

FileExplorer for ASP.NET Web Forms

2018.04.12 更新

グレープシティ株式会社

目次

製品の概要	2
ComponentOne for ASP.NET Web Formsのヘルプ	2
ビジュアル要素	3
クイックスタート:フォルダパスの追加	4-5
エクスプローラーの機能	6
複数ファイルの選択	6-7
フォルダの作成と管理	7
ファイル操作の無効化	7
キーボードのユーザー補助機能	8-9
ソートとフィルタ処理	9-11
ページング	11-12
表示モード	12-14
外観のカスタマイズ	14-17
クライアントイベント	17-18

製品の概要

FileExplorer for ASP.NET Web Forms は、ファイルやフォルダの整理に役立ちます。ComponentOne のコントロールを使用して開発されており、Windows エクスプローラーと同様の機能を提供します。作成、削除、名前変更、ドラッグ & ドロップなどの基本的なファイル操作をすべて FileExplorer で実行できます。

主要な機能

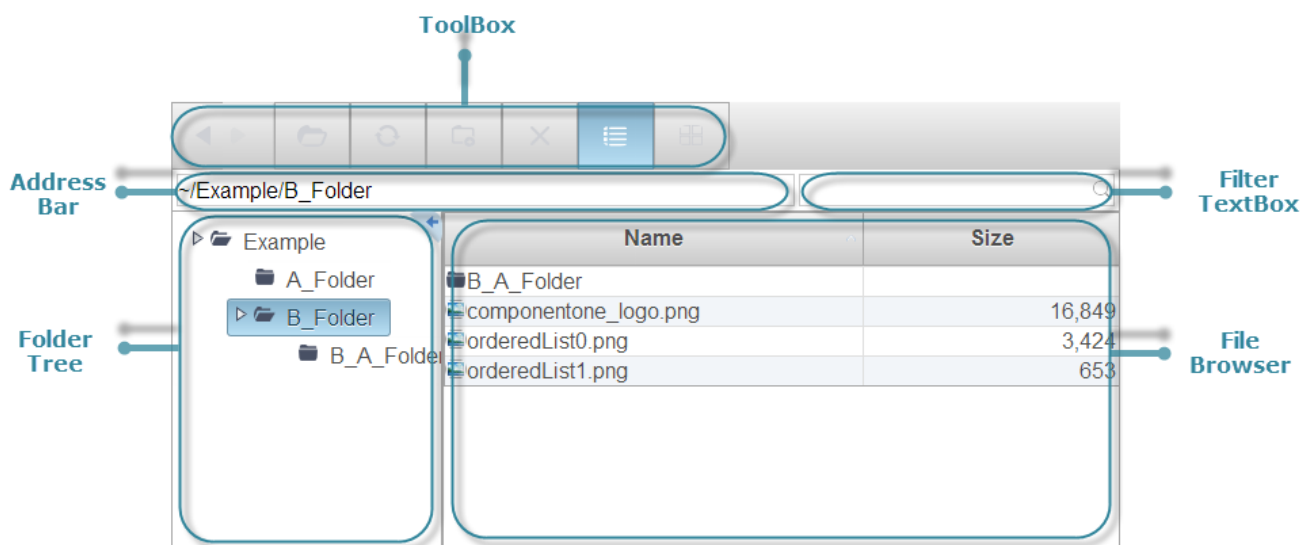
- **複数ファイルの選択**: 複数のファイルまたはフォルダを選択できます。
- **サムネイルの自動生成**: 選択された画像ファイルのサムネイルが自動的に表示されます。
- **フォルダの管理**: 切り取り、貼り付け、削除、名前の変更など、基本的なファイル操作をすべて行えます。
- **フォルダの新規作成**: コントロールを使用して新しいフォルダを作成できます。
- **キーボードサポート**: 主要なファイル操作のキーボードショートカットを追加できます。
- **ソート**: 名前に基づいてファイルやフォルダを自動的にソートできます。
- **フィルタ処理**: 組み込みフィルタを使用して、リスト内のファイルをすばやく見つけることができます。
- **ページング**: 右ペインに表示された項目のページングを有効にできます。
- **ドラッグ & ドロップ**: フォルダ間でファイルをドラッグアンドドロップできます。

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。

ビジュアル要素

次の図は、FileExplorer に表示されるコントロールを示しています。



- **ツールボックス**: ツールボックスは、次の表で説明するいくつかのボタンで構成されます。

アイコン	ボタン	説明
	戻る/進む	前のフォルダに戻る、または次のフォルダに進みます。
	開く	選択されたファイルを開きます。
	更新	フォルダの内容を更新します。
	新しいフォルダ	選択されているフォルダ内に新しいフォルダを作成します。
	削除	選択されているファイルまたはフォルダを削除します。
	リストビュー	フォルダ内のファイルをリストとして表示します。
	サムネイルビュー	フォルダ内のファイルをサムネイルとして表示します。

- **アドレスバー**: 開いているフォルダのアドレスを表示します。
- **フィルタテキストボックス**: このテキストボックスに入力された文字に基づいて、右ペインに表示されるファイルとフォルダをフィルタ処理します。
- **フォルダツリー**: すべてのフォルダとサブフォルダを表示します。
- **ファイルブラウザー**: フォルダツリーで選択されているフォルダ内にあるすべてのファイルとフォルダを表示します。

クイックスタート:フォルダパスの追加

FileExplorer を Web フォームに追加し、初期フォルダパスを追加するには、次の手順に従います。

手順1/2:フォームへのコントロールの追加

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 FileExplorer 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 FileExplorer を見つけます。

トピックの内容

- 手順1/2:フォームへのコントロールの追加
- 手順2/2:初期フォルダパスの追加

手順2/2:初期フォルダパスの追加

1. FileExplorer コントロールを右クリックし、**[プロパティ]** を選択します。
2. **[プロパティ]** ウィンドウで、**InitPath** プロパティにフォルダの場所を追加します。

ソースビューの場合

<cc1:C1FileExplorer> タグ内の **InitPath** プロパティに初期フォルダパスを追加します。

ASP.Net

```
<cc1:C1FileExplorer ID="C1FileExplorer1" InitPath="~/Example" runat="server" >
</cc1:C1FileExplorer>
```

