search
EN IT

PHP class for generating PDF documents

home | docs | examples | services | download | project

TCPDF Documentation

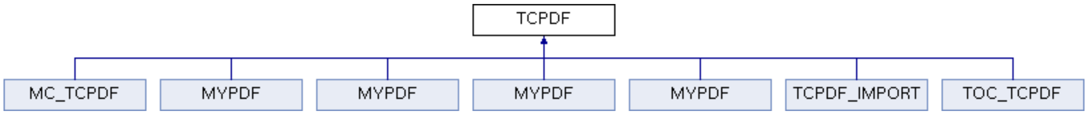
Main Page Related Pages Namespaces Classes Files

Class List Class Index Class Hierarchy Class Members

Public Member Functions | Protected Member Functions | Protected Attributes

TCPDF Class Reference

Inheritance diagram for TCPDF:



List of all members.

Public Member Functions

__construct ($orientation='P', $unit='mm', $format='A4', $unicode=true, $encoding='UTF-8', $diskcache=false, $pdfa=false)

__destruct ()

setPageUnit ($unit)

setPageOrientation ($orientation, $autopagebreak='', $bottommargin='')

setSpacesRE ($re='/[^\S\xa0]/')

setRTL ($enable, $resetx=true)

getRTL ()

setTempRTL ($mode)

isRTLTextDir ()

setLastH ($h)

getCellHeight ($fontsize, $padding=TRUE)

resetLastH ()

getLastH ()

setImageScale ($scale)

getImageScale ()

getPageDimensions ($pagenum='')

getPageWidth ($pagenum='')

getPageHeight ($pagenum='')

getBreakMargin ($pagenum='')

getScaleFactor ()

SetMargins ($left, $top, $right=-1, $keepmargins=false)

SetLeftMargin ($margin)

SetTopMargin ($margin)

SetRightMargin ($margin)

SetCellPadding ($pad)

setCellPaddings ($left='', $top='', $right='', $bottom='')

getCellPaddings ()

setCellMargins ($left='', $top='', $right='', $bottom='')

getCellMargins ()

SetAutoPageBreak ($auto, $margin=0)

getAutoPageBreak ()

SetDisplayMode ($zoom, $layout='SinglePage', $mode='UseNone')

SetCompression ($compress=true)

setSRGBmode ($mode=false)

SetDocInfoUnicode ($unicode=true)

SetTitle ($title)

SetSubject ($subject)

SetAuthor ($author)

SetKeywords ($keywords)

SetCreator ($creator)

Error ($msg)

Open ()

Close ()

setPage ($pnum, $resetmargins=false)

lastPage ($resetmargins=false)

getPage ()

getNumPages ()

addTOCPage ($orientation='', $format='', $keepmargins=false)

endTOCPage ()

AddPage ($orientation='', $format='', $keepmargins=false, $tocpage=false)

endPage ($tocpage=false)

startPage ($orientation='', $format='', $tocpage=false)

setPageMark ()

setHeaderData ($ln='', $lw=0, $ht='', $hs='', $tc=array(0, 0, 0), $lc=array(0, 0, 0))

setFooterData ($tc=array(0, 0, 0), $lc=array(0, 0, 0))

getHeaderData ()

setHeaderMargin ($hm=10)

getHeaderMargin ()

setFooterMargin ($fm=10)

getFooterMargin ()

setPrintHeader ($val=true)

setPrintFooter ($val=true)

getImageRBX ()

getImageRBY ()

resetHeaderTemplate ()

setHeaderTemplateAutoreset ($val=true)

Header ()

Footer ()

PageNo ()

getAllSpotColors ()

AddSpotColor ($name, $c, $m, $y, $k)

setSpotColor ($type, $name, $tint=100)

SetDrawSpotColor ($name, $tint=100)

SetFillSpotColor ($name, $tint=100)

SetTextSpotColor ($name, $tint=100)

setColorArray ($type, $color, $ret=false)

SetDrawColorArray ($color, $ret=false)

SetFillColorArray ($color, $ret=false)

