

# TECH-BEE dII マニュアル

第6版

テロップ  
ドロップファイルリスト  
優先選択コンボボックス  
ラベル  
テキストボックス  
カラフルキューブ  
パスワードポリシーチェック

## TECH-BEE

ホームページ <http://techbee.web.fc2.com/>

ご意見ご要望 <http://techbee.blog14.fc2.com/?all>

メールアドレス [tech-bee@mail.goo.ne.jp](mailto:tech-bee@mail.goo.ne.jp)

# 改定履歴

版	公開日	摘要
第6版	2011/05/15	パスワードポリシーチェック(1.0.0) 追加 本文書名称変更 (カスタムコントローラマニュアル → dll マニュアル)
第5版	2011/04/28	カラフルキューブ(2.0.0) 追加
第4版	2011/04/25	テキストボックス(1.1.0) 追加
第3版	2011/04/16	ラベル(1.0.0) 追加
第2版	2011/04/03	優先選択コンボボックス(1.0.0) 追加
第1版	2011/03/26	初版公開      テロップ(1.1.0)・ドロップファイルリスト(1.0.0) 追加

# 目次

---

カスタムコントロールの準備	1
dll の準備	2
優先選択コンボボックス (cctFrequencyComboBox)	3
ラベル (cctLabel)	5
テロップ (cctTelop)	6
テキストボックス (cctTextBox)	7
パスワードポリシーチェック (clsPassword)	8
ドロップファイルリスト (DropFileList)	10
バイナリーキューブ (uctBinaryCube)	12
カラフルキューブ (uctColorfulCube)	13
クルクルキューブ (uctCurucuruCube)	14

# カスタムコントロールの準備

## はじめに

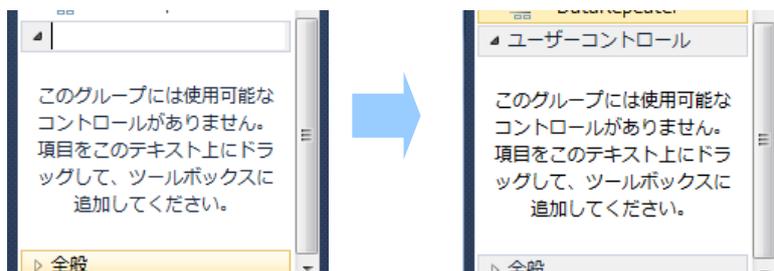
カスタムコントロールとは、.NET Framework クラス ライブラリに属さないコントロールで、ユーザー コントロールを含めた総称です。  
ASP.NET で使用する カスタム サーバー コントロールと、Windows Form アプリケーションで使用する カスタム クライアント コントロールがあります。  
また、一般的には標準コントロールの機能を拡張したものをカスタムコントロール、ユーザーコントロール用のベースに単体もしくは複数のコントロールを配置し意味のある一連の機能をまとめたものをユーザーコントロールと呼び、区別します。

## 登録方法

VB.net, C# などの開発環境でフォームもしくはユーザーコントロールのデザイン画面を開きます。

### ユーザーコントロールを登録するタブの追加

デフォルトでは画面の左側に配置されているツールボックスの中で右クリックをし、タブの追加をクリックします。  
タブ名の入力を促されますので、ユーザーコントロールなど分かりやすい名称を指定してください



### ユーザーコントロールの追加

エクスプローラー上から任意のコントロールの dll をドラッグして、上記にて作成したタブにドロップすることにより追加することが出来ます。



※ 利用環境が適切でない場合ドラッグアンドドロップしてもタブ内に表示されないことがあります。  
例) コントロールが .netFrameWork4.0 で開発されていて、開発環境が 2.0 となっているなど

