About RTF Generator – Professional Edition

RTF Generator

The main purpose of the script is generating (not parsing) RTF files. It consists in a PHP class and it is based on Rich Text Format (RTF) Specification Version 1.7 as is published by Microsoft Corporation.

Professional Edition

The Professional Edition of the RTFGen V 2.0 contains a complete set of functions which allow you to create enhanced RTF documents on the fly.

With RTFGen - Professional Edition you will be able to set the format of the document as you want, to add as many section as you want, to add your own color in the color table of the document, to add the fonts you want, to create your own formatting styles in the style sheet of the document, to write text within the paragraphs with the styles you've created, to add headers and footers, to add images, to create dynamically the tables you need and more than that you will be able to add footnotes, endnotes, bookmarks, fields, comments, etc.

If you need foot-notes or end-notes or fields or annotations or bookmarks or fields and images in the headers/footers or more sophisticated tables or even if you want to create custom user defined properties on the document or anything else then the RTFGen - Professional Edition, will offers you a total control over the RTF file.

What is the RTF ?

The rich-text format (RTF) standard is a method of encoding formatted text and graphics for easy transfer between different programs and different operations.

The RTF Specification provides a format for text and graphics interchange that can be used with different output devices, operating environments, and operating systems. Generally, it is used by all Microsoft Word programs — Word for Windows, Word for the Macintosh, and Word for MS-DOS — for moving word-processing documents between different platforms without having to rely on special translation software or conversion utilities. RTF uses the ANSI, PC-8, Macintosh, or IBM PC character set to control the representation and formatting of a document, both on the screen and in print. With the RTF Specification, documents created under different operating systems and with different software applications can be transferred between those operating systems and applications.

RTF version 1.7 includes all new control words introduced by Microsoft Word for Windows 95 version 7.0, Word 97 for Windows, Word 98 for the Macintosh, Word 2000 for Windows, and Word 2002 for Windows, as well as other Microsoft products.

Technical information

The RTF Generator is written in PHP, in fact it is a PHP class who uses no additional components. It could be easily installed and used to create enhanced rich text documents you need. The script was developed in PHP 4.3.0 running as a module in Apache 1.3.14.

Installing the RTF Generator

What the package contains?

The RTFGen V 2.0 – Professional Edition contains the following files :

- "frtf.php" which is the main file containing the PHP class
- "conf.inc.php" which is used to configure the class
- "param.inc.php" which is containing all the class parameters
- "rtfkeywords.inc.php" which contains all the keywords defined by RTF Specification V 1.7

Installing the script

In order to install the script all you have to do is to unzip the archive in the directory at your choice.

Methods

Index of methods

- Document formatting methods

- SetAuthor
- SetBackupTime
- SetBookfold
- SetBookfoldrev
- SetBookfoldsheets
- SetBorderArt
- SetComment
- SetCompany
- SetCreationTime
- SetDefaultTab
- SetDefFormat
- SetDefLang
- SetDefLangfe
- SetDoccomm
- SetDocSource
- SetDocTemp
- SetDocType
- SetDocumentView
- SetEditionTime
- SetFacingp
- SetFracWidth
- SetFromHtml
- SetFromText
- SetGutter
- SetGutterPos
- SetHlinkbase
- SetHorzDoc
- SetHyphenAuto
- SetHyphenCaps
- SetHyphenConsec
- SetHyphenHot
- SetKeywords
- SetLineStart
- SetMakeBackup

- SetManager
- SetMargb
- SetMargins
- SetMargl
- SetMargmirror
- SetMargr
- SetMargt
- SetNextFile
- SetOperator
- SetPageBorderFoot
- SetPageBorderHead
- SetPageBorderOption
- SetPageBorders
- SetPageOrientation
- SetPaperh
- SetPaperSize
- SetPaperw
- SetParBorderAlign
- SetPgnStart
- SetPrintTime
- SetPsOver
- SetRender
- SetRevisionTime
- SetSubject
- SetTemplate
- SetTitle
- SetTwoonone
- SetVersionDoc
- SetVertDoc
- SetViewKind
- SetViewScale
- SetViewZk
- SetWidowctrl
- SetWindowCaption

  - Section formatting methods

- AddNewSection
- Sect_BeginLineNb
- Sect_Break
- Sect_CharSpaceBasement
- Sect_ColNumber
- Sect_ColWidth
- Sect_DistLineNbTextLeft
- Sect_FirstPage
- Sect_IncludeEndNote
- Sect_LineBetCols
- Sect_LineGrid
- Sect_LineModulus
- Sect_LineNbRestart
- Sect_NbColForSnaking
- Sect_PageNbIncludeChapter
- Sect_PageNbRestartPolicy
- Sect_PageNbStyle
- Sect_PageNumberFormat
- Sect_PageOrientation

- Sect_PrintBin
- Sect_PrintBinFirst
- Sect_ResetToDefaultProp
- Sect_SetBidirectional
- Sect_SetBorderArt
- Sect_SetFooterY
- Sect_SetGutter
- Sect_SetHeaderY
- Sect_SetMargb
- Sect_SetMargins
- Sect_SetMargl
- Sect_SetMargmirror
- Sect_SetMargr
- Sect_SetMargt
- Sect_SetPageBorderFoot
- Sect_SetPageBorderHead
- Sect_SetPageBorderOption
- Sect_SetPageBorders
- Sect_SetPageNbRight
- Sect_SetPageNbTop
- Sect_SetPaperh
- Sect_SetPaperSize
- Sect_SetPaperw
- Sect_SetParBorderAlign
- Sect_SetRender
- Sect_SpaceBetCols
- Sect_SpaceToRightOfCol
- Sect_StateOf
- Sect_TextFlow
- Sect_TextShouldSnap
- Sect_UnlockedForForms
- Sect_VertAlign

  - Functional methods

- AddColor
- AddFont
- AddStyle
- Bullet
- Close
- Date
- DefineFooters
- DefineHeaders
- Error
- FRTF
- InsertFooter
- InsertHeader
- InsertPicture
- InsertTable
- LastWordSupported
- LineBreak
- Open
- Output
- PageBreak
- Tab
- Write

Document formatting methods

1. DefineEndNote
2. DefineFootNote
3. DocAllProtected
4. DrawGridToMargins
5. ProtectDocForComments
6. ProtectDocForForms
7. ProtectDocForRevision
8. SetAuthor
9. SetBackupTime
10. SetBookfold
11. SetBookfoldrev
12. SetBookfoldsheets
13. SetBorderArt
14. SetClickAndTypeStyle
15. SetComment
16. SetCompany
17. SetCreationTime
18. SetDefaultTab
19. SetDefFormat
20. SetDefLang
21. SetDefLangfe
22. SetDoccomm
23. SetDocDirection
24. SetDocFormShade
25. SetDocRevisions
26. SetDocSource
27. SetDocTemp
28. SetDocType
29. SetDocumentView
30. SetDrawGridHorizOrig
31. SetDrawGridHorizSpace
32. SetDrawGridVertOrig
33. SetDrawGridVertSpace
34. SetEditionTime
35. SetFacingp
36. SetFormDisp
37. SetFormPrintData
38. SetFracWidth
39. SetFromHtml
40. SetFromText
41. SetGutter
42. SetGutterPos
43. SetHlinkbase
44. SetHorzDoc
45. SetHyphenAuto
46. SetHyphenCaps
47. SetHyphenConsec
48. SetHyphenHot
49. SetKeywords
50. SetLineStart
51. SetMakeBackup

Compatibility options

52. CmptOpt_Word_6_95_BorderRules
53. CmptOpt_Word95AutoSpacing
54. CmptOpt_WordPerfectJustify
55. CmptOpt_WordPerfectSpaceWidth
56. CmptOpt_WordPerfectTabStop
57. CmptOpt_WrapTrsp

CmptOpt_AllowEntireFieldSelection

Signature

function CmptOpt_AllowEntireFieldSelection ()

Description

Enables selecting the entire field with the first or last character.

Parameters

CmptOpt_AllowHangingPunctuation

Signature

function CmptOpt_AllowHangingPunctuation ()

Description

Allow hanging punctuation in character grid.

Parameters

CmptOpt_AsianRulesForLineBreak

Signature

function CmptOpt_AsianRulesForLineBreak ()

Description

Use Asian rules for line breaks with character grid.

Parameters

CmptOpt_CvMME

Signature

function CmptOpt_CvMME ()

Description

Treat old-style escaped quotation marks (\") as current style ("") in mail merge data documents.

Parameters

CmptOpt_DontAdjustLineHeightInTable

Signature

function CmptOpt_DontAdjustLineHeightInTable ()

Description

Don't adjust line height in table.

Parameters

CmptOpt_DontAlignTableInd

Signature

function CmptOpt_DontAlignTableInd ()

Description

Don’t align table rows independently.

Parameters

CmptOpt_DontAllowTableRowLayoutApart

Signature

function CmptOpt_DontAllowTableRowLayoutApart ()

Description

Don’t allow table rows to lay out apart.

Parameters

CmptOpt_DontBalanceSBDB

Signature

function CmptOpt_DontBalanceSBDB ()

Description

Don't balance SBCS/DBCS characters. Option for compatibility with Word 6.0 (Japanese).

Parameters

CmptOpt_DontBreakWrappedTable

Signature

function CmptOpt_DontBreakWrappedTable ()

Description

Don’t break wrapped tables across pages.

Parameters

CmptOpt_DontCenter

Signature

function CmptOpt_DontCenter ()

Description

Don’t center exact line height lines.

Parameters

CmptOpt_DontLayoutAutoShape

Signature

function CmptOpt_DontLayoutAutoShape ()

Description

Don’t lay out AutoShapes like Word 97.

Parameters

CmptOpt_DontLayoutFootnotes

Signature

function CmptOpt_DontLayoutFootnotes ()

Description

Don’t lay out footnotes like Word 6.0, Word 95, and Word 97.

Parameters

CmptOpt_DontLayoutTableWithRawWidth

Signature

function CmptOpt_DontLayoutTableWithRawWidth ()

Description

Don’t lay out tables with raw width.

Parameters

CmptOpt_DontShowComments

Signature

function CmptOpt_DontShowComments ()

Description

Don't show comments while reviewing.

Parameters

CmptOpt_DontShowFormatting

Signature

function CmptOpt_DontShowFormatting ()

Description

Don't show formatting while reviewing.

Parameters

CmptOpt_DontShowInsDel

Signature

function CmptOpt_DontShowInsDel ()

Description

Don't show insertions and deletions while reviewing.

Parameters

CmptOpt_DontShowMarkup

Signature

function CmptOpt_DontShowMarkup ()

Description

Don't show markup while reviewing.

Parameters

CmptOpt_ExpandShiftReturn

Signature

function CmptOpt_ExpandShiftReturn ()

Description

Expand character spaces on line-ending with shift+return. Option for compatibility with Word 6.0 (Japanese).

Parameters

CmptOpt_HideWhiteSpaceBetweenPages

Signature

function CmptOpt_HideWhiteSpaceBetweenPages ()

Description

Hide white space between pages.

Parameters

CmptOpt_HTMLAutoSpacing

Signature

function CmptOpt_HTMLAutoSpacing ()

Description

Use HTML paragraph auto spacing.

Parameters

CmptOpt_LineBreakRule

Signature

function CmptOpt_LineBreakRule ()

Description

Don’t use Word 97 line breaking rules for Asian text.

Parameters

CmptOpt_MsmCap

Signature

function CmptOpt_MsmCap ()

Description

Small caps like Word for the Macintosh 5.x.

Parameters

CmptOpt_NoBackslashToYen

Signature

function CmptOpt_NoBackslashToYen ()

Description

Don't translate backslash to Yen sign. Option for compatibility with Word 6.0 (Japanese).

Parameters

CmptOpt_NoColBal

Signature

function CmptOpt_NoColBal ()

Description

Don't balance columns.

Parameters

CmptOpt_NoCompatibility

Signature

function CmptOpt_NoCompatibility ()

Description

Specifies that all compatibility options should be set to default.

Parameters

CmptOpt_NoExtLead

Signature

function CmptOpt_NoExtLead ()

Description

No external leading. Option for compatibility with Word for the Macintosh 5.x.

Parameters

CmptOPt_PrColBl

Signature

function CmptOPt_PrColBl ()

Description

Print all colors as black.

Parameters

CmptOpt_NoSpaceforUnderline

Signature

function CmptOpt_NoSpaceforUnderline ()

Description

Don't add space for underline. Option for compatibility with Word 6.0 (Japanese).

Parameters

CmptOpt_NoTabIndent

Signature

function CmptOpt_NoTabIndent ()

Description

Don't add automatic tab stop for hanging indent.

Parameters

CmptOpt_NOUnderlineTrlSpace

Signature

function CmptOpt_NOUnderlineTrlSpace ()

Description

Don't underline trailing spaces. Option for compatibility with Word 6.0 (Japanese).

Parameters

CmptOpt_OldLineWrap

Signature

function CmptOpt_OldLineWrap ()

Description

Lines wrap like Word 6.0.

Parameters

CmptOpt_OldTableRule

Signature

function CmptOpt_OldTableRule ()

Description

Combine table borders as done in Word 5.x for the Macintosh. Contradictory table border information is resolved in favor of the first cell.

Parameters

CmptOPt_PrColBl

Signature

function CmptOPt_PrColBl ()

Description

Print all colors as black.

Parameters

CmptOpt_PrintBodyBefore

Signature

function CmptOpt_PrintBodyBefore ()

Description

Print body before header/footer. Option for compatibility with Word for the Macintosh 5.x.

Parameters

Cmptopt_RemovePersonalInfo

Signature

function Cmptopt_RemovePersonalInfo ()

Description

This will indicate to the emitting program to remove personal information such as the author's name as a document property or in a comment.

Parameters

CmptOpt_ShowBrkFrame

Signature

function CmptOpt_ShowBrkFrame ()

Description

Show hard (manual) page breaks and column breaks in frames.

Parameters

CmptOpt_SnapTxtToGrid

Signature

function CmptOpt_SnapTxtToGrid ()

Description

Snap text to grid inside table with inline objects.

Parameters

CmptOpt_Sprstsm

Signature

function CmptOpt_Sprstsm ()

Description

Does nothing. This keyword should be ignored.

Parameters

CmptOpt_SubFontbySize

Signature

function CmptOpt_SubFontbySize ()

Description

Substitute fonts based on size first.

Parameters

CmptOpt_SuppExtraLine

Signature

function CmptOpt_SuppExtraLine ()

Description

Suppress extra line spacing at top of page. Basically, this means to ignore any line spacing larger than Auto at the top of a page.

Parameters

CmptOpt_SuppExtraLine_WordPerfect

Signature

function CmptOpt_SuppExtraLine_WordPerfect ()

Description

Suppress extra line spacing like WordPerfect version 5.x.

Parameters

CmptOpt_SuppExtraSpace

Signature

function CmptOpt_SuppExtraSpace ()

Description

Suppress space before paragraph property after hard page or column break.

Parameters

CmptOpt_SuppLineAtBottomPage

Signature

function CmptOpt_SuppLineAtBottomPage ()

Description

Suppress extra line spacing at bottom of page.

Parameters

CmptOpt_SwitchParagraphBorder

Signature

function CmptOpt_SwitchParagraphBorder ()

Description

If a paragraph has a left border (not a box) and the Different Odd And Even or Mirror Margins check box is selected, Word will print the border on the right for odd-numbered pages.

Parameters

CmptOpt_ThaiBreakingRules

Signature

function CmptOpt_ThaiBreakingRules ()

Description

Use line breaking rules compatible with Thai text.

Parameters

CmptOpt_TopLinePunct

Signature

function CmptOpt_TopLinePunct ()

Description

Turns on a check box in the Paragraph Formatting dialogue box with a setting to allow punctuation at the start of the line to compress.

Parameters

CmptOpt_TransparentMetafile

Signature

function CmptOpt_TransparentMetafile ()

Description

Metafiles are considered transparent; don't blank the area behind metafiles.

Parameters

CmptOpt_TruncateEx

Signature

function CmptOpt_TruncateEx ()

Description

Don't add leading (extra space) between rows of text.

Parameters

CmptOpt_TruncateFontHeight

Signature

function CmptOpt_TruncateFontHeight ()

Description

Round down to the nearest font size instead of rounding up.

Parameters

CmptOpt_UseLastTab

Signature

function CmptOpt_UseLastTab ()

Description

Don’t forget last tab alignment.

Parameters

CmptOpt_UsePrinterMetrics

Signature

function CmptOpt_UsePrinterMetrics ()

Description

Use printer metrics to lay out document.

Parameters

CmptOpt_Word_6_95_BorderRules

Signature

function CmptOpt_Word_6_95_BorderRules ()

Description

Use Word 6.0/Word 95 borders rules.

Parameters

CmptOpt_Word95AutoSpacing

Signature

function CmptOpt_Word95AutoSpacing ()

Description

Use Word 95 Auto spacing.

Parameters

CmptOpt_WordPerfectJustify

Signature

function CmptOpt_WordPerfectJustify ()

Description

Do full justification like WordPerfect 6.x for Windows.

Parameters

CmptOpt_WordPerfectSpaceWidth

Signature

function CmptOpt_WordPerfectSpaceWidth ()

Description

Set the width of a space like WordPerfect 5.x.

Parameters

CmptOpt_WordPerfectTabStop

Signature

function CmptOpt_WordPerfectTabStop ()

Description

Advance to next tab stop like WordPerfect 6.x.

Parameters

CmptOpt_WrapTrsp

Signature

function CmptOpt_WrapTrsp ()

Description

Wrap trailing spaces onto the next line.

Parameters

DefineEndNote

Signature

function DefineEndNote ($AEPlace = '', $AEJustif = '', $AEBeginNumb = '', $AENumbPolicy = '',$AENumbStyle = '')

Description

This method define the endnote type for the current document.

Parameters

$AEPlace – specify the place of endnote (end of document/ end of section).

Allowed values:

- 'section | sect | sec | s' endnote at end of section
- 'document | docum | doc | d' endnote at end of document

By default : 'Section'

$AEJustify – a string specifying the endnote alignment.

Allowed values:

- top | t | beneath text endnote is top justified
- bottom | bot | b | bottom of page endnote is bottom justified

By default : 'Bottom of page'

$AEBeginNumb – beginning endnote number - default value 1.

$AENumbPolicy – a string specifying the endnote restart number policy.

Allowed values:

- 'page | pag | p | pagina' restart numbering each page
- 'section | sec | s | sectiune' restart numbering each section
- 'continuous | cont | c | continuu' continuous footnote numbering

By default : 'Continuous'

$AENumbStyle – a string specifying the endnote numbering style.

Allowed values:

- 'arabic' —Arabic numbering (1, 2, 3, ¼).
- 'alphabetic lowercase | alfabetic mic | a l | al | a mic | amic' —Alphabetic lowercase (a, b, c, ¼).
- 'alphabetic uppercase | alfabetic mare | a u | au | a mare | amare' —Alphabetic uppercase (A, B, C, ¼).
- 'roman lowercase | roman mic | r l | rl | r mic | rmic' —Roman lowercase (i, ii, iii, ¼).
- 'roman uppercase | roman mare | r u | ru | r mare | rmare' —Roman uppercase (I, II, III, ¼).

- **'chicago'** —Chicago Manual of Style (*, †, §, §).
- **'korean 1 | korean1 | kor1'** —Korean numbering 1 (*chosung).
- **'korean 2 | korean2 | kor2'** —Korean numbering 2 (*ganada).
- **'circle'** —Circle numbering (*circlenum).
- **'kanji 1 | kanji1 | k1'** —kanji numbering without the digit character (*dbnum1).
- **'kanji 2 | kanji2 | k2'** —kanji numbering with the digit character (*dbnum2).
- **'kanji 3 | kanji3 | k3'** —kanji numbering 3 (*dbnum3).
- **'kanji 4 | kanji4 | k4'** —kanji numbering 4 (*dbnum4).
- **'double_byte | double byte'** —double-byte numbering (*dbchar).
- **'chinese 1 | chi 1 | c1'** —Chinese numbering 1 (*gb1).
- **'chinese 2 | chi 2 | c2'** —Chinese numbering 2 (*gb2).
- **'chinese 3 | chi 3 | c3'** —Chinese numbering 3 (*gb3).
- **'chinese 4 | chi 4 | c4'** —Chinese numbering 4 (*gb4).
- **'chinese zodiac 1 | zodiac 1 | zodiac1 | z1'** —Chinese Zodiac numbering 1 (* zodiac1). 甲、乙、丙… ~~甲、乙、丙…~~ 甲、乙、丙…
- **'chinese zodiac 2 | zodiac 2 | zodiac2 | z2'**—Chinese Zodiac numbering 2 (* zodiac2). 子、丑、寅…
- **'chinese zodiac 3 | zodiac 3 | zodiac3 | z3'**—Chinese Zodiac numbering 3 (* zodiac3).

By default : 'Arabic'

DefineFootNote

Signature

function DefineFootNote ($FNPlace = '', $FNJustif = '', $FNBeginNumb = '', $FNNumbPolicy = '',$FNNumbStyle = '')

Description

This method define the footnote type for the current document.

Parameters

$FNPlace – specify the place of footnote (end of document/ end of section).

Allowed values:

'section | sect | sec | s'	footnote at end of section
'document | docum | doc | d'	footnote at end of document

By default : 'Section'

$FNJustify – a string specifying the footnote alignment.

Allowed values:

top | t | beneath text	footnote is top justified
bottom | bot | b | bottom of page	footnote is bottom justified

By default : 'Bottom of page'

$FNBeginNumb – beginning footnote number - default value 1.

$FNNumbPolicy – a string specifying the footnote restart number policy.

Allowed values:

'page | pag | p | pagina'	restart numbering each page
'section | sec | s | sectiune'	restart numbering each section
'continuous | cont | c | continuu'	continuous footnote numbering

By default : 'Continuous'

$FNNumbStyle – a string specifying the footnote numbering style.

Allowed values:

'arabic'	—Arabic numbering (1, 2, 3, ¼).
'alphabetic lowercase | alfabetic mic | a l | al | a mic | amic'	—Alphabetic lowercase (a, b, c, ¼).
'alphabetic uppercase | alfabetic mare | a u | au | a mare | amare'	—Alphabetic uppercase (A, B, C, ¼).
'roman lowercase | roman mic | r l | rl | r mic | rmic'	—Roman lowercase (i, ii, iii, ¼).
'roman uppercase | roman mare | r u | ru | r mare | rmare'	—Roman uppercase (I, II, III, ¼).

- **'chicago'** —Chicago Manual of Style (*, †, §, §).
- **'korean 1 | korean1 | kor1'** —Korean numbering 1 (*chosung).
- **'korean 2 | korean2 | kor2'** —Korean numbering 2 (*ganada).
- **'circle'** —Circle numbering (*circlenum).
- **'kanji 1 | kanji1 | k1'** —kanji numbering without the digit character (*dbnum1).
- **'kanji 2 | kanji2 | k2'** —kanji numbering with the digit character (*dbnum2).
- **'kanji 3 | kanji3 | k3'** —kanji numbering 3 (*dbnum3).
- **'kanji 4 | kanji4 | k4'** —kanji numbering 4 (*dbnum4).
- **'double_byte | double byte'** —double-byte numbering (*dbchar).
- **'chinese 1 | chi 1 | c1'** —Chinese numbering 1 (*gb1).
- **'chinese 2 | chi 2 | c2'** —Chinese numbering 2 (*gb2).
- **'chinese 3 | chi 3 | c3'** —Chinese numbering 3 (*gb3).
- **'chinese 4 | chi 4 | c4'** —Chinese numbering 4 (*gb4).
- **'chinese zodiac 1 | zodiac 1 | zodiac1 | z1'** —Chinese Zodiac numbering 1 (* zodiac1). 甲、乙、丙…
~~甲、乙、丙…~~ 甲、乙、丙…
- **'chinese zodiac 2 | zodiac 2 | zodiac2 | z2'**—Chinese Zodiac numbering 2 (* zodiac2). 子、丑、寅…
- **'chinese zodiac 3 | zodiac 3 | zodiac3 | z3'**—Chinese Zodiac numbering 3 (* zodiac3).

By default : 'Arabic'

DocAllProtected

Signature

function DocAllProtected ()

Description

Specifies that the current document has no unprotected areas.

Parameters

DrawGridToMargins

Signature

function DrawGridToMargins ()

Description

Drawing grid to follow margins.

Parameters

ProtectDocForComments

Signature

function ProtectDocForComments ()

Description

This document is protected for comments (annotations). The user cannot edit the document but can insert comments (annotations).

Parameters

ProtectDocForForms

Signature

function ProtectDocForForms ()

Description

Specifies that the current document is protected for forms.

Parameters

ProtectDocForRevision

Signature

function ProtectDocForRevision ()

Description

Specifies that this document is protected for revisions. The user can edit the document, but revision marking cannot be disabled.

Parameters

SetAuthor

Signature

function SetAuthor($Author = '')

Description

Set the author of the document.

Parameters

$Author - the name you want to be the author of the document.

SetBackupTime

Signature

function SetBackupTime($dtNewTime)

Description

Set the backup time of the document.

Parameters

$dtNewTime - the new time you want to became the backup time of the document.

SetBookfold

Signature

function SetBookfold( )

Description

Book fold printing. Allows for printing documents that can easily be made into pamphlets.
This will print two pages side by side in landscape mode, and will print to the back of the sheet if the printer supports duplex printing.

Parameters

SetBookfoldrev

Signature

function SetBookfoldrev( )

Description

Reverse book fold printing for bidirectional languages.

Parameters

SetBookfoldsheets

Signature

function SetBookfoldsheets($BookFoldSheets = '')

Description

Set the number of sheets per booklet; this should be a multiple of four.

Parameters

$BookFoldSheets - the number of sheets per booklet.

SetBorderArt

Signature

function SetBorderArt($BorderArt = '')

Description

Set the page border art.

Parameters

$BorderArt - the argument is a value from 1 to 165 representing the number of the border. By default is 1.

SetClickAndTypeStyle

Signature

function SetClickAndTypeStyle ($Index = '')

Description

Set the index to the style to be used for Click-and-Type (0 is the default).

Parameters

$Break – the index to the style to be used - default value 0.

SetComment

Signature

function SetComment($Comment = '')

Description

Set the comments for the document.

Parameters

$Comment - a string value representing the comments of the document.

SetCompany

Signature

function SetCompany($Company = '')

Description

Set the Company for the document.

Parameters

$Company - a string value representing the name of the Company for the document.

SetCreationTime

Signature

function SetCreationTime($dtNewTime = '')

Description

Set the creation time of the document.

Parameters

$dtNewTime - the new time you want to became the creation time of the document. If null then the current time will be used.

SetDefaultTab

Signature

function SetDefaultTab($TabWidth = '')

Description

Set the default tab size in twips for the document.

Parameters

$TabWidth - an integer representing the default tab width in twips for the document. If null or missing then the default value wich is 720 will be used.

SetDefFormat

Signature

function SetDefFormat()

Description

Tells the RTF reader that the document should be saved in RTF format.

Parameters

SetDefLang

Signature

function SetDefLang($Lang = '')

Description

Defines the default language used in the document.

Parameters

$Lang - a string representing the language to be used in the document.

Allowed values:

- Afrikaans
- Albanian
- Arabic
- Arabic Algeria
- Arabic Bahrain
- Arabic Egypt
- Arabic General
- Arabic Iraq
- Arabic Jordan
- Arabic Kuwait
- Arabic Lebanon
- Arabic Libya
- Arabic Morocco
- Arabic Oman
- Arabic Qatar
- Arabic Syria
- Arabic Tunisia
- Arabic U.A.E.
- Arabic Yemen
- Armenian
- Assamese
- Azeri Cyrillic
- Azeri Latin
- Basque
- Bengali
- Bosnia Herzegovina
- Bulgarian
- Burmese
- Byelorussian
- Catalan
- Chinese China
- Chinese General
- Chinese Hong Kong
- Chinese Macao
- Chinese Singapore
- Chinese Taiwan

- Croatian
- Czech
- Danish
- Dutch Belgium
- Dutch Standard
- English Australia
- English Belize
- English British
- English Canada
- English Caribbean
- English General
- English Ireland
- English Jamaica
- English New Zealand
- English Philippines
- English South Africa
- English Trinidad
- English United States
- English Zimbabwe
- Estonian
- Faeroese
- Farsi
- Finnish
- French
- French Belgium
- French Cameroon
- French Canada
- French Cote d'Ivoire
- French Luxemburg
- French Mali
- French Monaco
- French Reunion
- French Senegal
- French Swiss
- French West Indies
- French Zaire
- Frisian
- Gaelic
- Gaelic Ireland
- Galician
- Georgian
- German
- German Austrian
- German Liechtenstein
- German Luxemburg
- German Switzerland
- Greek
- Gujarati
- Hebrew
- Hindi
- Hungarian
- Icelandic
- Indonesian
- Italian
- Italian Switzerland

- Japanese
- Kannada
- Kashmiri
- Kashmiri India
- Kazakh
- Khmer
- Kirghiz
- Konkani
- Korean
- Korean Johab
- Lao
- Latvian
- Lithuanian
- Lithuanian Classic
- Macedonian
- Malay
- Malay Brunei Darussalam
- Malayalam
- Maltese
- Manipuri
- Marathi
- Mongolian
- Nepali
- Nepali India
- Norwegian Bokmal
- Norwegian Nynorsk
- Oriya
- Polish
- Portuguese Brazil
- Portuguese Iberian
- Punjabi
- Rhaeto-Romanic
- Romanian
- Romanian Moldova
- Russian
- Sami Lappish
- Sanskrit
- Serbian Cyrillic
- Serbian Latin
- Sindhi
- Slovak
- Slovenian
- Sorbian
- Spanish Argentina
- Spanish Bolivia
- Spanish Chile
- Spanish Colombia
- Spanish Costa Rica
- Spanish Dominican Republic
- Spanish Ecuador
- Spanish El Salvador
- Spanish Guatemala
- Spanish Honduras
- Spanish Mexico
- Spanish Modern

- Spanish Nicaragua
- Spanish Panama
- Spanish Paraguay
- Spanish Peru
- Spanish Puerto Rico
- Spanish Traditional
- Spanish Uruguay
- Spanish Venezuela
- Sutu
- Swahili
- Swedish
- Swedish Finland
- Tajik
- Tamil
- Tatar
- Telugu
- Thai
- Tibetan
- Tsonga
- Tswana
- Turkish
- Turkmen
- Ukrainian
- Urdu
- Urdu India
- Uzbek Cyrillic
- Uzbek Latin
- Venda
- Vietnamese
- Welsh
- Xhosa
- Yiddish
- Zulu

By default: 'Romanian'

SetDefLangfe

Signature

function SetDefLangfe($Lang = '')

Description

Defines the default language ID for the Asian versions of Word.

Parameters

$Lang - a string representing the language to be used in the document.

Allowed values:

- Afrikaans
- Albanian
- Arabic
- Arabic Algeria
- Arabic Bahrain
- Arabic Egypt
- Arabic General
- Arabic Iraq
- Arabic Jordan
- Arabic Kuwait
- Arabic Lebanon
- Arabic Libya
- Arabic Morocco
- Arabic Oman
- Arabic Qatar
- Arabic Syria
- Arabic Tunisia
- Arabic U.A.E.
- Arabic Yemen
- Armenian
- Assamese
- Azeri Cyrillic
- Azeri Latin
- Basque
- Bengali
- Bosnia Herzegovina
- Bulgarian
- Burmese
- Byelorussian
- Catalan
- Chinese China
- Chinese General
- Chinese Hong Kong
- Chinese Macao
- Chinese Singapore
- Chinese Taiwan

- Croatian
- Czech
- Danish
- Dutch Belgium
- Dutch Standard
- English Australia
- English Belize
- English British
- English Canada
- English Caribbean
- English General
- English Ireland
- English Jamaica
- English New Zealand
- English Philippines
- English South Africa
- English Trinidad
- English United States
- English Zimbabwe
- Estonian
- Faeroese
- Farsi
- Finnish
- French
- French Belgium
- French Cameroon
- French Canada
- French Cote d'Ivoire
- French Luxemburg
- French Mali
- French Monaco
- French Reunion
- French Senegal
- French Swiss
- French West Indies
- French Zaire
- Frisian
- Gaelic
- Gaelic Ireland
- Galician
- Georgian
- German
- German Austrian
- German Liechtenstein
- German Luxemburg
- German Switzerland
- Greek
- Gujarati
- Hebrew
- Hindi
- Hungarian
- Icelandic
- Indonesian
- Italian
- Italian Switzerland

- Japanese
- Kannada
- Kashmiri
- Kashmiri India
- Kazakh
- Khmer
- Kirghiz
- Konkani
- Korean
- Korean Johab
- Lao
- Latvian
- Lithuanian
- Lithuanian Classic
- Macedonian
- Malay
- Malay Brunei Darussalam
- Malayalam
- Maltese
- Manipuri
- Marathi
- Mongolian
- Nepali
- Nepali India
- Norwegian Bokmal
- Norwegian Nynorsk
- Oriya
- Polish
- Portuguese Brazil
- Portuguese Iberian
- Punjabi
- Rhaeto-Romanic
- Romanian
- Romanian Moldova
- Russian
- Sami Lappish
- Sanskrit
- Serbian Cyrillic
- Serbian Latin
- Sindhi
- Slovak
- Slovenian
- Sorbian
- Spanish Argentina
- Spanish Bolivia
- Spanish Chile
- Spanish Colombia
- Spanish Costa Rica
- Spanish Dominican Republic
- Spanish Ecuador
- Spanish El Salvador
- Spanish Guatemala
- Spanish Honduras
- Spanish Mexico
- Spanish Modern

- Spanish Nicaragua
- Spanish Panama
- Spanish Paraguay
- Spanish Peru
- Spanish Puerto Rico
- Spanish Traditional
- Spanish Uruguay
- Spanish Venezuela
- Sutu
- Swahili
- Swedish
- Swedish Finland
- Tajik
- Tamil
- Tatar
- Telugu
- Thai
- Tibetan
- Tsonga
- Tswana
- Turkish
- Turkmen
- Ukrainian
- Urdu
- Urdu India
- Uzbek Cyrillic
- Uzbek Latin
- Venda
- Vietnamese
- Welsh
- Xhosa
- Yiddish
- Zulu

By default: 'Romanian'

SetDoccomm

Signature

function SetDoccomm($DocComments)

Description

Set the comments displayed in the Summary Info or Properties dialog box in Word.

Parameters

$DocComments - a string representing the Summary Inof of the document.

