

ComboBox for ASP.NET Web Forms

2018.04.24 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
主な特長	3
クイックスタート	4
手順 1: アプリケーションの作成	4
手順 2: コントロールへの項目の追加	4
手順 3: 選択した項目のイベントハンドラの作成	4-5
手順 4: プロジェクトの実行	5
デザイン時のサポート	6
C1ComboBox スマートタグ	6-7
C1ComboBox コンテキストメニュー	7
C1ComboBox のコレクションエディタ	8
C1ComboBox エディタ	8
列デザイナー	8-9
C1ComboBox の外観	10
テーマ	10
C1ComboBox CSS セレクタ	10-11
クライアント側の機能	12
クライアント側イベント	12
タスク別ヘルプ	13
データソースへのコンボボックスの連結	13

製品の概要

ComboBox for ASP.NET Web Forms は、編集が可能なテキストボックスと自動検索が可能なドロップダウンリストを組み合わせたフル機能のコンボボックスコントロールです。

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。

主な特長

ComboBox for ASP.NET Web Forms は、以下の独特な主要機能を備えています。

- **自動検索が可能なドロップダウンリスト**
最初の数文字を入力するだけで、項目をすばやく検索します。ComboBox はリストを自動的に検索し、入力した文字に応じて項目を選択します。
- **ロードオンデマンド**
AJAX を使用してサーバーから項目を動的にロードします。これは、ページを小さく管理可能なサイズに保つのに便利です。
- **項目の選択**
C1ComboBox コントロールには、エンドユーザーがドロップダウンリストで1つまたは複数の項目を選択できるシングルおよびマルチ選択モードが用意されています。
- **テーマ**
スマートタグをクリックするだけで、6種類のプレミアムテーマ (Arctic、Midnight、Aristo、Rocket、Cobalt、および Sterling) のいずれかを選択して外観を変更します。オプションとして、jQuery UI からThemeRoller を使用してカスタマイズしたテーマを作成します。
- **CSS のサポート**
CSS (Cascading Style Sheet) のスタイルを使用して、カスタムスキンを定義します。

クイックスタート

C1ComboBox クイックスタートでは、ASP.NET コントロール **C1ComboBox** の基本的な使用方法を説明します。このクイックスタートでは、1つの **C1ComboBox** コントロールと3つの **C1ComboBoxItem** を含む ASP.NET アプリケーションを作成します。.aspx ファイルで1つの Label コントロールを宣言します。次に、選択したコンボボックス項目の値を表示する **SelectedItem** イベントのイベントハンドラを作成します。

手順 1: アプリケーションの作成

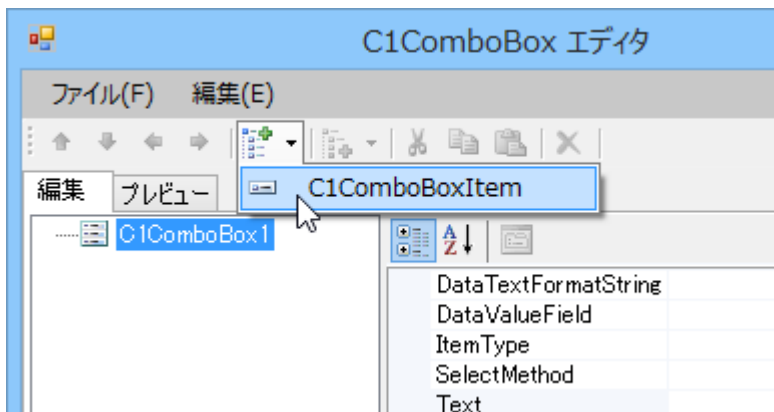
このトピックでは、ASP.NET Web サイトを作成し、**C1ComboBox** コントロールを Default.aspx ページに追加します。

1. まず、ASP.NET Web アプリケーションを作成します。
2. コントロールをツールボックスに追加します。
3. 「デザイン」タブを選択します。
4. デザインビューで、Visual Studio ツールボックスに移動し、**C1ComboBox** コントロールをダブルクリックしてページのメインコンテンツにコンボボックスを追加します。

