

---

ComponentOne

# SplitContainer for WinForms

**ComponentOne, a division of GrapeCity**

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

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

**Sales:** [sales@componentone.com](mailto: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 \$25 (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

SplitContainer for WinForms Overview	3
Help with WinForms Edition	3
embeddingcomponentinproject	3
runtimefilescomponent	3
Key Features	4
SplitContainer for WinForms Quick Start	5
Step 1 of 4: Adding C1SplitContainer to the Page	5
Step 2 of 4: Adding C1SplitterPanels to the C1SplitContainer	5-7
Step 3 of 4: Creating Collapsible Panels	7-8
Step 4 of 4: Sizing the C1SplitContainer Control	8-9
Design-Time Support	10
C1SplitContainer Smart Tags	10
C1SplitContainer Tasks Menu	10-11
C1Splitter Panel Tasks Menu	11-13
SplitContainer Elements	14
Splitter Panels	14-15
Panel Header	15
Splitter Bar	15-16
Expander Button	16
SplitContainer Appearance	17
SplitContainer Visual Styles	17-19
SplitContainer Appearance Properties	19-20
SplitContainer Layout	21
SplitContainer Layout Options	21-22
Horizontal Split	22-23
Vertical Split	23
Compound Split	23-24
Full-Size Split	24
SplitterPanel Behavior	25
Collapsible and Expandable Panels	25
SplitterPanel Scrolling	25-26
SplitterPanel and Splitter Bar Sizing	26
SplitterPanel Header and Splitter Bar Drawing Events	26
SplitContainer for WinForms Samples	27

SplitContainer for WinForms Task-Based Help	28
Adding Content to the Splitter Panels	28
Adding Arbitrary Controls to the C1SplitterPanel	28-30
Adding Text to a Splitter Panel	30-32
Adding Multiple Panels to the C1SplitContainer	32-34
Changing the C1SplitContainer's Appearance	34
Changing the Background Color of Panels	34-36
Changing the Border Color	36-38
Changing the Visual Style	38-40
Changing the Splitter Width	40-42
Creating Different Split Types	42
Creating a Horizontal Split	42-44
Creating a Vertical Split	44-46
Creating a Nested Split	46-47
Creating a Full-Size Split	47-49
Setting C1SplitContainer Behaviors	49
Adding a ToolTip to the Collapsed Splitter Panel	49-50
Setting a Minimum Size for a Splitter Panel	50-51
Setting a Collapsible Splitter Panel	51-53

## SplitContainer for WinForms Overview

Create highly sophisticated resizable panels in your applications without using any resizing code with **SplitContainer for WinForms** (C1SplitContainer). C1SplitContainer's WYSIWYG designer, support for multiple panels inside a container, and wide range of appearance properties gives you a customizable and flexible split container control that is very easy to use.

C1SplitContainer is a container control for resizable docked panels. The C1SplitContainer control allows you to dock multiple panels to either side of the container control. **SplitContainer for WinForms** includes many features such as flexible layout (nested panels, resizable panels, and collapsing handles), and user interface features (including Office 2007 and Office 2010 Visual Styles), that you can use to create complex user interfaces.

### Getting Started

- [SplitContainer for WinForms Quick Start](#)
- [SplitContainer Layout](#)
- [SplitContainer for WinForms Task-Based Help](#)

## Help with WinForms Edition

### Getting Started

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

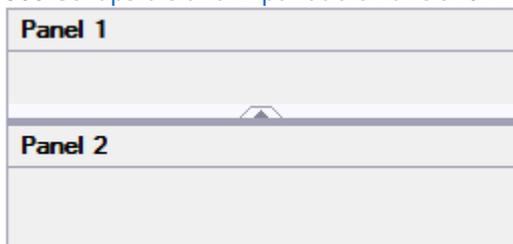
**embeddingcomponentinproject**

**runtimefilescomponent**

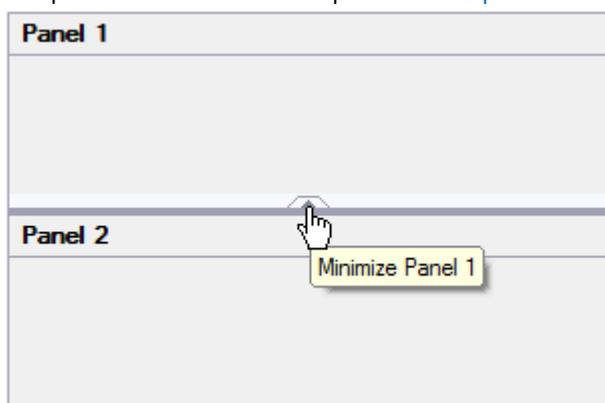
## Key Features

The following are some of the main features of [C1SplitContainer](#) that you may find useful:

- **Expand and Collapse Panels** Just set one property, [Collapsible](#), to enable expandable or collapsible panels. See [Collapsible and Expandable Panels](#) for more information.



- **Flexible Layout Options** With just one property, [Dock](#), you can create a complex interface by docking panels or nested panels to the top, bottom, left, or right of the container control. For more information see [SplitContainer Layout](#).
- **Multiple Panels** You can easily add, remove, and modify multiple panels through the **C1SplitContainer.Panels Collection editor**. For more information see [Adding Multiple Panels to the C1SplitContainer](#).
- **Unlimited Nesting** Organize massive amounts of data into one page using SplitContainer's nesting feature. **SplitContainer for WinForms** enables you to nest multiple splits of any orientation type as well accommodate the resizing for the nested splitters when you resize the panel. See [Compound Split](#) for more information.
- **Collapsed or Expanded Splitter Bar ToolTip** Create a more user-friendly Windows application by adding a ToolTip to the collapsed or expanded splitter bar. The ToolTip can let end-users know to click on the expander button to expand the hidden panel or click on the collapsed button to hide the panel. See [Expander Button](#) for more information.



## SplitContainer for WinForms Quick Start

The goal of this quick start guide is to get you acquainted with **SplitContainer for WinForms**. In the first step of this Quick Start guide, you will add a **C1SplitContainer** control to your WinForms project. This quick start guide will also explain how to set the layout, behavior, and appearance properties of C1SplitContainer.

### Step 1 of 4: Adding C1SplitContainer to the Page

In this step, you will create a .NET project and add a **C1SplitContainer** control to it.

Complete the following steps:

1. Begin by creating a new Windows Forms Application
2. While in **Design** view, navigate to the Visual Studio Toolbox and double-click the **C1SplitContainer** control to add it to your form. The **C1SplitContainer** appears as a container with one **C1SplitterPanel**.



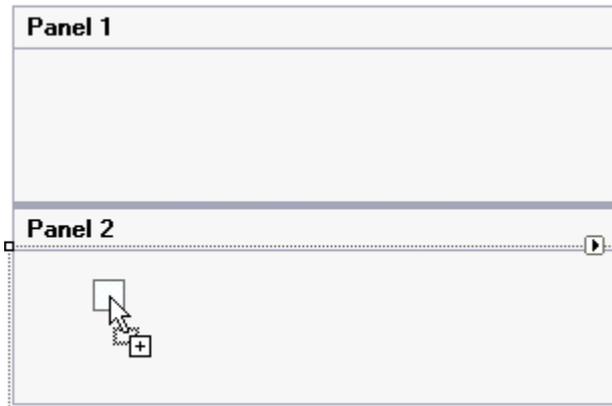
In the next step, you will add several C1SplitterPanels to the C1SplitContainer control.

### Step 2 of 4: Adding C1SplitterPanels to the C1SplitContainer

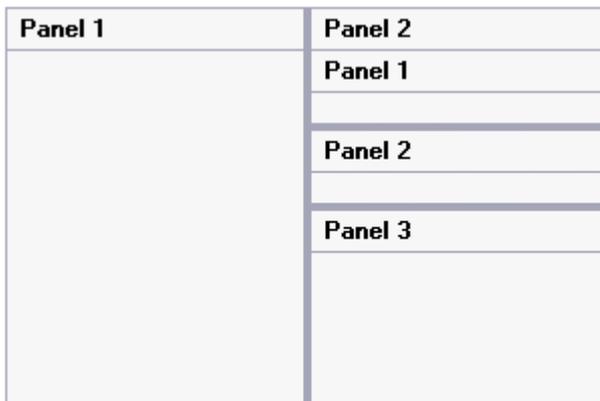
In this step you learn how to add the C1SplitterPanels to the C1SplitContainer through the **C1SplitContainer.Panels Collection** editor as well as set the panel's **Dock** property to control the layout of the panels.

Complete the following steps:

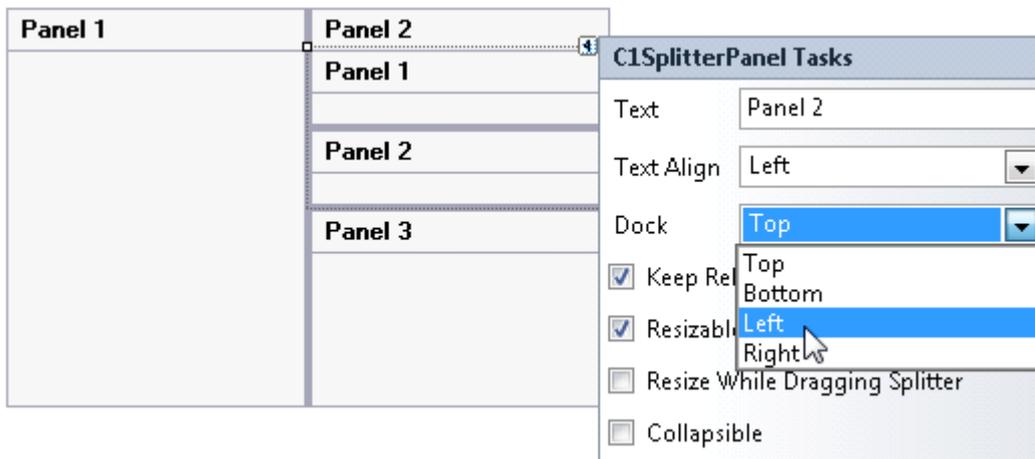
1. Click on the C1SplitContainer's smart tag to open its tasks menu.
2. Select **Add Panel** from the **C1SplitContainer Tasks Menu**. One C1SplitterPanel is added below the existing one.
3. Drag and drop another **C1SplitContainer** into Panel2 of the C1SplitContainer.



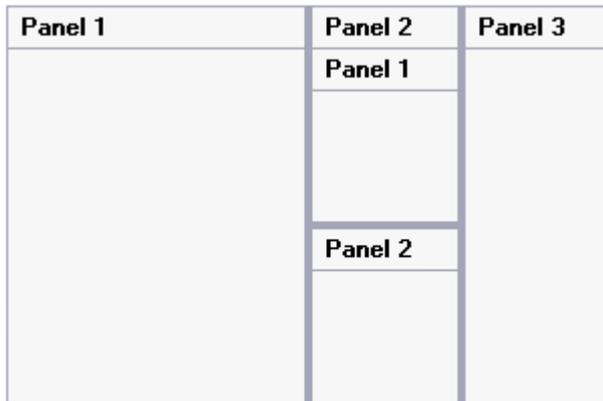
4. Select C1SplitContainer2 and click **Add Panel** from its tasks menu to add a new C1SplitterPanel.
5. Select C1SplitContainer1 and click **Add Panel** from its tasks menu to add a new C1SplitterPanel.
6. Select C1SplitContainer1 Panel1 dock property to left.



7. Select Panel 2 of C1SplitContainer1 and set its Dock property to left.



Panel2 which contains Panel1 and Panel2 of C1SplitContainer2 will be docked to the left of its parent control, Panel 3.

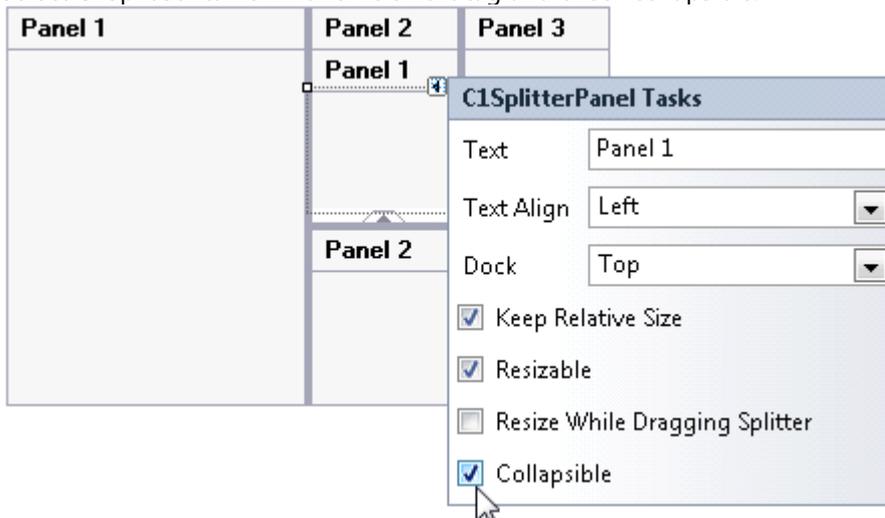


## Step 3 of 4: Creating Collapsible Panels

In this step you will modify a few common elements of the C1SplitContainer control to change its appearance as well as create collapsible panels.