SetDocDirection

Signature

function SetDocDirection ($Direction = '')

Description

Specifies that the document will be formatted to have Arabic-style pagination or English-style pagination (the default).

Parameters

$Direction – direction of document formatting.

Allowed values:

- 'l | left | left to right | left_to_right | left_to right | left to_right | left-to-right | left-to right | left to-right' - Left to right
- 'r | right | right to left | right_to_left | right to_left | right_to left | right-to-left | right-to left | right to-left' -Right to left

By default : 'Left to right'

SetDocFormShade

Signature

function SetDocFormShade ()

Description

Specifies that the current document has form field shading on.

Parameters

SetDocRevisions

Signature

function SetDocRevisions ()

Description

Turns on revision marking.

Parameters

SetDocSource

Signature

function SetDocSource($DocSource = '')

Description

Set the source of the document.

Parameters

$DocSource - a string representing the source of the document. 'H' or a string who is begining with 'h' for HTML, anything else for text source.

SetDocTemp

Signature

function SetDocTemp()

Description

Specify that document is a boilerplate document. For Word for Windows, this is a template; for Word for the Macintosh, this is a stationery file.

Parameters

SetDocType

Signature

function SetDocType($DocType = '')

Description

Set the document type.

Parameters

$DocType – a string specifying the document type.

Allowed values:

- General Document | General | Document
- Letter
- E-mail | Email | Mail

By default: 'General Document'

SetDocumentView

Signature

function SetDocumentView ($Kind = '', $Scale = '', $Zoom = '')

Description

This method set the view mode, the zoom level and the zoom kind of the document.

Parameters

$Kind - represents the view mode of the document.

Allowed values:

- **'None'**
- **'Page Layout'**
- **'Outline'**
- **'Master Document' |' Master'**
- **'Normal'**
- **'Online Layout' | 'Online'**

By default : 'None'

$Scale - zoom level of the document; the argument is a value representing a percentage (the default is 100).

$Zoom - represents the zoom kind of the document

Allowed values:

- **'None'**
- **'Full Page' | 'Full' | 'Page'**
- **'Best Fit' | 'Best' | 'Fit'**

By default : 'None'

SetDrawGridHorizOrig

Signature

function SetDrawGridHorizOrig ($GridHorizOrig = '')

Description

Set the drawing grid horizontal origin in twips (the default is 1701).

Parameters

$GridHorizOrig – grid horizontal origin in twips - default value 1701.

SetDrawGridHorizSpace

Signature

function SetDrawGridHorizSpace ($GridHorizSpace = '')

Description

Set the drawing grid horizontal spacing in twips (the default is 120).

Parameters

$GridHorizSpace – grid horizontal spacing in twips - default value 120.

SetDrawGridVertOrig

Signature

function SetDrawGridVertOrig ($GridVertOrig = '')

Description

Set the drawing grid vertical origin in twips (the default is 1984).

Parameters

$GridVertOrig – grid vertical origin in twips - default value 1984.

SetDrawGridVertSpace

Signature

function SetDrawGridVertSpace ($GridVertSpace = '')

Description

Set the drawing grid vertical spacing in twips (the default is 120).

Parameters

$GridVertSpace – grid vertical spacing in twips - default value 120.

SetEditionTime

Signature

function SetEditionTime($dtNewTime = '')

Description

Set the creation time of the document.

Parameters

$dtNewTime - the new time you want to became the edit time of the document.

SetFacingp

Signature

function SetFacingp()

Description

Set facing pages for the document. Activates odd/even headers and gutters.

Parameters

SetFormDisp

Signature

function SetFormDisp ()

Description

Specifies that the document currently has a forms drop-down box or check box selected.

Parameters

SetFormPrintData

Signature

function SetFormPrintData ()

Description

Specifies that the document has print form data only on.

Parameters

SetFracWidth

Signature

function SetFracWidth()

Description

Set the fractional character widths when printing (QuickDraw only).

Parameters

SetFromHtml

Signature

function SetFromHtml()

Description

Set document source to HTML. Indicates document was originally HTL and may contain encapsulated HTML tags.

Parameters

SetFromText

Signature

function SetFromText()

Description

Set document source to text. Indicates document was originally plain text.

Parameters

SetGutter

Signature

function SetGutter($GutterWidth = '')

Description

Set the gutter width in twips.

Parameters

$GutterWidth - a number representing the gutter width in twips.

SetGutterPos

Signature

function SetGutterPos($GutterPos = '')

Description

Set the gutter position.

Parameters

$GutterPos - a string representing the gutter position. 'T' or a string beginning with 'T' for top position, anything else for right position.

SetHlinkbase

Signature

function SetHlinkbase($HLinkBase)

Description

Set the base address that is used for the path of all relative hyperlinks inserted in the document.

Parameters

$HLinkBase - a string containing the base address.

SetHorzDoc

Signature

function SetHorzDoc()

Description

Set the rendering of the document to Horizontal.

Parameters

SetHyphenAuto

Signature

function SetHyphenAuto($HyphenAuto = '')

Description

Toggles automatic hyphenation.

Parameters

$HyphenAuto - use '1 (one)' to toggle the property on, anything else to turn it off.

SetHyphenCaps

Signature

function SetHyphenCaps($HyphenCaps = '')

Description

Toggles hyphenation of capitalized words.

Parameters

$HyphenCaps - use '1 (one)' to toggle the property on, anything else to turn it off.

SetHyphenConsec

Signature

function SetHyphenConsec($HyphenConsec = '')

Description

Set the maximum number of lines that will be allowed to end in a hyphen.

Parameters

$HyphenConsec - an integer representing the maximum number of consecutive lines that will be allowed to end in a hyphen. 0 means no limit.

SetHyphenHot

Signature

function SetHyphenHot($HyphenHot = '')

Description

Set the hyphenation hot zone in twips (the amount of space at the right margin in which words are hyphenated).

Parameters

$HyphenHot - an integer representing the hyphenation hot zone in twips. By default is 72.

SetKeywords

Signature

function SetKeywords($Keywords = '')

Description

Set the selected keywords for the document.

Parameters

$Keywords - a string containing the keywords of the document.

SetLineStart

Signature

function SetLineStart($LineStartNb = '')

Description

Set the beginning line number.

Parameters

$LineStartNb - an integer representing the beginning line number for the document.

SetMakeBackup

Signature

function SetMakeBackup()

Description

Set the 'Make Backup' property. Backup copy is made automatically when the document is saved.

Parameters

SetManager

Signature

function SetManager($Manager)

Description

Set the manager of the author.

Parameters

$Manager - a sring representing the name of the manager of the author of the document.

SetMargb

Signature

function SetMargb($Margin = '')

Description

Set the bottom margin of the page in twips.

Parameters

$Margin - an integer representing the bottom margin of the page in twips. The default value is 1440.

SetMargins

Signature

function SetMargins($Left = '', $Right = '', $Top = '', $Bottom = '')

Description

Set the margins of the page in twips.

Parameters

$Left - an integer representing the left margin of the page in twips. The default value is 1800.

$Right - an integer representing the right margin of the page in twips. The default value is 1800.

$Top - an integer representing the top margin of the page in twips. The default value is 1440.

$Bottom - an integer representing the left margin of the page in twips. The default value is 1440.

SetMargl

Signature

function SetMargl($Margin = '')

Description

Set the left margin of the page in twips.

Parameters

$Margin - an integer representing the left margin of the page in twips. The default value is 1800.

SetMargmirror

Signature

function SetMargmirror()

Description

Switches margin definitions on left and right pages. Used in conjunction with SetFancingp**.**

Parameters

SetMargr

Signature

function SetMargr($Margin = '')

Description

Set the right margin of the page in twips.

Parameters

$Margin - an integer representing the right margin of the page in twips. The default value is 1800.

SetMargt

Signature

function SetMargt($Margin = '')

Description

Set the top margin of the page in twips.

Parameters

$Margin - an integer representing the top margin of the page in twips. The default value is 1440.

SetNextFile

Signature

function SetNextFile($NextFile = '')

Description

Set the name of the file to print or index next.

Parameters

$NextFile - a string representing the name of the file to print or index next.

SetOperator

Signature

function SetOperator($Operator = '')

Description

Set the person who last made changes to the document.

Parameters

$Operator - a string representing the name of operator.

SetPageBorderFoot

Signature

function SetPageBorderFoot()

Description

Tells RTF reader that page borders surrounds.

Parameters

SetPageBorderHead

Signature

function SetPageBorderHead()

Description

Tells RTF reader that page borders surrounds header.

Parameters

SetPageBorderOption

Signature

function SetPageBorderOption($BorderOption = '')

Description

Set the reference for the page border measurement.

Parameters

$BorderOption - an integer specifying the option.

Allowed values:

- 8 -> Page border measure from text. Always display in front option is set to off.
- 32 -> Page border measure from edge of page. Always display in front option is set to on.
- 4032 -> Page border measure from edge of page. Always display in front option is set to off.

By default: '8'

SetPageBorders

Signature

function SetPageBorders($BrdPos = '', $BrdType = '', $BrdPen = '', $BrdColor = '', $BrdSpace = '', $BrdArt = '', $BrdOpt = '', $BrdAlign = '')

Description

Set the page borders for the RTF document.

Parameters

$BrdPos - a string representing the position or the positions the border/borders will be defined for.

Allowed values:

't'	Top.
'b'	Bottom
'l'	Left.
'r'	Right
'tblr' | 'tbl' | 'tb' | 'bl' | 'lr' | 'tl' etc - any combination	

By default: 'tblr'

$BrdType - a string specifying the border style.

Allowed values:

'Single-thickness | Single thickness | Single'	Single-thickness border.
'Double-thickness | Double thickness'	Double-thickness border.
'Shadowed'	Shadowed border.
'Double'	Double border.
'Dotted'	Dotted border.
'Dotted'	Dotted border.
'Dashed'	Dashed border.
'Hairline'	Hairline border.
'Resembles a frame | Frame'	Border resembles a “Frame.”
'No border specified | No'	No border specified.
'Table cell has no borders | table cell has no'	Table cell has no borders.
'Inset'	Inset border.
'Dashed small'	Dashed small.
'Dot-dashed | Dot dashed'	Dot-dashed border.
'Dot-dot-dashed | Dot-dot dashed | Dot dot-dashed | Dot dot dashed'	Dot-dot-dashed border.
'Outset'	Outset border.
'Triple'	Triple border.
'Thick-thin small | Thick thin small'	Thick-thin border (small).
'Thin-thick small | Thin thick small'	Thin-thick border (small).
'Thin-thick thin small | Thin thick thin small'	Thin-thick thin border (small).
'Thick-thin medium | Thick thin medium'	Thick-thin border (medium).
'Thin-thick medium | Thin thick medium'	Thin-thick border (medium).
'Thin-thick thin medium | Thin thick thin medium'	Thin-thick thin border

(medium).

'Thick-thin large | Thick thin large'	Thick-thin border (large).
'Thin-thick large | Thin thick large'	Thin-thick border (large).
'Thin-thick thin large | Thin-thick thin large | Thin thick-thin large | Thin thick thin large'	Thin-thick-thin border (large).
'Wavy'	Wavy border.
'Double wavy'	Double wavy border.
'Striped'	Striped border.
'Embossed'	Embossed border.
'Engraved'	Engraved border.

By default: 'No border specified'

$BrdPen - this argument is representing the border width in twips. It is the width in twips of the pen used to draw the paragraph border line and cannot be greater than 75. To obtain a larger border width, please prefix your width with the letter 'D' and you will obtain a width double that the one you had specified. Ex: 'D60' will draw a border with 120 twips in width. If null or missing by default the width will 1.

$BrdColor - a string representing the color code of the border to be drawn. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a border with red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'.

$BrdArt - this argument is a number between 1 and 165 representing the number of the border art of the page. By default is null that means no border art will be used to draw the border of the page.

$BrdSpace - this argument specify the space in twips between borders and the paragraph. By default the space is 480 twips.

$BrdOpt - an integer specifying the reference for the page border measurement.

Allowed values:

- 8 -> Page border measure from text. Always display in front option is set to off.
- 32 -> Page border measure from edge of page. Always display in front option is set to on.
- 4032 -> Page border measure from edge of page. Always display in front option is set to off.

By default: '8'

SetPageOrientation

Signature

function SetPageOrientation($Orientation = '')

Description

This method set the page orientation for the RTF document. By default the page orientation is portrait and it is not necessary to call this method with the 'Portrait' parameter in order to specify it.

Parameters

$Orientation - specify the orientation of the page Portrait/Landscape

Allowed values:

- ‘L’,’l’ or anything that starts with ‘L’,‘l’ for Landscape
- Anything else for Portrait

By default: 'Portrait'

SetPaperh

Signature

function SetPaperh($Paperh = '')

Description

Set the paper height in twips.

Parameters

$Paperh - a number representing the paper height in twips. By default the paper height is 15840 twips.

SetPaperSize

Signature

function SetPaperSize ($PaperFormat = '')

Description

Define the size of the paper for the RTF document. The size can be specified by passing as parameter the paper format. You can also define your custom paper size by using SetPaperh and SetPaperw.

Parameters

$PaperFormat - the type of paper used for this document

Allowed values:

Letter	(8.5 x 11 in.)
Letter Small	(8.5 x 11 in.)
Note	(8.5 x 11 in.)
Tabloid	(11 x 17 in.)
11 x 17 in	(11x17 in.)
Ledger	(17 x 11 in.)
Legal	(8.5 x 14 in.)
Statement	(5.5 x 8.5 in.)
Executive	(7.25 x 10.5 in.)
A3	(297 x 420 mm)
A4	(210 x 297 mm)
A4 Small	(210 x 297 mm)
A5	(148 x 210 mm)
B4 (jis)	(250 x 354)
B5 (jis)	(182 x 257 mm)
Folio	(8.5 x 13 in.)
Quarto	(215 x 275 mm)
10x14 in	(10x14 in.)
Envelope #9	(3.875 x 8.875 in.)
Envelope #10	(4.125 x 9.5 in.)
Envelope #11	(4.5 x 10.375 in.)
Envelope #12	(4.75 x 11 in.)
Envelope #14	(5 x 11.5 in.)
C size sheet	(17 x 22 in.)
D size sheet	(22 x 34 in.)
E size sheet	(34 x 44 in.)
Envelope DL	(110 x 220 mm)
Envelope C3	(324 x 458 mm)
Envelope C4	(229 x 324 mm)
Envelope C5	(162 x 229 mm)
Envelope C6	(114 x 162 mm)
Envelope C65	(114 x 229 mm)
Envelope B4	(250 x 353 mm)
Envelope B5	(176 x 250 mm
Envelope B6	(176 x 125 mm)

Envelope	(110 x 230 mm)
Envelope Monarch	(3.875 x 7.5 in.)
6-3/4 Envelope	(3.625 x 6.5 in.)
US Std Fanfold	(14.875 x 11 in.)
German Std Fanfold	(8.5 x 12 in.)
German Legal Fanfold	(8.5 x 13 in.)
A6	(105x142 mm)
index card 4x6 in	(4x6 in.)
index card 5x8 in	(5x8 in.)
hagaki card 100x148 mm	(100x 148 mm)

By default : 'Letter'

SetPaperw

Signature

function SetPaperw($Paperw = '')

Description

Set the paper width in twips.

Parameters

$Paperw - a number representing the paper width in twips. By default the paper width is 12240 twips.

SetParBorderAlign

Signature

function SetParBorderAlign()

Description

Align paragraph borders and table edges with page border..

Parameters

SetPgnStart

Signature

function SetPgnStart($PageNb = '')

Description

Set the beginning page number.

Parameters

$PageNb - a number representing the beginning page number of the document. By default is 1.

SetPrintTime

Signature

function SetPrintTime($dtNewTime)

Description

Set the last print time of the document.

Parameters

$dtNewTime - the date time when document was last printed. Ex: '12/23/2004 20:45:53'.

SetPsOver

Signature

function SetPsOver()

Description

Set the Print PostScrip over the text property.

Parameters

SetRender

Signature

function SetRender($Render = '')

Description

Set the rendering of the document.

Parameters

$Render - a string specifying the rendering style of the document. Any string beginning with 'V' for vertical rendering. Anything else even null or missing argument for horizontal rendering.

SetRevisionBar

Signature

function SetRevisionBar ($RevBar = '')

Description

Specifies how a vertical lines mark altered text, based on the argument:

Parameters

$RevBar – indicates how vertical line mark altered text:

Allowed values:

No	for no properties shown
Left	for bold
Right	for italic
Outside	for underline (the default)

By default : 'No'

SetRevisionProp

Signature

function SetRevisionProp ($RevProp = '')

Description

Specifies how the revised text will be displayed.

Parameters

$RevProp – indicates how revised text will be displayed:

Allowed values:

No	**for no properties shown**
Bold	**for bold**
Italic	**for italic**
Underline	**for underline (the default)**
Double	**for double underline**

By default : 'Underline'

SetRevisionTime

Signature

function SetRevisionTime($dtNewTime)

Description

Set the revision time of the document.

Parameters

$dtNewTime - the date time when document was last revised. Ex: '12/23/2004 20:45:53'.

SetShowHorizGrid

Signature

function SetShowHorizGrid ($ShowHorizOrig = '')

Description

Show Nth horizontal gridline (the default is 3).

Parameters

$ShowHorizOrig – the Nth horizontal gridline - default value 3.

SetShowVertGrid

Signature

function SetShowVertGrid ($ShowVertOrig = '')

Description

Show Nth vertical gridline (the default is 0).

Parameters

$ShowVertOrig – the Nth vertical gridline - default value 0.

SetSubject

Signature

function SetSubject($Subject = '')

Description

Set the subject of the document.

Parameters

$Subject - a string specifying the subject of the document.

SetTemplate

Signature

function SetTemplate($Template = '')

Description

Set the name of the related template file.

Parameters

$Template - a string specifying the name of the template file.

SetTitle

Signature

function SetTitle($Title = '')

Description

Set the title of the document.

Parameters

$Title - a string specifying the title of the document.

SetTwoonone

Signature

function SetTwoonone($Title = '')

Description

Print two logical pages on one physical page.

Parameters

SetUnicodeCorresp

Signature

function SetUnicodeCorresp ($ByteNr)

Description

Set the number of bytes corresponding to a given Unicode character.

Parameters

$ByteNr – **the number of bytes for a given Unicode char.**

SetUniCodePage

Signature

function SetUniCodePage ($UnicodePage = '')

Description

Set the unicode code page.

Parameters

$UnicodePage – the index to the style to be used - default value 0.

Allowed values:

- "united states ibm"
- "arabic (asmo 708)"
- "arabic (asmo 449+, bcon v4)"
- "arabic (transparent arabic)"
- "arabic (nafitha enhanced)"
- "arabic (transparent asmo)"
- "windows 3.1 (united states and western europe)"
- "ibm multilingual"
- "eastern european"
- "portuguese"
- "hebrew_1"
- "french canadian"
- "arabic_1"
- "norwegian"
- "soviet union"
- "thai"
- "japanese"
- "simplified chinese"
- "korean"
- "traditional chinese"
- "windows 3.1 (eastern european)"
- "windows 3.1 (cyrillic)"
- "western european"
- "greek"
- "turkish"
- "hebrew"
- "arabic"
- "baltic"
- "vietnamese"
- "johab"

By default : 'Western European'

SetVersionDoc

Signature

function SetVersionDoc($Version = '')

Description

Set the version number of the document.

Parameters

$Version - a string representing the version of the document.

SetVertDoc

Signature

function SetVertDoc()

Description

Set the rendering of the document to Vertical.

Parameters

SetViewKind

Signature

function SetViewKind($ViewKind = '')

Description

This method set the view mode of the document.

Parameters

$ViewKind - represents the view mode of the document.

Allowed values:

- 'None'
- 'Page Layout | Page'
- 'Outline'
- 'Master Document' |' Master'
- 'Normal'
- 'Online Layout' | 'Online'

By default : 'None

SetViewScale

Signature

function SetViewScale($Scale = '')

Description

This method set the zoom level of the document.

Parameters

$Scale **-** zoom level of the document; the argument is a value representing a percentage (the default is 100).

SetViewZk

Signature

function SetViewZk ($ZoomKind = '')

Description

This method set the zoom kind of the document.

Parameters

$ZoomKind - represents the zoom kind of the document

Allowed values:

- **'None'**
- **'Full Page' | 'Full' | 'Page'**
- **'Best Fit' | 'Best' | 'Fit'**

By default : 'None'

SetWidowctrl

Signature

function SetWidowctrl()

Description

Enable widow and orphan control.

Parameters

SetWindowCaption

Signature

function SetWindowCaption($Caption = '')

Description

This method sets the caption text for the document window.

Parameters

$Caption - the string to be displayed in the window caption.

SnapToDrawingGrid

Signature

function SnapToDrawingGrid ()

Description

Snap to drawing grid.

Parameters

Section formatting methods

1. AddNewSection
2. Sect_BeginLineNb
3. Sect_Break
4. Sect_CharSpaceBasement
5. Sect_ColNumber
6. Sect_ColWidth
7. DefineEndNote
8. DefineFootNote
9. Sect_DistLineNbTextLeft
10. Sect_FirstPage
11. Sect_IncludeEndNote
12. Sect_LineBetCols
13. Sect_LineGrid
14. Sect_LineModulus
15. Sect_LineNbRestart
16. Sect_NbColForSnaking
17. Sect_PageNbIncludeChapter
18. Sect_PageNbRestartPolicy
19. Sect_PageNbStyle
20. Sect_PageNumberFormat
21. Sect_PageOrientation
22. Sect_PrintBin
23. Sect_PrintBinFirst
24. Sect_ResetToDefaultProp
25. Sect_SetBidirectional
26. Sect_SetBorderArt
27. Sect_SetFooterY
28. Sect_SetGutter
29. Sect_SetHeaderY
30. Sect_SetMargb
31. Sect_SetMargins
32. Sect_SetMargl
33. Sect_SetMargmirror
34. Sect_SetMargr
35. Sect_SetMargt
36. Sect_SetPageBorderFoot
37. Sect_SetPageBorderHead
38. Sect_SetPageBorderOption
39. Sect_SetPageBorders
40. Sect_SetPageNbRight
41. Sect_SetPageNbTop
42. Sect_SetPaperh
43. Sect_SetPaperSize
44. Sect_SetPaperw
45. Sect_SetParBorderAlign
46. Sect_SetRender
47. Sect_SpaceBetCols
48. Sect_SpaceToRightOfCol
49. Sect_StateOf

50. Sect_TextFlow
51. Sect_TextShouldSnap
52. Sect_UnlockedForForms
53. Sect_VertAlign

AddNewSection

Signature

function AddNewSection ($DefaultProp = 'Yes', $Break = '')

Description

Add a new section to the RTF document.

Parameters

$DefaultProp – **if Yes reset to default section properties; if No, the current section inherits all section properties defined in the previous section.**

$Break – describe the break preceding the text.

Allowed values:

- None | N | Nimic
- New column | Column
- New page | Page
- Even page | E page | Even
- Odd page | O page | Odd

By default : 'New page'

Sect_BeginLineNb

Signature

function Sect_BeginLineNb ($LineNb = '')

Description

Set the beginning line number for the current section.

Parameters

$LineNb – **a number representing the line number. By default is 1.**

Sect_Break

Signature

function Sect_Break ($Break = '')

Description

Describe the break preceding the text of the current section.

Parameters

$Break – a string specifying the break preceding the text of the section

Allowed values:

- None | N | Nimic
- New column | Column
- New page | Page
- Even page | E page | Even
- Odd page | O page | Odd

By default : 'New page'

Sect_CharSpaceBasement

Signature

function Sect_CharSpaceBasement ($CharSpace = '')

Description

Set the character space basement (character pitch minus font size).

Parameters

$CharSpace – in device-independent units (a device-independent unit is 1/294912th of an inch).

Sect_ColNumber

Signature

function Sect_ColNumber ($ColNb = '')

Description

Column number to be formatted; used to specify formatting for variable-width columns.

Parameters

$ColNb – column number. By default is 1.

Sect_ColWidth

Signature

function Sect_ColWidth ($ColWidth = '')

Description

Set the width of column in twips; used to override the default constant width setting for variable-width columns.

Parameters

$ColWidth – the width of column in twips. By default is 720.

Sect_DistLineNbTextLeft

Signature

function Sect_DistLineNbTextLeft ($Dist = '')

Description

Set the distance from the line number to the left text margin in twips. The automatic distance is 0..

Parameters

$Dist – a number representing the distance from the line number to the left text margin in twips. By default is 360.

Sect_DefineEndNote

Signature

function Sect_DefineEndNote ($AEBeginNumb = '', $AENumbPolicy = '',$AENumbStyle = '')

Description

This method define the endnote type for the current section.

Parameters

$AEBeginNumb – **beginning endnote number - default value 1.**

$AENumbPolicy – **a string specifying the endnote restart number policy.**

Allowed values:

- **'page | pag | p | pagina'** **restart numbering each page**
- **'section | sec | s | sectiune'** **restart numbering each section**
- **'continuous | cont | c | continuu'** **continuous footnote numbering**

By default : 'Continuous'

$AENumbStyle – **a string specifying the endnote numbering style.**

Allowed values:

- **'arabic'** —Arabic numbering (1, 2, 3, ¼).
- **'alphabetic lowercase | alfabetic mic | a l | al | a mic | amic'** —Alphabetic lowercase (a, b, c, ¼).
- **'alphabetic uppercase | alfabetic mare | a u | au | a mare | amare'** —Alphabetic uppercase (A, B, C, ¼).
- **'roman lowercase | roman mic | r l | rl | r mic | rmic'** —Roman lowercase (i, ii, iii, ¼).
- **'roman uppercase | roman mare | r u | ru | r mare | rmare'** —Roman uppercase (I, II, III, ¼).
- **'chicago'** —Chicago Manual of Style (*, †, §, §).
- **'korean 1 | korean1 | kor1'** —Korean numbering 1 (*chosung).
- **'korean 2 | korean2 | kor2'** —Korean numbering 2 (*ganada).
- **'circle'** —Circle numbering (*circlenum).
- **'kanji 1 | kanji1 | k1'** —kanji numbering without the digit character (*dbnum1).
- **'kanji 2 | kanji2 | k2'** —kanji numbering with the digit character (*dbnum2).
- **'kanji 3 | kanji3 | k3'** —kanji numbering 3 (*dbnum3).
- **'kanji 4 | kanji4 | k4'** —kanji numbering 4 (*dbnum4).
- **'double_byte | double byte'** —double-byte numbering (*dbchar).
- **'chinese 1 | chi 1 | c1'** —Chinese numbering 1 (*gb1).
- **'chinese 2 | chi 2 | c2'** —Chinese numbering 2 (*gb2).
- **'chinese 3 | chi 3 | c3'** —Chinese numbering 3 (*gb3).
- **'chinese 4 | chi 4 | c4'** —Chinese numbering 4 (*gb4).

- **'chinese zodiac 1 | zodiac 1 | zodiac1 | z1'** —Chinese Zodiac numbering 1 (* zodiac1). 甲、乙、丙… ~~甲、乙、丙…~~ 甲、乙、丙…
- **'chinese zodiac 2 | zodiac 2 | zodiac2 | z2'**—Chinese Zodiac numbering 2 (* zodiac2). 子、丑、寅…
- **'chinese zodiac 3 | zodiac 3 | zodiac3 | z3'**—Chinese Zodiac numbering 3 (* zodiac3).

By default : 'Arabic'

Sect_DefineFootNote

Signature

function Sect_DefineFootNote($FNJustif = '', $FNBeginNumb = '', $FNNumbPolicy = '', $FNNumbStyle = '')

Description

This method define the footnote type for the current section.

Parameters

$FNJustify – **a string specifying the footnote alignment.**

Allowed values:

- **top | t | beneath text** **footnote is top justified**
- **bottom | bot | b | bottom of page** **footnote is bottom justified**

By default : 'Bottom of page'

$FNBeginNumb – **beginning footnote number - default value 1.**

$FNNumbPolicy – **a string specifying the footnote restart number policy.**

Allowed values:

- **'page | pag | p | pagina'** **restart numbering each page**
- **'section | sec | s | sectiune'** **restart numbering each section**
- **'continuous | cont | c | continuu'** **continuous footnote numbering**

By default : 'Continuous'

$FNNumbStyle – **a string specifying the footnote numbering style.**

Allowed values:

- **'arabic'** —Arabic numbering (1, 2, 3, ¼).
- **'alphabetic lowercase | alfabetic mic | a l | al | a mic | amic'** —Alphabetic lowercase (a, b, c, ¼).
- **'alphabetic uppercase | alfabetic mare | a u | au | a mare | amare'** —Alphabetic uppercase (A, B, C, ¼).
- **'roman lowercase | roman mic | r l | rl | r mic | rmic'** —Roman lowercase (i, ii, iii, ¼).
- **'roman uppercase | roman mare | r u | ru | r mare | rmare'** —Roman uppercase (I, II, III, ¼).
- **'chicago'** —Chicago Manual of Style (*, †, §, §).
- **'korean 1 | korean1 | kor1'** —Korean numbering 1 (*chosung).
- **'korean 2 | korean2 | kor2'** —Korean numbering 2 (*ganada).
- **'circle'** —Circle numbering (*circlenum).
- **'kanji 1 | kanji1 | k1'** —kanji numbering without the digit character

(*dbnum1).

- **'kanji 2 | kanji2 | k2'** —kanji numbering with the digit character (*dbnum2).
- **'kanji 3 | kanji3 | k3'** —kanji numbering 3 (*dbnum3).
- **'kanji 4 | kanji4 | k4'** —kanji numbering 4 (*dbnum4).
- **'double_byte | double byte'** —double-byte numbering (*dbchar).
- **'chinese 1 | chi 1 | c1'** —Chinese numbering 1 (*gb1).
- **'chinese 2 | chi 2 | c2'** —Chinese numbering 2 (*gb2).
- **'chinese 3 | chi 3 | c3'** —Chinese numbering 3 (*gb3).
- **'chinese 4 | chi 4 | c4'** —Chinese numbering 4 (*gb4).
- **'chinese zodiac 1 | zodiac 1 | zodiac1 | z1'** —Chinese Zodiac numbering 1 (* zodiac1). 甲、乙、丙… ~~甲、乙、丙…~~ 甲、乙、丙…
- **'chinese zodiac 2 | zodiac 2 | zodiac2 | z2'**—Chinese Zodiac numbering 2 (* zodiac2). 子、丑、寅…
- **'chinese zodiac 3 | zodiac 3 | zodiac3 | z3'**—Chinese Zodiac numbering 3 (* zodiac3).

By default : 'Arabic'

Sect_FirstPage

Signature

function Sect_FirstPage ()

Description

Tells RTF reader the first page has a special format.

Parameters

Sect_IncludeEndNote

Signature

function Sect_IncludeEndNote ()

Description

Tells RTF reader the current section contains endnotes.

Parameters

Sect_LineBetCols

Signature

function Sect_LineBetCols ()

Description

Tells RTF reader it will be lines between columns.

Parameters

Sect_LineGrid

Signature

function Sect_LineGrid ($LineGrid = '')

Description

Set the line grid of the current section.

Parameters

$LineGrid – a number specifying the line pitch in 20ths of a point.

Sect_LineModulus

Signature

function Sect_LineModulus ($LineMod = '')

Description

Set the line-number modulus amount to increase each line number.

Parameters

$LineMod – a number specifying the line-number modulus. By default is 1.

Sect_LineNbRestart

Signature

function Sect_LineNbRestart ($LineNbRestart = '')

Description

Set the line numbers restart policy for the current section.

Parameters

$LineNbRestart – a string specifying the line numbers restart policy.

Allowed values:

‘Restart’	Line numbers restart at the value specified by Sect_BeginLineNb
'Page'	Line numbers restart on each page
'Continue'	Line numbers continues from the preceding section

By default: 'Restart'

Sect_NbColForSnaking

Signature

function Sect_NbColForSnaking ($LNbCol = '')

Description

Set the number of columns for "snaking".

Parameters

$NbCol – the number of columns. By default is 1.

Sect_PageNbIncludeChapter

Signature

function Sect_PageNbIncludeChapter ($ChapterStyle = '', $Separator = '')

Description

Indicates which heading level is used to prefix a heading number to the page number of the current section. Also describe the separator which appear between the heading level number and the page number of the current section.

Parameters

$ChapterStyle – a string containing the name of the heading level.

Allowed values:

heading 1 | heading1 | head 1 | head1 | h 1 | h1	Heading 1
heading 2 | heading2 | head 2 | head2 | h 2 | h2	Heading 2
heading 3 | heading3 | head 3 | head3 | h 3 | h3	Heading 3
heading 4 | heading4 | head 4 | head4 | h 4 | h4	Heading 4
heading 5 | heading5 | head 5 | head5 | h 5 | h5	Heading 5
heading 6 | heading6 | head 6 | head6 | h 6 | h6	Heading 6
heading 7 | heading7 | head 7 | head7 | h 7 | h7	Heading 7
heading 8 | heading8 | head 8 | head8 | h 8 | h8	Heading 8
heading 9 | heading9 | head 9 | head9 | h 9 | h9	Heading 9

By default: 'Heading 1'

$Separator – a string containing the separator between the heading level number and the page number.

Allowed values:

Hyphen | H	Hyphen separator character
Period | P	Period separator character
Colon | C	Colon separator character
Em dash | Em	Em dash (—) separator character
En dash | En	En dash (–) separator character

By default: 'Hyphen'

Sect_PageNbRestartPolicy

Signature

function Sect_PageNbRestartPolicy ($Rule = '',$Numb = '')

Description

Set the section page number restart policy for the current section.

Parameters

$Rule – a string containing the rule of the page number restart policy.

Allowed values:

Continue | Cont | C	Continue page numbering
Restart | Rest | R	Page number restart at $Numb value

By default: 'Continue'

$Numb – a number representing the beginning page number. If null or missing then default value is 1

Sect_PageNbStyle

Signature

function Sect_PageNbStyle ($PageNbStyle = '')

Description

Set the page number style for the current section.

Parameters

$PageNbStyle – a string containing the page number style.