コードの場合

次のコードを **Page_Load** イベントに追加して、**InitPath** プロパティに初期フォルダパスを追加します。

C# でコードを書く場合

C#

```
C1FileExplorer1.InitPath= "Example\\A_Folder";
```

Visual Basic でコードを書く場合

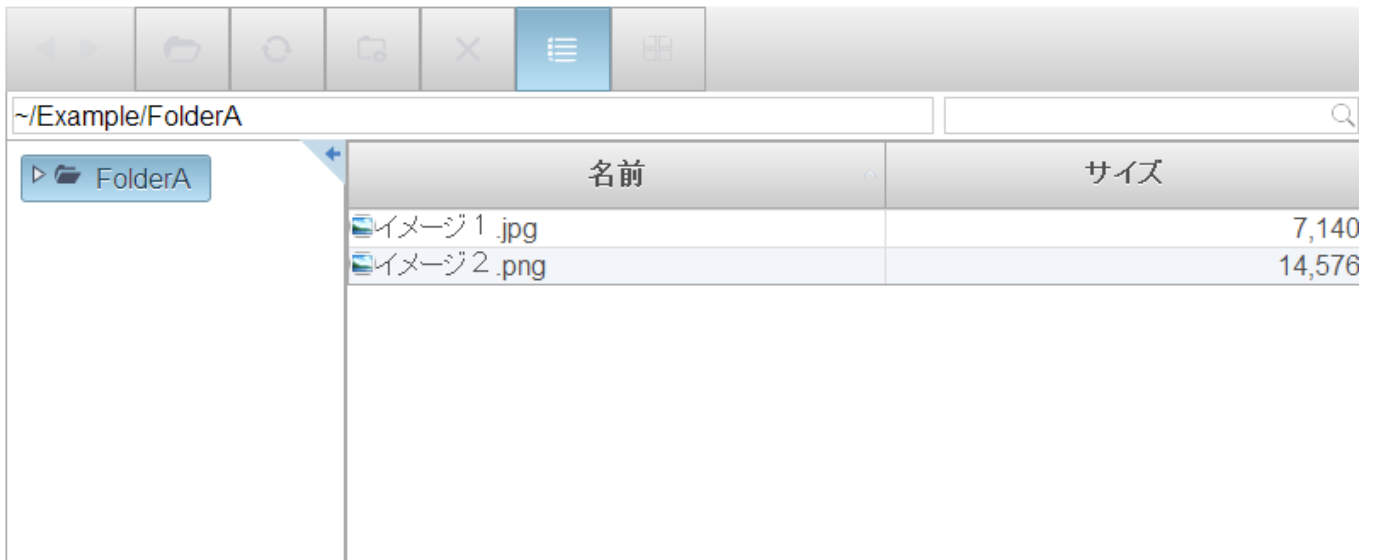
VB

```
C1FileExplorer1.InitPath= "Example\\A_Folder"
```

ここまでの成果

プロジェクトを実行すると、FileExplorer に Example\A_Folder の内容が表示されます。

FileExplorer for ASP.NET Web Forms



[先頭に戻る](#)

エクスプローラーの機能

複数ファイルの選択

AllowMultipleSelection プロパティを使用すると、ファイルブラウザ内での複数ファイルの選択を有効または無効にできます。デフォルトでは、**AllowMultipleSelection** は False に設定されています。

デザイナの場合

1. Visual Studio で FileExplorer を右クリックし、**[プロパティ]**を選択します。
2. **[プロパティ]**ウィンドウで、**AllowMultipleSelection** を True に設定して複数選択を有効にします。

ソースビューの場合

AllowMultipleSelection プロパティを `<cc1:C1FileExplorer>` タグ内で設定して、ファイルの複数選択を有効または無効にします。

```
<cc1:C1FileExplorer ID="C1FileExplorer1" runat="server" InitPath="~/Example"
AllowMultipleSelection="True">
```

コードの場合

次のコードを **Page_Load** イベントに追加して、複数ファイルの選択を有効または無効にします。

C# でコードを書く場合

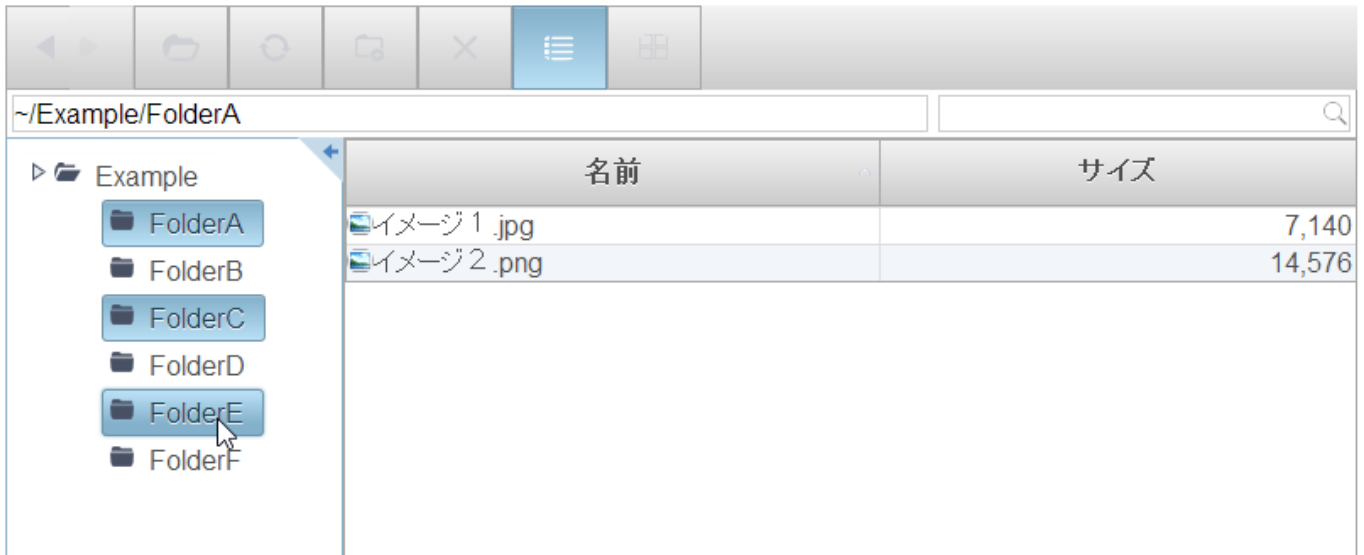
```
C#
C1FileExplorer1.AllowMultipleSelection = true;
```

Visual Basic でコードを書く場合

```
VB
C1FileExplorer1.AllowMultipleSelection = True
```

ここまでの成果

プロジェクトを実行し、**[Ctrl]**キーを押しながらファイル/フォルダをクリックすると、複数のファイル/フォルダを選択できます。

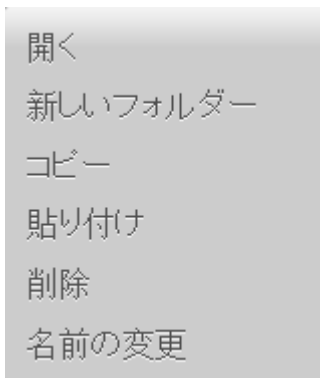


フォルダの作成と管理

FileExplorer を使用して、次のファイル操作を実行できます。


- 開く
- フォルダの新規作成
- コピー
- 貼り付け
- 名前の変更
- 削除

ファイルまたはフォルダを右クリックすると、次のようなコンテキストメニューが表示されます。



フォルダツリーまたはファイルブラウザでファイル操作を実行するには、次のようにします。

- **開く**: ファイルまたはフォルダを右クリックし、コンテキストメニューから**[開く]**を選択します。ツールバーの**[開く]**ボタンをクリックして、ファイルまたはフォルダを開くこともできます。
- **フォルダの新規作成**: フォルダを右クリックし、コンテキストメニューから**[新しいフォルダ]**を選択します。ツールバーの**[新しいフォルダ]**ボタンを使用して、選択されているフォルダ内に新しいフォルダを作成することもできます。
- **コピー**: 任意のファイルまたはフォルダを右クリックし、コンテキストメニューから**[コピー]**を選択します。
- **貼り付け**: 任意のファイルまたはフォルダを右クリックし、コンテキストメニューから**[貼り付け]**を選択します。
- **削除**: ファイルまたはフォルダを右クリックし、コンテキストメニューから**[削除]**を選択します。ツールバーの**[削除]**ボタンを使用して、選択されているファイルまたはフォルダを削除することもできます。
- **名前の変更**: 任意のファイルまたはフォルダを右クリックし、コンテキストメニューから**[名前の変更]**を選択します。

