

Sparkline for ASP.NET Web Forms

2018.04.11 更新

グレースィティ株式会社

目次

製品の概要	2
ComponentOne for ASP.NET Web Forms のヘルプ	2
クイックスタート: データの追加と外観のカスタマイズ	3-4
設計時サポート	5
スパークラインの操作	6
アニメーションの追加	6-7
グラフタイプ	7-9
データ連結	9-10
グリッドへの追加	10-12
負の値の表示	12-15

製品の概要

Sparkline for ASP.NET Web Forms は、単純なグラフを表示してデータを視覚化します。比較的狭い領域に多数のデータポイントを表示する場合に便利です。複数のスパークライングラフをインラインに追加したり、グリッド内の1行に追加することもできます。

主要な機能

- **グラフタイプ**: 折れ線グラフ、縦棒グラフ、面グラフの3種類のグラフタイプをサポートします。
- **データ連結**: **DataSource** プロパティと **DataBind** プロパティを使用して、コードでスパークラインのデータを設定できます。
- **グリッド表示**: C1GridView コントロール内にスパークラインを表示します。
- **ValueAxis**: このプロパティを True に設定すると、**Sparkline** は、負の値を値軸の下に表示し、正の値を値軸の上に表示します。

ComponentOne for ASP.NET Web Forms のヘルプ

ComponentOne for ASP.NET Web Forms の各コントロールで共通したトピック、アセンブリの追加、テーマの適用、クライアント側情報などについては「[ASP.NET Web Forms ユーザーガイド](#)」を参照してください。

クイックスタート: データの追加と外観のカスタマイズ

このクイックスタートでは、**Sparkline** for ASP.NET Web Forms を初めて使用するための手順について説明します。このトピックでは、**Sparkline** コントロールをページに追加する方法、外観の変更方法、データポイントのリストの追加方法、**Sparkline** の実行時の動作の確認方法について説明します。

この例では Visual Studio 2012 を使用しています。他のバージョンの Visual Studio では、手順が多少異なる場合があります。

手順1/3: フォームへのコントロールの追加

単純なアプリケーションを作成し、それに Sparkline コントロールを追加するには、次の手順に従います。

1. Visual Studio で、新しい **ASP.Net Web アプリケーション** を作成し、新しい **Web フォーム** を追加します。
2. **ツールボックス** で **Sparkline** コントロールを見つけ、Web フォームにドラッグします。ツールボックスでコントロールが見つからない場合は、右クリックして **[項目の選択]** を選択し、**[ツールボックス項目の選択]** ダイアログボックスで **Sparkline** を見つけます。

トピックの内容

手順1/3: フォームへのコントロールの追加

手順2/3: コントロールのカスタマイズ

手順3/3: コントロールへのデータの追加

手順2/3: コントロールのカスタマイズ

最後の手順で作成したスパークラインの外観と動作を変更するには、次の手順に従います。

1. **Sparkline** を右クリックし、**[プロパティ]** を選択します。**プロパティウィンドウ** で、次の内容を設定します。
 - **Width** = 200px
 - **Height** = 150px
 - **Theme** = midnight
2. **[C1Sparkline のタスク]** メニューを開き、**[系列の編集]** を選択します。これにより、**SparklineSeries コレクションエディタ** が開きます。
3. 右ウィンドウで、**[タイプ]** として **[面]** を選択します。デフォルトでは、**[折れ線]** タイプが設定されます。

ソースビューの場合

コントロールをカスタマイズするには、`<cc1:C1Sparkline></cc1:C1Sparkline>` タグを変更します。

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

コードの場合

次のコードを **Page_Load** イベントに追加して、Sparkline コントロールをカスタマイズします。

C# でコードを書く場合

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

VB でコードを書く場合

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

手順3/3:コントロールへのデータの追加

コントロールにデータを追加するには、次の手順に従います。

ソースビューの場合

コントロールをカスタマイズするには、`<cc1:C1Sparkline></cc1:C1Sparkline>` タグを変更します。

```
<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>
```

コードの場合

次のコードを **Page_Load** イベントに追加して、データを追加します。

C# でコードを書く場合

```
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;
```

Visual Basic でコードを書く場合

```
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
```

