

電子印鑑システム

パソコン決裁6

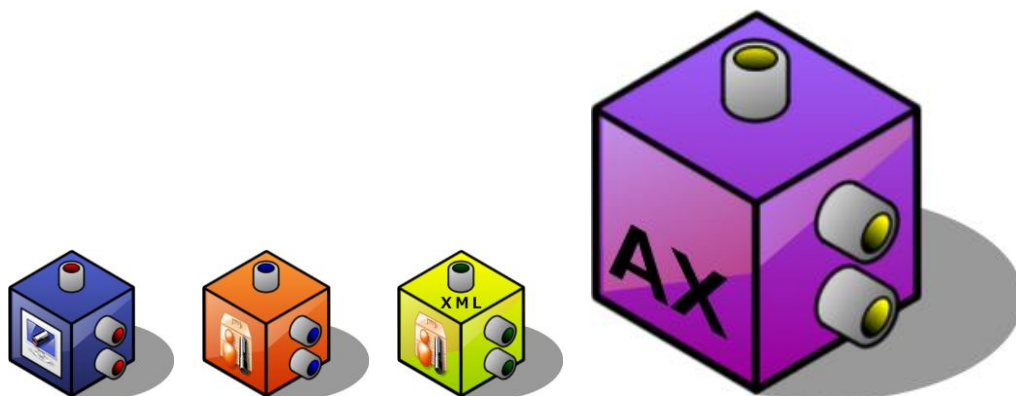
D i g i t a l S t a m p S e r i e s

E x t e n s i o n K i t

サービスリリース 1 試用版

リファレンス マニュアル

電子印鑑 コントロール編



内容

1	はじめに	1
2	電子印鑑 コントロールの概要	1
3	電子印鑑 コントロールのファイル名	1
4	セットアップ	1
5	電子印鑑 コントロールの操作	2
5.1	Visual Basic(Microsoft Visual Basic 2008)の場合	2
5.2	HTML ファイルの場合	4
6	メソッドとプロパティ	5
	BackColor	5
	BackColorColor	6
	BackColorStyle	7
	BackStyle	8
	CtrlStatus	9
	DataBlockFile	10
	DeviceID	11
	DrawPatternMode	12
	EnableDateLabel	13
	ErrorNumber	14
	ErrorTrap	15
	EsdData	16
	FaceText	17
	ForeColor	18
	ForegroundColor	19
	PropertyMode	20
	ReadOnly	21
	ReadyState	22
	StampAreaHeight	23
	StampAreaWidth	24
	StampHeight	25
	StampWidth	26
	UserID	27
	UserPass	28
	VersionText	29
	Zoom	30
	AboutBox	31
	DoLogin	32

GetDevice	33
Init.....	34
Refresh.....	35
Click	36
Loaded	37
OverArea	38

1 はじめに

このドキュメントには、[パソコン決裁6 Extension Kit]内の電子印鑑 コントロールモジュールについての説明がされています。また、本ドキュメントには Windows オペレーティングシステムの操作方法やパソコン決裁6についての操作方法などを理解している方を対象に記載されています。

試用版のご利用について

ダウンロードいただいた、モジュールおよび関連するドキュメントは、ご評価を目的としたモジュールになります。そのため、本モジュールを利用して発生したコンピュータについての障害や現象については、当社は一切の保障を行いません。また、本モジュールにつきましてのご質問やお問い合わせについてはご回答を差し上げることができませんので、あらかじめご了承ください。

2 電子印鑑 コントロールの概要

電子印鑑 コントロールは参照された ESD ファイル(拡張子.ESD)から、印影イメージや捺印プロパティ情報を取得し、コントロール内に印影イメージの表示を行います。