Complete the following steps:

1. Select C1SplitContainer2 Panel1's smart tag and check collapsible.



The expander button appears and now you will be able to expand or collapse the panels of Panel 2 at run time.

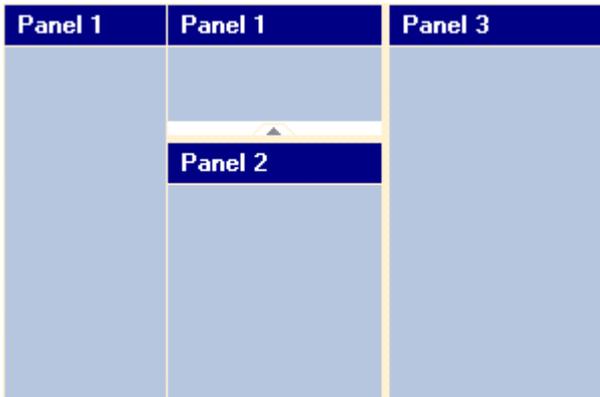
2. Select **C1SplitContainer1** to reveal its list of properties and then complete the following:
  - o Set the **C1SplitContainer1.BackColor** property to **LightSteelBlue**.
  - o Set the **C1SplitContainer1.FixedLineColor** property to **BlanchedAlmond**.
  - o Set the **C1SplitContainer1.HeaderBackColor** property to **Navy**.
  - o Set the **C1SplitContainer1.HeaderForeColor** property to **White**.
  - o Set the **C1SplitContainer1.SplitterColor** property to **BlanchedAlmond**.

For more information see [SplitContainer Appearance Properties](#).

3. Set the **C1SplitContainer1.Dock** property to **Fill**.
4. Select C1SplitContainer1 and click Edit Panels from its tasks menu.
5. In the Members list select the c1SplitterPanel2 [Panel 2] and delete the default, "Panel 2" text from the **Text** property.

This removes the panel header from the C1SplitContainer.

The appearance settings will be updated at design time, and your control will resemble the following image:



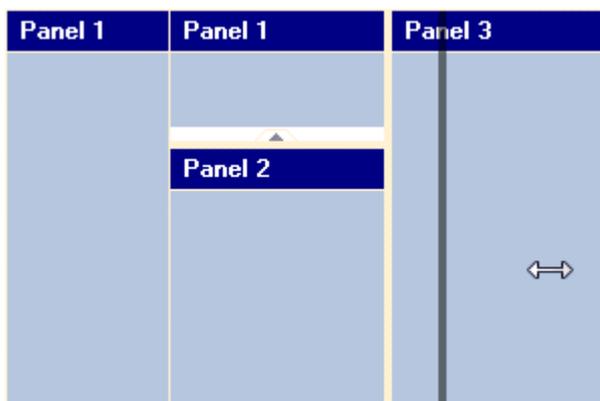
In the next step, you'll control the sizing for the C1SplitterPanels.

## Step 4 of 4: Sizing the C1SplitContainer Control

In this step, you will create a fixed size and absolute size panel and run the project to see the effect of the panels sizing.

Complete the following steps:

1. Select `c1SplitterPanel1` of `C1SplitContainer1` and set its following properties:
  - Set the `c1SplitterPanel1.Width` property to **80**.
  - Set the `c1SplitterPanel1.KeepRelativeSize` property to **False**.
  - Set the `c1SplitterPanel1.Resizable` property to **False**.
  - Set the `c1SplitterPanel1.Text` property to **Fixed**.
2. Select `c1SplitterPanel4` of the `C1SplitContainer1` and set its the `c1SplitterPanel4.MinWidth` property to **80**.
3. Press **F5** to build the project. Observe the following behaviors:
  - Hover over the expander button and notice that a tooltip appears with the default text, "minimize panel 1".
  - Click the expander button and notice panel1 collapses.
  - Notice how you can't resize the Fixed panel since you set the `Resizable` property to **False**.
  - Click the splitter bar to activate it and then attempt to drag the splitter bar to the bottom of the control.



As you've probably noticed by now – and as you can see in the image above – you can't expand the bar all the way over to the right of the control. This is because you set the `MinWidth` to **80** so that users wouldn't be able to minimize **Panel 3** to a width of less than 80 pixels. Thus, upon releasing your cursor, Panel 3 will appear no smaller than 80 pixels.

## What You've Accomplished

Congratulations! You have successfully completed the [C1SplitContainer](#) quick start. In this topic, you added a C1SplitContainer control to your window page, added panels, customized its behavior and appearance, and manipulated the control at run time.

## Design-Time Support

**C1SplitContainer** provides customized context menus, smart tags, and a designer that offers rich design-time support and simplifies working with the object model.

The following topics describe how to use **C1SplitContainer** design-time environment to configure **C1SplitContainer**.

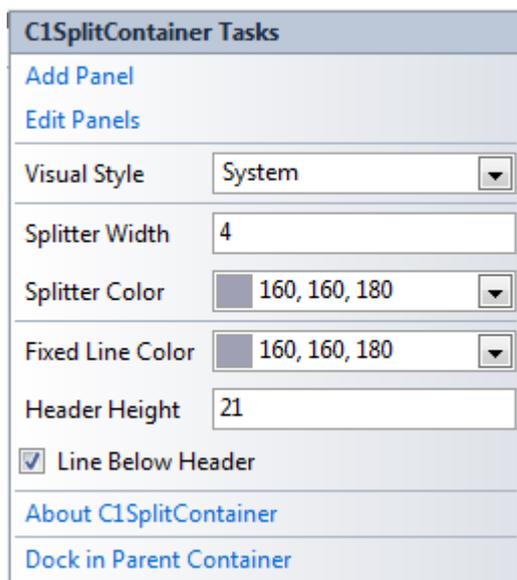
## C1SplitContainer Smart Tags

In Visual Studio, each component in **SplitContainer for WinForms** includes a smart tag. A smart tag represents a short-cut tasks menu that provides the most commonly used properties in each control.

The following section introduces each smart tag for C1SplitContainer's components.

## C1SplitContainer Tasks Menu

To access the **C1SplitContainer Tasks** menu, click the smart tag (🔗) in the upper right corner of the **C1SplitContainer** control. This will open the **C1SplitContainer Tasks** menu.



The C1SplitContainer Tasks menu operates as follows:

### Add Panel

Clicking the Add Panel item adds a new C1SplitterPanel beneath the existing one.

### Edit Panels

Clicking the Edit Panels item opens the C1SplitContainer.Panels Collection Editor where you can add, remove, or modify C1SplitterPanels.

### Visual Style

The Visual Style dropdown box provides a list of styles to choose from: Custom, System, Office2007Black, Office2007Blue, Office2007Silver, Office2010Black, Office2010Blue, Office2010Silver, and Windows7.

## Splitter Width

Specifies the width of the splitter element. The default splitter width is 4 pixels.

## Splitter Color

Click the dropdown arrow and select a color to specify the background color for the splitter element.

## Fixed Line Color

Click the dropdown arrow and select a color to specify the fixed line background color for the C1SplitContainer.

## Header Height

Specifies the height of each C1SplitterPanel header. The default height is 21 pixels.

## Line Below Header

Unchecking the checkbox removes the line below the header item.

## About C1SplitContainer

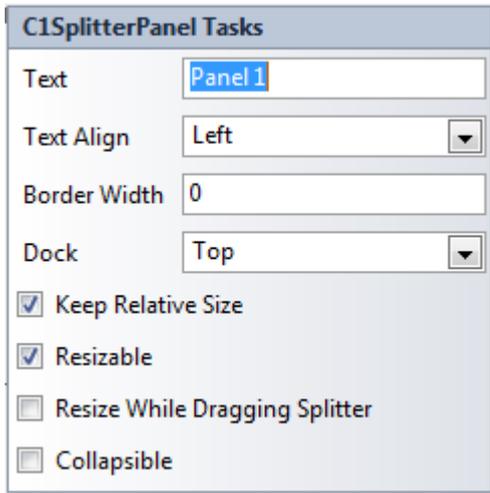
Clicking **About C1SplitContainer** reveals the **About** dialog box. This dialog box displays the version number and licensing information for the SplitContainer product.

## Dock in Parent Container

Clicking **Dock in Parent Container** docks the C1SplitContainer inside the parent container.

## C1Splitter Panel Tasks Menu

To access the **C1SplitterPanel Tasks** menu, click the smart tag (🔗) in the upper right corner of the **C1SplitterPanel** component. This will open the **C1SplittePanel Tasks** menu.



The C1SplitterPanel Tasks menu operates as follows:

## Text

Specifies the name that appears in the Header element of the C1SplitterPanel.

## Text Align

Click on the dropdown arrow to specify the alignment of the header text: Left, Center, and Right.

## Border Width

Specifies the width of the C1SplitterPanel border, in pixels. The default Border width is 1 pixel.

## Dock

Click on the dropdown arrow to specify the dock position of the C1SplitterPanel: Top, Bottom, Left, or Right.

## Keep Relative Size

Unchecking this checkbox sets the KeepRelativeSize property to False.

## Resizable

Unchecking this checkbox sets the Resizable property to False. If set to False, the panel will not be resized with a splitter.

## Resize While Dragging Splitter

Indicates whether or not to resize the panel while the user is dragging the splitter. By default the checkbox is unchecked so the user can't resize the panel while dragging the splitter.

## Collapsible

Indicates whether or not the C1SplitterPanel can be collapsed. Check this checkbox to make the C1SplitterPanel collapsible. By default the panel is not collapsible.

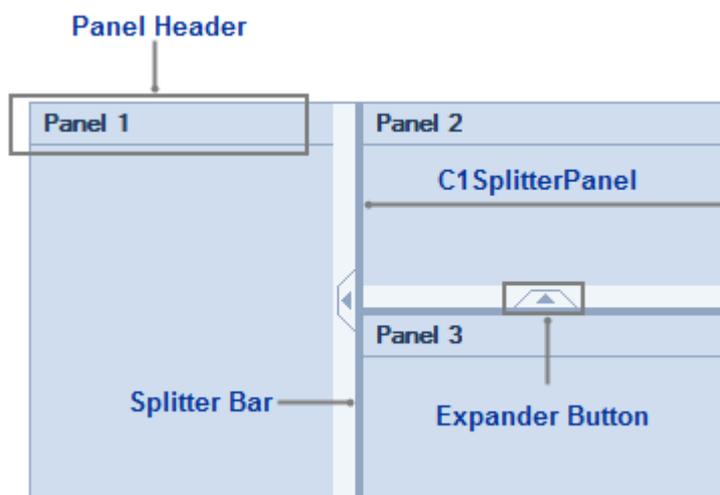
## SplitContainer Elements

The topics in this section are categorized into distinct elements that represent different aspects of the C1SplitContainer control. Each individual topic provides the following information:

- The element's purpose and its position within a control.
- The element's essential parts.
- The element's properties that affect the appearance and functionality of the element.

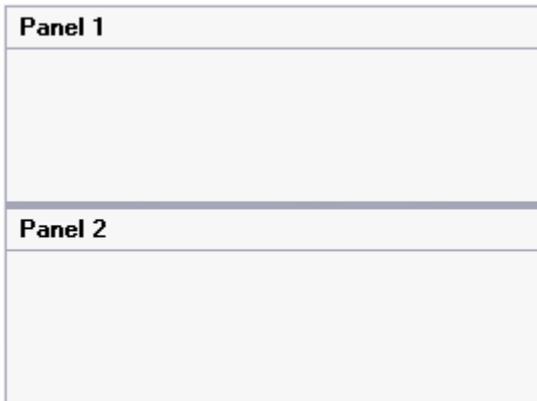
Four elements exist within the C1SplitContainer control: a C1SplitterPanel, Splitter Bar, SplitterPanel Header, and an expander button. The **C1SplitContainer** represents a container to hold the panels and its elements. The **Splitter Bar** represents the divider that separates the two panels. The **C1SplitterPanel** represents the panel inside the C1SplitContainer. The **Panel Header** represents the header that appears as a rectangular title bar above each panel. By default, a fixed line appears below the panel header and its color and width can be controlled by the [FixedLineColor](#) and [FixedLineWidth](#) properties. The expander button appears when the [Collapsible](#) property is true.

