|  |
| --- |
|  |
| Universal Windows Platform  tutorialr.com |
| |  |  |  | | --- | --- | --- | |  |  |  | |

## Setup

You will need to enable **Developer Mode** in **Windows 10** if this has not been done already by completing the following:

|  |  |
| --- | --- |
| A close up of a sign  Description automatically generated | Choose **Start** then **Settings** from the **Start Menu** |
| A picture containing screenshot  Description automatically generated | Next from **Windows Settings** choose **Update & Security** |
| A close up of a sign  Description automatically generated | Then from **Update & Security** choose **For developers** |
|  | Finally make sure **Developer mode** is selected |

## Install

You will need to install **Visual Studio 2019 Community**, if this has not been done already you just need to do the following:

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | Visit [**VisualStudio.com**](http://visualstudio.com) and then from the **Visual Studio IDE** section choose **Download Visual Studio** then **Community 2019** |
|  | Next on the **Thank you for downloading Visual Studio** page when the download prompt appears, select **Run** |
| A screenshot of a cell phone  Description automatically generated | Once downloaded, this should start the **Visual Studio Installer** and select **Continue** to begin the installation |
| A screenshot of a cell phone  Description automatically generated | Next once ready select **Universal Windows Platform development** from the **Workloads** section |
| A screenshot of a cell phone  Description automatically generated | Then make sure the latest **Windows 10 SDK** has been selected from **Installation details** if not already included then select **Install** and follow any instructions to complete the installation |

## Start

You will need to open **Visual Studio 2019 Community** and **Create a new project**, if this has not been done already you just need to do the following:

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | In **Windows 10** choose **Start**, and then from the **Start Menu** find and select **Visual Studio 2019** |
|  | Once done, from the **Get started** screen for **Visual Studio 2019** select **Create a new project** |
| A screenshot of a cell phone  Description automatically generated | Choose **Blank App (Universal Windows)** and select **Next** |
| A screenshot of a cell phone  Description automatically generated | Then in **Configure your new project** enter a **Project name** and **Location** and select **Create** |
| A screenshot of a social media post  Description automatically generated | Finally, in **New Universal Windows Platform Project** pick the latest **Target version** and **Minimum version** and select **OK** |

## Guide

In **Visual Studio 2019** there is a **Menu** at the top

A close up of a logo

Description automatically generated

Below this is the **Toolbar**, options there include **Local Machine** to start debugging

A screenshot of a cell phone

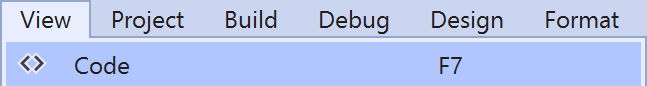
Description automatically generated

When an Application is running, the **Toolbar** will change to include other options including **Stop** to finish Debugging.