 ファイル操作を無効にする方法については、「[ファイル操作の無効化](#)」を参照してください。

ファイル操作の無効化

キーボードのユーザー補助機能

FileExplorer では、いくつかの基本的なファイル操作を実行できる組み込みのキーボードショートカットが提供されています。次の表に、デフォルトのキーボードショートカットを示します。

Shortcuts	C1.Web.Wijmo.Controls.C1FileExplorer.Shortcuts
Back	{Ctrl}+{K}
ContextMenu	{Shift}+{M}
Delete	{Delete}
FocusAddressBar	{Shift}+{2}
FocusFileExplorer	{Ctrl}+{F2}
FocusGrid	{Shift}+{4}
FocusGridPagingSlider	{Shift}+{5}
FocusToolBar	{Shift}+{1}
FocusTreeView	{Shift}+{3}
Forward	{Ctrl}+{L}
NewFolder	{Ctrl}+{N}
Open	{Enter}
PopupWindowClose	{Esc}
Refresh	{Ctrl}+{F3}
UploadFile	{Ctrl}+{U}

既存のショートカットを変更するには、次の手順に従います。

デザイナーの場合

1. Visual Studio で FileExplorer を右クリックし、**[プロパティ]**を選択します。
2. **[プロパティ]**ウィンドウで、**Shortcuts** プロパティグループを展開します。
3. Shortcuts.**Back** を '{Ctrl}+{B}' に、Shortcuts.**FocusFileExplorer** を '{Ctrl}+{K}' に設定します。同様に、他のショートカットも**[プロパティ]**ウィンドウで変更できます。

ソースビューの場合

次に示すように、ショートカットのマークアップを `<cc1:C1FileExplorer></cc1:C1FileExplorer>` タグ内に追加して、既存のショートカットを変更します。

```
<cc1:C1FileExplorer ID="C1FileExplorer1" runat="server" AccessKey="k">
  <Shortcuts Back="{Ctrl}+{b}" ContextMenu="{Shift}+{c}" FocusAddressBar="{Shift}+{a}" FocusGrid="{Shift}+{g}" FocusGridPagingSlider="{Shift}+{1}" FocusTreeView="{Shift}+{2}" Forward="{Ctrl}+{f}" NewFolder="{Ctrl}+{Shift}+{N}" FocusFileExplorer="{Ctrl}+{K}"/>
</cc1:C1FileExplorer>
```

コードの場合

次のコードを **Page_Load** イベントに追加して、**AccessKey** を追加し、既存のショートカットを変更します。

C# でコードを書く場合

```
C#
//AccessKey を追加します
C1FileExplorer1.AccessKey = "k";
```

FileExplorer for ASP.NET Web Forms

//既存のショートカットを変更します

```
C1FileExplorer1.Shortcuts.Back = "{Ctrl}+{B}";
C1FileExplorer1.Shortcuts.ContextMenu="{Shift}+{c}";
C1FileExplorer1.Shortcuts.FocusAddressBar="{Shift}+{a}";
C1FileExplorer1.Shortcuts.FocusGrid="{Shift}+{g}";
C1FileExplorer1.Shortcuts.FocusGridPagingSlider="{Shift}+{1}";
C1FileExplorer1.Shortcuts.FocusTreeView="{Shift}+{2}";
C1FileExplorer1.Shortcuts.Forward="{Ctrl}+{f}";
C1FileExplorer1.Shortcuts.NewFolder = "{Ctrl}+{Shift}+{N}";
C1FileExplorer1.Shortcuts.FocusFileExplorer="{Ctrl}+{K}";
```

Visual Basic でコードを書く場合

VB

'AccessKey を追加します

```
C1FileExplorer1.AccessKey = "k"
```

'既存のショートカットを変更します

```
C1FileExplorer1.Shortcuts.Back = "{Ctrl}+{B}"
C1FileExplorer1.Shortcuts.ContextMenu="{Shift}+{c}"
C1FileExplorer1.Shortcuts.FocusAddressBar="{Shift}+{a}"
C1FileExplorer1.Shortcuts.FocusGrid="{Shift}+{g}"
C1FileExplorer1.Shortcuts.FocusGridPagingSlider="{Shift}+{1}"
C1FileExplorer1.Shortcuts.FocusTreeView="{Shift}+{2}"
C1FileExplorer1.Shortcuts.Forward="{Ctrl}+{f}"
C1FileExplorer1.Shortcuts.NewFolder = "{Ctrl}+{Shift}+{N}"
C1FileExplorer1.Shortcuts.FocusFileExplorer="{Ctrl}+{K}"
```

ここまでの成果

プロジェクトを実行すると、操作に割り当てたキーボードショートカットを使用して、基本的な操作を実行できます。たとえば、**[Ctrl]+[B]**キーを押すと、ファイルブラウザ内を戻ることができます。

