



FREE eBook

LEARNING

telerik

Free unaffiliated eBook created from
Stack Overflow contributors.

#telerik

Table of Contents

About.....	1
Chapter 1: Getting started with telerik.....	2
Remarks.....	2
Examples.....	2
Installation or Setup.....	2
Chapter 2: Telerik WPF RadGridView.....	3
Examples.....	3
RowDoubleClick.....	3
Credits.....	5

About

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [telerik](#)

It is an unofficial and free telerik ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official telerik.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapter 1: Getting started with telerik

Remarks

This section provides an overview of what telerik is, and why a developer might want to use it.

It should also mention any large subjects within telerik, and link out to the related topics. Since the Documentation for telerik is new, you may need to create initial versions of those related topics.

Examples

Installation or Setup

Detailed instructions on getting telerik set up or installed.

Read **Getting started with telerik** online: <https://riptutorial.com/telerik/topic/7577/getting-started-with-telerik>

Chapter 2: Telerik WPF RadGridView

Examples

RowDoubleClick

It's a common request to be able to handle a double-click on a row in a RadGridView. The solution proposed by Telerik (<http://demos.telerik.com/silverlight/#GridView/ClickEvents>) is based around code behind:

```
this.grid.AddHandler(GridViewCellBase.CellDoubleClickEvent, new  
EventHandler<RadRoutedEventArgs>(OnCellDoubleClick), true);
```

if you're using MVVM, you would probably prefer to attach an ICommand, which will be passed the DataContext of the row that has been clicked on, as a parameter. Here's how to do that. XAML:

```
<telerik:RadGridView  
    ItemsSource="{Binding Rows}"  
    IsReadOnly="True"  
    ShowGroupPanel="False"  
    ShowColumnFooters="True"  
    local:RadGridViewAttachedProperties.RowDoubleClickCommand="{Binding DoubleClickRow}"/>
```

AttachedProperty:

```
public static class RadGridViewAttachedProperties  
{  
    public static readonly DependencyProperty RowDoubleClickCommandProperty =  
        DependencyProperty.RegisterAttached("RowDoubleClickCommand", typeof(ICommand),  
        typeof(RadGridViewAttachedProperties), new UIPropertyMetadata(null,  
        OnRowDoubleClickCommandChanged));  
  
    public static ICommand GetRowDoubleClickCommand(DependencyObject obj)  
    {  
        return (ICommand)obj.GetValue(RowDoubleClickCommandProperty);  
    }  
  
    public static void SetRowDoubleClickCommand(DependencyObject obj, ICommand value)  
    {  
        obj.SetValue(RowDoubleClickCommandProperty, value);  
    }  
  
    private static void OnMouseDoubleClick(object sender, RoutedEventArgs e)  
    {  
        var obj = sender as DependencyObject;  
        if (obj != null)  
        {  
            var command = GetRowDoubleClickCommand(obj);  
            if (command != null)  
            {  
                var frameworkElement = e.OriginalSource as FrameworkElement;  
                var dataContext = frameworkElement?.DataContext;  
                if (command.CanExecute(dataContext))
```

```

        {
            command.Execute(dataContext);
        }
    }
}

private static void OnRowDoubleClickCommandChanged(DependencyObject d,
DependencyPropertyChangedEventArgs e)
{
    var controlIsTheRightType = d as RadGridView;
    if (controlIsTheRightType == null)
    {
        return;
    }

    controlIsTheRightType.MouseDoubleClick += OnMouseDoubleClick;
}
}

```

Example ViewModel (uses a Telerik DelegateCommand as the command; but you can do what you like):

```

public class MainWindowViewModel
{
    public MainWindowViewModel()
    {
        Rows = new ObservableCollection<string>();
        Rows.Add("kljshndfoa");
        DoubleClickRow = new DelegateCommand(e => MessageBox.Show("Hello " + e));
    }

    public ICommand DoubleClickRow { get; }

    public ObservableCollection<string> Rows { get; }
}

```

NOTE: you will know what the type of things in your RadGridView's ItemsSource are. In the Execute method of your ICommand, make sure you check what the type of the parameter to the command is, since clicking on the header or footer will also trigger this command; but in those scenarios, the parameter will not be one of your items.

Read Telerik WPF RadGridView online: <https://riptutorial.com/telerik/topic/7903/telerik-wpf-radgridview>

Credits

S. No	Chapters	Contributors
1	Getting started with telerik	Community
2	Telerik WPF RadGridView	Richardissimo