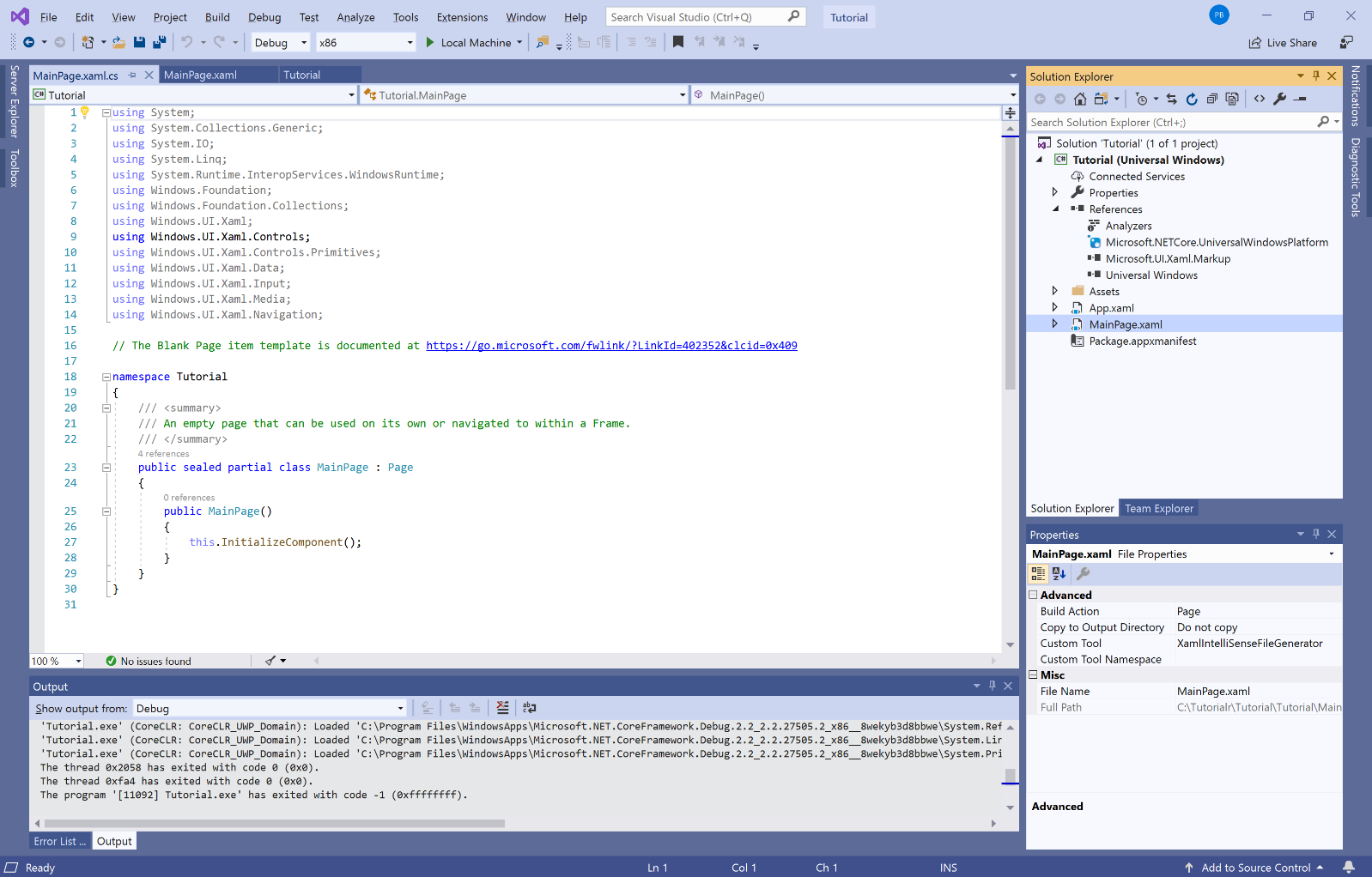
A picture containing object

Description automatically generated

After a **Solution** or **Project** has been opened or created, choose **View** then **Code** from the **Menu**



This will display the **Code** View for the code in a **Class** or for a **XAML** page such as **MainPage.xaml.cs**

