# How to bind command to a button in the DataTemplate in Universal Windows Platform apps

This sample demonstrates how bind command to a button in the DataTemplate in Universal Windows Platform apps.

This sample is upgraded from the following universal windows app sample:

<https://code.msdn.microsoft.com/How-to-bind-command-to-a-299f7759>

**Sample prerequisites**

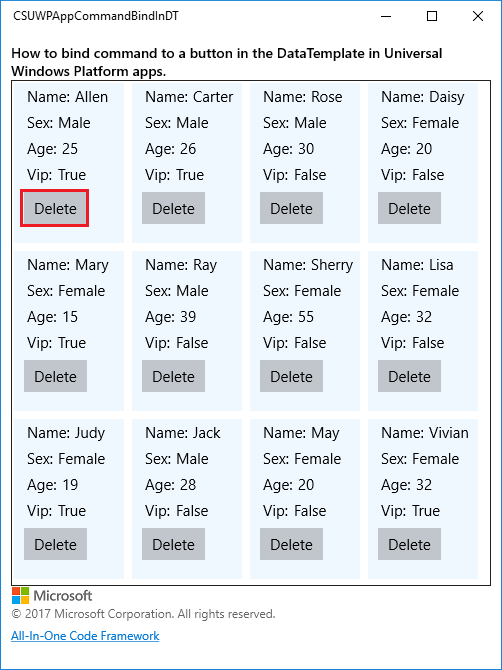
* Visual Studio 2017 or above [[Visual Studio Home Page](https://www.visualstudio.com/)]

## Building the sample

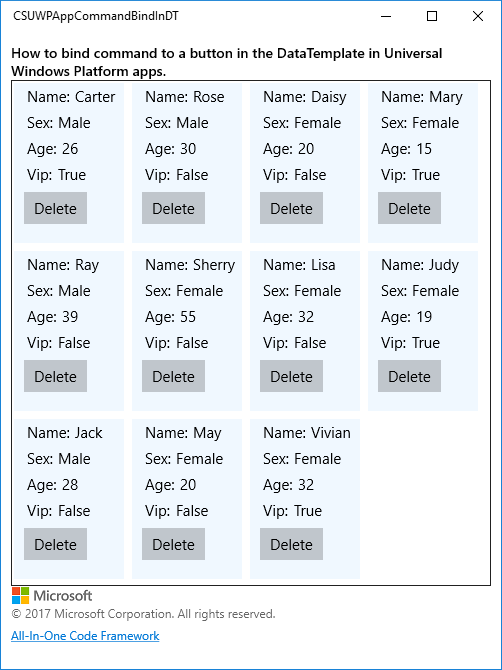
* Open the sample solution “CSUWPContextMenu.sln” using Visual Studio
* Right click the project “CSUWPContextMenu” and select Restore Packages
* Press **F6** Key or select **Build -> Build Solution** from the menu to build the sample.

## Running the sample

* Open the sample solution using Visual Studio, select **Local Machine** in the tool bar then press F5 Key or select Debug -> Start Debugging from the menu.



* Click Delete button of the first item in the gridview.



**Using the code**

DataTemplate of GridView.

<GridView.ItemTemplate>

<DataTemplate>

<Grid Margin="10,0,0,10" Width="100" Height="150">

<StackPanel>

<StackPanel Orientation="Horizontal" Margin="3,3,0,3">

<TextBlock Text="Name:" Style="{StaticResource AppBodyTextStyle}" Margin="0,0,5,0"/>

<TextBlock Text="{Binding Name}" Style="{StaticResource AppBodyTextStyle}"/>

</StackPanel>

<StackPanel Orientation="Horizontal" Margin="3,3,0,3">

<TextBlock Text="Sex:" Style="{StaticResource AppBodyTextStyle}" Margin="0,0,5,0"/>

<TextBlock Text="{Binding Sex, Converter={StaticResource SexConverter}}" Style="{StaticResource AppBodyTextStyle}"/>

</StackPanel>

<StackPanel Orientation="Horizontal" Margin="3,3,0,3">

<TextBlock Text="Age:" Style="{StaticResource AppBodyTextStyle}" Margin="0,0,5,0"/>

<TextBlock Text="{Binding Age}" Style="{StaticResource AppBodyTextStyle}"/>

</StackPanel>

<StackPanel Orientation="Horizontal" Margin="3,3,0,3">

<TextBlock Text="Vip:" Style="{StaticResource AppBodyTextStyle}" Margin="0,0,5,0"/>

<TextBlock Text="{Binding Vip}" Style="{StaticResource AppBodyTextStyle}"/>

</StackPanel>

<Button Content="Delete" Margin="0,5,0,0" Command="{Binding DataContext.DeleteCommand, ElementName=CustomerGridView}" CommandParameter="{Binding}"/>

</StackPanel>

</Grid>

</DataTemplate>

</GridView.ItemTemplate>

DelegateCommand

class DelegateCommand : ICommand

{

private Action<object> execute;

private Func<object, bool> canExecute;

public DelegateCommand(Action<object> execute)

{

this.execute = execute;

this.canExecute = (x) => { return true; };

}

public DelegateCommand(Action<object> execute, Func<object, bool> canExecute)

{

this.execute = execute;

this.canExecute = canExecute;

}

public bool CanExecute(object parameter)

{

return canExecute(parameter);

}

public event EventHandler CanExecuteChanged;

public void RaiseCanExecuteChanged()

{

if (CanExecuteChanged != null)

{

CanExecuteChanged(this, EventArgs.Empty);

}

}

public void Execute(object parameter)

{

execute(parameter);

}

}

**More information**

[ICommand interface](https://docs.microsoft.com/en-us/uwp/api/Windows.UI.Xaml.Input.ICommand)