The following image labels the elements of the C1SplitContainer control:



## Splitter Panels

The default C1SplitContainer has 1 panel. One or more panels can be added to the C1SplitContainer at design time through the C1SplitContainer's Tasks menu or properties window or programmatically through the **Add** method. When a second panel is added it appears below the previous panel and a horizontal splitter bar separates the two panels.



Arbitrary controls can be added within each panel. In the **C1SplitContainer.Panels Collection** editor, you can add, remove, and edit each panel's appearance and behavior through the **C1SplitterPanel** properties included in it.

In the object model for **C1SplitContainer**, all panels are referred as the **C1SplitterPanel**. The **C1SplitterPanel** object contains properties and methods for all panels. You can apply different behaviors and styles to each panel since you can set each panel individually.

## Panel Header

The **Panel Header** represents the header that appears as a rectangular title bar above each panel. By default, a fixed line appears below the panel header and its color and width can be controlled by the **FixedLineColor** and **FixedLineWidth** properties.

A panel header can be removed by deleting the text from the **Text** property.

The following table lists and describes the properties that effect the panel header:

Property	Description
<a href="#">HeaderBackColor</a>	Gets the background color of the header in the <b>C1SplitterPanel</b> .
<a href="#">HeaderForeColor</a>	Gets the foreground color of the header in the <b>C1SplitterPanel</b> .
<a href="#">HeaderTextAlign</a>	Gets the text alignment for the header in the <b>C1SplitterPanel</b> .
<a href="#">Text</a>	Gets the text for the header in the <b>C1SplitterPanel</b> .
<a href="#">FixedLineColor</a>	Gets or sets the color of a fixed dividing line.
<a href="#">FixedLineWidth</a>	Gets or sets the width of a fixed dividing line.
<a href="#">LineBelowHeader</a>	Gets or sets whether a horizontal line should appear below panel headers.
<a href="#">HeaderHeight</a>	Gets or sets the height of the <b>C1SplitterPanel</b> header.
<a href="#">HeaderLineColor</a>	Gets or sets the color of the horizontal line that appears below the panel header.
<a href="#">HeaderLineWidth</a>	Specifies the width of a horizontal line that appears below panel headers. If it's equal to 0 the header line width depends on the <b>FixedLineWidth</b> property.
<a href="#">HeaderTextOffset</a>	Gets or sets the offset value (in pixels) of the panel caption text.

For more information see [SplitContainer Appearance Properties](#).

## Splitter Bar

The **Splitter Bar** represents the divider that separates the two panels. The splitter bar appears when two or more panels are added to the C1SplitContainer control.

The following table lists and describes the properties that effect the splitter bar:

Property	Description
<a href="#">SplitterColor</a>	Gets or sets the color of a splitter element.
<a href="#">SplitterWidth</a>	Gets or sets the width of a splitter element.
<a href="#">SplitterMovingColor</a>	Gets or sets the opacity level of a splitter being moved by the user.
<a href="#">SplitterMovingOpacity</a>	Gets or sets the opacity level of a splitter being moved by the user.

For more information see [SplitContainer Appearance Properties](#).

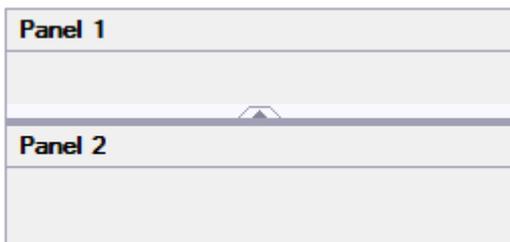
## Expander Button

The expander button appears on the splitter bar of the C1SplitContainer when the [Collapsible](#) property is True. When the panel is expanded the arrow points up and when the panel is collapsed the arrow points down. You can make the expander button appear larger by setting the [EnlargeCollapsingHandle](#) property to True.

The expander button image and the splitter bar will appear different depending on the visual style applied to the control.

Appearance of the Splitter Bar Styles

The following image illustrates the Collapsible property set to True:



The following image illustrates the Collapsible property set to True and the EnlargeCollapsingHandle property set to True:



## SplitContainer Appearance

The following topics describe C1SplitContainer's built-in visual styles and appearance properties.

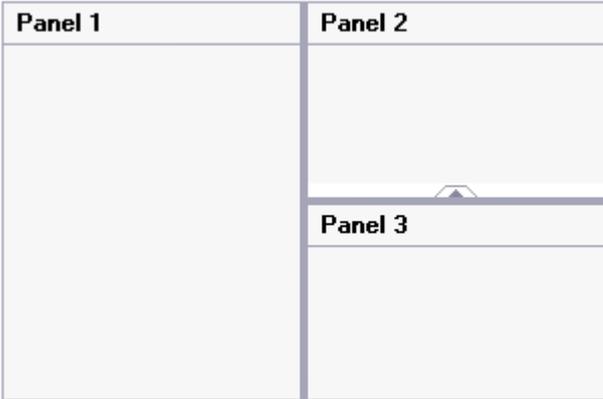
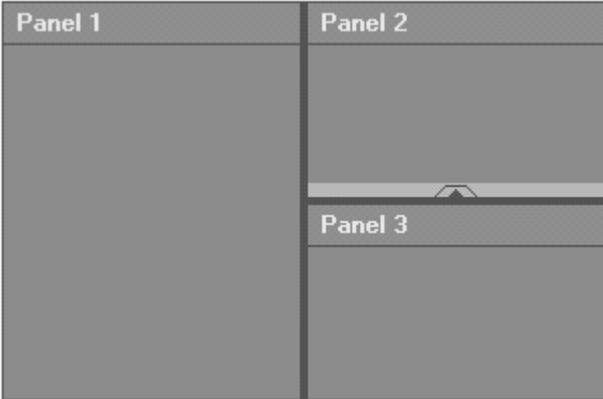
## SplitContainer Visual Styles

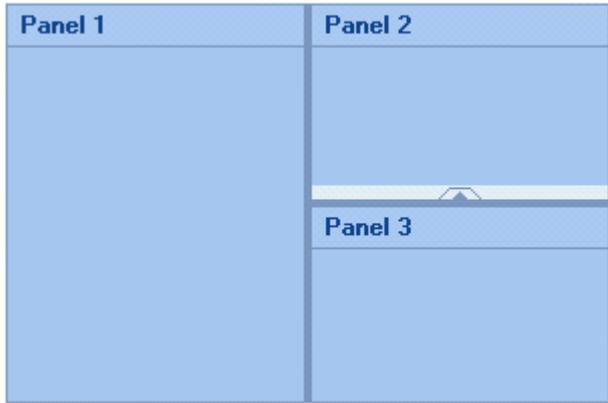
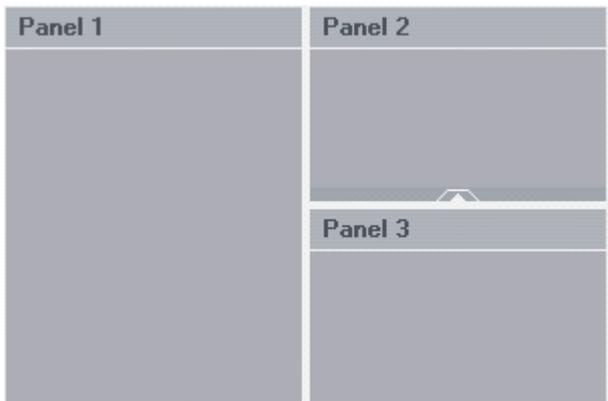
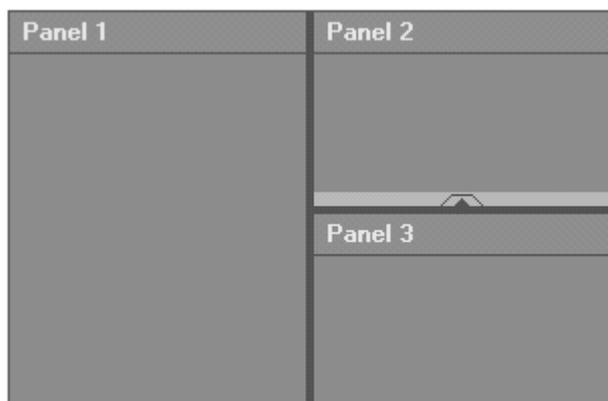
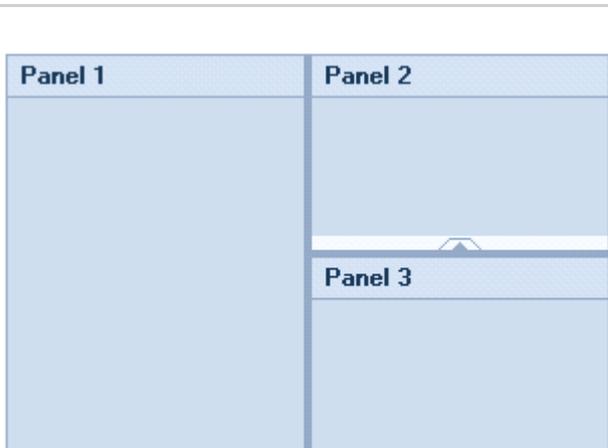
**C1SplitContainer** provides eight built-in visual styles for the control – **System**, **Office2007Black**, **Office2007Blue**, **Office2007Silver**, **Office2010Black**, **Office2010Blue**, **Office2010Silver**, and **Windows7** - that can be easily applied to the control by setting the **VisualStyle** property. The visual styles change the appearance of all of the elements of the C1SplitContainer such as the C1SplitterPanel, splitter bar, expander button, and panel header.

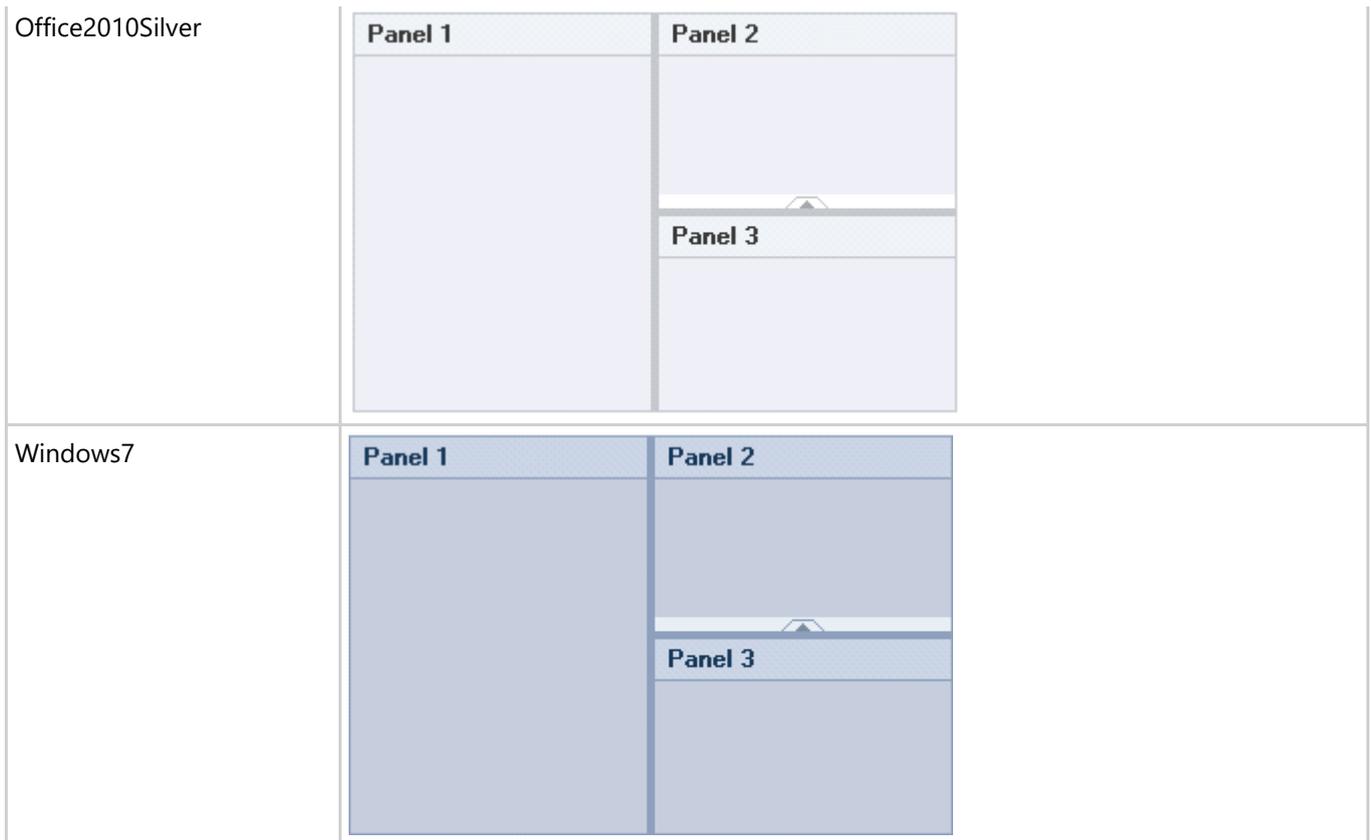
There is no built-in **Custom** visual style in the C1SplitContainer. Any style which doesn't match the predefined styles is considered as Custom. If you want to use a **Custom** style you should assign custom values to any of the following properties: **BackColor**, **ForeColor**, **ToolTipGradient**, **CollapsingAreaColor**, **CollapsingCueColor**, **SplitterColor**, **FixedLineColor**, **HeaderBackColor**, and **HeaderForeColor**. Then, the value of the **VisualStyle** property changes automatically to the **Custom** value. Actually, the **VisualStyle** property is just an indicator, it's not persistable. The value of this property depends on the other properties (listed above).