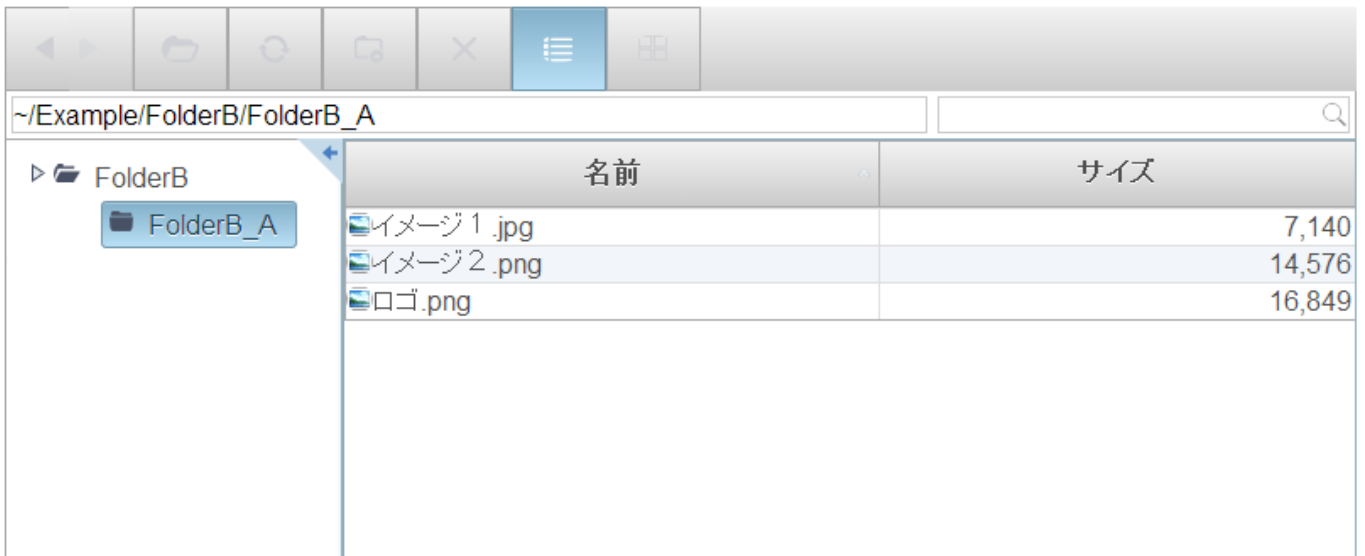
ソートとフィルタ処理

FileExplorer では、コードを記述することなく、ファイルやフォルダを名前に基づいてソートしたりフィルタ処理することができます。

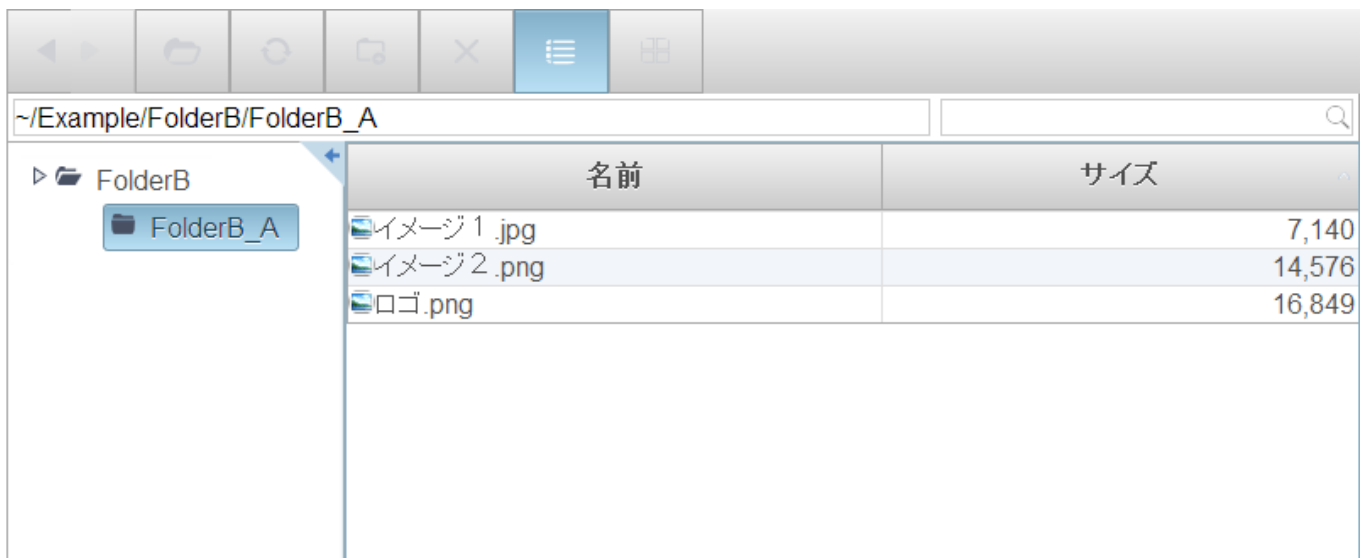
ソート

ソートするには、ファイルブラウザで列ヘッダーをクリックします。たとえば、ファイルやフォルダをアルファベット順にソートするには、**[名前]**ヘッダー行をクリックします。もう一度クリックすると、逆順にソートされます。

次の図は、ファイルブラウザでファイルとフォルダをアルファベット順にソートした例を示しています。

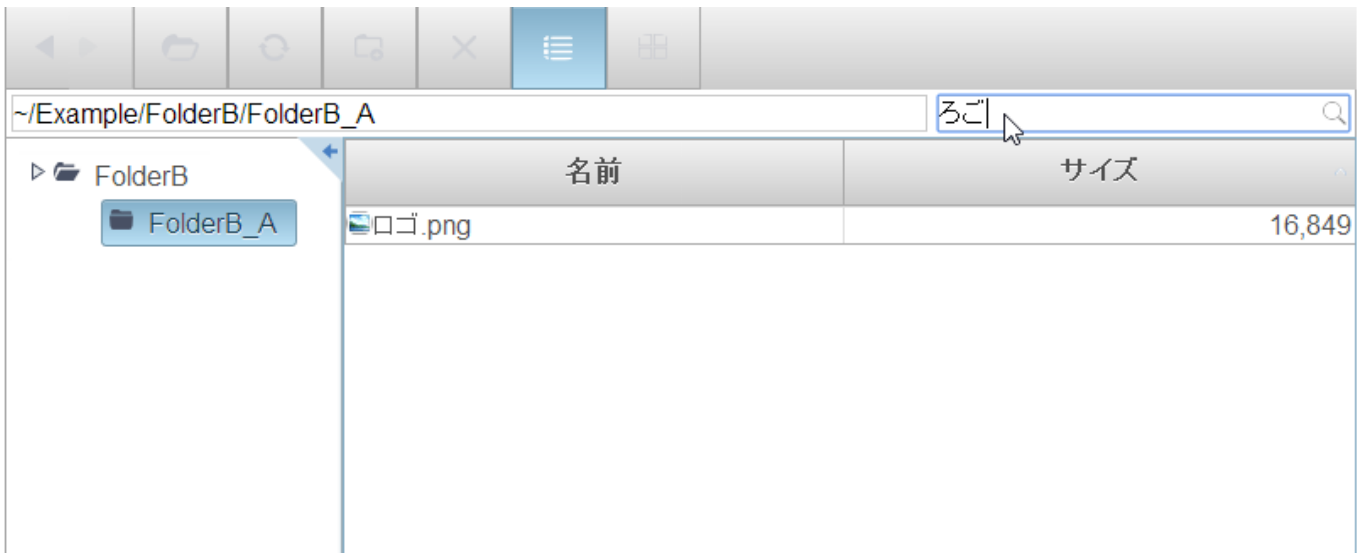


次の図は、右ペインでファイルとフォルダをサイズの昇順でソートした例を示しています。



フィルタ処理

FileExplorer の上部にあるフィルタテキストボックスを使用すると、ファイルを簡単に見つけることができます。ファイルをフィルタ処理するには、探したいファイルの名前をフィルタテキストボックスに入力します。次の図に示すように、入力されたアルファベットに基づいて FileExplorer がファイルを自動的に抽出します。



ページング

AllowPaging プロパティを使用して、FileExplorer 内でのページングを有効または無効にします。

デザイナーの場合

1. Visual Studio で FileExplorer を右クリックし、**[プロパティ]**を選択します。
2. **[プロパティ]**ウィンドウで、**AllowPaging** を True に設定して、ファイルやフォルダを複数ページに表示できるようにします。デフォルトでは、**AllowPaging** は False に設定されています。
3. **PageSize** プロパティを設定して、1ページあたりの項目の最大数を指定します。

ソースビューの場合

AllowPaging プロパティと **PageSize** プロパティを `<cc1:C1FileExplorer>` タグ内で設定して、ファイルやフォルダを複数ページに表示できるようにし、1ページあたりに表示する項目数を設定します。

```
<cc1:C1FileExplorer ID="C1FileExplorer1" runat="server" InitPath="~/Example"
AllowPaging="True" PageSize="5">
```

