchecked`** and when it is **`Checked`** for the **Custom CheckBox**.

Step 5

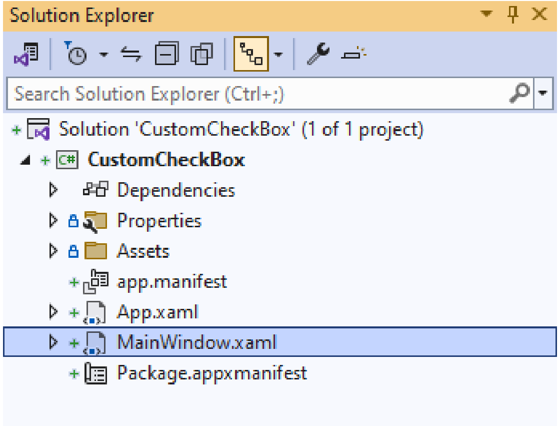
Then in the **XAML** for **App.xaml** below the **Comment** of **`<!-- Content -->`** type the following **XAML:**

```
<Grid>
    <Rectangle x:Name="NormalRectangle"
    Height="30" Width="30" RadiusY="10" RadiusX="10"
    UseLayoutRounding="False" StrokeThickness="2"
    Stroke="Salmon" Fill="LightSalmon"/>
    <Rectangle x:Name="InnerRectangle" Height="20"
    Width="20" RadiusY="4" RadiusX="4"
    UseLayoutRounding="False" StrokeThickness="2"
    Stroke="Goldenrod" Fill="Gold" Opacity="0"/>
</Grid>
<ContentPresenter x:Name="ContentPresenter" Grid.Column="1"
AutomationProperties.AccessibilityView="Raw"
ContentTemplate="{TemplateBinding ContentTemplate}"
ContentTransitions="{TemplateBinding ContentTransitions}"
Content="{TemplateBinding Content}"
HorizontalAlignment="{TemplateBinding HorizontalContentAlignment}"
Margin="{TemplateBinding Padding}" TextWrapping="Wrap"
VerticalAlignment="{TemplateBinding VerticalContentAlignment}"/>
```

This **XAML** is the **Content** for the layout of a **`CheckBox`** when the **`Style`** is applied for the **Custom CheckBox**.

Step 6

Within **Solution Explorer** for the **Solution** double-click on **MainWindow.xaml** to see the **XAML** for the **Main Window**.



Step 7

In the **XAML** for **MainWindow.xaml** there be some **XAML** for a **StackPanel**, this should be **Removed** by removing the following:

```
<StackPanel Orientation="Horizontal"
HorizontalAlignment="Center" VerticalAlignment="Center">
    <Button x:Name="myButton" Click="myButton_Click">Click Me</Button>
</StackPanel>
```

Step 8

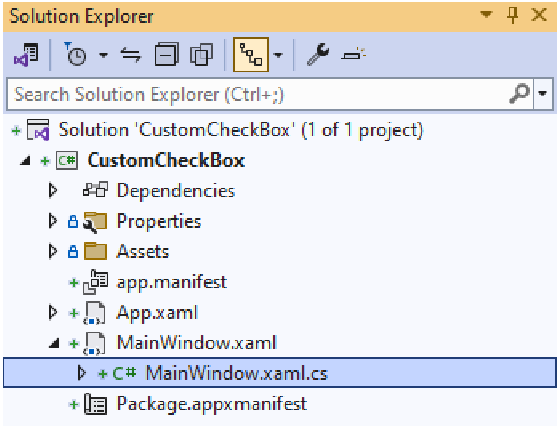
While still in the **XAML** for **MainWindow.xaml** above **</Window>**, type in the following **XAML**:

```
<CheckBox HorizontalAlignment="Center" Content="CheckBox"
Style="{StaticResource CustomCheckBox}"/>
```

This **XAML** contains a **CheckBox** with **Style** set to the **StaticResource** of **CustomCheckBox** from **App.xaml**.

Step 9

Then, within **Solution Explorer** for the **Solution** select the arrow next to **MainWindow.xaml** then double-click on **MainWindow.xaml.cs** to see the **Code** for the **Main Window**.



