

ComponentOne

Sparkline for ASP.NET WebForms

Cover Page Info

ComponentOne, a division of GrapeCity

201 South Highland Avenue, Third Floor
Pittsburgh, PA 15206 USA

Website: <http://www.componentone.com>

Sales: sales@componentone.com

Telephone: 1.800.858.2739 or 1.412.681.4343 (Pittsburgh, PA USA Office)

Trademarks

The ComponentOne product name is a trademark and ComponentOne is a registered trademark of GrapeCity, Inc. All other trademarks used herein are the properties of their respective owners.

Warranty

ComponentOne warrants that the media on which the software is delivered is free from defects in material and workmanship, assuming normal use, for a period of 90 days from the date of purchase. If a defect occurs during this time, you may return the defective media to ComponentOne, along with a dated proof of purchase, and ComponentOne will replace it at no charge. After 90 days, you can obtain a replacement for the defective media by sending it and a check for \$2 5 (to cover postage and handling) to ComponentOne.

Except for the express warranty of the original media on which the software is delivered is set forth here, ComponentOne makes no other warranties, express or implied. Every attempt has been made to ensure that the information contained in this manual is correct as of the time it was written. ComponentOne is not responsible for any errors or omissions. ComponentOne's liability is limited to the amount you paid for the product. ComponentOne is not liable for any special, consequential, or other damages for any reason.

Copying and Distribution

While you are welcome to make backup copies of the software for your own use and protection, you are not permitted to make copies for the use of anyone else. We put a lot of time and effort into creating this product, and we appreciate your support in seeing that it is used by licensed users only.

Table of Contents

Overview	2
Help with ASP.NET Web Forms Edition	2
Quick Start: Add Data and Customize Appearance	3-4
Design-Time Support	5
Explore Sparkline	6
Add Animation	6-7
Chart Types	7-9
Data Binding	9-10
Add to Grid	10-12
Display Negative Values	12-15

Overview

[Sparkline](#) for ASP.NET Web Forms allows you to display simple charts for data visualization. They are useful to represent multiple data points in a smaller region. You can also add multiple sparkline charts inline or in a single row of a grid.

Key Features

- **Chart Types:** Supports three different chart types: line, column, area charts.
- **Data Binding:** You can populate sparklines in code using **DataSource** and **DataBind** properties.
- **Grid Representation:** Display sparklines in the C1GridView control.
- **Value Axis:** When set to True, [Sparkline](#) displays negative values below the value axis and positive values above it.

Help with ASP.NET Web Forms Edition

Getting Started

For information on installing **ComponentOne Studio ASP.NET Web Forms Edition**, licensing, technical support, namespaces and creating a project with the control, please visit [Getting Started with ASP.NET Web Forms Edition](#).

Quick Start: Add Data and Customize Appearance

This quick start describes how to get started with [Sparkline](#) for ASP.NET Web Forms. In this topic you will learn how to add a [Sparkline](#) control to the page, change the appearance, add a list of data points, and observe the [Sparkline](#)'s run-time behavior.

Visual Studio 2012 was used in this example, the steps may be slightly different in other versions of Visual Studio.

Step 1 of 3: Add Control to the Form

Complete the steps below to create a simple application and add the Sparkline control to it.

1. In Visual Studio, create a new **ASP.Net Web Application** and add a new **Web Form**.
2. Locate the [Sparkline](#) control in the **ToolBox** and drag it to the Web Form. If you cannot find the control in the ToolBox, right click and select **Choose Items** to locate the [Sparkline](#) control in the **Choose Toolbox Items** dialog box.

[Back to Top](#)

Step 2 of 3: Customize the Control

Complete the following steps to change the appearance and behavior of the sparkline you created in the last step:

1. Right click the [Sparkline](#) and select **Properties**. In the **Properties Window**, set the following:
 - o **Width** = 200px
 - o **Height** = 150px
 - o **Theme** = midnight
2. Open the C1Sparkline Tasks menu and select **SeriesList**. This opens the [SparklineSeries Collection Editor](#).
3. From the right window, select the Type as Area. By default the type Line is set.

In Source View

Modify the `<cc1:C1Sparkline></cc1:C1Sparkline>` tags, to customize the control:

```
<cc1:C1Sparkline ID="C1Sparkline1" runat="server" height="150" Width="200" >
  <SeriesList>
    <cc1:SparklineSeries Type="Area" >
      </cc1:SparklineSeries>
    </SeriesList>
</cc1:C1Sparkline>
```

In Code

Add the following code to the **Page_Load** event, to customize the Sparkline control.