ここまでの成果

プロジェクトを実行すると、結果が次の図のように表示されます。

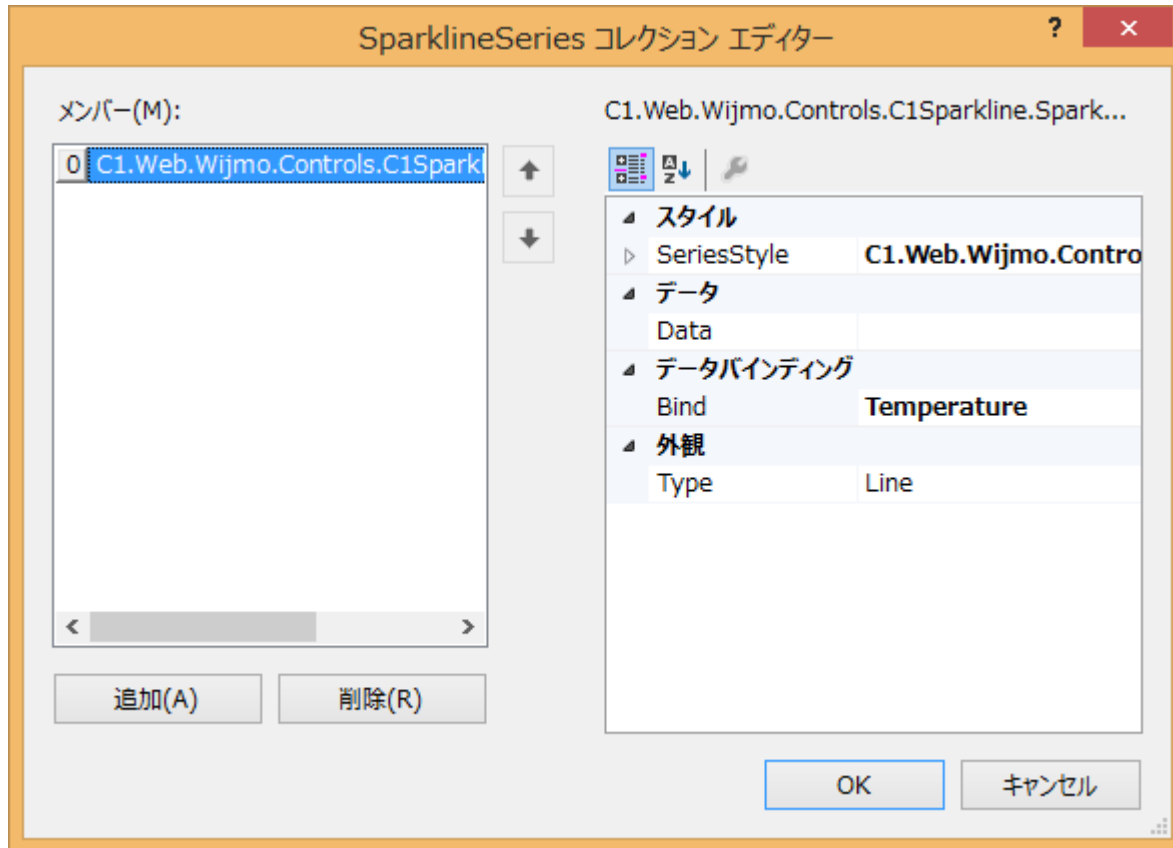


設計時サポート

このトピックでは、Sparkline の設計時機能を使用してコントロールを構成する方法について説明します。

[**C1Sparkline のタスク**]メニューにアクセスするには、コントロールの右上隅にあるスマートタグをクリックします。[**系列の編集**]にアクセスします。これにより、**SparklineSeries コレクションエディタ**が開きます。

