





4D Write Proリファレンス

-  はじめに
-  4D Write Pro エリアの作成
-  4D Write Pro エリアを使用する
-  4D Write Proランゲージ

はじめに

概要

4D Write Proは、4Dユーザーに対して、4Dデータベースに完全に統合した進化したワードプロセストールを提供します。4D Write Proを使用すれば、プリフォーマットされたEメールや文章に画像、スキャン済みの署名、フォーマット済みのテキストやダイナミック変数用のプレースホルダーなどを含めることができます。また請求書やレポートを動的に作成し、フォーマット済みのテキストや画像を含めることができます。

この製品の主な機能は以下の通りです：

- 4D Write との互換性: 4D Write Proオブジェクトは、以前の4D Writeドキュメントを開いたり変換したりでき、また4D Writeのプロパティの大部分をサポートしています。
- ワードプロセス: フォームに埋め込まれた4D Write Proオブジェクトは、テキストとスタイルの操作、画像の挿入、読み込みと書き出しなどを含めた様々な標準のワープロ機能を持ち合わせます。
- データベースとの統合:
 - 4D Write Pro オブジェクトはデータベースからのデータや、4Dが計算をしたデータなどを表示する変数部分を表示することができます。
 - 4D Write Pro ドキュメントはデータベースフィールド内や、ディスク上に保存することもできます。

インストールとアクティベート

4D Write Pro はプラグインではなく、4D自身に完全に統合されており、配布と管理がより簡単になっています。追加のインストールは何も必要なく、フォーム内に4D Write Proエリアを追加するだけで、4Dアプリケーション内において4D Write Pro変数を直接管理することができます。


しかしながら、4D Write Proは4D Writeと同じライセンスを使用することに注意して下さい。機能を有効化するためには、アプリケーションにそのライセンスをインストールする必要があります。

このマニュアルについて

このマニュアルは4D Write Pro リファレンスガイド です。ここでは全ての4D Write Proの機能に加え、ユーザーインターフェースとランゲージコマンドについてカバーします。

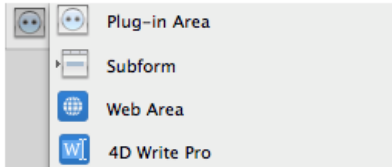
4D Write Proオブジェクトは、特定のコマンド("4D Write Pro" テーマ)に加えて、4Dランゲージリファレンス内にある他の4Dテーマ(**オブジェクト(フォーム)** テーマと**スタイル付きテキスト** テーマ)のコマンドを使用しても管理する事ができるという点に注意して下さい。

4D Write Pro

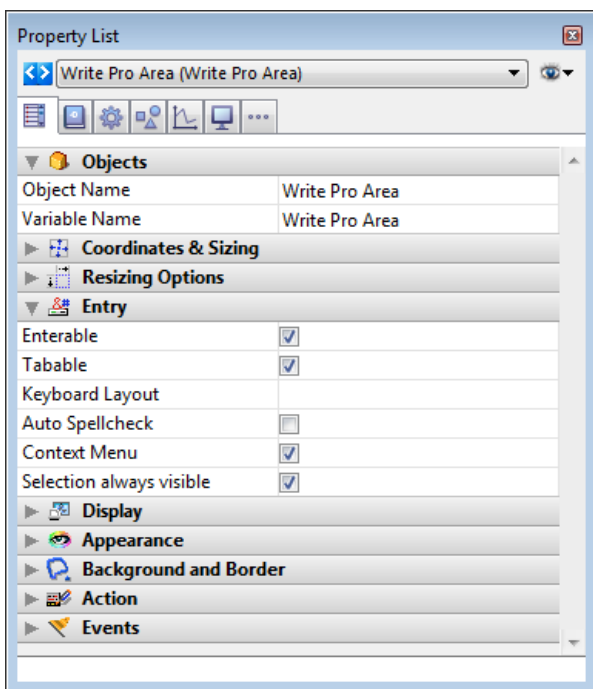
-  WP EXPORT DOCUMENT
-  WP EXPORT VARIABLE
-  WP Import document
-  WP New