To write code in C#

```
C1Sparkline1.Theme = "midnight";
C1Sparkline1.Height = 150;
C1Sparkline1.Width=200;
```

In this Topic

- [Step 1 of 3: Add Control to the Form](#)
- [Step 2 of 3: Customize the Control](#)
- [Step 3 of 3: Add data to the Control](#)

To write code in VB

```
C1Sparkline1.Theme = "midnight"  
C1Sparkline1.Height = 150  
C1Sparkline1.Width=200
```

Back to Top

Step 3 of 3: Add data to the Control

Complete the following steps to add data to the control.

In Source View

Modify the `<cc1:C1Sparkline></cc1:C1Sparkline>` tags, to customize the control:

```
<SeriesList>  
  <cc1:SparklineSeries Data="33,11,15,26,16,27,37,-13,8,-8,-3,  
  17,0,22,-13,-29,19,8">  
  </cc1:SparklineSeries>  
</SeriesList>
```

In Code

Add the following code to the **Page_Load** event, to add data.

To write code in C#

```
double[] data = { 33, 11, 15, 26, 16, 27, 37, -13, 8, -8, -3, 17, 0, 22, -13,  
  -29, 19,8}; C1Sparkline1.SeriesList[0].Data = data;
```

To write code in Visual Basic

```
Dim data As Double() = {33, 11, 15, 26, 16, 27, 37, -13, 8, -8, -3, 17, 0, 22,  
  -13, -29, 19, 8}  
C1Sparkline1.SeriesList(0).Data = data
```

What You've Accomplished

When you run the project, the result appears as the image below:



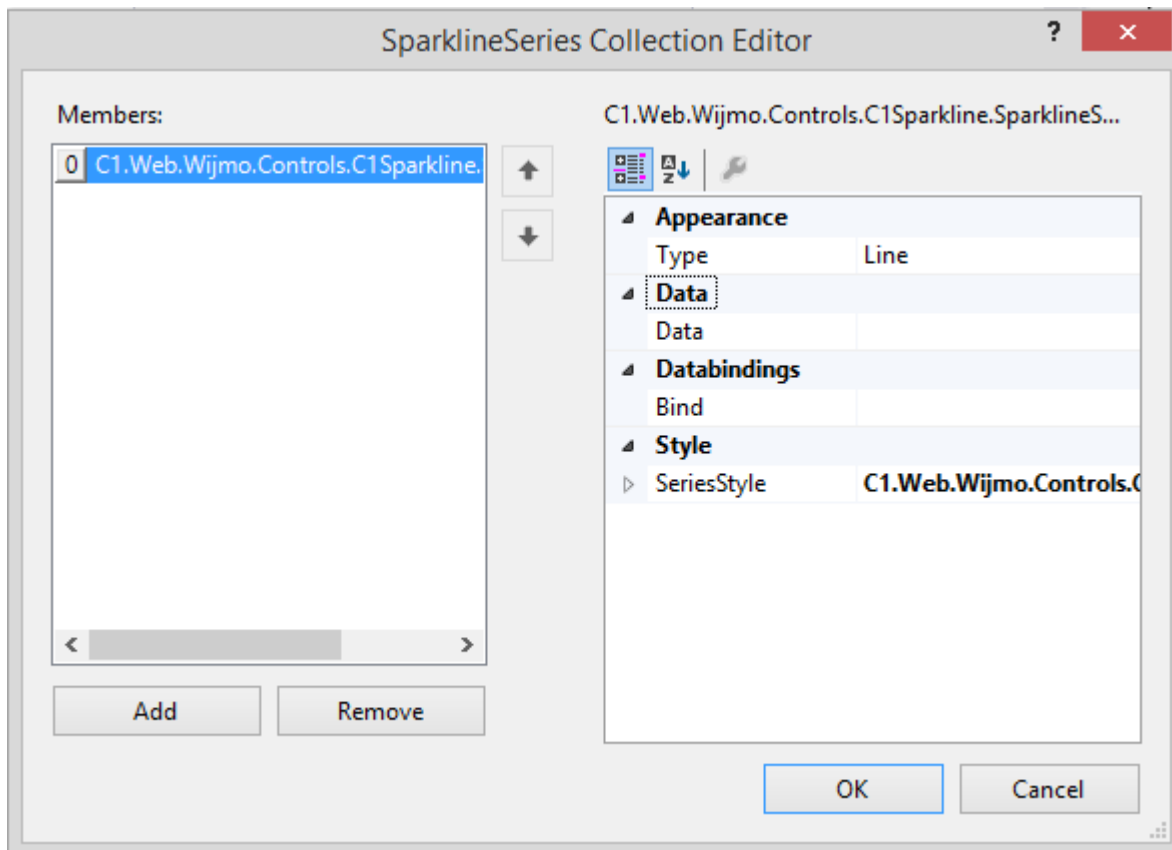
Back to Top

Design-Time Support

This topic describes how to use Sparkline's design-time features to configure the control.