To learn how to set the **VisualStyle** property, see [Changing the Visual Style](#).

The following table illustrates each of the eight built-in visual schemes.

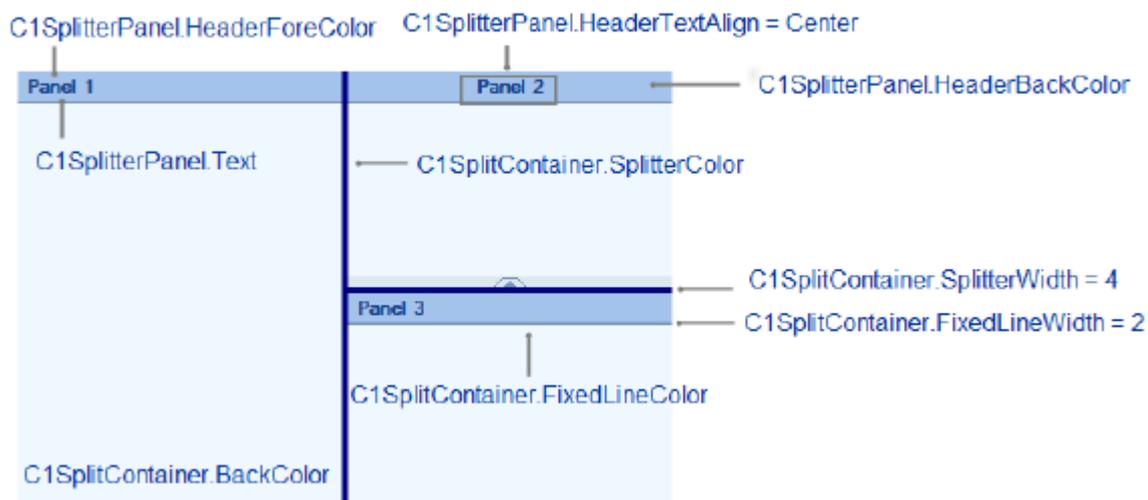
Formatting	Appearance
System	
Office2007Black	

Office2007Blue	 A screenshot of a WinForms SplitContainer in the Office 2007 Blue theme. The container is split vertically into two panels. The left panel is labeled 'Panel 1' and occupies approximately 60% of the width. The right panel is labeled 'Panel 2' and occupies the remaining 40%. Within the right panel, there is a horizontal split. The top part of the right panel is a sub-panel, and the bottom part is labeled 'Panel 3'. A small white arrow icon is visible on the horizontal split line, indicating it is adjustable.
Office2007Silver	 A screenshot of a WinForms SplitContainer in the Office 2007 Silver theme. The layout is identical to the Office 2007 Blue version, with a vertical split into 'Panel 1' and 'Panel 2', and a horizontal split within 'Panel 2' into a sub-panel and 'Panel 3'. A small white arrow icon is visible on the horizontal split line.
Office2010Black	 A screenshot of a WinForms SplitContainer in the Office 2010 Black theme. The layout is identical to the previous versions, with a vertical split into 'Panel 1' and 'Panel 2', and a horizontal split within 'Panel 2' into a sub-panel and 'Panel 3'. A small white arrow icon is visible on the horizontal split line.
Office2010Blue	 A screenshot of a WinForms SplitContainer in the Office 2010 Blue theme. The layout is identical to the previous versions, with a vertical split into 'Panel 1' and 'Panel 2', and a horizontal split within 'Panel 2' into a sub-panel and 'Panel 3'. A small white arrow icon is visible on the horizontal split line.



## SplitContainer Appearance Properties

This section describes a few of C1SplitContainer and C1SplitterPanel's main appearance properties along with a C1SplitContainer showing the effect of each property setting on the C1SplitContainer.



The following table describes each of the C1SplitContainer appearance properties used in the image above:

Property	Description
<b>C1SplitContainer.BackColor</b>	Gets the background color of the C1SplitContainer control.

<a href="#">SplitterWidth</a>	Gets the width of the splitter element.
<a href="#">SplitterColor</a>	Gets the color of the splitter element.
<a href="#">FixedLineColor</a>	Gets the color of the fixed dividing line.
<a href="#">FixedLineWidth</a>	Gets the width of the fixed dividing line.

The following table describes each of the C1SplitterPanel appearance properties used in the image above:

Property	Description
<a href="#">HeaderBackColor</a>	Gets the background color of the header in the C1SplitterPanel.
<a href="#">HeaderForeColor</a>	Gets the forecolor of the header in the C1SplitterPanel.
<a href="#">HeaderTextAlign</a>	Gets the text alignment for the header in the C1SplitterPanel.
<a href="#">Text</a>	Gets the text for the header in the C1SplitterPanel.

For task-based help on modifying the C1SplitterPanel's appearance, see [Changing the C1SplitContainer's Appearance](#).

## SplitContainer Layout

**C1SplitContainer** has a WYSIWYG editor that allows you to view the end result without having to run the project. With the WYSIWYG designer interface **C1SplitContainer** supports, it makes it simple to arrange child controls in the containers of the panels because **C1SplitContainer** displays them as it would at run time. You can add as many child controls to each panel by dragging and dropping each control into the desired panel.

Panels can be selected on the Form by clicking anywhere inside the rectangular box.

The WYSIWYG designer interface makes it simple to add child controls to a specific splitter panel via drag and drop. You also can add child controls programmatically to a specific splitter panel through the **C1SplitterPanel** object since it is a control.

To add a **C1SplitterPanel** programmatically to a specified panel, use the following code:

### Visual Basic

#### Visual Basic

```
Private Sub Form1_Load(sender As Object, e As EventArgs)
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'create a new panel for the split container
    Dim panel1 As New C1SplitterPanel()
    Dim panel2 As New C1SplitterPanel()
    'add panel1 to the splitcontainer
    split.Panels.Add(panel1)
    split.Panels.Add(panel2)
    Controls.Add(split)
End Sub
```

### To write code in C#

#### C#

```
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panel1 = new C1SplitterPanel();
    C1SplitterPanel panel2 = new C1SplitterPanel();
    //add panel1 to the splitcontainer
    split.Panels.Add(panel1);
    split.Panels.Add(panel2);
    Controls.Add(split);
}
```

For more information on adding multiple panels to the **C1SplitContainer** see, [Adding Multiple Panels to the C1SplitContainer](#).

## SplitContainer Layout Options

The default **C1SplitContainer** has a simple horizontal layout. The horizontal layout has one panel with a header. The

default C1SplitterPanel appears docked to the top. When another C1SplitterPanel is added it appears below the initial C1SplitterPanel.



The C1SplitterPanel's layout can easily be controlled by the [Dock](#) property. The C1SplitterPanel's can be docked to the top, left, right or bottom on the container that the C1SplitterPanel has been assigned to.

Use the Dock property to define how a C1SplitterPanel is automatically resized as its parent control is resized. For example, setting the Dock to **Left** causes the panel to align itself with the left edges of its parent control and to resize as the parent control is resized. If Left, Right, Top, or Bottom is selected, the specified and opposite edges of the control are resized to the size of the containing **C1SplitContainer's** corresponding edges.

The [PanelDockStyle](#) settings are:

Member Name	Description
Top	The C1SplitterPanel's top edge is docked to the top of its containing C1SplitContainer.
Bottom	The C1SplitterPanel's bottom edge is docked to the bottom of its containing C1SplitContainer.
Left	The C1SplitterPanel's left edge is docked to the left edge of its containing C1SplitContainer.
Right	The C1SplitterPanel's right edge is docked to the right edge of its containing C1SplitContainer.

The last visible panel in the [Panels](#) collection is the special case. It always behaves as if its Dock property is set to Fill (all four sides of the control are resized to match the containing control's edges). Values of the following properties are ignored for the last visible panel: **Collapsible**, **ResizeWhileDragging**, **Dock**, **SizeRatio**, **KeepRelativeSize**, and **Resizable**. Also, an attempt to change the **Width** or the **Height** property of such a panel has no visible effect. If this panel becomes not the last, or if any of the subsequent panels becomes visible, all the mentioned properties start to work as one may expect.

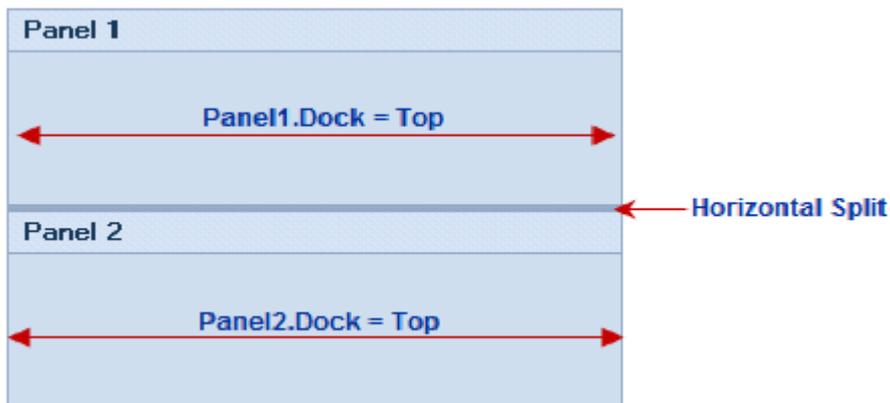
C1SplitterPanel's Dock property makes it easy to create the following splits in the C1SplitContainer:

- Horizontal Split
- Vertical Split
- Nested Split
- Full Split

## Horizontal Split

A horizontal split is the default split type for the C1SplitContainer. It divides the panels into two or more rows and is represented in the window's page by one or more splitter bars. In a horizontal split the panels are both docked to the top.

The following image illustrates a horizontal split:

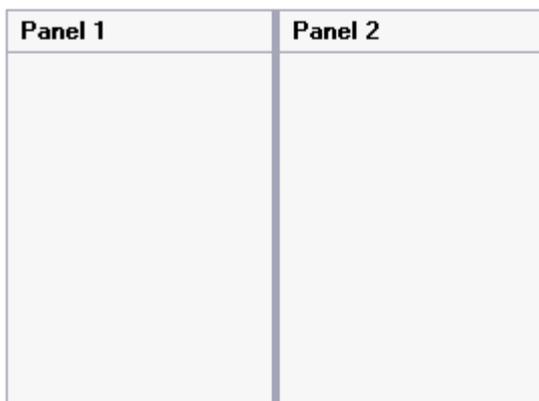


For task-based help on creating a horizontal split, see [Creating a Horizontal Split](#).

## Vertical Split

A vertical split divides the panels into two or more columns and is represented in the Window page by one or more splitter bars.

The following image illustrates a vertical split:



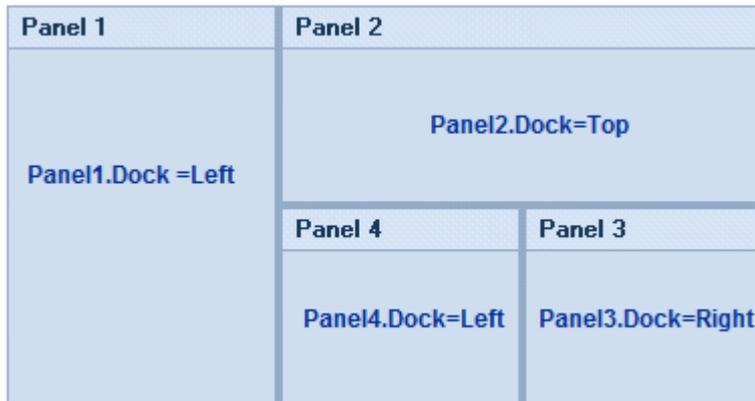
For task-based help on creating a vertical split, see [Creating a Vertical Split](#).

## Compound Split

A compound split is a nested split, meaning that the initial [C1SplitContainer](#) control contains one or more [C1SplitContainer](#) controls. A compound split can contain two or more vertical splitters, two or more horizontal splitters, or a combination of vertical and horizontal splitters.