SetTextColorArray ($color, $ret=false)

setColor ($type, $col1=0, $col2=-1, $col3=-1, $col4=-1, $ret=false, $name='')

SetDrawColor ($col1=0, $col2=-1, $col3=-1, $col4=-1, $ret=false, $name='')

SetFillColor ($col1=0, $col2=-1, $col3=-1, $col4=-1, $ret=false, $name='')

SetTextColor ($col1=0, $col2=-1, $col3=-1, $col4=-1, $ret=false, $name='')

GetStringWidth ($s, $fontname='', $fontstyle='', $fontsize=0, $getarray=false)

GetArrStringWidth ($sa, $fontname='', $fontstyle='', $fontsize=0, $getarray=false)

GetCharWidth ($char, $notlast=true)

getRawCharWidth ($char)

GetNumChars ($s)

unichr ($c)

addTTFfont ($fontfile, $fonttype='', $enc='', $flags=32, $outpath='', $platid=3, $encid=1, $addcbbox=false)

AddFont ($family, $style='', $fontfile='', $subset='default')

SetFont ($family, $style='', $size=null, $fontfile='', $subset='default', $out=true)

SetFontSize ($size, $out=true)

getFontBBox ()

getAbsFontMeasure ($s)

getCharBBox ($char)

getFontDescent ($font, $style='', $size=0)

getFontAscent ($font, $style='', $size=0)

isCharDefined ($char, $font='', $style='')

replaceMissingChars ($text, $font='', $style='', $subs=array())

SetDefaultMonospacedFont ($font)

AddLink ()

SetLink ($link, $y=0, $page=-1)

Link ($x, $y, $w, $h, $link, $spaces=0)

Annotation ($x, $y, $w, $h, $text, $opt=array('Subtype'=>'**Text**'), $spaces=0)

Text ($x, $y, $txt, $fstroke=false, $fclip=false, $ffill=true, $border=0, $ln=0, $align='', $fill=false, $link='', $stretch=0, $ignore_min_height=false, $calign='T', $valign='M', $rtloff=false)

AcceptPageBreak ()

Cell ($w, $h=0, $txt='', $border=0, $ln=0, $align='', $fill=false, $link='', $stretch=0, $ignore_min_height=false, $calign='T', $valign='M')

MultiCell ($w, $h, $txt, $border=0, $align='J', $fill=false, $ln=1, $x='', $y='', $reseth=true, $stretch=0, $ishtml=false, $autopadding=true, $maxh=0, $valign='T', $fitcell=false)

getNumLines ($txt, $w=0, $reseth=false, $autopadding=true, $cellpadding='', $border=0)

getStringHeight ($w, $txt, $reseth=false, $autopadding=true, $cellpadding='', $border=0)

Write ($h, $txt, $link='', $fill=false, $align='', $ln=false, $stretch=0, $firstline=false, $firstblock=false, $maxh=0, $wadj=0, $margin='')

Image ($file, $x='', $y='', $w=0, $h=0, $type='', $link='', $align='', $resize=false, $dpi=300, $palign='', $ismask=false, $imgmask=false, $border=0, $fitbox=false, $hidden=false, $fitonpage=false, $alt=false, $altimgs=array())

Ln ($h='', $cell=false)

GetX ()

GetAbsX ()

GetY ()

SetX ($x, $rtloff=false)

SetY ($y, $resetx=true, $rtloff=false)

SetXY ($x, $y, $rtloff=false)

SetAbsX ($x)

SetAbsY ($y)

SetAbsXY ($x, $y)

Output ($name='doc.pdf', $dest='I')

_destroy ($destroyall=false, $preserve_objcopy=false)

setExtraXMP ($xmp)

setDocCreationTimestamp ($time)

setDocModificationTimestamp ($time)

getDocCreationTimestamp ()

getDocModificationTimestamp ()

setHeaderFont ($font)

getHeaderFont ()

setFooterFont ($font)

getFooterFont ()

setLanguageArray ($language)

getPDFData ()

addHtmlLink ($url, $name, $fill=false, $firstline=false, $color='', $style=-1, $firstblock=false)

pixelsToUnits ($px)

unhtmlentities ($text_to_convert)