SparklineSeries コレクションエディタ



[**追加**]ボタンは、新しいスパークライン **SeriesList** を追加します。

[**削除**]ボタンは、選択したスパークライン **SeriesList** を削除します。

スパークラインの操作

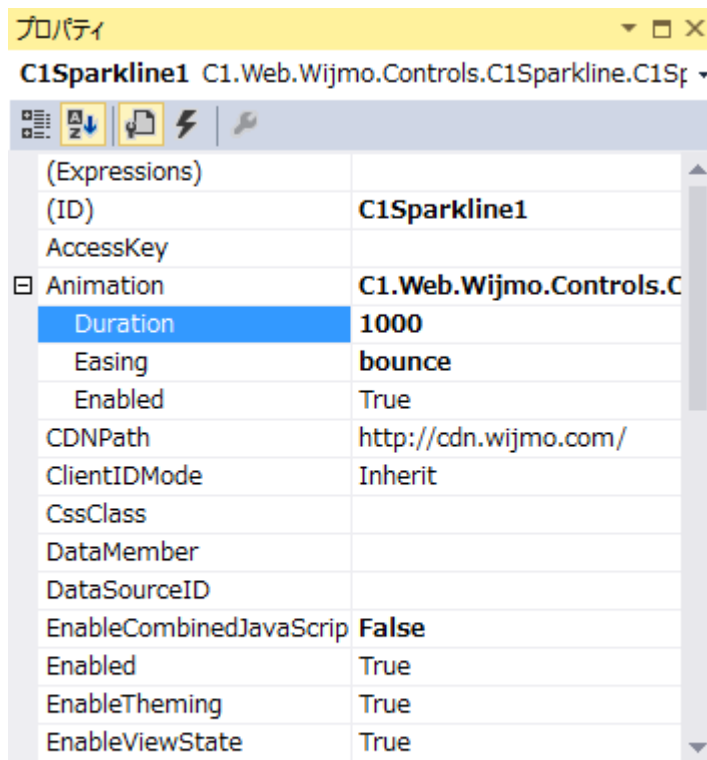
アニメーションの追加

Sparkline コントロールにアニメーションを追加するには、**Animation** プロパティからアニメーションの期間やイーasingタイプを設定します。

- **有効**: このプロパティを **False** に設定して、アニメーションなしでスパークラインを表示できます。デフォルトでは、このプロパティは **True** に設定されています。
- **時間**: スパークラインの表示にかかる時間を設定します。デフォルト値は **2000** ミリ秒です。
- **イーasing**: アニメーションのイーasingタイプを変更します。デフォルトでは、イーasingは **[リニア]** に設定されています。

デザイナの場合

1. Sparkline コントロールを右クリックし、**[プロパティ]**を選択します。これで、**[プロパティ]**ウィンドウが表示されます。
2. **プロパティウィンドウ**で、**Animation** プロパティを展開し、**Duration** を 1000 に変更し、**Easing** を bounce に変更します。



Duration

アニメーションの長さを示す値。

- 3.