手順 2: コントロールへの項目の追加

この手順では、C1ComboBox エディタを使用して、実行時にそのドロップ矢印をクリックしたときに表示される C1ComboBox コントロールに項目を追加します。

1. **C1ComboBox** のスマートタグをクリックし、[**C1ComboBox タスク**]メニューから[**項目の編集**]を選択します。**C1ComboBox エディタ**が表示されます。
2. <子項目の追加>ボタンから C1ComboBoxItem を3回クリックして、3つの C1ComboBoxItem を取得します。



3. 各コンボボックスアイテムの Text と Value プロパティを以下のように設定します。
 - C1ComboBoxItem1 の Text プロパティを「小林」、Value プロパティを「小林正男」に設定します。
 - C1ComboBoxItem2 の Text プロパティを「秋本」、Value プロパティを「秋本武」に設定します。
 - C1ComboBoxItem3 の Text プロパティを「中村」、Value プロパティを「中村功」に設定します。
4. <OK>をクリックして C1ComboBox コントロールに変更内容を適用し、C1ComboBox エディタを閉じます。

手順 3: 選択した項目のイベントハンドラの作成

このトピックでは、コンボボックスの項目を選択したときに selectedItem イベントがトリガーされます。

1. 「ソース」タブを選択し、<body> タグの前に下記の **C1ComboBox1_OnClientChanged** 関数のスクリプトを追加します。

スクリプト

ComboBox for ASP.NET Web Forms

```
<script id="script1nit" type="text/javascript">
  function C1ComboBox1_OnClientChanged(e, data) {
    var val = data.selectedItem.value;
    $('#output').html('社員「' + val + '」を選択しました。');
  }
</script>
```

2. ラベルをソースページへC1ComboBox タグの前に追加します。

スクリプト

```
<label id="output">
  社員名を選択してください。</label>
```

3. C1ComboBox タグ内で、onclientchanged プロパティに onclientchanged="C1ComboBox1_OnClientChanged" 関数を割り当てます。その結果、次のようになります。

スクリプト

```
<cc1:C1ComboBox runat="server" Width="160px"
  onclientchanged="C1ComboBox1_OnClientChanged">
```

手順 4: プロジェクトの実行

[F5]を押して、プロジェクトを実行します。次のような表示になります。

- リストから項目を選択し、選択した項目の値が下記のテキストに更新されて表示されることを確認します。



社員「小林正男」を選択しました。

おめでとうございます！ **ComboBox for ASP.NET Web Forms クイックスタート**が正常に完了しました。

デザイン時のサポート

C1ComboBox は、カスタマイズされたコンテキストメニュー、スマートタグ、および充実したデザイン時サポートを提供するデザイナを備えており、オブジェクトモデルの操作が簡素化されています。

以下の各セクションでは、C1ComboBox のデザイン時環境を使用して、**C1ComboBox** コントロールを設定する方法について説明します。

C1ComboBox スマートタグ

Visual Studio では、**C1ComboBox** コントロールにスマートタグが用意されています。スマートタグは、C1ComboBox で最もよく使用されるプロパティを提供するショートカットタスクメニューです。

[**C1ComboBox タスク** メニュー]にアクセスするには、**C1ComboBox** コントロールの右上端にあるスマートタグ () をクリックします。[**C1ComboBox タスク** メニュー]が開きます。