Allowed values:

- Decimal'
- Roman uppercase | roman u | r u | ru | roman mare | rmare
- Roman lowercase | roman l | r l | rl | roman mic | rmic
- Letter uppercase | letter u | l u | lu
- Letter lowercase | letter l | l l | ll
- Abjad jawaz | biblical standard
- Alif ba tah | non-standard decimal
- Korean 1 | chosung
- Circle
- Kanji 1
- kanji 2
- Kanji 3
- Kanji 4
- Double-byte decimal
- Korean 2 | ganada
- Chinese 1
- Chinese 2
- Chinese 3
- Chinese 4
- Chinese zodiac 1
- Chinese zodiac 2
- Chinese zodiac 3
- Hindi 1 | hindi vowel
- Hindi 2 | hindi consonants
- Hindi 3 | hindi digits
- Hindi 4 | hindi descriptive | hindi cardinal
- Thai 1 | thai letters
- Thai 2 | thai digits
- Thai 3 | thai descriptive
- Vietnamese descriptive
- Korean 3 | dashes

By default: 'Decimal'

Sect_PageNumberFormat

Signature

function Sect_PageNumberFormat ($PageNbStyle = '', $ChapterStyle = '', $Separator = '', $RestartRule = '', $RestartNumb = '')

Description

Set the page number format for the current section.

Parameters

$PageNbStyle – a string containing the page number style.

Allowed values:

- Decimal'
- Roman uppercase | roman u | r u | ru | roman mare | rmare
- Roman lowercase | roman l | r l | rl | roman mic | rmic
- Letter uppercase | letter u | l u | lu
- Letter lowercase | letter l | l l | ll
- Abjad jawaz | biblical standard
- Alif ba tah | non-standard decimal
- Korean 1 | chosung
- Circle
- Kanji 1
- kanji 2
- Kanji 3
- Kanji 4
- Double-byte decimal
- Korean 2 | ganada
- Chinese 1
- Chinese 2
- Chinese 3
- Chinese 4
- Chinese zodiac 1
- Chinese zodiac 2
- Chinese zodiac 3
- Hindi 1 | hindi vowel
- Hindi 2 | hindi consonants
- Hindi 3 | hindi digits
- Hindi 4 | hindi descriptive | hindi cardinal
- Thai 1 | thai letters
- Thai 2 | thai digits
- Thai 3 | thai descriptive
- Vietnamese descriptive
- Korean 3 | dashes

By default: 'Decimal'

$ChapterStyle – a string containing the name of the heading level.

Allowed values:

heading 1 | heading1 | head 1 | head1 | h 1 | h1	Heading 1
heading 2 | heading2 | head 2 | head2 | h 2 | h2	Heading 2
heading 3 | heading3 | head 3 | head3 | h 3 | h3	Heading 3
heading 4 | heading4 | head 4 | head4 | h 4 | h4	Heading 4
heading 5 | heading5 | head 5 | head5 | h 5 | h5	Heading 5
heading 6 | heading6 | head 6 | head6 | h 6 | h6	Heading 6
heading 7 | heading7 | head 7 | head7 | h 7 | h7	Heading 7
heading 8 | heading8 | head 8 | head8 | h 8 | h8	Heading 8
heading 9 | heading9 | head 9 | head9 | h 9 | h9	Heading 9

By default: 'Heading 1'

$Separator – a string containing the separator between the heading level number and the page number.

Allowed values:

Hyphen | H	Hyphen separator character
Period | P	Period separator character
Colon | C	Colon separator character
Em dash | Em	Em dash (—) separator character
En dash | En	En dash (–) separator character

By default: 'Hyphen'

$Rule – a string containing the rule of the page number restart policy.

Allowed values:

Continue | Cont | C	Continue page numbering
Restart | Rest | R	Page number restart at $Numb value

By default: 'Continue'

$Numb – a number representing the beginning page number. If null or missing then default value is 1

Sect_PageOrientation

Signature

function Sect_PageOrientation($Orientation = '')

Description

This method set the page orientation for the current section of the RTF document. By default the page orientation is portrait and it is not necessary to call this method with the 'Portrait' parameter in order to specify it.

Parameters

$Orientation - specify the orientation of the page Portrait/Landscape

Allowed values:

- **‘L’,’l’ or anything that starts with ‘L’,‘l’ for Landscape**
- **Anything else for Portrait**

By default: 'Portrait'

Sect_PrintBin

Signature

function Sect_PrintBin ($PrintBin = '')

Description

Set the printer bin used for the pages of the section.

Parameters

$PrintBin - number of the printer bin. By default is 0.

Sect_PrintBinFirst

Signature

function Sect_PrintBinFirst ($PrintBin = '')

Description

Set the printer bin used for the first page of the section.

Parameters

$PrintBin - number of the printer bin. By default is 0.

Sect_ResetToDefaultProp

Signature

function Sect_ResetToDefaultProp ()

Description

Reset to default section properties.

Parameters

Sect_SetBidirectional

Signature

function Sect_SetBidirectional ($Bidi = '')

Description

Set the bi-directional support for the current section.

Parameters

$Bidi – a string containing the bi-directional support

Allowed values:

- 'l | left | left to right | left_to_right | left_to right | left to_right | left-to-right | left-to right | left to-right'
  Left to right
- 'r | right | right to left | right_to_left | right to_left | right_to left | right-to-left | right-to left |right to-left'
  Right to left

By default: 'Left to right'.

Sect_SetBorderArt

Signature

function Sect_SetBorderArt($BorderArt = '')

Description

Set the page border art for the current section.

Parameters

$BorderArt - the argument is a value from 1 to 165 representing the number of the border. By default is 1.

Sect_SetFooterY

Signature

function Sect_SetFooterY ($Marg = '')

Description

Set the footer margin = distance from the footer to the bottom of the page for the current section .

Parameters

$Marg – a number representing the distance in twips from the footer to the bottom of the page (the default is 720)

Sect_SetGutter

Signature

function Sect_SetGutter($GutterWidth = '')

Description

Set the gutter width in twips for the current section.

Parameters

$GutterWidth - a number representing the gutter width in twips.

Sect_SetHeaderY

Signature

function Sect_SetHeaderY ($Marg = '')

Description

Set the header margin = distance from the heder to the top of the page for the current section .

Parameters

$Marg – a number representing the distance in twips from the header to the top of the page (the default is 720)

Sect_SetMargb

Signature

function Sect_SetMargb($Margin = '')

Description

Set the bottom margin of the page in twips for the current section.

Parameters

$Margin - an integer representing the bottom margin of the page in twips. The default value is 1440.

Sect_SetMargins

Signature

function Sect_SetMargins($Left = '', $Right = '', $Top = '', $Bottom = '')

Description

Set the margins of the page in twips for the current section.

Parameters

$Left - an integer representing the left margin of the page in twips. The default value is 1800.

$Right - an integer representing the right margin of the page in twips. The default value is 1800.

$Top - an integer representing the top margin of the page in twips. The default value is 1440.

$Bottom - an integer representing the left margin of the page in twips. The default value is 1440.

Sect_SetMargl

Signature

function Sect_SetMargl($Margin = '')

Description

Set the left margin of the page in twips for the current section.

Parameters

$Margin - an integer representing the left margin of the page in twips. The default value is 1800.

Sect_SetMargmirror

Signature

function Sect_SetMargmirror()

Description

Switches margin definitions on left and right pages for the current section. Used in conjunction with SetFancingp**.**

Parameters

Sect_SetMargr

Signature

function Sect_SetMargr($Margin = '')

Description

Set the right margin of the page in twips for the current section.

Parameters

$Margin - an integer representing the right margin of the page in twips. The default value is 1800.

Sect_SetMargt

Signature

function Sect_SetMargt($Margin = '')

Description

Set the top margin of the page in twips for the current section.

Parameters

$Margin - an integer representing the top margin of the page in twips. The default value is 1440.

Sect_SetPageBorderFoot

Signature

function Sect_SetPageBorderFoot()

Description

Tells RTF reader that page borders surrounds footer in the current section.

Parameters

Sect_SetPageBorderHead

Signature

function Sect_SetPageBorderHead()

Description

Tells RTF reader that page borders surrounds header in the current section.

Parameters

Sect_SetPageBorderOption

Signature

function Sect_SetPageBorderOption($BorderOption = '')

Description

Set the reference for the page border measurement in the current section.

Parameters

$BorderOption - an integer specifying the option.

Allowed values:

- 8 -> Page border measure from text. Always display in front option is set to off.
- 32 -> Page border measure from edge of page. Always display in front option is set to on.
- 4032 -> Page border measure from edge of page. Always display in front option is set to off.

By default: '8'

Sect_SetPageBorders

Signature

function Sect_SetPageBorders($BrdPos = '', $BrdType = '', $BrdPen = '', $BrdColor = '', $BrdSpace = '', $BrdArt = '', $BrdOpt = '', $BrdAlign = '')

Description

Set the page borders for the current section.

Parameters

$BrdPos - a string representing the position or the positions the border/borders will be defined for.

Allowed values:

't'	Top.
'b'	Bottom
'l'	Left.
'r'	Right
'tblr' | 'tbl' | 'tb' | 'bl' | 'lr' | 'tl' etc - any combination	

By default: 'tblr'

$BrdType - a string specifying the border style.

Allowed values:

'Single-thickness | Single thickness | Single'	Single-thickness border.
'Double-thickness | Double thickness'	Double-thickness border.
'Shadowed'	Shadowed border.
'Double'	Double border.
'Dotted'	Dotted border.
'Dotted'	Dotted border.
'Dashed'	Dashed border.
'Hairline'	Hairline border.
'Resembles a frame | Frame'	Border resembles a “Frame.”
'No border specified | No'	No border specified.
'Table cell has no borders | table cell has no'	Table cell has no borders.
'Inset'	Inset border.
'Dashed small'	Dashed small.
'Dot-dashed | Dot dashed'	Dot-dashed border.
'Dot-dot-dashed | Dot-dot dashed | Dot dot-dashed | Dot dot dashed'	Dot-dot-dashed border.
'Outset'	Outset border.
'Triple'	Triple border.
'Thick-thin small | Thick thin small'	Thick-thin border (small).
'Thin-thick small | Thin thick small'	Thin-thick border (small).
'Thin-thick thin small | Thin thick thin small'	Thin-thick thin border (small).
'Thick-thin medium | Thick thin medium'	Thick-thin border (medium).
'Thin-thick medium | Thin thick medium'	Thin-thick border (medium).
'Thin-thick thin medium | Thin thick thin medium'	Thin-thick thin border

(medium).

- 'Thick-thin large | Thick thin large' Thick-thin border (large).
- 'Thin-thick large | Thin thick large' Thin-thick border (large).
- 'Thin-thick thin large | Thin-thick thin large | Thin thick-thin large | Thin thick thin large' Thin-thick-thin border (large).
- 'Wavy' Wavy border.
- 'Double wavy' Double wavy border.
- 'Striped' Striped border.
- 'Embossed' Embossed border.
- 'Engraved' Engraved border.

By default: 'No border specified'

$BrdPen - this argument is representing the border width in twips. It is the width in twips of the pen used to draw the paragraph border line and cannot be greater than 75. To obtain a larger border width, please prefix your width with the letter 'D' and you will obtain a width double that the one you had specified. Ex: 'D60' will draw a border with 120 twips in width. If null or missing by default the width will 1.

$BrdColor - a string representing the RGB color code of the border to be drawn. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a border with red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'.

$BrdArt - this argument is a number between 1 and 165 representing the number of the border art of the page. By default is null that means no border art will be used to draw the border of the page.

$BrdSpace - this argument specify the space in twips between borders and the paragraph. By default the space is 480 twips.

$BrdOpt - an integer specifying the reference for the page border measurement.

Allowed values:

- 8 -> Page border measure from text. Always display in front option is set to off.
- 32 -> Page border measure from edge of page. Always display in front option is set to on.
- 4032 -> Page border measure from edge of page. Always display in front option is set to off.

By default: '8'

Sect_SetPageNbRigh

Signature

function Sect_SetPageNbRigh($PageRight = '')

Description

Set the Page number right margin for the current section. This control word is understood but not used by current versions (6.0 or later) of Word

Parameters

$PageRight - a number representing the distance in twips between page number and the right margin (the default is 720).

Sect_SetPageNbTop

Signature

function Sect_SetPageNbTop($PageTop = '')

Description

Set the Page number top margin for the current section. This control word is understood but not used by current versions (6.0 or later) of Word

Parameters

$PageTop - a number representing the distance in twips between page number and the top margin (the default is 720).

Sect_SetPaperh

Signature

function Sect_SetPaperh($Paperh = '')

Description

Set the paper height in twips for the current section.

Parameters

$Paperh - a number representing the paper height in twips. By default the paper height is 15840 twips.

Sect_SetPaperSize

Signature

function Sect_SetPaperSize ($PaperFormat = '')

Description

Define the size of the paper for the current section. The size can be specified by passing as parameter the paper format. You can also define your custom paper size by using Sect_SetPaperh and Sect_SetPaperw.

Parameters

$PaperFormat - the type of paper used for this document

Allowed values:

Letter	(8.5 x 11 in.)
Letter Small	(8.5 x 11 in.)
Note	(8.5 x 11 in.)
Tabloid	(11 x 17 in.)
11 x 17 in	(11x17 in.)
Ledger	(17 x 11 in.)
Legal	(8.5 x 14 in.)
Statement	(5.5 x 8.5 in.)
Executive	(7.25 x 10.5 in.)
A3	(297 x 420 mm)
A4	(210 x 297 mm)
A4 Small	(210 x 297 mm)
A5	(148 x 210 mm)
B4 (jis)	(250 x 354)
B5 (jis)	(182 x 257 mm)
Folio	(8.5 x 13 in.)
Quarto	(215 x 275 mm)
10x14 in	(10x14 in.)
Envelope #9	(3.875 x 8.875 in.)
Envelope #10	(4.125 x 9.5 in.)
Envelope #11	(4.5 x 10.375 in.)
Envelope #12	(4.75 x 11 in.)
Envelope #14	(5 x 11.5 in.)
C size sheet	(17 x 22 in.)
D size sheet	(22 x 34 in.)
E size sheet	(34 x 44 in.)
Envelope DL	(110 x 220 mm)
Envelope C3	(324 x 458 mm)
Envelope C4	(229 x 324 mm)
Envelope C5	(162 x 229 mm)
Envelope C6	(114 x 162 mm)
Envelope C65	(114 x 229 mm)
Envelope B4	(250 x 353 mm)
Envelope B5	(176 x 250 mm
Envelope B6	(176 x 125 mm)

Envelope	(110 x 230 mm)
Envelope Monarch	(3.875 x 7.5 in.)
6-3/4 Envelope	(3.625 x 6.5 in.)
US Std Fanfold	(14.875 x 11 in.)
German Std Fanfold	(8.5 x 12 in.)
German Legal Fanfold	(8.5 x 13 in.)
A6	(105x142 mm)
index card 4x6 in	(4x6 in.)
index card 5x8 in	(5x8 in.)
hagaki card 100x148 mm	(100x 148 mm)

By default : 'Letter'

Sect_SetPaperw

Signature

function Sect_SetPaperw($Paperw = '')

Description

Set the paper width in twips for the current section.

Parameters

$Paperw - a number representing the paper width in twips. By default the paper width is 12240 twips.

Sect_SetParBorderAlign

Signature

function Sect_SetParBorderAlign()

Description

Align paragraph borders and table edges with page border in the current section.

Parameters

Sect_SetRender

Signature

function Sect_SetRender($Render = '')

Description

Set the rendering of the current section.

Parameters

$Render - a string specifying the rendering style of the document. Any string beginning with 'V' for vertical rendering. Anything else even null or missing argument for horizontal rendering.

Sect_SpaceBetCols

Signature

function Sect_SpaceBetCols ($SpaceBetCols = '')

Description

Set the space, in twips, between columns in the current section.

Parameters

$SpaceBetCols – a number representing the space wbetween columns. By default is 720.

Sect_SpaceToRightOfCol

Signature

function Sect_SpaceToRightOfCol ($SpaceCols = '')

Description

Space to right of column in twips in the current section.; used to specify formatting for variable-width columns

Parameters

$SpaceCols	–	a number representing the space to right of column. By default is 720.

Sect_StateOf

Signature

function Sect_StateOf ($StateOf = '')

Description

Set the default state of section. Or specify number of chars per line only or both number of char per line and number of line per page.

Parameters

$StateOf – a string containing the state of section.

Allowed values:

- 'Default | D' Indicates number of chars per line and number of line per page are not emitted.
- 'Characters per line | Char/line | c/l | cl' Specify number of chars per line only.
- 'Both | B | Char/line and line/page | c/l l/p' Specify both number of char per line and number of line per page.

By default : 'Default'

Sect_TextFlow

Signature

function Sect_TextFlow ($TextFlow = '')

Description

Set the section properties for specifying text flow.

Parameters

$TextFlow – a string containing text flow properties of the current section

Allowed values:

- 'Left to right and top to bottom | lrtb'
- 'Top to bottom and right to left, vertical | tbrlv'
- 'Left to right and bottom to top | lrbt'
- 'Right to left and top to bottom | rltb'
- 'Left to right and top to bottom, vertical | lrtbv'
- 'Vertically, non-vertical font | vert | v'

By default : 'Left to right and top to bottom'

Sect_TextShouldSnap

Signature

function Sect_TextShouldSnap ()

Description

Indicates that text should snap to the character grid in the current section.

Parameters

Sect_UnlockedForForms

Signature

function Sect_UnlockedForForms ()

Description

Specify this section is unlocked for forms..

Parameters

Sect_VertAlign

Signature

function Sect_VertAlign ($VertAlign ='')

Description

Set the vertical alignment of the text in the current section..

Parameters

$VertAlign – a string specifying the vertical alignment of the text.

Allowed values:

- 'Top-aligned | top | t'
- 'Bottom-aligned | bottom | b'
- 'Centered vertically | centered | center | c'
- 'Justified vertically | justified | justify | justif | j'

By default : 'Top'

Functional methods

1. AddColor
2. AddFont
3. AddStyle
4. AddUserProperties
5. Bullet
6. Close
7. CloseBookmark
8. Date
9. DefineFooters
10. DefineHeaders
11. EmbedField
12. EmbedPicture
13. EmbedShape
14. Error
15. FRTF
16. InsertAnnotation
17. InsertBookmark
18. InsertDocumentVariable
19. InsertEndNote
20. InsertField
21. InsertFooter
22. InsertFootNote
23. InsertFormField_CheckBox
24. InsertFormField_DropDown
25. InsertFormField_Text
26. InsertHeader
27. InsertPicture
28. InsertShape
29. InsertTable
30. InsertUnicodeChar
31. InsertUnicodeCorresp
32. LastWordSupported
33. LineBreak
34. Open
35. Output
36. PageBreak
37. Tab
38. Write

AddColor

Signature

function AddColor ($Red, $Green, $Blue)

Description

Add a new color in the color table of the document.

This method return the index of the color in the color table or -1 if the color hasn't been added to the color table.

Parameters

$Red – a number between 000-255 representing the code for the red color.

$Green – **a number between 000-255 representing the code for the green color.**

$Blue – a number between 000-255 representing the code for the blue color.

AddFont

Signature

function AddFont ($Name, $Family, $CharSet, $Pitch = 0, $Panose = '', $Tagg = '', $AltName = '', $EmbType = '', $EmbFile = '', $EmbCPage = '')

Description

Add a new font in the font table of the document.

This method return the index of the font in the font table or -1 if the font hasn't been added to the font table.

Parameters

$Name – **a** string specifying the name of the font to be added

$Family – **a string specifying the family of the font**

Allowed values:

- Unknown   Unknown or default font (the default)
- Roman   Roman, proportionally spaced serifs font   -> Times New Roman, Palatino
- Swiss   Swiss proportionally spaced sans serifs font -> Arial
- Modern | **Fixed-pitch** | **Fixed**   **Fixed-pitch serif and sans serif fonts**   **-> Courier New, Pica**
- **Script**   **Script fonts**   **-> Cursive**
- **Decorative**   **Decorative fonts**   **-> Old English, ITC Zapf Chancery**
- **Technical** | **Symbol** | **Mathematical**   **Technical, symbol and mathematical fonts**   **-> Symbol**
- **Bidirectional** | **arabic** | **hebrew**   **Arabic, Hebrew or other bidirectional font**   **-> Miriam**

By default : 'Unknown'

$Charset – **a string specifying the character set of the font**

Allowed values:

- **Ansi**
- **Default**
- **Symbol**
- **Invalid**
- **Mac**
- **Shift jis**
- **Hangul**
- **Johab**
- **Gb2312**
- **Big5**
- **Greek**
- **Turkish**
- **Vietnamese**
- **Hebrew**

- Arabic
- Arabic traditional
- Arabic user
- Hebrew user
- Baltic
- Russian
- Thai
- Eastern european
- Pc 437
- Oem

By default : 'Ansi'

$Pitch – a string specifying the font pitch

Allowed values:

- Default
- Fixed | Fix | F
- Variable | Var | V

By default : 'Default'

$Panose – this string contains a 10-byte Panose 1 number. Each byte represents a single font property as described by the Panose 1 standard specification.

$Tagg – This is an optional control word in the font table to define the nontagged font name. This is the actual name of the font without the tag, used to show which character set is being used. For example, Arial is a nontagged font name, and Arial (Cyrillic) is a tagged font name. This control word is used by WordPad. Word ignores this control word (and never creates it).

$AltName – Indicates alternate font name to use if the specified font in the font table is not available.

$EmbType – a string containing the type of the embedded font

Allowed values:

- Unknown
- True type | True | T

By default : 'Unknown'

$EmbFile – the file name containing the embedded font

$EmbCPage – specify the character set for the file name

Allowed values:

- United states ibm
- Arabic (asmo 708)
- Arabic (asmo 449+, bcon v4)
- Arabic (transparent arabic)
- Arabic (nafitha enhanced)
- Arabic (transparent asmo)
- Windows 3.1 (united states and western europe)
- Ibm multilingual
- Eastern european
- Portuguese

- Hebrew_1
- French canadian
- Arabic_1
- Norwegian
- Soviet union
- Thai
- Japanese
- Simplified chinese
- Korean
- Traditional chinese
- Windows 3.1 (eastern european)
- Windows 3.1 (cyrillic)
- Western european
- Greek
- Turkish
- Hebrew
- Arabic
- Baltic
- Vietnamese
- Johab

By default : 'Western european'

AddStyle

Signature

function AddStyle ($StyleFormatting, $StyleDef, $StyleName **= '',** $StyleKeyCode **= '',** $StyleAdditive **= '',** $StyleBased **= '',** $StyleNext **= '',** $StyleAutoUpd **= '',** $StyleHidden **= '',** $StylePers **= '',** $StyleCompose **= '',** $StyleReply **= '',** $StyleSemiHidden **= '',** $StyleID **= '')**

Description

Add a style in the style sheet table of the document.

This method return the index of the style sheet in the style sheet table or -1 if the style sheet hasn't been added to the style sheet table.

Parameters

$StyleFormatting – a string specifying the format to be applied to the style

All style formatting elements are configured in the *conf.inc.php* file.
So it is easy to change labels for objects or separators in the style formatting string.

By default these labels are:

Group separator begin	='['
Group separator end	= ']'
Multi definitions begin	= '{'
Multi definitions end	= '}'
Value separator	='_'
Missing value	='^'
Label for BORDER properties	='BRDRDEF'
Label for PARAGRAPH properties	='PARFMT'
Label for FRAME properties	='APOCTL'
Label for TAB properties	='TABDEF'
Label for SHADING properties	='SHADING'
Label for CHARACTER properties	='CHRFMT'
Label for FONT properties	='FONTDEF'
Label for LANGUAGE properties	='LANGDEF'

The structure of the style formatting string is:
$StyleFormatting=(Object[{Definition}+])* ; Definition=((value | ^) _)+ where +, |, * have PHP Eregi significations

Objects to which we can change properties in the style sheet are:

- FONT
- PARAGRAPH
- TABS
- BORDERS
- SHADING
- LANGUAGE
- FRAME

FONT

FONTDEF:[{FontFamily_FontStyle_FontSize_FontColor_FontUnderline_FontEffects_CharScale_CharSpacing_CharPosition_CharKerning_TextAnimation}]

FontFamily - the name of the font family (it has to be added in the font table !)
FontStyle - the font style

Allowed values:

- Regular
- Italic
- Bold
- Bold Italic

By default : 'Regular'

FontSize - the size of the font between 1 and 1638. By default 12.
FontColor - a string representing the color code of the font. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a border with red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'.
FontUnderline - specify the underline style and color for the font. It can be specified like 'style255000128' or '255000128style' or any mixture '2s55t0y00l1e28'. For the underline color see FontColor.

Allowed values for style:

- Continuous
- Dotted
- Dashed
- Dash-dotte
- Dash-dot-dotted
- Double
- Heavy wave
- Long dashed
- None
- Thick
- Thick dotted
- Thick dashed
- Thick dash-dotted
- Thick dash-dot-dotted
- Thick long dashed
- Double wave
- Word
- Wave

By default : 'None'

FontEffects - specify the effects to be applied to the font. It can contains one or more (even all) of the followings values:

Allowed values for effects: "strikethrough | double strikethrough | superscript | subscript | shadow | outline | emboss | engrave | small capitals | all capitals | hidden".

CharScale - specify the character scale of the font. Allowed values between 1 and 600%. By default 100.
CharSpacing - specify the character spacing for the font. Values between -1584 and 1584 in points. If CharSpacing < 0 that means Condensed else means Expanded. By default 0.
CharPosition - specify the character position for the font. Values between -1584 and 1584 in points. If CharPosition < 0 that means Lowered else means Raised. By default 0.
CharKerning - specify the character kerning for the font. Values between 1 and 1638 in points. By default 0.
TextAnimation - specify the animation for the font

Allowed values:

- None
- Las Vegas Lights
- Blinking Background
- Sparkle Text
- Marching Black Ants
- Marching Red Ants
- Shimmer

By default : 'None'

EX: FONTDEF:[{Times New Roman CE_Bold italic_12_128000255_word128000255_strikethrough double strikethrough superscript subscript shadow outline emboss engrave small capitals all capitals hidden_^_^_^_^_Las Vegas Lights}] , the char '^' is used to specify a missing value; in that case by default values are used: CharScale=100, CharSpacing=0, CharPosition=0, CharKerning=0.

PARAGRAPH

PARFMT:[{Align_Outline_Indent_Special_Spacing_LineSpace_PageBreak_FontAlign}]

Align - define the paragraph alignement

Allowed values:

- Centered | center | c
- Justified | just | j
- Left-aligned | left | l
- Right-aligned | right | r
- Distributed | distri | d
- Kashida-low | kashida-l | kashidal | k-l | kl
- Kashida-medium | kashida-m | kashidam | k-m | km
- Kashida-high | kashida-h | kashidah | k-h | kh
- Thai | t

By default : 'Left-aligned'

Outline - define the outline level of paragraph.

Allowed values:

- Body text | bodytext | body
- Level1 | L1 | 1
- Level2 | L2 | 2
- Level3 | L3 | 3
- Level4 | L4 | 4
- Level5 | L5 | 5
- Level6 | L6 | 6
- Level7 | L7 | 7

- Level8 | L8 | 8
- Level9 | L9 | 9
-

By default : 'Body Text'

Indent - specify the indentation for the paragraph. Possible values: Left9999Right9999 | L9999Right9999 |Left9999R9999 | L9999R99999 where 9 means a digit. The number after Left and Right is the indentation in twips and must be between -31680 and 31680 that correspond to -22 to 22 inches.
Special - define special indentation for the paragraph. In the list below 9 mean a digit and the range of indentation is 0 - 28800 twips that correspond to 0-20 inches.

Allowed values:

- None | n
- First line | firstline | fline | fl | first | line | f | 1+ 9999
- Hanging | hang | h + 9999

By default : 'None'

Spacing - specify the spacing for the paragraph. Possible values: Before9999After9999 | B9999After9999 | B9999A9999 | Before9999A9999 BAutoA9999 | BautoAauto where 9 means a digit. The values for before and after must be between -1 and 1584 pts
LineSpace - specify the line spacing. For At least and Exactly values you an add or mix (E1x3a4c6tly<=>Exactly1346) a space in twips. For Multiply value you can add or mix (Mu3ltiply<=>Multliply3) a number of lines.

Allowed values:

- Single | sing | s
- .lines | lines | line | .line | line | lin | l
- double | doub | do | db | d
- at least | at | least | le | a
- exactly | exact | ex | e | x
- multiple | multi | mul | m

By default : 'Single'

PageBreak - define the page breaks for the paragraph.
Possible values: W-L-X-P-S-H. Presence of letter means checkbox =true and the absence means checkbox=false.

W=>Widow/Orphan control
L=> Keep lines together
X=> Keep with next
P=> Page break before
S=> Suppress line numbers
H=> Don't hyphenate

FontAlign - define the font alignment in the paragraph

Allowed values:

- Auto | a
- Hanging | hang | h
- Center | c
- Roman | r
- Upholding variable | u var | variable | var | v
- Upholding fixed | u f | fixed | fix | f

By default : 'Auto'

EX: PARFMT:[{Left_Body text_L0R0_^_Before3A3_single_W_^}], the char '^' is used to specify a missing value; in that case by default values are used: Special=None, FontAlign=Auto.

TABS

TABDEF:[{TabStop_Align_Lead}]

TabStop - define tab stop position. A number specifying the tab stop position in inches.
Align - define the alignment of the tab

Allowed values:

- Left | L
- Center | C
- Right | R
- Decimal | Deci | D
- Bar | B

By default : 'Left'

Lead - define tab leader

Allowed values:

- None
- Dots | D
- Middle dots | middle | m
- Hyphens | H
- Underline | U
- Thick line | Thick | line | t
- Equal sign | equal | sign | e

By default : 'None'

EX: TABDEF:[{2_center_u} {1.5_right_equal sign}].

BORDERS

BRDRDEF:[{BorderStyle_BorderColor_BorderWidth_BorderSpace_BorderSettings}]

BorderStyle - specify the border style.

Allowed values:

• 'Single-thickness | Single thickness | Single'	Single-thickness border.
• 'Double-thickness | Double thickness'	Double-thickness border.
• 'Shadowed'	Shadowed border.
• 'Double'	Double border.
• 'Dotted'	Dotted border.
• 'Dotted'	Dotted border.
• 'Dashed'	Dashed border.
• 'Hairline'	Hairline border.
• 'Resembles a frame | Frame'	Border resembles a “Frame.”
• 'No border specified | No'	No border specified.
• 'Table cell has no borders | table cell has no'	Table cell has no borders.

'Inset'	Inset border.
'Dashed small'	Dashed small.
'Dot-dashed | Dot dashed'	Dot-dashed border.
'Dot-dot-dashed | Dot-dot dashed | Dot dot-dashed | Dot dot dashed'	Dot-dot-dashed border.
'Outset'	Outset border.
'Triple'	Triple border.
'Thick-thin small | Thick thin small'	Thick-thin border (small).
'Thin-thick small | Thin thick small'	Thin-thick border (small).
'Thin-thick thin small | Thin thick thin small'	Thin-thick thin border (small).
'Thick-thin medium | Thick thin medium'	Thick-thin border (medium).
'Thin-thick medium | Thin thick medium'	Thin-thick border (medium).
'Thin-thick thin medium | Thin thick thin medium'	Thin-thick thin border (medium).
'Thick-thin large | Thick thin large'	Thick-thin border (large).
'Thin-thick large | Thin thick large'	Thin-thick border (large).
'Thin-thick thin large | Thin-thick thin large | Thin thick-thin large | Thin thick thin large'	Thin-thick-thin border (large).
'Wavy'	Wavy border.
'Double wavy'	Double wavy border.
'Striped'	Striped border.
'Embossed'	Embossed border.
'Engraved'	Engraved border.

By default: 'No border specified'

BorderColor - a string representing the color code of the border to be drawn. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a border with red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'.

BorderWidth -this argument is representing the border width in twips. It is the width in twips of the pen used to draw the paragraph border line and cannot be greater than 75. To obtain a larger border width, please prefix your width with the letter 'D' and you will obtain a width double that the one you had specified. Ex: 'D60' will draw a border with 120 twips in width. If null or missing by default the width will 1.

BorderSpace - this argument specify the space in twips between borders and the paragraph. By default the space is 480 twips.

BorderSettings - specify the margins to which properties apply. Possible values: T -> Top, B->Bottom, L->Left, R->Right, I->Individual, X->boX

EX: BRDRDEF:[{shadowed_000000000_32_3_tlr}{shadowed_255000000_32_3_b}].

SHADING

SHADING:[{ShadingStyle_FillColor_BackColor}]

ShadingStyle - define the shading style.

Allowed values:

- 0 - 100, specifying a percentage
- Dk Horizontal
- Dk Vertical
- Dk Dwn Diagonal
- Dk Up Diagonal
- Dk Grid
- Dk Trellis

- Lt Horizontal
- Lt Vertical
- Lt Dwn Diagonal
- Lt Up Diagonal
- Lt Grid
- Lt Trellis

By default : '0'

FillColor - a color code
BackColor - a color code

EX: SHADING[{20_255255_255255255}].

LANGUAGE

LANGDEF:[{Lang_Langfe}]

Lang - primary language
Langfe - secondary language

Allowed values for both above items:

- Afrikaans
- Albanian
- Arabic
- Arabic Algeria
- Arabic Bahrain
- Arabic Egypt
- Arabic General
- Arabic Iraq
- Arabic Jordan
- Arabic Kuwait
- Arabic Lebanon
- Arabic Libya
- Arabic Morocco
- Arabic Oman
- Arabic Qatar
- Arabic Syria
- Arabic Tunisia
- Arabic U.A.E.
- Arabic Yemen
- Armenian
- Assamese
- Azeri Cyrillic
- Azeri Latin
- Basque
- Bengali
- Bosnia Herzegovina
- Bulgarian
- Burmese
- Byelorussian
- Catalan
- Chinese China
- Chinese General
- Chinese Hong Kong

- Chinese Macao
- Chinese Singapore
- Chinese Taiwan
- Croatian
- Czech
- Danish
- Dutch Belgium
- Dutch Standard
- English Australia
- English Belize
- English British
- English Canada
- English Caribbean
- English General
- English Ireland
- English Jamaica
- English New Zealand
- English Philippines
- English South Africa
- English Trinidad
- English United States
- English Zimbabwe
- Estonian
- Faeroese
- Farsi
- Finnish
- French
- French Belgium
- French Cameroon
- French Canada
- French Cote d'Ivoire
- French Luxemburg
- French Mali
- French Monaco
- French Reunion
- French Senegal
- French Swiss
- French West Indies
- French Zaire
- Frisian
- Gaelic
- Gaelic Ireland
- Galician
- Georgian
- German
- German Austrian
- German Liechtenstein
- German Luxemburg
- German Switzerland
- Greek
- Gujarati
- Hebrew
- Hindi
- Hungarian
- Icelandic

- Indonesian
- Italian
- Italian Switzerland
- Japanese
- Kannada
- Kashmiri
- Kashmiri India
- Kazakh
- Khmer
- Kirghiz
- Konkani
- Korean
- Korean Johab
- Lao
- Latvian
- Lithuanian
- Lithuanian Classic
- Macedonian
- Malay
- Malay Brunei Darussalam
- Malayalam
- Maltese
- Manipuri
- Marathi
- Mongolian
- Nepali
- Nepali India
- Norwegian Bokmal
- Norwegian Nynorsk
- Oriya
- Polish
- Portuguese Brazil
- Portuguese Iberian
- Punjabi
- Rhaeto-Romanic
- Romanian
- Romanian Moldova
- Russian
- Sami Lappish
- Sanskrit
- Serbian Cyrillic
- Serbian Latin
- Sindhi
- Slovak
- Slovenian
- Sorbian
- Spanish Argentina
- Spanish Bolivia
- Spanish Chile
- Spanish Colombia
- Spanish Costa Rica
- Spanish Dominican Republic
- Spanish Ecuador
- Spanish El Salvador
- Spanish Guatemala

- Spanish Honduras
- Spanish Mexico
- Spanish Modern
- Spanish Nicaragua
- Spanish Panama
- Spanish Paraguay
- Spanish Peru
- Spanish Puerto Rico
- Spanish Traditional
- Spanish Uruguay
- Spanish Venezuela
- Sutu
- Swahili
- Swedish
- Swedish Finland
- Tajik
- Tamil
- Tatar
- Telugu
- Thai
- Tibetan
- Tsonga
- Tswana
- Turkish
- Turkmen
- Ukrainian
- Urdu
- Urdu India
- Uzbek Cyrillic
- Uzbek Latin
- Venda
- Vietnamese
- Welsh
- Xhosa
- Yiddish
- Zulu

By default: 'Romanian'

EX: LANGDEF:[{^_^}] equivalent to LANGDEF:[{Romanian_Romanian}].