コードの場合

次のコードを **Page_Load** イベントに追加して、ページングを有効または無効にし、ページサイズを指定します。

C# でコードを書く場合

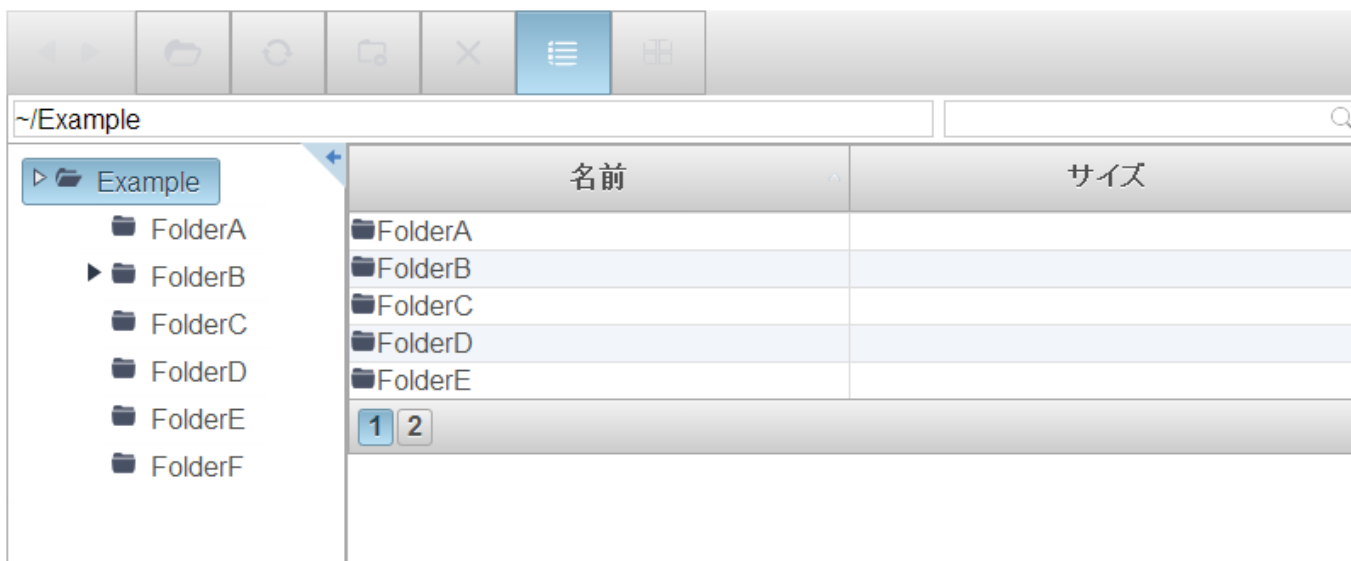
```
C#
C1FileExplorer1.AllowPaging = true;
C1FileExplorer1.PageSize = 5;
```

Visual Basic でコードを書く場合

```
VB
C1FileExplorer1.AllowPaging = True
C1FileExplorer1.PageSize = 5
```

ここまでの成果

プロジェクトを実行すると、ファイルやフォルダが1ページに5つだけ表示され、FileExplorer の下部にはページャが表示されます。



表示モード

FileExplorer コントロールおよび右ペインに表示されるファイルグリッドの表示モードを変更できます。

FileExplorer は2つのモードで表示できます。

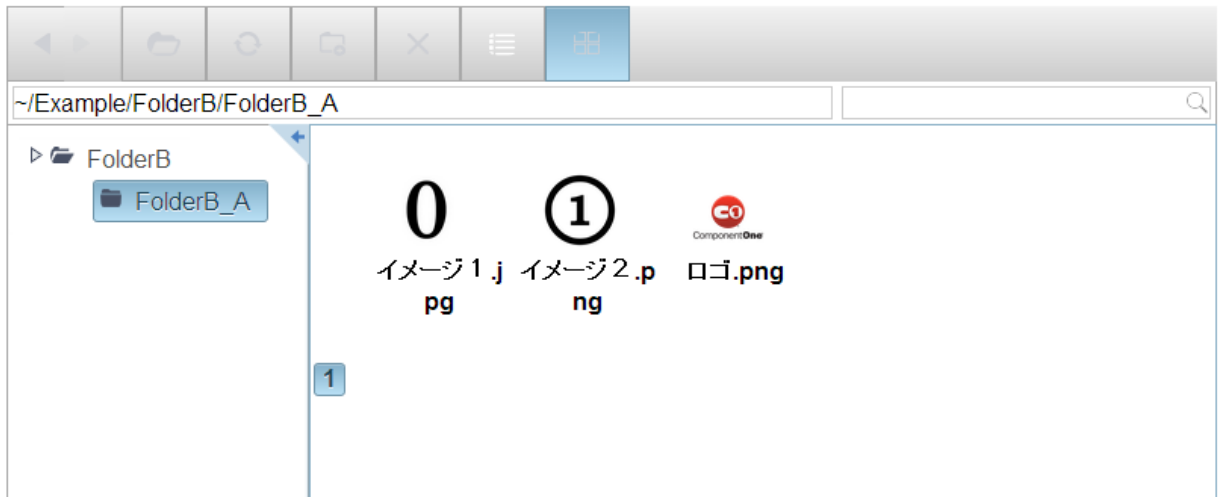
- Default:** このモードでは、左ペインにフォルダツリーが表示され、右ペインのファイルブラウザには、すべてのファイルとサブフォルダが表示されます。



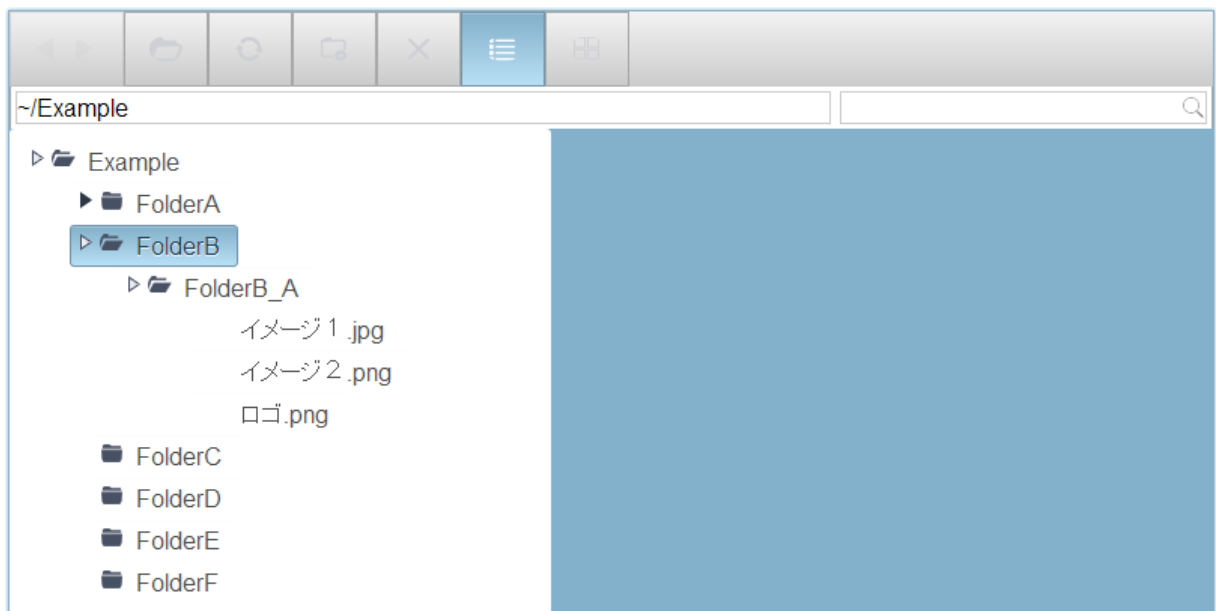
デフォルトモードでは、ファイルブラウザのビューモードを次のいずれかに変更することもできます。