To access the **C1Sparkline Tasks** menu, click the smart tag in the upper-right corner of the control. Go to **SeriesList**, this will open the SparklineSeries Collection Editor.

SparklineSeries Collection Editor



The **Add** button adds a new Sparkline [SeriesList](#).

The **Remove** button removes the selected Sparkline [SeriesList](#).

Explore Sparkline

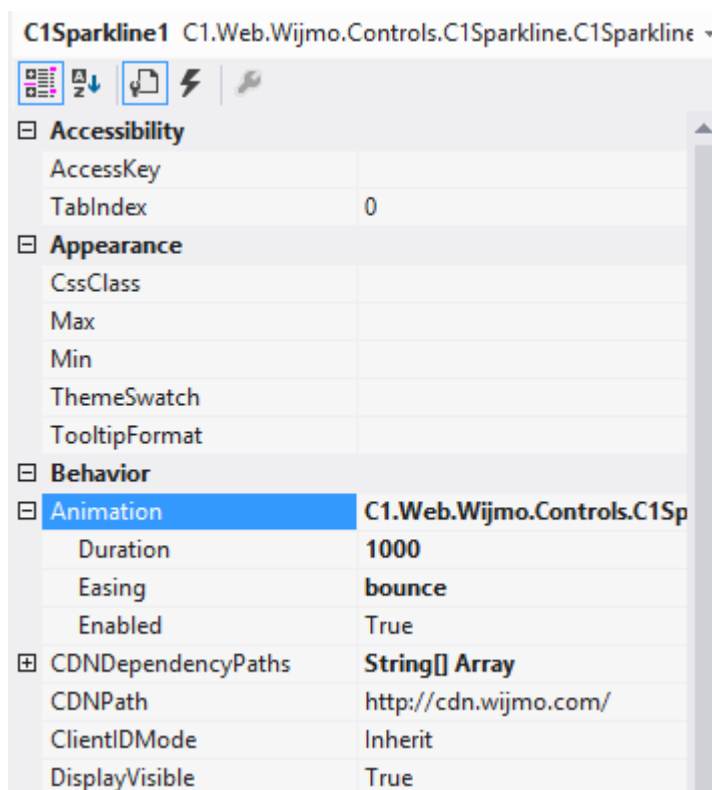
Add Animation

You can add animation to the Sparkline control by setting animation duration or easing type through the [Animation](#) property.

- **Enabled**: You can display sparkline without any animation by setting this property as **False**. By default, the property is set **True**.
- **Duration**: Set the amount of time it takes to display a sparkline. By default the value is **2000** milliseconds.
- **Easing**: Change the easing type for the animation. By default, Easing is set to **Linear**.

In the Designer

1. Right click the Sparkline control and select **Properties**. This opens the **Properties Window**.
2. In the **Properties Window**, expand the [Animation](#) property, and change the [Duration](#) to 1000 and [Easing](#) to bounce.



In Source View

Modify the [Animation](#) property within the `<cc1:C1Sparkline>` tag, to customize the animation of the control.

```
<cc1:C1Sparkline ID="C1Sparkline1" runat="server"
Animation-Duration="1000" Animation-Easing="bounce" >
</cc1:C1Sparkline>
```

In Code

Add the following code to the Page_Load event to customize the animation of the control.

To write code in C#

```
C1Sparkline1.Animation.Duration = 1000;  
C1Sparkline1.Animation.Easing = "bounce";
```

To write code in Visual Basic

```
C1Sparkline1.Animation.Duration = 1000  
C1Sparkline1.Animation.Easing = "bounce"
```


 To disable the Animation, set **Animation.Enabled** to False.

