

電子印鑑システム

# パソコン決裁6

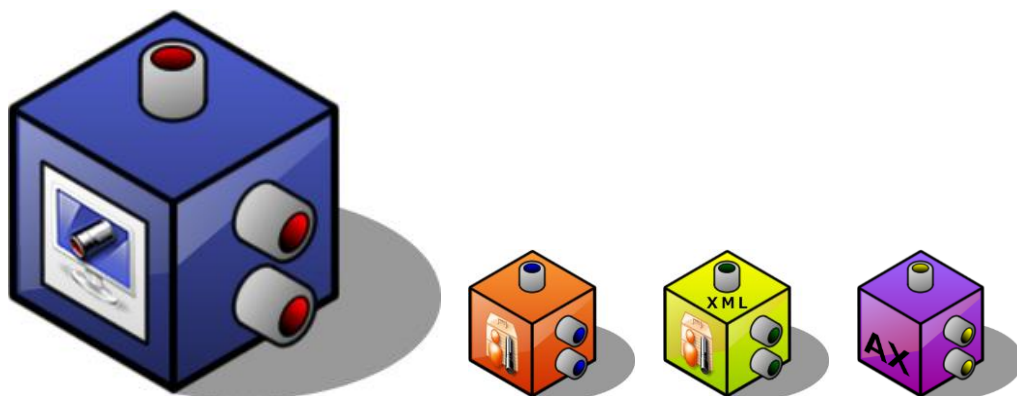
D i g i t a l S t a m p S e r i e s

## 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

捺印ツール Extension 編



## 内容

1	はじめに .....	1
2	捺印ツール Extension の概要 .....	1
3	捺印ツール Extension のファイル名 .....	1
4	セットアップ .....	1
5	捺印ツール Extension の操作 .....	2
5.1	VBScript (Visual Basic Script) の場合 .....	2
5.2	Visual Basic (Microsoft Visual Basic 2008) の場合 .....	3
6	捺印ツール Extension のコード例 .....	4
7	メソッドとプロパティ .....	5
	ActiveStamp .....	5
	ActiveStampName .....	6
	ActiveStampPosition .....	7
	AdditionDate .....	8
	AdditionDateFontBold .....	9
	AdditionDateFontItalic .....	10
	AdditionDateFontName .....	11
	AdditionDateFontSize .....	12
	AdditionDateFormat .....	13
	AdditionDateLogFlag .....	14
	AdditionDateStatus .....	15
	AdditionText .....	16
	AdditionTextFontBold .....	17
	AdditionTextFontItalic .....	18
	AdditionTextFontName .....	19
	AdditionTextFontSize .....	20
	AdditionTextLogFlag .....	21
	AdditionTextStatus .....	22
	ApplicationPath .....	23
	AttachmentFileCount .....	24
	AttachmentFileName .....	25
	AutoQuit .....	26
	BitBlt .....	27
	ColorBits .....	28
	CommentDialogCaption .....	29
	CommentMark .....	30
	CommentPassword .....	31

CommentPasswordDialogCaption.....	32
ComputerName .....	33
ComputerNameLogFlag .....	34
ConvertBitmapFont .....	35
CustomDrawPropertyDialogCaption.....	36
DatacutMode.....	37
DefaultFilePath .....	38
DialogTopMost.....	39
DocInfoDialogCaption.....	40
DocumentFileName.....	41
DocumentFileNameLogFlag.....	42
DocumentItem .....	43
DocumentItemLogFlag .....	44
DocumentNumber.....	45
DocumentNumberLogFlag.....	46
DocumentTitle.....	47
DocumentTitleLogFlag.....	48
DomainName .....	49
DomainNameLogFlag .....	50
DrawPattern.....	51
DrawPropertyDialogCaption .....	52
EnableBackDate.....	53
EnableDatacutMode .....	54
ErrorTrap .....	55
FileCID .....	56
FileDialogCaption.....	57
FilePath .....	58
GroupNamePath .....	59
GroupNamePathLogFlag .....	60
ImpressCount.....	61
ImpressCountFontBold .....	62
ImpressCountFontItalic .....	63
ImpressCountFontName .....	64
ImpressCountFontSize.....	65
ImpressCountLogFlag.....	66
ImpressCountStatus.....	67
ImpressID.....	68
ImpressIDFontBold .....	69
ImpressIDFontItalic .....	70

ImpressIDFontName .....	71
ImpressIDFontSize.....	72
ImpressIDLogFlag.....	73
ImpressIDStatus.....	74
ImpressPropertyDialogCaption .....	75
ImpressTime .....	76
LoginName.....	77
LoginNameLogFlag.....	78
NotesFlag.....	79
OptionDialogCaption .....	80
PaletteColor.....	81
Password .....	82
PasswordDialogCaption .....	83
QueryUpdate.....	84
Rop .....	85
RopCode.....	86
SerializeLog.....	87
SerializeStampBitmap.....	88
StampCID .....	89
StampCIDLogFlag .....	90
StampColor .....	91
StampCount .....	92
StampFaceDialogCaption .....	93
StampSerialNumber .....	94
StampSerialNumberLogFlag .....	95
StampType.....	96
StampVisible .....	97
StampVisiblePosition .....	98
StretchBltMode .....	99
UnlessDocInfoImpressStatus.....	100
UserCID.....	101
UserCIDLogFlag.....	102
UserDialogCaption .....	103
UserFirstName.....	104
UserLastName .....	105
UserMiddleName.....	106
UserName.....	107
UserPassword.....	108
UserPasswordDialogCaption .....	109

UserPasswordMinLength .....	110
WindowCaption .....	111
WindowDialogCaption .....	112
WindowLeft.....	113
WindowTop .....	114
WindowTopMost .....	115
WindowVisible .....	116
WriteLog .....	117
Zoom.....	118
AppendAttachmentFile .....	119
CommentDialog .....	120
Copy .....	121
DeleteAttachmentFile .....	122
DocInfoDialog .....	123
FileDialog.....	124
GetStamp.....	125
GetStampByE9 .....	126
GetStampByInpplet.....	127
GetUser .....	128
ImpressPropertyDialog .....	129
LoadStampProperties .....	130
OptionDialog .....	131
PasswordDialog .....	132
Quit.....	133
Quit_ .....	134
ResetFontStyles .....	135
ResetStampProperties .....	136
SaveStampProperties .....	137
StampFaceDialog .....	138
UserDialog .....	139
UserPasswordDialog.....	140
WindowDialog .....	141

## 1 はじめに

このドキュメントには、[パソコン決裁6 Extension Kit]内の捺印ツール Extension 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。また、本ドキュメントは作成時点のベータモジュールを対象にして記載されており、製品リリース時点で内容が変更される場合がございますのであらかじめご了承ください。

### 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 捺印ツール Extension の概要

捺印ツール Extension はクライアントにセットアップされている捺印ツールの機能を OLE オートメーションで動作させ、ユーザのログインや各種ダイアログボックスの表示、捺印などの操作をプログラムから制御を行います。

## 3 捺印ツール Extension のファイル名

捺印ツール 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
捺印ツール	Dstmp32.exe	捺印機能を操作する OLE サーバモジュールです。

※32 ビットで動作するモジュールです。

## 4 セットアップ

捺印ツール Extension の利用は既にクライアントに捺印ツールがセットアップされている場合には必要ありません。クライアントにセットアップされていない場合には、以下の手順で行います。

1. メディア内の [SetupDstmp32.msi] を実行します。
2. 捺印ツールのインストールが開始されます。
3. ウィザードに沿ってインストールを完了します。

## 5 捺印ツール Extension の操作

捺印ツール Extension で処理を行うために次のような実装を行います。

### 5.1 VBScript (Visual Basic Script) の場合

捺印ツール Extension をVBScript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. メモ帳を起動します。
2. 起動したメモ帳に以下のコードを入力します。

```
Dim objDstmp32
Set objDstmp32 = CreateObject("Dstmp.StampObject.6")
objDstmp32.UserDialog()
Set objDstmp32 = Nothing
msgbox "プログラムの実行が終了しました"
```

3. [ファイル]メニュー内の[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]画面で保存する場所を選択して[ファイル名]に"sample.vbs"と入力します。
5. [保存]ボタンを選択します。
6. 保存されたファイルをダブルクリックして実行を行います。
7. [ログイン機能の選択]ダイアログボックスが表示されます。
8. [パソコン決裁認証]を選択して[OK]ボタンを選択します。
9. [パソコン決裁 ログイン]ダイアログボックスが表示されます。
10. [OK]または[キャンセル]ボタンを選択してダイアログボックスを閉じます。
11. "プログラムの実行が終了しました"というメッセージが表示されます。

## 5.2 Visual Basic(Microsoft Visual Basic 2008)の場合

電子印鑑 XML Extension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]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フォーム内に移動します。
9. [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)
10. 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。
11. 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
```

12. 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。

```
Dim objDstmp32 As Object
objDstmp32 = CreateObject("Dstmp.StampObject.6")
objDstmp32.ShowDialog()
objDstmp32 = Nothing
msgbox "プログラムの実行が終了しました"
```

13. [デバッグ | デバッグ開始]メニューを選択します。
14. ビルドが開始され、フォームが表示されます。
15. 表示された[Button1]ボタンを選択します。
16. [ログイン機能の選択]ダイアログボックスが表示されます。
17. [パソコン決裁認証]を選択して[OK]ボタンを選択します。
18. [パソコン決裁 ログイン]ダイアログボックスが表示されます。
19. [OK]または[キャンセル]ボタンを選択してダイアログボックスを閉じます。
20. "プログラムの実行が終了しました"というメッセージが表示されます。



## 6 捺印ツール Extension のコード例

よく利用される捺印ツール Extension の VBScript での実装例として以下に記載します。

パソコン決裁のログインを行うダイアログを表示してユーザを取得する

```
Dim objDstmp32
Set objDstmp32 = CreateObject("Dstmp.StampObject.6")
if objDstmp32.UserDialog() = True then
    msgbox "ログインされたユーザ名: " & objDstmp32.UserName
end if
Set objDstmp32 = Nothing
```

パソコン決裁のログイン後に有効な印面を捺印オブジェクトとしてクリップボードにコピーする

```
Dim objDstmp32
Set objDstmp32 = CreateObject("Dstmp.StampObject.6")
if objDstmp32.UserDialog() = True then
    if objDstmp32.Copy(True) = True then `引数に False を指定した場合にはログに記録されません
        msgbox "クリップボードに捺印オブジェクトをコピーしました"
    end if
end if
Set objDstmp32 = Nothing
```

ユーザ名とパスワードを引数として受け渡してユーザを取得する