## 使用方法

標準のコントロールと同じようにフォームやユーザーコントロールに貼り付けて使用してください  
詳細は個々のコントロールのマニュアルをご確認ください

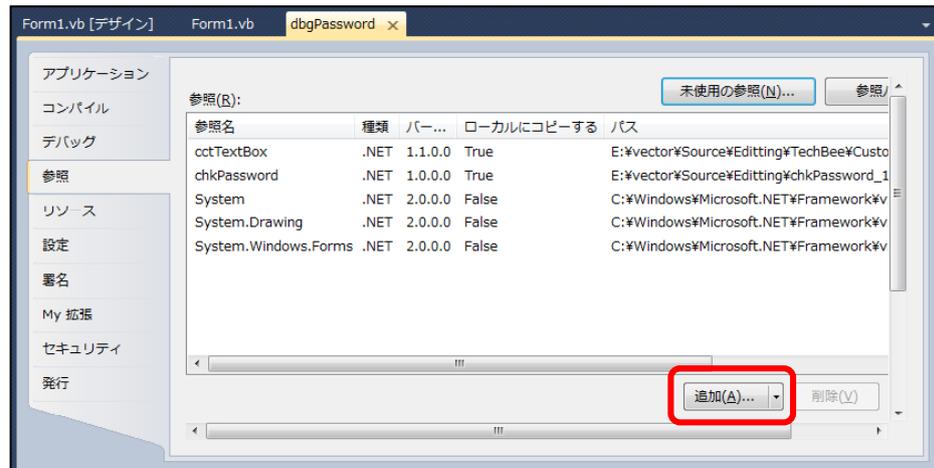
# dll の準備

## はじめに

本来カスタムコントロールも dll ですが、本文書ではカスタムコントロールを除く dll の事を dll と呼びます。

## 登録方法

VB.net, C# などの開発環境で追加したい dll の参照を追加します。



## 使用方法

オブジェクトを生成した上で利用してください  
詳細は個々のコントロールのマニュアルをご確認ください

**概要** Windows Form 上で、直近の選択項目および選択頻度が高い項目を優先的に表示するコンボボックスです。  
 選択項目にない文言を随時追加することが出来ます。  
 管理用のデータベースとして SQLite のデータベースファイルを利用しています。

**分類** カスタムコントロール

**ファイル名** cctFrequencyComboBox.dll

**参照ファイル**  
 conSQLite.dll データベース接続用 dll  
 System.Data.SQLite.dll SQLite 管理用 dll  
 ComboDataFile.db データベースファイル

**準備** カスタムコントロールの準備方法を参照してください

**利用方法** フォーム上に貼り付けてください

**サブプロシージャ**

名称	引数	摘要
gsub_SelectedItem	なし	選択した項目を最新としてデータベースに登録します。

**ファンクション**

名称	引数	型	摘要
gfnc_blnGetData	なし	Boolean	データベースに登録された内容から直近に選択されたもの、選択頻度が高いものの順に並べ替えられたデータをコンボボックスにセットしま

**プロパティ**

名称	Read	Write	型	摘要
DatabaseFile		<input type="radio"/>	String	SQLite のデータベースファイルを絶対パスもしくは相対パスで指定します。
Password		<input type="radio"/>	String	SQLite のデータベースファイルに設定したパスワードを指定します。設定されていない場合は指定不要です。
TableName		<input type="radio"/>	String	SQLite のデータベースファイル中のテーブル名を指定します。
MoreRecentlyCNT		<input type="radio"/>	Integer	選択候補の先頭に表示する直近に指定した項目の件数を指定します。直近から順番に並べます。
RecentlyCNT		<input type="radio"/>	Integer	リスト中の全ファイル名を取得します。引数に True を与えると選択データのみを返します。

**概念**

右のようなデータがデータベースに登録されており、MoreRecentlyCNT が 2 RecentlyCNT が 4 であった場合、表示順は赤丸の順番になります。  
 それぞれ 3 と 3 だった場合表示順は青丸の順番になります。

項目名	選択年月日時分秒	選択回数
項目 1	20100101_011010	7
項目 2	20100101_011011	6
項目 3	20100101_011012	5
項目 4	20100101_011013	4
項目 5	20110101_011014	3
項目 6	20110101_011015	2
項目 7	20100101_011016	1

MoreRecentlyCNT	2	一番新しい 二番目に新しい
RecentlyCNT	3	三番目から五番目に新しい中で一番選択回数が多い 三番目から五番目に新しい中で二番目に選択回数が多い 三番目から五番目に新しい中で三番目に選択回数が多い 六番目以降の中で一番選択回数が多い 六番目以降の中で二番目に選択回数が多い

MoreRecentlyCNT と RecentlyCNT を同じ値にすると、RecentlyCNT の指定は無視されます。

## 利用例

```
Public Class Form1

    Private Sub Form1_Load(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles MyBase.Load
        CctFrequencyComboBox1.DatabaseFile = ".\ComboDataFile.db"
        CctFrequencyComboBox1.TableName = "ComboData"
        CctFrequencyComboBox1.gfnc_blnGetData()
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles Button1.Click
        CctFrequencyComboBox1.gsub_SelectedItem()
        CctFrequencyComboBox1.gfnc_blnGetData()
    End Sub

End Class
```