3 電子印鑑 コントロールのファイル名

電子印鑑 XML Extension は次のファイルで構成されています。

名称	ファイル名	機能
電子印鑑 コントロール	Axstmpr.dll	印影イメージを表示する ActiveX コントロールです。

※32ビットで動作するモジュールです。

4 セットアップ

電子印鑑 コントロールのインストールは次のように行います。

- メディア内の[bin]フォルダから[Axstmpr.dll]を適当なフォルダにコピーします。
登録の際に、英数字以外の文字が利用されている場合(例: "C:¥サンプル" フォルダなど)では正常に動作しない場合がございますので必ず半角英数字(例: "C:¥ComModule" フォルダなど)が利用されたフォルダを指定してください。
- コマンドプロンプトを開きます。
Windows Vista の場合には管理権限で操作を行う必要があります。
- 表示されたコマンドプロンプトの画面に次のように入力します。
regsvr32 <コピーしたモジュールのパス>¥Axstmpr.dll
- 登録が成功した旨のメッセージが表示されます。

5 電子印鑑 コントロールの操作

電子印鑑 コントロールで処理を行うために次のような実装を行います。また、この操作を行う場合にはあらかじめ電子印鑑 Extension を利用して ESD 形式のファイルを準備しておく必要があります。

5.1 Visual Basic(Microsoft Visual Basic 2008)の場合

電子印鑑 コントロールを Visual Basic で実装するためには次のような手順で行います。以下に手順またはコードは、実装を理解しやすくするための例示になります。開発される環境などによって操作方法や名称などが異なる場合がございますので、あらかじめご了承ください。

1. Visual Basic を起動します。
2. [ファイル | 新規作成 | プロジェクト]の順にメニューを選択します。
3. [新しいプロジェクト]画面で[プロジェクトの種類]に[Visual Basic]が選択されていることを確認します。
(選択されていない場合には、[他の言語]などから探して選択します)
4. [テンプレート]から[Windows フォーム アプリケーション]を選択します。
5. [プロジェクト名]に"DstmpWindowsApplication"と入力して[OK]ボタンを選択します。
6. 新しいプロジェクトファイルが作成され、[Form1.vb]のデザイン画面が表示されます。
7. [表示 | ツールボックス]メニューを選択します。
8. 表示された[ツールボックス]を右クリックして表示されたポップアップメニューから[アイテムの選択]メニューを選択します。
9. [ツールボックス アイテムの選択]内の[COM コンポーネント]タブを選択し、一覧で表示された名前から[AXstamper Class]の左側のチェックボックスを付けて[OK]ボタンを選択します。
10. ツールボックス内に[AXstamper Class]が追加されます。
11. 追加された[AXstamper Class]をマウスの左ボタンで選択し、マウスの左ボタンを押したままの状態ですべての[Form1]フォーム内に移動します。
12. [Form1]フォーム内の適当な場所でマウスの左ボタンを離します。
(フォーム内に[AxAXstamper1]コントロールが追加されます)
13. マウスを使って適当なサイズになるように調整します。
14. [表示 | ツールボックス]メニューを選択します。
15. 表示された[ツールボックス]から[コモン コントロール]内の[Button]をマウスの左ボタンで選択し、マウスの左ボタンを押したままの状態ですべての[Form1]フォーム内に移動します。
16. [Form1]フォーム内の適当な場所でマウスの左ボタンを離します。
(フォーム内にボタンコントロールが追加されます)
17. 追加されたボタンコントロールをマウスの左ボタンでダブルクリックします。

18. 次のようなコードが追加されます。

```
Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

    End Sub
End Class
```

19. 追加されたコード内の"Private Sub..." と、"End Sub" の間に次のコードを挿入します。

```
AxAXstamper1.set_CtrlStatus(0)
AxAXstamper1.DataBlockFile = "<準備した ESD 形式ファイルの場所>"
```

20. [Form1]フォームのデザイン画面に移動して、表示されたフォーム内の適当な場所(ボタンやコントロールが配置されていない背景の部分)をマウスの左ボタンでダブルクリックします。

21. 次のようなコードが追加されます。

```
Public Class Form1
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load

    End Sub
End Class
```

22. 追加されたコード内の"Private Sub..." と、"End Sub" の間に次のコードを挿入します。

```
AxAXstamper1.set_CtrlStatus(1)
```

23. [デバッグ | デバッグ開始]メニューを選択します。

24. ビルドが開始され、フォームが表示されます。

(追加したコントロールには"印"の文字列が表示されています)