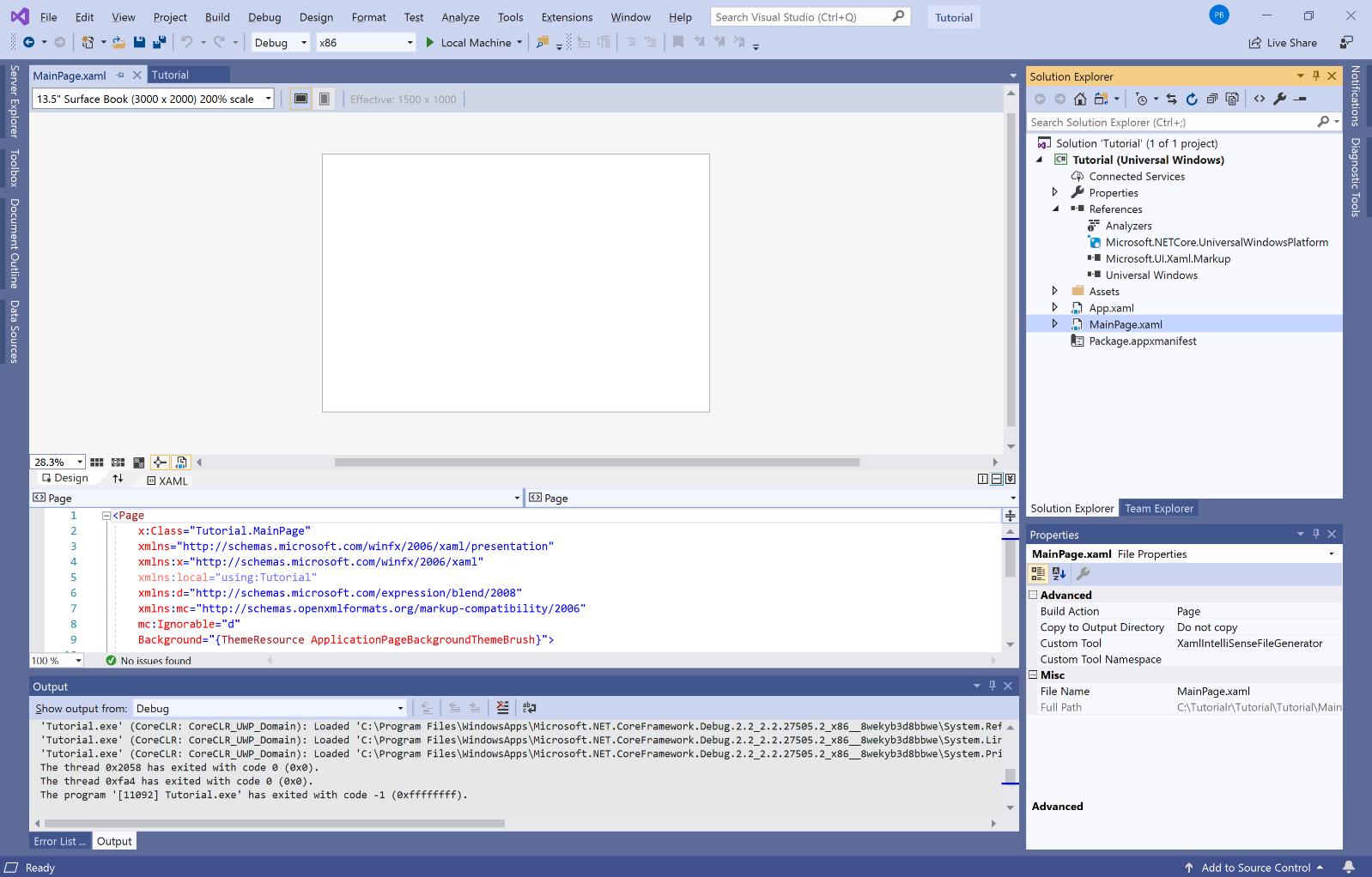


Also, after a **Solution** or **Project** has been opened or created, choose **View** then **Designer** from the **Menu**

A screenshot of a cell phone

Description automatically generated

This will display the **Design** and **XAML** views for a **XAML** page such as **MainPage.xaml**



**Hello World**, is used to introduce many new programming language examples in this case it is an introduction to the **Universal Windows Platform** where a message will be displayed on screen when a **Button** is clicked

## Step 1

|  |  |
| --- | --- |
|  | Follow **Setup and Start** on how to Install and/or Get Started with **Visual Studio 2019** if not already or in **Windows 10** choose **Start**, find and select **Visual Studio 2019** then from the **Get started** screen select **Create a new project** |
| A screenshot of a cell phone  Description automatically generated | Then choose **Blank App (Universal Windows)** and select **Next** and then in **Configure your new project** enter the **Project name** as **HelloWorld** and select **Create** |
| A screenshot of a social media post  Description automatically generated | Finally, in **New Universal Windows Platform Project** pick the **Target version** and **Minimum version** to be at least **Windows 10, version 1903 (10.0; Build 18362)** and then select **OK** |