```
Dim objDstmp32
Dim strUserName:strUserName = "<取得するユーザ名>"
Dim strUserPassword:strUserPassword = "<ユーザパスワード>"
Set objDstmp32 = CreateObject("Dstmp.StampObject.6")
if objDstmp32.GetUser(strUserName, strUserPassword) = True then
    msgbox "ユーザを取得しました"
else
    msgbox "ユーザの取得に失敗しました"
end if
Set objDstmp32 = Nothing
```

## 7 メソッドとプロパティ

捺印ツール Extension > IDstmp32	プロパティ
ActiveStamp	
プロパティで指定した登録インデックスの番号に対応した印面を選択します。	
<b>ActiveStamp = long</b>	
<b>パラメータ</b> ありません	
<b>戻り値</b> long: 数値による設定・取得が行えます。	
<b>備考</b> 登録インデックスは、パソコン決裁 管理ツールで印鑑データを登録した際に自動的に設定される 0 から始まる数値で管理ツールで印鑑データのプロパティを表示した際に確認できます。	
<b>使用例</b> 設定: オブジェクト.ActiveStamp = 選択する登録インデックス 取得: 現在選択されている登録インデックス = オブジェクト.ActiveStamp	
<b>サンプルコード</b> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '選択されている印面の登録インデックスによる取得 msgbox objDstamp32.ActiveStamp Set objDstamp32 = Nothing	

捺印ツール Extension > IDstmp32	プロパティ
ActiveStampName	
プロパティで指定した品番文字列（印鑑データに登録されている機種名）の文字列に対応した印面を選択します。	
<b>ActiveStampName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> 品番文字列は、印鑑データを申込時に自動的に登録され変更することはできません。品番文字列を確認するには管理ツールで印鑑データの プロパティを表示します。また、同一の機種名が重複して登録されている場合には本メソッドを利用した場合に選択される印面は不定に なります。</p>	
<p><b>使用例</b> 設定:オブジェクト.ActiveStampName = 選択する機種文字列 取得:現在選択されている機種文字列 = オブジェクト.ActiveStampName</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '選択されている印面の機種による取得 msgbox objDstamp32.ActiveStampName '選択されている印面 Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>ActiveStampPosition</b>	
プロパティで指定したポジション（印鑑選択ウィンドウの左を 0 とした整数値）の番号に対応した印面を選択します。	
<b>ActiveStampPosition = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。</p> <p><b>備考</b> ポジションは印鑑選択ウィンドウの左を 0 とした整数値の番号を指定します。</p>	
<p><b>使用例</b>          設定:オブジェクト.ActiveStampPosition = 選択するポジション          取得:現在選択されているポジション = オブジェクト.ActiveStampPosition</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '選択されている印面のポジションによる取得 msgbox objDstamp32.ActiveStampPosition Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionDate	
印面の上部に表示される文字列の設定と取得を行います。	
<b>AdditionDate = date</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> date: 日付型による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>            設定: オブジェクト.AdditionDate = 表示する日付            取得: 現在設定されている日付 = オブジェクト.AdditionDate</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '印面日時のステータスを取得 msgbox objDstamp32.AdditionDateStatus '印面日時の書式を取得 msgbox objDstamp32.AdditionDateFormat '印面日時を取得 msgbox objDstamp32.AdditionDate Set objDstamp32 = Nothing           </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionDateFontBold	
印面の中央または下部に表示される日付の強調スタイルの設定と取得を行います。	
<b>AdditionDateFontBold = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:強調スタイルにしない 1:強調スタイルにする</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.AdditionDateFontBold = 1 取得:現在の設定値 = オブジェクト.AdditionDateFontBold</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印面日時の手体名の取得 msgbox objDstmp32.AdditionDateFontName '印面日時の手体斜体の取得 msgbox objDstmp32.AdditionDateFontItalic '印面日時の手体強調の取得 msgbox objDstmp32.AdditionDateFontBold '印面日時の手体サイズの取得 msgbox objDstmp32.AdditionDateFontSize Set objDstmp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionDateFontItalic	
印面の中央または下部に表示される日付の斜体スタイルの設定と取得を行います。	
<b>AdditionDateFontItalic = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:斜体スタイルにしない 1:斜体スタイルにする</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.AdditionDateFontItalic = 1 取得:現在の設定値 = オブジェクト.AdditionDateFontItalic</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印面日時の手体名の取得 msgbox objDstamp32.AdditionDateFontName '印面日時の手体斜体の取得 msgbox objDstamp32.AdditionDateFontItalic '印面日時の手体強調の取得 msgbox objDstamp32.AdditionDateFontBold '印面日時の手体サイズの取得 msgbox objDstamp32.AdditionDateFontSize Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionDateFontName	
印面の中央または下部に表示される日付の書体名の設定と取得を行います。	
<b>AdditionDateFontName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.AdditionDateFontName = "書体名"          取得:現在設定されている書体名 = オブジェクト.AdditionDateFontName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印面日時の書体名の取得 msgbox objDstmp32.AdditionDateFontName '印面日時の書体斜体の取得 msgbox objDstmp32.AdditionDateFontItalic '印面日時の書体強調の取得 msgbox objDstmp32.AdditionDateFontBold '印面日時の書体サイズの取得 msgbox objDstmp32.AdditionDateFontSize Set objDstmp32 = Nothing </pre>	



捺印ツール Extension > IDstmp32	プロパティ
AdditionDateFontSize	
印面の中央または下部に表示される日付の文字サイズの設定と取得を行います。	
<b>AdditionDateFontSize = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.AdditionDateFontSize = 1          取得:現在の設定値 = オブジェクト.AdditionDateFontSize</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印面日時の書体名の取得 msgbox objDstmp32.AdditionDateFontName '印面日時の書体斜体の取得 msgbox objDstmp32.AdditionDateFontItalic '印面日時の書体強調の取得 msgbox objDstmp32.AdditionDateFontBold '印面日時の書体サイズの取得 msgbox objDstmp32.AdditionDateFontSize Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionDateFormat	
印面の中央または下部に表示される日付書式の設定と取得を行います。	
<b>AdditionDateFormat = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による設定・取得が行えます。 氏名印（下部に日付が表示される印面の場合） 1:'yy.mm.dd 2:'yy.mm.dd 3:ee.mm.dd 4:ee/mm/dd 5:gee.mm.dd 6:gee/mm/dd 7:'yy/mm/dd hh:mm:ss 8:hh:mm:ss 9:YYYY.mm.dd 10:YYYY/mm/dd 日付印（中央に日付が表示される印面の場合） 1:'yy.mm.dd 2:'yy.mm.dd 3:ee.mm.dd 4:ee/mm/dd 5:gee.mm.dd 6</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定: オブジェクト.AdditionDateFormat = 1 取得: 現在設定されている印面日時の書式 = オブジェクト.AdditionDateFormat</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '印面日時のステータスを取得 msgbox objDstamp32.AdditionDateStatus '印面日時の書式を取得 msgbox objDstamp32.AdditionDateFormat '印面日時を取得 msgbox objDstamp32.AdditionDate Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionDateLogFlag	
印鑑データの印面日時に対応する捺印ログの収集状態を取得します。	
<b>AdditionDateLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.AdditionDateLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印面日時のログ収集状態の取得 if objDstamp32.AdditionDateLogFlag = True then     msgbox objDstamp32.AdditionDate end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionDateStatus	
印面の中央または下部に表示される日付の表示と表示色の設定と取得を行います。	
<b>AdditionDateStatus = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:非表示 1:黒色で表示 2:カラー（印面と同色）で表示</p> <p><b>備考</b> 印面の種類が日付印の場合には非表示(0)の設定を行うことはできません。</p>	
<p><b>使用例</b> 設定:オブジェクト.AdditionDateStatus = 1 取得:現在設定されている印面日時の表示と色 = オブジェクト.AdditionDateStatus</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '印面日時のステータスを取得 msgbox objDstamp32.AdditionDateStatus '印面日時の書式を取得 msgbox objDstamp32.AdditionDateFormat '印面日時を取得 msgbox objDstamp32.AdditionDate Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionText	
印面の上部に表示される文字列の設定と取得を行います。	
<b>AdditionText = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>            設定:オブジェクト.AdditionText = "表示する文字列"            取得:現在設定されている文字列 = オブジェクト.AdditionText</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '印面テキストのステータスを取得 msgbox objDstamp32.AdditionTextStatus '印面テキストの文字列を取得 msgbox objDstamp32.AdditionText Set objDstamp32 = Nothing           </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionTextFontBold	
印面の上部に表示される文字列の強調スタイルの設定と取得を行います。	
<b>AdditionTextFontBold = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:強調スタイルにしない 1:強調スタイルにする</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.AdditionTextFontBold = 1 取得:現在の設定値 = オブジェクト.AdditionTextFontBold</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印面テキストの書体名の取得 msgbox objDstamp32.AdditionTextFontName '印面テキストの書体斜体の取得 msgbox objDstamp32.AdditionTextFontItalic '印面テキストの書体強調の取得 msgbox objDstamp32.AdditionTextFontBold '印面テキストの書体サイズの取得 msgbox objDstamp32.AdditionTextFontSize Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionTextFontItalic	
印面の上部に表示される文字列の斜体スタイルの設定と取得を行います。	
<b>AdditionTextFontItalic = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:斜体スタイルにしない 1:斜体スタイルにする</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.AdditionTextFontItalic = 1 取得:現在の設定値 = オブジェクト.AdditionTextFontItalic</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印面テキストの書体名の取得 msgbox objDstamp32.AdditionTextFontName '印面テキストの書体斜体の取得 msgbox objDstamp32.AdditionTextFontItalic '印面テキストの書体強調の取得 msgbox objDstamp32.AdditionTextFontBold '印面テキストの書体サイズの取得 msgbox objDstamp32.AdditionTextFontSize Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionTextFontName	
印面の上部に表示される文字列の書体名の設定と取得を行います。	
<b>AdditionTextFontName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.AdditionTextFontName = "書体名"          取得:現在設定されている書体名 = オブジェクト.AdditionTextFontName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印面テキストの書体名の取得 msgbox objDstmp32.AdditionTextFontName '印面テキストの書体斜体の取得 msgbox objDstmp32.AdditionTextFontItalic '印面テキストの書体強調の取得 msgbox objDstmp32.AdditionTextFontBold '印面テキストの書体サイズの取得 msgbox objDstmp32.AdditionTextFontSize Set objDstmp32 = Nothing </pre>	



捺印ツール Extension > IDstmp32	プロパティ
AdditionTextFontSize	
印面の上部に表示される文字列の文字サイズの設定と取得を行います。	
<b>AdditionTextFontSize = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.AdditionTextFontSize = 1          取得:現在の設定値 = オブジェクト.AdditionTextFontSize</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印面テキストの書体名の取得 msgbox objDstmp32.AdditionTextFontName '印面テキストの書体斜体の取得 msgbox objDstmp32.AdditionTextFontItalic '印面テキストの書体強調の取得 msgbox objDstmp32.AdditionTextFontBold '印面テキストの書体サイズの取得 msgbox objDstmp32.AdditionTextFontSize Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionTextLogFlag	
印鑑データの印面テキストに対応する捺印ログの収集状態を取得します。	
<b>AdditionTextLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.AdditionTextLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印面テキストのログ収集状態の取得 if objDstamp32.AdditionTextLogFlag = True then     msgbox objDstamp32.AdditionText end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AdditionTextStatus	
印面の上部に表示される文字列の表示と表示色の設定と取得を行います。	
<b>AdditionTextStatus = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:非表示 1:黒色で表示 2:カラー（印面と同色）で表示</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.AdditionTextStatus = 1 取得:現在設定されている印面テキストの表示と色 = オブジェクト.AdditionTextStatus</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '印面テキストのステータスを取得 msgbox objDstmp32.AdditionTextStatus '印面テキストの文字列を取得 msgbox objDstmp32.AdditionText Set objDstmp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
ApplicationPath	
パソコン決裁 捺印ツールがインストールされている場所のパス情報を取得します。	
<b>ApplicationPath = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b> 取得される文字列は終端が¥マークで終了します。</p>	
<p><b>使用例</b>            設定:オブジェクト.ApplicationPath = "書体名"            取得:現在設定されている書体名 = オブジェクト.ApplicationPath</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'アプリケーションのインストール先を取得 msgbox objDstamp32.ApplicationPath Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
AttachmentFileCount	
添付ファイルとして設定されているファイルの数を取得します。	
<b>AttachmentFileCount = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:現在設定されている添付ファイル数 = オブジェクト.AttachmentFileCount</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True objDstmp32.AppendAttchmentFile("&lt;追加するファイルの場所&gt;") for nLoop = 0 to objDstmp32.AttachmentFileCount     msgbox objDstmp32.AttachmentFileName(nLoop)     objDstmp32.DeleteAttchemntFile(nLoop) next Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