- Detail:** ファイルブラウザで、ファイル/フォルダの名前とサイズの2つの列でファイルやフォルダを表示します。
- Thumbnail:** ファイルブラウザで、ファイルやフォルダをサムネイル画像として表示します。

FileExplorer for ASP.NET Web Forms




- **FileTree**: このモードでは、すべてのファイルとフォルダが階層ツリーの形式で1つのペインに表示されます。



FileExplorer の表示モードを変更するには、次の手順に従います。

デザインビューの場合

1. FileExplorer を右クリックし、**[プロパティ]**を選択します。
2. **[プロパティ]**ウインドウで、**Mode** プロパティを Default または TreeView に設定します。
3. **ViewMode** プロパティを Detail または Thumbnail に設定します。

 ファイルブラウザ内のファイルとフォルダの **ViewMode** プロパティは、**Mode** プロパティが Default に設定されている場合にのみ設定できます。

ソースビューの場合

Mode プロパティと **ViewMode** プロパティを <cc1:C1FileExplorer> タグ内で設定して、FileExplorer の表示を変更します。

```
<cc1:C1FileExplorer ID="C1FileExplorer1" runat="server" VisibleControls="All"
InitPath="~/Example" Mode="Default" ViewMode="Thumbnail">
```

コードの場合

次のコードを **Page_Load** イベントに追加して、FileExplorer の表示を変更します。

C# でコードを書く場合

C#

```
C1FileExplorer1.Mode = C1.Web.Wijmo.Controls.C1FileExplorer.ExplorerMode.Default;
C1FileExplorer1.ViewMode = C1.Web.Wijmo.Controls.C1FileExplorer.ViewMode.Detail;
```

Visual Basic でコードを書く場合

VB

```
C1FileExplorer1.Mode = C1.Web.Wijmo.Controls.C1FileExplorer.ExplorerMode.Default
C1FileExplorer1.ViewMode = C1.Web.Wijmo.Controls.C1FileExplorer.ViewMode.Detail
```

外観のカスタマイズ

Theme プロパティで組み込みのテーマを使用したり、**Themeroller** を使用して独自のテーマを作成して、FileExplorer の外観をカスタマイズできます。