FRAME

APOCTL:[{FrameSize_HorizPos_VertPos_TextWrap_DropCap_TextFlow_OverLap}]

FrameSize - specify the size of the frame. Possible values: W9999H9999 where 9 means a digit. The numbers means the width and the height of the frame. A positive value for height indicates the minimum height of the frame, and a negative value indicates the exact height of the frame. A value of zero indicates that the height of the frame adjusts to the contents of the frame.
HorizPos - specify horizontal position. 99AlignReference.

99->distance from text in twips,

Allowed values for Align:

- Left | L

- Center | C
- Right | R
- Inside | I
- Outside | O

By default : 'Left'

Allowed values for Reference:

- Margin | m
- Page | p
- Column | c

By default : 'Column'

VertPos - **specify vertical position. 99AlignReferenceLock.**

99->distance from text in twips,

Lock -> Lock anchor (values 'lock | k')

Allowed values for Align:

- **Top | T**
- **Bottom | B**
- **Center | C**
- **Inside | I**
- **Inline | IL**
- **Outside | O**

By default : 'Top'

Allowed values for Reference:

- **Margin**
- **Page**
- **Paragraph | par**

By default : 'Margin'

TextWrap - **define the text wrap mode like: WrapMode?+OverlayMode? +B9999|W9999H9999**

Possible values :
WrapMode -> NoWrap | Now
OverlayMode -> Overlay | O
B9999 or W9999H9999 specifying the distance in twips from text in all directions respectively horizontally and vertically.

DropCap - **define number of lines drop cap is occupy to and the drop cap type : 9DropType**

Allowed values for DropType:

- **In-text | in | i**
- **Margin | mar | m**

By default : 'In-text'

TextFlow - **specify how the frame flows.**

Allowed values:

- lrtb   Frame box flows from left to right and top to bottom (default).
- rltb   Frame box flows right to left and top to bottom.
- lrbt   Frame box flows left to right and bottom to top
- lrtbv | vlrtb   Frame box flows left to right and top to bottom, vertical.
- rlbtv | vrlbt   Frame box flows top to bottom and right to left, vertical.

By default : 'lrtb'

OverLap - define overlap mode. 0 -> Allow overlap (default) 1 -> Do not allow overlap

EX: APOCTL:[{W0H0_leftcol_toppar_noWrapB1440W187H187_^_^_0}].

EXAMPLE of a style formatting definition string:

FONTDEF:[{Times New Roman_Bold_12_255_word000000255_^_^_^_^_^_^}]APOCTL:[{W0H0_leftcol_toppar_noWrapB1440W187H187_^_^_0}]TABDEF:[{2_center_u}{1.5_right_equal sign}]SHADING[{20_255255_255255255}]BRDRDEF:[{shadowed_000000000_32_3_tblr}]LANGDEF[]PARFMT:[{Left_Body text_L0R0_^_Before3A3_single_W_^}]

$StyleDef –   a string specifying the the object the style will be applied to

Allowed values:

- Character | char | c
- Paragraph | par | p
- Section | sect | s
- Table | tab | t

$StyleName –   a string specifying the name of the style sheet

$StyleKeyCode –   a string specifying the key code ALT SHIFT CTRL N 99, where 9 is a digit and the number 99 will be treated as F99

$StyleAdditive –   specify if the style sheet is additive or not

Allowed values:

- Yes | Y | Oui | o | Da | D
- No | N   - default

$StyleBased –   specify the style on which the current style sheet is based on.

$StyleNext –   Defines the next style associated with the current style; if omitted (or '__'), the next style is the current style.

$StyeleAutoUpd – specify if automatically update style

Allowed values:

- Yes | Y | Oui | o | Da | D
- No | N   - default

$StyleHidden –   specify if the style sheet is hidden or not. Style does not appear in the Styles drop-down list in the Style dialog box (on the Format menu, click Styles). The hidden style property can only be accessed using Microsoft Visual Basic® for Applications.

Allowed values:

- Yes | Y | Oui | o | Da | D
- No | N - default

$StylePers – specify if the style sheet is personal e-mail style or not

Allowed values:

- Yes | Y | Oui | o | Da | D
- No | N - default

$StyleCompose – specify if is the e-mail compose style

Allowed values:

- Yes | Y | Oui | o | Da | D
- No | N - default

$StyleReply – specify if is the e-mail reply style

Allowed values:

- Yes | Y | Oui | o | Da | D
- No | N - default

$StyleSemiHidden – specify if style does not appear in drop-downs menus.

Allowed values:

- Yes | Y | Oui | o | Da | D
- No | N - default

$StyleID – not used.

AddUserProperties

Signature

function AddUserProperties ($NewUsrPropname, $NewUsrProptype, $NewUsrStaticval, $NewUsrLinkval = '')

Description

This method add a custom user property to the document.

Parameters

$NewUsrPropname – the name of the property to be added.

$NewUsrProptype – the type of the property.

Allowed values:

- 'integer'
- 'real number'
- 'date'
- 'boolean'
- 'text'

By default : 'Section'

$NewUsrStaticval – the value of the property.

$NewUsrLinkval – the source of the link if this property is linked to content.

Bullet

Signature

function Bullet ()

Description

This method will insert a bullet character at the current position in the RTF document.

Parameters

Close

Signature

function Close ()

Description

This method will terminate the RTF document. It is not necessary to call it explicitly because the Output() method will do that for you.

Parameters

CloseBookmark

Signature

function CloseBookmark ($BookmarkTag = '')

Description

This method indicates the end of a bookmark for the document.

Note: this method is used in conjunction with InsertBookmark.

Parameters

$BookmarkTag – the tag that specify the bookmark. If it is not specified then the last bookmark tag will be used.

Date

Signature

function Date($DateFormat = '')

Description

This method will insert the current date, as specified in the argument, at the current position in the RTF document.

Parameters

$DateFormat – a string specifying the format of the date to be inserted in the document.

Allowed values:

- Long format | long | l -> Current date in long format
- Abbreviated | abb | a -> Current date in abbreviated format
- Time | t -> Current time as in headers
- Header | anything else -> Current date as in headers

By default : 'Header'

DefineFooters

Signature

function DefineFooters($FooterType = '')

Description

This method define the footer type for the current section.

Parameters

$FooterType – a string specifying the type of footer.

Allowed values:

- **All | a -> Footers will appear on all pages**
- **Left | l -> Footers will appear only on left pages, only if** SetFacingp **has been called for the document.**
- **Right | r -> Footers will appear only on right pages, only if** SetFacingp **has been called for the document.**
- **First | f -> Footers will appear only on first page if the first page has a special format (** Sect_FirstPage **has been called)**

By default : 'All'

DefineHeaders

Signature

function DefineHeaders($HeaderType = '')

Description

This method define the header type for the current section.

Parameters

$HeaderType – a string specifying the type of header.

Allowed values:

- **All | a -> Headers will appear on all pages**
- **Left | l -> Headers will appear only on left pages, only if SetFacingp has been called for the document.**
- **Right | r -> Headers will appear only on right pages, only if SetFacingp has been called for the document.**
- **First | f -> Headers will appear only on first page if the first page has a special format (Sect_FirstPage has been called)**

By default : 'All'

EmbedField

Signature

function EmbedField ($FieldToInsert, $FieldResult = '', $FieldMod = '', $FieldAlt = '', $FmtTxt = '')

Description

This method return the code of the field defined by parameters. With this method you can embed a field into the cells of a table.

Parameters

$FieldToInsert – a string specifying the field instruction.

$FieldResult – the most recent calculated result of the field.

$FieldMod – control words that alter the interpretation of the field.

Allowed values:

- 'dirty | d'   A formatting change has been made to the field result since the field was last updated.
- 'edit | e'   Text has been added to, or removed from, the field result since the field was last updated.
- 'locked | lock | l'   Field is locked and cannot be updated.
- 'private | priv | p'   Result is not in a form suitable for display (for example, binary data used by fields whose result is a picture).

By default : 'Section'

$FieldAlt – specify that the given field reference is to an endnote.

$FmtTxt – describe the formatting specifications for the text to be write - the result of the field. It can be the name of a style in the style sheet of the document previously added with AddStyle or a string with formatting specifications as it is defined by StyleFormatting in AddStyle method

EmbedPicture

Signature

function EmbedPicture($PictPath, $PictFormat = '', $ShapeFormat = '', $OtherProp = '')

Description

This method return the code of a picture from the file specified by $PictPath according to the format detailed by $PictFormat, $ShapeFormat and $OtherProp. With this method you can embed a picture into the cells of a table.

Parameters

$PictPath – path to the file the picture will be inserted from. Only .JPG and .PNG accepted. The images can be inserted either from a local disk or from World Wide Web.

$PictFormat – a string defining the format of the picture. It has the following structure: {PictFillColor_PictLineDef_PictSize_PictScale_PictCrop_PictImageControl_PictWeb}

PictFillColor - a string representing the fill color code of the picture. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a border with red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'.
PictLineDef - define the borders properties for the picture. It has the following structure: 'Color<123123123>Dashed<>Style<>Weight<>' or 'C<123123123>D<>S<>W<>' where

Color: is a RGB color code see PictFillColor for details about RGB color code.
Dash style Dashing: this is the dashed style of the line.

Allowed values (the values are self-explanatory):

- 'Solid line | solid | s'
- 'Dashed line windows | dashed windows | dash w'
- 'Dotted line windows | dotted windows | dot w'
- 'Dash-dotted line windows | dash dotted line windows | dash dotted windows | dash-dotted windows | d-d w | d d w | dd w | ddw'
- 'Dash-dot-dotted line windows | dash-dot-dotted windows | d-d-d w | ddd w | dddw'
- 'Dotted line | dotted | dot'
- 'Dashed line | dashed | dash'
- 'Long dashed line | long dashed | l d | ld'
- 'Dash-dotted line | dash dotted line | dash dotted | dash-dotted | d d |dd'
- 'Long dash-dotted line | long dash dotted line | long dash dotted | long dash-dotted | l d-d | l dd | ldd'
- 'Long dash-dot-dotted line | long dash-dot-dotted | l d-d-d | l ddd | lddd'

By default : 'Solid line'

Line style: this is the style of the line.

Allowed values (the values are self-explanatory):

- 'Single | s'
- 'Double of equal width | double equal | d e | de'
- 'Dotted line windows | dotted windows | dot w'
- 'Double one thick one thin | double one one | d o o | doo'
- 'Double reverse order | double reverse | d r | dr'
- 'Three'

By default : 'Single'

Weigth: the width of the line (border) [0 - 1584 pt]
Example:"Color<128128128>Dashed<Solid>Style<Single>Weight<2>"or "C<128128128>D<Solid>S<Single>W<2>", in this case for Dashed and Style the values by default will be used.

PictSize - define the size of the picture. It has this structure: Height99Width99 or H99W99, where 99 is a number between 0 - 22".
Example:"Height3.12Width6.76"
PictScale - define the scale of the picture. It has this structure: Height99Width99LR or H99W99LR, where 99 is a percent between 0-10675.
L -> if present then *Lock aspect ration* checkbox will be checked.
R -> if present then *Relative to original size* checkbox will be checked.
Example:"Height100Width1000LR"
PictCrop - define the crop parameters for the picture. It has this structure: Left99Top99Right99Bottom99 or L99T99R99B99, where 99 is a number between -22"-22".
Example:"Left12Top13Right4Bottom5"
PictImageControl - define the brightness of the picture. It has this structure: <TYPE>Brightness99Contrast99. where 99 is a number between 0-100.

Allowed values for TYPE (the values are self-explanatory):

- 'Automatic | auto | a'
- 'Grayscale | gray | g'
- 'Black ?
- 'Watermark | water | w'

By default : 'Automatic'

Example:"<Automatic>Brightness100Contrast100"

PictWeb - the text to be displayed in internet environment. Web browsers display alternative text while pictures are loading or if they are missing. Web search engines use alternative text to help find Web pages.

$ShapeFormat – a string defining the format of the shape which will contains the picture. It has the following structure:
{ShapeWrapStyle_ShapeWrapText_ShapeDistance_ShapeHoriz_ShapeVert_ShapeOptions}

ShapeWrapStyle - define the wrap style.

Allowed values (the values are self-explanatory):

- 'Wrap around shape | square | squ | q'
- 'Wrap tightly around shape | tight | tightly | ti'
- 'Wrap text through shape | through | th'
- 'Wrap around top and bottom of shape | top and bottom | tb'
- 'Wrap behind text | behind | b'
- 'In front of text | front | f'

- 'In line with text | line | l'

By default : 'Wrap around shape'

ShapeWrapText - define the wrap text mode on sides of shape.

Allowed values (the values are self-explanatory):

- 'Wrap both sides of shape | both sides | both | b'
- 'Wrap left side only | left side | left | l'
- 'Wrap right side only | right side | right | r'
- 'Wrap only on largest side | largest side | largest | la'

By default : 'Wrap both sides of shape'

ShapeDistance - define the distances from text. It has this structure: Left99Top99Right99Bottom99 or L99T99R99B99, where 99 is a number between 0"-22".

Example:"Left12Top13Right4Bottom5"

ShapeHoriz - define the horizontal alignment of the shape. It has the following structure: ALIGN:RELATIVETO.

Allowed values for ALIGN (the values are self-explanatory):

- Number between -22"-22" - that correspond to absolute position
- 'Left | l'
- 'Centered | center | c'
- 'Right | r'
- 'Inside | i'
- 'Outside | o'

By default : 'Left'

Allowed values for RELATIVETO (the values are self-explanatory):

- 'Margin | mar | m'
- 'Page | pg | p'
- 'Column | col | c'
- 'Character | char | ch'

By default : 'Column'

Example:"'Left:Column' or 'Centered:Page' or 'Inside:Margin' or '-12:Character'"

ShapeVert - define the vertical alignment of the shape. It has the following structure: ALIGN:BELLOWTO.

Allowed values for ALIGN (the values are self-explanatory):

- Number between -22"-22" - that correspond to absolute position
- 'Top | t'
- 'Centered | center | c'
- 'Bottom | b'
- 'Inside | i'
- 'Outside | o'

By default : 'Left'

Allowed values for BELLOWTO (the values are self-explanatory):

- 'Margin | mar | m'

- **'Page | pg | p'**
- **'Paragraph | para | par'**
- **'Line | lin | l'**

By default : 'Paragraph'

Example:"'Top:Page' or 'Centered:Line' or 'Inside:Margin' or '-12:Paragraph'"

ShapeOptions - define the supplementary options for shape positioning. These options are *'Move object with text'* | *'Lock anchor'* | *'Allow overlap'*.The structure of this parameter is : (M?)(L?)(A?). If the letter is present the the corresponding option is set.
Example:"'MLA' or 'LA' or 'MA'"

$OtherProp – an array with properties to be applied to the shape. It allows to customize in detail the shape. The format of the array should be: array key = name of the property, array value = value of the property.
Example:array('fLockAspectRatio'=>'1','fLockRotation'=>'0','Rotation'=>65536*10)
The set of properties/values which can be used:

Property	**Meaning**	**Type of value**	**Default**
Lock			
fLockRotation	Boolean	Lock rotation.	FALSE
fLockAspectRatio	Boolean	Lock aspect ratio.	FALSE
fLockAgainstSelect	Boolean	Lock against selection.	FALSE
fLockCropping	Boolean	Lock against cropping.	FALSE
fLockVerticies	Boolean	Lock against edit mode.	FALSE
fLockText	Boolean	Lock text against editing.	FALSE
fLockAdjustHandles	Boolean	Lock adjust handles.	FALSE
fLockAgainstGrouping	Boolean	Lock against grouping.	FALSE
fLockShapeType	Boolean	Lock the shape type (don’t allow Change Shape).	FALSE
Text Box			
dxTextLeft	EMU	Left internal margin of the text box.	91,440
dyTextTop	EMU	Top internal margin of the text box.	45,720
dxTextRight	EMU	Right internal margin of the text box.	91,440
dyTextBottom	EMU	Bottom internal margin of the text box.	45,720

WrapText	Not applicable	Wrap text at shape margins: 0 Square 1 Tight 2 None 3 Top bottom 4 Through	0
anchorText	Not applicable	Text anchor point: 0 Top 1 Middle 2 Bottom 3 Top centered 4 Middle centered 5 Bottom centered 6 Bottom centered baseline	0
txflTextFlow	Not applicable	Text flow: 0 Horizontal non-ASCII font 1 Top to bottom ASCII font 2 Bottom to top non-ASCII font 3 Top to bottom non-ASCII font 4 Horizontal ASCII font	0
cdirFont	Direction	Font rotation: 0 Right 1 Down 2 Left 3 Up	0
fAutoTextMargin	Boolean	Use host's margin calculations.	FALSE
scaleText	Long integer	Text zoom and scale.	0

lTxid	Long integer	ID for the text. The value is determined by the host.	0
fRotateText	Boolean	Rotate text with shape.	FALSE
fSelectText	Boolean	TRUE if single click selects text, FALSE if two clicks select text.	TRUE
fFitShapeToText	Boolean	Adjust shape to fit text size.	FALSE
fFitTextToShape	Boolean	Adjust text to fit shape size.	FALSE
WordArt Effect			
gtextUNICODE	String	Unicode text string.	NULL
gtextAlign	Not applicable	Alignment on curve: 0 Stretch each line of text to fit width 1 Center text on width 2 Left justify 3 Right justify 4 Spread letters out to fit width 5 Spread words out to fit width	1
gtextSize	Fixed	Default point size.	2,359,296
gtextSpacing	Fixed	Adjust the spacing between characters (1.0 is normal).	65,536
gtextFont	String	Font name.	NULL
fGtext	Boolean	**True** if the text effect properties (**gtext***) are used. **False** if these properties are ignored.	FALSE
gtextFVertical	Boolean	If available, an @ font should be used. Otherwise, rotate individual characters 90 degrees counter-clockwise.	FALSE
gtextFKern	Boolean	Use character pair kerning if it is supported by the font.	FALSE
gtextFTight	Boolean	Adjust the spacing between characters rather than the character advance by the **gtextSpacingratio**.	FALSE
gtextFStretch	Boolean	Stretch the text to fit the shape.	FALSE

gtextFShrinkFit	Boolean	When laying out the characters, consider the glyph bounding box rather than the nominal font character bounds.	FALSE
gtextFBestFit	Boolean	Scale text laid out on a path to fit the path.	FALSE
gtextFNormalize	Boolean	Stretch individual character heights independently to fit.	FALSE
gtextFDxMeasure	Boolean	When laying out characters, measure the distances along the x-axis rather than along the path.	FALSE
gtextFBold	Boolean	Bold font (if available).	FALSE
gtextFItalic	Boolean	Italic font (if available).	FALSE
gtextFUnderline	Boolean	Underline font (if available).	FALSE
gtextFShadow	Boolean	Shadow font (if available).	FALSE
gtextFSmallcaps	Boolean	Small caps font (if available).	FALSE
gtextFStrikethrough	Boolean	Strikethrough font (if available).	FALSE
fGtextOK	Boolean	Text effect (WordArt) supported.	FALSE
gtextFReverseRows	Boolean	Reverse row order.	FALSE
gtextRTF	String	RTF text string.	NULL
3-D Effects			
c3DSpecularAmt	Fixed	Specular amount for the material.	0
c3DDiffuseAmt	Fixed	Diffusion amount for the material.	65,536
c3DShininess	Long integer	Shininess of the material.	5
c3DEdgeThickness	EMU	Specular edge thickness.	12,700
c3DExtrudeForward	EMU	Extrusion amount forward.	0
c3DExtrudeBackward	EMU	Extrusion amount backward.	457,200
c3DExtrusionColor	Color	Color of the extrusion.	
f3D	Boolean	**True** if shape has a three-dimensional (3D) effect, **False** if it does not.	FALSE
fc3DMetallic	Boolean	**True** if shape uses metallic specularity, **False** if it does not.	FALSE
fc3DUseExtrusionColor	Boolean	Extrusion color is set explicitly.	FALSE

fc3DLightFace	Boolean	Light the face of the shape.	TRUE
c3DYRotationAngle	Angle	Degrees about y-axis. If **fc3DconstrainRotation** (a Boolean property which defaults to **True**) is **True**, then the rotation is restricted to x-y rotation. In addition, the final rotation results from first rotating by **c3DYRotationAngle** degrees about the y-axis and then by **c3DXRotationAngle** degrees about the z-axis. If **fc3DconstrainRotation** is **False**, then the final rotation results from a single rotation of **c3DrotationAngle** about the axis specified by **c3DrotationAxisX**, **c3DrotationAxisY**, and **c3DrotationAxisZ**.	0
c3DXRotationAngle	Angle	Degrees about x-axis.	0
c3DRotationAxisX	Long integer	These keywords specify the rotation axis. Only their relative magnitudes matter.	100
c3DRotationAxisY	Long integer	See meaning for **c3DYRotationAxisX**.	0
c3DRotationAxisZ	Long integer	See meaning for **c3DYRotationAxisX**.	0
c3DRotationAngle	Angle	The rotation about the axis (defined previously in the **c3DRotationAxisX**, **Y**, and **Z** parameter sections)	0
fC3DRotationCenterAuto	Boolean	If **fC3DRotationCenterAuto** is **True**, then the rotation will be about the center of the 3-D bounding cube of the 3-D group; otherwise, the rotation center will be about **c3DRotationCenterX**, **c3DRotationCenterY**, and **c3DRotationCenterZ**.	FALSE
c3DRotationCenterX	Fixed	Rotation center (X). The **X** and **Y** values are a 16.16 fraction of the geometry width and height, with (0,0) being at the center of the geometry. The **Z** value must be in absolute units (EMUs).	0

c3DRotationCenterY	Fixed	Rotation center (Y). If **fC3DRotationCenterAuto** is **True**, then the rotation will be about the center of the 3-D bounding cube of the 3-D group; otherwise, the rotation center will be about **c3DRotationCenterX**, **c3DRotationCenterY**, and **c3DRotationCenterZ**. The **X** values and **Y** values are a fraction of the geometry width and height, with (0,0) being at the center of the geometry. The **Z** value is in absolute units.	0
c3DRotationCenterZ	EMU	See meaning for **c3DRotationCenterY**.	0
c3DRenderMode	Long integer	0 Render with full detail 1 Render as a wire frame 2 Render a bounding cube	Not applicable
c3DXViewpoint	EMU	X view point.	1,250,000
c3DYViewpoint	EMU	Y view point.	-1,250,000
c3DZViewpoint	EMU	Z view distance.	9,000,000
c3DOriginX	Fixed	The following **c3DOriginY** and **c3DSkewAngle** values define the origin relative to the viewpoint origin measured. These values are 16.16 numbers that specify the position of the origin within the shape bounding box, as multiples of the width and height of that bounding box and relative to the center (that is, they are displaced from the center). When these values are applied the actual transformed shape path is used, rather than the shape geometry (compare with the shadow and perspective values that work on the geometry bounding box, not the actual points). This means that a shape that extends outside the geometry bounding box (such as a text effect) is handled "correctly" for the calculation of the 3-D origin.	32,768
c3DOriginY	Fixed	See meaning for **c3DOriginX**.	-32,768
c3DSkewAngle	Fixed	Skew angle.	-8,847,360
c3DSkewAmount	Long integer	Percentage skew amount.	50

c3DAmbientIntensity	Fixed	Ambient intensity should be low (0 to .1) to avoid washed out appearance.	20,000
c3DKeyX	Long integer	Key light source direction. Values may be any number; only their relative magnitudes matter.	50,000
c3DKeyY	Long integer	See meaning for **c3DKeyX**.	0
c3DKeyZ	Long integer	See meaning for **c3DKeyX**.	10,000
c3DKeyIntensity	Fixed	Fixed point intensity. Theoretical maximum is 1, but may be higher.	38,000
c3DFillX	Long integer	Fill light source direction; only their relative magnitudes matter. This direction defines a second light source arbitrarily called the "fill light." Generally this will be positioned 90-180 degrees away from the key light and very roughly in front of the scene to fill in any harsh shadows. This fill will be dim compared to the first light source. Theoretically it should be non-harsh, but harsh fill lighting looks better sometimes.	-50,000
c3DFillY	Long integer	See meaning for **c3DfillX**.	0
c3DFillZ	Long integer	See meaning for **c3DfillX**.	10,000
c3DFillIntensity	Fixed	Theoretical maximum is 1, but may be higher.	38,000
fc3DParallel	Boolean	**True** if the fill has parallel projection, **False** if it does not. If **fc3DParallel** is **True**, the **fc3DKeyHarsh** and **fc3DFillHarsh** properties determine the parallel projection used. A skew amount of 0 means the projection is orthographic.	TRUE
fc3DKeyHarsh	Boolean	**True** if key lighting is harsh, **False** if it is not.	TRUE
fc3DFillHarsh	Boolean	**True** if fill lighting harsh, **False** if it is not.	FALSE
c3DCrMod	Color	Modification for BW views.	Undefined
c3DTolerance	Fixed	3D tolerance.	30,000
Perspective			
perspectiveOffsetX	Fixed	The values define a transformation matrix. Each value is scaled by the perspectiveWeight parameter.	0
perspectiveOffsetY	Fixed	See meaning for **perspectiveOffsetX**.	0

perspectiveOriginX	Fixed	Perspective x origin.	32,768
perspectiveOriginY	Fixed	Perspective y origin.	32,768
perspectivePerspectiveX	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectivePerspectiveY	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleXToX	Fixed	See meaning for **perspectiveOffsetX**.	65,536
perspectiveScaleXToY	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleYToX	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleYToY	Fixed	See meaning for **perspectiveOffsetX**.	65,536
perspectiveType	Transform type	Where transform applies: 0 Absolute 1 Shape 2 Drawing	1
perspectiveWeight	Fixed	Scaling factor.	256
fPerspective	Boolean	On/off.	Not applicable

Black and White Modes

bWMode	Black and white mode	Settings for modifications to be made when in different forms of black and white mode: 0 Color 1 Automatic 2 Grayscale 3 Light grayscale 4 Inverse gray 5 Gray outline 6 Black TextLine 7 High contrast 8 Black 9 White 10 Don’t show 11 Number of black and white modes	1
bWModeBW	Black and white mode	See meaning for **bWMode**.	1
bWModePureBW	Black and White Mode	See meaning for **bWmode**.	1

The format of the value depends on the property name it is paired with. Many values are simple single numbers. Distances are expressed in EMU units. There are 12,700 EMU units in a point hence 914,400 in an inch and 360,000 cm-1. Fractional or fixed values are expressed using units that are 1/65536th of a whole. Angles are expressed as fractions of a degree. Colors are 24-bit color values. Booleans have two possible values: 1 for **True** and 0 for **False**.

EmbedShape

Signature

function EmbedShape ($ShapeType, $ShapeText, $TextFmt, $ShapeFormat, $ShapePosition='',$OtherProp='')

Description

This method return the code of the shape defined by parameters. With this method you can embed a shape into the cells of a table.

Parameters

See InsertShape

EmbedTable

Signature

function EmbedTable($TblCnt, $TableFmt = '', $CellFmt = '', $Header = False, $CellWidths='', $RowWidth='')

Description

This method return the cod of the table that can be embeded in shapes for instance. Cannot be used to create nested tables!

Parameters

$TblCnt - this is the content of the table. It is an array of arrays with the following structure:

```
$TblCnt=array(
0=>array("Col_11","Col_12",...,"Col_1n"),              -> this is the Row 1 of the table
1=>array("Col_21","Col_22",...,"Col_2m),               -> this is the Row 2 of the table
2=>array("Col_31","Col_32",...,"Col_3s")               -> this is the Row 3 of the table
...
R=>array("Col_(r+1)1","Col_(r+1)2",...,"Col_(r+1)t")); -> this is the Row R of the table
```

$TableFmt - a string or an array of strings (you can have as many elements in array as many rows in the table or less and in that case the last rows will have the same format) defining the format of the table. The string has the following structure:
{TblRowFormat_TblRowBorder_TblRowShading_TblAutoFormat}

TblRowFormat - define the row format for the table. It has the following structure:
'WriteDirection%AutoFit%Align%CellMargins%CellSpacing'

WriteDirection - LR or RL <=> Left to Right or Right to Left
AutoFit - CNT | WIN <=> Auto fit to contents | Auto fit to window
Align - L+Nr | C | R <=> Left + indent in twips | Center | Right
CellMargins - Left99Top99Righ99Bottom99 , where 99 is the distance in twips between the margin and the text inside the table
CellSpacing - Left99 Top99 Righ99 Bottom99 , where 99 is the distance in twips between the two cells (horizontal or vertical)

Example of TblRowFormat :"LR%WIN%C%L11T12R13B14%L21T22R23B24%"

TblRowBorder - define the border styles for the table. It has the following structure:
'DefinitionOfBorder%Width%SpaceBetwenBorder%Color%BordersMargins'

DefinitionOfBorder - a string specifying the border style.

Allowed values:

'Single-thickness | Single thickness | Single'	Single-thickness border.
'Double-thickness | Double thickness'	Double-thickness border.
'Shadowed'	Shadowed border.
'Double'	Double border.

'Dotted'	Dotted border.
'Dotted'	Dotted border.
'Dashed'	Dashed border.
'Hairline'	Hairline border.
'Resembles a frame | Frame'	Border resembles a “Frame.”
'No border specified | No'	No border specified.
'Table cell has no borders | table cell has no'	Table cell has no borders.
'Inset'	Inset border.
'Dashed small'	Dashed small.
'Dot-dashed | Dot dashed'	Dot-dashed border.
'Dot-dot-dashed | Dot-dot dashed | Dot dot-dashed | Dot dot dashed'	Dot-dot-dashed border.
'Outset'	Outset border.
'Triple'	Triple border.
'Thick-thin small | Thick thin small'	Thick-thin border (small).
'Thin-thick small | Thin thick small'	Thin-thick border (small).
'Thin-thick thin small | Thin thick thin small'	Thin-thick thin border (small).
'Thick-thin medium | Thick thin medium'	Thick-thin border (medium).
'Thin-thick medium | Thin thick medium'	Thin-thick border (medium).
'Thin-thick thin medium | Thin thick thin medium'	Thin-thick thin border (medium).
'Thick-thin large | Thick thin large'	Thick-thin border (large).
'Thin-thick large | Thin thick large'	Thin-thick border (large).
'Thin-thick thin large | Thin-thick thin large | Thin thick-thin large | Thin thick thin large'	Thin-thick-thin border (large).
'Wavy'	Wavy border.
'Double wavy'	Double wavy border.
'Striped'	Striped border.
'Embossed'	Embossed border.
'Engraved'	Engraved border.

By default: 'No border specified'

Width - this argument is representing the border width in twips. It is the width in twips of the pen used to draw the paragraph border line and cannot be greater than 75. To obtain a larger border width, please prefix your width with the letter 'D' and you will obtain a width double that the one you had specified. Ex: 'D60' will draw a border with 120 twips in width. If null or missing by default the width will 1.

SpaceBetwenBorder - this argument specify the space in twips between borders and the paragraph.

Color - a string representing the RGB color code of the border to be drawn. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a border with red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'.

BordersMargins - a string representing the border or the borders the format will be applied for. IT has this format: (T?)(L?)(B?)(R?)(H?)(V?), where T ->TOP, L ->LEFT, B->BOTTOM, R ->RIGHT, H ->DIAGONAL TL-BR, V-> DIAGONAL TR-BL

TblRowShading - define the shading for the table. The structure of the field is:
ShadingStyle%ShadingForeColor%ShadingBackColor

ShadingStyle - define the shading style.

Allowed values:

- 0 - 100, specifying a percentage

- Dk Horizontal
- Dk Vertical
- Dk Dwn Diagonal
- Dk Up Diagonal
- Dk Grid
- Dk Trellis
- Lt Horizontal
- Lt Vertical
- Lt Dwn Diagonal
- Lt Up Diagonal
- Lt Grid
- Lt Trellis

By default : '0'

ShadingForeColor - a RGB color code
ShadingBackColor - a RGBcolor code

TblAutoFormat - set the auto formatting flags for the table. This flags are Bordres+Shading+Font+Color+BestFit+HeadeRow+LastRow+FirstColumn+LastColumn. The structure of the field is:
(B?)(S?)(F?)(C?)(E?)(H?)(L?)(I?)(A?), the presence of letter means that the flag is set:

B - Flag sets table auto format to format borders.
S - Flag sets table auto format to affect shading.
F - Flag sets table auto format to affect font.
C - Flag sets table auto format to affect color.
E - Flag sets table auto format to apply best fit.
H - Flag sets table auto format to format the first (header) row.
L - Flag sets table auto format to format the last row.
I - Flag sets table auto format to format the first (header) column.
A - Flag sets table auto format to format the last column.

