[C# WPF Tutorial #11 - FolderBrowserDialog - How to use Winforms In WPF!](https://www.youtube.com/watch?v=Heq8qve1Vts&list=PLih2KERbY1HHOOJ2C6FOrVXIwg4AZ-hk1&index=11)

A screenshot of a computer

Description automatically generated

MainWindow.xaml

using System.Windows;

using WinForms = System.Windows.Forms;

namespace \_11\_WPF\_Tutorial\_v1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

WinForms.FolderBrowserDialog dialog = new WinForms.FolderBrowserDialog();

dialog.InitialDirectory = "C:\\\_csc\_projects\\Complete\_csharp\_Programming\_Course\\WPF\_Iteration\_02\\11\_WPF\_Tutorial\_v1";

WinForms.DialogResult result = dialog.ShowDialog();

if(result == WinForms.DialogResult.OK)

{

// Do something with the result (OK)

string folder = dialog.SelectedPath;

// Returns the folder string and displays in MessageBox

WinForms.MessageBox.Show(folder);

}

else

{

// Do something else if Cancelled

}

}

}

}

MainWindow.xaml.cs

<Window x:Class="\_11\_WPF\_Tutorial\_v1.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:\_11\_WPF\_Tutorial\_v1"

mc:Ignorable="d"

Title="MainWindow" Height="150" Width="250">

<Grid>

<Button Name="btnFire"

Content="Fire"

Width="100"

Height="40"

Click="btnFire\_Click"

/>

</Grid>

</Window>

# Start new WPF solution in Visual Studio.

A black screen with white text

Description automatically generated

A black rectangular object with a black background

Description automatically generated

# Name it:

A screenshot of a computer

Description automatically generated

# Starting point. Code looks like this initially.

MainWindow.xaml

<Window x:Class="\_11\_WPF\_Tutorial\_v1.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:\_11\_WPF\_Tutorial\_v1"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800">

<Grid>

</Grid>

</Window>

# Add the following code to the .xaml file to create the GUI.

A screenshot of a computer

Description automatically generated

MainWindow.xaml

<Window x:Class="\_11\_WPF\_Tutorial\_v1.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:\_11\_WPF\_Tutorial\_v1"

mc:Ignorable="d"

Title="MainWindow" Height="150" Width="250">

<Grid>

<Button Name="btnFire"

Content="Fire"

Width="100"

Height="40"

/>

</Grid>

</Window>

# The MainWindow.xaml.cs code behind page will initially look like this:

MainWindow.xaml.cs

using System.Windows;

namespace \_11\_WPF\_Tutorial\_v1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

}

}

# Add a Click event handler to the button

MainWindow.xaml

<Window x:Class="\_11\_WPF\_Tutorial\_v1.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:\_11\_WPF\_Tutorial\_v1"

mc:Ignorable="d"

Title="MainWindow" Height="150" Width="250">

<Grid>

<Button Name="btnFire"

Content="Fire"

Width="100"

Height="40"

Click="btnFire\_Click"

/>

</Grid>

</Window>

# THIS DOES NOT WORK SO DON’T JUST DO THIS…

This is because we are using WPF and not accessing Windows.Forms. So Windows.Forms must be enabled before this. Code is the same but Windows Forms must be enabled in Properties.

MainWindow.xaml.cs

using System.Windows;

using System.Windows.Forms;

namespace \_11\_WPF\_Tutorial\_v1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

FolderBrowserDialog dialog = new FolderBrowserDialog();

dialog.ShowDialog();

}

}

}

# Correct method is this…

Right Click project and select Properties

A screenshot of a computer

Description automatically generated

Scroll down and TICK Enable Windows Forms for this project.

A black background with white text

Description automatically generated

Now add this code:

MainWindow.xaml.cs

using System.Windows;

using System.Windows.Forms;

namespace \_11\_WPF\_Tutorial\_v1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

FolderBrowserDialog dialog = new FolderBrowserDialog();

dialog.ShowDialog();

}

}

}

# FolderBrowserDialog works fine BUT…

BIG WARNING: When Windows Forms is enabled it has a lot of the same components as WPF so there will be ambiguity and it will not work.

You must call either Windows.Forms or WPF explicitly.

Best solution is to ALIAS the using System.Windows.Forms with something obvious.

The example below shows the changes to be made and also adds a MessageBox which is present in both Windows.Forms an WPF.

MainWindow.xaml.cs

using System.Windows;

using WinForms = System.Windows.Forms;

namespace \_11\_WPF\_Tutorial\_v1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

WinForms.FolderBrowserDialog dialog = new WinForms.FolderBrowserDialog();

dialog.ShowDialog();

DEMONSTATION PURPOSES ONLY

// Calls WPF version

MessageBox.Show("Called WPF version");

// Calls Windows.Forms version

WinForms.MessageBox.Show("Called Windows.Forms version");

}

}

}

# Add initialDirectory

Copy full path of Solution by right clicking on Solution in Solution Explorer:

A screenshot of a computer

Description automatically generated

My current solution is here:

C:\\_code\WPF\_v2\11\_WPF\_Tutorial\_v1\11\_WPF\_Tutorial\_v1.sln

And add the following code but remove the solution file name from the end.

dialog.InitialDirectory = "C:\\\_code\\WPF\_v2\\11\_WPF\_Tutorial\_v1";

MainWindow.xaml.cs

using System.Windows;

using WinForms = System.Windows.Forms;

namespace \_11\_WPF\_Tutorial\_v1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

WinForms.FolderBrowserDialog dialog = new WinForms.FolderBrowserDialog();

dialog.InitialDirectory = "C:\\\_Dev\\WPF\_v2\\11\_WPF\_Tutorial\_v1";

dialog.ShowDialog();

}

}

}

# Working with the returned data

MainWindow.xaml.cs

using System.Windows;

using WinForms = System.Windows.Forms;

namespace \_11\_WPF\_Tutorial\_v1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

WinForms.FolderBrowserDialog dialog = new WinForms.FolderBrowserDialog();

dialog.InitialDirectory = "C:\\\_Dev\\WPF\_v2\\11\_WPF\_Tutorial\_v1";

WinForms.DialogResult result = dialog.ShowDialog();

if(result == WinForms.DialogResult.OK)

{

// Do something with the result (OK)

string folder = dialog.SelectedPath;

// Returns the folder string and displays in MessageBox

WinForms.MessageBox.Show(folder);

}

else

{

// Do something else if cancelled

}

}

}

}