[**C1ComboBox タスク** メニュー]は、以下のように動作します。

- データソースの選択**
 [データソースの選択]項目をクリックすると、既存のデータソースやバインドする新しいデータソースを選択できるドロップダウンリストが開きます。
- 項目の編集**
 [項目の編集]リンク項目をクリックすると、**C1ComboBox エディタ**が開きます。
- 列の編集**
 [列の編集]リンク項目をクリックすると、列デザイナが開きます。
- テーマ**
 [テーマ]ドロップダウンボックスでは、**Theme** プロパティを設定し、**C1ComboBox** コントロールの外観を6種類の定義済みのテーマの1つに変更できます。デフォルトでは、これは **Aristo** テーマに設定されます。使用可能なテーマの詳細については、「[テーマ](#)」を参照してください。
- 新しいテーマの作成**
 [新しいテーマの作成]オプションをクリックすると、**ThemeRoller for Visual Studio** が開きます。したがって、開発環境内でテーマをカスタマイズすることができます。アプリケーションで **ThemeRoller for Visual Studio** を使用する方法については、「[ThemeRoller for Visual Studio](#)」を参照してください。
- CDN の使用**
 [CDN の使用]チェックボックスを選択すると、クライアントリソースが CDN からロードされます。これはデフォルトで

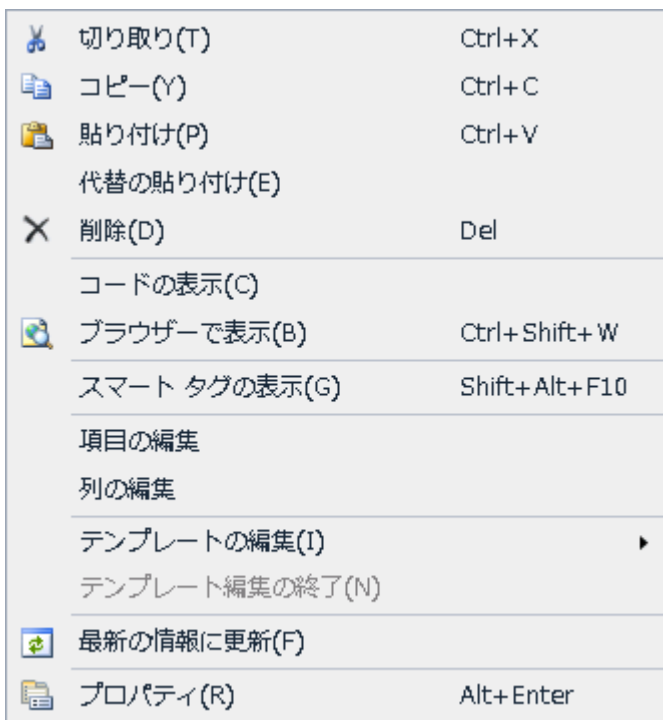
ComboBox for ASP.NET Web Forms

- OFF です。
- **CDN パス**
CDN の URL パスを表示します。
- **Bootstrap の使用**
[**Bootstrap の使用**] オプションを選択すると、コントロールに Bootstrap テーマを適用することができます。アプリケーションで Bootstrap テーマを使用する方法については、「[Bootstrap for ASP.NET Web Forms クイックスタート](#)」を参照してください。
- **バージョン情報**
[**バージョン情報**] をクリックすると、製品のバージョン情報を確認できるダイアログボックスが表示されます。
- **テンプレートの編集**
[**テンプレートの編集**] をクリックすると、C1ComboBox コントロールが **テンプレート編集モード** に切り替えます。**テンプレート編集モード** では、C1ComboBox のタスクメニューが以下のように表示されます。
 - **表示**
[**表示**] ドロップダウン矢印を選択すると、カスタマイズできるアイテムテンプレートが開きます。この一覧からアイテムテンプレートを選択して編集するテンプレートを開きます。
 - **テンプレートの編集を解除**
[**テンプレートの編集を解除**] をクリックすると、テンプレート編集モードを解除し、C1ComboBox のメインタスクメニューに戻します。

C1ComboBox コンテキストメニュー

C1ComboBox には、Visual Studio がすべての .NET および ASP.NET コントロールに提供しているコンテキストメニューで利用できる追加的なコマンドがあります。

リスト上の任意の場所を右クリックし、**C1ComboBox** のコンテキストメニューを表示します。



コンテキストメニューのコマンドは、以下のように動作します。

- **項目の編集**
[**項目の編集**] メニュー項目をクリックすると、**C1ComboBox エディタ**が開きます。
- **列の編集**
[**列の編集**] メニュー項目をクリックすると、**列デザイナー**が開きます。
- **テンプレートの編集**
[**テンプレートの編集**] をクリックすると、C1ComboBox コントロールが **テンプレート編集モード** に切り替えます。