**Target Version** will control the most recent features of **Windows 10** your application can use. To make sure you always have the most recent version, check for any **Notifications** or **Updates** in **Visual Studio 2019**

## Step 2

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | In the **Solution Explorer** of **Visual Studio 2019** select **MainPage.xaml** |

## Step 3

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | Choose **View** then **Designer** from the **Menu** in **Visual Studio 2019** |

## Step 4

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | In the **Toolbox** of **Visual Studio 2019** from **Common XAML Controls**, double-click **Button** to add it to the **Design** View |

**MainPage.xaml** makes up the look of the application by placing **Controls** on the **Design** View

## Step 5

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | When the **Button** has been added to the **Design** View go to **Properties** set **HorizontalAlignment** to **Center**, **VerticalAlignment** to **Center** and **Content** to **Display** |

The **Button** will appear in the middle of the **Design** View with the **Content** of **Display** once the **Properties** have been set correctly

## Step 6

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | While still in the **Properties** select **Events** and then set **Click** to **Button\_Click** then either double-click on the text or press Enter once that has been typed in |

## Step 7

Finally, once done the **Code** View will be displayed and inside the **Button\_Click(...)** method the following should be entered:

|  |
| --- |
| \_ = new Windows.UI.Popups.MessageDialog("Hello World").ShowAsync(); |

The **Button\_Click(...)** method should then appear as follows:

private void Button\_Click(object sender, RoutedEventArgs e)

{

\_ = new Windows.UI.Popups.MessageDialog("Hello World").ShowAsync();

}

Clicking on the **Button** the **Event** of Button\_Click(...) will be triggered and this display a **MessageDialog** with the Text **Hello World**

## Step 8

|  |  |
| --- | --- |
|  | That completes the **Universal Windows Platform** Application, in **Visual Studio 2019** select **Local Machine** to run the Application |

## Step 9

Once the running, you can click **Display** to show the **MessageDialog** and dismiss it with **Close**

A screenshot of a cell phone

Description automatically generated

## Step 10

|  |  |
| --- | --- |
| A picture containing object  Description automatically generated | To Exit the Application, select the **Close** button in the top right of the Application |

**Command Bar** is where **AppBarButton** Controls can be added, these allow a standard-looking interface for applications to perform actions or access options

## Step 1

|  |  |
| --- | --- |
|  | Follow **Setup and Start** on how to Install and/or Get Started with **Visual Studio 2019** if not already or in **Windows 10** choose **Start**, find and select **Visual Studio 2019** then from the **Get started** screen select **Create a new project** |
| A screenshot of a cell phone  Description automatically generated | Then choose **Blank App (Universal Windows)** and select **Next** and then in **Configure your new project** enter the **Project name** as **CommandBar** and select **Create** |
| A screenshot of a social media post  Description automatically generated | Finally, in **New Universal Windows Platform Project** pick the **Target version** and **Minimum version** to be at least **Windows 10, version 1903 (10.0; Build 18362)** and then select **OK** |

**Target Version** will control the most recent features of **Windows 10** your application can use. To make sure you always have the most recent version, check for any **Notifications** or **Updates** in **Visual Studio 2019**

## Step 2

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | In the **Solution Explorer** of **Visual Studio 2019** select **MainPage.xaml** |

## Step 3

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | Choose **View** then **Designer** from the **Menu** in **Visual Studio 2019** |

## Step 4

In the **Design** View and **XAML** View of **Visual Studio 2019** will be displayed, and in this between the **Grid** and **/Grid** elements enter the following **XAML**:

|  |
| --- |
| <CommandBar IsOpen="True" IsSticky="True" VerticalAlignment="Bottom">  <CommandBar.SecondaryCommands>  <AppBarButton Name="Hide" Icon="Cancel" Label="Hide Other"  Visibility="Collapsed" Click="Show\_Click"/>  </CommandBar.SecondaryCommands>  <AppBarButton Name="Show" Icon="Accept" Label="Show Other"  Click="Show\_Click"/>  </CommandBar> |