SetProtection ($permissions=array('print', 'modify', 'copy', 'annot-forms', 'fill-forms', 'extract', 'assemble', 'print-high'), $user_pass='', $owner_pass=null, $mode=0, $pubkeys=null)

StartTransform ()

StopTransform ()

ScaleX ($s_x, $x='', $y='')

ScaleY ($s_y, $x='', $y='')

ScaleXY ($s, $x='', $y='')

Scale ($s_x, $s_y, $x='', $y='')

MirrorH ($x='')

MirrorV ($y='')

MirrorP ($x='', $y='')

MirrorL ($angle=0, $x='', $y='')

TranslateX ($t_x)

TranslateY ($t_y)

Translate ($t_x, $t_y)

Rotate ($angle, $x='', $y='')

SkewX ($angle_x, $x='', $y='')

SkewY ($angle_y, $x='', $y='')

Skew ($angle_x, $angle_y, $x='', $y='')

SetLineWidth ($width)

GetLineWidth ()

SetLineStyle ($style, $ret=false)

Line ($x1, $y1, $x2, $y2, $style=array())

Rect ($x, $y, $w, $h, $style='', $border_style=array(), $fill_color=array())

Curve ($x0, $y0, $x1, $y1, $x2, $y2, $x3, $y3, $style='', $line_style=array(), $fill_color=array())

Polycurve ($x0, $y0, $segments, $style='', $line_style=array(), $fill_color=array())

Ellipse ($x0, $y0, $rx, $ry='', $angle=0, $astart=0, $afinish=360, $style='', $line_style=array(), $fill_color=array(), $nc=2)

Circle ($x0, $y0, $r, $angstr=0, $angend=360, $style='', $line_style=array(), $fill_color=array(), $nc=2)

PolyLine ($p, $style='', $line_style=array(), $fill_color=array())

Polygon ($p, $style='', $line_style=array(), $fill_color=array(), $closed=true)

RegularPolygon ($x0, $y0, $r, $ns, $angle=0, $draw_circle=false, $style='', $line_style=array(), $fill_color=array(), $circle_style='', $circle_outLine_style=array(), $circle_fill_color=array())

StarPolygon ($x0, $y0, $r, $nv, $ng, $angle=0, $draw_circle=false, $style='', $line_style=array(), $fill_color=array(), $circle_style='', $circle_outLine_style=array(), $circle_fill_color=array())

RoundedRect ($x, $y, $w, $h, $r, $round_corner='1111', $style='', $border_style=array(), $fill_color=array())

RoundedRectXY ($x, $y, $w, $h, $rx, $ry, $round_corner='1111', $style='', $border_style=array(), $fill_color=array())

Arrow ($x0, $y0, $x1, $y1, $head_style=0, $arm_size=5, $arm_angle=15)

setDestination ($name, $y=-1, $page='', $x=-1)

getDestination ()

setBookmark ($txt, $level=0, $y=-1, $page='', $style='', $color=array(0, 0, 0), $x=-1, $link='')

Bookmark ($txt, $level=0, $y=-1, $page='', $style='', $color=array(0, 0, 0), $x=-1, $link='')

IncludeJS ($script)

addJavascriptObject ($script, $onload=false)

setFormDefaultProp ($prop=array())

getFormDefaultProp ()

TextField ($name, $w, $h, $prop=array(), $opt=array(), $x='', $y='', $js=false)

RadioButton ($name, $w, $prop=array(), $opt=array(), $onvalue='On', $checked=false, $x='', $y='', $js=false)

ListBox ($name, $w, $h, $values, $prop=array(), $opt=array(), $x='', $y='', $js=false)

ComboBox ($name, $w, $h, $values, $prop=array(), $opt=array(), $x='', $y='', $js=false)

CheckBox ($name, $w, $checked=false, $prop=array(), $opt=array(), $onvalue='Yes', $x='', $y='', $js=false)

