

CUS

Microsoft Announces Visual Studio 2019



C# Corner



WPF ICommand in MVVM

ASK A QUESTION

CONTRIBUTE



Prabhat Kumar

Jan 09 2013

Article

5

10

277.4k

[WPFMVVMCommand.zip](#)
[Download Free Office API](#)

## Introduction

Commands provide a mechanism for the view to update the model in the MVVM architecture. Commands provide a way to

```
bool CanExecute(object parameter);
void Execute(object parameter);
event EventHandler CanExecuteChanged;
```

The Execute method is only invoked when CanExecute returns true. In case the CanExecute method returns false then the binding control is disabled automatically.

In order to know the CanExecute value, listen to the CanExecuteChanged event, that may vary based on the parameter passed.

## Using Code

### Use of ICommand

The ICommand interface is generally used in the MVVM architecture. Here in the Button control the Command property is bound to the "UpdateCommand". Since UpdateCommand is nothing but an ICommand instance, while loading the window it will check the CanExecute return value and if it returns true then it will enable the button control and the Execute method is ready to be used otherwise the button control is disabled.

```
<Button x:Name="btnUpdate" Width="100" Height="20" HorizontalAlignment="Center" Grid.Row="1" Content="Update"
Command="{Binding Path=UpdateCommand}"/>
```

### Passing a parameter to the CanExecute and Execute methods

A parameter can be passed through the "CommandParameter" property. Once the button is clicked the selected address value is passed to the ICommand.Execute method.

The CommandParameter is sent to both CanExecute and Execute events.

CUS

Microsoft Announces Visual Studio 2019



ASK A QUESTION

CONTRIBUTE

## Tasks of CanExecuteChanged

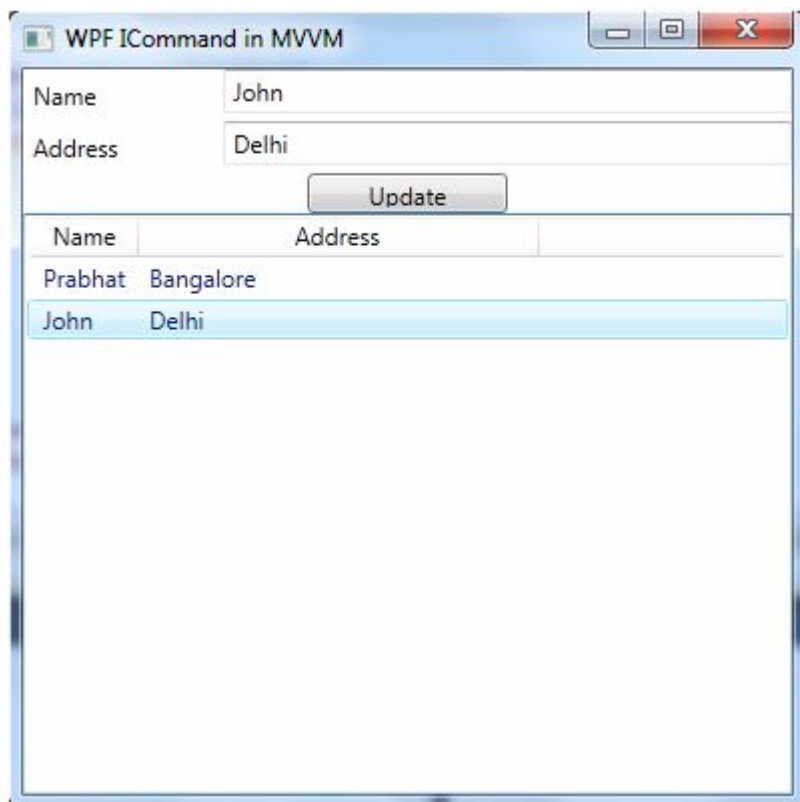
CanExecuteChanged notifies any command sources (like a Button or CheckBox) that are bound to that ICommand that the value returned by CanExecute has changed. Command sources care about this because they generally need to update their status accordingly (eg. a Button will disable itself if CanExecute() returns false).

## Listen CanExecuteChanged

The CommandManager.RequerySuggested event is raised whenever the CommandManager thinks that something has changed that will affect the ability of commands to execute. This might be a change of focus, for example. This event happens to fire a lot.

```
public event EventHandler CanExecuteChanged
{
    add { CommandManager.RequerySuggested += value; }
    remove { CommandManager.RequerySuggested -= value; }
}
```

## UI



## Model

Create a folder named Model and a class named Person. Implement INotifyPropertyChanged and override the PropertyChanged event. Define two properties, Name and Address.