## その他

添付のデータベースファイルを利用させていただくことにより本 dll を利用していただけますが、次のSQLにより、SQLite の任意のデータベースファイルに管理テーブルを登録していただくことができます。

```
CREATE TABLE [ComboData] (
    [ItemData] TEXT NOT NULL UNIQUE,
    [SelectDate] TEXT,
    [SelectCount] INTEGER,
    PRIMARY KEY (ItemData)
)
```

※ テーブル名は任意のものに変えて使用してください

## 概要

標準の機能を拡張したラベルです。

Caption に指定した文字列を、デザイン上のコントロールの幅で均等割付します。

## 分類

ユーザーコントロール

## ファイル名

cctLabel.dll

## 準備

カスタムコントロールの準備方法を参照してください

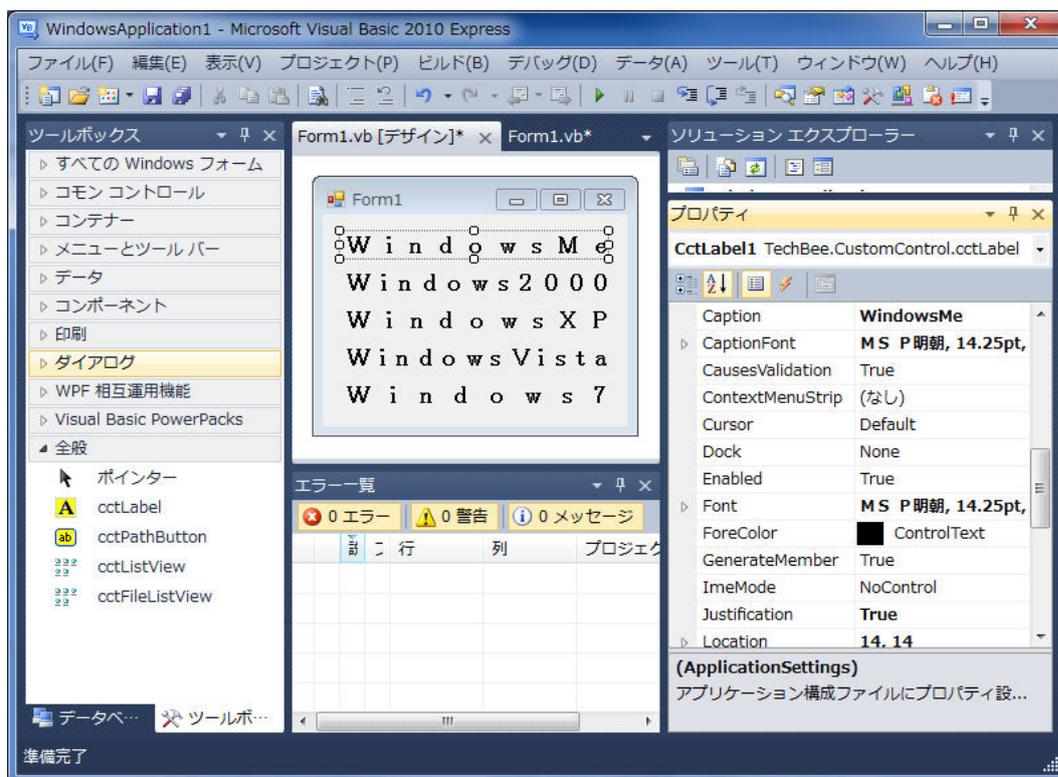
## 利用方法

フォーム上に貼り付けてください

## プロパティ

名称	Read	Write	型	摘要
Caption	<input type="radio"/>	<input type="radio"/>	String	コントロール上に表示する文字列を指定します。
CaptionFont	<input type="radio"/>	<input type="radio"/>	Font	一般的なコントロールで指定する Font と同様のものです。プロポーショナルフォントを指定していただくことも出来ます。
Justification	<input type="radio"/>	<input type="radio"/>	Boolean	均等割付をするかどうかを指定します。 True : 均等割付をする False : 均等割付をしない 今後機能拡張をした際に、別の機能だけを使うこともあると思われるために実装してあります。 False にすればほぼ通常の Label としてご利用いただけます

## 利用例



プロパティの設定だけで設定をすることが出来ます。  
フォントは、プロポーショナルフォントやボールドも利用できます。

**概要** Windows Form 上で任意の文字列をスクロールして表示します。  
同時に複数の文字列をセットすることが出来、順番に表示します。  
実行途中での追加・割り込みができます。  
ひとつの文字列ごとに表示終了時にイベントを発生させることができます。