4D Write Pro エリアの作成

4D Write Proドキュメントは**Write Pro**という4D フォームオブジェクト内にて手動で表示・編集することができます。このオブジェクトはオブジェクトバーの最終ツールの一部として提供されています:



4D Write Proフォームエリアはプロパティリスト内の標準のプロパティを通して設定することができます。すなわし、**オブジェクト名** そして **変数名**、**座標**、**入力**、**表示** それから **外見**、**イベント**です。



変数名のプロパティはランゲージにおいて4D Write Proエリアの参照として使用します。変数は`object`型でなければならないという点に注意して下さい(詳細については**C_OBJECT** コマンドを参照して下さい)。

"入力" プロパティはテキスト入力の基本的な機能を管理します:

- **入力可能:** エリアをロック・アンロックすることによって編集を許可または許可しないことを選択できます。
- **自動スペルチェック:** 4D Write Pro において利用可能です。
- **コンテキストメニュー:** アプリケーションモードにおいてコンテキストメニューを有効化・無効化します(**4D Write Pro エリアを使用する** の章を参照して下さい)。
- **選択を常に表示:** 標準のテキストエリアにおいてテキストの選択を管理します。

```
C_OBJECT ( {method ;} variable {; variable2 ; ... ; variableN} )
```

引数	型		説明
method	メソッド	→	メソッド名
variable	変数	→	宣言する変数の名前、または引数\${...}

説明

C_OBJECT は、指定されたそれぞれの変数をランゲージオブジェクト型としてタイプ定義します。

Object 型はv14より4Dランゲージでサポートされるようになりました。この型のオブジェクトは **オブジェクト(ランゲージ)** テーマのコマンドで管理されます。

プロセス、インタープロセスまたはローカル変数を宣言・タイプするためには(*method* 引数が渡されない)第一記法を使用します。この記法はインタープリテッドデータベースにて使用可能です。

メソッドの結果と引数 (\$0, \$1, \$2, 等)をコンパイラへと先に宣言するためには(*method* 引数が渡される)第二記法を使用します。データベースがコンパイルされる際に変数タイピングフェーズを省略して時間を節約したいときにはこの記法を使用しなければなりません。

警告: 第二記法はインタープリタモードで実行することはできません。このため、この記法を使用する際には、この記法をインタープリタモードでは実行されないメソッド(名前が"COMPILER"で始まる必要があります)に保存する必要があります。

特殊使用法: **C_OBJECT**(\${...}) の記法を使用すると、メソッドの最後の引数が同じ型であればメソッドに対して同じ型の不定数の引数を指定することができます。例えば、**C_OBJECT**(\${5}) は5番目の引数以降、メソッドはこの型の引数を不定数受け取ることができるということをコンパイラに意味します。