AttachmentFileName	
引数で指定された位置に設定された添付ファイルのファイル名を取得します。	
<b>AttachmentFileName(long nPosition) = string</b>	
<p><b>パラメータ</b> nPosition:取得を行う添付ファイルの位置</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b> 指定する添付ファイルの位置は AppendAttachmentFile メソッドの実行された順番に 0 から始まる番号になります。</p>	
<p><b>使用例</b> 設定:オブジェクト.AttachmentFileName (0) = 1 取得:表示状態 = オブジェクト.AttachmentFileName (0)</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.AppendAttachmentFile("&lt;追加するファイルの場所&gt;") for nLoop = 0 to objDstamp32.AttachmentFileCount     msgbox objDstamp32.AttachmentFileName(nLoop)     objDstamp32.DeleteAttachmentFile(nLoop) next Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
AutoQuit	
ドラッグアンドドロップでの捺印操作の後にパソコン決裁を終了するかどうかの設定・取得を行います。	
<b>AutoQuit = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:ドラッグアンドドロップによる捺印後に終了しない 1:ドラッグアンドドロップによる捺印後に終了する</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.AutoQuit = 1 取得:現在の設定値 = オブジェクト.AutoQuit</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.WindowCaption = "&lt;印鑑選択ウィンドウのタイトルに表示する文字列&gt;" nResult = objDstamp32.UserDialog objDstamp32.WindowVisible = True objDstamp32.AutoQuit = 1 Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>BitBlt</b>	
印影を描画する際のビット転送の方法を設定・取得を行います。範囲外を指定した場合、その設定は無効になり、既定値および以前の設定値が残ります。	
<b>BitBlt = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。  1:印面イメージを捺印オブジェクトにビット転送を直接行います  2:印面イメージをメモリ上のデバイス非依存ビットマップ（DDB）に描画してから捺印オブジェクトにビット転送を直接行います  3:印面イメージをメモリ上のデバイス非依存ビットマップ（DDB）に描画してからデバイス依存ビットマップ（DIB）にパッケージ後、捺印オブジェクトにビット転送を直接行います</p> <p><b>備考</b></p>	
<p><b>使用例</b>  設定:オブジェクト.BitBlt = 1  取得:現在の設定値 = オブジェクト.BitBlt</p>	
<b>サンプルコード</b>	



捺印ツール Extension > IDstmp32	プロパティ
<b>ColorBits</b>	
印影を描画する際に使用するカラービットの設定・取得を行います。範囲外を指定した場合、その設定は無効になり、既定値および以前の設定値が残ります。	
<b>ColorBits = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 1:印面イメージを1ビット（2色）で描画します 4:印面イメージを1ビット（16色）で描画します 8:印面イメージを1ビット（256色）で描画します</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.ColorBits = 1 取得:現在の設定値 = オブジェクト.ColorBits</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
CommentDialogCaption	
コメントと添付ファイルの設定を行うダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>CommentDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"コメント"が表示されます。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b> 設定:オブジェクト.CommentDialogCaption = "キャプション文字列" 取得:"キャプション文字列" = オブジェクト.CommentDialogCaption</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() 'コメントダイアログのタイトル文字列を変更 objDstamp32.CommentDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstamp32.ShowDialog() Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
CommentMark	
捺印時にコメントや添付ファイルがある場合にコメントマークを表示する設定状態を設定・取得します。	
<b>CommentMark = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:コメントマークなし 1:コメントマークあり</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.CommentMark = 1 取得:現在の設定値 = オブジェクト.CommentMark</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'コメントマークの取得 msgbox objDstamp32.CommentMark Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
CommentPassword	
コメントや添付ファイルを確認するために入力するパスワードの設定・取得を行います。	
<b>CommentPassword = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"印鑑ファイルの指定"が表示されます。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b> 設定:オブジェクト.CommentPassword = "パスワード" 取得:"パスワード" = オブジェクト.CommentPassword</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'コメントパスワードの取得 msgbox objDstamp32.CommentPassword Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
CommentPasswordDialogCaption	
コメントパスワードを入力する際に表示されるダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>CommentPasswordDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"パスワードの入力"が表示されます。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b>            設定:オブジェクト.CommentPasswordDialogCaption = "キャプション文字列"            取得:"キャプション文字列" = オブジェクト.CommentPasswordDialogCaption</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
ComputerName	
捺印する環境設定のコンピュータ名に関する情報の取得を行います。	
<b>ComputerName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"コンピュータ名" = オブジェクト.ComputerName</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'コンピュータ名のログ収集状態の取得 if objDstamp32.ComputerNameLogFlag = True then     msgbox objDstamp32.ComputerName end if Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
ComputerNameLogFlag	
コンピュータ名に対応する捺印ログの収集状態を取得します。	
<b>ComputerNameLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.ComputerNameLogFlag</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'コンピュータ名のログ収集状態の取得 if objDstamp32.ComputerNameLogFlag = True then     msgbox objDstamp32.ComputerName end if Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>ConvertBitmapFont</b>	
印影を描画する際に印面テキスト、印面日時および捺印カウントのフォントもイメージに変換するかどうかの設定・取得を行います。設定した値は、印鑑毎に保存され、次回利用時の初期値となります。	
<b>ConvertBitmapFont = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0(FALSE):フォントをビットマップに変換しないで描画する 1 (TRUE):フォントをビットマップに変換して描画する</p> <p><b>備考</b> 範囲外の値を設定した場合には無効となり、既定値または設定以前の設定が残ります。</p>	
<p><b>使用例</b> 設定:オブジェクト.ConvertBitmapFont = 1 取得:現在設定されている値 = オブジェクト.ConvertBitmapFont</p>	
<b>サンプルコード</b>	



捺印ツール Extension > IDstmp32	プロパティ
CustomDrawPropertyDialogCaption	
描画パターンの詳細を入力する際に表示されるダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>CustomDrawPropertyDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"描画パターンの詳細"が表示されます。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b>          設定:オブジェクト.CustomDrawPropertyDialogCaption = "キャプション文字列"          取得:"キャプション文字列" = オブジェクト.CustomDrawPropertyDialogCaption</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
DatacutMode	
捺印モードの設定と取得を行います。	
<b>DatacutMode = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 1:通常 2:簡易</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。このプロパティは必ず通常（1）が設定、取得されます。</p>	
<p><b>使用例</b> 設定:オブジェクト.DatacutMode = 1 取得:1 = オブジェクト.DatacutMode</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '捺印モードの取得 msgbox objDstamp32.DatacutMode '捺印モードの指定を可能にする状態の取得 msgbox objDstamp32.EnableDatacutMode Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
DefaultFilePath	
参照を行う捺印用印鑑データファイルの場所の設定、取得を行います。	
<b>DefaultFilePath = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、捺印ツールが起動した場合に参照を行う場所（前回起動時に参照を行った場所）が参照されます。</p>	
<p><b>使用例</b>            設定:オブジェクト.DefaultFilePath = "捺印用印鑑データファイルの場所"            取得:"捺印用印鑑データファイルの場所" = オブジェクト.DefaultFilePath</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;設定する捺印用印鑑データファイルの場所&gt;" Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>DialogTopMost</b>	
ダイアログボックス関連の表示を最前面で表示するかどうかを設定・取得を行います。	
<b>DialogTopMost = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0 (FALSE) :ダイアログボックスを最前面に表示しない 1 (TRUE) :ダイアログボックスを最前面に表示する</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。WindowTopMost プロパティを利用してください。</p>	
<p><b>使用例</b> 設定:オブジェクト.DialogTopMost = 1 取得:現在選択されている値 = オブジェクト.DialogTopMost</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
DocInfoDialogCaption	
文書情報を入力する際に表示されるダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>DocInfoDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"文書情報"が表示されます。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b> 設定:オブジェクト.DocInfoDialogCaption = "キャプション文字列" 取得:"キャプション文字列" = オブジェクト.DocInfoDialogCaption</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstamp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstamp32.DocumentFileNameLogFlag = True then     msgbox objDstamp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstamp32.DocumentTitleLogFlag = True then     msgbox objDstamp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstamp32.DocumentNumberLogFlag = True then     msgbox objDstamp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstamp32.DocumentItemLogFlag(1) = True then     msgbox objDstamp32.DocumentItem(1) end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
DocumentFileName	
捺印する印影オブジェクトに設定する文書ファイル名の設定・取得を行います。	
<b>DocumentFileName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>            設定:オブジェクト.DocumentFileName = "文書ファイル名"            取得:"文書ファイル名" = オブジェクト.DocumentFileName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstamp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstamp32.DocumentFileNameLogFlag = True then     msgbox objDstamp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstamp32.DocumentTitleLogFlag = True then     msgbox objDstamp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstamp32.DocumentNumberLogFlag = True then     msgbox objDstamp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstamp32.DocumentItemLogFlag(1) = True then     msgbox objDstamp32.DocumentItem(1) end if Set objDstamp32 = Nothing           </pre>	

捺印ツール Extension > IDstmp32	プロパティ
DocumentFileNameLogFlag	
文書情報の文書ファイル名に対応する捺印ログの収集状態を取得します。	
<b>DocumentFileNameLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.DocumentFileNameLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstamp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstamp32.DocumentFileNameLogFlag = True then     msgbox objDstamp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstamp32.DocumentTitleLogFlag = True then     msgbox objDstamp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstamp32.DocumentNumberLogFlag = True then     msgbox objDstamp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstamp32.DocumentItemLogFlag(1) = True then     msgbox objDstamp32.DocumentItem(1) end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
DocumentItem	
引数で指定された番号に対応する「文書情報」の「承認項目」に設定される文字列を設定・取得します。	
<b>DocumentItem (long nCount) = string</b>	
<p><b>パラメータ</b> nCount: 設定・取得を行う承認項目の番号 (0 から 4 までの整数値)</p> <p><b>戻り値</b> string: 文字列による設定・取得が行えます。</p> <p><b>備考</b> 引数で指定可能な数値は 1 から 5 までを指定します。</p>	
<p><b>使用例</b> 設定: オブジェクト.DocumentItem(0) = "承認項目" 取得: "承認項目" = オブジェクト.DocumentItem(0)</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstamp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstamp32.DocumentFileNameLogFlag = True then     msgbox objDstamp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstamp32.DocumentTitleLogFlag = True then     msgbox objDstamp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstamp32.DocumentNumberLogFlag = True then     msgbox objDstamp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstamp32.DocumentItemLogFlag(1) = True then     msgbox objDstamp32.DocumentItem(1) end if Set objDstamp32 = Nothing </pre>	



捺印ツール Extension > IDstmp32	プロパティ
<b>DocumentItemLogFlag</b>	
引数で指定された番号に対応する文書情報の承認項目に対応する捺印ログの収集状態を取得します。	
<b>DocumentItemLogFlag (long nCount) = long</b>	