Themeroller を使用した新しいテーマの作成方法については、「[Themeroller for Visual Studio](#)」を参照してください。

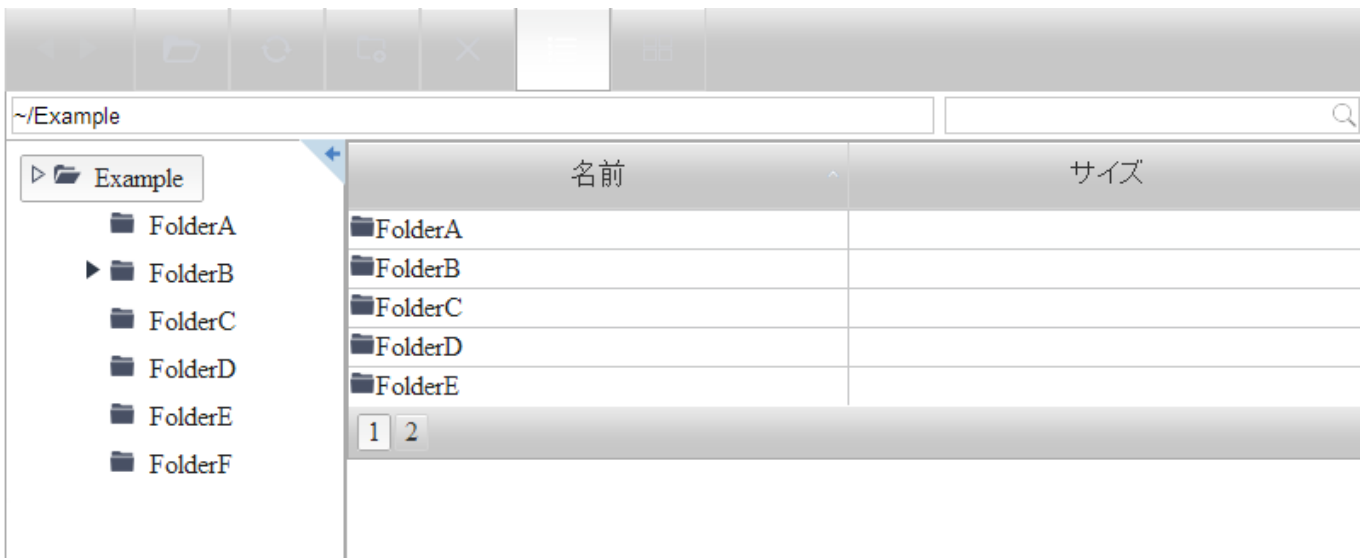
組み込まれているテーマ

FileExplorer for ASP.NET Web Forms には、次のテーマが組み込まれています。

- arctic
- aristo
- cobalt
- lucid
- metro-dark
- midnight
- rocket
- sterling

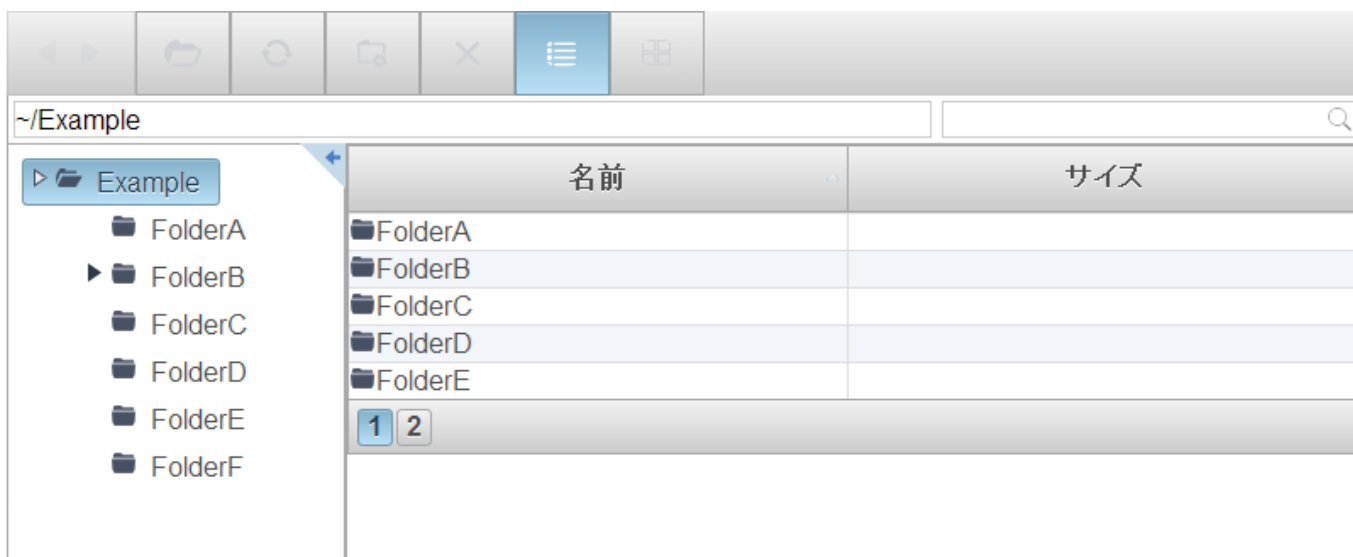
Arctic

次の図に、arctic テーマを示します。

**Aristo**

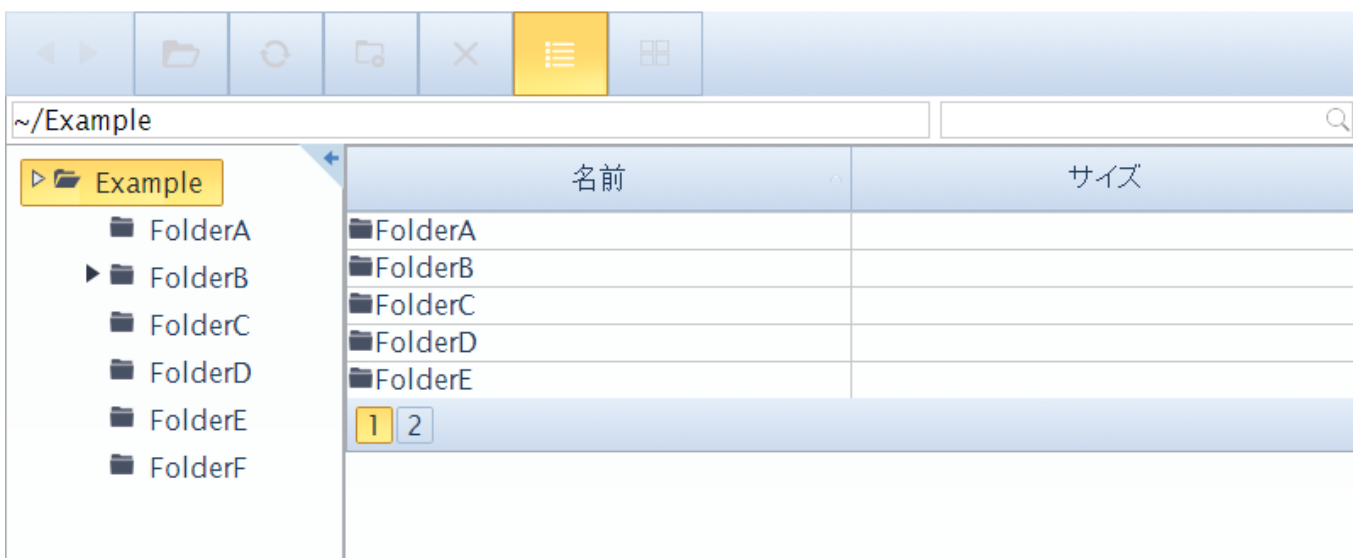
次の図に、aristo テーマを示します。

FileExplorer for ASP.NET Web Forms



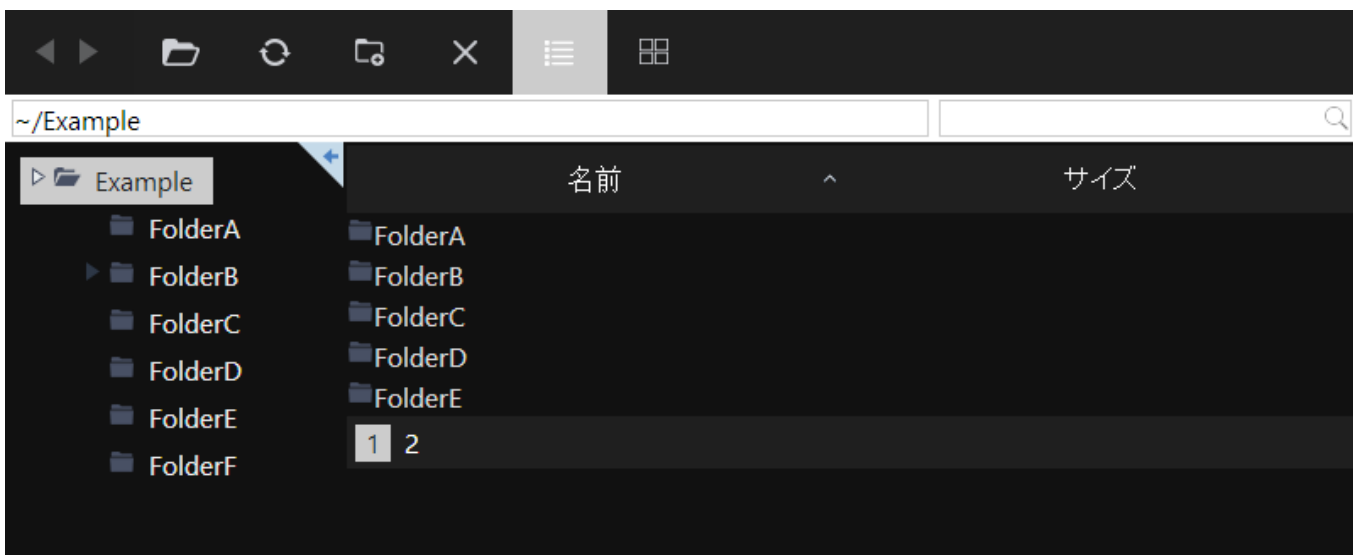
Cobalt

次の図に、cobalt テーマを示します。



Metro-Dark

次の図に、metro-dark テーマを示します。

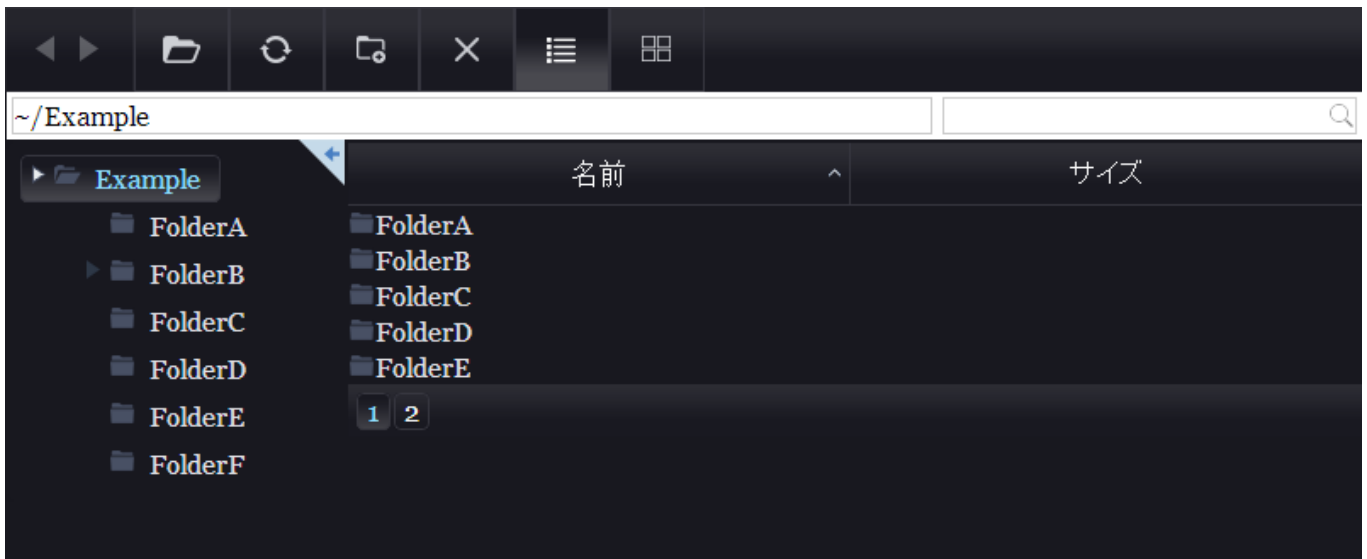


Metro

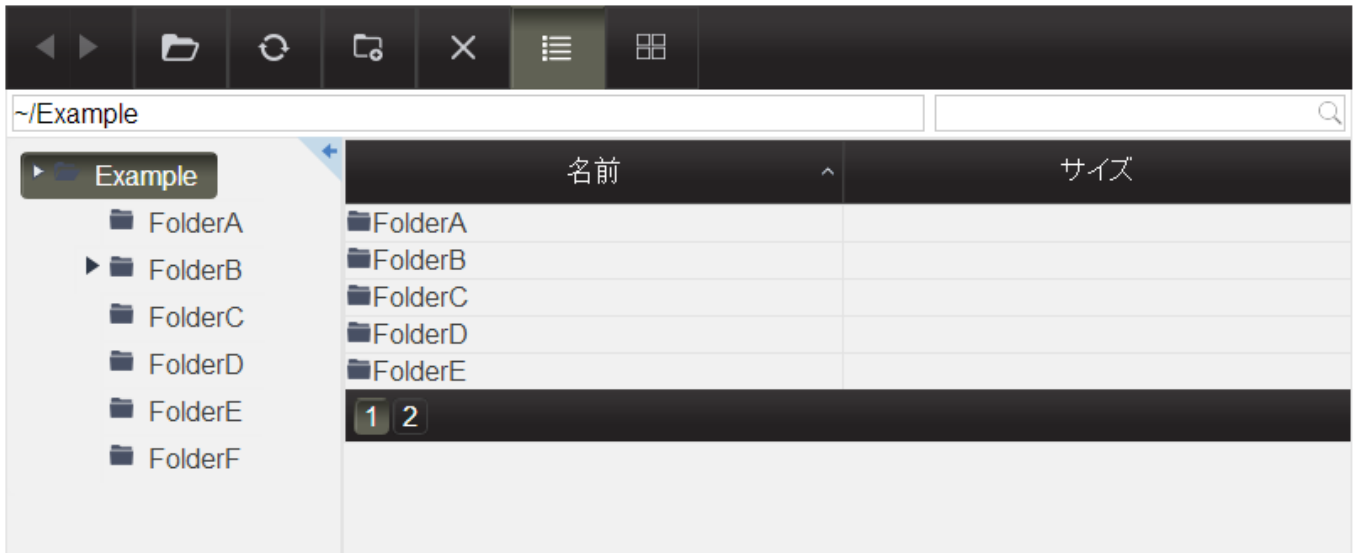
次の図に、metro テーマを示します。

**Midnight**

次の図に、midnight テーマを示します。

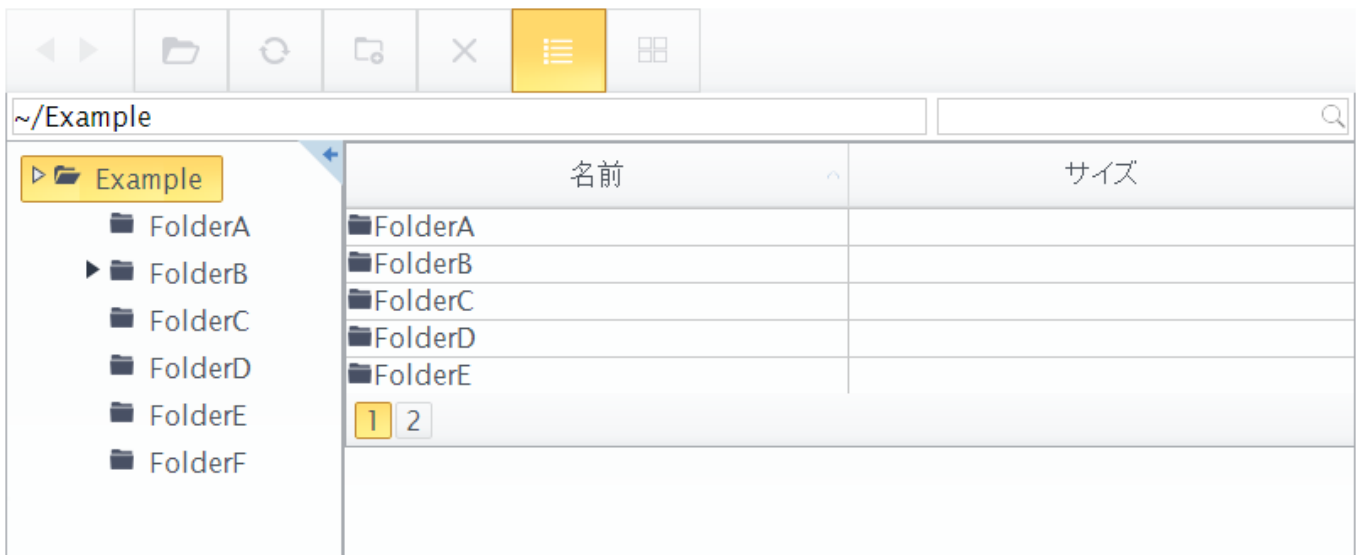
**Rocket**

次の図に、rocket テーマを示します。



Sterling

次の図に、sterling テーマを示します。



クライアントイベント

次の表に、FileExplorer で使用できるクライアント側イベントを示します。

イベント	説明
OnClientItemPasting	項目が貼り付けられる前に発生するイベント。
OnClientItemDeleting	項目が削除される前に発生するイベント。
OnClientFileOpened	ファイルが開かれた後に発生するイベント。
OnClientFiltered	フィルタ処理の後に発生するイベント。
OnClientFolderChanged	現在のアクティブフォルダが変更された後に発生するイベント。
OnClientFolderLoaded	フォルダコンテンツがロードされた後に発生するイベント。
OnClientItemSelected	ノードが選択された後に発生するイベント。
OnClientItemMoved	項目が移動された後に発生するイベント。

OnClientNewFolderCreated

新しいフォルダが作成された後に発生するイベント。