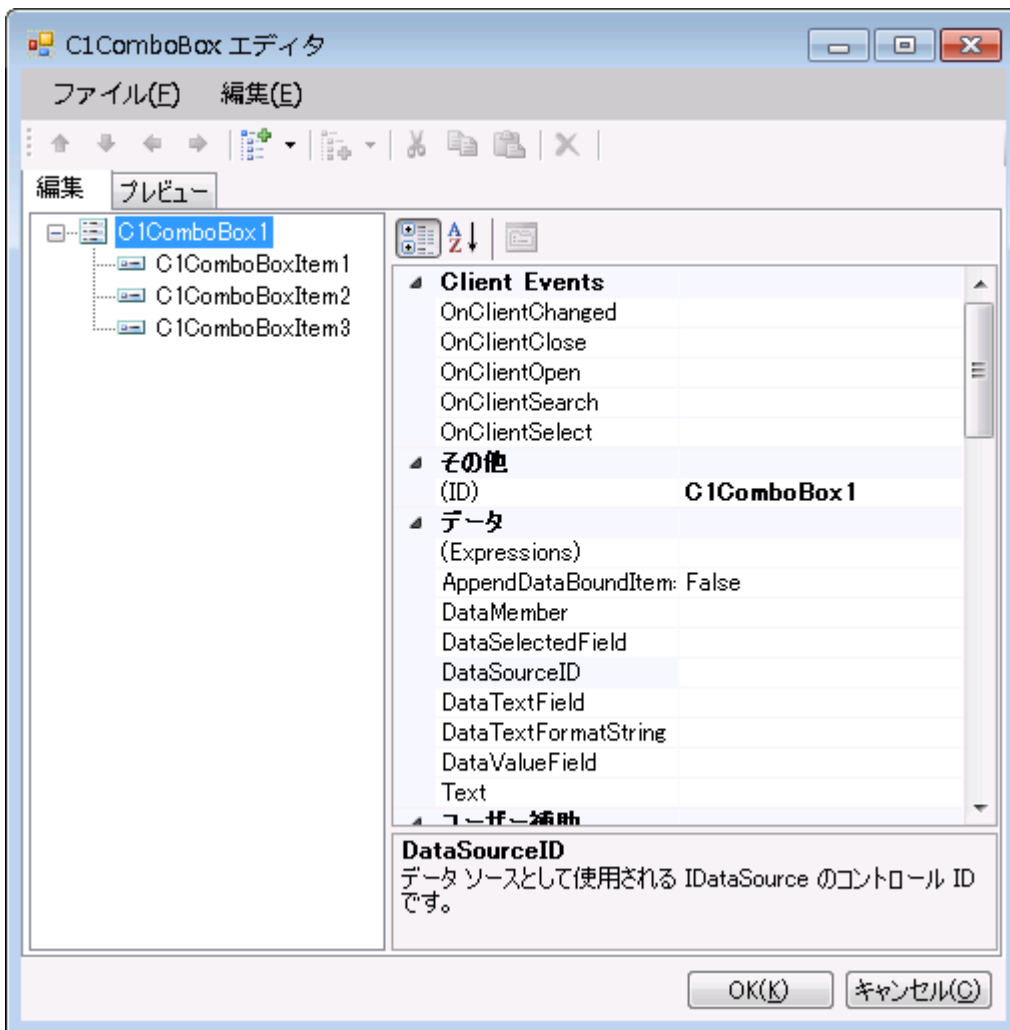
C1ComboBox のコレクションエディタ

C1ComboBox には、C1ComboBoxItem および列を追加／削除／変更するために次の2つのコレクションエディタが組み込まれています。

- **C1ComboBox エディタ**
- **列デザイナー**

C1ComboBox エディタ

C1ComboBox エディタでは、ユーザーは C1ComboBox コントロールに C1ComboBoxItem を追加／削除／変更することができます。

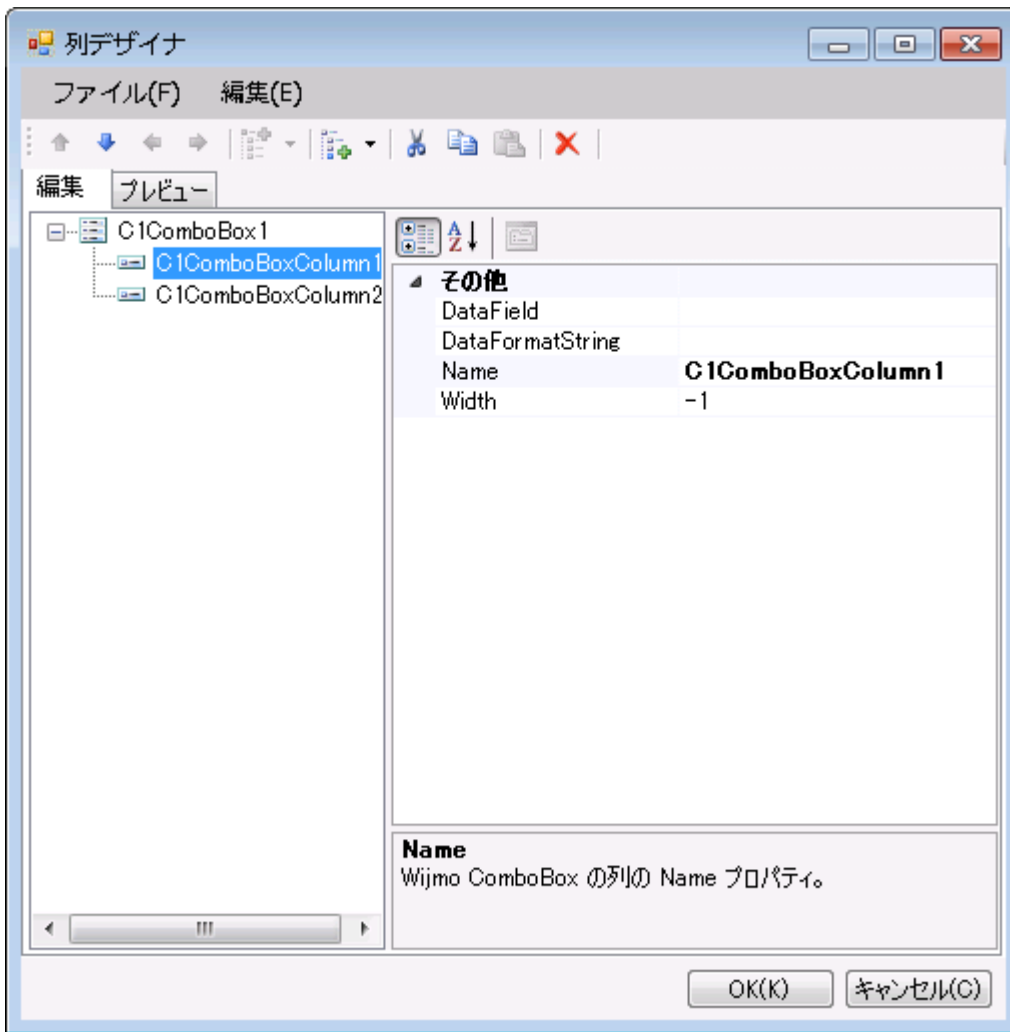


C1ComboBox エディタにアクセスするには、**C1ComboBox** コントロールを右クリックし、コンテキストメニューから[項目の編集]を選択します。

列デザイナー

列デザイナーでは、C1ComboBox コントロールに列を追加／削除／変更できます。

ComboBox for ASP.NET Web Forms



列デザイナーにアクセスするには、C1ComboBox コントロールを右クリックし、コンテキストメニューから[列デザイナー]を選択します。

C1ComboBox の外観

コンボボックスの外観は、組み込みテーマまたは CSS スタイルによって制御されます。コンボボックスのテーマは、コンボボックスの項目や列など、すべてのコンボボックス要素の外観に影響を与えます。