$CellFmt - describe the formatting specifications for the text to be written in the cells of the table. It can be the name of a style in the style sheet of the document previously added with AddStyle or a string with formatting specifications as it is defined by StyleFormatting in AddStyle method. Also it can be a matrix - a row in the matrix <=> row in the table, a column in the matrix <=> a column in the table. If the table has more rows than the matrix you defined then the last rows will have the same format; if a row in the table has more columns than the row in the matrix then the last column will have the same format !

$Header - if TRUE then the first row of the table will be a header row, that means it will be emitted on each page the table appear.

$CellWidths - an array of arrays of integers specifying the width (in percentage) of each cell of the table. In the case of missing values (rows of cells or only cells) the values from the precedent row of cell width will be used. For accuracy the first row of cells must have all width values specified.

Example for a table with at least 3 rows and 4 columns:

```
$CellWidth=array(
0=>array(15,5,30,50),
0,
2=>array(10,10,30,50));
```

$RowWidth - an integer specifying the width of the table in percentage.

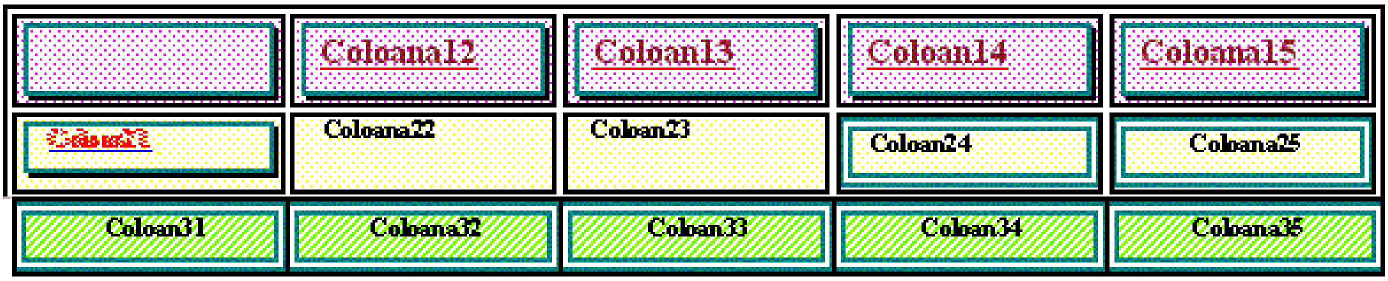
EXAMPLE: for the arguments bellow:

```
$TblContent=array(
0=>array("","Coloana12","Coloan13","Coloan14","Coloana15"),
1=>array("Coloan21","Coloana22","Coloan23","Coloan24","Coloana25"),
2=>array("Coloan31","Coloana32","Coloan33","Coloan34","Coloana35"));

$DefaultTableFormat2
="RL%WIN%L-23%L108T0R108B0%L18T18R18B18%_single%32%2%000000000%TLBR_20%25
5000255%255255255_BSFCEHLIA";
$DefaultTableFormat1
="RL%WIN%L-23%L108T0R108B0%L18T18R18B18%_single%32%2%000000000%TLBR_20%25
5255000%255255255_BSFCEHLIA";
$DefaultTableFormat
="RL%WIN%L0%L108T0R108B0%L18T0RB0%_single%32%2%000000000%TLBR_dk Up
Diagonal%128255000%255255255_BSFCEHLIA";
$DefaultCellFormat="PARFMT:[{left_body
text_Left0Right0_^_Before0After0_Single_W_^}]BRDRDEF:[{shadowed_000128128_32_3_tblr}]FONT
DEF:[{^_Bold_18_140020029_word255000000_superscript_^_^_^_^_^}]";
$DefaultCellFormat1="PARFMT:[{left_body
text_Left0Right0_^_Before0After0_Single_W_^}]FONTDEF:[{Times New
Roman_Bold_12_255_word000000255_^_^_^_^_^_Shimmer}]";
$StyleFormatEndNoteRef="FONTDEF:[{^_^_^_^_^_superscript_^_^_^_^_^}]";
$StyleFormatHeaderText="PARFMT:[{centered_body
text_Left4320Right0_H3_Before0After0_Single_W_^}]";

$TabelFormat=array($DefaultTableFormat2,$DefaultTableFormat1,$DefaultTableFormat);
$CellFormat=array(
0=>array($DefaultCellFormat,$DefaultCellFormat,$DefaultCellFormat,$DefaultCellFormat,$DefaultCell
Format),
1=>array($DefaultCellFormat1,"BRDRDEF:[{none_000128128_32_3_tblr}]FONTDEF:[{^_^_^_^_^_sup
erscript_^_^_^_^_Shimmer}]",$StyleFormatEndNoteRef,"BRDRDEF:[{double_000128128_32_3_tblr}]F
ONTDEF:[{^_^_^_^_^_superscript_^_^_^_^_Las Vegas}]",$StyleFormatHeaderText));
```

with this call:

```
$rtf->InsertTable($TblContent,$TableFormat,$CellTextFormat)
```

then you'll get this table :



	Coloana12	Coloan13	Coloan14	Coloana15
Coloan21	Coloana22	Coloan23	Coloan24	Coloana25
Coloan31	Coloana32	Coloan33	Coloan34	Coloana35

Error

Signature

function Error($ErrorCode = '')

Description

Will bring up an error message according to the error code. The message will be in the language selected for the object.

In order to extend error messages please see the _getError function.

Parameters

$ErrorCode – a number corresponding to the number of the message.

FRTF – the constructor

Signature

function FRTF ($Orientation =** *'P'*, **$PageFormat = *'A4'*, **$Lang =** *'EN'*)

Description

This is the constructor of the class.

Parameters

$Orientation – specify the orientation of the page Portrait/Landscape

Allowed values:

- **'L','l' or anything that starts with 'L','l' for Landscape**
- **Anything else for Portrait**

By default: 'Portrait'

$PageFormat – specify the size of the page

Allowed values:

- **Letter**
- **Legal**
- **Executive**
- **A3**
- **A4**

By default : 'Letter'

$Lang – specify the language the error messages will be displayed in

Allowed values:

- **English | Engleza | En | E**
- **Romanian | Romana | Ro | R**
- **French | Franceza | Fr | F**
- **Italian | Italiana | It | I**
- **German | Germana | De | D**

By default: 'English'

InsertAnnotation

Signature

function InsertAnnotation ($ID, $Author, $Comments, $FormatComments = '', $Time = '', $Icon = '', $Data = '', $Ref = '', $Parent = '')

Description

This method insert an annotation (comment) at the current position in the current section or in the document if no section was added.

Parameters

$ID – the ID of the user who made the comment.

$Author – the name of the author of the comment.

$Comments – the text of the annotation.

$FormatComments – the paragraph style used for the comment .

$Time – the time stamp of the annotation

$Icon – the icon linked to the annotation.

$Data – the data stamp of the annotation.

$Ref – specify the ID of the parent of annotation.

InsertBookmark

Signature

function InsertBookmark ($BookmarkTag, $FirstCol = '', $LastCol = '')

Description

This method indicates the start of a bookmark for the document.

Note: this method is used in conjunction with CloseBookmark.

Parameters

$BookmarkTag – the tag that mark the start of the bookmark.

$FirstCol – is used to denote the first column of a table covered by a bookmark. If it is not included, the first column is assumed.

$Lastcol – is used to denote the last column. If it is not used, the last column is assumed.

InsertDocumentVariable

Signature

function InsertDocumentVariable ($VarName, $VarType)

Description

This method insert a document variable which can be accessed through macros.

Parameters

$VarName – variable name.

$VarType – variable type

InsertEndNote

Signature

function InsertEndNote ($EndText , $EndNoteStyleFormat = '', $EndNoteTextFormat = '', $AEPlace = '', $AEJustif = '', $AEBeginNumb = '', $AENumbPolicy = '',$AENumbStyle = '')

Description

This method define the endnote type for the current document.

Parameters

$EndText – the text of the endnote.

$EndNoteStyleFormat – this is the paragraph style for the reference of the endnote.

$EndNoteTextFormat – this is the paragraph style for the text of the endnote.

$AEPlace – specify the place of endnote (end of document/ end of section).

Allowed values:

'section | sect | sec | s'	endnote at end of section
'document | docum | doc | d'	endnote at end of document

By default : 'Section'

$AEJustify – a string specifying the endnote alignment.

Allowed values:

top | t | beneath text	endnote is top justified
bottom | bot | b | bottom of page	endnote is bottom justified

By default : 'Bottom of page'

$AEBeginNumb – beginning endnote number - default value 1.

$AENumbPolicy – a string specifying the endnote restart number policy.

Allowed values:

'page | pag | p | pagina'	restart numbering each page
'section | sec | s | sectiune'	restart numbering each section
'continuous | cont | c | continuu'	continuous footnote numbering

By default : 'Continuous'

$AENumbStyle – a string specifying the endnote numbering style.

Allowed values:

'arabic'	—Arabic numbering (1, 2, 3, ¼).
'alphabetic lowercase | alfabetic mic | a l | al | a mic | amic'	—Alphabetic lowercase (a, b, c,

¼).

- **'alphabetic uppercase | alfabetic mare | a u | au | a mare | amare'** —Alphabetic uppercase (A, B, C, ¼).
- **'roman lowercase | roman mic | r l | rl | r mic | rmic'** —Roman lowercase (i, ii, iii, ¼).
- **'roman uppercase | roman mare | r u | ru | r mare | rmare'** —Roman uppercase (I, II, III, ¼).
- **'chicago'** —Chicago Manual of Style (*, †, §, §).
- **'korean 1 | korean1 | kor1'** —Korean numbering 1 (*chosung).
- **'korean 2 | korean2 | kor2'** —Korean numbering 2 (*ganada).
- **'circle'** —Circle numbering (*circlenum).
- **'kanji 1 | kanji1 | k1'** —kanji numbering without the digit character (*dbnum1).
- **'kanji 2 | kanji2 | k2'** —kanji numbering with the digit character (*dbnum2).
- **'kanji 3 | kanji3 | k3'** —kanji numbering 3 (*dbnum3).
- **'kanji 4 | kanji4 | k4'** —kanji numbering 4 (*dbnum4).
- **'double_byte | double byte'** —double-byte numbering (*dbchar).
- **'chinese 1 | chi 1 | c1'** —Chinese numbering 1 (*gb1).
- **'chinese 2 | chi 2 | c2'** —Chinese numbering 2 (*gb2).
- **'chinese 3 | chi 3 | c3'** —Chinese numbering 3 (*gb3).
- **'chinese 4 | chi 4 | c4'** —Chinese numbering 4 (*gb4).
- **'chinese zodiac 1 | zodiac 1 | zodiac1 | z1'** —Chinese Zodiac numbering 1 (* zodiac1). 甲、乙、丙… ~~甲、乙、丙…~~ 甲、乙、丙…
- **'chinese zodiac 2 | zodiac 2 | zodiac2 | z2'**—Chinese Zodiac numbering 2 (* zodiac2). 子、丑、寅…
- **'chinese zodiac 3 | zodiac 3 | zodiac3 | z3'**—Chinese Zodiac numbering 3 (* zodiac3).

By default : 'Arabic'

Note: The arguments :$AEPlace, $AEJustif, $AEBeginNumb, $AENumbPolicy,$AENumbStyle are used by InsertEndNote only if the DefineEndNote method was not called for the current document.

InsertField

Signature

function InsertField ($FieldToInsert, $FieldResult = '', $FieldMod = '', $FieldAlt = '', $FmtTxt = '')

Description

This method insert a field at the current position in the current section or in the document if no section was added.

Parameters

$FieldToInsert – a string specifying the field instruction.

$FieldResult – the most recent calculated result of the field.

$FieldMod – control words that alter the interpretation of the field.

Allowed values:

- 'dirty | d' A formatting change has been made to the field result since the field was last updated.
- 'edit | e' Text has been added to, or removed from, the field result since the field was last updated.
- 'locked | lock | l' Field is locked and cannot be updated.
- 'private | priv | p' Result is not in a form suitable for display (for example, binary data used by fields whose result is a picture).

By default : 'Section'

$FieldAlt – specify that the given field reference is to an endnote.

$FmtTxt – describe the formatting specifications for the text to be write - the result of the field. It can be the name of a style in the style sheet of the document previously added with AddStyle or a string with formatting specifications as it is defined by StyleFormatting in AddStyle method

InsertFooter

Signature

function InsertFooter($FooterText, $FooterType = '', $FooterStyleFormat = '')

Description

This method insert a footer containing the $FooterText in the current section.

Parameters

$FooterText – the text to be displayed in the footer. You can insert fields and pictures in the header by using '!~'/'!#' separators (see $FieldSep[0]/$FieldSep[1] in *conf.inc.php*). That it if you pass the following string "Page !~{PAGE *MERGEFORMAT }~!/!~{NUMPAGES *MERGEFORMAT }~!" in the footer text you'll get page number from page numbers <=> "Page X of Y". The following string will insert a picture in the footer :"Picture !#E:\\imagini\\log.jpg,^_^_^_^_^_^,line_both_^_0.1:Column_0:par_MA#!'" It is possible to use some HTML tags in the $HeaderTags, for a detailed list see the array $AuthTags in *conf.inc.php*

$FooterType – a string specifying the type of footer. It will be used if a footer type has not been defined for the current section, see DefineFooters**.**

Allowed values:

- **All | a -> Footers will appear on all pages**
- **Left | l -> Footers will appear only on left pages, only if** SetFacingp **has been called for the document.**
- **Right | r -> Footers will appear only on right pages, only if** SetFacingp **has been called for the document.**
- **First | f -> Footers will appear only on first page if the first page has a special format (** Sect_FirstPage **has been called)**

By default : 'All'

$FooterStyleFormat – it is a string which define the style of the footer. For more details about how to define a style please look at AddStyle**. If it is not specified then the style defined by $StyleFormatFooterRef in *conf.inc.php* will be used.**

InsertFootNote

Signature

function InsertFootNote ($FootText, $FootNoteStyleFormat = '', $FootNoteTextFormat = '', $FNPlace = '', $FNJustif = '', $FNBeginNumb = '', $FNNumbPolicy = '', $FNNumbStyle = '')

Description

This method insert a footnote at the current position in the current section, or document if no section was added.

Parameters

$FootText – the text of the footnote.

$FootNoteStyleFormat – this is the paragraph style for the reference of the footnote.

$FootNoteTextFormat – this is the paragraph style for the text of the footnote.

$FNPlace – specify the place of footnote (end of document/ end of section).

Allowed values:

'section | sect | sec | s'	footnote at end of section
'document | docum | doc | d'	footnote at end of document

By default : 'Section'

$FNJustify – a string specifying the footnote alignment.

Allowed values:

top | t | beneath text	footnote is top justified
bottom | bot | b | bottom of page	footnote is bottom justified

By default : 'Bottom of page'

$FNBeginNumb – beginning footnote number - default value 1.

$FNNumbPolicy – a string specifying the footnote restart number policy.

Allowed values:

'page | pag | p | pagina'	restart numbering each page
'section | sec | s | sectiune'	restart numbering each section
'continuous | cont | c | continuu'	continuous footnote numbering

By default : 'Continuous'

$FNNumbStyle – a string specifying the footnote numbering style.

Allowed values:

'arabic'	—Arabic numbering (1, 2, 3, ¼).

- 'alphabetic lowercase | alfabetic mic | a l | al | a mic | amic' —Alphabetic lowercase (a, b, c, ¼).
- 'alphabetic uppercase | alfabetic mare | a u | au | a mare | amare' —Alphabetic uppercase (A, B, C, ¼).
- 'roman lowercase | roman mic | r l | rl | r mic | rmic' —Roman lowercase (i, ii, iii, ¼).
- 'roman uppercase | roman mare | r u | ru | r mare | rmare' —Roman uppercase (I, II, III, ¼).
- 'chicago' —Chicago Manual of Style (*, †, §, §).
- 'korean 1 | korean1 | kor1' —Korean numbering 1 (*chosung).
- 'korean 2 | korean2 | kor2' —Korean numbering 2 (*ganada).
- 'circle' —Circle numbering (*circlenum).
- 'kanji 1 | kanji1 | k1' —kanji numbering without the digit character (*dbnum1).
- 'kanji 2 | kanji2 | k2' —kanji numbering with the digit character (*dbnum2).
- 'kanji 3 | kanji3 | k3' —kanji numbering 3 (*dbnum3).
- 'kanji 4 | kanji4 | k4' —kanji numbering 4 (*dbnum4).
- 'double_byte | double byte' —double-byte numbering (*dbchar).
- 'chinese 1 | chi 1 | c1' —Chinese numbering 1 (*gb1).
- 'chinese 2 | chi 2 | c2' —Chinese numbering 2 (*gb2).
- 'chinese 3 | chi 3 | c3' —Chinese numbering 3 (*gb3).
- 'chinese 4 | chi 4 | c4' —Chinese numbering 4 (*gb4).
- 'chinese zodiac 1 | zodiac 1 | zodiac1 | z1' —Chinese Zodiac numbering 1 (* zodiac1). 甲、乙、丙… ~~甲、乙、丙…~~ 甲、乙、丙…
- 'chinese zodiac 2 | zodiac 2 | zodiac2 | z2'—Chinese Zodiac numbering 2 (* zodiac2). 子、丑、寅…
- 'chinese zodiac 3 | zodiac 3 | zodiac3 | z3'—Chinese Zodiac numbering 3 (* zodiac3).

By default : 'Arabic'

Note: The arguments :$FNPlace, $FNJustif, $FNBeginNumb, $FNNumbPolicy,$FNNumbStyle are used by InsertFootNote only if the DefineFootNote method was not called for the current document.

InsertFormField_CheckBox

Signature

function InsertFormField_CheckBox($FieldName, $Size, $Default, $Calc_On_Exit=TRUE, $Protected=FALSE, $StatusBar_Help="", $Key_Help="", $Entry_Macro="", $Exit_Macro="")

Description

This method insert a check box form field at the current position in the current section or in the document if no section was added.

Parameters

$FieldName – a string specifying the form field name.

$Size – type of size selected for check box field.

Allowed values:

- Auto The size of the chekc box field will be determined automatically.
- xxx An integer between 1 and 1584 specifying the exactly size of the check box field.

By default : Auto

$Default – specify if the check box field is checked or not.

Allowed values:

- TRUE The check box field is checked.
- FALSE The check box field is not checked.

By default : TRUE

$Calc_On_Exit – control words that alter the interpretation of the field.

Allowed values:

- TRUE The value of the field will be calculated on the field is loosing the focus.
- FALSE The value of the field will be not calculated on leaving the field.

By default : TRUE

$Protected – specify if the field is protected or not.

Allowed values:

- TRUE The field si protected.
- FALSE The is not protected.

By default : FALSE

$Status_Bar_Help – the string to be dispalyed on the status bar when the field is getting focus.

$Key_Help – the string to be displayed when the field is getting focus and the help key is pressed.

$Entry_Macro – the name of the macro to be executed when the field is getting focus.

$Exit_Macro – the name of the macro to be executed when the field is loosing the focus.

InsertFormField_DropDown

Signature

function InsertFormField_DropDown($FieldName, $ItemList, $Default, $Calc_On_Exit=TRUE, $Protected=FALSE, $StatusBar_Help="", $Key_Help="", $Entry_Macro="", $Exit_Macro="")

Description

This method insert a drop down form field at the current position in the current section or in the document if no section was added.

Parameters

$FieldName – a string specifying the form field name.

$ItemList – an array with the items to be into the drop down field.

$Default – an integer specifying the number of the default item of the list.

$Calc_On_Exit – control words that alter the interpretation of the field.

Allowed values:

- TRUE The value of the field will be calculated on the field is loosing the focus.
- FALSE The value of the field will be not calculated on leaving the field.

By default : TRUE

$Protected – specify if the field is protected or not.

Allowed values:

- TRUE The field si protected.
- FALSE The is not protected.

By default : FALSE

$Status_Bar_Help – the string to be dispalyed on the status bar when the field is getting focus.

$Key_Help – the string to be displayed when the field is getting focus and the help key is pressed.

$Entry_Macro – the name of the macro to be executed when the field is getting focus.

$Exit_Macro – the name of the macro to be executed when the field is loosing the focus.

InsertFormField_Text

Signature

function InsertFormField_Text($FieldName, $Type, $Format, $Default="", $Length="Unlimited", $Result="", $Calc_On_Exit=TRUE, $Protected=FALSE, $StatusBar_Help="", $Key_Help="", $Entry_Macro="", $Exit_Macro="")

Description

This method insert a text box form field at the current position in the current section or in the document if no section was added.

Parameters

$FieldName – a string specifying the form field name.

$Type – the type of the text box field.

Allowed values:

0, Regular text, Regular Text	Regular Text
1, Number	Number
2, Date	Date
3, Current date	Current date
4, Current time	Current time
5, Calculation	Calculation

By default : Regular Text

$Format – a string specifying the format of the text field.

Allowed values:

Uppercase, Upper, U	Uppercase
Lowercase, Low, L	Lowercase
First capital, First, Capital, F	First Capital
Title case, Title, Case, T	Title case

By default : No format

$Default – specify if the check box field is checked or not.

Allowed values:

TRUE	The check box field is checked.
FALSE	The check box field is not checked.

By default : TRUE

$Length – the length of the text field = number of characters for the text field.

Allowed values:

Unlimited	The lenght of the text field is not limited.

- xxx An integer between 1 and 32767 specifying the exactly lenght of the text field.

By default : Auto

$Result – a string specifying the most recent calculated result of the text field.

$Calc_On_Exit – control words that alter the interpretation of the field.

Allowed values:

- TRUE The value of the field will be calculated on the field is loosing the focus.
- FALSE The value of the field will be not calculated on leaving the field.

By default : TRUE

$Protected – specify if the field is protected or not.

Allowed values:

- TRUE The field si protected.
- FALSE The is not protected.

By default : FALSE

$Status_Bar_Help – the string to be dispalyed on the status bar when the field is getting focus.

$Key_Help – the string to be displayed when the field is getting focus and the help key is pressed.

$Entry_Macro – the name of the macro to be executed when the field is getting focus.

$Exit_Macro – the name of the macro to be executed when the field is loosing the focus.

InsertHeader

Signature

function InsertHeader($HeaderText, $HeaderType = '', $HeaderStyleFormat = '')

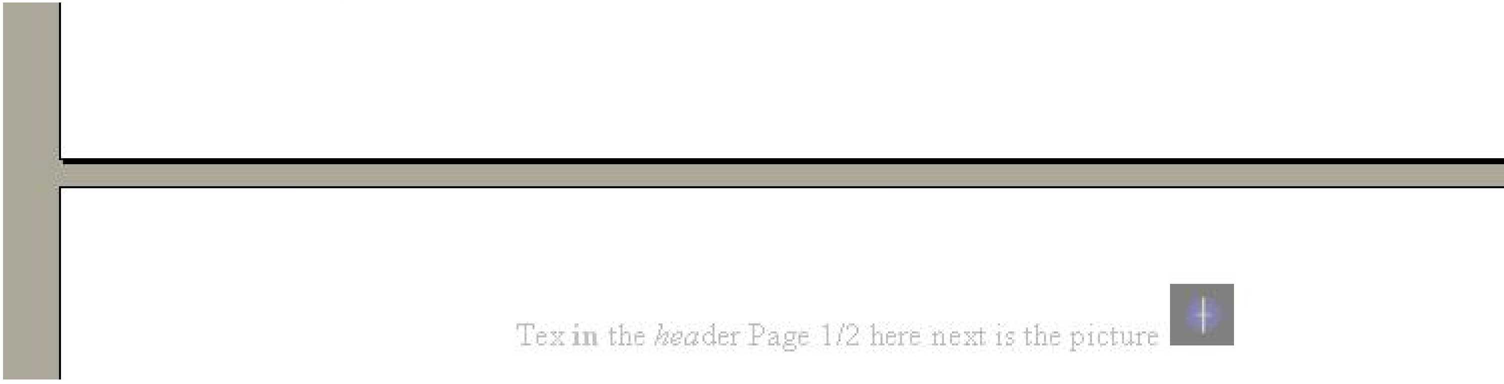
Description

This method insert a header containing the $HeaderText in the current section.

Parameters

$HeaderText – the text to be displayed in the header. You can insert fields and pictures in the header by using '!~'/'!#' separators (see $FieldSep[0]/$FieldSep[1] in *conf.inc.php*). That it if you pass the following string "Page !~{PAGE *MERGEFORMAT }~!/!~{NUMPAGES *MERGEFORMAT }~!"in the header text you'll get page number from page numbers <=> "Page X of Y". The following string will insert a picture in the header :"Picture !#E:\\imagini\\log.jpg,^_^_^_^_^_^,line_both_^_0.1:Column_0:par_MA#!'"
It is possible to use some HTML tags in the $HeaderTags, for a detailed list see the array $AuthTags in *conf.inc.php*.

Example:
This method call: $rtf->InsertHeader('Tex<script language=Javascript>t <b>in</b> the <I>hea</i>der Page !~{PAGE *MERGEFORMAT }~!/!~{NUMPAGES *MERGEFORMAT }~! here next is the picture !#D:\\imagini\\log.jpg,^_^_^_^_^_^,line_both_^_0.1:Column_0:par_MA#!');
will generate the following header:



Tex in the *header* Page 1/2 here next is the picture

$HeaderType – a string specifying the type of footer. It will be used if a header type has not been defined for the current section, see DefineHeaders.

Allowed values:

- All | a -> Headers will appear on all pages
- Left | l -> Headers will appear only on left pages, only if SetFacingp has been called for the document.
- Right | r -> Headers will appear only on right pages, only if SetFacingp has been called for the document.
- First | f -> Headers will appear only on first page if the first page has a special format (Sect_FirstPage has been called)

By default : 'All'

$HeaderStyleFormat – it is a string which define the style of the header. For more details about how to define a style please look at AddStyle. If it is not specified then the style defined by $StyleFormatHeaderRef in *conf.inc.php* will be used.

InsertPicture

Signature

function InsertPicture($PictPath, $PictFormat = '', $ShapeFormat = '', $OtherProp = '')

Description

This method insert a picture from the file specified by $PictPath according to the format detailed by $PictFormat, $ShapeFormat and $OtherProp.

Parameters

$PictPath – path to the file the picture will be inserted from. Only .JPG and .PNG accepted. The images can be inserted either from a local disk or from World Wide Web.

$PictFormat – a string defining the format of the picture. It has the following structure: {PictFillColor_PictLineDef_PictSize_PictScale_PictCrop_PictImageControl_PictWeb}

PictFillColor - a string representing the fill color code of the picture. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a border with red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'. PictLineDef - define the borders properties for the picture. It has the following structure: 'Color<123123123>Dashed<>Style<>Weight<>' or 'C<123123123>D<>S<>W<>' where

Color: is a RGB color code see PictFillColor for details about RGB color code. Dash style Dashing: this is the dashed style of the line.

Allowed values (the values are self-explanatory):

- **'Solid line | solid | s'**
- **'Dashed line windows | dashed windows | dash w'**
- **'Dotted line windows | dotted windows | dot w'**
- **'Dash-dotted line windows | dash dotted line windows | dash dotted windows | dash-dotted windows | d-d w | d d w | dd w | ddw'**
- **'Dash-dot-dotted line windows | dash-dot-dotted windows | d-d-d w | ddd w | dddw'**
- **'Dotted line | dotted | dot'**
- **'Dashed line | dashed | dash'**
- **'Long dashed line | long dashed | l d | ld'**
- **'Dash-dotted line | dash dotted line | dash dotted | dash-dotted | d d |dd'**
- **'Long dash-dotted line | long dash dotted line | long dash dotted | long dash-dotted | l d-d | l dd | ldd'**
- **'Long dash-dot-dotted line | long dash-dot-dotted | l d-d-d | l ddd | lddd'**

By default : 'Solid line'

Line style: this is the style of the line.

Allowed values (the values are self-explanatory):

- **'Single | s'**

- 'Double of equal width | double equal | d e | de'
- 'Dotted line windows | dotted windows | dot w'
- 'Double one thick one thin | double one one | d o o | doo'
- 'Double reverse order | double reverse | d r | dr'
- 'Three'

By default : 'Single'

Weigth: the width of the line (border) [0 - 1584 pt]
Example:"Color<128128128>Dashed<Solid>Style<Single>Weight<2>"or "C<128128128>D<Solid>S<Single>W<2>", in this case for Dashed and Style the values by default will be used.

PictSize - define the size of the picture. It has this structure: Height99Width99 or H99W99, where 99 is a number between 0 - 22".
Example:"Height3.12Width6.76"
PictScale - define the scale of the picture. It has this structure: Height99Width99LR or H99W99LR, where 99 is a percent between 0-10675.
L -> if present then *Lock aspect ration* checkbox will be checked.
R -> if present then *Relative to original size* checkbox will be checked.
Example:"Height100Width1000LR"
PictCrop - define the crop parameters for the picture. It has this structure: Left99Top99Right99Bottom99 or L99T99R99B99, where 99 is a number between -22"-22".
Example:"Left12Top13Right4Bottom5"
PictImageControl - define the brightness of the picture. It has this structure: <TYPE>Brightness99Contrast99. where 99 is a number between 0-100.

Allowed values for TYPE (the values are self-explanatory):

- 'Automatic | auto | a'
- 'Grayscale | gray | g'
- 'Black ?
- 'Watermark | water | w'

By default : 'Automatic'

Example:"<Automatic>Brightness100Contrast100"

PictWeb - the text to be displayed in internet environment. Web browsers display alternative text while pictures are loading or if they are missing. Web search engines use alternative text to help find Web pages.

$ShapeFormat – a string defining the format of the shape which will contains the picture. It has the following structure:
{ShapeWrapStyle_ShapeWrapText_ShapeDistance_ShapeHoriz_ShapeVert_ShapeOptions}

ShapeWrapStyle - define the wrap style.

Allowed values (the values are self-explanatory):

- 'Wrap around shape | square | squ | q'
- 'Wrap tightly around shape | tight | tightly | ti'
- 'Wrap text through shape | through | th'
- 'Wrap around top and bottom of shape | top and bottom | tb'
- 'Wrap behind text | behind | b'
- 'In front of text | front | f'
- 'In line with text | line | l'

By default : 'Wrap around shape'

ShapeWrapText - define the wrap text mode on sides of shape.

Allowed values (the values are self-explanatory):

- 'Wrap both sides of shape | both sides | both | b'
- 'Wrap left side only | left side | left | l'
- 'Wrap right side only | right side | right | r'
- 'Wrap only on largest side | largest side | largest | la'

By default : 'Wrap both sides of shape'

ShapeDistance - define the distances from text. It has this structure: Left99Top99Right99Bottom99 or L99T99R99B99, where 99 is a number between 0"-22".

Example:"Left12Top13Right4Bottom5"

ShapeHoriz - define the horizontal alignment of the shape. It has the following structure: ALIGN:RELATIVETO.

Allowed values for ALIGN (the values are self-explanatory):

- Number between -22"-22" - that correspond to absolute position
- 'Left | l'
- 'Centered | center | c'
- 'Right | r'
- 'Inside | i'
- 'Outside | o'

By default : 'Left'

Allowed values for RELATIVETO (the values are self-explanatory):

- 'Margin | mar | m'
- 'Page | pg | p'
- 'Column | col | c'
- 'Character | char | ch'

By default : 'Column'

Example:"'Left:Column' or 'Centered:Page' or 'Inside:Margin' or '-12:Character'"

ShapeVert - define the vertical alignment of the shape. It has the following structure: ALIGN:BELLOWTO.