**分類** ユーザーコントロール

**ファイル名** cctTelop.dll

**準備** カスタムコントロールの準備方法を参照してください

**利用方法** フォーム上に貼り付けてください

**プロパティ**

名称	Read	Write	型	摘要
Caption	<input type="radio"/>	<input type="radio"/>	String	テロップで表示する文字列を指定します。 設定と共に表示しなおします。
AddCaption		<input type="radio"/>	String	テロップで表示する文字列を追加します。
AddCaptions		<input type="radio"/>	String()	テロップで表示する文字列を配列で追加します。
InsCaption		<input type="radio"/>	String	実行時に現在表示している文字列と次の文字列の間に文字列を割りこませることができます。
CaptionColor	<input type="radio"/>	<input type="radio"/>	Color	文字色を指定します。
BackColor	<input type="radio"/>	<input type="radio"/>	Color	背景色を指定します。
Speed	<input type="radio"/>	<input type="radio"/>	Integer	文字列が流れるスピードを指定します。
CaptionFont	<input type="radio"/>	<input type="radio"/>	Font	文字列のフォントを指定します。
Active	<input type="radio"/>	<input type="radio"/>	Boolean	True にすると文字列が流れ、False にすると止まります。
BeforeSpaces		<input type="radio"/>	Integer	それぞれの文字列の先頭に自動的にスペースを付加することが出来、スペースの数を指定します。

**イベント**

名称	摘要
MessageEnd	個々の文字列が流れ終わったときにイベントが発生します。

**利用例**

```
Public Class Form1

    Private Sub Form1_Load(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles MyBase.Load
        Dim strCaptions() As String = Nothing
        Dim intCNT As Integer
        CctTelop1.Caption = "最初からのメッセージ"
        For intCNT = 0 To 2
            ReDim Preserve strCaptions(intCNT)
            strCaptions(intCNT) = CStr(intCNT + 1) & "つ目の追加"
        Next
        CctTelop1.AddCaptions = strCaptions
        CctTelop1.AddCaption = "最後の追加"
    End Sub

    Private Sub CctTelop1_MessageEnd(ByVal sender As Object, _
        ByVal e As System.EventArgs) Handles CctTelop1.MessageEnd
        MsgBox("終わったよ!")
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles Button1.Click
        CctTelop1.InsCaption = "割り込み"
    End Sub

End Class
```

**概要**

標準の機能を拡張したテキストボックスです。  
 文字列・数値・日付・時刻・大文字・小文字 を登録するために最適化したコントロールとして動作します。  
 フォーカスが当たるとバックカラーが変化し、外れると戻ります。  
 入力値がエラーの場合、エラーの色になります。  
 桁数の指定やフォントサイズの変更により自動的に最適なサイズに変化します。  
 サイズは MS ゴシックを基準としているために、他のフォントでは適切に表示できません

**分類**

カスタムコントロール

**ファイル名**

cctTextBox.dll

**準備**

カスタムコントロールの準備方法を参照してください

**利用方法**

フォーム上に貼り付けてください

**スタイルモード**

次の各スタイルがあります。

- 0 文字列      ほとんどすべての文字列が入力できます。
- 1 数値        数字とマイナス・ピリオドのみが入力できます。  
指定範囲外の数値入力・数値以外の入力エラーになります。
- 2 日付        数字とスラッシュのみが入力できます。  
指定範囲外の日付入力・日付以外の入力エラーになります。
- 3 時刻        数字とコロンのみが入力できます。  
指定範囲外の時刻入力・時刻以外の入力エラーになります。
- 4 大文字      入力したアルファベットが大文字になります。
- 5 小文字      入力したアルファベットが小文字になります。
- 0・4・5 文字列全般 指定バイト数以上の入力・ホワイトリストやブラックリストに違反する入力エラーになります。