25. 表示された[Button1]ボタンを選択します。

26. コントロールに ESD ファイル内の印影イメージが表示されます。

(印影をマウスの左ボタンでダブルクリックするとプロパティ情報が表示されます)

27. [デバッグ | デバッグの停止]メニューを選択します。

5.2 HTML ファイルの場合

電子印鑑 XML Extension を HTML 上に配置して実装するためには次のような手順で行います。以下に手順またはコードは、実装を理解しやすくするための例示になります。開発される環境などによって操作方法や名称などが異なる場合がございますので、あらかじめご了承ください。

1. メモ帳を起動します。
2. 表示されたメモ帳に次のサンプルを入力します。

```
<html>
<head>
<title>電子印鑑コントロール 表示サンプル</title>
</head>
<body>
<object width="64px" height="64px" id="AXstmpr1"
classid="CLSID:DAFDB38F-79E4-45AA-B969-A1C57B5DC7CE">
  <param name="CtrlStatus" value="0">
  <param name="DataBlockFile" value="＜準備したESD形式ファイルの場所＞" >
</object>

</body>
</html>
```

3. [ファイル | 名前を付けて保存]メニューを選択します。
4. 表示された[名前を付けて保存]ダイアログボックスの[ファイル名]に適切なファイル名(拡張子は.html にしてください)を入力して[保存]ボタンを選択します。
5. 保存された HTML ファイルをダブルクリックなどで開きます。
6. セキュリティの警告が表示される場合には表示の許可などの設定または選択を行い、HTML ファイルを開きます。
7. Web ブラウザ内で HTML に配置されたコントロール内に ESD ファイル内の印影が表示されます。
(印影をマウスの左ボタンでダブルクリックするとプロパティ情報が表示されます)