Allowed values for ALIGN (the values are self-explanatory):

- Number between -22"-22" - that correspond to absolute position
- 'Top | t'
- 'Centered | center | c'
- 'Bottom | b'
- 'Inside | i'
- 'Outside | o'

By default : 'Top'

Allowed values for BELLOWTO (the values are self-explanatory):

- 'Margin | mar | m'
- 'Page | pg | p'

- **'Paragraph | para | par'**
- **'Line | lin | l'**

By default : 'Paragraph'

Example:"'Top:Page' or 'Centered:Line' or 'Inside:Margin' or '-12:Paragraph'"

ShapeOptions - define the supplementary options for shape positioning. These options are *'Move object with text'* | *'Lock anchor'* | *'Allow overlap'*.The structure of this parameter is : (M?)(L?)(A?). If the letter is present the the corresponding option is set.
Example:"'MLA' or 'LA' or 'MA'"

$OtherProp – an array with properties to be applied to the shape. It allows to customize in detail the shape. The format of the array should be: array key = name of the property, array value = value of the property.
Example:array('fLockAspectRatio'=>'1','fLockRotation'=>'0','Rotation'=>65536*10)
The set of properties/values which can be used:

Property	**Meaning**	**Type of value**	**Default**
Lock			
fLockRotation	Boolean	Lock rotation.	FALSE
fLockAspectRatio	Boolean	Lock aspect ratio.	FALSE
fLockAgainstSelect	Boolean	Lock against selection.	FALSE
fLockCropping	Boolean	Lock against cropping.	FALSE
fLockVerticies	Boolean	Lock against edit mode.	FALSE
fLockText	Boolean	Lock text against editing.	FALSE
fLockAdjustHandles	Boolean	Lock adjust handles.	FALSE
fLockAgainstGrouping	Boolean	Lock against grouping.	FALSE
fLockShapeType	Boolean	Lock the shape type (don't allow Change Shape).	FALSE
Text Box			
dxTextLeft	EMU	Left internal margin of the text box.	91,440
dyTextTop	EMU	Top internal margin of the text box.	45,720
dxTextRight	EMU	Right internal margin of the text box.	91,440
dyTextBottom	EMU	Bottom internal margin of the text box.	45,720

WrapText	Not applicable	Wrap text at shape margins: 0 Square 1 Tight 2 None 3 Top bottom 4 Through	0
anchorText	Not applicable	Text anchor point: 0 Top 1 Middle 2 Bottom 3 Top centered 4 Middle centered 5 Bottom centered 6 Bottom centered baseline	0
txflTextFlow	Not applicable	Text flow: 0 Horizontal non-ASCII font 1 Top to bottom ASCII font 2 Bottom to top non-ASCII font 3 Top to bottom non-ASCII font 4 Horizontal ASCII font	0
cdirFont	Direction	Font rotation: 0 Right 1 Down 2 Left 3 Up	0
fAutoTextMargin	Boolean	Use host's margin calculations.	FALSE
scaleText	Long integer	Text zoom and scale.	0

lTxid	Long integer	ID for the text. The value is determined by the host.	0
fRotateText	Boolean	Rotate text with shape.	FALSE
fSelectText	Boolean	TRUE if single click selects text, FALSE if two clicks select text.	TRUE
fFitShapeToText	Boolean	Adjust shape to fit text size.	FALSE
fFitTextToShape	Boolean	Adjust text to fit shape size.	FALSE
WordArt Effect			
gtextUNICODE	String	Unicode text string.	NULL
gtextAlign	Not applicable	Alignment on curve: 0 Stretch each line of text to fit width 1 Center text on width 2 Left justify 3 Right justify 4 Spread letters out to fit width 5 Spread words out to fit width	1
gtextSize	Fixed	Default point size.	2,359,296
gtextSpacing	Fixed	Adjust the spacing between characters (1.0 is normal).	65,536
gtextFont	String	Font name.	NULL
fGtext	Boolean	**True** if the text effect properties (**gtext***) are used. **False** if these properties are ignored.	FALSE
gtextFVertical	Boolean	If available, an @ font should be used. Otherwise, rotate individual characters 90 degrees counter-clockwise.	FALSE
gtextFKern	Boolean	Use character pair kerning if it is supported by the font.	FALSE
gtextFTight	Boolean	Adjust the spacing between characters rather than the character advance by the **gtextSpacingratio**.	FALSE
gtextFStretch	Boolean	Stretch the text to fit the shape.	FALSE

gtextFShrinkFit	Boolean	When laying out the characters, consider the glyph bounding box rather than the nominal font character bounds.	FALSE
gtextFBestFit	Boolean	Scale text laid out on a path to fit the path.	FALSE
gtextFNormalize	Boolean	Stretch individual character heights independently to fit.	FALSE
gtextFDxMeasure	Boolean	When laying out characters, measure the distances along the x-axis rather than along the path.	FALSE
gtextFBold	Boolean	Bold font (if available).	FALSE
gtextFItalic	Boolean	Italic font (if available).	FALSE
gtextFUnderline	Boolean	Underline font (if available).	FALSE
gtextFShadow	Boolean	Shadow font (if available).	FALSE
gtextFSmallcaps	Boolean	Small caps font (if available).	FALSE
gtextFStrikethrough	Boolean	Strikethrough font (if available).	FALSE
fGtextOK	Boolean	Text effect (WordArt) supported.	FALSE
gtextFReverseRows	Boolean	Reverse row order.	FALSE
gtextRTF	String	RTF text string.	NULL
3-D Effects			
c3DSpecularAmt	Fixed	Specular amount for the material.	0
c3DDiffuseAmt	Fixed	Diffusion amount for the material.	65,536
c3DShininess	Long integer	Shininess of the material.	5
c3DEdgeThickness	EMU	Specular edge thickness.	12,700
c3DExtrudeForward	EMU	Extrusion amount forward.	0
c3DExtrudeBackward	EMU	Extrusion amount backward.	457,200
c3DExtrusionColor	Color	Color of the extrusion.	
f3D	Boolean	**True** if shape has a three-dimensional (3D) effect, **False** if it does not.	FALSE
fc3DMetallic	Boolean	**True** if shape uses metallic specularity, **False** if it does not.	FALSE
fc3DUseExtrusionColor	Boolean	Extrusion color is set explicitly.	FALSE

fc3DLightFace	Boolean	Light the face of the shape.	TRUE
c3DYRotationAngle	Angle	Degrees about y-axis. If **fc3DconstrainRotation** (a Boolean property which defaults to **True**) is **True**, then the rotation is restricted to x-y rotation. In addition, the final rotation results from first rotating by **c3DYRotationAngle** degrees about the y-axis and then by **c3DXRotationAngle** degrees about the z-axis. If **fc3DconstrainRotation** is **False**, then the final rotation results from a single rotation of **c3DrotationAngle** about the axis specified by **c3DrotationAxisX**, **c3DrotationAxisY**, and **c3DrotationAxisZ**.	0
c3DXRotationAngle	Angle	Degrees about x-axis.	0
c3DRotationAxisX	Long integer	These keywords specify the rotation axis. Only their relative magnitudes matter.	100
c3DRotationAxisY	Long integer	See meaning for **c3DYRotationAxisX**.	0
c3DRotationAxisZ	Long integer	See meaning for **c3DYRotationAxisX**.	0
c3DRotationAngle	Angle	The rotation about the axis (defined previously in the **c3DRotationAxisX**, **Y**, and **Z** parameter sections)	0
fC3DRotationCenterAuto	Boolean	If **fC3DRotationCenterAuto** is **True**, then the rotation will be about the center of the 3-D bounding cube of the 3-D group; otherwise, the rotation center will be about **c3DRotationCenterX**, **c3DRotationCenterY**, and **c3DRotationCenterZ**.	FALSE
c3DRotationCenterX	Fixed	Rotation center (X). The **X** and **Y** values are a 16.16 fraction of the geometry width and height, with (0,0) being at the center of the geometry. The **Z** value must be in absolute units (EMUs).	0

c3DRotationCenterY	Fixed	Rotation center (Y). If **fC3DRotationCenterAuto** is **True**, then the rotation will be about the center of the 3-D bounding cube of the 3-D group; otherwise, the rotation center will be about **c3DRotationCenterX**, **c3DRotationCenterY**, and **c3DRotationCenterZ**. The **X** values and **Y** values are a fraction of the geometry width and height, with (0,0) being at the center of the geometry. The **Z** value is in absolute units.	0
c3DRotationCenterZ	EMU	See meaning for **c3DRotationCenterY**.	0
c3DRenderMode	Long integer	0 Render with full detail 1 Render as a wire frame 2 Render a bounding cube	Not applicable
c3DXViewpoint	EMU	X view point.	1,250,000
c3DYViewpoint	EMU	Y view point.	-1,250,000
c3DZViewpoint	EMU	Z view distance.	9,000,000
c3DOriginX	Fixed	The following **c3DOriginY** and **c3DSkewAngle** values define the origin relative to the viewpoint origin measured. These values are 16.16 numbers that specify the position of the origin within the shape bounding box, as multiples of the width and height of that bounding box and relative to the center (that is, they are displaced from the center). When these values are applied the actual transformed shape path is used, rather than the shape geometry (compare with the shadow and perspective values that work on the geometry bounding box, not the actual points). This means that a shape that extends outside the geometry bounding box (such as a text effect) is handled "correctly" for the calculation of the 3-D origin.	32,768
c3DOriginY	Fixed	See meaning for **c3DOriginX**.	-32,768
c3DSkewAngle	Fixed	Skew angle.	-8,847,360
c3DSkewAmount	Long integer	Percentage skew amount.	50

c3DAmbientIntensity	Fixed	Ambient intensity should be low (0 to .1) to avoid washed out appearance.	20,000
c3DKeyX	Long integer	Key light source direction. Values may be any number; only their relative magnitudes matter.	50,000
c3DKeyY	Long integer	See meaning for **c3DKeyX**.	0
c3DKeyZ	Long integer	See meaning for **c3DKeyX**.	10,000
c3DKeyIntensity	Fixed	Fixed point intensity. Theoretical maximum is 1, but may be higher.	38,000
c3DFillX	Long integer	Fill light source direction; only their relative magnitudes matter. This direction defines a second light source arbitrarily called the "fill light." Generally this will be positioned 90-180 degrees away from the key light and very roughly in front of the scene to fill in any harsh shadows. This fill will be dim compared to the first light source. Theoretically it should be non-harsh, but harsh fill lighting looks better sometimes.	-50,000
c3DFillY	Long integer	See meaning for **c3DfillX**.	0
c3DFillZ	Long integer	See meaning for **c3DfillX**.	10,000
c3DFillIntensity	Fixed	Theoretical maximum is 1, but may be higher.	38,000
fc3DParallel	Boolean	**True** if the fill has parallel projection, **False** if it does not. If **fc3DParallel** is **True**, the **fc3DKeyHarsh** and **fc3DFillHarsh** properties determine the parallel projection used. A skew amount of 0 means the projection is orthographic.	TRUE
fc3DKeyHarsh	Boolean	**True** if key lighting is harsh, **False** if it is not.	TRUE
fc3DFillHarsh	Boolean	**True** if fill lighting harsh, **False** if it is not.	FALSE
c3DCrMod	Color	Modification for BW views.	Undefined
c3DTolerance	Fixed	3D tolerance.	30,000
Perspective			
perspectiveOffsetX	Fixed	The values define a transformation matrix. Each value is scaled by the perspectiveWeight parameter.	0
perspectiveOffsetY	Fixed	See meaning for **perspectiveOffsetX**.	0

perspectiveOriginX	Fixed	Perspective x origin.	32,768
perspectiveOriginY	Fixed	Perspective y origin.	32,768
perspectivePerspectiveX	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectivePerspectiveY	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleXToX	Fixed	See meaning for **perspectiveOffsetX**.	65,536
perspectiveScaleXToY	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleYToX	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleYToY	Fixed	See meaning for **perspectiveOffsetX**.	65,536
perspectiveType	Transform type	Where transform applies: 0 Absolute 1 Shape 2 Drawing	1
perspectiveWeight	Fixed	Scaling factor.	256
fPerspective	Boolean	On/off.	Not applicable

Black and White Modes

bWMode	Black and white mode	Settings for modifications to be made when in different forms of black and white mode: 0 Color 1 Automatic 2 Grayscale 3 Light grayscale 4 Inverse gray 5 Gray outline 6 Black TextLine 7 High contrast 8 Black 9 White 10 Don’t show 11 Number of black and white modes	1
bWModeBW	Black and white mode	See meaning for **bWMode**.	1
bWModePureBW	Black and White Mode	See meaning for **bWmode**.	1

The format of the value depends on the property name it is paired with. Many values are simple single numbers. Distances are expressed in EMU units. There are 12,700 EMU units in a point hence 914,400 in an inch and 360,000 cm-1. Fractional or fixed values are expressed using units that are 1/65536th of a whole. Angles are expressed as fractions of a degree. Colors are 24-bit color values. Booleans have two possible values: 1 for **True** and 0 for **False**.

InsertShape

Signature

function InsertShape($ShapeType, $ShapeText = '', $TextFmt = '', $ShapeFormat = '', $ShapePosition = '', $OtherProp = '')

Description

This method insert a shape with the type specified by $ShapeType with the text $ShapeText according to the format detailed by $TextFmt, $ShapeFormat and $ShapePosition. Also you can specify more properties for the shape with the $OtherProp parameter.

Parameters

$ShapeType – a string specifying the type of the shape to be inserted.

Allowed values(the values are self-explanatory):

- Rectangle
- Round rectangle
- Ellipse
- Diamond
- Isosceles triangle
- Right triangle
- Parallelogram
- Trapezoid
- Hexagon
- Octagon
- Plus Sign
- Star
- Arrow
- Thick arrow
- Home plate
- Cube
- Balloon
- Seal
- Arc
- Line
- Plaque
- Can
- Donut
- Text simple
- Text octagon
- Text hexagon
- Text curve
- Text wave
- Text ring
- Text on curve
- Text on ring
- Callout 1
- Callout 2

- Callout 3
- Accent callout 1
- Accent callout 2
- Accent callout 3
- Border callout 1
- Border callout 2
- Border callout 3
- Accent border callout 1
- Accent border callout 2
- Accent border callout 3
- Ribbon
- Ribbon2
- Chevron
- Pentagon
- No smoking
- Seal8
- Seal16
- Seal32
- Wedge rectangle callout
- Wedge RRect callout
- Wedge ellipse callout
- Wave
- Folded corner
- Left arrow
- Down arrow
- Up arrow
- Left right arrow
- Up down arrow
- IrregularSeal1
- IrregularSeal2
- Lightning bolt
- Heart
- Picture frame
- Quad arrow
- Left arrow callout
- Right arrow callout
- Up arrow callout
- Down arrow callout
- Left right arrow callout
- Up down arrow callout
- Quad arrow callout
- Bevel
- Left bracket
- Right bracket
- Left brace
- Right brace
- Left up arrow
- Bent up arrow
- Bent arrow
- Seal24
- Striped right arrow
- Notched right arrow
- Block arc
- Smiley face
- Vertical scroll

- Horizontal scroll
- Circular arrow
- Notched circular arrow
- U-turn arrow
- Curved right arrow
- Curved left arrow
- Curved up arrow
- Curved down arrow
- Cloud callout
- Ellipse ribbon
- Ellipse ribbon 2
- Flow chart process
- Flow chart decision
- Flow chart input output
- Flow chart predefined process
- Flow chart internal storage
- Flow chart document
- Flow chart multidocument
- Flow chart terminator
- Flow chart preparation
- Flow chart manual input
- Flow chart manual operation
- Flow chart connector
- Flow chart punched card
- Flow chart punched tape
- Flow chart summing junction
- Flow chart or
- Flow chart collate
- Flow chart sort
- Flow chart extract
- Flow chart merge
- Flow chart offline storage
- Flow chart online storage
- Flow chart magnetic tape
- Flow chart magnetic disk
- Flow chart magnetic drum
- Flow chart display
- Flow chart delay
- Text plain text
- Text stop
- Text triangle
- Text triangle inverted
- Text chevron
- Text chevron inverted
- Text ring inside
- Text ring outside
- Text arch up curve
- Text arch down curve
- Text circle curve
- Text button curve
- Text arch up pour
- Text arch down pour
- Text circle pour
- Text button pour
- Text curve up

- Text curve down
- Text cascade up
- Text cascade down
- Text wave1
- Text wave2
- Text wave3
- Text wave4
- Text inflate
- Text deflate
- Text inflate bottom
- Text deflate bottom
- Text inflate top
- Text deflate top
- Text deflate inflate
- Text deflate inflate deflate
- Text fade right
- Text fade left
- Text fade up
- Text fade down
- Text slant up
- Text slant down
- Text can up
- Text can down
- Flow chart alternate process
- Flow chart off-page connector
- Callout 90
- Accent callout 90
- Border callout 90
- Accent border callout 90
- Left right up arrow
- Sun
- Moon
- Bracket pair
- Brace pair
- Seal4
- Double wave
- Host control
- Text box

By default : 'Text box'

$ShapeText – the text to be inserted into the shape.

$TextFmt – describe the formatting specifications for the text to be writen into the shape. It can be the name of a style in the style sheet of the document previously added with AddStyle or a string with formatting specifications as it is defined by StyleFormatting in AddStyle method

$ShapeFormat – a string defining the format of the shape. It has the following structure: {ShapeFillColor_ShapeLineDef_ShapeSize_ShapeScale_ShapeWeb}

ShapeFillColor - a string representing the fill color code of the shape. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a shape filled with the red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till

length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'.
ShapeLineDef - define the borders properties for the shape. It has the following structure: 'Color<123123123>Dashed<>Style<>Weight<>' or 'C<123123123>D<>S<>W<>' where

Color: is a RGB color code see ShapeFillColor for details about RGB color code.
Dash style Dashing: this is the dashed style of the line.

Allowed values (the values are self-explanatory):

- 'Solid line | solid | s'
- 'Dashed line windows | dashed windows | dash w'
- 'Dotted line windows | dotted windows | dot w'
- 'Dash-dotted line windows | dash dotted line windows | dash dotted windows | dash-dotted windows | d-d w | d d w | dd w | ddw'
- 'Dash-dot-dotted line windows | dash-dot-dotted windows | d-d-d w | ddd w | dddw'
- 'Dotted line | dotted | dot'
- 'Dashed line | dashed | dash'
- 'Long dashed line | long dashed | l d | ld'
- 'Dash-dotted line | dash dotted line | dash dotted | dash-dotted | d d |dd'
- 'Long dash-dotted line | long dash dotted line | long dash dotted | long dash-dotted | l d-d | l dd | ldd'
- 'Long dash-dot-dotted line | long dash-dot-dotted | l d-d-d | l ddd | lddd'

By default : 'Solid line'

Line style: this is the style of the line.

Allowed values (the values are self-explanatory):

- 'Single | s'
- 'Double of equal width | double equal | d e | de'
- 'Dotted line windows | dotted windows | dot w'
- 'Double one thick one thin | double one one | d o o | doo'
- 'Double reverse order | double reverse | d r | dr'
- 'Three'

By default : 'Single'

Weigth: the width of the line (border) [0 - 1584 pt]
Example:"Color<128128128>Dashed<Solid>Style<Single>Weight<2>"or "C<128128128>D<Solid>S<Single>W<2>", in this case for Dashed and Style the values by default will be used.

ShapeSize - define the size of the shape. It has this structure: Height99Width99 or H99W99, where 99 is a number between 0 - 22".
Example:"Height3.12Width6.76"
ShapeScale - define the scale of the shape. It has this structure: Height99Width99LR or H99W99LR, where 99 is a percent between 0-10675.
L -> if present then *Lock aspect ration* checkbox will be checked.
R -> if present then *Relative to original size* checkbox will be checked.
Example:"Height100Width1000LR"
ShapeWeb - the text to be displayed in internet environment. Web browsers display alternative text while pictures are loading or if they are missing. Web search engines use alternative text to help find Web pages.

$ShapePosition – a string defining the position of the shape. It has the following structure: {ShapeWrapStyle_ShapeWrapText_ShapeDistance_ShapeHoriz_ShapeVert_ShapeOptions}

ShapeWrapStyle - define the wrap style.

Allowed values (the values are self-explanatory):

- 'Wrap around shape | square | squ | q'
- 'Wrap tightly around shape | tight | tightly | ti'
- 'Wrap text through shape | through | th'
- 'Wrap around top and bottom of shape | top and bottom | tb'
- 'Wrap behind text | behind | b'
- 'In front of text | front | f'
- 'In line with text | line | l'

By default : 'Wrap around shape'

ShapeWrapText - define the wrap text mode on sides of shape.

Allowed values (the values are self-explanatory):

- 'Wrap both sides of shape | both sides | both | b'
- 'Wrap left side only | left side | left | l'
- 'Wrap right side only | right side | right | r'
- 'Wrap only on largest side | largest side | largest | la'

By default : 'Wrap both sides of shape'

ShapeDistance - define the distances from text. It has this structure: Left99Top99Right99Bottom99 or L99T99R99B99, where 99 is a number between 0"-22".

Example:"Left12Top13Right4Bottom5"

ShapeHoriz - define the horizontal alignment of the shape. It has the following structure: ALIGN:RELATIVETO.

Allowed values for ALIGN (the values are self-explanatory):

- Number between -22"-22" - that correspond to absolute position
- 'Left | l'
- 'Centered | center | c'
- 'Right | r'
- 'Inside | i'
- 'Outside | o'

By default : 'Left'

Allowed values for RELATIVETO (the values are self-explanatory):

- 'Margin | mar | m'
- 'Page | pg | p'
- 'Column | col | c'
- 'Character | char | ch'

By default : 'Column'

Example:"'Left:Column' or 'Centered:Page' or 'Inside:Margin' or '-12:Character'"

ShapeVert - define the vertical alignment of the shape. It has the following structure: ALIGN:BELLOWTO.

Allowed values for ALIGN (the values are self-explanatory):

- Number between -22"-22" - that correspond to absolute position
- 'Top | t'

- 'Centered | center | c'
- 'Bottom | b'
- 'Inside | i'
- 'Outside | o'

By default : 'Top'

Allowed values for BELLOWTO (the values are self-explanatory):

- 'Margin | mar | m'
- 'Page | pg | p'
- 'Paragraph | para | par'
- 'Line | lin | l'

By default : 'Paragraph'

Example:"'Top:Page' or 'Centered:Line' or 'Inside:Margin' or '-12:Paragraph'"

ShapeOptions - define the supplementary options for shape positioning. These options are *'Move object with text'* | *'Lock anchor'* | *'Allow overlap'*.The structure of this parameter is : (M?)(L?)(A?). If the letter is present the the corresponding option is set.

Example:"'MLA' or 'LA' or 'MA'"

$OtherProp – an array with properties to be applied to the shape. It allows to customize in detail the shape. The format of the array should be: array key = name of the property, array value = value of the property.

Example:array('fLockAspectRatio'=>'1','fLockRotation'=>'0','Rotation'=>65536*10)

Samples of paires properties/values that can be used (the exhaustive list can be found in the RTF 1.7 specification):

Property	**Meaning**	**Type of value**	**Default**
Lock			
fLockRotation	Boolean	Lock rotation.	FALSE
fLockAspectRatio	Boolean	Lock aspect ratio.	FALSE
fLockAgainstSelect	Boolean	Lock against selection.	FALSE
fLockCropping	Boolean	Lock against cropping.	FALSE
fLockVerticies	Boolean	Lock against edit mode.	FALSE
fLockText	Boolean	Lock text against editing.	FALSE
fLockAdjustHandles	Boolean	Lock adjust handles.	FALSE
fLockAgainstGrouping	Boolean	Lock against grouping.	FALSE
fLockShapeType	Boolean	Lock the shape type (don’t allow Change Shape).	FALSE
Text Box			
dxTextLeft	EMU	Left internal margin of the text box.	91,440

dyTextTop	EMU	Top internal margin of the text box.	45,720
dxTextRight	EMU	Right internal margin of the text box.	91,440
dyTextBottom	EMU	Bottom internal margin of the text box.	45,720
WrapText	Not applicable	Wrap text at shape margins: 0 Square 1 Tight 2 None 3 Top bottom 4 Through	0
anchorText	Not applicable	Text anchor point: 0 Top 1 Middle 2 Bottom 3 Top centered 4 Middle centered 5 Bottom centered 6 Bottom centered baseline	0
txflTextFlow	Not applicable	Text flow: 0 Horizontal non-ASCII font 1 Top to bottom ASCII font 2 Bottom to top non-ASCII font 3 Top to bottom non-ASCII font 4 Horizontal ASCII font	0

cdirFont	Direction	Font rotation: 0 Right 1 Down 2 Left 3 Up	0
fAutoTextMargin	Boolean	Use host's margin calculations.	FALSE
scaleText	Long integer	Text zoom and scale.	0
lTxid	Long integer	ID for the text. The value is determined by the host.	0
fRotateText	Boolean	Rotate text with shape.	FALSE
fSelectText	Boolean	TRUE if single click selects text, FALSE if two clicks select text.	TRUE
fFitShapeToText	Boolean	Adjust shape to fit text size.	FALSE
fFitTextToShape	Boolean	Adjust text to fit shape size.	FALSE
WordArt Effect			
gtextUNICODE	String	Unicode text string.	NULL
gtextAlign	Not applicable	Alignment on curve: 0 Stretch each line of text to fit width 1 Center text on width 2 Left justify 3 Right justify 4 Spread letters out to fit width 5 Spread words out to fit width	1
gtextSize	Fixed	Default point size.	2,359,296
gtextSpacing	Fixed	Adjust the spacing between characters (1.0 is normal).	65,536
gtextFont	String	Font name.	NULL
fGtext	Boolean	**True** if the text effect properties (**gtext***) are used. **False** if these properties are ignored.	FALSE

gtextFVertical	Boolean	If available, an @ font should be used. Otherwise, rotate individual characters 90 degrees counter-clockwise.	FALSE
gtextFKern	Boolean	Use character pair kerning if it is supported by the font.	FALSE
gtextFTight	Boolean	Adjust the spacing between characters rather than the character advance by the **gtextSpacingratio**.	FALSE
gtextFStretch	Boolean	Stretch the text to fit the shape.	FALSE
gtextFShrinkFit	Boolean	When laying out the characters, consider the glyph bounding box rather than the nominal font character bounds.	FALSE
gtextFBestFit	Boolean	Scale text laid out on a path to fit the path.	FALSE
gtextFNormalize	Boolean	Stretch individual character heights independently to fit.	FALSE
gtextFDxMeasure	Boolean	When laying out characters, measure the distances along the x-axis rather than along the path.	FALSE
gtextFBold	Boolean	Bold font (if available).	FALSE
gtextFItalic	Boolean	Italic font (if available).	FALSE
gtextFUnderline	Boolean	Underline font (if available).	FALSE
gtextFShadow	Boolean	Shadow font (if available).	FALSE
gtextFSmallcaps	Boolean	Small caps font (if available).	FALSE
gtextFStrikethrough	Boolean	Strikethrough font (if available).	FALSE
fGtextOK	Boolean	Text effect (WordArt) supported.	FALSE
gtextFReverseRows	Boolean	Reverse row order.	FALSE
gtextRTF	String	RTF text string.	NULL
3-D Effects			
c3DSpecularAmt	Fixed	Specular amount for the material.	0
c3DDiffuseAmt	Fixed	Diffusion amount for the material.	65,536
c3DShininess	Long integer	Shininess of the material.	5
c3DEdgeThickness	EMU	Specular edge thickness.	12,700
c3DExtrudeForward	EMU	Extrusion amount forward.	0

c3DExtrudeBackward	EMU	Extrusion amount backward.	457,200
c3DExtrusionColor	Color	Color of the extrusion.	
f3D	Boolean	**True** if shape has a three-dimensional (3D) effect, **False** if it does not.	FALSE
fc3DMetallic	Boolean	**True** if shape uses metallic specularity, **False** if it does not.	FALSE
fc3DUseExtrusionColor	Boolean	Extrusion color is set explicitly.	FALSE
fc3DLightFace	Boolean	Light the face of the shape.	TRUE
c3DYRotationAngle	Angle	Degrees about y-axis. If **fc3DconstrainRotation** (a Boolean property which defaults to **True**) is **True**, then the rotation is restricted to x-y rotation. In addition, the final rotation results from first rotating by **c3DYRotationAngle** degrees about the y-axis and then by **c3DXRotationAngle** degrees about the z-axis. If **fc3DconstrainRotation** is **False**, then the final rotation results from a single rotation of **c3DrotationAngle** about the axis specified by **c3DrotationAxisX**, **c3DrotationAxisY**, and **c3DrotationAxisZ**.	0
c3DXRotationAngle	Angle	Degrees about x-axis.	0
c3DRotationAxisX	Long integer	These keywords specify the rotation axis. Only their relative magnitudes matter.	100
c3DRotationAxisY	Long integer	See meaning for **c3DYRotationAxisX**.	0
c3DRotationAxisZ	Long integer	See meaning for **c3DYRotationAxisX**.	0
c3DRotationAngle	Angle	The rotation about the axis (defined previously in the **c3DRotationAxisX**, **Y**, and **Z** parameter sections)	0
fC3DRotationCenterAuto	Boolean	If **fC3DRotationCenterAuto** is **True**, then the rotation will be about the center of the 3-D bounding cube of the 3-D group; otherwise, the rotation center will be about **c3DRotationCenterX**, **c3DRotationCenterY**, and **c3DRotationCenterZ**.	FALSE

c3DRotationCenterX	Fixed	Rotation center (X). The **X** and **Y** values are a 16.16 fraction of the geometry width and height, with (0,0) being at the center of the geometry. The **Z** value must be in absolute units (EMUs).	0
c3DRotationCenterY	Fixed	Rotation center (Y). If **fC3DRotationCenterAuto** is **True**, then the rotation will be about the center of the 3-D bounding cube of the 3-D group; otherwise, the rotation center will be about **c3DRotationCenterX**, **c3DRotationCenterY**, and **c3DRotationCenterZ**. The **X** values and **Y** values are a fraction of the geometry width and height, with (0,0) being at the center of the geometry. The **Z** value is in absolute units.	0
c3DRotationCenterZ	EMU	See meaning for **c3DRotationCenterY**.	0
c3DRenderMode	Long integer	0 Render with full detail 1 Render as a wire frame 2 Render a bounding cube	Not applicable
c3DXViewpoint	EMU	X view point.	1,250,000
c3DYViewpoint	EMU	Y view point.	-1,250,000
c3DZViewpoint	EMU	Z view distance.	9,000,000
c3DOriginX	Fixed	The following **c3DOriginY** and **c3DSkewAngle** values define the origin relative to the viewpoint origin measured. These values are 16.16 numbers that specify the position of the origin within the shape bounding box, as multiples of the width and height of that bounding box and relative to the center (that is, they are displaced from the center). When these values are applied the actual transformed shape path is used, rather than the shape geometry (compare with the shadow and perspective values that work on the geometry bounding box, not the actual points). This means that a shape that extends outside the geometry bounding box (such as a text effect) is handled "correctly" for the calculation of the 3-D origin.	32,768
c3DOriginY	Fixed	See meaning for **c3DOriginX**.	-32,768

c3DSkewAngle	Fixed	Skew angle.	-8,847,360
c3DSkewAmount	Long integer	Percentage skew amount.	50
c3DAmbientIntensity	Fixed	Ambient intensity should be low (0 to .1) to avoid washed out appearance.	20,000
c3DKeyX	Long integer	Key light source direction. Values may be any number; only their relative magnitudes matter.	50,000
c3DKeyY	Long integer	See meaning for **c3DKeyX**.	0
c3DKeyZ	Long integer	See meaning for **c3DKeyX**.	10,000
c3DKeyIntensity	Fixed	Fixed point intensity. Theoretical maximum is 1, but may be higher.	38,000
c3DFillX	Long integer	Fill light source direction; only their relative magnitudes matter. This direction defines a second light source arbitrarily called the "fill light." Generally this will be positioned 90-180 degrees away from the key light and very roughly in front of the scene to fill in any harsh shadows. This fill will be dim compared to the first light source. Theoretically it should be non-harsh, but harsh fill lighting looks better sometimes.	-50,000
c3DFillY	Long integer	See meaning for **c3DfillX**.	0
c3DFillZ	Long integer	See meaning for **c3DfillX**.	10,000
c3DFillIntensity	Fixed	Theoretical maximum is 1, but may be higher.	38,000
fc3DParallel	Boolean	**True** if the fill has parallel projection, **False** if it does not. If **fc3DParallel** is **True**, the **fc3DKeyHarsh** and **fc3DFillHarsh** properties determine the parallel projection used. A skew amount of 0 means the projection is orthographic.	TRUE
fc3DKeyHarsh	Boolean	**True** if key lighting is harsh, **False** if it is not.	TRUE
fc3DFillHarsh	Boolean	**True** if fill lighting harsh, **False** if it is not.	FALSE
c3DCrMod	Color	Modification for BW views.	Undefined
c3DTolerance	Fixed	3D tolerance.	30,000
Perspective			

perspectiveOffsetX	Fixed	The values define a transformation matrix. Each value is scaled by the perspectiveWeight parameter.	0
perspectiveOffsetY	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveOriginX	Fixed	Perspective x origin.	32,768
perspectiveOriginY	Fixed	Perspective y origin.	32,768
perspectivePerspectiveX	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectivePerspectiveY	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleXToX	Fixed	See meaning for **perspectiveOffsetX**.	65,536
perspectiveScaleXToY	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleYToX	Fixed	See meaning for **perspectiveOffsetX**.	0
perspectiveScaleYToY	Fixed	See meaning for **perspectiveOffsetX**.	65,536
perspectiveType	Transform type	Where transform applies: 0 Absolute 1 Shape 2 Drawing	1
perspectiveWeight	Fixed	Scaling factor.	256
fPerspective	Boolean	On/off.	Not applicable

Black and White Modes

bWMode	Black and white mode	Settings for modifications to be made when in different forms of black and white mode: 0 Color 1 Automatic 2 Grayscale 3 Light grayscale 4 Inverse gray 5 Gray outline 6 Black TextLine 7 High contrast 8 Black 9 White 10 Don’t show 11 Number of black and white modes	1
bWModeBW	Black and white mode	See meaning for **bWMode**.	1
bWModePureBW	Black and White Mode	See meaning for **bWmode**.	1

The format of the value depends on the property name it is paired with. Many values are simple single numbers. Distances are expressed in EMU units. There are 12,700 EMU units in a point hence 914,400 in an inch and 360,000 cm-1. Fractional or fixed values are expressed using units that are 1/65536th of a whole. Angles are expressed as fractions of a degree. Colors are 24-bit color values. Booleans have two possible values: 1 for **True** and 0 for **False**.

InsertTable

Signature

function InsertTable($TblCnt, $TableFmt = '', $CellFmt = '', $Header = False, $CellWidths='', $RowWidth='')

Description

This method insert a table at the current position in the document.

Parameters

$TblCnt - this is the content of the table. It is an array of arrays with the following structure:

```
$TblCnt=array(
0=>array("Col_11","Col_12",...,"Col_1n"),              -> this is the Row 1 of the table
1=>array("Col_21","Col_22",...,"Col_2m),               -> this is the Row 2 of the table
2=>array("Col_31","Col_32",...,"Col_3s")               -> this is the Row 3 of the table
...
R=>array("Col_(r+1)1","Col_(r+1)2",...,"Col_(r+1)t")); -> this is the Row R of the table
```

$TableFmt - a string or an array of strings (you can have as many elements in array as many rows in the table or less and in that case the last rows will have the same format) defining the format of the table. The string has the following structure:
{TblRowFormat_TblRowBorder_TblRowShading_TblAutoFormat}

TblRowFormat - define the row format for the table. It has the following structure:
'WriteDirection%AutoFit%Align%CellMargins%CellSpacing'

WriteDirection - LR or RL <=> Left to Right or Right to Left
AutoFit - CNT | WIN <=> Auto fit to contents | Auto fit to window
Align - L+Nr | C | R <=> Left + indent in twips | Center | Right
CellMargins - Left99Top99Righ99Bottom99 , where 99 is the distance in twips between the margin and the text inside the table
CellSpacing - Left99 Top99 Righ99 Bottom99 , where 99 is the distance in twips between the two cells (horizontal or vertical)

Example of TblRowFormat :"LR%WIN%C%L11T12R13B14%L21T22R23B24%"

TblRowBorder - define the border styles for the table. It has the following structure:
'DefinitionOfBorder%Width%SpaceBetwenBorder%Color%BordersMargins'

DefinitionOfBorder - a string specifying the border style.

Allowed values:

'Single-thickness | Single thickness | Single'	Single-thickness border.
'Double-thickness | Double thickness'	Double-thickness border.
'Shadowed'	Shadowed border.
'Double'	Double border.
'Dotted'	Dotted border.

- 'Dotted' Dotted border.
- 'Dashed' Dashed border.
- 'Hairline' Hairline border.
- 'Resembles a frame | Frame' Border resembles a “Frame.”
- 'No border specified | No' No border specified.
- 'Table cell has no borders | table cell has no' Table cell has no borders.
- 'Inset' Inset border.
- 'Dashed small' Dashed small.
- 'Dot-dashed | Dot dashed' Dot-dashed border.
- 'Dot-dot-dashed | Dot-dot dashed | Dot dot-dashed | Dot dot dashed' Dot-dot-dashed border.
- 'Outset' Outset border.
- 'Triple' Triple border.
- 'Thick-thin small | Thick thin small' Thick-thin border (small).
- 'Thin-thick small | Thin thick small' Thin-thick border (small).
- 'Thin-thick thin small | Thin thick thin small' Thin-thick thin border (small).
- 'Thick-thin medium | Thick thin medium' Thick-thin border (medium).
- 'Thin-thick medium | Thin thick medium' Thin-thick border (medium).
- 'Thin-thick thin medium | Thin thick thin medium' Thin-thick thin border (medium).
- 'Thick-thin large | Thick thin large' Thick-thin border (large).
- 'Thin-thick large | Thin thick large' Thin-thick border (large).
- 'Thin-thick thin large | Thin-thick thin large | Thin thick-thin large | Thin thick thin large' Thin-thick-thin border (large).
- 'Wavy' Wavy border.
- 'Double wavy' Double wavy border.
- 'Striped' Striped border.
- 'Embossed' Embossed border.
- 'Engraved' Engraved border.