**プロパティ**

名称	StyleMode	型	摘要
StyleMode		Integer	0 : 文字列、1 : 数値、2 : 日付、3 : 時刻、4 : 大文字、5 : 小文字の入力を指定します。
BlackList	0・4・5	String	入力を禁止する文字列を指定します。
WhiteList	0・4・5	String	入力を許可する文字列を指定します。
clrGotFocus	0・1・2・3・4・5	Color	フォーカス取得時の BackColor を指定します。
clrError	0・1・2・3・4・5	Color	エラー発生時の BackColor を指定します。
MaxLength	0・4・5	Integer	入力可能バイト数を指定します。
MaxFigure	1	Integer	入力可能桁数を指定します。
FormatString	1・2・3	String	数値・日付・時刻の表示書式を指定します。
Value	0・1・2・3・4・5	String	数値・日付・時刻は指定した書式で、それ以外はそのままの値を String 型で返します。
MaxValue	1	Decimal	数値の最大値を指定します。 デフォルト : 2147483647
MinValue	1	Decimal	数値の最小値を指定します。 デフォルト : -2147483648
MaxDate	2	Date	日付の最大値を指定します。 デフォルト : 9999/12/31
MinDate	2	Date	日付の最小値を指定します。 デフォルト : 1900/01/01
MaxTime	3	Date	時刻の最大値を指定します。 デフォルト : 23:59:59
MinTime	3	Date	時刻の最小値を指定します。 デフォルト : 00:00:00

**概要** 指定した文字列が任意のポリシーに則ったものかどうかを判定します。  
 最大桁数・最小桁数の文字数範囲、禁止文字使用の有無、大文字・小文字・数字・記号の利用可否および最小要求桁数などを判定します。  
 ポリシーに則ったランダムな文字列を生成することが出来ます。

**分類** dll

**ファイル名** chkPassword.dll

**準備** dll の準備方法を参照してください

**利用方法** 次のようにしてオブジェクトを生成してください  
 Dim objPassword As New TechBee.clsPassword

**ファンクション**

名称	引数	型	摘要
gfnc_strRandomString	なし	String()	ポリシーに則ったランダムな文字列を返します。

**プロパティ**

名称	型	摘要
MaxLength	Integer	判定対象文字列の最大桁数を指定します。 デフォルト：8
MinLength	Integer	判定対象文字列の最小桁数を指定します。 デフォルト：6
NeedUpperCount	Integer	判定対象文字列のうち最低限必要な大文字の文字数を指定します。 デフォルト：0
NeedLowerCount	Integer	判定対象文字列のうち最低限必要な小文字の文字数を指定します。 デフォルト：0
NeedNumericCount	Integer	判定対象文字列のうち最低限必要な数字の文字数を指定します。 デフォルト：0
NeedSignCount	Integer	判定対象文字列のうち最低限必要な記号の文字数を指定します。 デフォルト：0
IsUpperUse	Boolean	判定対象文字列中の大文字の使用可否を指定します。 デフォルト：True
IsLowerUse	Boolean	判定対象文字列中の小文字の使用可否を指定します。 デフォルト：True
IsNumericUse	Boolean	判定対象文字列中の数字の使用可否を指定します。 デフォルト：True
IsSignUse	Boolean	判定対象文字列中の記号の使用可否を指定します。 デフォルト：True
SignList	String	記号として使用可能な文字列を指定します。 デフォルト：“-+/_”
BlackList	String	使用を禁止する文字列を指定します。 デフォルト：“アイウエオィウエイオカキクケコサンスセソタチツテトナニネノハヒフホマミムメヤユヨラリルロワヅ”