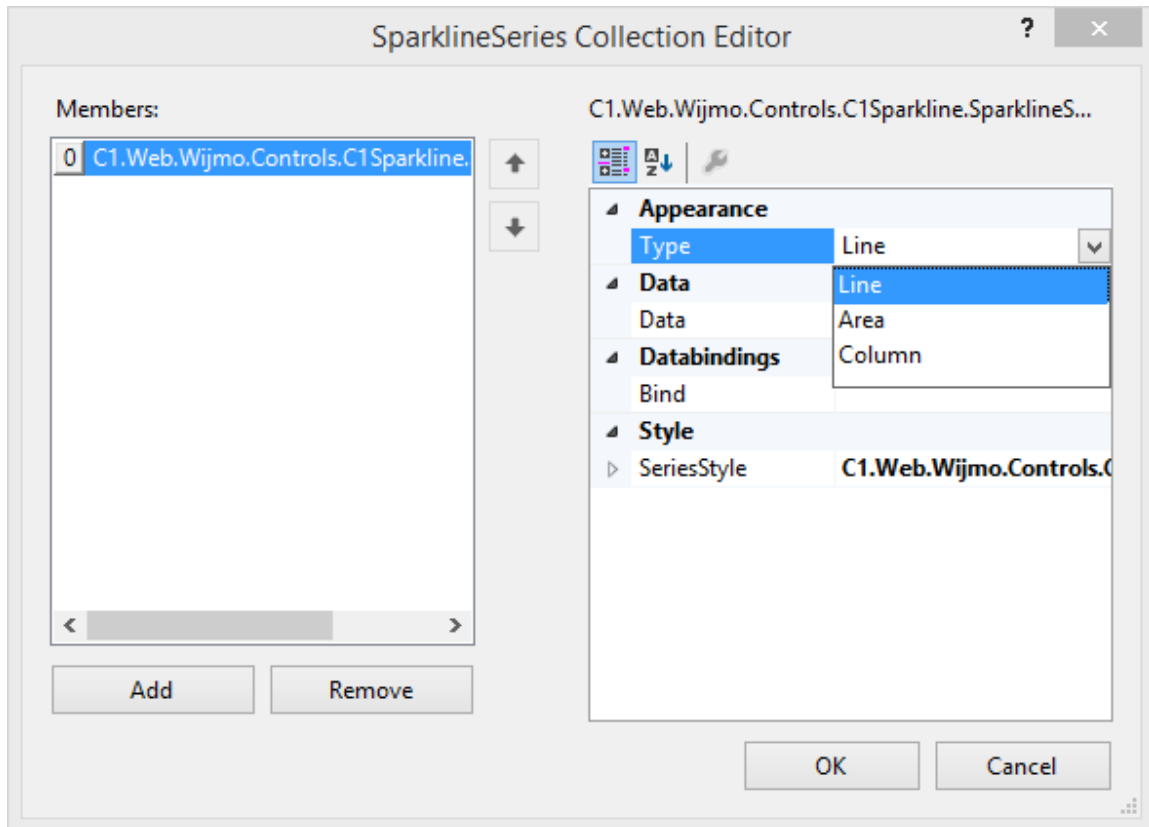
Chart Types

You can represent the data using three types of sparklines: Line, Area and Column. By default the chart type is set to Line.

- **Line:** The Line Sparkline is useful for representing a continuous flow of data, such as stock values or sales data.
- **Area:** The area sparkline draws the values as an area or region and is useful for representing cumulative data over a period of time.
- **Column:** The Column Sparkline is useful for representing data where previous values and the current value don't closely affect each other, such as sports scores and cash register receipts.

In the Designer

1. Click the smart tag in the upper-right corner of the control, and select **SeriesList**. This opens the [SparklineSeries Collection Editor](#).
2. In the SparklineSeries Collection Editor, click **Add** the add button to add a series.
3. In the window on the right, select Area, Column or Line type from the [Type](#) dropdown box. By default the type **Line** is set.



In Source View

Modify the `<cc1:C1SiteMap></cc1:C1SiteMap>` tags, to customize and add data to the control:

```
<cc1:C1Sparkline ID="C1Sparkline1" runat="server">
  <SeriesList>
    <cc1:SparklineSeries Data="33,11,15,26,16,27,37,-13,8,-8,-3,17,0,22,-13,-29,19,8"
      Type="Area" >
    </cc1:SparklineSeries>
  </SeriesList>
</cc1:C1Sparkline>
```

In Code

Add the following code to the **Page_Load** event, to customize and add data to the Sparkline control.