By default: 'No border specified'

Width - this argument is representing the border width in twips. It is the width in twips of the pen used to draw the paragraph border line and cannot be greater than 75. To obtain a larger border width, please prefix your width with the letter 'D' and you will obtain a width double that the one you had specified. Ex: 'D60' will draw a border with 120 twips in width. If null or missing by default the width will 1.

SpaceBetwenBorder - this argument specify the space in twips between borders and the paragraph.

Color - a string representing the RGB color code of the border to be drawn. The color will be specified in this format 'RRRGGGBBB' where RRR is the code for the red color and can be any number between 000 and 255, GGG is the code for the green color and can be any number between 000 and 255, BBB is the code for the blue color and can be any number between 000 and 255. Ex: '255000000' to draw a border with red color. If the length of the argument is less than 9 then it will be filled out with '0' at the right till length is 9. Ex: '00012' became '000120000'. If null or missing by default the color will be '000000000'.

BordersMargins - a string representing the border or the borders the format will be applied for. IT has this format: (T?)(L?)(B?)(R?)(H?)(V?), where T ->TOP, L ->LEFT, B->BOTTOM, R ->RIGHT, H ->DIAGONAL TL-BR, V-> DIAGONAL TR-BL

TblRowShading - define the shading for the table. The structure of the field is:
ShadingStyle%ShadingForeColor%ShadingBackColor

ShadingStyle - define the shading style.

Allowed values:

- 0 - 100, specifying a percentage
- Dk Horizontal

- Dk Vertical
- Dk Dwn Diagonal
- Dk Up Diagonal
- Dk Grid
- Dk Trellis
- Lt Horizontal
- Lt Vertical
- Lt Dwn Diagonal
- Lt Up Diagonal
- Lt Grid
- Lt Trellis

By default : '0'

ShadingForeColor - a RGB color code
ShadingBackColor - a RGBcolor code

TblAutoFormat - set the auto formatting flags for the table. This flags are Bordres+Shading+Font+Color+BestFit+HeadeRow+LastRow+FirstColumn+LastColumn. The structure of the field is:
(B?)(S?)(F?)(C?)(E?)(H?)(L?)(I?)(A?), the presence of letter means that the flag is set:

B - Flag sets table auto format to format borders.
S - Flag sets table auto format to affect shading.
F - Flag sets table auto format to affect font.
C - Flag sets table auto format to affect color.
E - Flag sets table auto format to apply best fit.
H - Flag sets table auto format to format the first (header) row.
L - Flag sets table auto format to format the last row.
I - Flag sets table auto format to format the first (header) column.
A - Flag sets table auto format to format the last column.

$CellFmt - describe the formatting specifications for the text to be written in the cells of the table. It can be the name of a style in the style sheet of the document previously added with AddStyle or a string with formatting specifications as it is defined by StyleFormatting in AddStyle method. Also it can be a matrix - a row in the matrix <=> row in the table, a column in the matrix <=> a column in the table. If the table has more rows than the matrix you defined then the last rows will have the same format; if a row in the table has more columns than the row in the matrix then the last column will have the same format !

$Header - if TRUE then the first row of the table will be a header row, that means it will be emitted on each page the table appear.

$CellWidths - an array of arrays of integers specifying the width (in percentage) of each cell of the table. In the case of missing values (rows of cells or only cells) the values from the precedent row of cell width will be used. For accuracy the first row of cells must have all width values specified.

Example for a table with at least 3 rows and 4 columns:

```
$CellWidth=array(
0=>array(15,5,30,50),
0,
2=>array(10,10,30,50));
```

$RowWidth - an integer specifying the width of the table in percentage.

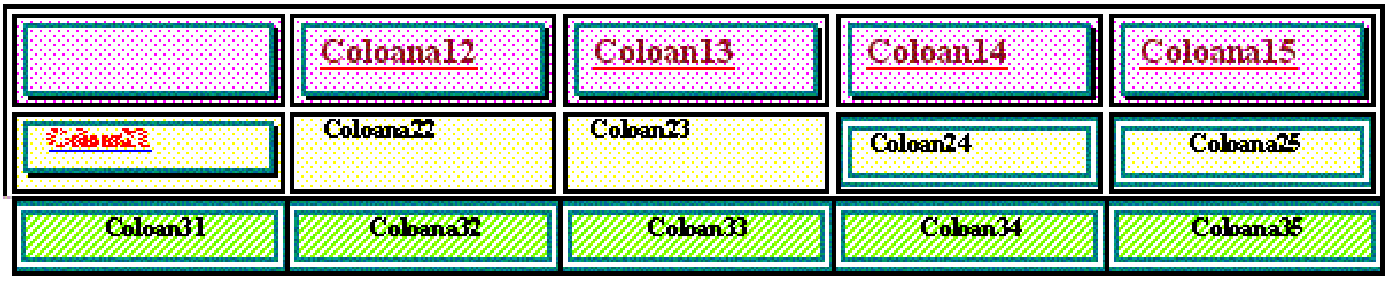
EXAMPLE: for the arguments bellow:

```
$TblContent=array(
0=>array("","Coloana12","Coloan13","Coloan14","Coloana15"),
1=>array("Coloan21","Coloana22","Coloan23","Coloan24","Coloana25"),
2=>array("Coloan31","Coloana32","Coloan33","Coloan34","Coloana35"));

$DefaultTableFormat2
="RL%WIN%L-23%L108T0R108B0%L18T18R18B18%_single%32%2%000000000%TLBR_20%25
5000255%255255255_BSFCEHLIA";
$DefaultTableFormat1
="RL%WIN%L-23%L108T0R108B0%L18T18R18B18%_single%32%2%000000000%TLBR_20%25
5255000%255255255_BSFCEHLIA";
$DefaultTableFormat
="RL%WIN%L0%L108T0R108B0%L18T0RB0%_single%32%2%000000000%TLBR_dk Up
Diagonal%128255000%255255255_BSFCEHLIA";
$DefaultCellFormat="PARFMT:[{left_body
text_Left0Right0_^_Before0After0_Single_W_^}]BRDRDEF:[{shadowed_000128128_32_3_tblr}]FONT
DEF:[{^_Bold_18_140020029_word255000000_superscript_^_^_^_^_^}]";
$DefaultCellFormat1="PARFMT:[{left_body
text_Left0Right0_^_Before0After0_Single_W_^}]FONTDEF:[{Times New
Roman_Bold_12_255_word000000255_^_^_^_^_^_Shimmer}]";
$StyleFormatEndNoteRef="FONTDEF:[{^_^_^_^_^_superscript_^_^_^_^_^}]";
$StyleFormatHeaderText="PARFMT:[{centered_body
text_Left4320Right0_H3_Before0After0_Single_W_^}]";

$TabelFormat=array($DefaultTableFormat2,$DefaultTableFormat1,$DefaultTableFormat);
$CellFormat=array(
0=>array($DefaultCellFormat,$DefaultCellFormat,$DefaultCellFormat,$DefaultCellFormat,$DefaultCell
Format),
1=>array($DefaultCellFormat1,"BRDRDEF:[{none_000128128_32_3_tblr}]FONTDEF:[{^_^_^_^_^_sup
erscript_^_^_^_^_Shimmer}]",$StyleFormatEndNoteRef,"BRDRDEF:[{double_000128128_32_3_tblr}]F
ONTDEF:[{^_^_^_^_^_superscript_^_^_^_^_Las Vegas}]",$StyleFormatHeaderText));
```

with this call:

```
$rtf->InsertTable($TblContent,$TableFormat,$CellTextFormat)
```

then you'll get this table :



	Coloana12	Coloan13	Coloan14	Coloana15
[illegible]	Coloana22	Coloan23	Coloan24	Coloana25
Coloan31	Coloana32	Coloan33	Coloan34	Coloana35

InsertUnicodeChar

Signature

function InsertUnicodeChar ($UniChar, $ANSIEquiv)

Description

Insert an Unicode character.

Parameters

$UniChar – **the Unicode charcater to be inserted.**

$ANSIEquiv – **the ANSI equivalent.**

InsertUnicodeCorresp

Signature

function InsertUnicodeCorresp ($ByteNr = '')

Description

Insert in document the keyword specifying the number of bytes corresponding to a given Unicode character. The calls of this method are scoped like character properties. That is, a method call will applies only to text following the call.

Parameters

$ByteNr – the number of bytes for a given Unicode char, if NULL then the value specified by SetUnicodeCorresp will be used.

LastWordSupported

Signature

function LastWordSupported ()

Description

This method return a string with the version of the Microsoft Word who fully support all the keywords used in the current RTF document.

Parameters

LineBreak

Signature

function LineBreak($Type = '')

Description

This method will insert a line break at the current position in the RTF document. If the argument is not null the a no required line break will be emitted.

Parameters

$Type – any string if you want to insert a non required line break, null for a hard line break.

Open

Signature

function Open()

Description

This method begin the generation of the RTF file. It has to be called before any other method.

Parameters

Output

Signature

function Output ($File = '', $download = *FALSE*)

Description

Send the RTF document to the browser or save it locally when the $File is present. If the $File is not null then the file can be saved locally or the file download dialog box can be displayed. This method begin by calling the Close() method if it is necessary.

Parameters

$File - the name of the file. If it is null the document will be sent to the browser who will use the plug-in if installed.

$download - if FALSE the file will be saved locally according to the $File, if TRUE then the file download dialog box will be bring up.

PageBreak

Signature

function PageBreak($Type = '')

Description

This method will insert a page break at the current position in the RTF document. If the argument is not null the a no required page break will be emitted.

Parameters

$Type – any string if you want to insert a non required page break, null for a hard page break.

Tab

Signature

function Tab()

Description

This method will insert a tab character at the current position in the RTF document.

Parameters

Write

Signature

function Write($Txt, $Fmt = '', $NewPar = *'Yes'*)

Description

Write a text in the RTF document at the current position of the cursor. This text will be formatted according to the $Fmt specifications. The text can be write continuously i the current paragraph or a new paragraph can be emitted.

Parameters

$Txt - the text to be written in the document

$Fmt - describe the formatting specifications for the text to be write. It can be the name of a style in the style sheet of the document previously added with AddStyle **or a string with formatting specifications as it is defined by StyleFormatting in** AddStyle **method.**

$NewPar - a flag specifying if the text will be written in a new paragraph.

Allowed values:

- **'Yes' - that means a new paragraph will be emitted in order to write the text**
- **'No' - that means the text will be written continuously in the current paragraph**

By default : 'Yes'

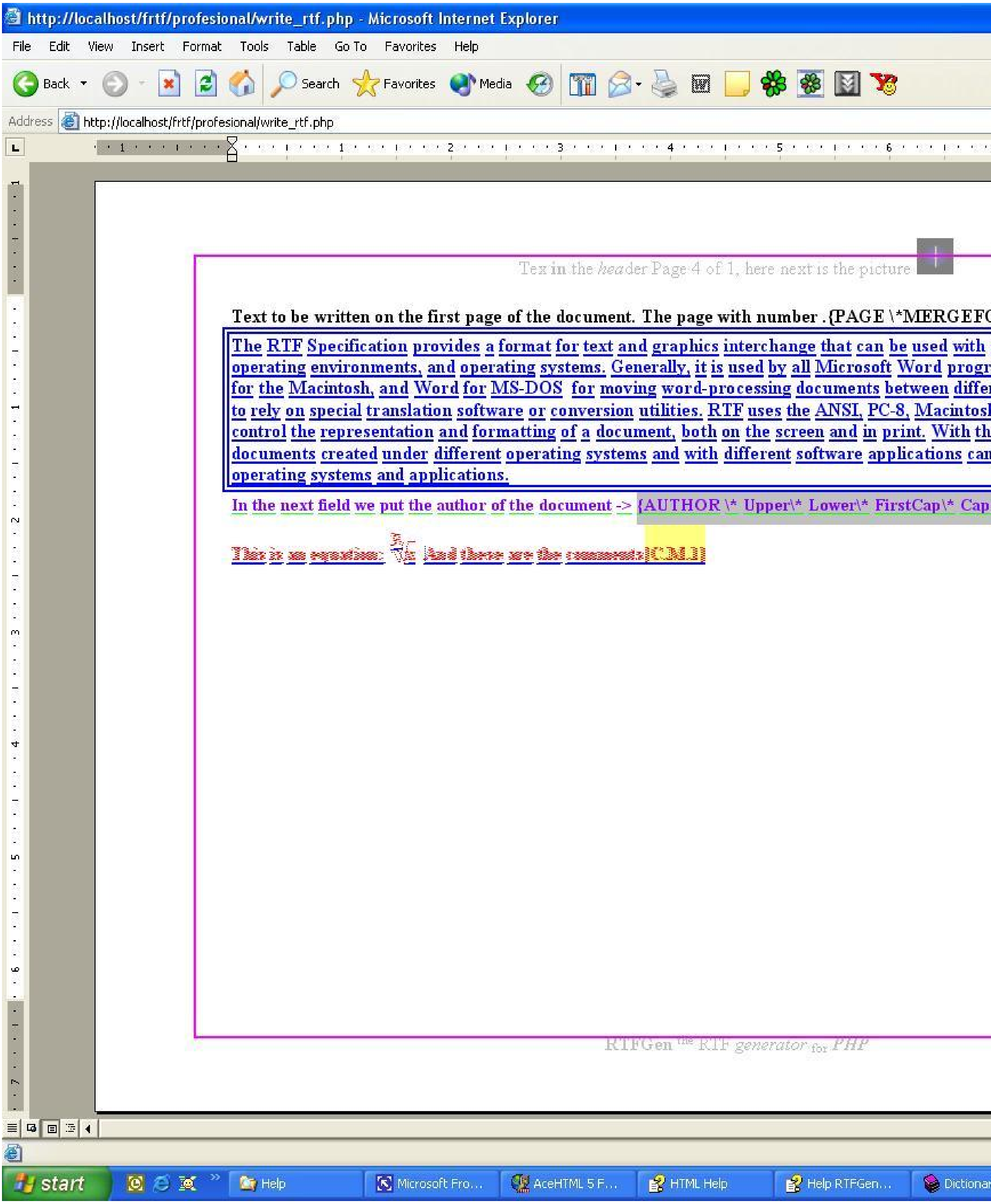
Samples for RTF Generator – Professional Edition

Sample 1

In order to create your first RTF document all you have to do is to write the following lines in your PHP file:

```
<?php
require('frtf.php');

$rtf=new FRTF();

$rtf->Open();

//some info document settings
$rtf->SetTitle("File generated by RTFGen V1.0 - Professional Edition");
$rtf->SetSubject("Subjectul Documentului");
$rtf->SetManager("Managerul Documentului");
$rtf->SetAuthor("Authorul Documentului");
$rtf->SetOperator("Operatorul Documentului");
$rtf->SetCompany("Compania Documentului");
$rtf->SetHlinkbase("Your BaseLink here");
$rtf->SetCreationTime();
$rtf->SetRevisionTime('12/23/2004 20:45:53');
$rtf->SetPrintTime('12/23/2004 20:45:53');
$rtf->SetBackupTime('12/23/2004 20:45:53');
$rtf->SetEditionTime('12');
$rtf->SetVersionDoc('3');
//some document formatting settings
$rtf->SetDefaultTab(0);
$rtf->SetHyphenHot('');
$rtf->SetHyphenConsec('');
$rtf->SetHyphenCaps('0');
$rtf->SetHyphenAuto('');
$rtf->SetLineStart(5);
$rtf->SetFracWidth();
$rtf->SetNextFile("NextFile here");
$rtf->SetTemplate("Template.doc");
$rtf->SetDefLang("Romanian");
$rtf->SetDefLangfe();

$rtf->SetWindowCaption('RTFGen for PHP');
$rtf->SetDocType('General Document');
$rtf->SetFromHtml();
$rtf->SetFromText();
$rtf->SetDocumentView('Page',100,'ful');
$rtf->SetPaperSize('a4');
//$rtf->SetMargins(10000,10000,2500,2500);
//$rtf->SetFacingp();
//$rtf->SetMargmirror();
//$rtf->SetTwoonone();
//$rtf->SetGutter(1000);
//$rtf->SetGutterPos('t');
//$rtf->SetRender();
//$rtf->SetDocSource();
$rtf->SetPageOrientation('l');// set the page orientation to landscape
$rtf->SetWidowCtrl();
$rtf->SetPgnStart(4); //the page number will start with 4
//$rtf->SetBookfold();
//$rtf->SetBookfoldrev();
//$rtf->SetBookfoldsheets();
//$rtf->SetPageBorderHead();
//$rtf->SetPageBorderFoot();
```

and this is what you will obtain:


http://localhost/frtf/profesional/write_rtf.php - Microsoft Internet Explorer
File Edit View Insert Format Tools Table Go To Favorites Help
Back
Search
Favorites
Media
Address http://localhost/frtf/profesional/write_rtf.php
start
Help
Microsoft Fro...
AceHTML 5 F...
HTML Help
Help RTFGen...
Dictionar

Tex in the *hea*der Page 4 of 1, here next is the picture

Text to be written on the first page of the document. The page with number .{PAGE *MERGEFO

The RTF Specification provides a format for text and graphics interchange that can be used with operating environments, and operating systems. Generally, it is used by all Microsoft Word progr for the Macintosh, and Word for MS-DOS for moving word-processing documents between differ to rely on special translation software or conversion utilities. RTF uses the ANSI, PC-8, Macintosh control the representation and formatting of a document, both on the screen and in print. With th documents created under different operating systems and with different software applications can operating systems and applications.

In the next field we put the author of the document -> {AUTHOR * Upper* Lower* FirstCap* Cap

~~**This is an equation:**~~ ~~**And there are the comments[CM1]**~~

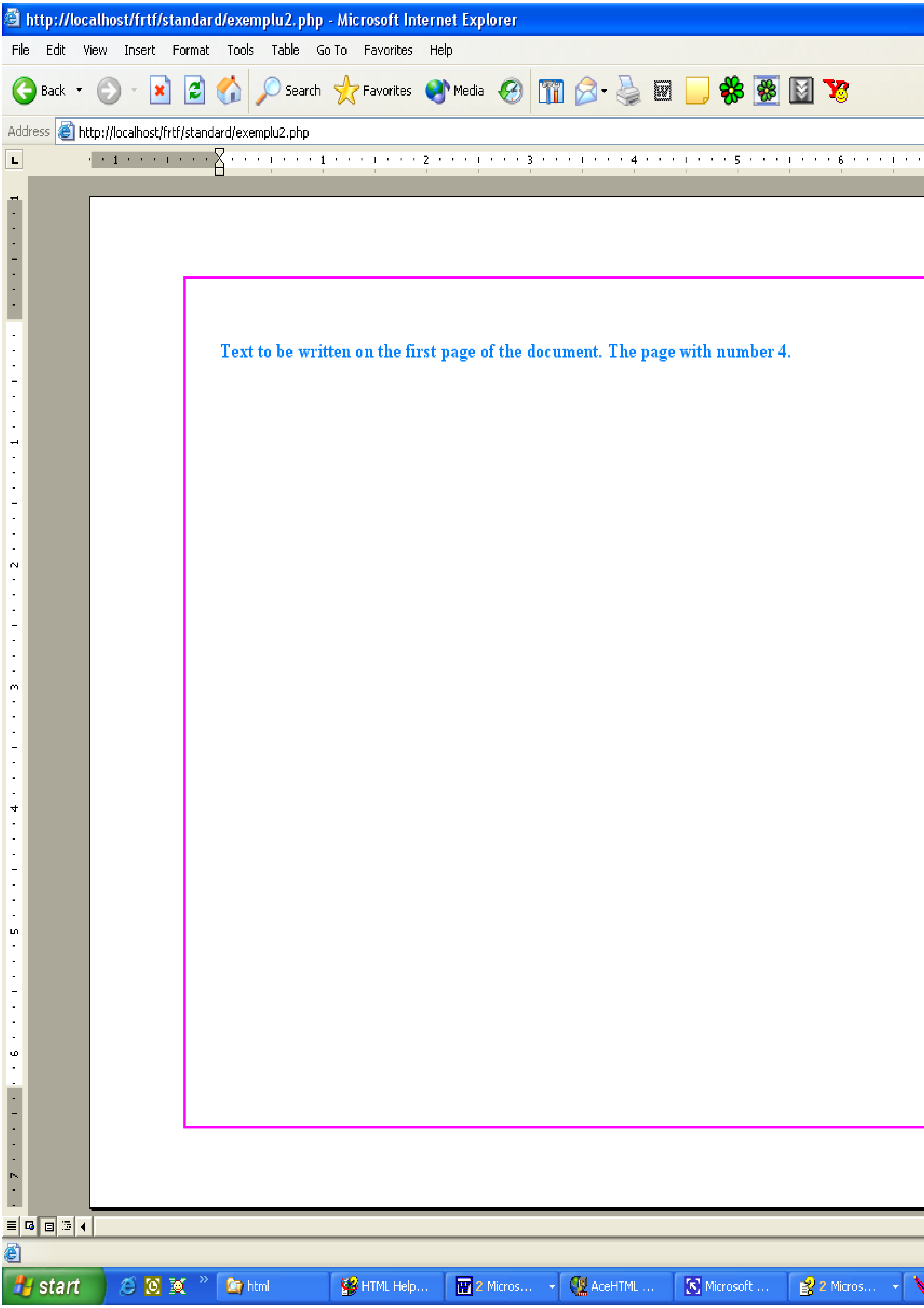
RTFGen the RTF generator for PHP

Sample 2

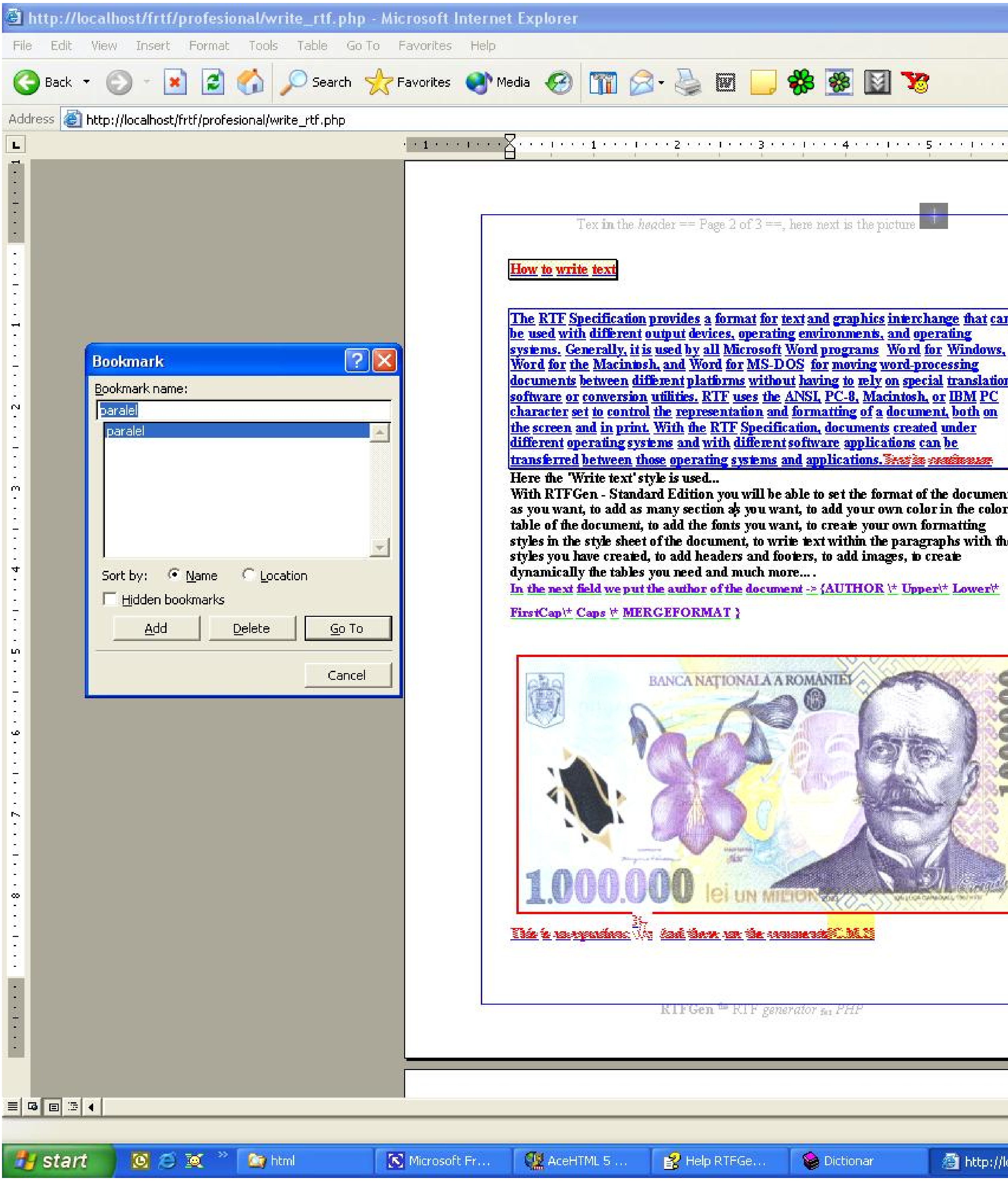
This will show how to add more sections to your document and how to add headers/footers with page number:

```
<?php
require('frtf.php');

$rtf=new FRTF();

$rtf->Open();

//some info document settings
$rtf->SetTitle("File generated by RTFGen V1.0 - Professional Edition");
$rtf->SetSubject("Subjectul Documentului");
$rtf->SetManager("Managerul Documentului");
$rtf->SetAuthor("Authorul Documentului");
$rtf->SetOperator("Operatorul Documentului");
$rtf->SetCompany("Compania Documentului");
$rtf->SetHlinkbase("Your BaseLink here");
$rtf->SetCreationTime();
$rtf->SetRevisionTime('12/23/2004 20:45:53');
$rtf->SetPrintTime('12/23/2004 20:45:53');
$rtf->SetBackupTime('12/23/2004 20:45:53');
$rtf->SetEditionTime('12');
$rtf->SetVersionDoc('3');
//some document formatting settings
$rtf->SetDefaultTab(0);
$rtf->SetHyphenHot('');
$rtf->SetHyphenConsec('');
$rtf->SetHyphenCaps('0');
$rtf->SetHyphenAuto('');
$rtf->SetLineStart(5);
$rtf->SetFracWidth();
$rtf->SetNextFile("NextFile here");
$rtf->SetTemplate("Template.doc");
$rtf->SetDefLang("Romanian");
$rtf->SetDefLangfe();

$rtf->SetWindowCaption('RTFGen for PHP');
$rtf->SetDocType('General Document');
$rtf->SetFromHtml();
$rtf->SetFromText();
$rtf->SetDocumentView('Page',100,'ful');
$rtf->SetPaperSize('a4');
//$rtf->SetMargins(10000,10000,2500,2500);
//$rtf->SetFacingp();
//$rtf->SetMargmirror();
//$rtf->SetTwoonone();
//$rtf->SetGutter(1000);
//$rtf->SetGutterPos('t');
//$rtf->SetRender();
//$rtf->SetDocSource();
$rtf->SetPageOrientation('l');// set the page orientation to landscape
$rtf->SetWidowCtrl();
//$rtf->SetPgnStart(4); //the page number will start with 4
//$rtf->SetBookfold();
//$rtf->SetBookfoldrev();
//$rtf->SetBookfoldsheets();
//$rtf->SetPageBorderHead();
//$rtf->SetPageBorderFoot();
```

and you'll obtain the followings two pages:



Text to be written on the first page of the document. The page with number 4.


http://localhost/frtf/profesional/write_rtf.php - Microsoft Internet Explorer
File Edit View Insert Format Tools Table Go To Favorites Help
Back
Search
Favorites
Media
Address
http://localhost/frtf/profesional/write_rtf.php
Bookmark
Add
Delete
Go To
Cancel
start
html
Microsoft Fr...
AceHTML 5 ...
Help RTFGe...
Dictionar
http://lo

Bookmark name: paralel

paralel

Sort by: ● Name ○ Location

□ Hidden bookmarks

Tex in the *header* == Page 2 of 3 ==, here next is the picture

How to write text

The RTF Specification provides a format for text and graphics interchange that can be used with different output devices, operating environments, and operating systems. Generally, it is used by all Microsoft Word programs Word for Windows, Word for the Macintosh, and Word for MS-DOS for moving word-processing documents between different platforms without having to rely on special translation software or conversion utilities. RTF uses the ANSI, PC-8, Macintosh, or IBM PC character set to control the representation and formatting of a document, both on the screen and in print. With the RTF Specification, documents created under different operating systems and with different software applications can be transferred between those operating systems and applications.~~Text is continuous~~

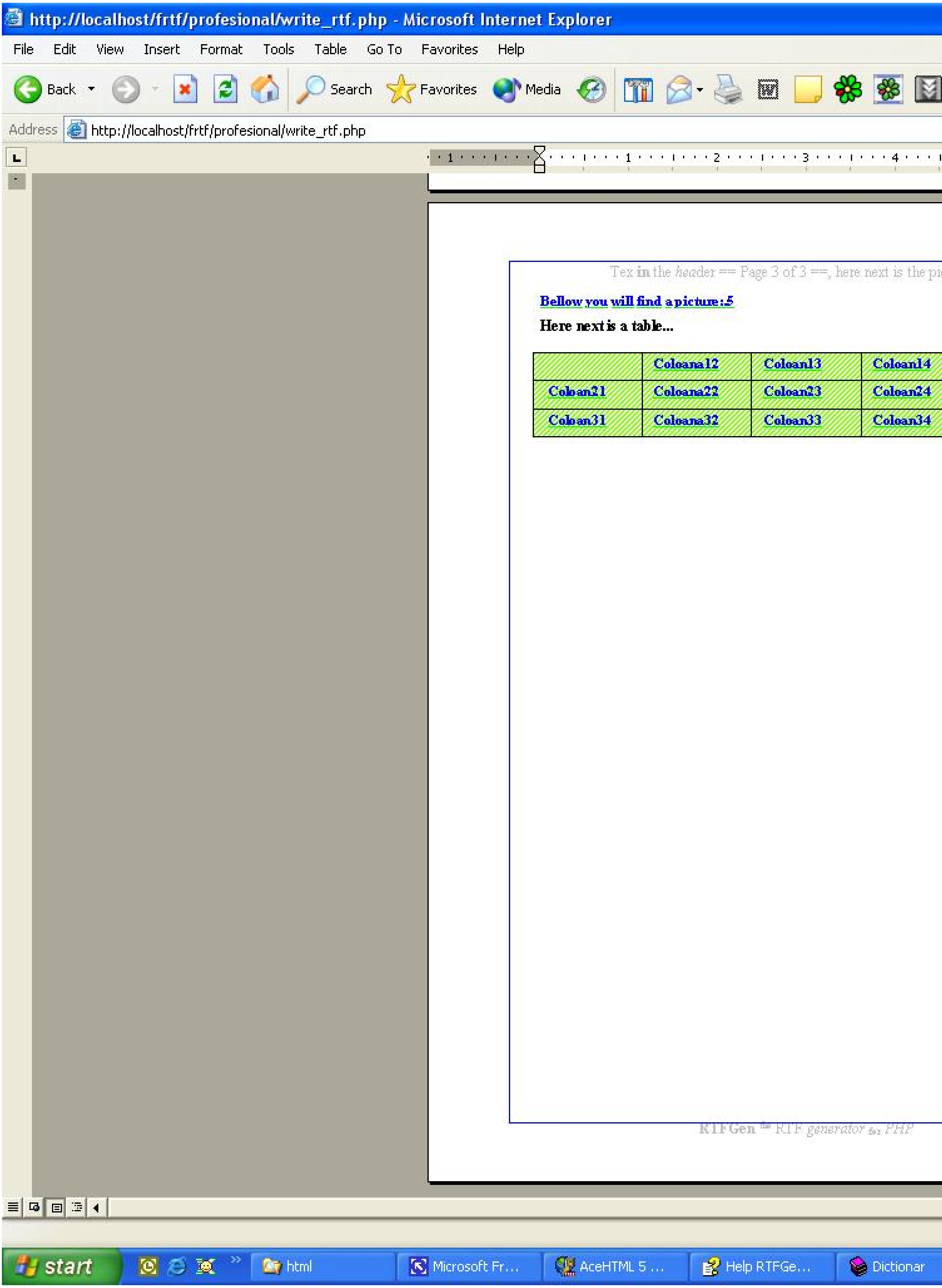
Here the 'Write text' style is used...

With RTFGen - Standard Edition you will be able to set the format of the document as you want, to add as many section as you want, to add your own color in the color table of the document, to add the fonts you want, to create your own formatting styles in the style sheet of the document, to write text within the paragraphs with the styles you have created, to add headers and footers, to add images, to create dynamically the tables you need and much more... .

In the next field we put the author of the document -> {AUTHOR * Upper* Lower* FirstCap* Caps * MERGEFORMAT }

~~This is an equation:~~ ~~And these are the comments~~[C.M.1]

RTFGen ™ RTF *generator* for *PHP*



Tex in the *header* == Page 3 of 3 ==, here next is the pi

Bellow you will find a picture:5

Here next is a table...