ソースビューの場合

コントロールのアニメーションをカスタマイズするには、`<cc1:C1Sparkline>` タグ内の **Animation** プロパティを変更します。

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

Sparkline for ASP.NET Web Forms

コードの場合

次のコードを Page_Load イベントに追加して、コントロールのアニメーションをカスタマイズします。

C# でコードを書く場合

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

Visual Basic でコードを書く場合

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

 アニメーションを無効にするには、**Animation.Enabled** を False に設定します。

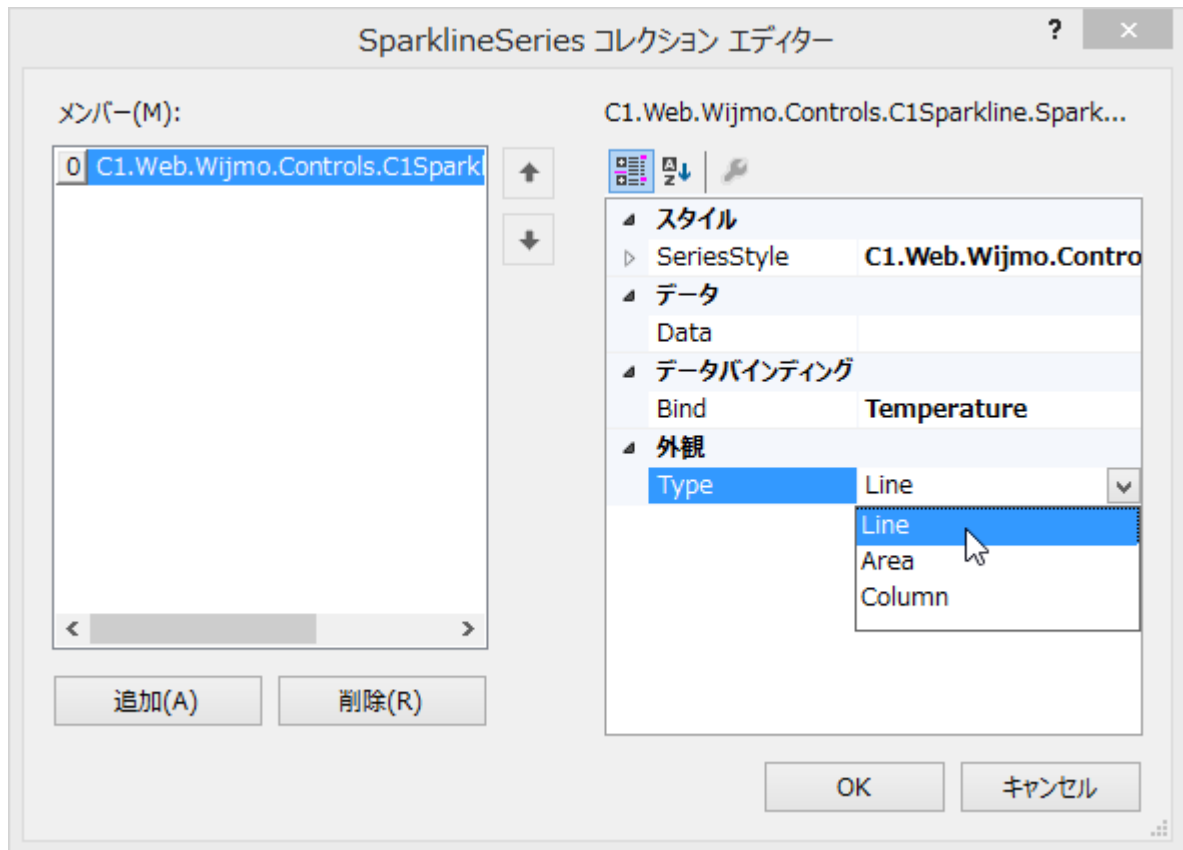
グラフタイプ

折れ線、面、縦棒の3つのタイプのスパークラインを使用してデータを表示できます。デフォルトでは、グラフタイプは[折れ線]に設定されています。

- **折れ線**: 折れ線スパークラインは、株価や売上データなど、連続的なデータを表す場合に便利です。
- **面**: 面スパークラインは、値を面または領域として描画し、一定期間の累積データを表す場合に便利です。
- **縦棒**: 縦棒スパークラインは、スポーツのスコアやレジのレシートなど、以前の値と現在の値があまり関係しないデータを表す場合に便利です。

デザイナーの場合

1. コントロールの右上隅にあるスマートタグをクリックし、**[系列の編集]**を選択します。これにより、**SparklineSeries コレクションエディタ**が開きます。
2. **SparklineSeries コレクションエディタ**で、**[追加]**ボタンをクリックして系列を追加します。
3. 右ウィンドウで、**[タイプ]**ドロップダウンボックスから[面]、[縦棒]、または[折れ線]タイプを選択します。デフォルトでは、**[折れ線]**タイプが設定されます。



ソースビューの場合

コントロールにデータを追加してカスタマイズするには、`<cc1:C1Sparkline></cc1:C1Sparkline>`タグを変更します。

```
<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>
```

コードの場合

次のコードを **Page_Load** イベントに追加して、Sparkline コントロールをカスタマイズし、データを追加します。

C# でコードを書く場合

```
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;
```

VB でコードを書く場合

```
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
```

ここまでの成果

Sparkline for ASP.NET Web Forms

[面]、[縦棒]、または[折れ線]タイプを選択し、プロジェクトを実行すると、スパークラインが次の図のように表示されます。



面スパークライン



縦棒スパークライン



折れ線スパークライン

[先頭に戻る](#)

データ連結

このトピックでは、**DataSource** プロパティと **DataBind** プロパティを使用したコントロールのローカルデータ連結について説明します。次のコードは、ある年の月ごとの**温度変化**を示します。