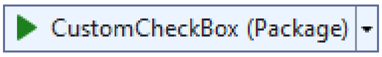
Step 10

In the **Code** for **MainWindow.xaml.cs** there be a **Method** of **myButton_Click(...)** this should be **Removed** by removing the following:

```
private void myButton_Click(object sender, RoutedEventArgs e)
{
    myButton.Content = "Clicked";
}
```

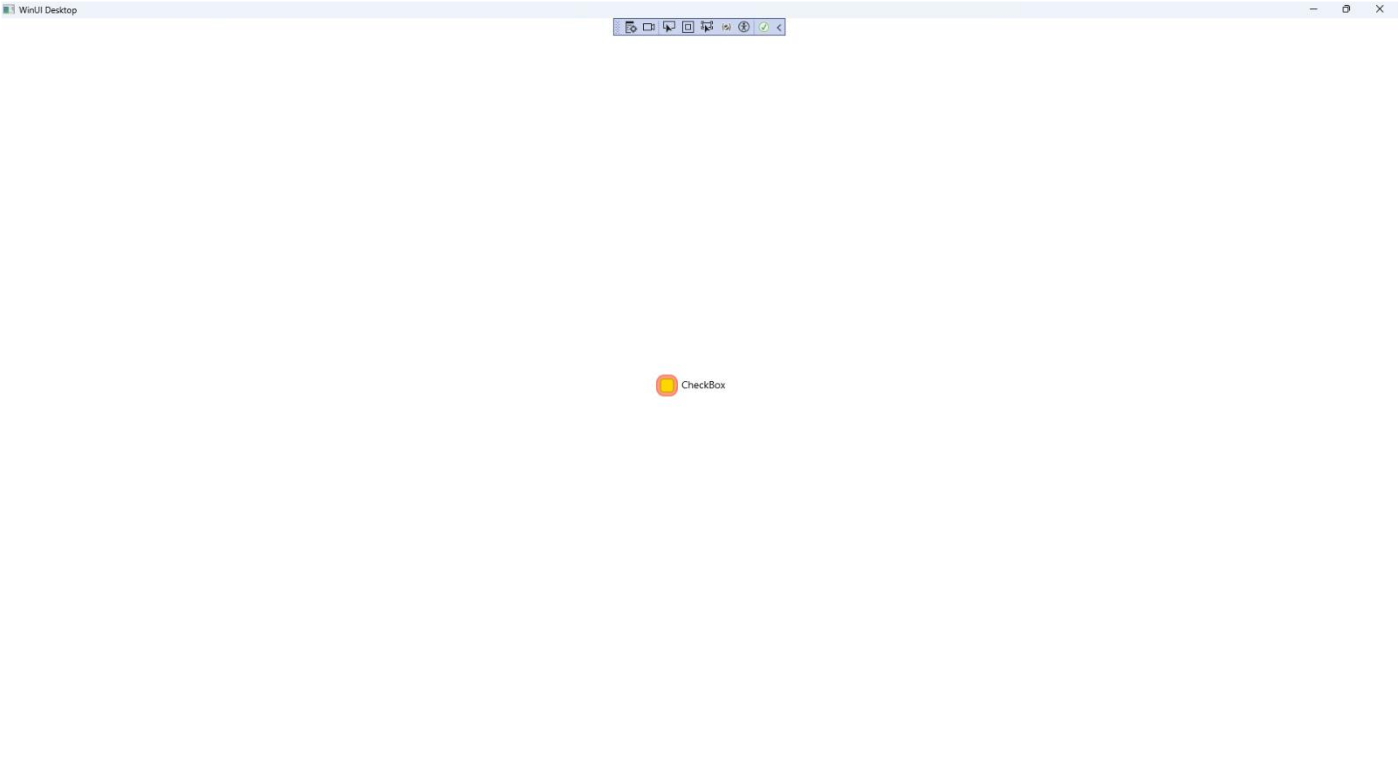
Step 11

That completes the **Windows App SDK** application. In **Visual Studio 2022** from the **Toolbar** select **CustomCheckBox (Package)** to **Start** the application.



Step 12

Once running you will see the **Custom Checkbox** displayed which you can then set to **Checked**.



Step 13

To **Exit** the **Windows App SDK** application, select the **Close** button from the top right of the application as that concludes this **Tutorial** for **Windows App SDK** from tutorialr.com!