<p><b>パラメータ</b> nCount:取得を行う承認項目の番号 (0 から 4 までの整数値)</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。引数で指定可能な数値は 1 から 5 までを指定します。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.DocumentItemLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstamp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstamp32.DocumentFileNameLogFlag = True then     msgbox objDstamp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstamp32.DocumentTitleLogFlag = True then     msgbox objDstamp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstamp32.DocumentNumberLogFlag = True then     msgbox objDstamp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstamp32.DocumentItemLogFlag(1) = True then     msgbox objDstamp32.DocumentItem(1) end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
DocumentNumber	
捺印する印影オブジェクトに設定する文書番号の設定・取得を行います。	
<b>DocumentNumber = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>            設定:オブジェクト.DocumentNumber = "文書番号"            取得:"文書番号" = オブジェクト.DocumentNumber</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstamp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstamp32.DocumentFileNameLogFlag = True then     msgbox objDstamp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstamp32.DocumentTitleLogFlag = True then     msgbox objDstamp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstamp32.DocumentNumberLogFlag = True then     msgbox objDstamp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstamp32.DocumentItemLogFlag(1) = True then     msgbox objDstamp32.DocumentItem(1) end if Set objDstamp32 = Nothing           </pre>	

捺印ツール Extension > IDstmp32	プロパティ
DocumentNumberLogFlag	
文書情報の文書番号に対応する捺印ログの収集状態を取得します。	
<b>DocumentNumberLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.DocumentNumberLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True objDstmp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstmp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstmp32.DocumentFileNameLogFlag = True then     msgbox objDstmp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstmp32.DocumentTitleLogFlag = True then     msgbox objDstmp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstmp32.DocumentNumberLogFlag = True then     msgbox objDstmp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstmp32.DocumentItemLogFlag(1) = True then     msgbox objDstmp32.DocumentItem(1) end if Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
DocumentTitle	
捺印する印影オブジェクトに設定する文書タイトルの設定・取得を行います。	
<b>DocumentTitle = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.DocumentTitle = "文書タイトル"          取得:"文書タイトル" = オブジェクト.DocumentTitle</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstamp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstamp32.DocumentFileNameLogFlag = True then     msgbox objDstamp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstamp32.DocumentTitleLogFlag = True then     msgbox objDstamp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstamp32.DocumentNumberLogFlag = True then     msgbox objDstamp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstamp32.DocumentItemLogFlag(1) = True then     msgbox objDstamp32.DocumentItem(1) end if Set objDstamp32 = Nothing         </pre>	

捺印ツール Extension > IDstmp32	プロパティ
DocumentTitleLogFlag	
文書情報の文書タイトルに対応する捺印ログの収集状態を取得します。	
<b>DocumentTitleLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.DocumentTitleLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True objDstmp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstmp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstmp32.DocumentFileNameLogFlag = True then     msgbox objDstmp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstmp32.DocumentTitleLogFlag = True then     msgbox objDstmp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstmp32.DocumentNumberLogFlag = True then     msgbox objDstmp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstmp32.DocumentItemLogFlag(1) = True then     msgbox objDstmp32.DocumentItem(1) end if Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
DomainName	
捺印する環境設定のドメイン名に関する情報の取得を行います。	
<b>DomainName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"ドメイン名" = オブジェクト.DomainName</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() 'ドメイン名のログ収集状態の取得 if objDstmp32.DomainNameLogFlag = True then     msgbox objDstmp32.DomainName end if Set objDstmp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
DomainNameLogFlag	
ドメイン名に対応する捺印ログの収集状態を取得します。	
<b>DomainNameLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.DomainNameLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'ドメイン名のログ収集状態の取得 if objDstamp32.DomainNameLogFlag = True then     msgbox objDstamp32.DomainName end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>DrawPattern</b>	
描画パターンの値を設定・取得を行います。描画に関するプロパティを変更した場合に、設定値は0（未設定）になります。	
<b>DrawPattern = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:(未設定) ユーザが描画に関する設定を独自で変更した場合 1:標準（透過なし） 印影の背景を白にする設定 2:標準（背景透過） 印影の背景を透過にする設定 3:特定環境（背景透過） 一太郎などの一部のアプリケーション用に背景を透過にする設定 4:以前のバージョン用の互換のために予約されています 5:データ削減（透過なし） 再描画用の情報を削除した形で背景を白に設定。但し、アプリケーションによっては、印影が正常に表示されなかったり、捺印プロパティが表示されない</p> <p><b>備考</b> 次のプロパティを変更した場合には、このメソッドは0（未設定）に変更されます。 影響を受けるプロパティまたは操作 〔描画パターンの詳細〕 ダイアログボックスから設定を変更操作、Rop プロパティ、ColorBits プロパティ、BitBlt プロパティ、StretchBltMode プロパティ、PaletteColor プロパティ、ConvertBitmapFont プロパティ、SerializeStampBitmap プロパティ</p> <p><b>使用例</b> 設定:オブジェクト.DrawPattern = 選択する描画パターン 取得:現在選択する描画パターン = オブジェクト.DrawPattern</p> <p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '描画パターンの取得 msgbox objDstamp32.DrawPattern Set objDstamp32 = Nothing</pre>	



捺印ツール Extension > IDstmp32	プロパティ
DrawPropertyDialogCaption	
描画プロパティの変更を入力する際に表示されるダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>DrawPropertyDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"描画プロパティ"が表示されます。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b>          設定:オブジェクト.DrawPropertyDialogCaption = "キャプション文字列"          取得:"キャプション文字列" = オブジェクト.DrawPropertyDialogCaption</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
<b>EnableBackDate</b>	
印面の日付を以前の日付に戻すことが許可されているかの状態を取得します。	
<b>EnableBackDate = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:バックデート禁止 1:バックデート許可</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているバックデートの状態 = オブジェクト.EnableBackDate</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'バックデート禁止設定の取得 msgbox objDstamp32.EnableBackDate Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
EnableDatacutMode	
捺印モードの指定を可能にする設定と取得を行います。	
<b>EnableDatacutMode = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による設定・取得が行えます。 0: 無効 1: 有効</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。このプロパティは必ず無効（0）が設定、取得されます。</p>	
<p><b>使用例</b> 設定: オブジェクト.EnableDatacutMode = 0 取得: 0 = オブジェクト.EnableDatacutMode</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '捺印モードの取得 msgbox objDstamp32.DatacutMode '捺印モードの指定を可能にする状態の取得 msgbox objDstamp32.EnableDatacutMode Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>ErrorTrap</b>	
エラーを捺印ツールでモジュールするかどうかの状態を設定・取得します。本プロパティの初期値は FALSE (0) が設定されており、エラーが発生した場合に、モジュールでエラーメッセージを表示します。TRUE (1) を設定すると、モジュールでのエラーメッセージを表示せず例外を投入します。例外の情報は OLE オートメーション対応アプリケーションで取得することができます。	
<b>ErrorTrap = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:エラーが発生した場合に、捺印ツールでエラーメッセージを表示する 1:エラーが発生した場合に、例外を投入してエラーを発生させる</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.ErrorTrap = 1 取得:現在の設定値 = オブジェクト.ErrorTrap</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'エラートラップを有効 objDstamp32.ErrorTrap = True '...モジュール側でエラーメッセージを表示 objDstamp32.ErrorTrap = False '...アプリケーション側でエラー対処 Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>FileCID</b>	
操作を行う捺印用印鑑データファイルを新規作成した際のクリエイト ID を数値型で取得します。取得される ID は管理ツールの捺印用印鑑データファイルのプロパティで 16 進数で表示確認できます。	
<b>FileCID = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:設定値 = オブジェクト.FileCID</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '印鑑 CID を取得 msgbox objDstamp32.StampCID 'ファイル CID を取得 msgbox objDstamp32.FileCID 'ユーザ名を取得 msgbox objDstamp32.UserName 'ユーザ CID を取得 msgbox objDstamp32.UserCID 'ユーザの姓を取得 msgbox objDstamp32.UserLastName 'ユーザの名を取得 msgbox objDstamp32.UserFirstName 'ユーザのミドルネームを取得 msgbox objDstamp32.UserMiddleName Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
FileDialogCaption	
捺印用印鑑データファイルの場所の設定を行うダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>FileDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"印鑑ファイルの指定"が表示されます。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b>            設定:オブジェクト.FileDialogCaption = "キャプション文字列"            取得:"キャプション文字列" = オブジェクト.FileDialogCaption</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
FilePath	
参照されている捺印用印鑑データファイルの場所を取得します。	
<b>FilePath = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"現在の捺印用印鑑データファイルの場所" = オブジェクト.FilePath</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") '参照先の捺印用印鑑データファイルの場所 objDstamp32.FilePath Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
GroupNamePath	
グループパス名に関する情報の取得を行います。	
<b>GroupNamePath = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b> グループパス名は、管理ツールによってログインを行うユーザがグループによる階層構造が設定されている場合に、階層構造の最上部からの構成を¥マークで区切られた文字列になります。</p>	
<p><b>使用例</b> 取得:"グループパス名" = オブジェクト.GroupNamePath</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'グループパス名のログ収集状態の取得 if objDstamp32.GroupNamePathLogFlag = True then     msgbox objDstamp32.GroupNamePath end if Set objDstamp32 = Nothing</pre>	



捺印ツール Extension > IDstmp32	プロパティ
GroupNamePathLogFlag	
グループパス名に対応する捺印ログの収集状態を取得します。	
<b>GroupNamePathLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.GroupNamePathLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'グループパス名のログ収集状態の取得 if objDstamp32.GroupNamePathLogFlag = True then     msgbox objDstamp32.GroupNamePath end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressCount	
印面の下部に表示される捺印カウンタの設定と取得を行います。	
<b>ImpressCount = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。</p> <p><b>備考</b> このプロパティは取得専用です。</p>	
<p><b>使用例</b> 取得:現在設定されている捺印カウンタ = オブジェクト.ImpressCount</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '捺印カウンタのステータスを取得 msgbox objDstamp32.ImpressCountStatus '捺印カウンタの取得 msgbox objDstamp32.ImpressCount Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressCountFontBold	
印面の下部に表示される捺印カウンタの強調スタイルの設定と取得を行います。	
<b>ImpressCountFontBold = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:強調スタイルにしない 1:強調スタイルにする</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.ImpressCountFontBold = 1 取得:現在の設定値 = オブジェクト.ImpressCountFontBold</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '捺印カウンタの書体名の取得 msgbox objDstamp32.ImpressCountFontName '捺印カウンタの書体斜体の取得 msgbox objDstamp32.ImpressCountFontItalic '捺印カウンタの書体強調の取得 msgbox objDstamp32.ImpressCountFontBold '捺印カウンタの書体サイズの取得 msgbox objDstamp32.ImpressCountFontSize Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressCountFontItalic	
印面の下部に表示される捺印カウンタの斜体スタイルの設定と取得を行います。	
<b>ImpressCountFontItalic = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:斜体スタイルにしない 1:斜体スタイルにする</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.ImpressCountFontItalic = 1 取得:現在の設定値 = オブジェクト.ImpressCountFontItalic</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstmp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '捺印カウンタの書体名の取得 msgbox objDstamp32.ImpressCountFontName '捺印カウンタの書体斜体の取得 msgbox objDstamp32.ImpressCountFontItalic '捺印カウンタの書体強調の取得 msgbox objDstamp32.ImpressCountFontBold '捺印カウンタの書体サイズの取得 msgbox objDstamp32.ImpressCountFontSize Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressCountFontName	