Button ($name, $w, $h, $caption, $action, $prop=array(), $opt=array(), $x='', $y='', $js=false)

setUserRights ($enable=true, $document='/FullSave', $annots='/Create/Delete/Modify/Copy/Import/Export', $form='/Add/Delete/FillIn/Import/Export/SubmitStandalone/SpawnTemplate', $signature='/Modify', $ef='/Create/Delete/Modify/Import', $formex='')

setSignature ($signing_cert='', $private_key='', $private_key_password='', $extracerts='', $cert_type=2, $info=array())

setSignatureAppearance ($x=0, $y=0, $w=0, $h=0, $page=-1, $name='')

addEmptySignatureAppearance ($x=0, $y=0, $w=0, $h=0, $page=-1, $name='')

startPageGroup ($page='')

setStartingPageNumber ($num=1)

getAliasRightShift ()

getAliasNbPages ()

getAliasNumPage ()

getPageGroupAlias ()

getPageNumGroupAlias ()

getGroupPageNo ()

getGroupPageNoFormatted ()

PageNoFormatted ()

startLayer ($name='', $print=true, $view=true, $lock=true)

endLayer ()

setVisibility ($v)

setOverprint ($stroking=true, $nonstroking='', $mode=0)

getOverprint ()

setAlpha ($stroking=1, $bm='Normal', $nonstroking='', $ais=false)

getAlpha ()

setJPEGQuality ($quality)

setDefaultTableColumns ($cols=4)

setCellHeightRatio ($h)

getCellHeightRatio ()

setPDFVersion ($version='1.7')

setViewerPreferences ($preferences)

colorRegistrationBar ($x, $y, $w, $h, $transition=true, $vertical=false, $colors='A, R, G, B, C, M, Y, K')

cropMark ($x, $y, $w, $h, $type='T, R, B, L', $color=array(100, 100, 100, 100,'All'))

registrationMark ($x, $y, $r, $double=false, $cola=array(100, 100, 100, 100,'All'), $colb=array(0, 0, 0, 0,'None'))

registrationMarkCMYK ($x, $y, $r)

LinearGradient ($x, $y, $w, $h, $col1=array(), $col2=array(), $coords=array(0, 0, 1, 0))

RadialGradient ($x, $y, $w, $h, $col1=array(), $col2=array(), $coords=array(0.5, 0.5, 0.5, 0.5, 1))

CoonsPatchMesh ($x, $y, $w, $h, $col1=array(), $col2=array(), $col3=array(), $col4=array(), $coords=array(0.00, 0.0, 0.33, 0.00, 0.67, 0.00, 1.00, 0.00, 1.00, 0.33, 1.00, 0.67, 1.00, 1.00, 0.67, 1.00, 0.33, 1.00, 0.00, 1.00, 0.00, 0.67, 0.00, 0.33), $coords_min=0, $coords_max=1, $antialias=false)

Gradient ($type, $coords, $stops, $background=array(), $antialias=false)

PieSector ($xc, $yc, $r, $a, $b, $style='FD', $cw=true, $o=90)

PieSectorXY ($xc, $yc, $rx, $ry, $a, $b, $style='FD', $cw=false, $o=0, $nc=2)

ImageEps ($file, $x='', $y='', $w=0, $h=0, $link='', $useBoundingBox=true, $align='', $palign='', $border=0, $fitonpage=false, $fixoutvals=false)

setBarcode ($bc='')

getBarcode ()

write1DBarcode ($code, $type, $x='', $y='', $w='', $h='', $xres='', $style='', $align='')

write2DBarcode ($code, $type, $x='', $y='', $w='', $h='', $style='', $align='', $distort=false)

getMargins ()

getOriginalMargins ()

getFontSize ()

getFontSizePt ()

getFontFamily ()

getFontStyle ()

fixHTMLCode ($html, $default_css='', $tagvs='', $tidy_options='')

getCSSPadding ($csspadding, $width=0)

getCSSMargin ($cssmargin, $width=0)

getCSSBorderMargin ($cssbspace, $width=0)

getHTMLFontUnits ($val, $refsize=12, $parent_size=12, $defaultunit='pt')

serializeTCPDFtagParameters ($pararray)

writeHTMLCell ($w, $h, $x, $y, $html='', $border=0, $ln=0, $fill=false, $reseth=true, $align='', $autopadding=true)

writeHTML ($html, $ln=true, $fill=false, $reseth=false, $cell=false, $align='')

setLIsymbol ($symbol='!')