**CommandBar** is a Control that can contain **AppBarButton** that will be displayed o show the main toolbar of a **Universal Windows Platform** Application in **Windows 10**

## Step 5

|  |  |
| --- | --- |
|  | Choose **View** then **Code** from the **Menu** in **Visual Studio 2019** |

## Step 6

Once in the **Code** View, below the end of public MainPage() { ... } the following Code should be entered:

|  |
| --- |
| private void Show\_Click(object sender, RoutedEventArgs e)  {  if (Hide.Visibility == Visibility.Collapsed)  {  Hide.Visibility = Visibility.Visible;  }  else  {  Hide.Visibility = Visibility.Collapsed;  }  } |

Show\_Click is an **Event** handler that will be triggered when **Hide Other** or **Show Other** is Clicked. This will if the Hide.Visibility is Visibility.Collapsed will set Hide.Visibility to Visibility.Visible or else it will set Hide.Visibility it to Visibility.Collapsed

## Step 7

|  |  |
| --- | --- |
|  | That completes the **Universal Windows Platform** Application, in **Visual Studio 2019** select **Local Machine** to run the Application |

## Step 8

Once the Application is running click **Show Other** to show an option on the bottom when **...** is tapped and tap **Hide Other** to hide this option again



## Step 9

|  |  |
| --- | --- |
| A picture containing object  Description automatically generated | To Exit the Application, select the **Close** button in the top right of the Application |

**Data Input** shows how to use **InputScope** for on-screen Keyboards where supported and and loading **ApplicationData**

## Step 1

|  |  |
| --- | --- |
|  | Follow **Setup and Start** on how to Install and/or Get Started with **Visual Studio 2019** if not already or in **Windows 10** choose **Start**, find and select **Visual Studio 2019** then from the **Get started** screen select **Create a new project** |
| A screenshot of a cell phone  Description automatically generated | Then choose **Blank App (Universal Windows)** and select **Next** and then in **Configure your new project** enter the **Project name** as **DataInput** and select **Create** |
| A screenshot of a social media post  Description automatically generated | Finally, in **New Universal Windows Platform Project** pick the **Target version** and **Minimum version** to be at least **Windows 10, version 1903 (10.0; Build 18362)** and then select **OK** |

**Target Version** will control the most recent features of **Windows 10** your application can use. To make sure you always have the most recent version, check for any **Notifications** or **Updates** in **Visual Studio 2019**

## Step 2

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | Choose **Project** then **Add New Item...** from the **Menu** in **Visual Studio 2019** |

## Step 3

|  |  |
| --- | --- |
| A close up of a logo  Description automatically generated | Then choose **Code File** from **Add New Item** in **Visual Studio 2019**, enter the **Name** as **Library.cs** and select **Add** |

## Step 4

In the **Code** View of **Library.cs** will be displayed and in this the following should be entered:

|  |
| --- |
| using Windows.Storage;  public class Library  {  public string LoadSetting(string key)  {  return (string)(ApplicationData.Current.LocalSettings.Values[key]  ?? string.Empty);  }  public void SaveSetting(string key, string value)  {  ApplicationData.Current.LocalSettings.Values[key] = value;  }  } |

There is a using statement to include functionality from Windows.Storage. LoadSetting(...) method takes a string parameter to return the LocalSettings with the key if present and using the **null coalesce** or ?? operator will be string.Empty if it is not. SaveSetting(...) method takes two string parameters to set the LocalSettings to be returned later with the key and value passed in

## Step 5

|  |  |
| --- | --- |
|  | In the **Solution Explorer** of **Visual Studio 2019** select **MainPage.xaml** |

## Step 6

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | Choose **View** then **Designer** from the **Menu** in **Visual Studio 2019** |