例題

コンパイラコマンド セクションを参照してください。

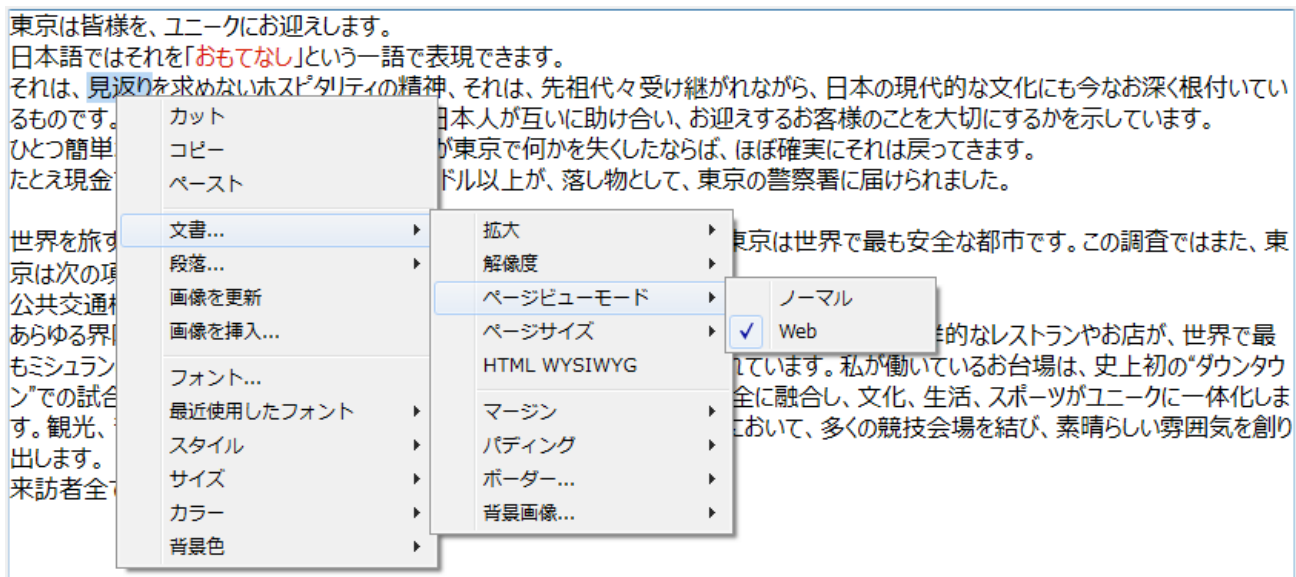
4D Write Pro エリアを使用する

4D Write Pro エリア内のドキュメントを管理する

4D アプリケーションでは、4D Write Proドキュメントは、4D Write Proテーマ内にある特定のコマンドを使用することによって作成、読み込み、書き出しをすることができます(**WP EXPORT DOCUMENT**、**WP EXPORT VARIABLE**、**WP Import document**、**WP New**)。

ユーザーインターフェース

4D Write Pro エリア内にて**コンテキストメニュー**プロパティがチェックされている場合、(**4D Write Pro エリアの作成**を参照して下さい)、アプリケーションモードにおいてユーザーは包括的なコンテキストメニューを使用することができます:



このメニューを通して、提供されている全ての4D Write Pro 機能へとアクセスすることができます。

WP EXPORT DOCUMENT (wpDoc ; filePath {; format {; option}})

引数	型		説明
wpDoc	Object	→	4D Write Pro 変数
filePath	文字	→	書き出されたファイルへのパス
format	倍長整数	→	ドキュメントの出力フォーマット
option	倍長整数	→	書き出しオプション

説明

WP EXPORT DOCUMENT コマンドは *wpDoc* 引数に渡したディスク上の4D Write Proオブジェクトを、*filePath* 引数で指定した場所に、他の任意の引数に基づいて書き出します。

wpDoc 引数には、書き出したい4D Write Proオブジェクトを渡します。

filePath 引数には、書き出し先の場所と、書き出したドキュメント名を渡します。ドキュメント名のみを渡した場合、4Dストラクチャーファイルと同階層に保存されます。デフォルトで、*format* 引数を省略した場合、コマンドはドキュメントの拡張子を見て書き出しのフォーマットを選択します。

format 引数には**4D Write Pro** テーマからの定数を渡す事もできます。この場合、4Dは必要に応じて適切な拡張子をファイル名に追加します。サポートされるフォーマットは以下の通りです：

定数	型	値	コメント
wk web page complete	倍長整数	2	.htm または .html 拡張子。このドキュメントは標準HTMLとして保存され、そのリソースは別に保存されます。4Dタグは除去され、式は値が計算されます。このフォーマットは特に4D Write Pro ドキュメントWeb ブラウザで表示したい場合に特に適しています。

注:

- "4D 特有のタグ"とは、4Dネームスペースと4D CSSスタイルを含めた4D XHTMLのことです。
- 式は、**ST FREEZE EXPRESSIONS**コマンドを使用することで、書き出す前でもいつでも値を確定させることができます。

options 引数には、書き出しの設定を指定するオプションを渡します。倍長整数 の値を渡す事で、HTMLコードのスタイルを定義することができます。使用できる定数は以下の通りです：

定数	型	値	コメント
wk html debug	倍長整数	1	フォーマット済みのHTMLコード("整形済みフォーマット")。デバッグが容易。
wk normal	倍長整数	0	標準のHTML コード