ソースビューの場合

<cc1: C1Sparkline> </cc1: C1Sparkline> 内に次のマークアップを追加します。

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

コードの場合

次のコードを **Page_Load** イベントに追加して、データをコントロールに追加します。

C# でコードを書く場合

```
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();
```

VB でコードを書く場合

```
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()
```

ここまでの成果

プロジェクトを実行すると、スパークラインが次の図のように表示されます。



グリッドへの追加

スパークラインを GridView コントロールに追加するには、次の手順に従います。

以下の手順は、Visual Studio 2012 で作成されたアプリケーション用です。使用する Visual Studio のバージョンによっては、手順が多少異なる場合があります。

デザイナーの場合

1. ツールボックスで、**C1GridView** コントロールを見つけ、それを Web フォームにドラッグします。

ソースビューの場合

GridView にスパークラインを追加するには、`<cc1: C1GridView></cc1:C1GridView>` タグ内に次のマークアップを追加します。これにより、**BoundField** と **ItemTemplate** が GridView に追加されます。

```
<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>
```

コードの場合

GridView コントロールにデータを追加するには、**Page_Load** イベントに次のコードを追加します。次のコードは、年ごとの販売データを示します。

C# でコードを書く場合

```
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, 108, 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();
}
```

VB でコードを書く場合

```
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

```

ここまでの成果

プロジェクトを実行すると、スパークライングラフが次の図のように表示されます。



負の値の表示

次のトピックでは、負の値を値軸の下に表示する方法、およびスパークラインの正の値と負の値を別の色で表示する方法について説明します。

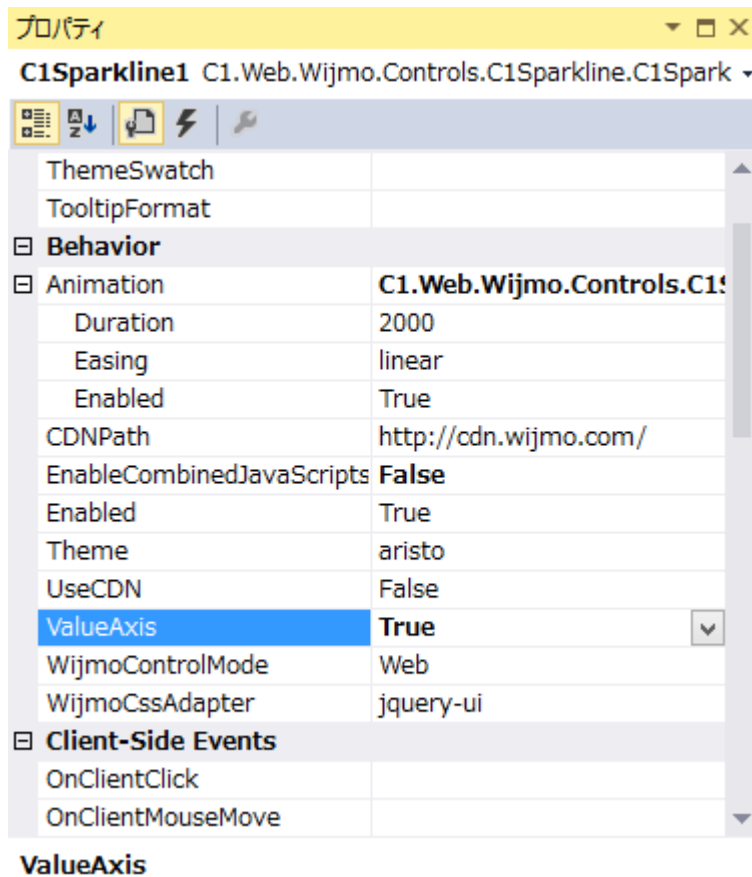
手順1/3: ValueAxis プロパティを True に設定

データの負の値をスパークラインコントロールの **ValueAxis** の下に表示するには、次の手順に従います。

1. **Sparkline** を右クリックし、**[プロパティ]**を選択します。プロパティウィンドウで、**ValueAxis** を **True** に設定します。デフォルトでは、**ValueAxis** は **False** に設定されています。