A compound split can be created by added two or more [C1SplitterPanels](#) into the panel of the parent [C1SplitContainer](#). For more information on creating a compound split, see [Creating a Nested Split](#).

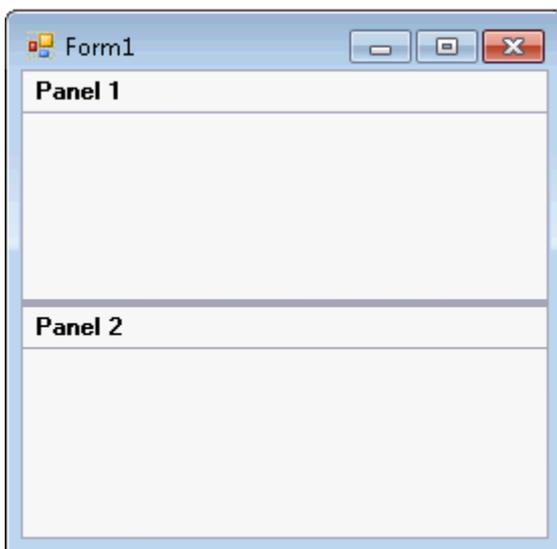
The following image illustrates a compound split:



For task-based help on creating a compound split, see [Creating a Nested Split](#).

## Full-Size Split

A full-size split is a horizontal or vertical split that stretches to fill the content area of a window. The following image illustrates a full-size split:



To create a full-size split, set the **C1SplitContainer.Dock** property to **Fill**. For task-based help on creating a full-size split, see [Creating a Full-Size Split](#).

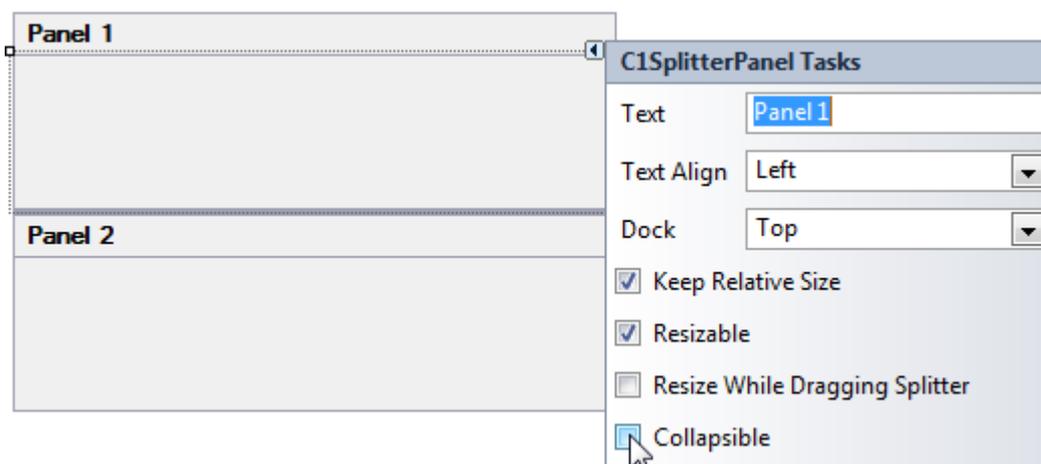
## SplitterPanel Behavior

The following section details the behavior properties used to control the behavior of the splitter panels.

## Collapsible and Expandable Panels

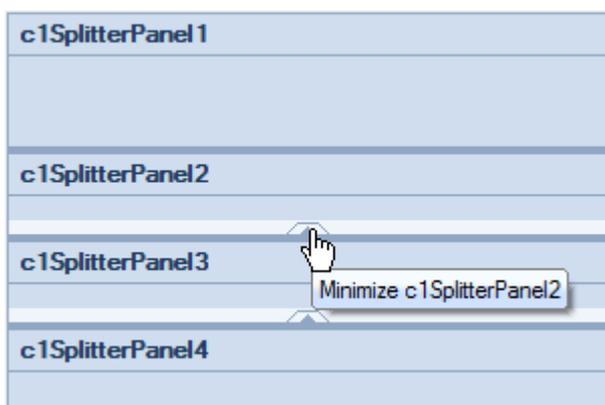
The panels in [C1SplitContainer](#) can easily be collapsed or expanded by setting the [Collapsible](#) property to **True** (see [Adding a Tooltip to the Collapsed Splitter Panel](#)). Once the property is set, you can click on the expander button to collapse or expand the panels.

The following image illustrates Panel1's Collapsible property set to **True**:



When the Collapsible property is set to **True** you can enter text for the collapsed or collapsing tooltip using the [CollapsedToolTip](#) and [CollapsingToolTip](#) properties.

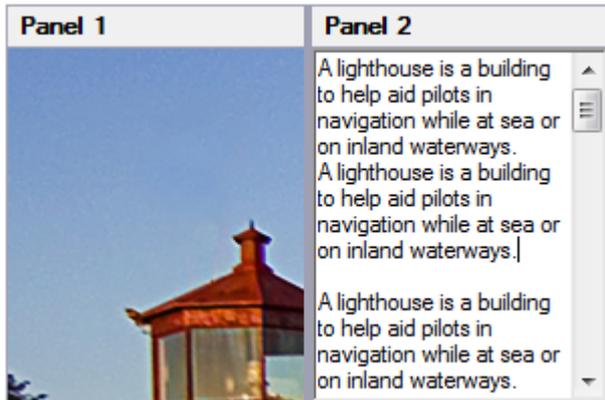
The following image shows the default tooltip when you hover over the expanded button:



When you hover over the expander button of a minimized panel the default tooltip text will read, expand panel.

## SplitterPanel Scrolling

The panels in the [C1SplitContainer](#) control are scrollable by default.



To disable scrolling in a specific panel, set the **C1SplitterPanel.AutoScroll** to False.

## SplitterPanel and Splitter Bar Sizing

You can determine the panel's minimum size as well as determine how the panels size when you move the splitter bar.

### Controlling the Panel Size

You have complete control over the size of the panels in the **C1SplitContainer** control. You can specify the minimum width or height of each panel to prevent the user from resizing it too small through the **MinWidth** property (see [Setting a Minimum Size for a Splitter Panel](#)). This allows more flexibility in the interface for the designer and prevents important information from being hidden.

### Splitter Bar Width

You can specify the width of your divider through the **SplitterWidth** property. To create a more defined split, you can increase the width of your splitter bar. The default splitter width size is 4 pixels.

## SplitterPanel Header and Splitter Bar Drawing Events

If you want to draw the custom header for a **C1SplitterPanel** or draw the custom Splitter Bar you can use the [DrawHeader](#) and [DrawSplitter](#) events.

The [DrawHeader](#) event is fired when the **C1SplitContainer** control needs to repaint the header of the specified **C1SplitterPanel**. The clipping region is restricted by the panel header area.

The [DrawSplitter](#) event is fired to draw the splitter element (the user can drag this element to resize the panel). The clipping region is restricted by the splitter element.

You can take a look at the [LayoutDemo](#) sample for an example of using both events.

## SplitContainer for WinForms Samples

Please be advised that this ComponentOne software tool is accompanied by various sample projects and/or demos which may make use of other development tools included with the ComponentOne Studio.

Please refer to the pre-installed product samples through the following path:

### **Documents\ComponentOne Samples\WinForms**

The following table provides a short overview of each sample.

Sample	Description
VisualStyles	Illustrates <b>C1SplitContainer</b> 's built-in visual styles.
LayoutDemo	Demonstrates different layout options for the C1SplitterPanel's such as fixed, nested, and absolute panels.

## SplitContainer for WinForms Task-Based Help

The task-based help section assumes that you are familiar with programming in the Visual Studio environment and have a general understanding of the **SplitContainer** control.

Each topic provides a solution for specific tasks using the [C1SplitContainer](#) control. By following the steps outlined in each topic, you will be able to create projects using a variety of **C1SplitContainer** features.

## Adding Content to the Splitter Panels

A [C1SplitContainer](#) control can hold arbitrary controls or display text. The following topics will instruct you on adding content to the pages of your [C1SplitContainer](#) control.

## Adding Arbitrary Controls to the C1SplitterPanel

You can add arbitrary controls to each [C1SplitterPanel](#) of the [C1SplitContainer](#) control using a simple drag-and-drop operation. In this topic, you will add a **Button** control to **Panel1** and a **TextBox** control to **Panel2**.

### In Design View:

Complete the following steps:

1. Add [C1SplitContainer](#) to the form.
2. Click on the [C1SplitContainer](#)'s smart tag to open its tasks menu.
3. Select **Add Panel** twice to add two panels to the **C1SplitContainer** control.
4. Click inside **Panel 1** and open its tasks menu. The **C1SplitterPanel Tasks** menu appears.
5. Select a **Button** control from the Visual Studio Toolbox and drag it into the **Panel 1**.

Select a **TextBox** control from the Visual Studio Toolbox and drag it into **Panel 2**.

### In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

#### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

#### To write code in C#

```
C#
using C1.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

#### Visual Basic

## Visual Basic

```
Private Sub Form1_Load(sender As Object, e As EventArgs)
    ' Create a Button control and add text to it
    Dim button1 As New Button()
    button1.Text = "Hello World!"
    ' Create a TextBox control
    Dim textbox1 As New TextBox()
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'create a new panel for the split container
    Dim panel1 As New C1SplitterPanel()
    Dim panel2 As New C1SplitterPanel()
    'add panel1 to the splitcontainer
    split.Panels.Add(panel1)
    split.Panels.Add(panel2)
    panel1.Text = "Panel 1"
    panel2.Text = "Panel 2"
    'add the splitcontainer
    Controls.Add(split)
    'add the button control to panel1
    panel1.Controls.Add(button1)
    'add the textbox control to panel2
    panel2.Controls.Add(textbox1)
End Sub
```

## To write code in C#

## C#

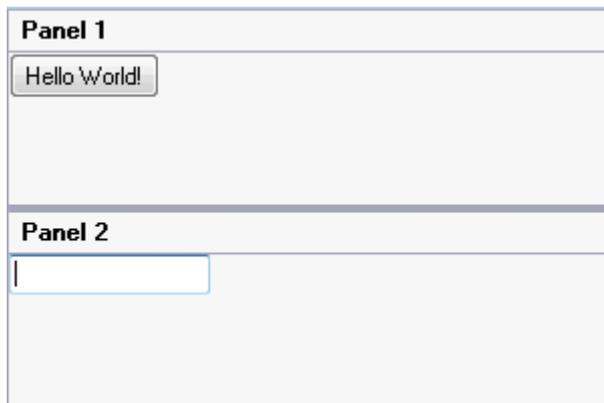
```
private void Form1_Load(object sender, EventArgs e)
{
    // Create a Button control and add text to it
    Button button1 = new Button();
    button1.Text = "Hello World!";
    // Create a TextBox control
    TextBox textbox1 = new TextBox();
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panel1 = new C1SplitterPanel();
    C1SplitterPanel panel2 = new C1SplitterPanel();
    //add panel1 to the splitcontainer
    split.Panels.Add(panel1);
    split.Panels.Add(panel2);
    panel1.Text = "Panel 1";
    panel2.Text = "Panel 2";
    //add the splitcontainer
    Controls.Add(split);
    //add the button control to panel1
    panel1.Controls.Add(button1);
    //add the textbox control to panel2
    panel2.Controls.Add(textbox1);
}
```

```
}
```

- Run the program.

✔ **This Topic Illustrates the Following:**

The following graphic depicts a **C1SplitContainer** control with a **Button** control in **Panel1** and a **TextBox** control in **Panel2**.



## Adding Text to a Splitter Panel

In this topic, you will learn how to add text to a [C1SplitContainer](#) control in design view and in code.

### In Design View:

To add text to a panel, add a **RichTextBox** control to panel1, add text to the **RichTextBox** control and then set the **Dock** property to Fill.

1. Add the **C1SplitContainer** to the form.
2. Click on the **C1SplitContainer**'s smart tag to open its tasks menu.
3. Select **Add Panel** once to add another panel to the **C1SplitContainer** control.
4. Click inside **Panel 1** and open its tasks menu. The **C1SplitterPanel Tasks** menu appears.
5. Select a **RichTextBox** control from the Visual Studio Toolbox and drag it into the **Panel 1**.
6. Set the **RichTextBox1.Text** property to "A lighthouse is a building to help aid pilots in navigation while at sea or on inland waterways."
7. Set the **RichTextBox1.Dock** property to Fill.
8. Set the **RichTextBox1.ForeColor** property to "DarkBlue".

### In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

#### Visual Basic

```
Visual Basic  
Imports C1.Win.C1SplitContainer
```

## To write code in C#

C#