CUS

Microsoft Announces Visual Studio 2019



ASK A QUESTION

CONTRIBUTE

```
private string name;
private string address;
```

```
public event PropertyChangedEventHandler PropertyChanged;
public void OnPropertyChanged(string propertyName)
{
    if (PropertyChanged != null)
    {
        PropertyChanged(this, new PropertyChangedEventArgs(propertyName));
    }
}

public string Name
{
    get

    set
    {
        name = value;
        OnPropertyChanged("Name");
    }
}

public string Address
{
    get
    {
        return address;
    }
    set
    {
        address = value;
        OnPropertyChanged("Address");
    }
}
}
```

## ViewModel

Create a folder named ViewModel and a class named PersonViewModel. Initialize a List object of type person. Expose this object through the Persons property. Also expose an ICommand instance in this class.

Define a class Updater and implement ICommand interface.

CUS  
Microsoft Announces Visual Studio 2019


[ASK A QUESTION](#)
[CONTRIBUTE](#)

```
private IList<Person> _personList;
public PersonViewModel()
{
    _personList = new List<Person>()
    {
        new Person(){Name="Prabhat", Address="Bangalore"},
        new Person(){Name="John",Address="Delhi"}
    };
}
public IList<Person> Persons
{
    get { return _personList; }
    set { _personList = value; }
}
private ICommand mUpdater;
public ICommand UpdateCommand

{
    if (mUpdater == null)
        mUpdater = new Updater();
    return mUpdater;
}
set
{
    mUpdater = value;
}
}
class Updater : ICommand
{
    #region ICommand Members

    public bool CanExecute(object parameter)
    {
        return true;
    }
    public event EventHandler CanExecuteChanged
    {
        add { CommandManager.RequerySuggested += value; }
        remove { CommandManager.RequerySuggested -= value; }
    }

    public void Execute(object parameter)
    {

```


[ASK A QUESTION](#)
[CONTRIBUTE](#)

## View

Create a view folder and a Person.xaml class. Bind the command to the button control. Once the button is clicked, the command parameter is passed to the Execute method. You can also bind a command to any control that supports the command property. the TextBox is bound with listView selectedItem.

```
<Grid>
  <Grid.RowDefinitions>
    <RowDefinition Height="auto"/>
    <RowDefinition Height="auto"/>
    <RowDefinition Height="*/"/>
  </Grid.RowDefinitions>
  <Grid>
    <Grid.ColumnDefinitions>

    </Grid.ColumnDefinitions>
    <Grid.RowDefinitions>
      <RowDefinition Height="auto"> </RowDefinition>
      <RowDefinition Height="auto"> </RowDefinition>
    </Grid.RowDefinitions>
    <Label x:Name="lblName" Content="Name" Grid.Row="0" Grid.Column="0" VerticalAlignment="Top"> </Label>
    <TextBox x:Name="txtName" Grid.Row="0" Grid.Column="1" VerticalAlignment="Top" Text="{Binding
ElementName=lstPerson, Path=SelectedItem.Name}">
    </TextBox>
    <Label x:Name="lblAddress" Content="Address" Grid.Row="1" Grid.Column="0" VerticalAlignment="Top">
</Label>
    <TextBox x:Name="txtAddress" Grid.Row="1" Grid.Column="1" VerticalAlignment="Top" Text="{Binding
ElementName=lstPerson
Path=SelectedItem.Address}"> </TextBox>
  </Grid>
  <Button x:Name="btnUpdate" Width="100" Height="20" HorizontalAlignment="Center" Grid.Row="1"
Content="Update"
  Command="{Binding Path=UpdateCommand}" CommandParameter="{Binding ElementName=lstPerson,
Path=SelectedItem.Address}"> </Button>
  <ListView x:Name="lstPerson" Grid.Row="2" ItemsSource="{Binding Persons}">

  <ListView.View>
    <GridView>
      <GridViewColumn Header="Name" DisplayMemberBinding="{Binding Name}" />

      <GridViewColumn Header="Address" Width="200" DisplayMemberBinding="{Binding Address}" />
    </GridView>
  </ListView.View>
</ListView>
```

CUS

Microsoft Announces Visual Studio 2019



App.xaml.cs

ASK A QUESTION

CONTRIBUTE

Override the OnStartup method of the Application class. Initialize a view and viewModel class and wrap a viewModel to the view by the DataContext property.

```
protected override void OnStartup(StartupEventArgs e)
{
    base.OnStartup(e);
    View.Person person = new View.Person();
    PersonViewModel personViewModel = new PersonViewModel();
    person.DataContext = personViewModel;
    person.Show();
}
```

### Summary

3. MVVM architecture flow.
4. How to pass parameter in CanExecute and Execute method.
5. How to listen to CanExecuteChanged event.