SetBooklet ($booklet=true, $inner=-1, $outer=-1)

setHtmlVSpace ($tagvs)

setListIndentWidth ($width)

setOpenCell ($isopen)

setHtmlLinksStyle ($color=array(0, 0, 255), $fontstyle='U')

getHTMLUnitToUnits ($htmlval, $refsize=1, $defaultunit='px', $points=false)

movePage ($frompage, $topage)

deletePage ($page)

copyPage ($page=0)

addTOC ($page='', $numbersfont='', $filler='.', $toc_name='TOC', $style='', $color=array(0, 0, 0))

addHTMLTOC ($page='', $toc_name='TOC', $templates=array(), $correct_align=true, $style='', $color=array(0, 0, 0))

startTransaction ()

commitTransaction ()

rollbackTransaction ($self=false)

setEqualColumns ($numcols=0, $width=0, $y='')

resetColumns ()

setColumnsArray ($columns)

selectColumn ($col='')

getColumn ()

getNumberOfColumns ()

setTextRenderingMode ($stroke=0, $fill=true, $clip=false)

setTextShadow ($params=array('enabled'=>false, 'depth_w'=>0, 'depth_h'=>0, 'color'=>false, 'opacity'=>1, 'blend_mode'=>'Normal'))

getTextShadow ()

hyphenateText ($text, $patterns, $dictionary=array(), $leftmin=1, $rightmin=2, $charmin=1, $charmax=8)

setRasterizeVectorImages ($mode)

setFontSubsetting ($enable=true)

getFontSubsetting ()

- **stringLeftTrim** ($str, $replace='')
- **stringRightTrim** ($str, $replace='')
- **stringTrim** ($str, $replace='')
- **isUnicodeFont** ()
- **getFontFamilyName** ($fontfamily)
- **startTemplate** ($w=0, $h=0, $group=false)
- **endTemplate** ()
- **printTemplate** ($id, $x='', $y='', $w=0, $h=0, $align='', $palign='', $fitonpage=false)
- **setFontStretching** ($perc=100)
- **getFontStretching** ()
- **setFontSpacing** ($spacing=0)
- **getFontSpacing** ()
- **getPageRegions** ()
- **setPageRegions** ($regions=array())
- **addPageRegion** ($region)
- **removePageRegion** ($key)
- **ImageSVG** ($file, $x='', $y='', $w=0, $h=0, $link='', $align='', $palign='', $border=0, $fitonpage=false)

Protected Member Functions

- **setPageFormat** ($format, $orientation='P')
- **adjustCellPadding** ($brd=0)
- **setContentMark** ($page=0)
- **setHeader** ()
- **setFooter** ()
- **inPageBody** ()
- **setTableHeader** ()
- **getFontsList** ()
- **_putEmbeddedFiles** ()
- **checkPageBreak** ($h=0, $y='', $addpage=true)
- **getCellCode** ($w, $h=0, $txt='', $border=0, $ln=0, $align='', $fill=false, $link='', $stretch=0, $ignore_min_height=false, $calign='T', $valign='M')
- **replaceChar** ($oldchar, $newchar)
- **getCellBorder** ($x, $y, $w, $h, $brd)
- **getRemainingWidth** ()
- **fitBlock** ($w, $h, $x, $y, $fitonpage=false)
- **ImagePngAlpha** ($file, $x, $y, $wpx, $hpx, $w, $h, $type, $link, $align, $resize, $dpi, $palign, $filehash='')
- **getGDgamma** ($img, $c)
- **_dochecks** ()
- **getInternalPageNumberAliases** ($a= '')
- **getAllInternalPageNumberAliases** ()
- **replaceRightShiftPageNumAliases** ($page, $aliases, $diff)
- **setPageBoxTypes** ($boxes)
- **_putpages** ()
- **_putannotsrefs** ($n)
- **_getannotsrefs** ($n)
- **_putannotsobjs** ()
- **_putAPXObject** ($w=0, $h=0, $stream='')
- **_putfonts** ()
- **_puttruetypeunicode** ($font)
- **_putcidfont0** ($font)
- **_putimages** ()
- **_putxobjects** ()
- **_putspotcolors** ()
- **_getxobjectdict** ()
- **_putresourcedict** ()
- **_putresources** ()
- **_putinfo** ()
- **_putXMP** ()
- **_putcatalog** ()
- **_putviewerpreferences** ()
- **_putheader** ()