To write code in C#

```
C1Sparkline1.Height = 150;
C1Sparkline1.Width = 200;
double[] data = { 33, 11, 15, 26, 16, 27, 37, -13, 8, -8, -3, 17,
  0, 22, -13, -29, 19, 80 };
C1Sparkline1.SeriesList[0].Data = data;
```

To write code in VB

```
C1Sparkline1.Height = 150
C1Sparkline1.Width = 200
Dim data As Double() = {33, 11, 15, 26, 16, 27,37, -13, 8, -8, -3, 17,
  0, 22, -13, -29, 19, 80}
C1Sparkline1.SeriesList(0).Data = data
```

What You've Accomplished

When you select **Area**, **Column** or **Line** type and run the project, sparkline would appear as shown in the images below:



Area Sparkline



Column Sparkline



Line Sparkline

[Back to Top](#)

Data Binding

This topic describes local data binding of the control using the **DataSource** and **DataBind** properties. The code below shows **temperature variation** for different months of a year.

In Source View

Add the following markup within the `<cc1:C1Sparkline></cc1:C1Sparkline>`.

```
<cc1:C1Sparkline ID="C1Sparkline1" runat="server">
    <SeriesList>
        <cc1:SparklineSeries Bind="Temperature" />
    </SeriesList>
</cc1:C1Sparkline>
```

In Code

Add the following code to the **Page_Load** event, to add data to the control.

To write code in C#

```
object[] data =
{
    new { Name = "Januray", Temperature = 10 },
    new { Name = "February", Temperature = 17 },
    new { Name = "March", Temperature = 25 },
    new { Name = "April", Temperature = 37 },
    new { Name = "May", Temperature = 39 },
    new { Name = "June", Temperature = 43 },
    new { Name = "July", Temperature = 34 },
    new { Name = "August", Temperature = 36 },
    new { Name = "September", Temperature = 30 },
    new { Name = "October", Temperature = 27 },
    new { Name = "November", Temperature = 20 },
    new { Name = "December", Temperature = 15 }
}
```

```
};  
C1Sparkline1.DataSource = data;  
C1Sparkline1.DataBind();
```

To write code in VB

```
Dim data As Object() = {  
    New With {Key .Name = "Januray", Key .Temperature = 10},  
    New With {Key .Name = "February", Key .Temperature = 17},  
    New With {Key .Name = "March", Key .Temperature = 25},  
    New With {Key .Name = "April", Key .Temperature = 37},  
    New With {Key .Name = "May", Key .Temperature = 39},  
    New With {Key .Name = "June", Key .Temperature = 43},  
    New With {Key .Name = "July", Key .Temperature = 34},  
    New With {Key .Name = "August", Key .Temperature = 36},  
    New With {Key .Name = "September", Key .Temperature = 30},  
    New With {Key .Name = "October", Key .Temperature = 27},  
    New With {Key .Name = "November", Key .Temperature = 20},  
    New With {Key .Name = "December", Key .Temperature = 15}  
}  
C1Sparkline1.DataSource = data  
C1Sparkline1.DataBind()
```

What You've Accomplished

When you run the project, Sparklines appear as shown in the image below.



Add to Grid

Complete the following steps to add a [Sparkline](#) to the GridView control.

The following steps are for an application created on Visual Studio 2012. The steps may differ slightly based on the version of Visual Studio you use.

In the Designer

1. In the Toolbox, locate **C1GridView** control and drag it onto the Web Form.

In Source View

Add the following markup within the `<cc1:C1GridView></cc1:C1GridView>` tags to add sparkline in the GridView. This adds a **BoundField** and **ItemTemplate** to the GridView.

```
<cc1:C1GridView ID="C1GridView1" runat="server" AutogenerateColumns="False" AllowColSizing="True">  
    <Columns>  
        <cc1:C1BoundField DataField="Year" HeaderText="YEAR" SortExpression="Year">  
            <ItemStyle HorizontalAlign="Center" />  
        </cc1:C1BoundField>  
        <cc1:C1TemplateField HeaderText="SALES">  
            <ItemStyle HorizontalAlign="Center" />  
            <ItemTemplate>  
                <cc1:C1Sparkline ID="C1Sparkline1" runat="server">  
                    <SeriesList>  
                        <cc1:SparklineSeries>
```

```

        </cc1:SparklineSeries>
    </SeriesList>
</cc1:C1Sparkline>
</ItemTemplate>
</cc1:C1TemplateField>
</Columns>

```

In Code

Add the code below to the **Page_Load** event, to add data to the GridView control. The code below shows sales data for different years.