```
using C1.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

## Visual Basic

Visual Basic

```
Private Sub Form1_Load(sender As Object, e As EventArgs)
    Dim richtextbox1 As New RichTextBox()
    richtextbox1.Text = "A lighthouse is a building to help aid pilots in
navigation while at sea or on inland waterways."
    richtextbox1.ForeColor = Color.DarkBlue
    richtextbox1.Dock = DockStyle.Fill
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'create a new panel for the split container
    Dim panel1 As New C1SplitterPanel()
    Dim panel2 As New C1SplitterPanel()
    'add panel1 to the splitcontainer
    split.Panels.Add(panel1)
    split.Panels.Add(panel2)
    panel1.Text = "Panel 1"
    panel2.Text = "Panel 2"
    'add the splitcontainer
    Controls.Add(split)
    'add the button control to panel1
    panel1.Controls.Add(richtextbox1)
End Sub
```

## To write code in C#

C#

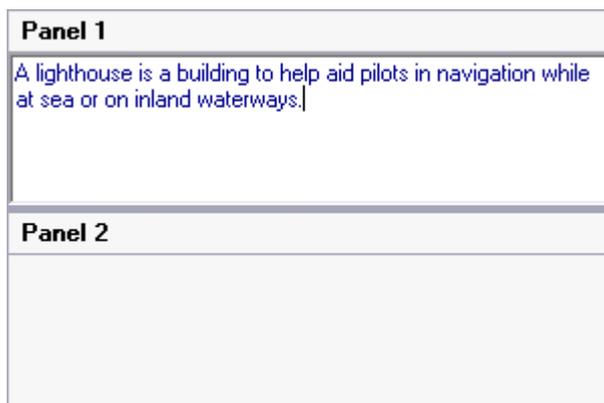
```
private void Form1_Load(object sender, EventArgs e)
{
    RichTextBox richtextbox1 = new RichTextBox();
    richtextbox1.Text = "A lighthouse is a building to help aid pilots in
navigation while at sea or on inland waterways.";
    richtextbox1.ForeColor = Color.DarkBlue;
    richtextbox1.Dock = DockStyle.Fill;
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panel1 = new C1SplitterPanel();
    C1SplitterPanel panel2 = new C1SplitterPanel();
    //add panel1 to the splitcontainer
    split.Panels.Add(panel1);
    split.Panels.Add(panel2);
    panel1.Text = "Panel 1";
}
```

```
panel2.Text = "Panel 2";  
//add the splitcontainer  
Controls.Add(split);  
//add the button control to panel1  
panel1.Controls.Add(richtextbox1);  
}End Sub}
```

- Run the program.

✔ **This Topic Illustrates the Following:**

The following graphic depicts a **C1SplitContainer** control with a **RichTextBox** control docked within **Panel 1**.



## Adding Multiple Panels to the C1SplitContainer

In this topic, you will learn how to add multiple **C1SplitterPanels** to the **C1SplitContainer** control in design view and in code.

### In Design View:

Complete the following steps:

1. Add the **C1SplitContainer** to the form.
2. Click on the **C1SplitContainer**'s smart tag to open its tasks menu.
3. Select **Edit Panels** from the **C1SplitContainer Tasks Menu**. The **C1SplitContainer.Panels Collection Editor** appears.
4. Click on the **Add** button twice to add two more **C1SplitterPanels** to the **C1SplitContainer**. Once a **C1SplitterPanel** is added to the **C1SplitContainer** the properties appear in the right pane so you can easily modify the **C1SplitterPanel**'s settings.
5. Click **OK** to save and close the **C1SplitContainer.Panels Collection Editor**.

### In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

**Visual Basic**

---

## Visual Basic

```
Imports Cl.Win.C1SplitContainer
```

### To write code in C#

## C#

```
using Cl.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

### Visual Basic

## Visual Basic

```
Private Sub Form1_Load(sender As Object, e As EventArgs)
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'create a new panel for the split container
    Dim panel1 As New C1SplitterPanel()
    Dim panel2 As New C1SplitterPanel()
    Dim panel3 As New C1SplitterPanel()
    'add the panels to the splitcontainer
    split.Panels.Add(panel1)
    split.Panels.Add(panel2)
    split.Panels.Add(panel3)
    panel1.Text = "Panel 1"
    panel2.Text = "Panel 2"
    panel3.Text = "Panel 3"
    'add the splitcontainer
    Controls.Add(split)
End Sub
```

### To write code in C#

## C#

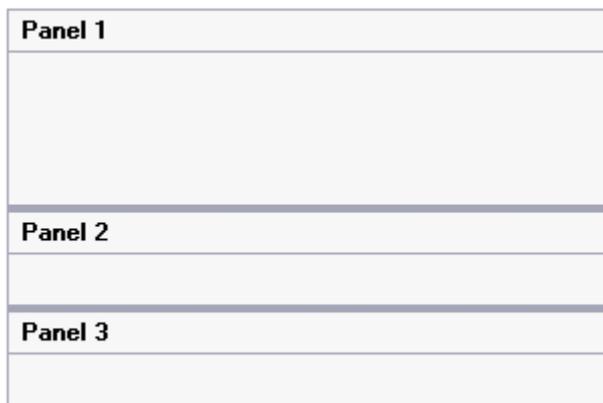
```
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panel1 = new C1SplitterPanel();
    C1SplitterPanel panel2 = new C1SplitterPanel();
    C1SplitterPanel panel3 = new C1SplitterPanel();
    //add panel1 to the splitcontainer
    split.Panels.Add(panel1);
    split.Panels.Add(panel2);
    split.Panels.Add(panel3);
    panel1.Text = "Panel 1";
    panel2.Text = "Panel 2";
    panel2.Text = "Panel 3";
    //add the splitcontainer
    Controls.Add(split);
}
```

```
}
```

4. Run the program.

✔ This Topic Illustrates the Following:

The following graphic depicts a `C1SplitContainer` control with three `C1SplitterPanels`.



## Changing the C1SplitContainers Appearance

The following topics detail how to modify the appearance of a `C1SplitContainer` control using the `C1SplitContainer` and `C1SplitterPanel` properties.

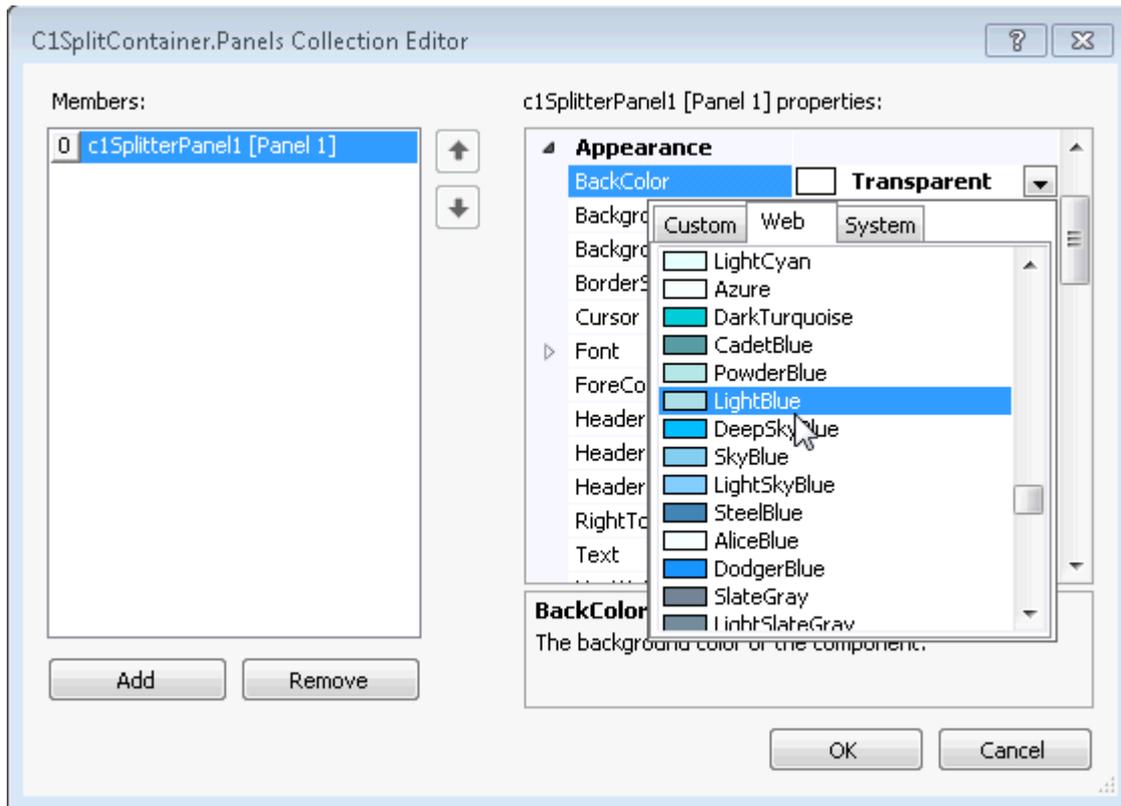
## Changing the Background Color of Panels

You can quickly set the background color of the `C1SplitterPanels` using the `C1SplitterPanel.BackColor` property. In this topic, you'll learn how to apply different colors to each panel of the `C1SplitContainer` control in design view and in code.

### In Design View:

Complete the following steps:

1. Add the `C1SplitContainer` to the form.
2. Click on the `C1SplitContainer`'s smart tag to open its tasks menu.
3. Select **Edit Panels** from the **C1SplitContainer Tasks Menu**. The **C1SplitContainer.Panels Collection Editor** appears.
4. Click Panel 1's **BackColor** drop-down arrow, select the **Web** tab, and choose **LightBlue** from the list and click **OK** to save and close the **C1SplitContainer.Panels Collection Editor**.



## In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

### To write code in C#

```
C#
using C1.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

### Visual Basic

```
Visual Basic
Private Sub Form1_Load(sender As Object, e As EventArgs)
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'create a new panel for the split container
    Dim panel1 As New C1SplitterPanel()
    'add the panel to the splitcontainer
```

```
split.Panels.Add(panell)
    panell.Text = "Panel 1"
    'set Panel 1 bgcolor to lightblue
Panel.BackColor = Color.LightBlue
    'add the splitcontainer
Controls.Add(split)
End Sub
```

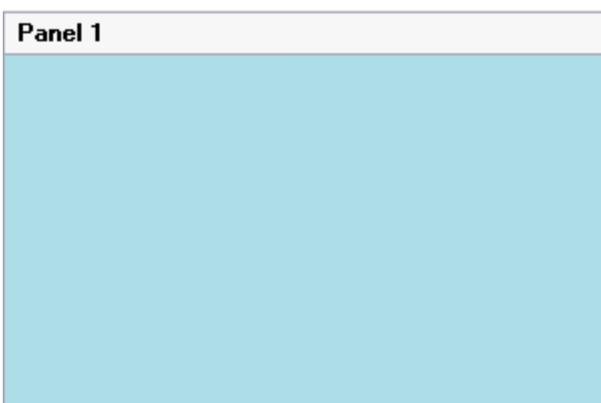
## To write code in C#

```
C#
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panell = new C1SplitterPanel();
    //add panell to the splitcontainer
    split.Panels.Add(panell);
    panell.Text = "Panel 1";
    //set Panel 1 bgcolor to lightblue
    Panel.BackColor = Color.LightBlue;
    //add the splitcontainer
    Controls.Add(split);
}
```

4. Run the program.

## ✔ This Topic Illustrates the Following:

The image below shows Panel 1 with a **LightBlue** background.



## Changing the Border Color

By default, the color of the [C1SplitContainer](#) border is determined by its visual style; however, you can change the color of that border using the **BorderColor** property. In this topic, you will learn how to set the **BorderColor** property in design view and in code.

## In Design View:

Complete the following steps:

1. Add the C1SplitContainer to the form.
2. In the C1SplitContainer Properties window set the **BorderColor** property to "Black".  
The border color is updated on the C1SplitContainer control.