6 メソッドとプロパティ

電子印鑑 コントロール > IAXstamper	プロパティ
BackColor	
コントロール背景色の設定・取得を行います。	
BackColor = OLE_COLOR	
パラメータ ありません	
戻り値 OLE_COLOR:数値による設定・取得が行えます。	
備考 本プロパティに利用する値は OLE_COLOR で設定されている値を設定します。また、BackColor プロパティが設定されている場合には、本プロパティに設定されている値は無視され BackColor プロパティの値が優先されます。	
使用例 設定:オブジェクト.BackColor = コントロールの背景色 取得:コントロールの背景色 = オブジェクト.BackColor	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
BackColor	
コントロール背景色の設定・取得を行います。	
BackColor = string	
<p>パラメータ ありません</p> <p>戻り値 string:文字列による設定・取得が行えます。 設定する値は#RRGGBB 形式で行います。(例 赤の場合 #FF0000)</p> <p>備考 本プロパティに利用する値は#RRGGBB で設定します。また、BackColor プロパティが設定されている場合でも、本プロパティに設定されている値が優先されます。</p>	
<p>使用例 設定:オブジェクト.BackColor = "#909090" 取得:"コントロールの背景色" = オブジェクト.BackColor</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
BackGroundStyle	
コントロール背景透過の設定・取得を行います。	
BackGroundStyle = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。 0:透過なし 1:背景透過</p> <p>備考</p>	
<p>使用例 設定:オブジェクト.BackGroundStyle = コントロールの背景透過 取得:コントロールの背景透過 = オブジェクト.BackGroundStyle</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
BackStyle	
コントロール背景透過の設定・取得を行います。	
BackStyle = VARIANT_BOOL	
<p>パラメータ ありません</p> <p>戻り値 VARIANT_BOOL: 数値による設定・取得が行えます。 VARIANT_FALSE: 透過なし VARIANT_TRUE: 背景透過</p> <p>備考 本プロパティに利用する値は VARIANT_BOOL で設定されている値を設定します。また、BackGroundStyle プロパティが設定されている場合には、本プロパティに設定されている値は無視され BackGroundStyle プロパティが優先されます。</p>	
<p>使用例 設定: オブジェクト.BackStyle = コントロールの背景透過 取得: コントロールの背景透過 = オブジェクト.BackStyle</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
CtrlStatus	
<p>コントロールの動作状態を設定・取得します。</p> <p>Visual Basicなどでフォーム上に配置した場合には set_CtrlStatus(設定)、get_CtrlStatus(取得)を利用してください。</p>	
CtrlStatus = long long nCtrlStatus = get_CtrlStatus() set_CtrlStatus(long nCtrlStatus)	
<p>パラメータ ありません</p> <p>戻り値 long: 数値による設定・取得が行えます。 次の何れかの値を設定または取得がされます。 0: ESD データによる印影表示を行います。 1: マウスのクリックを検知してログイン用のダイアログボックスを表示します。(既定値) 2: インプレットのデバイスによるクリックを検知してデバイス ID を取得します。</p> <p>備考 本プロパティは、コントロールの状態について設定を行います。0 が設定されている場合にはコントロールは DataBlockFile プロパティで設定された ESD データを読み込みコントロール内に印影を表示する処理を行い、1 および 2 が設定された場合には、マウスまたはインプレットのデバイスによるクリックイベントからそれぞれの処理を行います。</p>	
<p>使用例 設定: オブジェクト.CtrlStatus = コントロールの動作状態 取得: 現在のコントロールの動作状態 = オブジェクト.CtrlStatus</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
DataBlockFile	
ESD 形式のデータファイルの位置を設定・取得します。設定される文字列は“¥”マークで区切られたファイルパスや“/”で区切られた URL を利用できます。また、URL については相対パスを指定することもできます。	
DataBlockFile = string	
<p>パラメータ ありません</p> <p>戻り値 string:文字列による設定・取得が行えます。</p> <p>備考 設定可能な ESD 形式のデータファイルは、DstmpLib::IExtension 内の GenStampData メソッドで出力されたデータを設定します。</p>	
<p>使用例 設定:オブジェクト.DataBlockFile = ESD 形式のデータファイルの場所 取得:現在設定されている ESD 形式のデータファイルの場所 = オブジェクト.DataBlockFile</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
DeviceID	
インプレットのデバイス ID を取得します。 Visual Basic などフォーム上に配置した場合には set_DeviceID(設定)、get_DeviceID(取得) を利用してください。	
DeviceID = string string strDeviceID = get_DeviceID() set_DeviceID(string strDeviceID)	
パラメータ ありません	
戻り値 string:文字列による設定・取得が行えます。	
備考 本プロパティで取得される値は、GetDeviceID メソッドや CtrlStatus プロパティに 2 が設定された状態でインプレットのデバイスによるクリックによる取得で変化します。また直接文字列を指定することもできます。	
使用例 設定:オブジェクト.DeviceID = "インプレットのデバイス ID" 取得:インプレットのデバイス ID = オブジェクト.DeviceID	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
DrawPatternMode	
表示される描画される際の印影イメージの平均化設定を行います。	
DrawPatternMode = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。 8:描画するピクセル情報の周囲 8 ピクセルを利用してイメージの平均化を行います(初期値) 3:描画するピクセル情報の周囲 3(下、右、右下)ピクセルを利用して平均化を行います</p> <p>備考 本プロパティは、DataBlockFile プロパティなどで ESD 形式のデータがコントロール上に読み込まれた場合のイメージの描画方法について設定・取得を行います。</p>	
<p>使用例 設定:オブジェクト.DrawPatternMode = 8 取得:印影の表示高さ = オブジェクト.DrawPatternMode</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
EnableDateLabel	
印影の下部に日付が表示される場合に日付の表示・非表示を設定・取得します。	
EnableDateLabel = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。</p> <p>備考 印影の中央部に日付が表示される印面(日付印)の場合には、本プロパティに設定されている値は無視されます。</p>	