トピックの内容

手順1/3: ValueAxis プロパティを True に設定
 手順2/3: 正の値と負の値に別の色を設定
 手順3/3: コントロールへのデータの追加

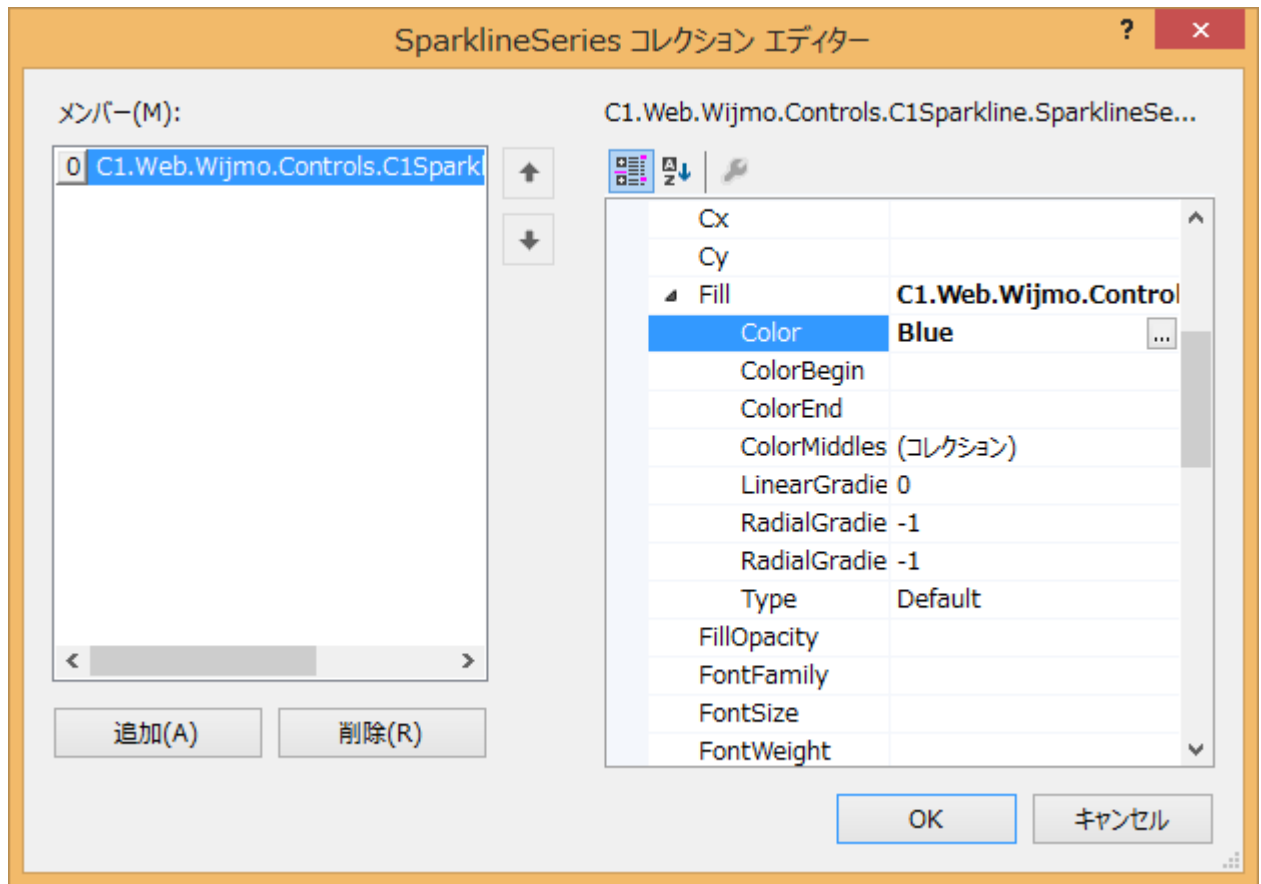


手順2/3: 正の値と負の値に別の色を設定

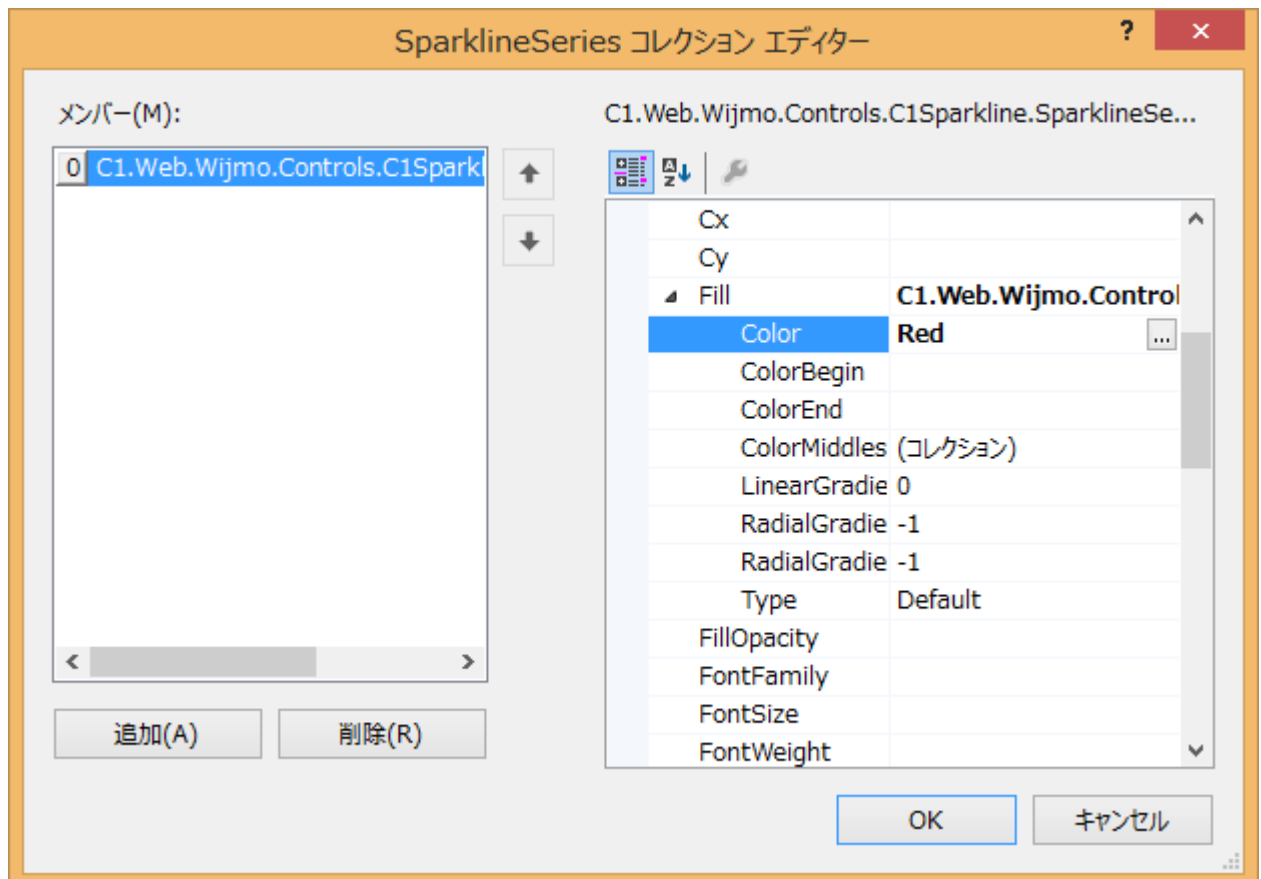
スパークラインに正の値と負の値を別の色で表示するには、次の手順に従います。

デザインビューの場合

1. コントロールを選択し、**[C1Sparkline のタスク]**メニューを開き、**[系列の編集]**を選択します。これにより、**SparklineSeries コレクションエディタ**が開きます。
2. **[追加]**ボタンをクリックし、右ウィンドウで**[系列スタイル]**>**[塗りつぶし]**>**[色]**の順に展開し、**[色]**ドロップダウンから**[青]**を選択します。



3. 同様に、[負のスタイル]>[塗りつぶし]>[色]の順に展開し、[赤]を選択して負の値を赤で表示します。



Sparkline for ASP.NET Web Forms

ソースビューの場合

```
<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>
```

手順3/3:コントロールへのデータの追加

ソースビューの場合

```
<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>
```

コードの場合

Page_Load イベントに次のコードを追加します。

C# でコードを書く場合

```
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;
```

VB でコードを書く場合

```
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
```

ここまでの成果

プロジェクトを実行すると、スパークラインが次の図のように表示されます。