印面の下部に表示される捺印カウンタの書体名の設定と取得を行います。	
<b>ImpressCountFontName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.ImpressCountFontName = "書体名"          取得:現在設定されている書体名 = オブジェクト.ImpressCountFontName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '捺印カウンタの書体名の取得 msgbox objDstmp32.ImpressCountFontName '捺印カウンタの書体斜体の取得 msgbox objDstmp32.ImpressCountFontItalic '捺印カウンタの書体強調の取得 msgbox objDstmp32.ImpressCountFontBold '捺印カウンタの書体サイズの取得 msgbox objDstmp32.ImpressCountFontSize Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressCountFontSize	
印面の下部に表示される捺印カウンタの文字サイズの設定と取得を行います。	
<b>ImpressCountFontSize = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.ImpressCountFontSize = 1          取得:現在の設定値 = オブジェクト.ImpressCountFontSize</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.ShowDialog() '捺印カウンタの書体名の取得 msgbox objDstmp32.ImpressCountFontName '捺印カウンタの書体斜体の取得 msgbox objDstmp32.ImpressCountFontItalic '捺印カウンタの書体強調の取得 msgbox objDstmp32.ImpressCountFontBold '捺印カウンタの書体サイズの取得 msgbox objDstmp32.ImpressCountFontSize Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressCountLogFlag	
印鑑データの捺印カウンタに対応する捺印ログの収集状態を取得します。	
<b>ImpressCountLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.ImpressCountLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '捺印カウンタのログ収集状態の取得 if objDstamp32.ImpressCountLogFlag = True then     msgbox objDstamp32.ImpressCount end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressCountStatus	
印面の下部に表示される捺印カウンタの表示と表示色の設定と取得を行います。	
<b>ImpressCountStatus = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:非表示 1:黒色で表示 2:カラー（印面と同色）で表示</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.ImpressCountStatus = 1 取得:現在設定されている捺印カウンタの表示と色 = オブジェクト.ImpressCountStatus</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '捺印カウンタのステータスを取得 msgbox objDstamp32.ImpressCountStatus '捺印カウンタの取得 msgbox objDstamp32.ImpressCount Set objDstamp32 = Nothing</pre>	



捺印ツール Extension > IDstmp32	プロパティ
ImpressID	
印面の下部に表示される捺印 ID の設定と取得を行います。	
<b>ImpressID = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による取得が行えます。</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは利用できません。 このメソッドは取得専用です。</p>	
<p><b>使用例</b> 取得: 現在設定されている捺印 ID = オブジェクト.ImpressID</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '捺印 ID のステータスを取得 msgbox objDstamp32.ImpressIDStatus '捺印 ID を取得 msgbox objDstamp32.ImpressID Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressIDFontBold	
印面の下部に表示される捺印 ID の強調スタイルの設定と取得を行います。	
<b>ImpressIDFontBold = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:強調スタイルにしない 1:強調スタイルにする</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b> 設定:オブジェクト.ImpressIDFontBold = 1 取得:現在の設定値 = オブジェクト.ImpressIDFontBold</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '捺印 ID の書体名の取得 msgbox objDstamp32.ImpressIDFontName '捺印 ID の書体斜体の取得 msgbox objDstamp32.ImpressIDFontItalic '捺印 ID の書体強調の取得 msgbox objDstamp32.ImpressIDFontBold '捺印 ID の書体サイズの取得 msgbox objDstamp32.ImpressIDFontSize Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressIDFontItalic	
印面の下部に表示される捺印 ID の斜体スタイルの設定と取得を行います。	
<b>ImpressIDFontItalic = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:斜体スタイルにしない 1:斜体スタイルにする</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b> 設定:オブジェクト.ImpressIDFontItalic = 1 取得:現在の設定値 = オブジェクト.ImpressIDFontItalic</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '捺印 ID の書体名の取得 msgbox objDstamp32.ImpressIDFontName '捺印 ID の書体斜体の取得 msgbox objDstamp32.ImpressIDFontItalic '捺印 ID の書体強調の取得 msgbox objDstamp32.ImpressIDFontBold '捺印 ID の書体サイズの取得 msgbox objDstamp32.ImpressIDFontSize Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressIDFontName	
印面の下部に表示される捺印 ID の書体名の設定と取得を行います。	
<b>ImpressIDFontName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b>            設定:オブジェクト.ImpressIDFontName = "書体名"            取得:現在設定されている書体名 = オブジェクト.ImpressIDFontName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '捺印 ID の書体名の取得 msgbox objDstamp32.ImpressIDFontName '捺印 ID の書体斜体の取得 msgbox objDstamp32.ImpressIDFontItalic '捺印 ID の書体強調の取得 msgbox objDstamp32.ImpressIDFontBold '捺印 ID の書体サイズの取得 msgbox objDstamp32.ImpressIDFontSize Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressIDFontSize	
印面の下部に表示される捺印 ID の文字サイズの設定と取得を行います。	
<b>ImpressIDFontSize = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b>            設定:オブジェクト.ImpressIDFontSize = 1            取得:現在の設定値 = オブジェクト.ImpressIDFontSize</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '捺印 ID の書体名の取得 msgbox objDstamp32.ImpressIDFontName '捺印 ID の書体斜体の取得 msgbox objDstamp32.ImpressIDFontItalic '捺印 ID の書体強調の取得 msgbox objDstamp32.ImpressIDFontBold '捺印 ID の書体サイズの取得 msgbox objDstamp32.ImpressIDFontSize Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressIDLogFlag	
印鑑データの捺印 ID に対応する捺印ログの収集状態を取得します。	
<b>ImpressIDLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による取得が行えます。 0: ログ収集なし 1: ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b> 取得: 現在設定されているログ収集状態 = オブジェクト.ImpressIDLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '捺印 ID のログ収集状態の取得 if objDstamp32.ImpressIDLogFlag = True then     msgbox objDstamp32.ImpressID end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>ImpressIDStatus</b>	
印面の下部に表示される捺印 ID の表示と表示色の設定と取得を行います。	
<b>ImpressIDStatus = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による設定・取得が行えます。 0: 非表示 1: 黒色で表示 2: カラー（印面と同色）で表示</p> <p><b>備考</b> このメソッドは下位互換性のために残されています。現在のバージョンでは利用できません。</p>	
<p><b>使用例</b> 設定: オブジェクト.ImpressIDStatus = 1 取得: 現在設定されている捺印 ID の表示と色 = オブジェクト.ImpressIDStatus</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '捺印 ID のステータスを取得 msgbox objDstamp32.ImpressIDStatus '捺印 ID を取得 msgbox objDstamp32.ImpressID Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
ImpressPropertyDialogCaption	
印面の設定を行うダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>ImpressPropertyDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"印面設定"が表示されます。</p>	
<p><b>使用例</b>            設定:オブジェクト.ImpressPropertyDialogCaption = "キャプション文字列"            取得:"キャプション文字列" = オブジェクト.ImpressPropertyDialogCaption</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印面設定ダイアログのタイトル文字列を変更 objDstamp32.ImpressPropertyDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstamp32.ImpressPropertyDialog() Set objDstamp32 = Nothing </pre>	



捺印ツール Extension > IDstmp32	プロパティ
<b>ImpressTime</b>	
Copy メソッドで引数に TRUE を指定して実行した際の印影オブジェクトに保存された捺印日時を取得します。Copy メソッドを実行した直後以外は本プロパティの値は不定となります。	
<b>ImpressTime = DATE</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> DATE: 日付型による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得: 捺印日時 = オブジェクト.ImpressTime</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() if objDstamp32.Copy(True) = True then     '捺印時間の取得     msgbox objDstamp32.ImpressTime end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
LoginName	
捺印する環境設定のログインユーザ名に関する情報の取得を行います。	
<b>LoginName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"コンピュータ名" = オブジェクト.LoginName</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'コンピュータへのログイン名のログ収集状態の取得 if objDstamp32.LoginNameLogFlag = True then     msgbox objDstamp32.LoginName end if Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
LoginNameLogFlag	
コンピュータのログイン名に対応する捺印ログの収集状態を取得します。	
<b>LoginNameLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.LoginNameLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() 'コンピュータへのログイン名のログ収集状態の取得 if objDstamp32.LoginNameLogFlag = True then     msgbox objDstamp32.LoginName end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>NotesFlag</b>	
捺印ツールの動作の設定を指定します。LotusScript で使用する場合には、1 (TRUE) を設定します。既定値は 0 (FALSE) になります。	
<b>NotesFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による設定・取得が行えます。 0: LotusScript で利用しない 1: LotusScript で利用する</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定: オブジェクト.NotesFlag = 1 取得: 追加印鑑数 = オブジェクト.NotesFlag</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
OptionDialogCaption	
印面設定を行うダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>OptionDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティは旧バージョンとの互換性のために用意されています。StampFaceDialogCaption プロパティを利用してください。</p>	
<p><b>使用例</b>          設定:オブジェクト.OptionDialogCaption = "キャプション文字列"          取得:"キャプション文字列" = オブジェクト.OptionDialogCaption</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.OptionDialogCaption = "&lt;変更するダイアログボックスのタイトル文字列&gt;" objDstamp32.OptionDialog '選択されている印面 Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>PaletteColor</b>	
印影を描画する際にパレットにカラーを指定して描画するかどうかの設定・取得を行います。設定した値は、印鑑毎に保存され、次回利用時の初期値となります。範囲外を指定した場合、その設定は無効になり、既定値および以前の設定値が残ります。	
<b>PaletteColor = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:パレットにカラーを指定しないで描画する 1:パレットにカラーを指定して描画する</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.PaletteColor = 1 取得:現在の設定値 = オブジェクト.PaletteColor</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
Password	
ログインをしているユーザのパスワードの設定を行います。本メソッドは設定専用です。値の取得はできません。	
<b>Password = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定が行えます。</p> <p><b>備考</b> このメソッドは、旧バージョンの互換のために残されています。ユーザのパスワードを設定・取得するには、UserPassword プロパティを使用してください。</p>	
<p><b>使用例</b>          設定:オブジェクト.Password = "ユーザパスワード"          取得:"ユーザパスワード" = オブジェクト.Password</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.Password = "&lt;変更を行うパスワード&gt;" objDstamp32.PasswordDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstamp32.PasswordDialog() Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
PasswordDialogCaption	
ユーザパスワードの変更を入力する際に表示されるダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>PasswordDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"ユーザパスワードの変更"が表示されます。このメソッドは下位互換性のために残されています。UserPasswordDialogCaption を利用してください。</p>	
<p><b>使用例</b> 設定:オブジェクト.PasswordDialogCaption = "キャプション文字列" 取得:"キャプション文字列" = オブジェクト.PasswordDialogCaption</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.Password = "&lt;変更を行うパスワード&gt;" objDstamp32.PasswordDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstamp32.PasswordDialog() Set objDstamp32 = Nothing</pre>	



捺印ツール Extension > IDstmp32	プロパティ
QueryUpdate	
ドラッグアンドドロップでの捺印操作の後にパソコン決裁を終了するかどうかの設定・取得を行います。	
<b>QueryUpdate = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0:捺印確認ダイアログボックスを表示し、ユーザ選択により終了 1:捺印確認ダイアログボックスを表示せず、捺印を行い自動終了する 2:捺印確認ダイアログボックスを表示せず、捺印しないで自動終了する</p> <p><b>備考</b> 捺印ツールの終了時（Quit メソッド実行時または、メインウィンドウ表示時にメニューから終了を選択した場合）の動作ステータスを設定します。既定値として、1（捺印して終了）が設定されています。範囲外の値を設定した場合には、その設定は無効となり以前の設定値がそのまま残ります。このプロパティは、LotusNotes（LotusScript など）での利用を前提に用意されています。その他の環境での動作は不定になります。</p>	
<p><b>使用例</b> 設定:オブジェクト.QueryUpdate = 1 取得:現在の設定値 = オブジェクト.QueryUpdate</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
<b>Rop</b>	
<p>印影が描画される際のラスタオペレーションの設定・取得を行います。既定値では 12 (SRCCOPY) が設定されています。本プロパティを設定し、捺印を行った場合にその設定値が有効になり次回の使用時にはその値が既定値となります。範囲外を指定した場合、その設定は無効になり、既定値および以前の設定値が残ります。</p>	