## Step 7

In the **Design** View and **XAML** View of **Visual Studio 2019** will be displayed, and in this between the **Grid** and **/Grid** elements enter the following **XAML**:

|  |
| --- |
| <StackPanel>  <TextBox Name="Email" PlaceholderText="Email"  InputScope="EmailSmtpAddress" Margin="20"/>  <TextBox Name="Website" PlaceholderText="Website"  InputScope="Url" Margin="20"/>  <TextBox Name="Telephone" PlaceholderText="Telephone"  InputScope="TelephoneNumber" Margin="20"/>  </StackPanel>  <CommandBar VerticalAlignment="Bottom">  <AppBarButton Icon="Page2" Label="New" Click="New\_Click"/>  <AppBarButton Icon="OpenLocal" Label="Open" Click="Open\_Click"/>  <AppBarButton Icon="Save" Label="Save" Click="Save\_Click"/>  </CommandBar> |

The first block of **XAML** comprises of **TextBox** Controls which will show the relevant on-screen Keyboard **InputScope** if supported. The second block of **XAML** is the **CommandBar** containing the operations

## Step 8

|  |  |
| --- | --- |
|  | Choose **View** then **Code** from the **Menu** in **Visual Studio 2019** |

## Step 9

Once in the **Code** View, below the end of **public MainPage() { ... }** the following Code should be entered:

|  |
| --- |
| Library library = new Library();  private void New\_Click(object sender, RoutedEventArgs e)  {  Email.Text = string.Empty;  Website.Text = string.Empty;  Telephone.Text = string.Empty;  }  private void Open\_Click(object sender, RoutedEventArgs e)  {  Email.Text = library.LoadSetting("Email");  Website.Text = library.LoadSetting("Website");  Telephone.Text = library.LoadSetting("Telephone");  }  private void Save\_Click(object sender, RoutedEventArgs e)  {  library.SaveSetting("Email", Email.Text);  library.SaveSetting("Website", Website.Text);  library.SaveSetting("Telephone", Telephone.Text);  } |

Below the **MainPage(...)** method an instance of the Library **Class** is created. In the New\_Click(...) **Event** handler the **TextBox** Controls have their **Text** property set to an Empty String. The Open\_Click(...) **Event** handler uses the LoadSetting method to load a value that has been previously Saved and the Save\_Click(...) **Event** handler will use SaveSetting to store a value to be loaded later

## Step 10

|  |  |
| --- | --- |
|  | That completes the **Universal Windows Platform** Application, in **Visual Studio 2019** select **Local Machine** to run the Application |

## Step 11

Once the Application is running you can then input some data such as an **Email Address**, **Website** and **Telephone Number** then store using the **Save** button and recall the data with the **Open** button or reset with the **New** button



## Step 12

|  |  |
| --- | --- |
| A picture containing object  Description automatically generated | To Exit the Application, select the **Close** button in the top right of the Application |

**Image Rotate** shows how to use a **Storyboard** to create a simple **DoubleAnimation** to rotate an image in the **X**, **Y** and **X** axis

## Step 1

|  |  |
| --- | --- |
|  | Follow **Setup and Start** on how to Install and/or Get Started with **Visual Studio 2019** if not already or in **Windows 10** choose **Start**, find and select **Visual Studio 2019** then from the **Get started** screen select **Create a new project** |
| A screenshot of a cell phone  Description automatically generated | Then choose **Blank App (Universal Windows)** and select **Next** and then in **Configure your new project** enter the **Project name** as **ImageRotate** and select **Create** |
| A screenshot of a social media post  Description automatically generated | Finally, in **New Universal Windows Platform Project** pick the **Target version** and **Minimum version** to be at least **Windows 10, version 1903 (10.0; Build 18362)** and then select **OK** |

**Target Version** will control the most recent features of **Windows 10** your application can use. To make sure you always have the most recent version, check for any **Notifications** or **Updates** in **Visual Studio 2019**

## Step 2

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | Choose **Project** then **Add New Item...** from the **Menu** in **Visual Studio 2019** |