To write code in C#

```

var data = new[]
{
    new { Year = "2008", Sales = new List<double>{95, 87, 103, 75, 91, 66, 112,
90, 83, 65, 99, 87}},
    new { Year = "2009", Sales = new List<double>{69, 76, 82, 92, 120, 102, 110,
95, 88, 75, 96, 107}},
    new { Year = "2010", Sales = new List<double>{75, 87, 92, 74, 89, 69, 101,
92, 97, 85, 94, 112}},
    new { Year = "2011", Sales = new List<double>{88, 87, 106, 95, 91, 78, 124, 1
08, 93, 85, 103, 85}},
    new { Year = "2012", Sales = new List<double>{86, 97, 112, 75, 81, 63, 89,
94, 83, 77, 120, 103}},
    new { Year = "2013", Sales = new List<double>{105, 107, 103, 95, 111, 86,
123, 135, 101, 95, 91, 117}},
};
C1GridView1.RowDataBound += C1GridView1_RowDataBound;
C1GridView1.DataSource = data;
C1GridView1.DataBind();
}
void C1GridView1_RowDataBound(object sender,

C1.Web.Wijmo.Controls.C1GridView.C1GridViewRowEventArgs e)
{
    var C1sparkline1 = e.Row.FindControl("C1Sparkline1") as C1Sparkline;
    var prop1 = e.Row.DataItem.GetType().GetProperty("Sales");
    C1sparkline1.DataSource = prop1.GetValue(e.Row.DataItem, null);
    C1sparkline1.DataBind();
}

```

To write code in VB

```

Dim data As Object() = {
    New With {Key .Year = "2008", Key .Sales = New List(Of Double)()
From {95, 87, 103, 75, 91, 66, 112, 90, 83, 65, 99, 87}},
    New With {Key .Year = "2009", Key .Sales = New List(Of Double)()
From {69, 76, 82, 92, 120, 102, 110, 95, 88, 75, 96, 107}},
    New With {Key .Year = "2010", Key .Sales = New List(Of Double)()
From {75, 87, 92, 74, 89, 69, 101, 92, 97, 85, 94, 112}},
    New With {Key .Year = "2011", Key .Sales = New List(Of Double)()
From {88, 87, 106, 95, 91, 78, 124, 108, 93, 85, 103, 85}},
    New With {Key .Year = "2012", Key .Sales = New List(Of Double)()
From {86, 97, 112, 75, 81, 63, 89, 94, 83, 77, 120, 103}},
    New With {Key .Year = "2013", Key .Sales = New List(Of Double)()
From {105, 107, 103, 95, 111, 86, 123, 135, 101, 95, 91, 117}}
}
AddHandler C1GridView1.RowDataBound, AddressOf C1GridView1_RowDataBound

```

```

C1GridView1.DataSource = data







C1GridView1.DataBind()
End Sub

Private Sub C1GridView1_RowDataBound(sender As Object,
e As C1.Web.Wijmo.Controls.C1GridView.C1GridViewRowEventArgs)
    Dim C1sparkline1 = TryCast(e.Row.FindControl("C1Sparkline1"), C1Sparkline)
    Dim prop1 = e.Row.DataItem.[GetType]() .GetProperty("Sales")
    C1sparkline1.DataSource = prop1.GetValue(e.Row.DataItem, Nothing)
    C1sparkline1.DataBind()
End Sub

```

What You've Accomplished

When you run the project, Sparkline charts appear as shown in the image below.

YEAR	SALES
2008	
2009	
2010	
2011	
2012	
2013	

[Back to Top](#)

Display Negative Values

The following topic describes how to represent negative values below the ValueAxis and show positive and negative values in different colors in a Sparkline.

Step 1 of 3: Set ValueAxis property to True

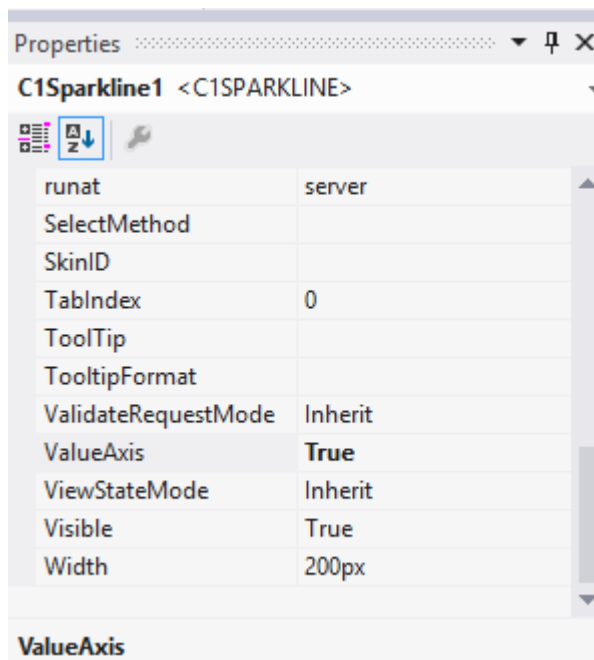
Complete the following steps to represent the negative values of the data below the ValueAxis of the Sparkline control.