<b>Rop = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。</p> <p>1 (オリジナル) :印影の文字や枠の部分のみを転送先にコピーする  2 (BLACKNESS) :物理パレットのインデックス 0 に対する色 (デフォルトは黒) でコピー先の矩形を塗りつぶします  3 (DSTINVERT) :コピー先の矩形の色を反転します  4 (MERGECOPY) :コピー元の色と、コピー先の色を論理 AND 演算子で結合します  5 (MERGEPAINT) :コピー元の色を反転した色と、コピー先の色を論理 OR 演算子で結合します  6 (NOTSRCCOPY) :コピー先の色を反</p> <p><b>備考</b></p>	
<p><b>使用例</b>  設定:オブジェクト.Rop = 12  取得:現在の設定値 = オブジェクト.Rop</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
RopCode	
印面を描画する際のラスターオペレーションコードの設定・取得を行います。	
<b>RopCode = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。</p> <p><b>備考</b> ラスターオペレーションコードとは、画像情報を描画する際の数値情報です。設定情報によっては描画されない可能性があるため設定には十分な注意を行ってください。</p>	
<p><b>使用例</b>            設定:オブジェクト.RopCode = &lt;ラスターオペレーション値&gt;            取得:現在設定されているラスターオペレーション値 = オブジェクト.RopCode</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
<b>SerializeLog</b>	
捺印時に印影オブジェクトに監査情報埋込を行うかどうかの状態を取得します。本プロパティは取得のみ可能です。	
<b>SerializeLog = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0 (FALSE) :捺印時に監査情報埋込を行わない 1 (TRUE) :捺印時に監査情報埋込を行う</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:現在選択されている値 = オブジェクト.SerializeLog</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '捺印ログの収集状態を取得 msgbox objDstamp32.WriteLog '監査情報埋込の状態を取得 msgbox objDstamp32.SerializeLog Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>SerializeStampBitmap</b>	
<p>印影を描画する際に再描画用のイメージをつけて捺印するかどうかの設定・取得を行います。再描画用の画像は、捺印した印影のプロパティを表示したり、アプリケーション上で捺印された印影を表示する際に必要となり、再描画用の画像がない場合にはアプリケーションによっては、印影が正常に表示されない場合がありますので。設定には十分な注意が必要です。また設定した値は、印鑑毎に保存され、次回利用時の初期値となります。</p>	
<b>SerializeStampBitmap = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0(FALSE):再描画用の画像をつけないで描画する 1 (TRUE):再描画用の画像をつけて描画する</p> <p><b>備考</b> 範囲外の値を設定した場合には無効となり、既定値または設定以前の設定が残ります。</p>	
<p><b>使用例</b>          設定:オブジェクト.SerializeStampBitmap = 1          取得:現在設定されている値 = オブジェクト.SerializeStampBitmap</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
StampCID	
印鑑クリエイト ID を取得します。クリエイト ID は数値型として取得されますが、取得される ID は管理ツールの印鑑データのプロパティで 16 進数で表示確認できます。	
<b>StampCID = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:設定値 = オブジェクト.StampCID</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '印鑑 CID を取得 msgbox objDstamp32.StampCID 'ファイル CID を取得 msgbox objDstamp32.FileCID 'ユーザ名を取得 msgbox objDstamp32.UserName 'ユーザ CID を取得 msgbox objDstamp32.UserCID 'ユーザの姓を取得 msgbox objDstamp32.UserLastName 'ユーザの名を取得 msgbox objDstamp32.UserFirstName 'ユーザのミドルネームを取得 msgbox objDstamp32.UserMiddleName Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
StampCIDLogFlag	
印鑑データのクリエイト ID に対応する捺印ログの収集状態を取得します。	
<b>StampCIDLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:ログ収集なし 1:ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得:現在設定されているログ収集状態 = オブジェクト.StampCIDLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑 CID のログ収集状態の取得 if objDstamp32.StampCIDLogFlag = True then     msgbox objDstamp32.StampCID end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
StampColor	
印面の表示色の設定と取得を行います。	
<b>StampColor = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 1:黒色で表示 2:カラー（印面と同色）で表示</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.StampColor = 1 取得:現在設定されている印面色 = オブジェクト.StampColor</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '印面色を取得 msgbox objDstmp32.StampColor Set objDstmp32 = Nothing</pre>	



捺印ツール Extension > IDstmp32	プロパティ
<b>StampCount</b>	
ログインしているユーザに追加されている印鑑データの数を取得します。	
<b>StampCount = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得: 追加印鑑数 = オブジェクト.StampCount</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.WindowCaption = "&lt;印鑑選択ウィンドウのタイトルに表示する文字列&gt;" nResult = objDstamp32.UserDialog '追加されている印鑑数の取得 msgbox objDstamp32.StampCount Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
StampFaceDialogCaption	
印面設定を行うダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>StampFaceDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"印面設定"が表示されます。</p>	
<p><b>使用例</b>          設定:オブジェクト.StampFaceDialogCaption = "キャプション文字列"          取得:"キャプション文字列" = オブジェクト.StampFaceDialogCaption</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.StampFaceDialogCaption = "&lt;変更するダイアログボックスのタイトル文字列&gt;" objDstamp32.StampFaceDialog '選択されている印面 Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
StampSerialNumber	
捺印する印鑑データのシリアル番号に関する情報の取得を行います。	
<b>StampSerialNumber = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"印鑑シリアル番号" = オブジェクト.StampSerialNumber</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑シリアル番号のログ収集状態の取得 if objDstamp32.StampSerialNumberLogFlag = True then     msgbox objDstamp32.StampSerialNumber end if Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
StampSerialNumberLogFlag	
印鑑データのシリアル番号に対応する捺印ログの収集状態を取得します。	
<b>StampSerialNumberLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による取得が行えます。 0: ログ収集なし 1: ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得: 現在設定されているログ収集状態 = オブジェクト.StampSerialNumberLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑シリアル番号のログ収集状態の取得 if objDstamp32.StampSerialNumberLogFlag = True then     msgbox objDstamp32.StampSerialNumber end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
StampType	
印面の種類（日付印と氏名印）の種類を取得します。	
<b>StampType = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 1:氏名印 2:日付印（印面日付が中央部に配置される種類）</p> <p><b>備考</b> 本メソッドは取得専用です。設定はできません。</p>	
<p><b>使用例</b> 取得:印面の種類 = オブジェクト.StampType</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '印鑑の種類を取得 msgbox objDstmp32.StampType Set objDstmp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>StampVisible</b>	
引数で指定された登録インデックスの印鑑データのメインウィンドウへの表示状態について設定・取得を行います。	
<b>StampVisible(long nIndex) = long</b>	
<p><b>パラメータ</b>  nIndex:設定を行う印鑑データの登録インデックス</p> <p><b>戻り値</b>  long:数値による設定・取得が行えます。  0:引数で指定された登録インデックスの印面を表示しない  1:引数で指定された登録インデックスの印面を表示する</p> <p><b>備考</b>  登録インデックスは、管理ツールで印鑑データのプロパティを表示した際に確認することができます。本プロパティは、StampVisiblePosition プロパティと連動しています。本プロパティで表示・非表示の設定を行った場合に、それに対応した印影の位置がStampVisiblePosition プロパティに設定されます。</p>	
<p><b>使用例</b>  設定:オブジェクト.StampVisible (0) = 1  取得:表示状態 = オブジェクト.StampVisible (0)</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.WindowCaption = "&lt;印鑑選択ウィンドウのタイトルに表示する文字列&gt;" nResult = objDstamp32.UserDialog '追加されている印鑑の表示設定を取得 msgbox objDstamp32.StampVisible(0) Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>StampVisiblePosition</b>	
引数で指定された表示位置の印鑑データのメインウィンドウへの表示状態について設定・取得を行います。	
<b>StampVisiblePosition(long nPosition) = long</b>	
<p><b>パラメータ</b>  nPosition:設定を行う印鑑データの表示位置</p> <p><b>戻り値</b>  long:数値による設定・取得が行えます。  0:引数で指定された表示位置の印面を表示しない  1:引数で指定された表示位置の印面を表示する</p> <p><b>備考</b>  表示位置は「表示設定」ダイアログボックスで「表示する印鑑」のリストに表示される順序になり、上から 0,1,2,3 という値になります。  本プロパティは StampVisible プロパティと連動しています。本プロパティで表示・非表示の設定を行った場合に、それに対応した印影の位置が StampVisible プロパティに設定されます。</p>	
<p><b>使用例</b>  設定:オブジェクト.StampVisiblePosition (0) = 1  取得:表示状態 = オブジェクト.StampVisiblePosition (0)</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.WindowCaption = "&lt;印鑑選択ウィンドウのタイトルに表示する文字列&gt;" nResult = objDstamp32.UserDialog '追加されている印鑑の表示設定を取得 msgbox objDstamp32.StampVisible(0) Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>StretchBltMode</b>	
印面を描画する際に使用するストレッチングモードの設定・取得を行います。	
<b>StretchBltMode = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。</p> <p>1 (STRETCH_ANDSCANS) :ビット毎の AND 演算子を使用して消去するラインを残すラインと結合する  2 (STRETCH_ORSCANS) :ビット毎の OR 演算子を使用して消去するラインを残すラインと結合する  3 (STRETCH_DELETESCANS) :論理演算を行わずに単純に削除する。この値を指定した場合にはラインの色がそのまま残ります。  4 (STRETCH_HALFTONE) :ビットの平均カラー値を取る</p> <p><b>備考</b> 範囲外の値を設定した場合には無効となり、既定値または設定以前の設定が残ります。</p> <p><b>使用例</b>  設定:オブジェクト.StretchBltMode = &lt;ラスターオペレーション値&gt;  取得:現在設定されているラスターオペレーション値 = オブジェクト.StretchBltMode</p>	
<b>サンプルコード</b>	



捺印ツール Extension > IDstmp32	プロパティ
UnlessDocInfoImpressStatus	
文書情報の状態による捺印操作の抑制ステータスを取得します。	
<b>UnlessDocInfoImpressStatus = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0:文書情報が入力されていない状態でも捺印可能 1:文書情報が入力されていない場合には警告表示 2:文書情報が入力されていない場合には捺印不可能</p> <p><b>備考</b></p>	
<p><b>使用例</b> 設定:オブジェクト.UnlessDocInfoImpressStatus = 1 取得:現在設定されている捺印抑制ステータス = オブジェクト.UnlessDocInfoImpressStatus</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '捺印抑制ステータスの取得 msgbox objDstamp32.UnlessDocInfoImpressStatus Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>UserCID</b>	
ユーザクリエイト ID を取得します。クリエイト ID は数値型として取得されますが、取得される ID は管理ツールのユーザのプロパティで 16 進数で表示確認できます。	
<b>UserCID = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:設定値 = オブジェクト.UserCID</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '印鑑 CID を取得 msgbox objDstmp32.StampCID 'ファイル CID を取得 msgbox objDstmp32.FileCID 'ユーザ名を取得 msgbox objDstmp32.UserName 'ユーザ CID を取得 msgbox objDstmp32.UserCID 'ユーザの姓を取得 msgbox objDstmp32.UserLastName 'ユーザの名を取得 msgbox objDstmp32.UserFirstName 'ユーザのミドルネームを取得 msgbox objDstmp32.UserMiddleName Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>UserCIDLogFlag</b>	
ユーザのクリエイト ID に対応する捺印ログの収集状態を取得します。	
<b>UserCIDLogFlag = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long: 数値による取得が行えます。 0: ログ収集なし 1: ログ収集あり</p> <p><b>備考</b> このメソッドは取得専用です。値の設定はできません。</p>	
<p><b>使用例</b> 取得: 現在設定されているログ収集状態 = オブジェクト.UserCIDLogFlag</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'ユーザ CID のログ収集状態の取得 if objDstamp32.UserCIDLogFlag = True then     msgbox objDstamp32.UserCID end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
UserDialogCaption	
ユーザ名とパスワードを入力するダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>UserDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"パソコン決裁 ログイン"が表示されます。</p>	
<p><b>使用例</b>            設定:オブジェクト.UserDialogCaption = "キャプション文字列"            取得:"キャプション文字列" = オブジェクト.UserDialogCaption</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.UserDialogCaption = "&lt;ログインダイアログボックスのタイトルに表示する文字列&gt;" objDstamp32.UserDialog Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
UserFirstName	
ログインをしているユーザの名の取得を行います。	
<b>UserFirstName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"名" = オブジェクト.UserFirstName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.ShowDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '印鑑 CID を取得 msgbox objDstmp32.StampCID 'ファイル CID を取得 msgbox objDstmp32.FileCID 'ユーザ名を取得 msgbox objDstmp32.UserName 'ユーザ CID を取得 msgbox objDstmp32.UserCID 'ユーザの姓を取得 msgbox objDstmp32.UserLastName 'ユーザの名を取得 msgbox objDstmp32.UserFirstName 'ユーザのミドルネームを取得 msgbox objDstmp32.UserMiddleName Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
UserLastName	
ログインをしているユーザの姓の取得を行います。	
<b>UserLastName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"姓" = オブジェクト.UserLastName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '印鑑 CID を取得 msgbox objDstmp32.StampCID 'ファイル CID を取得 msgbox objDstmp32.FileCID 'ユーザ名を取得 msgbox objDstmp32.UserName 'ユーザ CID を取得 msgbox objDstmp32.UserCID 'ユーザの姓を取得 msgbox objDstmp32.UserLastName 'ユーザの名を取得 msgbox objDstmp32.UserFirstName 'ユーザのミドルネームを取得 msgbox objDstmp32.UserMiddleName Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
UserMiddleName	