**利用例** サンプルプログラムをご確認ください

```
Public Class Form1

    Private m_objPassword As New TechBee.clsPassword

    Private Sub Form1_Load(ByVal sender As System.Object, _
        ByVal e As System.EventArgs) Handles MyBase.Load
        txtMaxLength.Value = m_objPassword.MaxLength
        txtMinLength.Value = m_objPassword.MinLength
        txtUpperCount.Value = m_objPassword.NeedUpperCount
    End Sub
End Class
```

```
txtLowerCount.Value = m_objPassword.NeedLowerCount
txtNumericCount.Value = m_objPassword.NeedNumericCount
txtSignCount.Value = m_objPassword.NeedSignCount
chkUpper.Checked = m_objPassword.IsUpperUse
chkLower.Checked = m_objPassword.IsLowerUse
chkNumeric.Checked = m_objPassword.IsNumericUse
chkSign.Checked = m_objPassword.IsSignUse
txtSignList.Text = m_objPassword.SignList
txtBlackList.Text = m_objPassword.BlackList
End Sub

Private Sub cmdCheck_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles cmdCheck.Click
    msub_setProperty()
    m_objPassword.Password = txtCheckString.Text
    If m_objPassword.Password.Length > 0 Then
        MsgBox("指定したパスワードは有効です。", _
            MsgBoxStyle.Information + MsgBoxStyle.OkOnly, "パスワード判定")
    End If
End Sub

Private Sub msub_setProperty()
    m_objPassword.MaxLength = txtMaxLength.Value
    m_objPassword.MinLength = txtMinLength.Value
    m_objPassword.NeedUpperCount = txtUpperCount.Value
    m_objPassword.NeedLowerCount = txtLowerCount.Value
    m_objPassword.NeedNumericCount = txtNumericCount.Value
    m_objPassword.NeedSignCount = txtSignCount.Value
    m_objPassword.IsUpperUse = chkUpper.Checked
    m_objPassword.IsLowerUse = chkLower.Checked
    m_objPassword.IsNumericUse = chkNumeric.Checked
    m_objPassword.IsSignUse = chkSign.Checked
    m_objPassword.SignList = txtSignList.Text
    m_objPassword.BlackList = txtBlackList.Text
End Sub

Private Sub cmdCreate_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles cmdCreate.Click
    msub_setProperty()
    MsgBox(m_objPassword.gfnc_strRandomString, _
        MsgBoxStyle.Information + MsgBoxStyle.OkOnly, "パスワード生成")
End Sub

Private Sub CheckBox_CheckedChanged(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles chkUpper.CheckedChanged, _
    chkLower.CheckedChanged, _
    chkNumeric.CheckedChanged, _
    chkSign.CheckedChanged

    If sender.checked Then
        sender.text = "許可"
    Else
        sender.text = "禁止"
    End If
End Sub

End Class
```

## 概要

Windows Form 上で、任意のファイルをドラッグアンドドロップすると、コントロール中のリストボックスにファイル名をフルパスで追加します。プロパティやファンクションによりリスト中のファイル名を取得することができます。右クリックメニューによりリスト中の任意のファイル名を除外することができます。リスト内の全選択・全解除をすることができます。

## 分類

ユーザーコントロール

## ファイル名

DropFileList.dll

## 準備

カスタムコントロールの準備方法を参照してください

## 利用方法

フォーム上に貼り付けてください

## サブプロシージャ

名称	引数	摘要
gsub_blnListClear	なし	リスト中の全ファイルを消去します。
gsub_setAllSelect	blnSelect	引数に True を指定するとリスト内の全ファイルを選択します。 False を指定すると全ファイルの選択を解除します。

## ファンクション

名称	引数	型	摘要
gfnc_strFileList	なし	String()	リスト中の全ファイル名を取得します。
gfnc_strFileList	blnSelect	String()	リスト中の全ファイル名を取得します。 引数に True を与えると選択データのみを返します。

## プロパティ

名称	Read	Write	型	摘要
FileName	<input type="radio"/>		String()	リスト中の全ファイル名を取得します。
FileName(blnSelect)	<input type="radio"/>		String()	リスト中の全ファイル名を取得します。 引数に True を与えると選択データのみを返します。

## 利用例

```
Public Class Form1
```

```
Private Sub cmdAllSelect_Click(ByVal sender As System.Object, _  
    ByVal e As System.EventArgs) Handles cmdAllSelect.Click  
    DropFileList1.gsub_setAllSelect(True)  
End Sub
```

```
Private Sub cmdExclusion_Click(ByVal sender As System.Object, _  
    ByVal e As System.EventArgs) Handles cmdExclusion.Click  
    DropFileList1.gsub_setAllSelect(False)  
End Sub
```

```
Private Sub cmdAllFile_Click(ByVal sender As System.Object, _  
    ByVal e As System.EventArgs) Handles cmdAllFile.Click  
    Dim strFileNames() As String = DropFileList1.FileName  
    Dim intCNT As Integer  
    Dim strMSG As String = ""  
    strMSG += "リスト中のファイル名は" & vbCrLf  
    If Not strFileNames Is Nothing Then  
        For intCNT = 0 To strFileNames.Length - 1  
            strMSG += strFileNames(intCNT) & vbCrLf  
        Next  
        strMSG += "です。"  
    Else  
        strMSG += "ありません"  
    End If  
    MsgBox(strMSG)  
End Sub
```

---

```
Private Sub cmdSelectedFile_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles cmdSelectedFile.Click
    Dim strFileNames() As String = DropFileList1.FileName(True)
    Dim intCNT As Integer
    Dim strMSG As String = ""
    strMSG += "選択されたファイル名は" & vbCrLf
    If Not strFileNames Is Nothing Then
        For intCNT = 0 To strFileNames.Length - 1
            strMSG += strFileNames(intCNT) & vbCrLf
        Next
        strMSG += "です。"
    Else
        strMSG += "ありません"
    End If
    MsgBox(strMSG)
End Sub

Private Sub cmdQuit_Click(ByVal sender As System.Object, _
    ByVal e As System.EventArgs) Handles cmdQuit.Click
    Me.Close()
End Sub

End Class
```