## In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

### To write code in C#

```
C#
using C1.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

### Visual Basic

```
Visual Basic
Private Sub Form1_Load(sender As Object, e As EventArgs)
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'set splitcontainer border color to black
    split.BorderColor = Color.Black
    'add the splitcontainer
    Controls.Add(split)
End Sub
```

### To write code in C#

```
C#
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //set splitcontainer border color to black
    split.BorderColor = Color.Black;
    //add the splitcontainer
    Controls.Add(split);
}
```

4. Run the program.

## ✔ This Topic Illustrates the Following:

The image below shows the C1SplitContainer with a black border.



## Changing the Visual Style

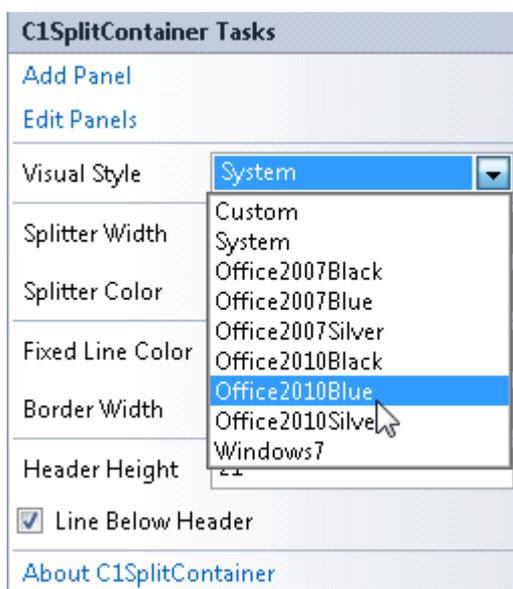
C1SplitContainer features eight embedded visual styles that you can apply to the control with just a few clicks. This topic illustrates how to change the control's visual style in Design view, Source view, and in code.

For more information on visual styles, see [SplitContainer Visual Styles](#).

## In Design View:

Complete the following steps:

1. Click **C1SplitContainer**'s smart tag to open the **C1SplitContainer Tasks** menu.
2. Click the VisualStyle drop-down arrow and then select a visual style from the list. For this example, choose **Office2010Blue**.



The **C1SplitContainer** control adopts the **Office2010Blue** visual style.

## In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

### To write code in C#

```
C#
using C1.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

### Visual Basic

```
Visual Basic
Private Sub Form1_Load(sender As Object, e As EventArgs)
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'set splitcontainer visual style to office2010blue
    split.VisualStudio = VisualStyle.Office2010Blue
    'add the splitcontainer
    Controls.Add(split)
End Sub
```

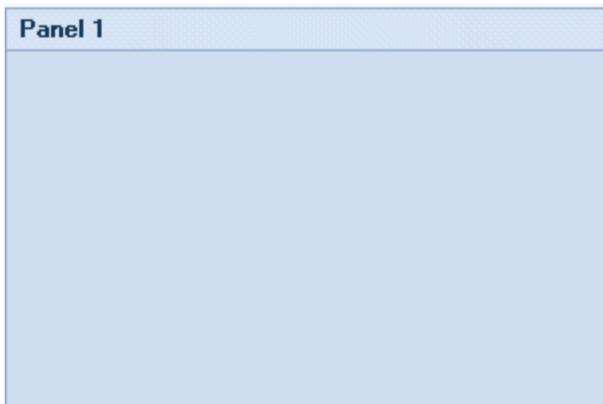
### To write code in C#

```
C#
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //set splitcontainer visual style to office2010blue
    split.VisualStudio = VisualStyle.Office2010Blue
    //add the splitcontainer
    Controls.Add(split);
}
```

4. Run the program.

## This Topic Illustrates the Following:

The following image shows a **C1SplitContainer** with the **Office2010Blue** visual scheme:



## Changing the Splitter Width

You can easily widen the splitter bar using the [SplitterWidth](#) property. In this topic, you will learn how to set the `SplitterWidth` property in design view and in code.

### In Design View:

Complete the following steps:

1. Add the [C1SplitContainer](#) to the form.
2. Click on the C1SplitContainer's smart tag to open its tasks menu.
3. Select **Add Panel** once to add another panel to the **C1SplitContainer** control.
4. In the C1SplitContainer Properties window, set the `SplitterWidth` property to "40".

### In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

#### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

#### To write code in C#

```
C#
using C1.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

#### Visual Basic

```
Visual Basic
Private Sub Form1_Load(sender As Object, e As EventArgs)
```

```
'create new splitcontainer
Dim split As New C1SplitContainer()
'create a new panel for the split container
Dim panel1 As New C1SplitterPanel()
Dim panel2 As New C1SplitterPanel()
'add the panels to the splitcontainer
split.Panels.Add(panel1)
split.Panels.Add(panel2)
panel1.Text = "Panel 1"
panel2.Text = "Panel 2"
'set the splitter width to 40 pixels
Split.SplitterWidth = 40
'add the splitcontainer
Controls.Add(split)
End Sub
```

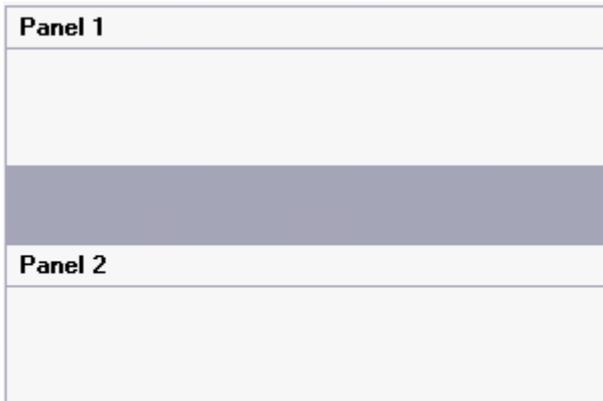
## To write code in C#

```
C#
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panel1 = new C1SplitterPanel();
    C1SplitterPanel panel2 = new C1SplitterPanel();
    //add panel1 to the splitcontainer
    split.Panels.Add(panel1);
    split.Panels.Add(panel2);
    panel1.Text = "Panel 1";
    panel2.Text = "Panel 2";
    //set the splitter width to 40 pixels
    split.SplitterWidth = 40;
    //add the splitcontainer
    Controls.Add(split);
}
```

4. Run the program.

## ✔ This topic illustrates the following:

The following image displays a vertical C1SplitContainer with its SplitterWidth property set to 40 pixels:



## Creating Different Split Types

There are four types of splits that can be created with the **C1SplitContainer** control: horizontal split, vertical split, nested split, and full-size split. This section contains procedures for creating each type of split.

For more information on split types, see [SplitContainer Layout](#).

## Creating a Horizontal Split

Creating a horizontal split is as simple as setting one property. In this topic, you'll learn how to set the **C1SplitContainer.Dock** property in design view and in code.

For more information on horizontal splits, see [Horizontal Split](#).

### In Design View:

Complete the following steps:

1. Add the [C1SplitContainer](#) to the form.
2. Click on the C1SplitContainer's smart tag to open its tasks menu.
3. Select **Add Panel** once to add another panel to the **C1SplitContainer** control. A new panel is added below and the panel's appear horizontal, by default.
4. In the C1SplitterPanel's Properties Window or the C1SplitterPanel Tasks menu and locate the Dock property and notice the [Dock](#) property is set to **Top** by default.

### In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

#### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

#### To write code in C#

C#

```
using Cl.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

## Visual Basic

Visual Basic

```
Private Sub Form1_Load(sender As Object, e As EventArgs)
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'create a new panel for the split container
    Dim panel1 As New C1SplitterPanel()
    Dim panel2 As New C1SplitterPanel()
    'add the panels to the splitcontainer
    split.Panels.Add(panel1)
    split.Panels.Add(panel2)
    panel1.Text = "Panel 1"
    panel2.Text = "Panel 2"
    'add the splitcontainer
    Controls.Add(split)
End Sub
```

## To write code in C#

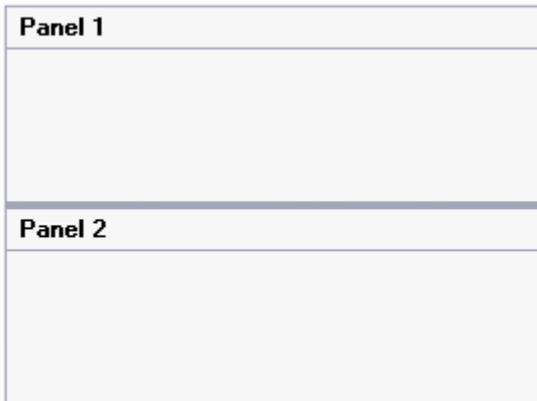
C#

```
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panel1 = new C1SplitterPanel();
    C1SplitterPanel panel2 = new C1SplitterPanel();
    //add panel1 to the splitcontainer
    split.Panels.Add(panel1);
    split.Panels.Add(panel2);
    panel1.Text = "Panel 1";
    panel2.Text = "Panel 2";
    //add the splitcontainer
    Controls.Add(split);
}
```

4. Run the program.

## ✔ This topic illustrates the following:

The splitter bar and panels appear horizontal. The result of this topic will resemble the following image:



## Creating a Vertical Split

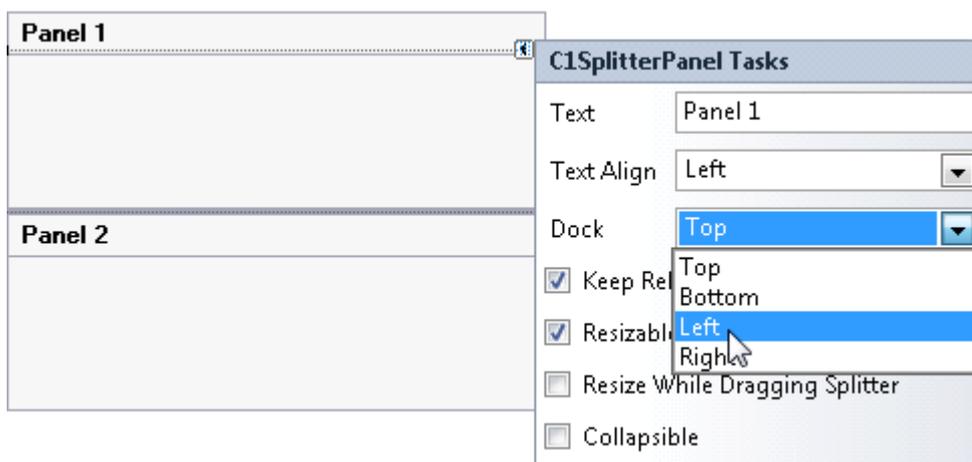
Creating a vertical split is as simple as setting one property. In this topic, you'll learn how to set the [Dock](#) property in design view and in code view.

For more information on vertical splits, see [Vertical Split](#).

### In Design View:

Complete the following steps:

1. Add the [C1SplitContainer](#) to the form.
2. Click on the C1SplitContainer's smart tag to open its tasks menu.
3. Select **Add Panel** once to add another panel to the **C1SplitContainer** control. A new panel is added below and the panels appear horizontal, by default.
4. Select **Panel 1** and set its Dock property to **Left** so it moves the panel to the left side and moves the splitter bar vertically.



### In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

## Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

## To write code in C#

```
C#
using C1.Win.C1SplitContainer
```

3. Add the following code in the **Form\_Load** event:

## Visual Basic

```
Visual Basic
Private Sub Form1_Load(sender As Object, e As EventArgs)
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'create a new panel for the split container
    Dim panel1 As New C1SplitterPanel()
    Dim panel2 As New C1SplitterPanel()
    'add the panels to the splitcontainer
    split.Panels.Add(panel1)
    split.Panels.Add(panel2)
    panel1.Text = "Panel 1"
    panel2.Text = "Panel 2"
    'dock panel 1 to the left
    panel1.Dock = DockStyle.Left
    'add the splitcontainer
    Controls.Add(split)
End Sub
```

## To write code in C#

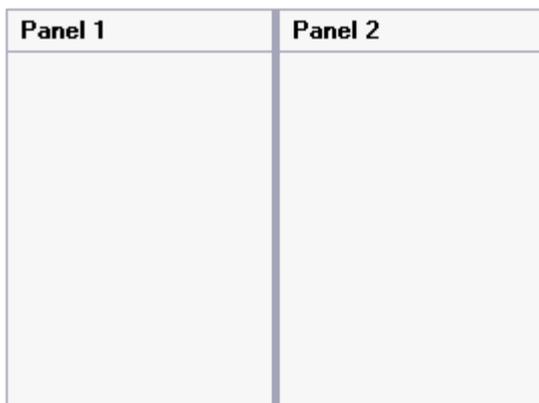
```
C#
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panel1 = new C1SplitterPanel();
    C1SplitterPanel panel2 = new C1SplitterPanel();
    //add panel1 to the splitcontainer
    split.Panels.Add(panel1);
    split.Panels.Add(panel2);
    panel1.Text = "Panel 1";
    panel2.Text = "Panel 2";
    //dock panel 1 to the left
    Panel1.Dock = DockStyle.Left;
    //add the splitcontainer
    Controls.Add(split);
}
```

```
}
```

4. Run the program.

## ✔ This Topic Illustrates the Following:

The splitter bar is now vertical. The final result of this topic will resemble the following image:



## Creating a Nested Split

You can combine multiple [C1SplitContainer](#) objects to create nested splits. In this topic, you will learn how to nest a horizontal split within the first panel of a vertical split.