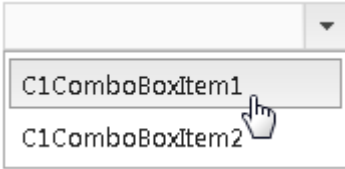
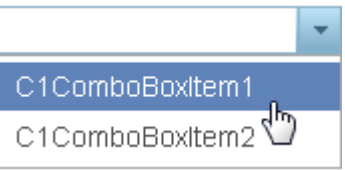
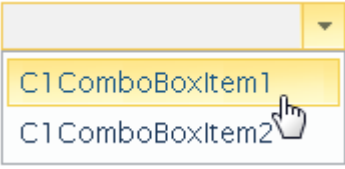
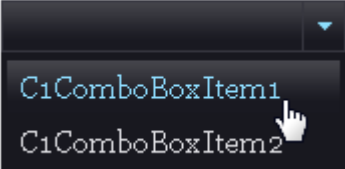
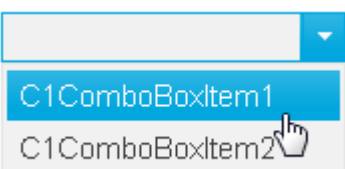
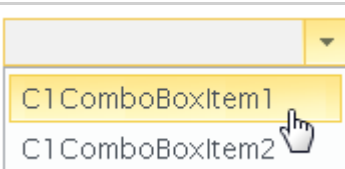
C1ComboBox は、簡単にカスタマイズできるように設計されています。デフォルトの外観を数かぎりなくさまざまに変更できます。

CSS スタイルを適用して、**C1ComboBox** の要素を変更できます。**C1ComboBox** には、**CSSClass** プロパティの横にあるドロップダウン矢印をクリックすると表示される組み込みの CSS セレクタが用意されています。

テーマ

C1ComboBox は、コントロール用に **Arctic**、**Aristo**、**Cobalt**、**Midnight**、**Rocket**、および **Sterling** という6種類の組み込みテーマを備えています。これらは **Theme** プロパティを設定して簡単にコントロールに適用できます。

マウスが上に置かれた状態で表示される6種類の組み込みテーマは次の通りです。

テーマ	外観
arctic	
Aristo(デフォルト)	
Cobalt	
Midnight	
Rocket	
Sterling	

C1ComboBox CSS セレクタ

CSS スタイルを使用して **C1ComboBox** の任意の要素をスタイル設定し、その外観を真に独特のものにすることができます。カスタマイズ処理を簡素化するために、ComponentOne for ASP.NET Web Forms には、その6種類の組み込みテーマごとに CSS セレクタが組み込まれています。

背景、テキスト、フォント、枠、輪郭、マージン、埋め込み、リスト、表などの一般的な CSS プロパティを該当する CSS セレクタに適用できます。

一般に使用される個々の CSS セレクタとグループ化された CSS セレクタのリストについては、プロジェクトの **C1ComboBox** コントロールを選択し、**C1ComboBox Visual Studio プロパティ** ウィンドウで **CssClass** プロパティの横にあるドロップダウンリストを表示します。

以下の表では、一般に使用される個々の CSS セレクタとグループ化された CSS セレクタについて詳細に説明します。個々の CSS セレクタをグループとして組み合わせ、CSS セレクタをより具体的かつ強力なものにすることができます。グループ化された CSS セレクタは、複数の C1ComboBox 要素のスタイルを定義します。