<p>使用例 設定:オブジェクト.EnableDateLabel = 1 取得:日付の表示状態 = オブジェクト.EnableDateLabel</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
ErrorNumber	
コントロールのエラー発生時のエラー番号を取得します。	
ErrorNumber = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による取得が行えます。</p> <p>備考</p>	
<p>使用例 取得:エラートラップの状態 = オブジェクト.ErrorNumber</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
ErrorTrap	
コントロールのエラー発生時の動作を設定・取得します。	
ErrorTrap = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。 0:エラートラップなし 1:エラートラップあり(既定値)</p> <p>備考 エラートラップが有効な場合にはエラー発生時にエラーメッセージが表示されるようになります。エラートラップが無効な場合には例外が投入されます。エラーの原因を取得するには ErrorNumber プロパティを利用します。</p>	
<p>使用例 設定:オブジェクト.ErrorTrap = エラートラップの状態 取得:エラートラップの状態 = オブジェクト.ErrorTrap</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
EsdData	
ESD 形式のデータファイルのデータを設定・取得します。	
EsdData = string	
<p>パラメータ ありません</p> <p>戻り値 string:文字列による設定・取得が行えます。</p> <p>備考 設定可能な ESD 形式のデータファイルは、DstmpLib::IExtension 内の GenStampDataBase64 メソッドで出力されたデータを設定します。</p>	
<p>使用例 設定:オブジェクト.EsdData = ESD 形式のデータ 取得:現在設定されている ESD 形式のデータ = オブジェクト.EsdData</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
FaceText	
コントロールがユーザによるイベント待機状態の場合に表示する文字列を設定・取得します。	
FaceText = string	
<p>パラメータ ありません</p> <p>戻り値 string:文字列による設定・取得が行えます。</p> <p>備考 本プロパティで取得される値は、CtrlStatus プロパティに 1 または 2 が設定された状態でマウスのクリックによるイベントを待つ状態になった場合にコントロール上に表示される文字列(初期値 "印")になります。</p>	
<p>使用例 設定:オブジェクト.FaceText = "印" 取得:コントロールに表示される文字列 = オブジェクト.FaceText</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
ForeColor	
コントロール前景色の設定・取得を行います。	
ForeColor = OLE_COLOR	
<p>パラメータ ありません</p> <p>戻り値 OLE_COLOR:数値による設定・取得が行えます。</p> <p>備考 本プロパティに利用する値は OLE_COLOR で設定されている値を設定します。また、ForegroundColor プロパティが設定されている場合には、本プロパティに設定されている値は無視され ForegroundColor プロパティの値が優先されます。</p>	
<p>使用例 設定:オブジェクト.ForeColor = コントロールの前景色 取得:コントロールの前景色 = オブジェクト.ForeColor</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
ForeColorColor	
コントロール前景色の設定・取得を行います。	
ForeColorColor = string	
<p>パラメータ ありません</p> <p>戻り値 string:文字列による設定・取得が行えます。 設定する値は#RRGGBB 形式で行います。(例 赤の場合 #FF0000)</p> <p>備考 本プロパティに利用する値は#RRGGBB で設定します。また、ForeColor プロパティが設定されている場合でも、本プロパティに設定されている値が優先されます。</p>	
<p>使用例 設定:オブジェクト.ForeColorColor = "#909090" 取得:"コントロールの前景色" = オブジェクト.ForeColorColor</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
PropertyMode	
読み込まれた印影のプロパティ情報の表示状態について設定・取得を行います。	
PropertyMode = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。 0:プロパティ情報を表示しません。 1:プロパティ情報を表示します。</p> <p>備考 本プロパティは、DataBlockFile プロパティなどで ESD 形式のデータがコントロール上に読み込まれた場合にデータに含まれるプロパティ情報を別ダイアログボックスで表示する動作状態を設定します。プロパティ情報が表示される状態に設定されたコントロールでは、データの読み込み後にコントロールをダブルクリックした際に別画面でプロパティ情報が表示されます。</p>	
<p>使用例 設定:オブジェクト.PropertyMode = 1 取得:プロパティの表示状態 = オブジェクト.PropertyMode</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
ReadOnly	
コントロールで発生するイベント(マウスのクリックなど)の発生による動作を設定・取得します。	
ReadOnly = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。 0:コントロールのイベントを無効にする 1:コントロールのイベントを有効にする(初期値)</p> <p>備考</p>	
<p>使用例 設定:オブジェクト.ReadOnly = 1 取得:コントロールのイベント状態 = オブジェクト.ReadOnly</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
ReadyState	
コントロールの状態の設定・取得を行います。	
ReadyState = long	
<p>パラメータ ありません</p> <p>戻り値 long: 数値による設定・取得が行えます。 次の何れかの値を設定または取得がされます。 0(READYSTATE_UNINITIALIZED): コントロールは初期化されていません。 1(READYSTATE_LOADING): コントロールは非同期通信中です。 2(READYSTATE_LOADED): コントロールは非同期通信を終了しデータを読み込みました。 4(READYSTATE_COMPLETE): コントロールは処理を終了しました。</p> <p>備考</p>	
<p>使用例 設定: オブジェクト.ReadyState = コントロールの状態 取得: 現在のコントロールの状態 = オブジェクト.ReadyState</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
StampAreaHeight	
読み込まれた印影データの表示高さについて取得を行います。	
StampAreaHeight = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。</p> <p>備考 本プロパティは、DataBlockFile プロパティなどで ESD 形式のデータがコントロール上に読み込まれた場合に表示された印影の高さを取得します。</p>	
<p>使用例 取得:印影の表示高さ = オブジェクト.StampAreaHeight</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