## 概要

プログラムの進捗状況を表示します。  
縦横 4 マス、計 16 マスのキューブを 2 進法で塗りつぶしてゆきます。  
定周期で進めることも、任意のタイミングで進めることも出来ます。

## 分類

ユーザーコントロール

## ファイル名

ColorfulCube.dll

## 準備

カスタムコントロールの準備方法を参照してください

## 利用方法

フォーム上に貼り付けてください

## サブプロシージャ

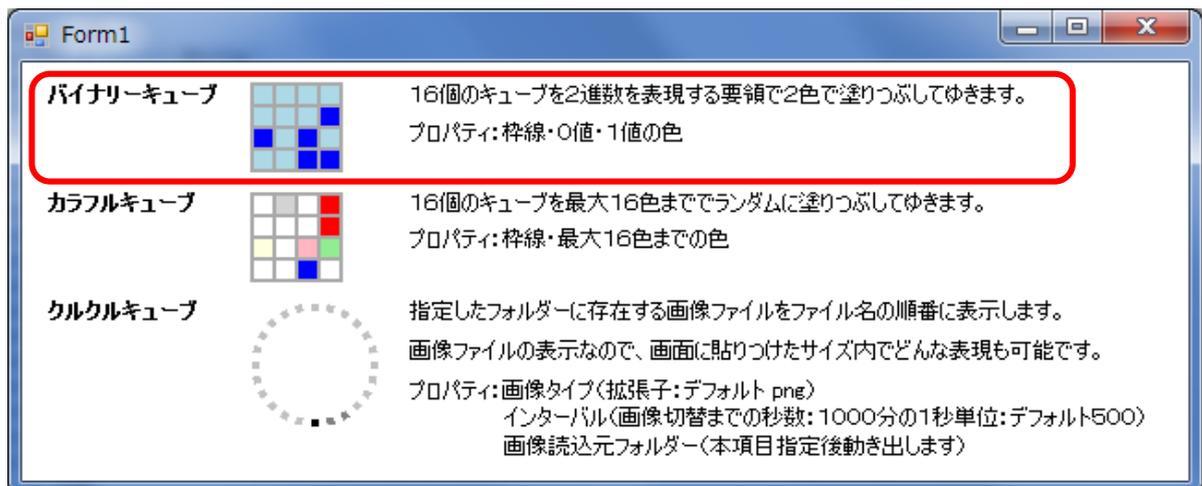
名称	引数	摘要
gsub_setCountUp	なし	任意のタイミングで手動でカウントアップします。

## プロパティ

名称	Read	Write	型	摘要
Activate		<input type="radio"/>	Boolean	True : 自動実行、False : 手動実行
CubeColor		<input type="radio"/>	Color	引数 0 : OFFの色、1 : ONの色
SquareColor		<input type="radio"/>	Color	枠の色
Interval		<input type="radio"/>	Integer	タイマーのインターバル

## 利用例

```
BinaryCube1.CubeColor(0) = Color.LightBlue
BinaryCube1.CubeColor(1) = Color.Blue
BinaryCube1.SquareColor = Color.DarkGray
BinaryCube1.Interval = 100
BinaryCube1.Activate = True
```