ログインをしているユーザのミドルネームの取得を行います。	
<b>UserMiddleName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"名" = オブジェクト.UserMiddleName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '印鑑 CID を取得 msgbox objDstmp32.StampCID 'ファイル CID を取得 msgbox objDstmp32.FileCID 'ユーザ名を取得 msgbox objDstmp32.UserName 'ユーザ CID を取得 msgbox objDstmp32.UserCID 'ユーザの姓を取得 msgbox objDstmp32.UserLastName 'ユーザの名を取得 msgbox objDstmp32.UserFirstName 'ユーザのミドルネームを取得 msgbox objDstmp32.UserMiddleName Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
UserName	
ログインをしているユーザのユーザ名の取得を行います。	
<b>UserName = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:"ユーザ名" = オブジェクト.UserName</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '印鑑 CID を取得 msgbox objDstmp32.StampCID 'ファイル CID を取得 msgbox objDstmp32.FileCID 'ユーザ名を取得 msgbox objDstmp32.UserName 'ユーザ CID を取得 msgbox objDstmp32.UserCID 'ユーザの姓を取得 msgbox objDstmp32.UserLastName 'ユーザの名を取得 msgbox objDstmp32.UserFirstName 'ユーザのミドルネームを取得 msgbox objDstmp32.UserMiddleName Set objDstmp32 = Nothing </pre>	



捺印ツール Extension > IDstmp32	プロパティ
UserPassword	
ログインをしているユーザのパスワードの設定を行います。本メソッドは設定専用です。値の取得はできません。	
<b>UserPassword = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.UserPassword = "ユーザパスワード"          取得:"ユーザパスワード" = オブジェクト.UserPassword</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True objDstmp32.UserPassword = "&lt;変更を行うパスワード&gt;" objDstmp32.UserPasswordDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstmp32.UserPasswordDialog() Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
UserPasswordDialogCaption	
ユーザパスワードの変更を入力する際に表示されるダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>UserPasswordDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"ユーザパスワードの変更"が表示されます。このメソッドは下位互換性のために残されています。</p>	
<p><b>使用例</b>            設定:オブジェクト.UserPasswordDialogCaption = "キャプション文字列"            取得:"キャプション文字列" = オブジェクト.UserPasswordDialogCaption</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.UserPassword = "&lt;変更を行うパスワード&gt;" objDstamp32.UserPasswordDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstamp32.UserPasswordDialog() Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>UserPasswordMinLength</b>	
ユーザのパスワードに必要な最小の長さを取得します。取得される値の設定は管理ツールのユーザのプロパティで行います。	
<b>UserPasswordMinLength = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:現在設定されている値 = オブジェクト.UserPasswordMinLength</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True 'ユーザパスワード最少文字数の取得 msgbox objDstmp32.UserPasswordMinLength Set objDstmp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
WindowCaption	
印鑑選択ウィンドウ（捺印ツールでログイン後に表示されるウィンドウ）のキャプションに表示される文字列の設定・取得を行います。	
<b>WindowCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"パソコン決裁 &lt;ログインユーザ名&gt;"が表示されます。</p>	
<p><b>使用例</b>          設定:オブジェクト.WindowCaption = "キャプション文字列"          取得:"キャプション文字列" = オブジェクト.WindowCaption</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.WindowCaption = "&lt;印鑑選択ウィンドウのタイトルに表示する文字列&gt;" nResult = objDstamp32.UserDialog objDstamp32.WindowVisible = True msgbox "オブジェクトが終了するのを防ぐためのメッセージ" Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
WindowDialogCaption	
表示設定の変更を行う際に表示されるダイアログボックスのキャプションに表示される文字列の設定・取得を行います。	
<b>WindowDialogCaption = string</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> string:文字列による設定・取得が行えます。</p> <p><b>備考</b> このプロパティが設定されていない場合には、既定値として"表示設定"が表示されます。このメソッドは下位互換性のために残されています。現在のバージョンでは廃止されました。</p>	
<p><b>使用例</b> 設定:オブジェクト.WindowDialogCaption = "キャプション文字列" 取得:"キャプション文字列" = オブジェクト.WindowDialogCaption</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	プロパティ
WindowLeft	
印鑑選択ウィンドウ（捺印ツールでログイン後に表示されるウィンドウ）が表示される際の左上の横座標の設定・取得を行います。	
<b>WindowLeft = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による値の設定・取得が行えます。</p> <p><b>備考</b> 印鑑選択ウィンドウの位置がログインユーザ毎に保存される設計になっています。そのため、本メソッドを利用して設定・取得を行う前に UserDialog メソッドなどでログインを行っておく必要があります。ログインを行う前に取得される値は不定となります。</p>	
<p><b>使用例</b> 設定:オブジェクト.WindowLeft = 印鑑選択ウィンドウ左上の横座標 取得:印鑑選択ウィンドウ左上の横座標 = オブジェクト.WindowLeft</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.UserDialog msgbox objDstamp32.WindowTop msgbox objDstamp32.WindowLeft Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
WindowTop	
印鑑選択ウィンドウ（捺印ツールでログイン後に表示されるウィンドウ）が表示される際の左上の縦座標の設定・取得を行います。	
<b>WindowTop = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による値の設定・取得が行えます。</p> <p><b>備考</b> 印鑑選択ウィンドウの位置がログインユーザ毎に保存される設計になっています。そのため、本メソッドを利用して設定・取得を行う前に UserDialog メソッドなどでログインを行っておく必要があります。ログインを行う前に取得される値は不定となります。</p>	
<p><b>使用例</b> 設定:オブジェクト.WindowTop = 印鑑選択ウィンドウ左上の縦座標 取得:印鑑選択ウィンドウ左上の縦座標 = オブジェクト.WindowTop</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.UserDialog msgbox objDstamp32.WindowTop msgbox objDstamp32.WindowLeft Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
<b>WindowTopMost</b>	
メインウィンドウやダイアログボックス関連の表示を最前面で表示するかどうかを設定・取得を行います。	
<b>WindowTopMost = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 0 (FALSE) :メインウィンドウおよびダイアログボックスを最前面に表示しない 1 (TRUE) :メインウィンドウおよびダイアログボックスを最前面に表示する</p> <p><b>備考</b> 印鑑選択ウィンドウの表示設定がログインユーザ毎に保存される設計になっています。そのため、本メソッドを利用して設定・取得を行う前に UserDialog メソッドなどでログインを行っておく必要があります。ログインを行う前に取得される値は不定となります。</p>	
<p><b>使用例</b> 設定:オブジェクト.WindowTopMost = 1 取得:現在選択されている値 = オブジェクト.WindowTopMost</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '常に手前に表示の取得と設定 msgbox objDstamp32.WindowTopMost Set objDstamp32 = Nothing</pre>	



捺印ツール Extension > IDstmp32	プロパティ
<b>WindowVisible</b>	
印鑑選択ウィンドウ（捺印ツールでログイン後に表示されるウィンドウ）の可視状態を設定・取得を行います。既定値は非表示状態 (FALSE) になっています。	
<b>WindowVisible = Boolean</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> Boolean 型による設定・取得が行えます。 TRUE:表示状態 FALSE:非表示状態（既定値）</p> <p><b>備考</b></p>	
<p><b>使用例</b>          設定:オブジェクト.WindowVisible = 可視状態の設定値          取得:現在の可視状態 = オブジェクト.WindowVisible</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.WindowCaption = "&lt;印鑑選択ウィンドウのタイトルに表示する文字列&gt;" nResult = objDstamp32.UserDialog objDstamp32.WindowVisible = True msgbox "オブジェクトが終了するのを防ぐためのメッセージ" Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
WriteLog	
捺印時にユーザ名や捺印日時などのプロパティ情報を捺印ログとして収集状態を取得します。	
<b>WriteLog = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による取得が行えます。 0 (FALSE) :捺印時にログ収集を行わない 1 (TRUE) :捺印時にログ収集を行う</p> <p><b>備考</b></p>	
<p><b>使用例</b> 取得:現在選択されている値 = オブジェクト.WriteLog</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstmp32 = CreateObject("Dstmp.StampObject.6") 'objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True '捺印ログの収集状態を取得 msgbox objDstmp32.WriteLog '監査情報埋込の状態を取得 msgbox objDstmp32.SerializeLog Set objDstmp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	プロパティ
Zoom	
メインウィンドウ上で表示される印面の表示倍率の設定・取得を行います。	
<b>Zoom = long</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> long:数値による設定・取得が行えます。 10~400:メインウィンドウで表示される印面の表示率</p> <p><b>備考</b> 範囲外の値を設定した場合には、その設定は無効になり、以前の値が初期値として反映されます。</p>	
<p><b>使用例</b> 設定:オブジェクト.Zoom = 100 取得:現在選択されている値 = オブジェクト.Zoom</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '表示倍率を取得 msgbox objDstamp32.Zoom Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	メソッド
<b>AppendAttachmentFile</b>	
引数で指定された場所で追加されている添付ファイルを削除します。	
<b>BOOL AppendAttachmentFile(string strFilePath)</b>	
<p><b>パラメータ</b> strFilePath:追加を行う添付ファイルの場所</p> <p><b>戻り値</b> BOOL:数値（または BOOL 値）で取得が行えます 0（FALSE）:追加失敗 1（TRUE）:追加成功</p> <p><b>備考</b> 追加を行う前に、GetUser メソッドなどでユーザの取得を行い、ActiveStamp などのプロパティで対象印鑑データをアクティブにしておく必要があります。</p>	
<p><b>使用例</b> nResult = オブジェクト.AppendAttachmentFile（）</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.AppendAttachmentFile("&lt;追加するファイルの場所&gt;") for nLoop = 0 to objDstamp32.AttachmentFileCount     msgbox objDstamp32.AttachmentFileName(nLoop)     objDstamp32.DeleteAttachmentFile(nLoop) next Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	メソッド
CommentDialog	
コメントと添付ファイルの設定を行う「コメント」ダイアログボックスを表示します。	
<b>BOOL CommentDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>          BOOL:数値（または BOOL 値）で取得が行えます          0（FALSE）:コメントと添付ファイルを行うダイアログボックスで [キャンセル] が選択された          1（TRUE）:コメントと添付ファイルを行うダイアログボックスで [OK] が選択された</p> <p><b>備考</b>          ダイアログボックスのタイトルを変更するには、CommentDialogCaption プロパティを使います。このメソッドは以前のバージョンとの互換性のために用意されています。ImpressPropertyDialog メソッドを利用してください。</p>	
<p><b>使用例</b>          nResult = オブジェクト.CommentDialog（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'コメントダイアログのタイトル文字列を変更 objDstamp32.CommentDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstamp32.CommentDialog() Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	メソッド
Copy	
選択されている印面を印影オブジェクトとしてクリップボードに転送します。	
<b>BOOL Copy(boolean bUpdate)</b>	
<p><b>パラメータ</b>  bUpdate:  false:実行時にログ収集を行いません  true:実行時にログ収集を行います</p> <p><b>戻り値</b>  BOOL:数値（または BOOL 値）で取得が行えます  0（FALSE）:失敗  1（TRUE）:成功</p> <p><b>備考</b>  印鑑データのログ収集に関するプロパティが設定されていない場合には、引数で指定してもログ収集は行われません。</p>	
<p><b>使用例</b>  bResult = オブジェクト.ResetStampProperties（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑データをクリップボードにコピー objDstamp32.Copy(True) Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	メソッド
<b>DeleteAttachmentFile</b>	
引数で指定された場所で追加されている添付ファイルを削除します。	
<b>BOOL DeleteAttachmentFile(string strFilePath)</b>	
<p><b>パラメータ</b> strFilePath: 削除を行う添付ファイルの場所</p> <p><b>戻り値</b> BOOL: 数値（または BOOL 値）で取得が行えます 0（FALSE）: 削除失敗 1（TRUE）: 削除成功</p> <p><b>備考</b> 削除を行う前に、GetUser メソッドなどでユーザの取得を行い、ActiveStamp などのプロパティで対象印鑑データをアクティブにしておく必要があります。</p>	
<p><b>使用例</b> nResult = オブジェクト.DeleteAttachmentFile（）</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.AppendAttachmentFile("&lt;追加するファイルの場所&gt;") for nLoop = 0 to objDstamp32.AttachmentFileCount     msgbox objDstamp32.AttachmentFileName(nLoop)     objDstamp32.DeleteAttachmentFile(nLoop) next Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	メソッド
<b>DocInfoDialog</b>	
文書情報を設定するダイアログボックスを表示します。ダイアログボックスのウィンドウタイトルを変更するには、DocInfoDialogCaption プロパティを使います。このメソッドは以前のバージョンとの互換性のために用意されています。ImpressPropertyDialog メソッドを利用します。	
<b>BOOL DocInfoDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>          BOOL:数値（または BOOL 値）で取得が行えます          0（FALSE）:文書情報の設定を行うダイアログボックスで［キャンセル］が選択された          1（TRUE）:文書情報の設定を行うダイアログボックスで［OK］が選択された</p> <p><b>備考</b></p>	
<p><b>使用例</b>          nResult = オブジェクト.DocInfoDialog（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.DocInfoDialogCaption = "&lt;変更を行うタイトル&gt;" objDstamp32.DocInfoDialog '文書ファイル名のログ収集状態の取得 if objDstamp32.DocumentFileNameLogFlag = True then     msgbox objDstamp32.DocumentFileName end if '文書タイトルのログ収集状態の取得 if objDstamp32.DocumentTitleLogFlag = True then     msgbox objDstamp32.DocumentTitle end if '文書番号のログ収集状態の取得 if objDstamp32.DocumentNumberLogFlag = True then     msgbox objDstamp32.DocumentNumber end if '承認項目のログ収集状態の取得 if objDstamp32.DocumentItemLogFlag(1) = True then     msgbox objDstamp32.DocumentItem(1) end if Set objDstamp32 = Nothing         </pre>	



捺印ツール Extension > IDstmp32	メソッド
FileDialog	
捺印用印鑑データファイルの参照を行う「印鑑データ指定」ダイアログボックスを表示します。	
<b>BOOL FileDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>          BOOL:数値（または BOOL 値）で取得が行えます          0（FALSE）:印鑑データ指定を行うダイアログボックスで［キャンセル］が選択された          1（TRUE）:印鑑データ指定を行うダイアログボックスで［開く］が選択された</p> <p><b>備考</b>          開かれるダイアログボックスのウィンドウタイトルを変更するには、FileDialogCaption プロパティを使います。このメソッドは以前のバージョンとの互換性のために用意されています。</p>	