	Coloana12	Coloan13	Coloan14
Coloan21	Coloana22	Coloan23	Coloan24
Coloan31	Coloana32	Coloan33	Coloan34

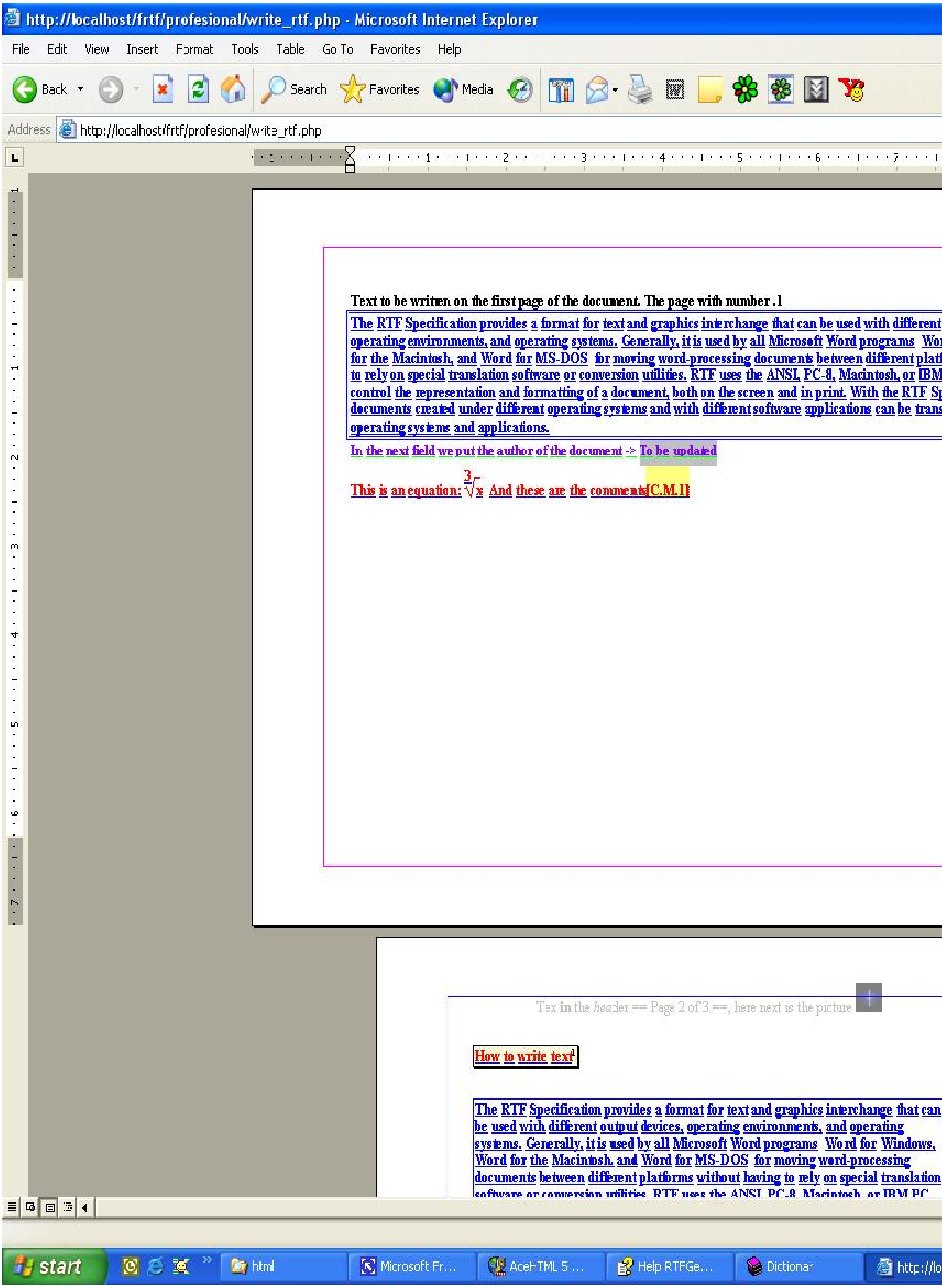
RTFGen ™ RTF *generator for PHP*

Sample 3

**How to add pictures and custom formatted tables to your document; how to add fields and pictures to the headers and footers,
Footnotes and endnotes:**

```
<?php
require('frtf.php');

$rtf=new FRTF();

$rtf->Open();

//some info document settings
$rtf->SetTitle("File generated by RTFGen V1.0 - Professional Edition");
$rtf->SetSubject("Subjectul Documentului");
$rtf->SetManager("Managerul Documentului");
$rtf->SetAuthor("Authorul Documentului");
$rtf->SetOperator("Operatorul Documentului");
$rtf->SetCompany("Compania Documentului");
$rtf->SetHlinkbase("Your BaseLink here");
$rtf->SetCreationTime();
$rtf->SetRevisionTime('12/23/2004 20:45:53');
$rtf->SetPrintTime('12/23/2004 20:45:53');
$rtf->SetBackupTime('12/23/2004 20:45:53');
$rtf->SetEditionTime('12');
$rtf->SetVersionDoc('3');
//some document formatting settings
$rtf->SetDefaultTab(0);
$rtf->SetHyphenHot('');
$rtf->SetHyphenConsec('');
$rtf->SetHyphenCaps('0');
$rtf->SetHyphenAuto('');
$rtf->SetLineStart(5);
$rtf->SetFracWidth();
$rtf->SetNextFile("NextFile here");
$rtf->SetTemplate("Template.doc");
$rtf->SetDefLang("Romanian");
$rtf->SetDefLangfe();

$rtf->SetWindowCaption('RTFGen for PHP');
$rtf->SetDocType('General Document');
$rtf->SetFromHtml();
$rtf->SetFromText();
$rtf->SetDocumentView('Page',100,'ful');
$rtf->SetPaperSize('a4');
//$rtf->SetMargins(10000,10000,2500,2500);
//$rtf->SetFacingp();
//$rtf->SetMargmirror();
//$rtf->SetTwoonone();
//$rtf->SetGutter(1000);
//$rtf->SetGutterPos('t');
//$rtf->SetRender();
//$rtf->SetDocSource();
$rtf->SetPageOrientation('l');// set the page orientation to landscape
$rtf->SetWidowCtrl();
//$rtf->SetPgnStart(4); //the page number will start with 4
//$rtf->SetBookfold();
//$rtf->SetBookfoldrev();
//$rtf->SetBookfoldsheets();
//$rtf->SetPageBorderHead();
//$rtf->SetPageBorderFoot();

//add 3 styles in the style sheets of the document. These style will be used to write formatted text in the document
$rtf->AddStyle("FONTDEF:[{Times New
Roman_Bold_12_255_word000000255_^_^_^_^_^_^}]APOCTL:[{W0H0_leftcol_toppar_noWrapB1440W187H187_^_^_0}]TABDEF:[{2_
sign}]SHADING[{20_255255_255255255}]BRDRDEF:[{shadowed_000000000_32_3_tblr}]LANGDEF[]PARFMT:[{Left_Body text_L0R
lui Cristi','','','Normal');
$rtf->AddStyle("FONTDEF:[{Times New Roman_Bold_12_000_word000000255_^_^_^_^_^_Sparkle Text}]TABDEF:[{2_center_u}{1.5_
sign}]SHADING[{20_255255_255255255}]BRDRDEF:[{shadowed_000000000_32_3_tblr}]LANGDEF[]PARFMT:[{Left_Body text_L0R
style 1','','','Normal');
$rtf->AddStyle("FONTDEF:[{Times New Roman_Bold_12_000_^_^_^_^_^_^_^}]PARFMT:[{Left_Body text_L0R0_^_Before3A3_single_W

$rtf->SetPageBorders('tlrb','single','d20','255000255',0,'','','no'); //set the borders for document
$rtf->Write("Text to be written on the first page of the document. The page with number .","Write text");
$rtf->InsertField('{PAGE \*MERGEFORMAT }');
$rtf->InsertBookmark('paralel');
```

The next 3 pictures will show the 3 pages of the document. Because the first row in the table is header it will be repeated on each page.



http://localhost/frtf/profesional/write_rtf.php

Text to be written on the first page of the document. The page with number .1

The RTF Specification provides a format for text and graphics interchange that can be used with different
operating environments, and operating systems. Generally, it is used by all Microsoft Word programs Wo
for the Macintosh, and Word for MS-DOS for moving word-processing documents between different platf
to rely on special translation software or conversion utilities. RTF uses the ANSI, PC-8, Macintosh, or IBM
control the representation and formatting of a document, both on the screen and in print. With the RTF Sp
documents created under different operating systems and with different software applications can be trans
operating systems and applications.

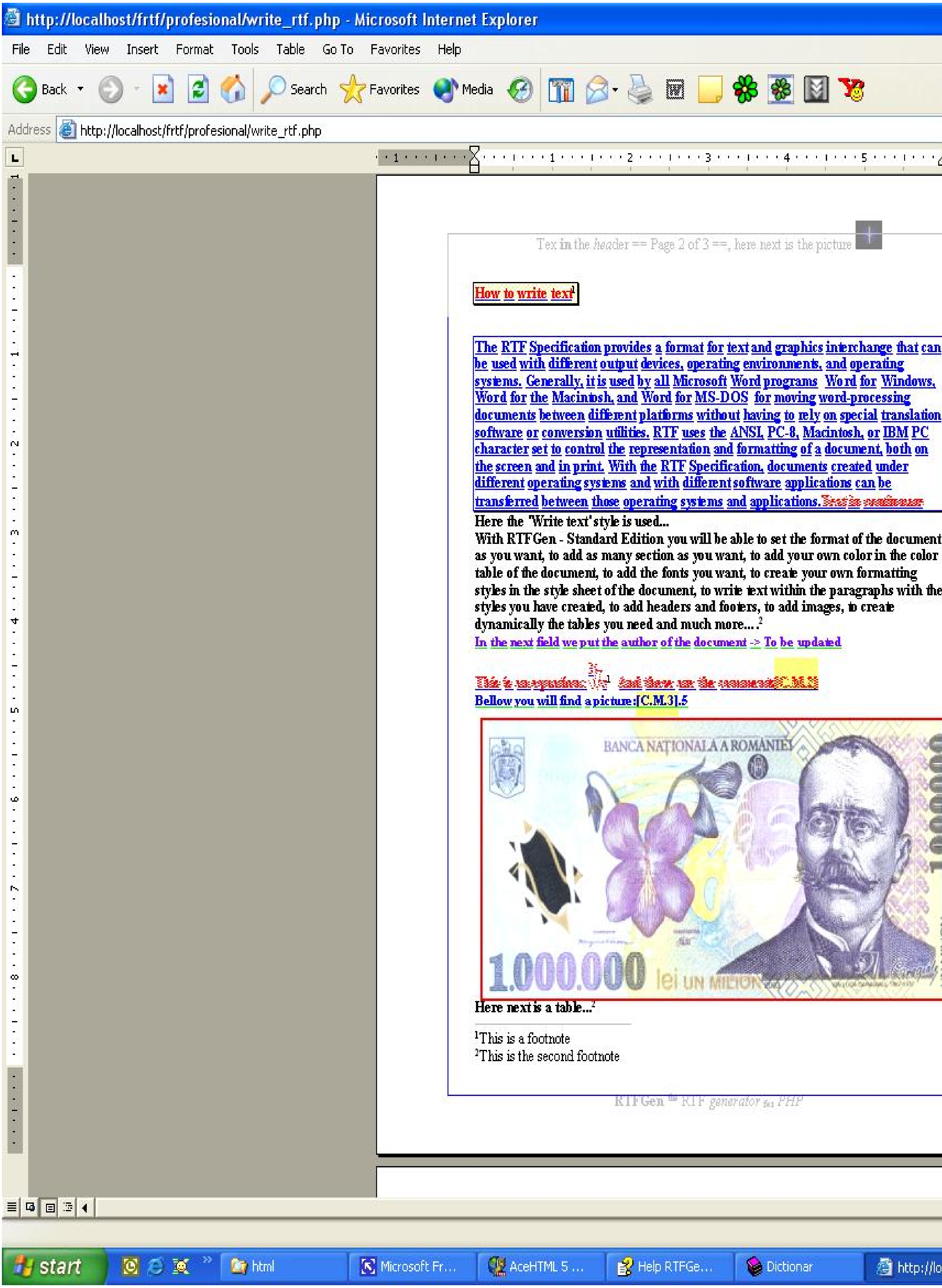
In the next field we put the author of the document -> To be updated

This is an equation: $\sqrt[3]{x}$ And these are the comments[C.M.1]

Tex in the *header* == Page 2 of 3 ==, here next is the picture

How to write text[1]

The RTF Specification provides a format for text and graphics interchange that can
be used with different output devices, operating environments, and operating
systems. Generally, it is used by all Microsoft Word programs Word for Windows,
Word for the Macintosh, and Word for MS-DOS for moving word-processing
documents between different platforms without having to rely on special translation
software or conversion utilities. RTF uses the ANSI, PC-8, Macintosh, or IBM PC


http://localhost/frtf/profesional/write_rtf.php - Microsoft Internet Explorer
File Edit View Insert Format Tools Table Go To Favorites Help
Back Search Favorites Media
Address http://localhost/frtf/profesional/write_rtf.php
start html Microsoft Fr... AceHTML 5 ... Help RTFGe... Dictionar http://lo

Tex in the *header* == Page 2 of 3 ==, here next is the picture

How to write text[1]

The RTF Specification provides a format for text and graphics interchange that can be used with different output devices, operating environments, and operating systems. Generally, it is used by all Microsoft Word programs Word for Windows, Word for the Macintosh, and Word for MS-DOS for moving word-processing documents between different platforms without having to rely on special translation software or conversion utilities. RTF uses the ANSI, PC-8, Macintosh, or IBM PC character set to control the representation and formatting of a document, both on the screen and in print. With the RTF Specification, documents created under different operating systems and with different software applications can be transferred between those operating systems and applications. ~~Text is [illegible]~~

Here the 'Write text' style is used...

With RTFGen - Standard Edition you will be able to set the format of the document as you want, to add as many section as you want, to add your own color in the color table of the document, to add the fonts you want, to create your own formatting styles in the style sheet of the document, to write text within the paragraphs with the styles you have created, to add headers and footers, to add images, to create dynamically the tables you need and much more... [2]

In the next field we put the author of the document -> To be updated

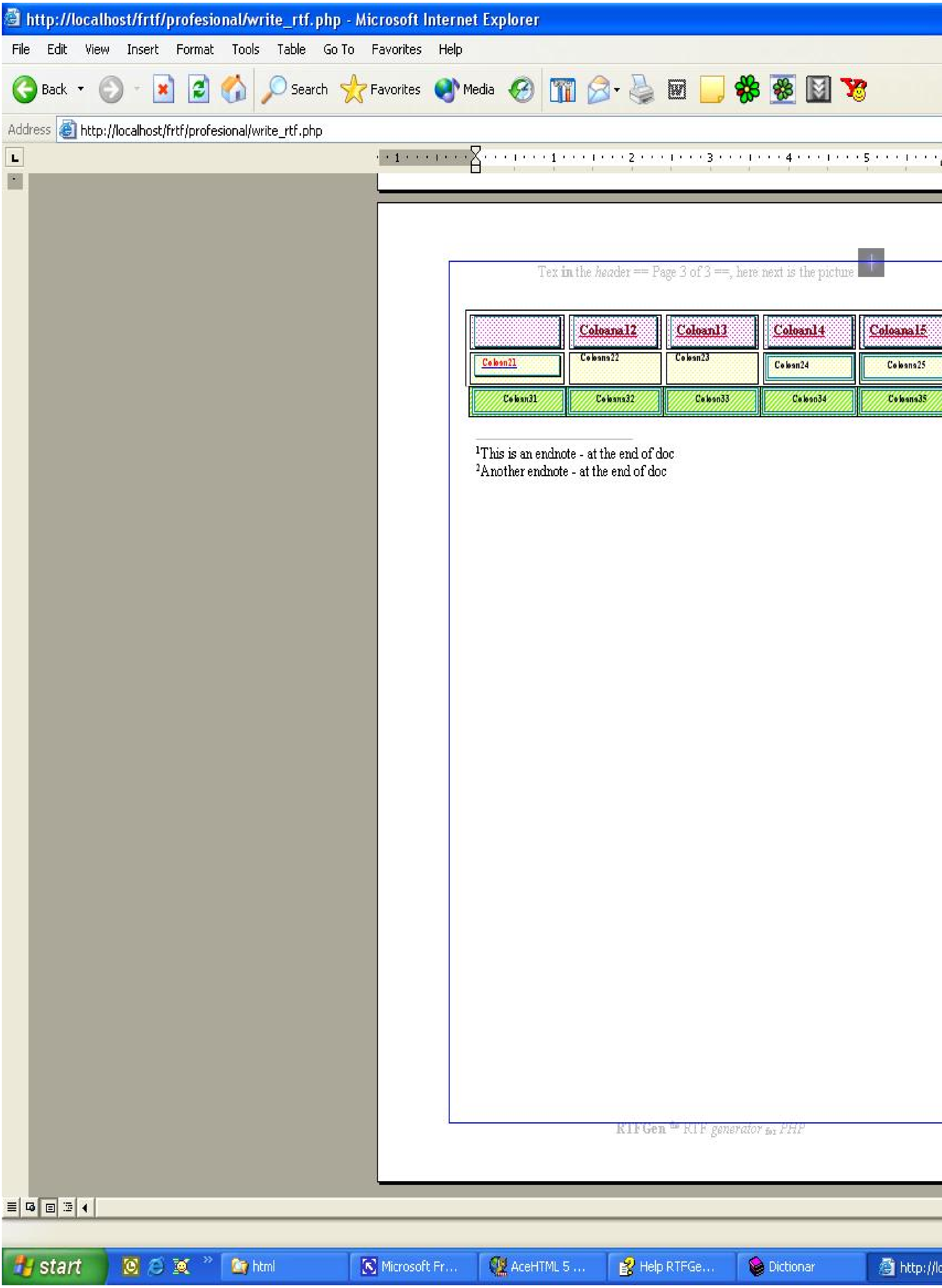
~~This is an equation:~~ [1] ~~And these are the comments[C.M.2]~~

Bellow you will find a picture:[C.M.3].5

Here next is a table...[2]

[1]This is a footnote
[2]This is the second footnote

RTFGen ™ RTF generator for PHP


http://localhost/frtf/profesional/write_rtf.php - Microsoft Internet Explorer
File Edit View Insert Format Tools Table Go To Favorites Help
Back
Search
Favorites
Media
Address http://localhost/frtf/profesional/write_rtf.php
start
html
Microsoft Fr...
AceHTML 5 ...
Help RTFGe...
Dictionar
http://lo

Tex in the *header* == Page 3 of 3 ==, here next is the picture

	Coloana12	Coloan13	Coloan14	Coloana15
Colean21	Coleana22	Colean23	Colean24	Coleana25
Colean31	Coleana32	Colean33	Colean34	Coleana35

[1] This is an endnote - at the end of doc
[2] Another endnote - at the end of doc

RTFGen ™ RTF *generator for PHP*

License agreement for RTFGen - Professional Edition

RTFGen Software License Agreement

(Version 1.00, 10, May, 2004 - the last version -> on the website)

0. SUBJECT MATTER.

This is a license agreement between Smart Systems, SRL (hereinafter referred to as the "Licensor") and ___________________________________ (hereinafter referred to as the "Licensee") in which Licensor grants Licensee certain rights in the software known as "RTFGen". The "Program", as used below, refers to the current release version of such program at the time of this agreement. The Program is copyrighted work, and Licensor holds its title and copyright.

BY USING, MODIFYING OR DISTRIBUTING THE PROGRAM OR ANY LARGER WORK THAT IS BASED ON THE PROGRAM, YOU INDICATE YOUR ACCEPTANCE OF THIS LICENSE, AND ALL ITS TERMS AND CONDITIONS FOR COPYING, DISTRIBUTING OR MODIFYING THE PROGRAM OR PRODUCING LARGER WORK BASED ON IT.

NOTHING OTHER THAN THIS LICENSE GRANTS YOU PERMISSION TO MODIFY OR DISTRIBUTE THE PROGRAM OR ITS DERIVATIVE WORKS. IF YOU DO NOT ACCEPT THESE TERMS AND CONDITIONS, DO NOT USE THE PROGRAM IN ANY FORM OR MANNER. COMMERCIAL USERS MUST ADDITIONALLY EXECUTE THE AGREEMENT WITH AUTHORIZED SIGNATURES.

1. DEFINITIONS.

1.1. "You" means the Licensee, an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 7.1. For legal entities, "you" includes any entity, which controls, is controlled by, or is under common control with you.

1.2. "Larger Work" or "Derivative Work" means any work that combines the Program or portions thereof with code not governed by the terms of this License.

1.3. "Source Code" means a collection of files for the Program in forms that are used for originally creating it or making modifications to it, including all modules it contains, plus any associated interface definition files, or a list of differential comparisons between different versions of it.

1.4. "Use" means an act of copying, editing, modifying, incorporating, distributing, and executing (or running) the Program or a subset thereof, or any Larger Work.

1.5. "Commercial Use" means any Use, as defined in section 1.4, of the Program or Larger Work for which any payment, profit or financial gain is sought directly or indirectly, regardless of whether the payment or gain is actually realized, and regardless of whether it is realized immediately or in the future (or in the past, if it is possible). The following examples are considered as instances of Commercial Use (This is not an exhaustive enumeration):

(a) Use of the Program or any Larger Work in any application programs, web based application, modules or plug-ins, shared libraries, loadable modules that are sold or licensed commercially in accordance with the definition of Commercial Use as defined in Section 1.5 of this License.

(b) Deployment of the Program or any Larger Work on a Internet or Intranet for server-side applications including, but not limited to, Commercial Uses as a part of World Wide Web services, or as a downloadable plug-in or applets in Web browser or any client application.

(c) Use of the Program or any Larger Work in an application program used in any business-related activities, including but not limited to industrial or laboratory testing, production or process control, product research and development, factory automation, quality and inventory control, report generation, production planning and executive or any level of decision making.

1.6. "Payment" (that you receive in the context of Commercial Use) means any form of monetary payment, securities and certificates of deposits, rights to exercise options for future financial gain, payment in employee or any form of benefits, promise or delivery of services, settlements of legal or commercial disputes, tangible materials and anything or right of financial value now or in the future. Direct payments include, but are not limited to, a payment for a copy of the Program or Larger Work, or for services or acts of using, modifying, developing or deploying the Program or Larger Work. Indirect payments include, but are not limited to, inclusion of the Program or Larger Work as a "no charge" or free "bundled" item in connection with services or product for which a payment is made. For the purpose of this License, a single instance of Commercial Use for a product or a service classifies the entire use for that product or service as Commercial Use, even if all other instances of uses are offered free of charge.

1.7. "License Fee" shall mean a one-time non-refundable fee payable to the Licensor in consideration for the Use of the Program (check items applicable):

1.7.1. For Standard Edition

___ A) Use in Single End-User Product or Application	200 €
___ B) Web Serving Single IP Address	200 €
___ C) Server-side Application License	2,000 €
___ D) Unlimited Corporate-wide License	2,000 €
___ E) Unlimited Server-side Application License	6,000 €
___ F) Unlimited Multi-national World-wide Corporate License	10,000 €

___ G) Inclusion in a commercial software development environment/library	<Please contact us>

1.7.2. For Professional Edition

___ A) Use in Single End-User Product or Application	400 €
___ B) Web Serving Single IP Address	400 €
___ C) Server-side Application License	4,000 €
___ D) Unlimited Corporate-wide License	4,000 €
___ E) Unlimited Server-side Application License	12,000 €
___ F) Unlimited Multi-national World-wide Corporate License	20,000 €
___ G) Inclusion in a commercial software development environment/library	<Please contact us>

If A) or B), list the product name or IP number/server hostname below:

____.

A Corporate-wide License applies to one incorporated unit and does not extend to subsidiaries. Corporations whose stocks are issued or traded separately are considered separate corporations by definition, and therefore cannot share a Corporate-wide license. Multi-national World-wide Corporate License covers groups of companies that are majority owned (ownership of more than 50%) by Licensee.

A Server-side application license allows Licensee to Use the Program in any number of server side applications and distribute them in unlimited manner.

Licensee, at its option, may elect to upgrade the License to a new version of the Program (the Upgrade Payment) at an additional fee, if more than two (2) years have elapsed from the date of this agreement or after the date of delivery of paid upgrades thereafter. Licensee is entitled to upgrades without charge, if and when they become available, for duration of two years from the date of this agreement or after the date of delivery of paid upgrades thereafter. Licensor solely determines availability, schedule and pricing of updates to the Program, and Licensor is not obligated to release updates for the Program. If you do not wish to upgrade, you may continue to Use the version of the Program obtained under a valid license in perpetuity. License Fee is non-refundable under any circumstance, even if Use ceases for any reason for a product, application, discontinuation of web servers, or due to corporate mergers or a termination of corporation. However, the License may be transferred to another product or Web server if the Use does not overlap within the Licensee organization.

1.8. "End-User Product" is software or application that is not capable of being used for developing or producing new applications that are Larger Work or Derivative Work.

2. LICENSES.

Licensor hereby grants Licensee a non-exclusive world-wide perpetual license to Use the Program, provided that Commercial License Fee as set forth in section 1.7 has been paid if applicable, and subject to third party intellectual property claims, and provided that you comply with all of the conditions set forth in this License.

2.1. You may copy and distribute literal (i.e., verbatim) copies of the Program's source code, documentation, and this License as you receive it throughout the world, in any medium.

2.2. You may Use the Program, create Larger Work based on the Program and distribute copies of such throughout the world, in any medium.

3. CONDITIONS.

This License is subject to the following conditions:

3.1. The Source Code of the Program, modified or unmodified, may be distributed only under the terms of this License or a future version of this License released under Section 7.1, and you must include a copy of this License (LICENSE.pdf or LICENSE.rtf) with every copy of the Source Code you distribute.

3.2. You must cause the modified files to carry prominent notices indicating that you have modified the files and the date of change.

3.3. If you received the Program or Larger Work from any party and you Use it to produce or develop further Derivative Work under Commercial Use, you must obtain a commercial license from Smart Systems, Ltd.

3.4. You must disclose any knowledge of intellectual property rights relevant to the modifications and additions as set forth in Section 6.

3.5. The Licensor, at his option, is entitled to incorporate modifications and additions to the Program to upgrade the Program that the Licensor distributes without any need for permission or any obligation (other than a moral obligation to acknowledge in relevant source files) if the modified source code is sent to the Licensor or is distributed via a generally accessible medium, including but not limited to mailing lists, Usenet News, Web and FTP sites, CDROMs and any other medium. However, you are not obligated to release or distribute modified source code of the Program or source code to any Larger Work.

3.6. You may not modify the data enclosed by { } (and including { } themselves) in the Generator destination of the Header Group of the generated RTF files. However, you may append any additional information to the Generator destination.

4. LICENSE FEE AND PAYMENT.

4.1. License Fee. In consideration of the license rights granted in Article 2 above, Licensee shall pay the License Fee for the Program and its documentation as set forth in 1.7 above. All amounts payable hereunder by Licensee shall be payable in European Union funds without deductions for taxes, assessments, fees, or charges of any kind. Checks shall be made payable to "Gabriela Muraru" and shall be forwarded, with two signed copies of this agreement to:

Smart Systems

Str. Cpt. Zaganescu, Nr.5, Sc. 4, Ap.20

Sector 6, Cod 026026

BUCURESTI

ROMANIA

or the current address noted on the Smart Systems' Web site: http://www.smartsystems.ro/

4.2. Taxes and Other Charges. Licensee shall be responsible for paying all (i) sales, use, excise, value-added, or other tax or governmental charges imposed on the licensing or use of the Program or documentation hereunder, (ii) freight, insurance and installation charges, and (iii) import or export duties or like charges, if any.

5. EXEMPTIONS AND DEFERMENT OF COMMERCIAL LICENSE FEE.

Exempt users listed herein are not required to send the signed license agreement to the Licensor. However, you are still bound by all applicable conditions and requirements as set forth in this License.

5.1. Non-Profit Organization Exemption

Uses, modifications, distributions, or deployments of the Program or Larger Work by educational institutions, non-profit organizations, and governments are exempt from Commercial Use license fees. Uses by consultants, contractors to non-profit organizations are exempt under this provision if the Use of the Program or Larger Work is solely in projects for these non-profit organizations and the rights to the resulting work including the source code belong to the nonprofit organization. If the contractors or consultants retain the rights to the resulting work including rights to sell or use the resulting work elsewhere, the exemption does not apply.

5.2. Private Non-Profit User Exemption

Non-profit uses, modifications, and distributions of the Program or Larger Work by a private individual are exempt from Commercial Use license fees. An example of use that does not fall under this exemption is the for-profit uses by individuals, such as in consulting or private business practices.

5.3. Posting the Program on a public access information storage and retrieval service for which a fee is received for retrieving information (such as an on-line service, an Internet service provider or a portal) is not considered a Commercial Use, provided that the fee is not content-dependent (i.e., the fee would be the same for retrieving the same volume of information consisting of random data) and that access to the service and to the Program is available independent of any other product or service. An example of a service that does not fall under this section is an on-line service that is operated by a company and that is only available to customers of that company. (This is not an exhaustive enumeration.)

5.4. Shareware Developer Deferment

Commercial Use in shareware applications that use the Program or any Larger Work is exempt from license fees until the cumulative total gross receipt of payments for all of your shareware applications that Use the Program exceeds ten (10) times the Commercial Use license fee for the Program. Thereafter, the shareware applications enjoy no special status and you must obtain Commercial Use license from the Licensor.

5.5. Distributing the Program on removable computer-readable media, provided that the files containing the Program are reproduced entirely and verbatim on such media, that all information on such media be redistributable for non-commercial purposes without charge, and that such media are distributed by themselves (except for accompanying documentation) independent of any other product or service. Examples of such media include CD-ROM, magnetic tape, and optical storage media. (This is not intended to be an exhaustive list.) An example of a distribution that does not fall under this section is a CD-ROM included in a book or magazine. (This is not an exhaustive enumeration.)

6. INTELLECTURAL PROPERTY MATTERS

6.1. Third Party Claims.

If You have knowledge that a party claims an intellectual property right in particular functionality or code (or its utilization under this License), you must include a text file with the source code titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. The LEGAL file must be prominently visible by any file or directory viewer and file listing commands, and reside in the same prominently visible directory or folder containing this License. If you obtain such knowledge after you make your modification or Larger Work available to others, You shall promptly modify the LEGAL file in all copies you make available thereafter, notify Smart Systems in writing or e-mail (sales@smartsystems.ro) and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Program that new knowledge has been obtained.

6.2. Patented API, technology and algorithm.

If your modification or Larger Work includes an application programming interface (API), technology or algorithm, and you own or control patents, which are reasonably necessary to implement that API, technology or algorithm, you must also include this information in the LEGAL file.

7. VERSIONS OF THE LICENSE.

7.1. New Versions.

Smart Systems may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number. The latest version of the License is available from http://www.smartsystems.ro/LICENSE.pdf or http://www.smartsystems.ro/LICENSE.rtf

7.2. Effect of New Versions.

Once the Program has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use the Program under the terms of any subsequent version of the License published by Smart Systems. No one other than Smart Systems has the right to modify the terms applicable to the Program created under this License.

7.3. Derivative Works.

If you create or use a modified version of this License (which you may only do in order to apply it to code which is not already part of the Program governed by this License), you must (a) rename Your license so that the phrases "Smart Systems", "RTFGen" or any confusingly similar phrase do not appear anywhere in your license and (b) otherwise make it clear that your version of the license contains terms which differ from the Smart Systems RTFGen Software License.

8. WARRANTY AND SUPPORT.

Smart Systems warrants that the program will perform substantially in accordance with the documentation when used as directed in the documentation. This warranty does not cover use of the program in ways that are not covered in the documentation (e.g., by calling undocumented functions, or by not obeying documented restrictions), or using modified copies of the program unless such modifications have been authorized by Smart Systems. Smart Systems warrants that it is owner of the program with authority to license the program to licensee, and that the program does not infringe third party intellectual property rights. Smart Systems agrees to indemnify, defend and hold harmless licensee from any claims either that Smart Systems does not own the program, or that the program infringes a third party’s intellectual property rights. In the event the unmodified program fails to satisfy this limited warranty during a warranty period of 30 days after the effective date of this agreement, Smart Systems shall promptly, at its expense and in its discretion, (i) provide a correction or workaround for any reproducible errors which are reported by licensee, and deliver an updated version of the program, or (ii) return a refund of any license fees paid pursuant to this agreement. In this event licensee will immediately terminate any use and distribution of the program.

Technical support as described above is available by e-mail only. Maintenance updates of the program, which are designated as such by Smart Systems, will do not extend the initial warranty period.

Neither party will be liable for any loss of use, interruption of business, or any indirect, special, incidental, or consequential damages of any kind (including lost profits), regardless of the form of action whether in contract, tort (including negligence), strict product liability or otherwise.

9. TERMINATION.

This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein, including non-payment of license fees, and fail to cure such breach within 30 days of becoming aware of the breach.

All sublicenses to the Program, which are properly granted, shall survive any termination of this License. Provisions, which, by their nature, must remain in effect beyond the termination of this License, shall survive.

10. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL SMART SYSTEMS, S.R.L.., ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF THE PROGRAM, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO YOU OR ANY OTHER PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR

DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THAT EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

11. APPLICABLE LAW.

The laws of Romania, excluding choice of law rules, govern this license. If any part of this license is found to be in conflict with the law, that part shall be interpreted in its broadest meaning consistent with the law, and no other parts of the license shall be affected.

12. RESPONSIBILITY FOR CLAIMS.

You are responsible for damages arising, directly or indirectly, out of Your utilization of rights under this License, based on the number of copies of the Program you made available, the revenues you received from utilizing such rights, and other relevant factors. You agree to work with affected parties to distribute responsibility on an equitable basis.

13. DELIVERY

The program and documentation are delivered in digital format only. Licensee agrees to receive the program and documentation, as well as any relevant maintenance updates from Smart System's by e-mail or retrieve (documentation and updates) from Smart Systems Web site at http://www.smartsystems.ro/. Smart Systems will deliver neither digital storage media nor printed documentation to licensee.

Commercial Users must fill out section 1.7 above and the sections below, and send two signed copies of this agreement along with the license fee payment to the address indicated in section 4.1.

One copy will be returned after an authorized representative of Smart Systems signs it.

THIS SOFTWARE LICENSE AGREEMENT is made and entered into this _____ day of

______________________ (month and year of the effective date) by and between Licensor and Licensee.

IN WITNESS WHEREOF, the parties have caused their duly authorized representatives to execute this Agreement as of the date first set forth above.

Licensor: Smart Systems, **Inc. Licensee: __________________________**

By ____________________________ By _______________________________________

(Authorized Company Signature) (Authorized Company
Signature)

_______________________________ ____

(Print Name) (Print Name)

Title ___________________________ Title ______________________________________

Str. Cpt.Zaganescu, Nr.5, Sc. 4, Ap.20

Sector 6, Cod 026026

BUCURESTI

ROMANIA ______________________________________

(Address)
(Address)

___