## Step 3

|  |  |
| --- | --- |
| A close up of a logo  Description automatically generated | Then choose **Code File** from **Add New Item** in **Visual Studio 2019**, enter the **Name** as **Library.cs** and select **Add** |

## Step 4

In the **Code** View of **Library.cs** will be displayed and in this the following should be entered:

|  |
| --- |
| using System;  using Windows.UI.Xaml.Controls;  using Windows.UI.Xaml.Media.Animation;  public class Library  {  private bool \_rotating = false;  private Storyboard \_rotation = new Storyboard();  public void Rotate(string axis, ref Image target)  {  if (\_rotating)  {  \_rotation.Stop();  \_rotating = false;  }  else  {  DoubleAnimation animation = new DoubleAnimation  {  From = 0.0,  To = 360.0,  BeginTime = TimeSpan.FromSeconds(1),  RepeatBehavior = RepeatBehavior.Forever  };  Storyboard.SetTarget(animation, target);  Storyboard.SetTargetProperty(animation,  $"(UIElement.Projection).(PlaneProjection.Rotation{axis})");  \_rotation.Children.Clear();  \_rotation.Children.Add(animation);  \_rotation.Begin();  \_rotating = true;  }  }  } |

There is a using statement to include functionality needed for the application. Storyboard is used as part of the later DoubleAnimation which will animate between **0** and **360** and this will repeat Forever after 1 second. The PlaneProjection.Rotation value for each Axis is set on the UIElement which in this case is the Image named target and uses **String Interpolation Syntax** or $

## Step 5

|  |  |
| --- | --- |
|  | In the **Solution Explorer** of **Visual Studio 2019** select **MainPage.xaml** |

## Step 6

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | Choose **View** then **Designer** from the **Menu** in **Visual Studio 2019** |

## Step 7

In the **Design** View and **XAML** View of **Visual Studio 2019** will be displayed, and in this between the **Grid** and **/Grid** elements enter the following **XAML**:

|  |
| --- |
| <Grid>  <Grid.RowDefinitions>  <RowDefinition Height="Auto"/>  <RowDefinition Height="\*"/>  </Grid.RowDefinitions>  <TextBox Grid.Row="0" Name="Value" Margin="20"  InputScope="Url" KeyDown="Go\_KeyDown"/>  <Image Grid.Row="1" Margin="100" Stretch="Uniform" Name="Display">  <Image.Projection>  <PlaneProjection/>  </Image.Projection>  </Image>  </Grid>  <CommandBar VerticalAlignment="Bottom">  <AppBarButton Icon="RepeatAll" Label="Pitch" Click="Pitch\_Click"/>  <AppBarButton Icon="Rotate" Label="Roll" Click="Roll\_Click"/>  <AppBarButton Icon="Refresh" Label="Yaw" Click="Yaw\_Click"/>  </CommandBar> |

The first block of **XAML** the main user interface features a **TextBox**. The second block of **XAML** is is the **CommandBar** which contains **Pitch** – to rotate the X Axis, **Roll** – to rotate the Y Axis and **Yaw** – to rotate the Z Axis

## Step 8

|  |  |
| --- | --- |
|  | Choose **View** then **Code** from the **Menu** in **Visual Studio 2019** |

## Step 9

Once in the **Code** View, below the end of **public MainPage() { ... }** the following Code should be entered:

|  |
| --- |
| Library library = new Library();  private void Go\_KeyDown(object sender, KeyRoutedEventArgs e)  {  if (e.Key == Windows.System.VirtualKey.Enter)  {  Display.Source = new Windows.UI.Xaml.Media.Imaging  .BitmapImage(new Uri(Value.Text));  }  }  private void Pitch\_Click(object sender, RoutedEventArgs e)  {  library.Rotate("X", ref Display);  }  private void Roll\_Click(object sender, RoutedEventArgs e)  {  library.Rotate("Y", ref Display);  }  private void Yaw\_Click(object sender, RoutedEventArgs e)  {  library.Rotate("Z", ref Display);  } |