1. **Right click** the [Sparkline](#) and select **Properties**. In the Properties Window, set [ValueAxis](#) to **True**. By default the [ValueAxis](#) is set **False**.

In this Topic

Step 1 of 3: Set ValueAxis property to True

Step 2 of 3: Set different colors for Positive and Negative



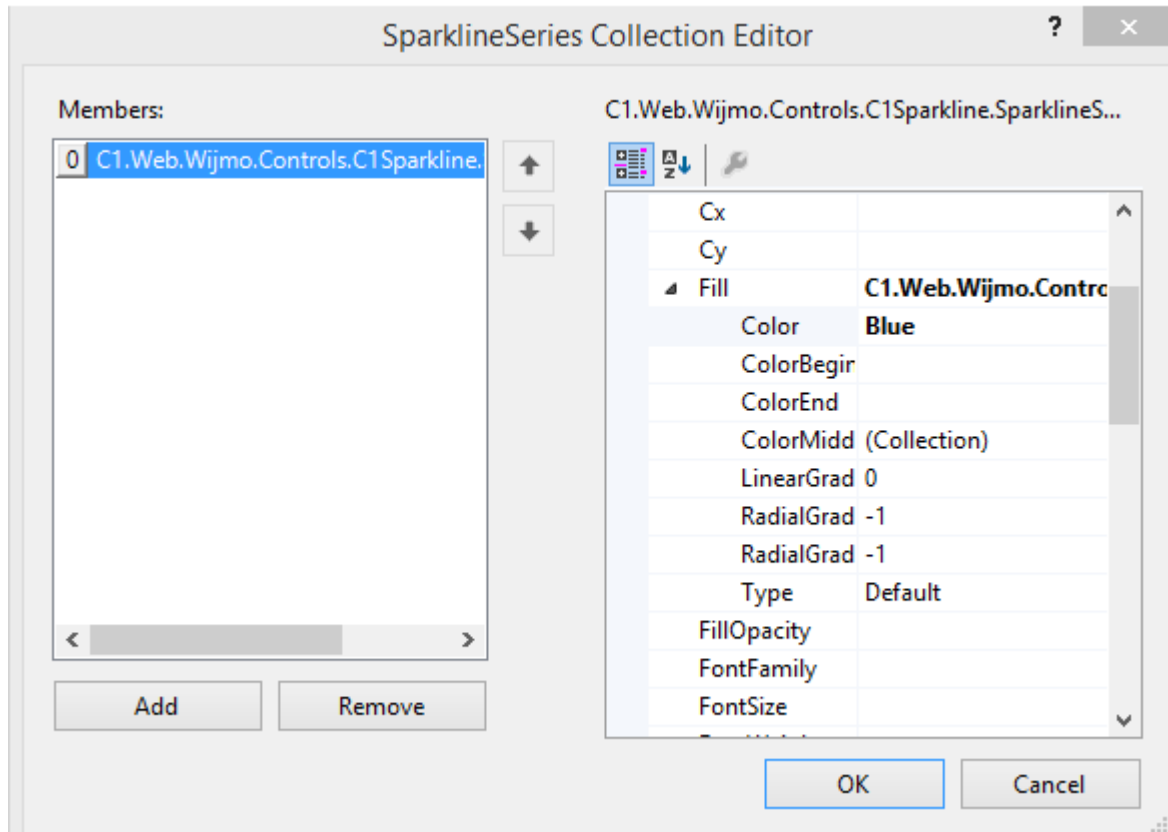
values
Step 3 of 3: Add Data
to the control