- **HTML デバッグオプションなし(デフォルト):**

```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }p,li { white-space:pre-wrap;margin:0pt;padding:0pt;font-family:'Times New Roman' }p.Normal,li.Normal { text-align:left }p_p1,li_p1 { font-family:'Arial';font-size:18pt;color:#1D1D1D }img_img1 { width:51pt;height:51pt }</style></head><body><p class="Normal_p1"><span>23/12/14</span></p><p class="Normal_p1">Dear <span style="color:#BE0E00">Shaun Stoltz</span><span style="color:#800062"> <span style="color:#000000">from
```

- **HTML デバッグオプションあり:**









```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }
```

例題

myArea という4D Write Proオブジェクトの中身をデータベースフォルダー内のドキュメントへと書き出したい場合を考えます。'pprint'ボタンを使用してデバッグオプションを設定することができます：

```
C_TEXT($filePath)
$filePath:=Get 4D folder(Database folder)+"Exported files"+Folder
separator+"WriteProExport.html"
If(pprint=0) //デバッグオプションがoffの場合
    WP EXPORT DOCUMENT(myArea;$filePath;wk web page complete;wk normal)
Else
    WP EXPORT DOCUMENT(myArea;$filePath;wk web page complete;wk html debug)
End if
```


4D Write Proランゲージ

-  4D Write Pro オブジェクトについて
-  オブジェクト(フォーム)テーマのコマンドの使用
-  スタイル付テキストテーマのコマンドの使用
-  4D Write Pro
-  WP EXPORT DOCUMENT
-  WP EXPORT VARIABLE
-  WP Import document
-  WP New

4D Write Pro オブジェクトについて

4D Write Pro オブジェクト

4Dアプリケーションにおいて、4D Write Proドキュメントは`object` 型の変数を使用して管理されます(詳細な情報に関しては、[C_OBJECT](#) コマンドを参照して下さい)。

4D Write Pro オブジェクトを一度メモリで参照した後は、以下の様な事をすることができます:

- "Write Pro エリア"というフォームオブジェクトを使用して、4Dフォームを通して表示・編集ができます。その場合、フォームエリアはオブジェクト変数と同じ名前である必要があります([4D Write Pro エリアの作成](#)を参照して下さい)。
- 複数のテーマのコマンドによって、プログラムで管理することができます(以下参照)。
- **WP EXPORT DOCUMENT** コマンドを使用して、ディスク上にHTMLドキュメントとして書き出す事ができます。

4D Write Pro オブジェクトを扱う事ができるコマンド

4D Write Pro オブジェクトは、複数のテーマの4Dコマンドを使用して管理することができます:

- **4D Write Pro** テーマの4D Write Pro専用のコマンド。これらのコマンドはこの章内にて詳細な説明があります。
- **オブジェクト(フォーム)** コマンド。フォーマット機能・オブジェクトプロパティ機能を管理します。詳細な情報に関しては、[オブジェクト\(フォーム\)テーマのコマンドの使用](#) の章を参照して下さい。
- **スタイル付きテキスト** コマンド。コンテンツ機能を管理します。詳細な情報に関しては、[スタイル付テキストテーマのコマンドの使用](#) の章を参照して下さい。

📄 オブジェクト(フォーム)テーマのコマンドの使用

以下のコマンドは4D Write Pro フォームオブジェクトをサポートします:

コマンド	補足
OBJECT DUPLICATE	
OBJECT Get auto spellcheck/OBJECT SET AUTO SPELLCHECK	
OBJECT Get border style/OBJECT SET BORDER STYLE	
OBJECT Get context menu/OBJECT SET CONTEXT MENU	
OBJECT GET COORDINATES/OBJECT SET COORDINATES	
OBJECT Get data source/OBJECT SET DATA SOURCE	
OBJECT GET DRAG AND DROP OPTIONS/OBJECT SET DRAG AND DROP OPTIONS	
OBJECT Get enabled/OBJECT SET ENABLED	
OBJECT Get enterable/OBJECT SET ENTERABLE	
OBJECT GET EVENTS/OBJECT SET EVENTS	
OBJECT Get focus rectangle invisible/OBJECT SET FOCUS RECTANGLE INVISIBLE	
OBJECT Get font/OBJECT SET FONT	カレントセクションがあればそこに適用
OBJECT Get font size/OBJECT SET FONT SIZE	カレントセクションがあればそこに適用
OBJECT Get font style/OBJECT SET FONT STYLE	カレントセクションがあればそこに適用
OBJECT Get horizontal alignment/OBJECT SET HORIZONTAL ALIGNMENT	カレントセクションがあればそこに適用
OBJECT GET RESIZING OPTIONS/OBJECT SET RESIZING OPTIONS	
OBJECT SET COLOR	カレントセクションがあればそこに適用
OBJECT GET RGB COLORS/OBJECT SET RGB COLORS	カレントセクションがあればそこに適用
OBJECT Get type	
OBJECT Get vertical alignment/OBJECT SET VERTICAL ALIGNMENT	段落の垂直方向の行揃え:段落の高さが段落のテキストの高さより高い場合にのみ効力を持ちます。
OBJECT Get visible/OBJECT SET VISIBLE	
OBJECT Is styled text	trueを返します
OBJECT MOVE	
OBJECT GET SCROLL POSITION/OBJECT SET SCROLL POSITION	
OBJECT GET SUBFORM CONTAINER SIZE	
OBJECT Get name	
OBJECT Get pointer	

上記にないOBJECT コマンドはどれも4D Write Pro エリアに対しては使用できません。

例題

4D Write Pro エリアの垂直方向の行揃えを中央に設定したい場合を考えます:

```
Case of
: (Form event=On_Clicked)
  OBJECT SET HORIZONTAL ALIGNMENT (*;"myObject";Align_center)
  hAlignLeft:=0 //次回表示のためにプロパティを保存
  hAlignRight:=0
End case
```

📄 スタイル付テキストテーマのコマンドの使用

以下のコマンドは4D Write Pro オブジェクトをサポートします:

コマンド	補足
ST COMPUTE EXPRESSIONS	
ST FREEZE EXPRESSIONS	
ST GET ATTRIBUTES/ST SET ATTRIBUTES	
ST Get content type	画像コンテンツの型に新しいタイプ(6)が追加されました
ST Get expression / ST INSERT EXPRESSION	
ST GET OPTIONS/ST SET OPTIONS	
ST Get plain text/ST SET PLAIN TEXT	
ST Get text / ST SET TEXT	
ST GET URL / ST INSERT URL	

例題

4D Write Pro エリアの選択範囲を変数の中身で置き換えたい場合を考えます:

```
C_TEXT(fullName)

Case of
: (Form event=On Clicked)
  ST INSERT EXPRESSION(myArea;"fullName";ST_Start_highlight;ST_End_highlight)
End case
```

4D Write Pro

定数	型	値	コメント
wk html debug	倍長整数	1	フォーマット済みのHTMLコード("整形済みフォーマット")。デバッグが容易。
wk mime html	倍長整数	1	4D Write Proドキュメントは標準のMIME HTMLとして保存され、htmlドキュメントと画像はMIMEパーツとして埋め込まれます(base64でエンコードされます)。式は計算され4D特有のタグは除去されます。このフォーマットはSMTP_QuickSend コマンドを使用してHTML Eメールを送信するのに特に適しています。
wk normal	倍長整数	0	標準のHTML コード
wk web page complete	倍長整数	2	.htm または .html 拡張子。このドキュメントは標準HTMLとして保存され、そのリソースは別に保存されます。4Dタグは除去され、式は値が計算されます。このフォーマットは特に4D Write Pro ドキュメントWeb ブラウザで表示したい場合に特に適しています。
wk web page html 4D	倍長整数	3	4D Write ProドキュメントはHTMLとして保存され、4D独自のタグを含みます。それぞれの式はノンブレイクのスペースとして挿入されます。このフォーマットはロスレスであるため、テキストフィールドへの保存目的に適しています。

WP EXPORT VARIABLE (wpDoc ; destination ; format {; option})

引数	型		説明
wpDoc	Object	→	4D Write Pro変数
destination	テキスト変数	←	書き出したコンテンツを受け取る変数
format	倍長整数	→	変数出力フォーマット
option	倍長整数, 文字	→	書き出しオプション