StampAreaWidth	
読み込まれた印影データの表示幅について取得を行います。	
StampAreaWidth = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。</p> <p>備考 本プロパティは、DataBlockFile プロパティなどで ESD 形式のデータがコントロール上に読み込まれた場合に表示された印影の幅を取得します。</p>	
<p>使用例 取得:印影の表示幅 = オブジェクト.StampAreaWidth</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
StampHeight	
コントロールに表示されている印鑑データの横方向の高さを取得します。	
StampHeight = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による取得が行えます。</p> <p>備考 DataBlockFile プロパティなどが設定されていない場合には本プロパティで取得される値は不定です。</p>	
<p>使用例 取得:現在の印鑑データの長さ = オブジェクト.StampHeight</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
StampWidth	
コントロールに表示されている印鑑データの横方向の長さを取得します。	
StampWidth = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による取得が行えます。</p> <p>備考 DataBlockFile プロパティなどが設定されていない場合には本プロパティで取得される値は不定です。</p>	
<p>使用例 取得:現在の印鑑データの幅 = オブジェクト.StampWidth</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
UserID	
ユーザ名 (ユーザ ID) を設定・取得します。 Visual Basic などフォーム上に配置した場合には set_UserID (設定)、get_UserID (取得) を利用してください。	
UserID = string string strUserID = get_UserID() set_UserID(string strUserID)	
パラメータ ありません	
戻り値 string: 文字列による設定・取得が行えます。	
備考 本プロパティで取得される値は、CtrlStatus プロパティに 1 が設定された状態でマウスのクリックにより表示されたログイン用のダイアログ上でユーザが入力された値になります。また直接文字列を指定することもできます。	
使用例 設定: オブジェクト.UserID = "ユーザ ID" 取得: ユーザ ID = オブジェクト.UserID	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
UserPass	
ユーザのパスワードを設定・取得します。 Visual Basicなどでフォーム上に配置した場合には set_UserPass (設定)、get_UserPass (取得) を利用してください。	
UserPass = string string strUserPass = get_UserPass() set_UserPass(string strUserPass)	
パラメータ ありません	
戻り値 string:文字列による設定・取得が行えます。	
備考 本プロパティで取得される値は、CtrlStatus プロパティに 1 が設定された状態でマウスのクリックにより表示されたログイン用のダイアログ上でユーザが入力された値になります。また直接文字列を指定することもできます。	
使用例 設定:オブジェクト.UserPass = "ユーザパスワード" 取得:ユーザパスワード = オブジェクト.UserPass	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
VersionText	
コントロールのバージョン情報を文字列で取得します。	
VersionText = string	
<p>パラメータ ありません</p> <p>戻り値 string:文字列による取得が行えます。</p> <p>備考</p>	
<p>使用例 取得:コントロールのバージョン情報 = オブジェクト.VersionText</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	プロパティ
Zoom	
コントロールで表示される印影の表示倍率を設定・取得します。	
Zoom = long	
<p>パラメータ ありません</p> <p>戻り値 long:数値による設定・取得が行えます。</p> <p>備考</p>	
<p>使用例 設定:オブジェクト.Zoom = 100 取得:コントロールに表示される印影の表示倍率 = オブジェクト.Zoom</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	メソッド
AboutBox	
コントロールのバージョン情報ダイアログボックスを表示します。	
void AboutBox()	
<p>パラメータ ありません</p> <p>戻り値 ありません</p> <p>備考</p>	
<p>使用例 オブジェクト.AboutBox()</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	メソッド
DoLogin	
コントロールでログインダイアログが表示される状態にします。	
void DoLogin()	
<p>パラメータ ありません</p> <p>戻り値 ありません</p> <p>備考 本メソッドは、別コントロールからログインを実行するために用意されています。</p>	
<p>使用例 オブジェクト.Init()</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	メソッド
GetDevice	
インプレットデバイスからデバイス ID を取得します。取得された値は DeviceID プロパティを利用して確認することができます。	
void GetDevice()	
<p>パラメータ ありません</p> <p>戻り値 ありません</p> <p>備考 取得されたデバイス ID は DstmpLib::IExtension::LoginByInpplet メソッドの引数として利用します。</p>	
<p>使用例 オブジェクト.GetDevice()</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	メソッド
Init	
コントロールの初期化を行います。	
void Init()	
<p>パラメータ ありません</p> <p>戻り値 ありません</p> <p>備考 本メソッドを利用した場合には、すべてのデータ、変数が初期化状態になります。ESD データによる印影イメージの表示なども初期化されます。</p>	
<p>使用例 オブジェクト.Init()</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	メソッド
Refresh	
コントロールの表示状態を再表示します。	
void Refresh()	
<p>パラメータ ありません</p> <p>戻り値 ありません</p> <p>備考</p>	
<p>使用例 オブジェクト.Refresh()</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	イベント
Click	
コントロール上でマウスなどでクリックが行われた際に発生します。	
void Click()	
<p>パラメータ ありません</p> <p>戻り値 ありません</p> <p>備考</p>	
<p>使用例 オブジェクト.Click()</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	イベント
Loaded	
コントロール上で ESD ファイルの読み込みが完了した際に発生します。	
void Loaded()	
<p>パラメータ ありません</p> <p>戻り値 ありません</p> <p>備考</p>	
<p>使用例 オブジェクト.Loaded()</p>	
サンプルコード	

電子印鑑 コントロール > IAXstamper	イベント
OverArea	
コントロール上で ESD ファイルの読み込みが完了した際に表示される幅または高さが現在のコントロールを超えている場合に発生します。	
void OverArea()	
<p>パラメータ ありません</p> <p>戻り値 ありません</p> <p>備考</p>	
<p>使用例 オブジェクト.OverArea()</p>	
サンプルコード	

著作権情報

© 2010 Shachihata Inc. All rights reserved.

本マニュアルの内容は、著作権法により保護されています。

本マニュアルが使用許諾契約を含むソフトウェアと共に提供される場合、本マニュアルおよびその中に記載されているソフトウェアは、使用許諾契約にもとづいて提供されるものであり、当該使用許諾契約の契約条件に従ってのみ使用または複製することが可能となるものです。本マニュアルのいかなる部分も、発行者（シャチハタ株式会社）への書面による許可なしに、いかなる形式・手段でも、複製、検索システムへの保存、または伝送を行うことはできません。

本マニュアルに記載される内容は、可能な限り正確であることを考慮しますが予告なしに変更されることを条件として提供されるものであります。従って、本マニュアルの情報がシャチハタ株式会社による確約として解釈されるものではありません。シャチハタ株式会社は、本マニュアルにおけるいかなる誤りまたは不正確な記述に対しても、いかなる義務や責任を負うものではありません。

本マニュアルに記載されている会社名、製品名などは各社の商標または商標登録です。

電子印鑑 コントロール

サービスリリース 1 リファレンスマニュアル

2010/06 発行

発行者

シャチハタ株式会社

<http://www.shachihata.co.jp/>

パソコン決裁についての情報

<http://www.shachihata.co.jp/interweb/>