<p><b>使用例</b>          nResult = オブジェクト.FileDialog（）</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	メソッド
GetStamp	
引数で指定されたユーザ名とパスワードを使って捺印用印鑑データファイル内のユーザを取得します。	
<b>BOOL GetStamp (string strUserName, string strUserPassword)</b>	
<p><b>パラメータ</b>  strUserName:取得するユーザのユーザ名を指定します  strPassword:取得するユーザのパスワードを指定します</p> <p><b>戻り値</b>  BOOL:数値（または BOOL 値）で取得が行えます  0 (FALSE) :失敗（ログイン失敗）  1 (TRUE) :成功（ログイン成功）</p> <p><b>備考</b>  このメソッドは以前のバージョンとの互換性のために用意されています。GetUser メソッドを利用してください。</p>	
<p><b>使用例</b>  bResult = オブジェクト.GetStamp ("ユーザ名", "パスワード")</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ダイアログボックスを表示しないでログインする if objDstamp32.GetStamp("&lt;ユーザ名&gt;", "&lt;パスワード&gt;") = True then     'ユーザ取得成功 else     'ユーザ取得失敗 end if 'アプリケーションを終了させる objDstamp32.Quit() Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	メソッド
<b>GetStampByE9</b>	
インプレットデバイスから取得した ID に一致する印鑑データを所有するユーザを取得します。	
<b>BOOL GetStampByE9()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>          BOOL:数値（または BOOL 値）で取得が行えます          0（FALSE）:失敗          1（TRUE）:成功</p> <p><b>備考</b>          本メソッドを実行する場合にはあらかじめ、実行するコンピュータに、e-tablet などインプレットの ID を取得するのに必要な機器およびドライバがあらかじめ設定されている必要があります。このメソッドは旧バージョンとの互換のために用意されています。ユーザの取得を行う場合には、GetStampByInpplet メソッドを使用してください。</p>	
<p><b>使用例</b>          bResult = オブジェクト.GetStampByE9（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" インプレットでユーザ取得を行う if objDstamp32.GetStampByE9() == True then     'ユーザ取得成功 else     'ユーザ取得失敗 end if Set objDstamp32 = Nothing         </pre>	

捺印ツール Extension > IDstmp32	メソッド
<b>GetStampByInpplet</b>	
インプレットデバイスから取得した ID に一致する印鑑データを所有するユーザを取得します。	
<b>BOOL GetStampByInpplet()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0（FALSE）:失敗            1（TRUE）:成功</p> <p><b>備考</b>            本メソッドを実行する場合にはあらかじめ、実行するコンピュータに、e-tablet などインプレットの ID を取得するのに必要な機器およびドライバがあらかじめ設定されている必要があります。</p>	
<p><b>使用例</b>            bResult = オブジェクト.GetStampByInpplet（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'インプレットでユーザ取得を行う if objDstamp32.GetStampByInpplet() == True then     'ユーザ取得成功 else     'ユーザ取得失敗 end if Set objDstamp32 = Nothing           </pre>	

捺印ツール Extension > IDstmp32	メソッド
GetUser	
引数で指定されたユーザ名とパスワードを使って捺印用印鑑データファイル内のユーザを取得します。	
<b>BOOL GetUser (string strUserName, string strUserPassword)</b>	
<p><b>パラメータ</b></p> <p>strUserName:取得するユーザのユーザ名を指定します  strPassword:取得するユーザのパスワードを指定します</p> <p><b>戻り値</b></p> <p>BOOL:数値（または BOOL 値）で取得が行えます  0 (FALSE):失敗（ログイン失敗）  1 (TRUE):成功（ログイン成功）</p> <p><b>備考</b></p>	
<p><b>使用例</b></p> <p>bResult = オブジェクト.GetUser ("ユーザ名", "パスワード")</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ダイアログボックスを表示しないでログインする if objDstamp32.GetUser("&lt;ユーザ名&gt;", "&lt;パスワード&gt;") = True then     'ユーザ取得成功 else     'ユーザ取得失敗 end if 'アプリケーションを終了させる objDstamp32.Quit() Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	メソッド
ImpressPropertyDialog	
印面の設定を行う「印面設定」ダイアログボックスを表示します。	
<b>BOOL ImpressPropertyDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>          BOOL:数値（または BOOL 値）で取得が行えます          0（FALSE）:印面設定を行うダイアログボックスで［キャンセル］が選択された          1（TRUE）:印面設定を行うダイアログボックスで［OK］が選択された</p> <p><b>備考</b>          ダイアログボックスのタイトルを変更するには、ImpressPropertyDialogCaption プロパティを使います。</p>	
<p><b>使用例</b>          nResult = オブジェクト.ImpressPropertyDialog（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印面設定ダイアログのタイトル文字列を変更 objDstamp32.ImpressPropertyDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstamp32.ImpressPropertyDialog() Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	メソッド
LoadStampProperties	
SaveStampProperties メソッドでメモリ内に一時保存した印鑑データのプロパティ情報を読み込み選択中の印鑑データに適用します。	
<b>BOOL LoadStampProperties()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0 (FALSE):失敗            1 (TRUE):成功</p> <p><b>備考</b>            本メソッドを利用する前に SaveStampProperties メソッドでメモリにプロパティ情報を保存する必要があります。</p>	
<p><b>使用例</b>            bResult = オブジェクト.LoadStampProperties ( )</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") 'objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑データのプロパティ情報をメモリ内にコピー if objDstamp32.SaveStampProperties() = True then     ...印鑑データのプロパティ情報を変更する処理...     '変更したプロパティを変更前に戻す     objDstamp32.LoadStampProperties() end if Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	メソッド
OptionDialog	
印面の設定を行う「印面設定」ダイアログボックスを表示します。	
<b>BOOL OptionDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0（FALSE）:印面設定を行うダイアログボックスで [キャンセル] が選択された            1（TRUE）:印面設定を行うダイアログボックスで [OK] が選択された</p> <p><b>備考</b>            ダイアログボックスのタイトルを変更するには、OptionDialogCaption プロパティを使います。このメソッドは以前のバージョンとの互換性のために用意されています。ImpressPropertyDialog メソッドを利用してください。</p>	
<p><b>使用例</b>            nResult = オブジェクト.OptionDialog（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.OptionDialogCaption = "&lt;変更するダイアログボックスのタイトル文字列&gt;" objDstamp32.OptionDialog Set objDstamp32 = Nothing           </pre>	



捺印ツール Extension > IDstmp32	メソッド
<b>PasswordDialog</b>	
パスワードを設定するダイアログボックスを表示します。ダイアログボックスのタイトルを変更するには、PasswordDialogCaption プロパティを使います。このメソッドは以前のバージョンとの互換性のために用意されています。UserPasswordDialog メソッドを利用してください。	
<b>BOOL PasswordDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0 (FALSE) :パスワード設定行うダイアログボックスで [キャンセル] が選択された            1 (TRUE) :パスワード設定を行うダイアログボックスで [OK] が選択された</p> <p><b>備考</b></p>	
<p><b>使用例</b>            bResult = オブジェクト.PasswordDialog ( )</p>	
<b>サンプルコード</b>	

捺印ツール Extension > IDstmp32	メソッド
Quit	
クリップボードの内容をクリアして、プログラムを終了します。捺印を行って終了する場合、捺印カウント（ImpressCount プロパティ）がカウントアップされます。	
<b>void Quit ()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> ありません</p> <p><b>備考</b> アプリケーションの予約語などで、Quit メソッドが定義されている場合には Quit_メソッドを利用します。</p>	
<p><b>使用例</b> オブジェクト.Quit ()</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() 'アプリケーションを終了させる objDstamp32.Quit() Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	メソッド
Quit_	
クリップボードの内容をクリアして、プログラムを終了します。捺印を行って終了する場合、捺印カウント（ImpressCount プロパティ）がカウントアップされます。	
<b>void Quit_ ()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b> ありません</p> <p><b>備考</b> アプリケーションの予約語などで、Quit メソッドが定義されている場合に利用します。</p>	
<p><b>使用例</b> オブジェクト.Quit_ ()</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() 'アプリケーションを終了させる objDstmp32.Quit() Set objDstmp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	メソッド
<b>ResetFontStyles</b>	
設定されているフォント情報をすべて初期化します。	
<b>BOOL ResetFontStyles()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0（FALSE）:失敗            1（TRUE）:成功</p> <p><b>備考</b>            本メソッドは印面テキスト、印面日時、捺印カウンタの書体情報をすべて初期化（既定値）に戻します。</p>	
<p><b>使用例</b>            bResult = オブジェクト.ResetFontStyles（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True 'フォント情報の初期化 objDstmp32.ResetFontStyles() Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	メソッド
ResetStampProperties	
印鑑データのプロパティ値を初期値（既定値）に戻します。	
<b>BOOL ResetStampProperties()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0（FALSE）:失敗            1（TRUE）:成功</p> <p><b>備考</b>            本メソッドは印鑑データのすべてのプロパティ情報を新規登録時の初期値（既定値）に戻します。</p>	
<p><b>使用例</b>            bResult = オブジェクト.ResetStampProperties（）</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '印鑑データのプロパティ情報の初期化 objDstamp32.ResetStampProperties() Set objDstamp32 = Nothing</pre>	

捺印ツール Extension > IDstmp32	メソッド
SaveStampProperties	
印鑑データのプロパティ情報をメモリ内に一時保存します。	
<b>BOOL SaveStampProperties()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0 (FALSE) :失敗            1 (TRUE) :成功</p> <p><b>備考</b>            一時保存されたプロパティ情報は LoadStampProperties メソッドで読み込むことができます。</p>	
<p><b>使用例</b>            bResult = オブジェクト.SaveStampProperties ( )</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑データのプロパティ情報をメモリ内にコピー if objDstamp32.SaveStampProperties() = True then     ...印鑑データのプロパティ情報を変更する処理...     '変更したプロパティを変更前に戻す     objDstamp32.LoadStampProperties() end if Set objDstamp32 = Nothing           </pre>	

捺印ツール Extension > IDstmp32	メソッド
<b>StampFaceDialog</b>	
印面の設定を行う「印面設定」ダイアログボックスを表示します。	
<b>BOOL StampFaceDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0（FALSE）:印面設定を行うダイアログボックスで [キャンセル] が選択された            1（TRUE）:印面設定を行うダイアログボックスで [OK] が選択された</p> <p><b>備考</b>            ダイアログボックスのタイトルを変更するには、StampFaceDialogCaption プロパティを使います。このメソッドは以前のバージョンとの互換性のために用意されています。ImpressPropertyDialog メソッドを利用してください。</p>	
<p><b>使用例</b>            nResult = オブジェクト.StampFaceDialog（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True objDstamp32.StampFaceDialogCaption = "&lt;変更するダイアログボックスのタイトル文字列&gt;" objDstamp32.StampFaceDialog Set objDstamp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	メソッド
UserDialog	
パソコン決裁のログインを行うダイアログボックスを表示します。	
<b>BOOL UserDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>          BOOL:数値（または BOOL 値）で取得が行えます          0（FALSE）:ログインを行うダイアログボックスで［キャンセル］が選択された          1（TRUE）:ログインを行うダイアログボックスで［OK］が選択された</p> <p><b>備考</b></p>	
<p><b>使用例</b>          nResult = オブジェクト.UserDialog（）</p>	
<p><b>サンプルコード</b></p> <pre>Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.UserDialogCaption = "&lt;ログインダイアログボックスのタイトルに表示する文字列&gt;" objDstamp32.UserDialog Set objDstamp32 = Nothing</pre>	



捺印ツール Extension > IDstmp32	メソッド
<b>UserPasswordDialog</b>	
パスワードを設定するダイアログボックスを表示します。ダイアログボックスのタイトルを変更するには、UserPasswordDialogCaption プロパティを使います。このメソッドは以前のバージョンとの互換性のために用意されています。、ImpressPropertyDialog メソッドを利用してください。	
<b>BOOL UserPasswordDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>          BOOL:数値（または BOOL 値）で取得が行えます          0（FALSE）:パスワード設定を行うダイアログボックスで [キャンセル] が選択された          1（TRUE）:パスワード設定を行うダイアログボックスで [OK] が選択された</p> <p><b>備考</b></p>	
<p><b>使用例</b>          bResult = オブジェクト.UserPasswordDialog（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstmp32 = CreateObject("Dstmp.StampObject.6") objDstmp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstmp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstmp32.WindowVisible = True objDstmp32.UserPassword = "&lt;変更を行うパスワード&gt;" objDstmp32.UserPasswordDialogCaption = "&lt;変更を行うタイトル文字列&gt;" objDstmp32.UserPasswordDialog() Set objDstmp32 = Nothing </pre>	

捺印ツール Extension > IDstmp32	メソッド
WindowDialog	
印面の表示設定するダイアログボックスを表示します。ダイアログボックスのタイトルを変更するには、WindowDialogCaption プロパティを使います。	
<b>BOOL WindowDialog()</b>	
<p><b>パラメータ</b> ありません</p> <p><b>戻り値</b>            BOOL:数値（または BOOL 値）で取得が行えます            0（FALSE）:表示設定行うダイアログボックスで [キャンセル] が選択された            1（TRUE）:表示設定を行うダイアログボックスで [OK] が選択された</p> <p><b>備考</b></p>	
<p><b>使用例</b>            bResult = オブジェクト.WindowDialog（）</p>	
<p><b>サンプルコード</b></p> <pre> Set objDstamp32 = CreateObject("Dstamp.StampObject.6") objDstamp32.DefaultFilePath = "&lt;捺印用印鑑データファイルの場所&gt;" 'ログインを行うダイアログボックスを表示 nResult = objDstamp32.UserDialog() '印鑑選択ウィンドウの表示設定 objDstamp32.WindowVisible = True '表示設定ダイアログの表示 objDstamp32.WindowDialog() Set objDstamp32 = Nothing           </pre>	



## 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。

捺印ツール Extension

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/>