説明

WP EXPORT VARIABLE コマンドはwpDoc 変数内の4D Write Proオブジェクトをformat 引数で指定したフォーマットでdestination 引数で指定した4D変数へと書き出します。

wpDoc 引数には、書き出したい4D Write Proオブジェクトを渡します。

destination 引数には、書き出された4D Write Pro オブジェクトを受け取るテキスト変数を渡します。

format 引数には、**4D Write Pro** テーマから、使用する書き出しフォーマットを設定する定数を渡します。それぞれのフォーマットは特定の用法と関連しています。サポートされているフォーマットは以下の通りです：

定数	型	値	コメント
wk mime html	倍長整数	1	4D Write Proドキュメントは標準のMIME HTMLとして保存され、htmlドキュメントと画像はMIME パーツとして埋め込まれます(base64でエンコードされます)。式は計算され4D特有のタグは除去されます。このフォーマットは SMTP_QuickSend コマンドを使用してHTML Eメールを送信するのに特に適しています。
wk web page html 4D	倍長整数	3	4D Write ProドキュメントはHTMLとして保存され、4D独自のタグを含みます。それぞれの式はノンブレークのスペースとして挿入されます。このフォーマットはロスレスであるため、テキストフィールドへの保存目的に適しています。

注:

- "4D独自のタグ" とは4Dネームスペースと4D CSSスタイルを含めた4D XHTMLのことです。
- 式は、書き出しをする前であれば**ST FREEZE EXPRESSIONS** コマンドを使用することでいつでも値を確定させることができます。

options 引数には、書き出しの詳細を設定するオプションを渡します。詳細は以下の通りです：

- 倍長整数 の値を渡してHTMLコードのスタイルを定義します。以下の定数を使用することができます。

定数	型	値	コメント
wk html debug	倍長整数	1	フォーマット済みのHTMLコード("整形済みフォーマット")。デバッグが容易。
wk normal	倍長整数	0	標準のHTML コード

- HTML デバッグオプションなし(デフォルト):

```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }p,li { white-space:pre-wrap;margin:0pt;padding:0pt;font-family:'Times New Roman' }p.Normal,li.Normal { text-align:left }p._p1,li._p1 { font-family:'Arial';font-size:18pt;color:#1D1D1D }img, img1 { width:51pt;height:51pt }</style></head><body><p class="Normal _p1"><span>23/12/14</span></p><p class="Normal _p1">Dear <span style="color:#BE0E00">Shaun Stoltz</span><span style="color:#800062"><span style="color:#000000">from
```

- HTML デバッグオプションあり:

```
<html xmlns="http://www.w3.org/1999/xhtml"><head><title>New 4D Write Pro Document</title><style type="text/css">body { background-color:#FFFFFF }ul, ol { margin:0;padding:0 }
```

- または文字列を渡す事もできます。以下のプロパティがサポートされています(ただし [wp mime html](#) フォーマットが使用されている場合に限ります):
 - CID ホストドメイン名: 生成されたCIDのURLに追加されるホストドメイン(区切り文字として'@'を含む)。例えば、"gmail.com"を渡した場合、CIDユニークIDが123のときには'123@gmail.com'が挿入されます。省略された場合、デフォルトで、CIDユニークIDのみが使用されます(ほとんどのメールサーバーで使用可能です)。

例題

スタイル付テキストと4D参照と画像を含んだEメールを送信したい場合を考えます。この場合、MIME HTMLフォーマットで書き出された4D Write Proエリアを、4D Internet Commandを使用して送信することができます:

```
C_LONGINT($smtpid_1;$err_1;$smtpOption_1;$smtpPort_1)
C_TEXT($str;$emailBody_t;$smtpHost_t;$emailTo_t;$emailFrom_t;$smtpPass_t)

//エリアを適切なフォーマットで書き出す
WP_EXPORT_VARIABLE(myWPArea;$str;wk_mime_html)
$emailTo_t:="johnsmith@4d.com"
$emailFrom_t:"testWritePro@gmail.com"
$emailBody_t:=$str
$smtpHost_t:"smtp.gmail.com"
$smtpOption_1:=9
$smtpPort_1:=465
$smtpPass_t:"QRN_on_bretzelburg"

$err_1:=SMTP_QuickSend($smtpHost_t;$emailFrom_t;$emailTo_t;$emailTitle_t;\
$emailBody_t;$smtpOption_1;$smtpPort_1;$smtpUser_t;$smtpPass_t)

If(($err_1=0))
    ALERT("Email sent to "+emailTo_t)
Else
    ALERT("Error in parameters, please try again.")
End if
```