Step 2 of 3: Set different colors for Positive and Negative values

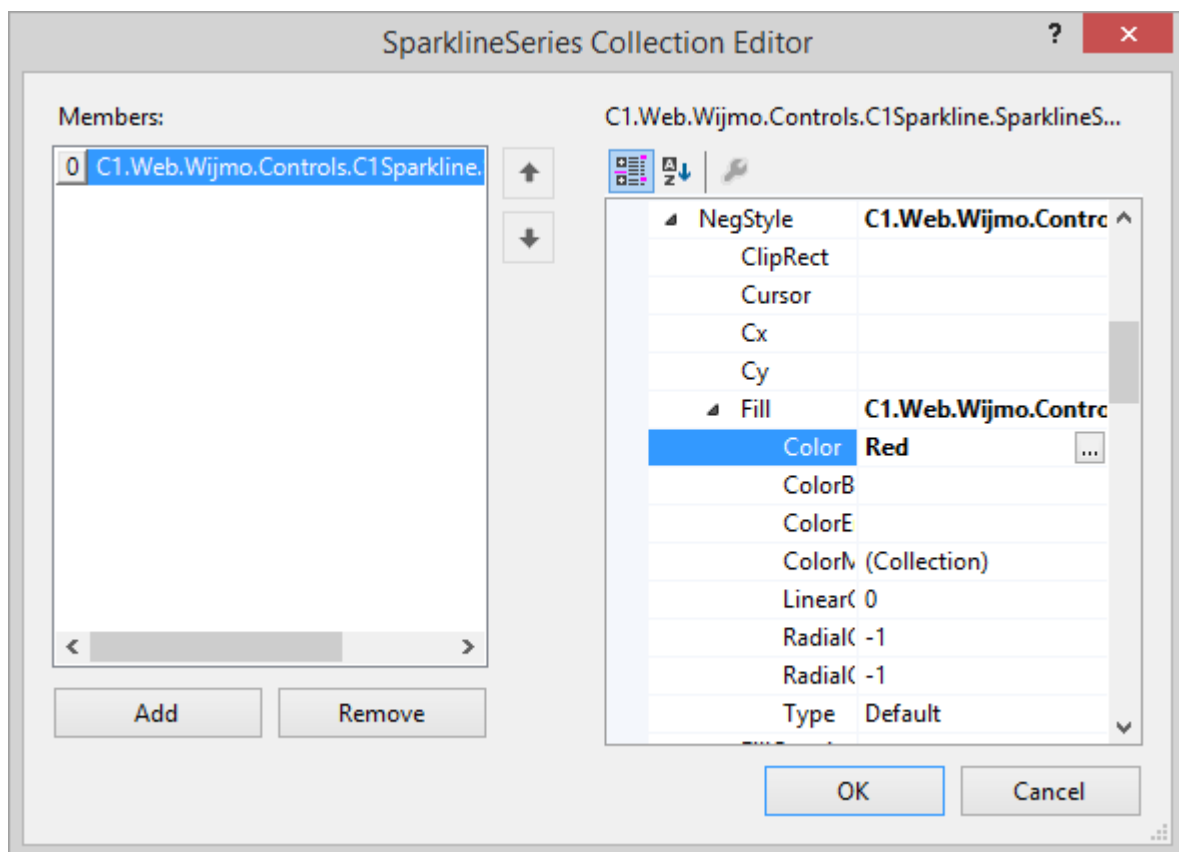
Complete the following steps to represent positive and negative values of the data in different colors on the [Sparkline](#):

In Design View

1. Select the control and open the **C1Sparkline Tasks** menu, and select **SeriesList**. This opens the [SparklineSeries Collection Editor](#).
2. Click the **Add** button, on the right window expand [SeriesStyle](#) > **Fill** > **Color**, and from the Color dropdown, select **Blue**.



3. Similarly, expand `NegStyle>Fill>Color`, and select **Red** to represent **negative** values in red.



[In Source View](#)


```
<cc1:C1Sparkline ID="C1Sparkline1" runat="server" ValueAxis="True" Height="50">
  <SeriesList>
    <cc1:SparklineSeries Type="column" SeriesStyle-Fill-Color="RoyalBlue"
SeriesStyle-NegStyle-Fill-Color="Red"> </cc1:SparklineSeries>
  </SeriesList>
</cc1:C1Sparkline>
```

Step 3 of 3: Add Data to the control

In Source View

```
<SeriesList>
  <cc1:SparklineSeries Data="20, 40, 0, 10, 20, 25, 0, 45, 15, 0, 10, 20, 5,
  20, -10, 0, -20, 20, 10">
  </cc1:SparklineSeries>
</SeriesList>
```

In Code

Add the following code to the **Page_Load** event.

To write code in C#

```
double[] data = {20, 40, 0, 10, 20, 25, 0, 45, 15, 0, 10, 20, 5, 20,
-10, 0, -20, 20, 10};
C1Sparkline1.SeriesList[0].Data = data;
```

To write code in VB

```
Dim data As Double() = {20, 40, 0, 10, 20, 25, 0, 45, 15, 0, 10, 20, 5,
20, -10, 0, -20, 20, 10}
C1Sparkline1.SeriesList(0).Data = data
```

What You've Accomplished

When you run the project, the sparkline appears like the image below:

