

## FlexViewクラス資料



## • はじめに

このドキュメントは、Biz/Browserでのアプリケーション構築を検討する際にポイントとなる項目をスムーズに理解していただく目的で作成されております。

このドキュメントが設計者、開発者の皆様のお役に立つことを心より願っております。

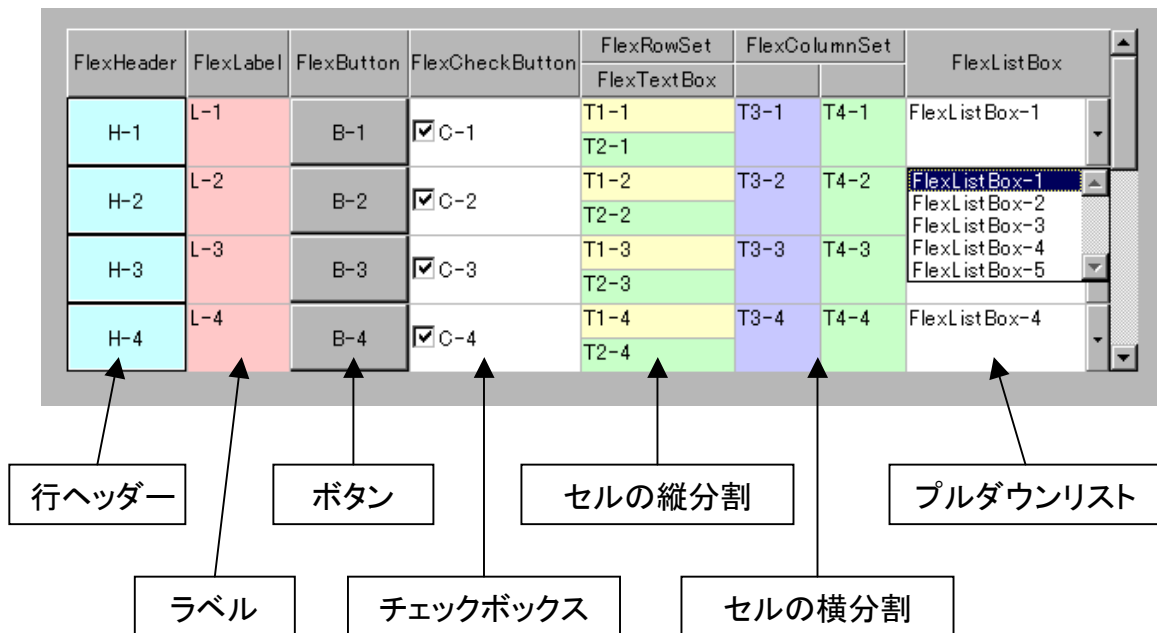
- ・本ドキュメントの所有権、知的財産権、その他全ての権利および権限は、アクシソフト株式会社が所有しています。
- ・本ドキュメントをアクシソフト株式会社の承諾を得ず、無断で複写、複製することを禁止します。
- ・本ドキュメントに記載されている内容は、将来予告なく変更されることがあります。
- ・Windowsは米国マイクロソフト社の登録商標です。
- ・その他製品名は各社の商標または登録商標です。



## 機能概要

FlexViewクラスは表形式のデータの表示および、入力を行うクラスです。任意のグルーピング表示、セルの分割、結合表示等の機能をもっています。また、各セルには専用のテキストボックス、ボタン、ラベル、プルダウンリストを直接配置する事が可能です。

FlexViewクラスは他の表形式のクラスとは異なり、個々のセルに対応するオブジェクトはありません。CSVDocumentクラスと同様に個々のセルはFlexViewの内部データとして管理され、メソッドを通してアクセスします。これによって、データ量が多い場合でも生成されるオブジェクト数を抑えることができるため、多くの行数を扱うときにSpreadやListViewよりも高いパフォーマンスが得られます。





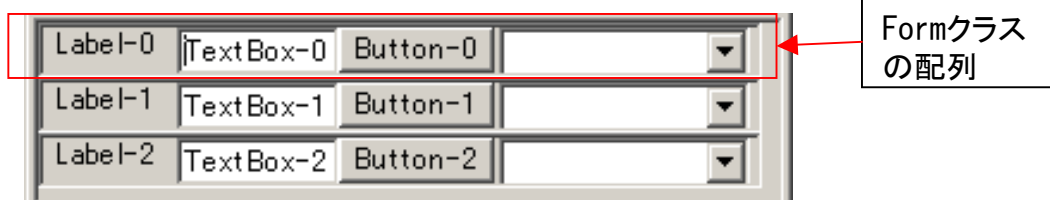
## • 一覧表示機能比較

Biz/Browserではデータを一覧形式で表現する方法は以下のようになります。

- 1、Formクラスの配列。
- 2、Spreadクラス。
- 3、FlexViewクラス。

### 1、Formクラスの配列の使用。

Formクラスを配列で定義し、Form上にその他のGUIクラスを配置することで、各行を定義します。



#### □メリット

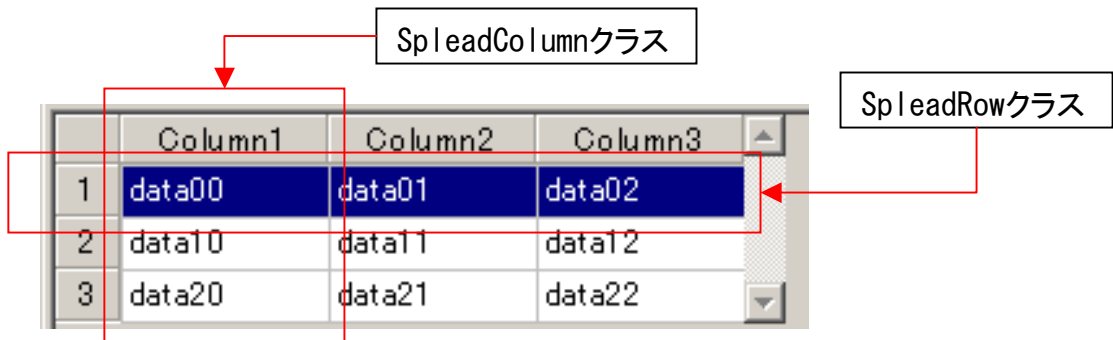
- ・Form上に配置可能なすべてのGUIクラスが使用可能。
- ・各GUIクラスの機能を完全に使用可能。
- ・一覧表示用クラスの機能上の制限をうけない。

#### □デメリット

- ・データ量に伴いオブジェクト数が増加するため、大量行データ表示時の画面表示のパフォーマンスが低下する。
- ・縦横スクロールの必要な場合の列ラベルの制御が困難。

### 2、Spreadクラスの使用。

Spreadクラスを配置し、クラスの機能を使用します。





□メリット

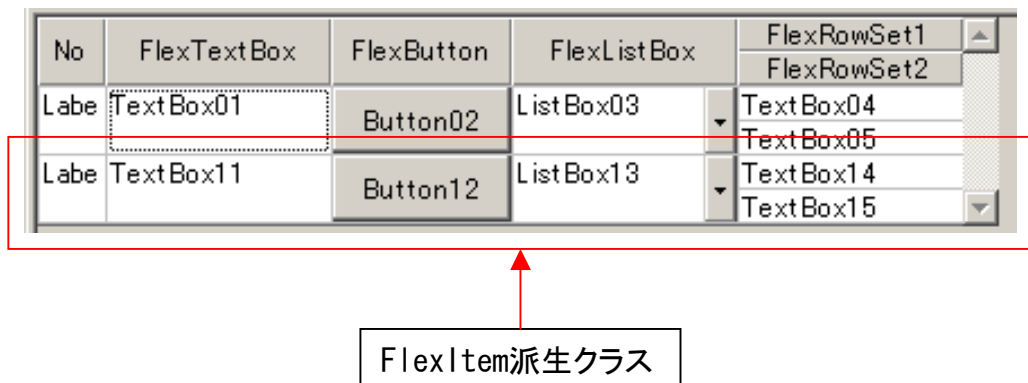
- ・画面作成時のレイアウトが容易。
- ・Spreadクラスの機能を利用することで、スクリプト実装が容易。
- ・クラスの機能である、ソート、列固定等の機能が利用可能。

□デメリット

- ・入力機能がない。(表示のみ)
- ・表示用のセル以外のオブジェクトは配置できない。
- ・データ量に伴いオブジェクト数が増加するため、大量行データ表示時の画面表示のパフォーマンスが低下する。

### 3、FlexViewクラスの使用。

FlexViewクラスを配置し、クラスの機能を使用する。



□メリット

- ・一覧形式での入力が可能。
- ・各セルに専用のFlexItem派生クラスを配置し多様な表現と機能が可能。
- ・大量データ表示時のパフォーマンスに優れている。

□デメリット

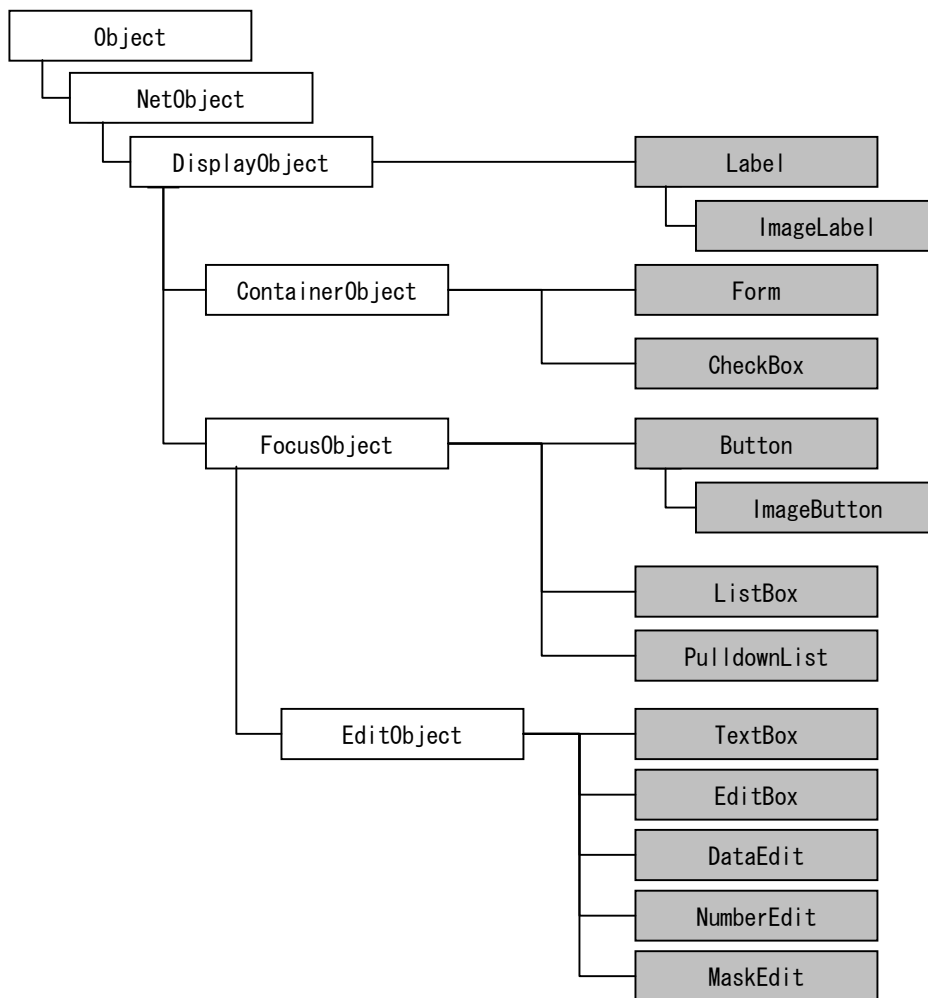
- ・画面作成時のレイアウトがやや複雑。
- ・各セルに配置されたオブジェクトに対するアクセスが独自形式となる。
- ・FlexItem派生クラスと一般的なGUIクラスとの機能差を理解する必要がある。



## • クラス派生関係

Biz/Browserではすべての要素はクラスで定義されています。また基本機能の定義は上位に位置する仮想クラスによって定義されていますので、各派生クラスの基本機能は各仮想クラスに依存しています。

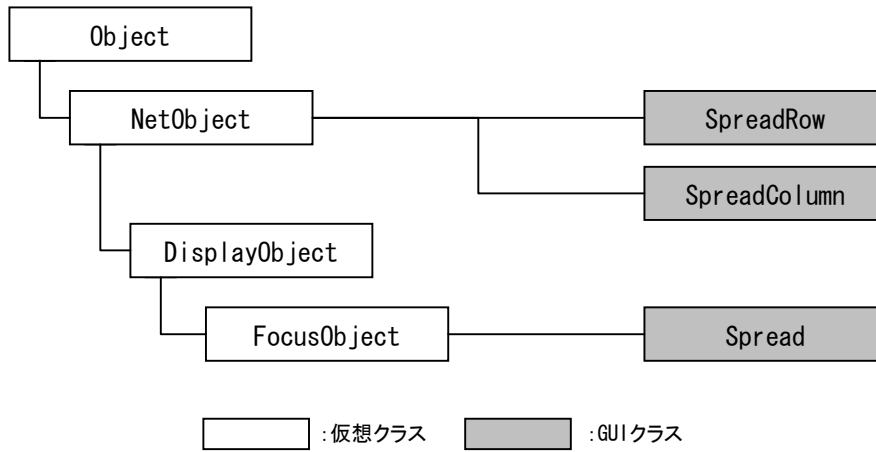
一般のGUIクラスの派生関係



□ :仮想クラス    ■ :GUIクラス

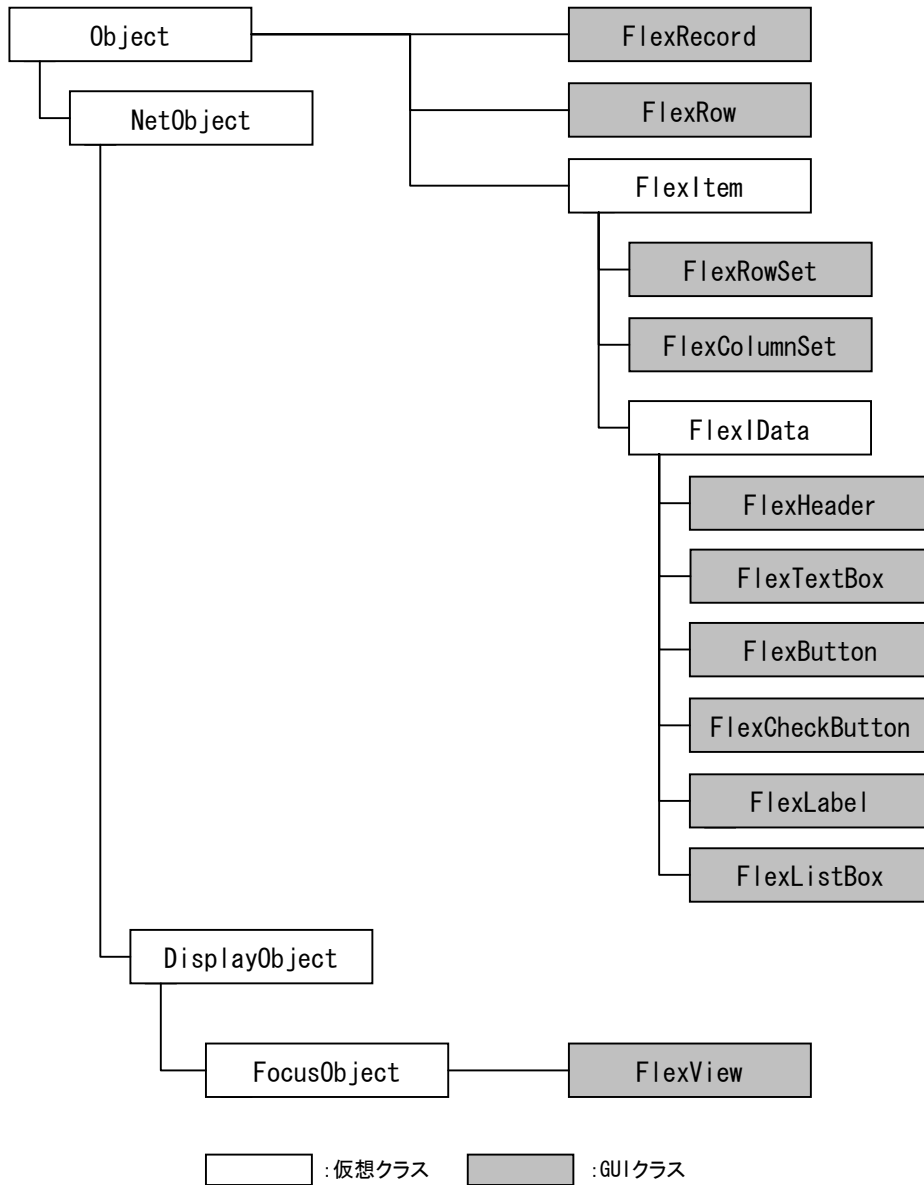


## Spreadクラスの派生関係





## FlexViewクラスの派生関係



FlexViewの各セルとして配置されるクラスはすべて、FlexItemクラスの派生クラスとして定義されています。一般のGUIのクラスとは基となる仮想クラスが異なります。このため一般のGUIクラスの機能と互換がない場合があります。

これはFlexViewクラスの特徴である、大量のデータ表示時の高いパフォーマンスを実現するために独自のクラス実装を行っているためです。





## • FlexItem派生クラスと通常のクラスとの差異

FlexViewクラスのセルとして配置される各FlexItem派生クラスは、テキストボックスやボタンといった通常のGUIクラスとほぼ同様の概観と機能をそなえています。しかし派生基となる基本クラスが違いため、機能上の差異が存在します。

以下の機能差異は代表的なものを抜粋したものであり、全ての差異を列挙したものではありません。ご注意ください。

### 1、FlexTextBoxクラス。

FlexTextBoxクラスは、FlexViewのセルの表示方法を定義するクラスで、対応するセルは、テキストボックスのように表示されます。

#### 1、TextBoxクラスとの差異。

TextBoxクラスは単一行の入力を行うクラスです。

#### 機能差異

	FlexTextBox	TextBox
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Valueの確定タイミング	編集状態が解除されたとき。	キーボードフォーカスを失ったときまたは、Enterキー押したとき。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
書式の反映するタイミング	編集状態が解除されたとき。	オブジェクトからカーソルが脱出したとき。
カーソル移動の代替キー設定	個別に設定できません。上位のFlexViewクラスのCursorMoveプロパティにて一括して指定されます。	NextTabKeyプロパティで個別に設定可能。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。
TextChanged (イベント)	セルの編集操作で発生します。	実装されていません。



## 2、EditBoxクラスとの差異。

複数行の文字を入力可能なクラスです。

### 機能差異

	FlexTextBox	EditBox
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Valueの確定タイミング	編集状態が解除されたとき。	キーボードフォーカスを失ったときまたは、Enterキー押したとき。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
書式の反映するタイミング	編集状態が解除されたとき。	オブジェクトからカーソルが脱出したとき。
複数行入力	単一行のみの入力です。	複数行入力が可能。
ワードラップ	ワードラップは機能はありません。	Wrapプロパティが有効で、横スクロールバーが表示されて内場合ワードラップが行われます。
カーソル移動の代替キー設定	個別に設定できません。上位のFlexViewクラスのCursorMoveプロパティにて一括して指定されます。	NextTabKeyプロパティで個別に設定可能。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。
TextChanged (イベント)	セルの編集操作で発生します。	実装されていません。



### 3、NumberEditクラスとの差異。 数値入力専用のTextBoxです。

#### 機能差異

	FlexTextBox	NumberEdit
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Valueの確定タイミング	編集状態が解除されたとき。	キーボードフォーカスを失ったときまたは、Enterキー押したとき。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
書式の反映するタイミング	編集状態が解除されたとき。	入力途中で表示に反映されます。
カーソル移動の代替キー設定	個別に設定できません。上位のFlexViewクラスのCursorMoveプロパティにて一括して指定されます。	NextTabKeyプロパティで個別に設定可能。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。
TextChanged (イベント)	セルの編集操作で発生します。	実装されていません。
isNull (プロパティ)	実装されていません。	ValueがNULLかどうかを調べる機能です。
簡易電卓機能	実装されていません。	使用可能。



#### 4、DateEditクラスとの差異。 日付入力専用のTextBoxです。

##### 機能差異

	FlexTextBox	DataEdit
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Valueの確定タイミング	編集状態が解除されたとき。	キーボードフォーカスを失ったときまたは、Enterキー押したとき。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
書式の反映するタイミング	編集状態が解除されたとき。	入力途中で表示に反映されます。
カーソル移動の代替キー設定	個別に設定できません。上位のFlexViewクラスのCursorMoveプロパティにて一括して指定されます。	NextTabKeyプロパティで個別に設定可能。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。
TextChanged (イベント)	セルの編集操作で発生します。	実装されていません。
InvalidDate (プロパティ)	実装されていません。	不正日付の確認をする機能です。
カレンダー機能	実装されていません。	使用可能。



## 5、MaskEditクラスとの差異。

MaskEditクラスは、Maskプロパティの指定により、入力形式や出力形式をカスタマイズ可能な固定長の文字列を入力するためのテキストボックスです。

### 機能差異

	FlexTextBox	MaskEdit
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Valueの確定タイミング	編集状態が解除されたとき。	キーボードフォーカスを失ったときまたは、Enterキー押したとき。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
カーソル移動の代替キー設定	個別に設定できません。上位のFlexViewクラスのCursorMoveプロパティにて一括して指定されます。	NextTabKeyプロパティで個別に設定可能。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。
TextChanged (イベント)	セルの編集操作で発生します。	実装されていません。
Mask (プロパティ)	実装されていません。	マスク文字列の設定。
InvalidText (プロパティ)	実装されていません。	不正な入力の有無を確認をする機能です。



## 2、FlexLabelクラス。

FlexLabelクラスは、FlexViewのセルの表示方法を定義するクラスで、対応するセルは、ラベルのように表示されます。

### 1、Labelクラスとの差異

文字列を表示するためのクラスです。

#### 機能差異

	FlexLabel	Label
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Fontの指定	個別に指定できません。 上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
Transparent (プロパティ)	実装されていません。	背景を透過します。



## 2、ImageLabelクラスとの差異。

文字とイメージを表示するクラスです。表面にはJPEGまたはPNGイメージと文字が表示されます。

### 機能差異

	FlexLabel	ImageLabel
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
イメージの表示	アイコンの表示を行います。表示するアイコンが行の高さよりも大きい場合、縦横比を維持したまま縮小されます。	アイコンとイメージの表示が可能です。Resizeプロパティにより、オリジナルサイズでの表示が可能です。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。



### 3、FlexButtonクラス。

FlexButtonクラスは、FlexViewのセルの表示方法を定義するクラスで、対応するセルは、ボタンのように表示されます。

#### 1、Buttonクラスとの差異。

押す事ができるボタンを表示するクラスです。

#### 機能差異

	FlexButton	Button
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Fontの指定	個別に指定できません。 上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
押下動作のイベント	Pushイベント。	Touchイベント。
押下代替キー設定	実装されていません。	AltKeyプロパティにて設定可能。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。





## 2、ImageButtonクラスとの差異。

押す事ができるボタンにイメージを表示できるクラスです。表面にはJPEGまたはPNGイメージとタイトルが表示されます。

### 機能差異

	FlexButton	ImageButton
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
イメージの表示	実装されていません。	アイコンとイメージの表示が可能です。 Resizeプロパティにより、オリジナルサイズでの表示が可能です。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
BgColor FgColor (プロパティ)	実装されていません。	使用可能。
トグルボタン 機能	実装されていません。	ButtonModeプロパティにより設定可能。



#### 4、FlexListBoxクラス

FlexListBoxクラスは、FlexViewのセルの表示方法を定義するクラスで、対応するセルは、プルダウンリストのように表示されます。

選択リストはListDataプロパティに設定されます。選択肢は、カンマ区切りの文字列で指定します。改行やタブを含むことはできません。

##### 1、PullDownListクラスとの差異。

プルダウンリストを表示するクラスです。PullDownListクラスは下位にPullDownListItemクラスを配列として動作します。選択肢はPullDownListItemクラスの配列要素として管理されます。

	FlexListBox	PullDownList
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
リスト選択イベント	SelectChangedイベント。	PullDownListItemのTouchイベント。
選択位置の確認	Valueプロパティには選択肢の値が格納されています。	PullDownListのValueプロパティに選択されたPullDownListItemの要素番号が格納されています。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。



## 2、ListBoxクラスとの差異

リストボックスを表示するクラスです。ListBoxクラスは下位にListItemクラスを配列として動作します。選択肢はListItemクラスの配列要素として管理されます。

### 機能差異

	FlexListBox	ListBox
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
リスト選択イベント	SelectChangedイベント。	ListItemのTouchイベント。
選択位置の確認	Valueプロパティには選択肢の値が格納されています。	ListBoxのValueプロパティに選択されたListItemの要素番号が格納されています。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。



### 3、ComboBoxクラスとの差異

コンボボックスを表示するクラスです。ComboBoxクラスは下位にComboBoxItemクラスを配列として動作します。選択肢はComboBoxItemクラスの配列要素として管理されます。

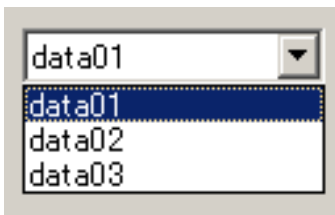
	FlexListBox	ComboBox
配置	FlexRecordオブジェクト、FlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置します。	ContainerObjectクラスの下位に配置します。
Fontの指定	個別に指定できません。上位のFlexViewクラスにて一括して指定されます。	個別に指定可能。
リスト選択イベント	SelectChangedイベント。	ComboBoxのChangeイベント。
入力機能	入力はできません。	表示部への直接入力が可能です。
Get (メソッド)	実装されていません。	使用可能。 WEBサーバからCRSファイルをダウンロードして実行する機能です。
SetFocus (メソッド)	実装されていません。	使用可能。
GetFocus LostFocus (イベント)	実装されていません。	使用可能。
DropdownWidth (プロパティ)	実装されていません。	ドロップダウンリスト部の幅を個別に設定可能です。



#### 4、通常の複合オブジェクトとの選択肢設定の違い

PullDownListクラスなどの複合構造を持ったクラスは選択肢を下位のクラスの配列として管理しています。これに対してFlexListBoxの場合はListDataプロパティにカンマで区切った文字列として設定します。

PullDownListクラスの選択肢の設定。

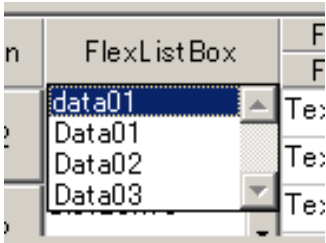


`PullDownList1[0]. Value = "data01"`

`PullDownList1[1]. Value = "data02"`

`PullDownList1[2]. Value = "data03"`

FlexListBoxの選択肢の設定。



`FlexListBox. ListData = "data01, data02, data03"`



## • FlexViewクラス使用時の留意点

### 1、オブジェクトへのアクセス方法。

FlexViewクラスのセルとして配置される、FlexItemクラスからの派生クラスは通常のクラスと異なり、直接オブジェクトを参照する事ができません。Valueプロパティ等の各プロパティを参照、設定する場合は必ずFlexRowオブジェクトを取得しその下位にある各セル毎のアクセッサクラスを通じてアクセスします。

#### 1、Spreadクラスのオブジェクトに対するアクセス。

1行目、1列目のオブジェクトのValueを参照する場合

	Column1	Column2	Column3
1	data00	data01	data02
2	data10	data11	data12
3	data20	data21	data22

```
var data = Spread1.SpreadRow[0].SpreadColumn1.Value;
```

#### 2、FlexTextBoxクラスのオブジェクトに対するアクセス。

1行目のFlexTextBoxオブジェクトのValueを参照する場合

No	FlexTextBox	FlexButton	FlexListBox	FlexRowSet1
Label	123,456	Button02	data01	FlexRowSet2
Label	321,654	Button12	List Box13	Text Box04
				Text Box05
				Text Box14
				Text Box15

誤)

```
var data = FlexView1.FlexRecord1[0].FlexTextBox1.Value;
```

正)

```
var row = FlexView1.GetRow(0);  
var data = row.FlexTextBox1.Value;
```

FlexRowオブジェクト

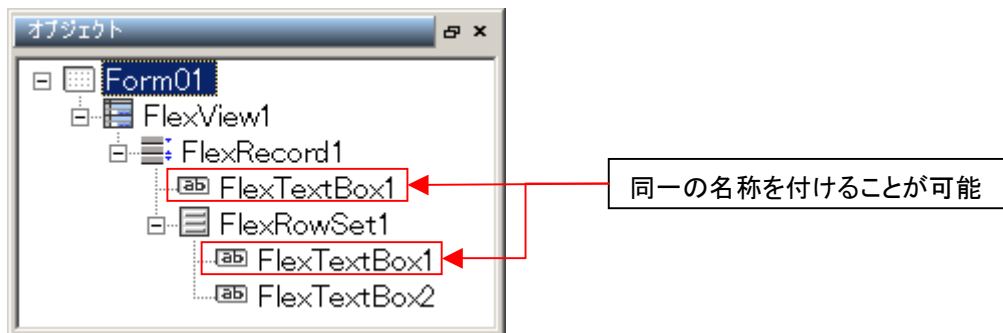


## 2、オブジェクトへの命名時の注意点。

FlexItemクラスからの派生クラスはFlexColumnSetオブジェクトまたは、FlexRowSetオブジェクトの下位に配置が可能です。この場合、オブジェクト階層としては他のオブジェクトの下位に位置するため同一のFlexView内で重複した命名が可能です。

しかし、各オブジェクトのデータにアクセスする場合FlexColumnSetオブジェクトとFlexRowSetオブジェクトはFlexRowオブジェクト上では構造に関連しません。このため重複したオブジェクト名を持つオブジェクトに正常にアクセスできなくなります。

同一のFlexViewクラスに配置されるFlexItemクラスからの派生クラスは名称をユニークに命名してください。



データへのアクセスは以下の様に行いますがFlexRowオブジェクト以下では同一階層上と扱われますので片方のオブジェクトにしかアクセスできません。

```
var row = FlexView.GetRow(0);  
Var data = row.FlexTextBox1.Value;
```