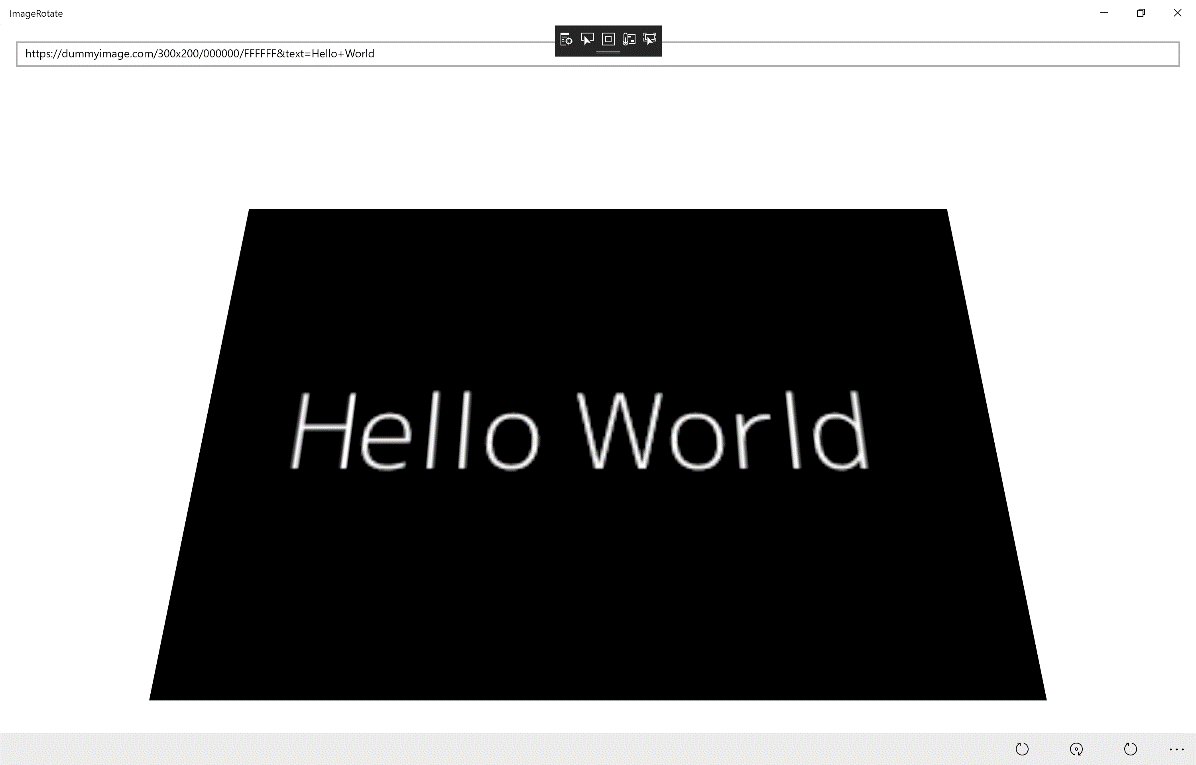
Below the **MainPage(...)** method an instance of the Library **Class** is created. In the Go\_KeyDown(...) **Event** handler the **Image** has the Source **property** set to the contents any URL entered in the **TextBox**, the Pitch\_Click(...), Roll\_Click(...) and Yaw\_Click(...) **event** handler will use the Rotate method to set which Axis the **Image** should be rotated by

## Step 10

|  |  |
| --- | --- |
|  | That completes the **Universal Windows Platform** Application, in **Visual Studio 2019** select **Local Machine** to run the Application |

## Step 11

Once the Application is running you can then type in the URL of any image e.g. <https://dummyimage.com/300x200/000000/FFFFFF&text=Hello+World> then press or tap Enter to load it, then use the **Pitch**, **Roll** or **Yaw** buttons to rotate the **Image**



## Step 12

|  |  |
| --- | --- |
| A picture containing object  Description automatically generated | To Exit the Application, select the **Close** button in the top right of the Application |