WP Import document

WP Import document (filePath) -> 戻り値

引数	型		説明
filePath	文字	→	4D Write ドキュメント(.4w7または.4wt)へのパス
戻り値	Object	↩	4D Write Proオブジェクト

説明

WP Import document コマンドは既存の4D Write ドキュメント(.4w7 または .4wt)を新規の4D Write Proオブジェクトへと変換します。

filePath 引数には、ディスク上に保存された4D Writeドキュメントへのフルパスを渡します。ドキュメントがストラクチャーファイルと同等の階層の保存されている場合には、ドキュメント名のみを渡す事もできます。

実行後、コマンドはこの変換の結果を格納した4D Write Proオブジェクトを返します。

注: 4D BLOBフィールドに保存された4D Write ドキュメントを読み込みたい場合には、**WP New** コマンドの使用も検討してみてください。

filePath 引数に渡したパスが無効な場合、またはファイルが見つからない場合、または指定したファイルがサポートされていないフォーマットだった場合、エラーが返されます。

例題

```
C_OBJECT (WPDoc)
WPDoc:=WP Import document ("C:\\documents\\4DWriteDocs\\Letter.4w7")
```

WP New {{ source }} -> 戻り値

引数	型		説明
source	文字, BLOB	→	4D HTMLソースまたは4D Write Blob
戻り値	Object	↩	4D Write Pro オブジェクト

説明

WP New コマンドは4D Write Pro オブジェクトを作成し、返します。

source 引数を省略した場合、コマンドはデフォルトで空の4D Write Proオブジェクトを返します。

また *source* 引数を使用した場合、新しい4D Write Pro オブジェクトは*source* 引数の中身をコンテンツとして返されます。渡せる内容は以下の通りです:

- 文字列 の引数: この場合、4D HTMLソースを渡します。つまり、`wk web page html 4D` オプションを使用した**WP EXPORT VARIABLE** で書き出されたテキストです。このテキストは参照(4Dタグと式)と埋め込まれた画像を含むことができます。
- *blob* 引数: この場合、以前の4D WriteエリアをBLOBに読み込ませて渡します(.4w7 または .4wt を含んだBlobがサポートされます)。4D Write Proオブジェクト内で現在サポートされている4D Writeの機能の詳細な一覧については、[4D Write ドキュメントの読み込み](#) の章を参照して下さい。
ディスク上に保存されている4D Write ドキュメント(.4w7 or .4wt) を読み込みたい場合、**WP Import document** コマンドの使用も検討してみてください。

例題 1

空の4D Write Proオブジェクトを作成したい場合を考えます:

```
myWPObject:=WP New
```

例題 2

簡単な4D式の参照を含んだ4D Write Proオブジェクトを作成したい場合を考えます:

```
C_TEXT(myText)
myText:="Today is "
ST INSERT EXPRESSION(myText;"string(current date;System date long)";ST_End text)
myWPA:=WP New(myText)
```

例題 3

以前作成したテンプレートを使用して4D Write Proエリアを初期化したい場合を考えます:

```
// 既存のエリアからテンプレートを書き出し
C_TEXT(wpTemplate)
WP EXPORT VARIABLE(myWPArea;wpTemplate;wk web page html 4D)

// 新規エリアに対してテンプレートを使用
C_OBJECT(myNewWPA)
myNewWPA:=WP New(wpTemplate)
```

例題 4

4Dフィールドに保存されている4D Write ドキュメントを新しい4D Write Proエリア内に読み込みたい場合を考えます:

```
C_OBJECT(wpArea)
wpArea=WP New([Templates]Reference_)
```