CSS セレクタ	説明
.wijmo-wijcombobox	C1ComboBox コントロールにスタイルを適用します。
.wijmo-wijcombobox-active-prev	前の Active 状態の C1ComboBox にスタイルを適用します。
.wijmo-wijcombobox-cell	C1ComboBox コントロールのセルにスタイルを適用します。
.wijmo-wijcombobox-input	入力 C1ComboBox にスタイルを適用します。
.wijmo-wijcombobox-item ui-state-active	C1ComboBox の Active 状態のリストされない項目にスタイルを適用します。
.wijmo-wijcombobox-item ui-state-hover	C1ComboBox の Hover 状態のリストされない項目にスタイルを適用します。
.wijmo-wijcombobox-label	C1ComboBox のラベルにスタイルを適用します。
.wijmo-wijcombobox-list	C1ComboBox のリストにスタイルを適用します。
.wijmo-wijcombobox-loading	ロード中の C1ComboBox にスタイルを適用します。
.wijmo-wijcombobox-multicolumn	複数列の C1ComboBox にスタイルを適用します。
.wijmo-wijcombobox-row	C1ComboBox の行にスタイルを適用します。
.wijmo-wijcombobox-rowheader	c1combobox の行ヘッダーにスタイルを適用します。
.wijmo-wijcombobox-trigger	トリガーされる c1combobox にスタイルを適用します。
.wijmo-wijcombobox-ul	リストされない c1combobox にスタイルを適用します。
.wijmo-wijcombobox-wrapper	ラッパー c1combobox にスタイルを適用します。

クライアント側の機能

C1ComboBox のクライアント側には、非常に充実したクライアント側オブジェクトモデルがあります。そのメンバは、ほとんどがサーバー側コントロールのメンバと同じです。

C1ComboBox コントロールが表示されると、クライアント側コンボボックスのインスタンスが自動的に生成されます。これは、サーバーにPostBackしなくても、**C1ComboBox** コントロールのプロパティやメソッドにアクセスできるということです。

C1ComboBox のクライアント側コードを使用すれば、時間をかけて Web サーバーに情報を送信しなくても、Web ページに多くの機能を実装できます。そのため、C1ComboBox のクライアント側オブジェクトモデルを使用して、Web サイトの効率を高めることができます。

クライアント側イベント

C1ComboBox には、複数のクライアント側イベントが含まれています。それらを利用すれば、項目の選択、ドロップダウンリストを開く、ドロップダウンリストを閉じるなどの処理が行われたときに、C1ComboBox コントロールのコンボボックス項目を操作できます。

各クライアント側イベントには、送信側の**C1ComboBox** を識別する ID である2つのパラメータが必要です。

クライアント側イベントの表にリストされたサーバー側プロパティを使用して、特定のクライアント側イベントに反応する JavaScript 関数の名前を指定できます。たとえば、「CloseList」という JavaScript 関数を割り当てて、閉じたドロップダウンリストに回答させるには、**OnClientClose** プロパティを CloseList に設定します。

下の表に、クライアントスクリプトで使用できるイベントを示します。これらのプロパティはサーバー側で定義されていますが、実際のイベントや各 JavaScript 関数用に宣言する名前はクライアント側で定義されます。

クライアント側イベント表

イベントのサーバー側プロパティ名	イベント名	説明>
OnClientChanged	Changed	選択項目が変更されたときに呼び出される関数。
OnClientClose	Close	ドロップダウンリストが閉じられたときに呼び出される関数。
OnClientOpen	Open	ドロップダウンリストが開かれたときに呼び出される関数。
OnClientSearch	Search	リストを検索する前に呼び出される関数。
OnClientSelect	Select	リスト内の項目が選択されたときに呼び出される関数。

タスク別ヘルプ

タスク別ヘルプは、Visual Studio のプログラミングに精通しているユーザーを対象としています。ヘルプに記述された手順に従うことによって、**ComboBox for ASP.NET Web Forms** のさまざまな機能を実証するプロジェクトを作成して、**ComboBox for ASP.NET Web Forms** の機能を理解することができます。

データソースへのコンボボックスの連結

下記の手順に従ってコンボボックスをデータベースと連結することができます。

1. **C1ComboBox** コントロールをページに追加します。
2. 適当なテーブルからデータを取得します。この例では、製品と共に C:\Users\<ユーザー名>\Documents\ComponentOneSamples\Common に自動的にインストールされる **NWind.mdb** を使用します。自分のプロジェクトの **App_Data** フォルダに **NWind.mdb** をコピーします。
3. **DataSourceID** プロパティにデータソースのIDを設定し、テキスト値を読み込むデータソースフィールドを設定します。