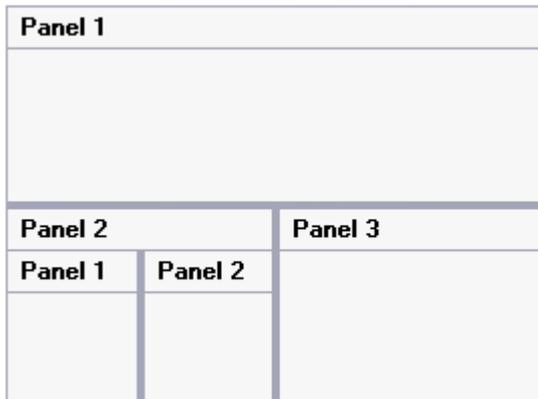
For more information on nested splits, see [Compound Split](#).

Complete the following steps:

1. Add the [C1SplitContainer](#) control to the form.
2. Click on the [C1SplitContainer](#)'s smart tag to open its tasks menu.
3. Select **Add Panel** twice to add two more [C1SplitterPanels](#) to the [C1SplitContainer](#) control.
4. Select Panel 2's smart tag and set its **Dock** property to **Left**.
5. Add another [C1SplitContainer](#) onto Panel 2 and add select **Add Panel** to add another [C1SplitterPanel](#) to [C1SplitContainer2](#).
6. Select Panel 1's smart tag in [C1SplitContainer2](#) and set its **Dock** property to **Left**.

## ✔ This Topic Illustrates the Following:

The following image displays [C1SplitContainer2](#) nested within the bottom left panel of [C1SplitContainer1](#):



## Creating a Full-Size Split

A full-size split is a horizontal or vertical split that stretches to fill the content area of a Windows Form. You can create a full-size split by setting one property: **C1SplitContainer.Dock** to Fill. In this topic, you'll learn how to set the **C1SplitContainer.Dock** property in design view and in code.

For more information on full-size splits, see [Full-Size Split](#).

### In Design View:

Complete the following steps:

1. Add [C1SplitContainer](#) to the form.
2. Click on the C1SplitContainer's smart tag to open its tasks menu.
3. Select **Add Panel** once to add one more [C1SplitterPanel](#) to the C1SplitContainer control.
4. Select C1SplitContainer1 from the Properties list dropdown box and set its **C1SplitContainer.Dock** property to **Fill**.
5. Run the program and observe that the control expands to the width and height of your Windows form.

### In Code View:

Complete the following steps:

1. Add the **C1.Win.C1SplitContainer.dll** reference to your project.
2. Declare the following **C1.Win.C1SplitContainer** namespace at the top of your code page:

#### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

#### To write code in C#

```
C#
using C1.Win.C1SplitContainer;
```

3. Add the following code in the **Form\_Load** event:

#### Visual Basic

## Visual Basic

```
Private Sub Form1_Load(sender As Object, e As EventArgs)
    'create new splitcontainer
    Dim split As New C1SplitContainer()
    'create a new panel for the split container
    Dim panel1 As New C1SplitterPanel()
    Dim panel2 As New C1SplitterPanel()
    'add the panels to the splitcontainer
    split.Panels.Add(panel1)
    split.Panels.Add(panel2)
    panel1.Text = "Panel 1"
    panel2.Text = "Panel 2"
    split.Dock = DockStyle.Fill
    'add the splitcontainer
    Controls.Add(split)
End Sub
```

## To write code in C#

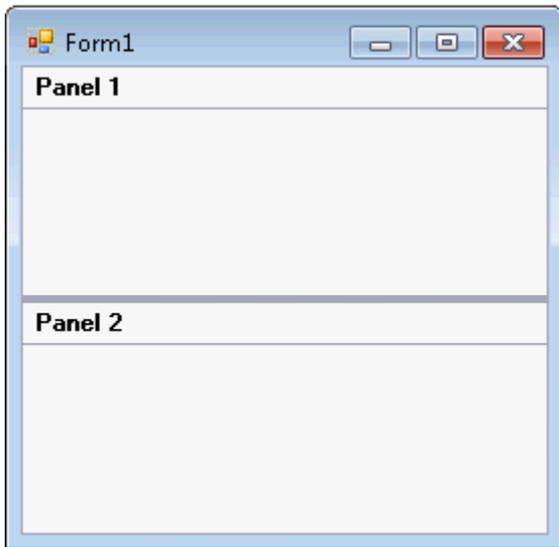
## C#

```
private void Form1_Load(object sender, EventArgs e)
{
    //create new splitcontainer
    C1SplitContainer split = new C1SplitContainer();
    //create a new panel for the split container
    C1SplitterPanel panel1 = new C1SplitterPanel();
    C1SplitterPanel panel2 = new C1SplitterPanel();
    //add panel1 to the splitcontainer
    split.Panels.Add(panel1);
    split.Panels.Add(panel2);
    panel1.Text = "Panel 1";
    panel2.Text = "Panel 2";
    //create a full size split
    split.Dock = DockStyle.Fill;
    //add the splitcontainer
    Controls.Add(split);
}
```

4. Run the program and observe that the control expands to the width and height of your Window Form.

## ✔ This Topic Illustrates the Following:

The C1SplitContainer expands to the width and height of your Window Form.



## Setting C1SplitContainer Behaviors

The [C1SplitContainer](#) control has a list of properties that affect how the control behaves at run time. Some of the properties affect how the control acts when loaded, whereas others affect the users' interactions with the control. The following topics will instruct you on how to modify the run-time actions of the control.

## Adding a ToolTip to the Collapsed Splitter Panel

To add a ToolTip to display text when you hover over the collapsed splitter bar, simply set [CollapsedToolTip](#) property. In this topic, you will learn how to set the [CollapsedToolTip](#) property in design view and in code.

For more information on ToolTips, see [Collapsible and Expandable Panels](#).

### In Design View:

Complete the following steps:

1. Add [C1SplitContainer](#) to the form.
2. Click on the C1SplitContainer's smart tag to open its tasks menu.
3. Select **Add Panel** to add a panel to the **C1SplitContainer** control.
4. Click inside **Panel1** and open its tasks menu. The **C1SplitterPanel Tasks** menu appears.
5. Check **Collapsible** to make **Panel1** collapsible and expandable.
6. Right-click inside **Panel1** and select **Properties**. In the Properties window, set the [CollapsedToolTip](#) to "Click the button to expand Panel 1."

### In Code View:

Complete the following steps:

- Import the following namespace into your project:

#### Visual Basic

```
Visual Basic
```

```
Imports Cl.Win.C1SplitContainer
```

## To write code in C#

C#

```
using Cl.Win.C1SplitContainer;
```

- Add the following code, which sets the CollapsedToolTip property, to the **Page\_Load** event:

## Visual Basic

Visual Basic

```
Panel1.CollapsedToolTip = "Click the button to expand Panel 1."
```

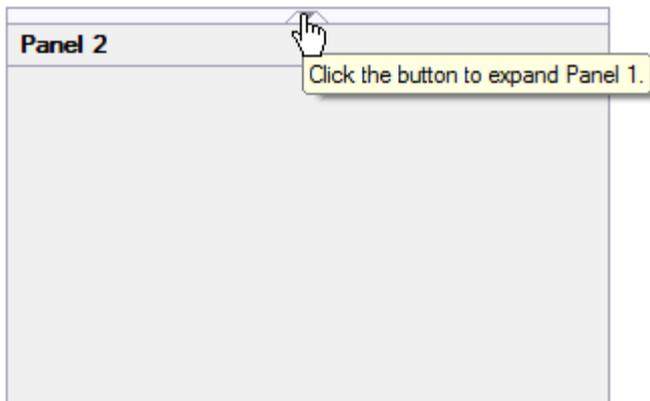
## To write code in C#

C#

```
Panel1.CollapsedToolTip = "Click the button to expand Panel 1.";
```

- Run the program.
  - ✓ **This Topic Illustrates the Following:**

After you've built the project, click the button to collapse panel1 and then hover over the splitter bar and observe that a ToolTip has been added to the collapsed splitter bar. The result will resemble the following image:



## Setting a Minimum Size for a Splitter Panel

In some instances, you might want to keep users from resizing a panel past a certain point. In **Panel1**, for example, you may have a stack of buttons that you want visible at all times. When confronted with that sort of situation, you can use the [MinWidth](#) property to specify, in pixels, the size of the area that you don't want users to drag past. In this topic, you will learn how to set the MinWidth property in Design view and in Code view.

For more information on panel sizing, see [SplitterPanel and Splitter Bar Sizing](#).

## In Design View:

Complete the following steps:

1. Add [C1SplitContainer](#) to the form.
2. Click on the C1SplitContainer's smart tag to open its tasks menu.
3. Select **Add Panel** to add a panel to the **C1SplitContainer** control.
4. Click inside **Panel 1** and open its tasks menu. The C1SplitterPanel Tasks menu appears.
5. Right-click inside **Panel 1** and select **Properties**. In the Properties window, set the MinWidth property for **Panel 1** to "30".

## In Code View:

Complete the following steps:

1. Declare the following namespace into your project:

### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

### To write code in C#

```
C#
using C1.Win.C1SplitContainer;
```

2. Add the following code, which sets the MinWidth property, to the **Page\_Load** event:

### Visual Basic

```
Visual Basic
Panel1.MinSize = 30
```

### To write code in C#

```
C#
Panel1.MinSize = 30;
```

3. Run the program.

 This Topic Illustrates the Following:

Once you've built the project, drag the splitter bar to the left and observe that it remains at 30 pixels.

## Setting a Collapsible Splitter Panel

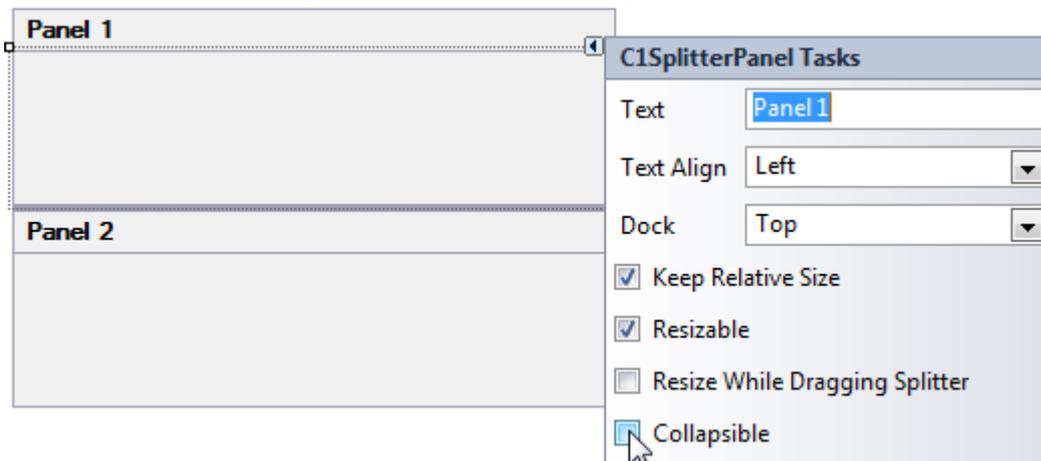
To create a collapsed panel, use the [Collapsible](#) property. In this topic, you will learn how to set the Collapsible property in design view and in code.

For more information on collapsed and expanded panels, see [Collapsible and Expandable Panels](#).

## In Design View

Complete the following steps:

1. Add [C1SplitContainer](#) to the form.
2. Click on the C1SplitContainer's smart tag to open its tasks menu.
3. Select **Add Panel** to add a panel to the **C1SplitContainer** control.
4. Click inside **Panel 1** and open its tasks menu. The C1SplitterPanel Tasks menu appears.



5. Check **Collapsible** to make Panel 1 collapsible and expandable.

## In Code

Complete the following steps:

1. Declare the following namespace into your project:

### Visual Basic

```
Visual Basic
Imports C1.Win.C1SplitContainer
```

### To write code in C#

```
C#
using C1.Win.C1SplitContainer;
```

2. Add the following code, which sets the Collapsible property, to the **Form\_Load** event:

### Visual Basic

```
Visual Basic
Panel1.Collapsible = True
```

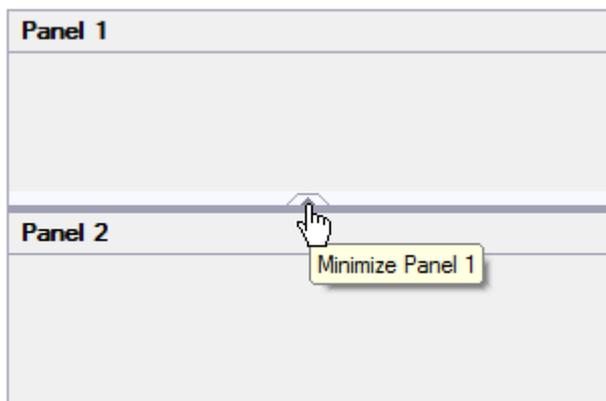
### To write code in C#

```
C#
Panel1.Collapsible = True;
```

3. Run the program.

 **This topic illustrates the following:**

**Panel1** is collapsed.



To expand the panel, simply click the expand button.