### Point Of Interest

Instead of events, commands have been used in the MVVM architecture. We cannot pass a parameter to the event but we can pass a parameter through a command.

F

Command in MVVM

WPF ICommand



Prabhat Kumar

<https://www.c-sharpcorner.com/members/prabhat-kumar6>

1092

975.4k

5

10



Type your comment here and press Enter Key (Minimum 10 characters)

How to clear text boxes when entered

Stuart Welsh

Oct 15

CUS

Microsoft Announces Visual Studio 2019



Using the MVVM pattern, all your textboxes are bound to corresponding property in the Model or ViewModel. You can use `<EnterKeyCommand"/>` and clear all the properties when the EnterKeyCommand will be executed

[Nick Var](#)

Nov 27, 2017

1541 23 0

0



thanks

[Mustafa KORUR](#)

Oct 20, 2014

1562 2 490

0 0 Reply



Newbie to WPF with MVVM , Could somebody let me know where to put INotifyPropertyChanged? Either in ViewModel or Model

[Muneer Akhtar](#)

Jul 08, 2014

1563 1 0

0 0 Reply



public void Execute(object parameter) {} never gets called

Feb 15, 2014



Thanks a lot prabhat... this example helped me to understand many things..

[Priyank Thakkar](#)

Jan 31, 2014

1563 1 0

0 0 Reply



Expose an event from your viewmodel and trigger it from your view

[Prabhat Kumar](#)

Sep 14, 2013

1092 504 975.4k

0 0 Reply



can we call a method which takes no parameter from view through command in viewmodel

[Sekhar Babu](#)

Sep 13, 2013

1459 105 369

0 0 Reply



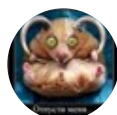
Sample is not working

[jenifer nishanth](#)

Jul 09, 2013

1562 2 430

0 0 Reply



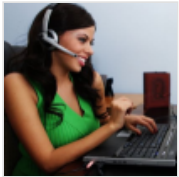
Thanks for article!

[Nick Var](#)

Feb 16, 2013

1541 23 0

0 0 Reply



### Hire a remote developer

Looking to add more bandwidth to your software team? Web Developers, designers, testers are now available on demand. Flexible hours, very competitive rates, expert team and High Quality work.

### TRENDING UP

- 01 ASP.NET Web API Using MVC And jQuery To Upload And Download Files - Part Seven
- 02 ASP.NET Web API Using MVC And HttpClient To Upload And Download Files - Part Eight
- 03 Building A Blockchain In .NET Core - Basic Blockchain
- 05 Introduction To Blockchain, Ethereum And Smart Contracts
- 06 ASP.NET Core Blazor CRUD Using Entity Framework And Web API
- 07 What Is ML.NET
- 08 Best Practices Of Writing C# Code
- 09 Comparison Between React And Angular
- 10 Azure Machine Learning Client And Server Applications

[View All](#) 



CUS

Microsoft Announces Visual Studio 2019

[ASK A QUESTION](#)[CONTRIBUTE](#)[Philadelphia](#)[New York](#)[London](#)[Delhi](#)

## JOIN C# CORNER

and millions of developer friends worldwide.

[Sign Up](#)[Learn ASP.NET MVC](#) [Learn ASP.NET Core](#) [Learn Python](#) [Learn JavaScript](#) [Learn Xamarin](#)[Learn Oracle](#) [More...](#)[Home](#) [Events](#) [Consultants](#) [Jobs](#) [Career Advice](#) [Stories](#) [Partners](#)[About Us](#) [Contact Us](#) [Privacy Policy](#) [Terms](#) [Media Kit](#) [Sitemap](#) [Report a Bug](#) [FAQ](#)

©2018 C# Corner. All contents are copyright of their authors.