**概要** プログラムの進捗状況を表示します。  
縦横 4 マス、計 16 マスのキューブを最大 16 色で塗りつぶしてゆきます。  
定周期で塗り替えることも、任意のタイミングで塗り替えることも出来ます。

**分類** ユーザーコントロール

**ファイル名** ColorfulCube.dll

**準備** カスタムコントロールの準備方法を参照してください

**利用方法** フォーム上に貼り付けてください

**サブプロシージャ**

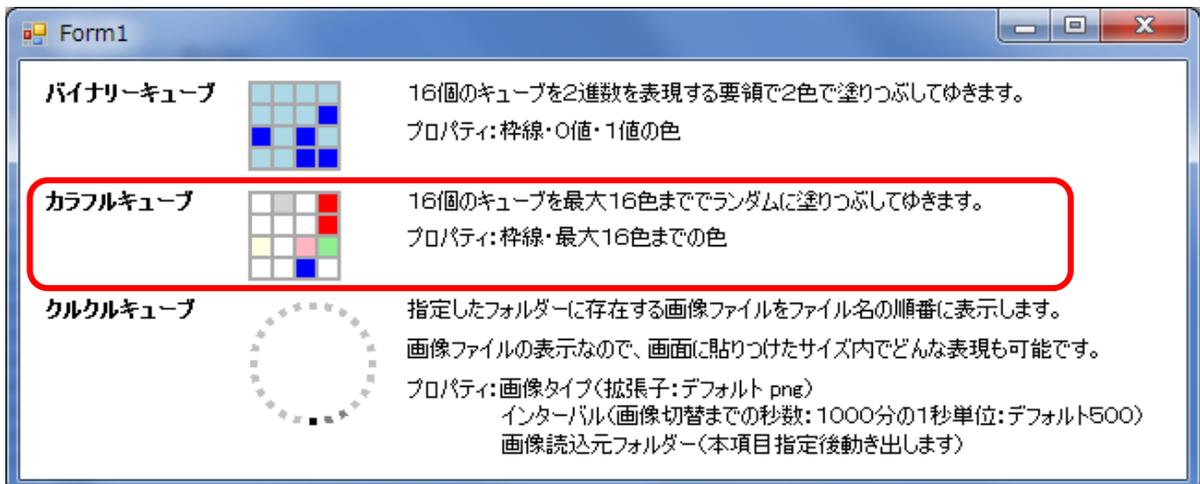
名称	引数	摘要
gsub_setCountUp	なし	任意のタイミングで手動でカウントアップします。

**プロパティ**

名称	Read	Write	型	摘要
Activate		<input type="radio"/>	Boolean	True : 自動実行、False : 手動実行
ColorCount		<input type="radio"/>	Integer	コントロール中で使用する色数
CubeColor		<input type="radio"/>	Color	0~15 のインデックスに割り当てたキューブの色 (ColorCountで指定した数までを使用します)
SquareColor		<input type="radio"/>	Color	枠の色
Interval		<input type="radio"/>	Integer	タイマーのインターバル

**利用例**

```
ColorfulCube1.ColorCount = 15
ColorfulCube1.CubeColor(0) = Color.White
ColorfulCube1.CubeColor(1) = Color.White
ColorfulCube1.CubeColor(2) = Color.White
ColorfulCube1.CubeColor(3) = Color.White
ColorfulCube1.CubeColor(4) = Color.White
ColorfulCube1.CubeColor(5) = Color.White
ColorfulCube1.CubeColor(6) = Color.White
ColorfulCube1.CubeColor(7) = Color.Blue
ColorfulCube1.CubeColor(8) = Color.Red
ColorfulCube1.CubeColor(9) = Color.Yellow
ColorfulCube1.CubeColor(10) = Color.LightBlue
ColorfulCube1.CubeColor(11) = Color.LightPink
ColorfulCube1.CubeColor(12) = Color.LightYellow
ColorfulCube1.CubeColor(13) = Color.LightGreen
ColorfulCube1.CubeColor(14) = Color.LightGray
ColorfulCube1.CubeColor(15) = Color.Red
ColorfulCube1.SquareColor = Color.DarkGray
ColorfulCube1.Interval = 100
ColorfulCube1.Activate = True
```



**概要**

プログラムの進捗状況を表示します。  
 指定したフォルダーに置かれた画像ファイルを定周期で切り替えて表示します。  
 存在するファイルすべてを表示後先頭ファイルに戻りますから無限ループで画像が切り替わります。  
 画像ファイルさえ用意していただければ事実上どんな表現も可能です。

**分類**

ユーザーコントロール

**ファイル名**

ColorfulCube.dll

**準備**

カスタムコントロールの準備方法を参照してください

**利用方法**

フォーム上に貼り付けてください

**サブプロシージャ**

名称	引数	摘要
gsub_setCountUp	なし	任意のタイミングで手動でカウントアップします。

**プロパティ**

名称	Read	Write	型	摘要
DataPath		<input type="radio"/>	String	画像を保存したフォルダーを指定します。
FileType		<input type="radio"/>	String	画像ファイルの拡張子を指定します。 デフォルト: png
Interval		<input type="radio"/>	Integer	タイマーのインターバル

**利用例**

```
CurucuruCube1.BackColor = Color.White
CurucuruCube1.Interval = 100
CurucuruCube1.FileType = "bmp"
CurucuruCube1.DataPath = ".\curucuru"
```