- **_enddoc** ()
- **_beginpage** ($orientation='', $format='')
- **_endpage** ()
- **_newobj** ()
- **_getobj** ($objid='')
- **_dounderline** ($x, $y, $txt)
- **_dounderlinew** ($x, $y, $w)
- **_dolinethrough** ($x, $y, $txt)
- **_dolinethroughw** ($x, $y, $w)
- **_dooverline** ($x, $y, $txt)
- **_dooverlinew** ($x, $y, $w)
- **_datastring** ($s, $n=0)
- **_datestring** ($n=0, $timestamp=0)
- **_textstring** ($s, $n=0)
- **_escapetext** ($s)
- **_getrawstream** ($s, $n=0)
- **_getstream** ($s, $n=0)
- **_putstream** ($s, $n=0)
- **_out** ($s)
- **_objectkey** ($n)
- **_encrypt_data** ($n, $s)
- **_putencryption** ()
- **_Uvalue** ()
- **_UEvalue** ()
- **_Ovalue** ()
- **_OEvalue** ()
- **_fixAES256Password** ($password)
- **_generateencryptionkey** ()
- **Transform** ($tm)
- **_outPoint** ($x, $y)
- **_outLine** ($x, $y)
- **_outRect** ($x, $y, $w, $h, $op)
- **_outCurve** ($x1, $y1, $x2, $y2, $x3, $y3)
- **_outCurveV** ($x2, $y2, $x3, $y3)
- **_outCurveY** ($x1, $y1, $x3, $y3)
- **_outellipticalarc** ($xc, $yc, $rx, $ry, $xang=0, $angs=0, $angf=360, $pie=false, $nc=2, $startpoint=true, $ccw=true, $svg=false)
- **_putdests** ()
- **sortBookmarks** ()
- **_putbookmarks** ()
- **_putjavascript** ()
- **_addfield** ($type, $name, $x, $y, $w, $h, $prop)
- **_putsignature** ()
- **getSignatureAppearanceArray** ($x=0, $y=0, $w=0, $h=0, $page=-1, $name='')
- **_putocg** ()
- **addExtGState** ($parms)
- **setExtGState** ($gs)
- **_putextgstates** ()
- **Clip** ($x, $y, $w, $h)
- **_putshaders** ()
- **getCSSBorderWidth** ($width)
- **getCSSBorderDashStyle** ($style)
- **getCSSBorderStyle** ($cssborder)
- **getCSSFontSpacing** ($spacing, $parent=0)
- **getCSSFontStretching** ($stretch, $parent=100)
- **getHtmlDomArray** ($html)
- **getSpaceString** ()
- **openHTMLTagHandler** ($dom, $key, $cell)
- **closeHTMLTagHandler** ($dom, $key, $cell, $maxbottomliney=0)
- **addHTMLVertSpace** ($hbz=0, $hb=0, $cell=false, $firsttag=false, $lasttag=false)
- **getBorderStartPosition** ()

drawHTMLTagBorder ($tag, $xmax)

swapMargins ($reverse=true)

putHtmlListBullet ($listdepth, $listtype='', $size=10)

getGraphicVars ()

setGraphicVars ($gvars, $extended=false)

_outSaveGraphicsState ()

_outRestoreGraphicsState ()

writeDiskCache ($filename, $data, $append=false)

readDiskCache ($filename)

setBuffer ($data)

replaceBuffer ($data)

getBuffer ()

setPageBuffer ($page, $data, $append=false)

getPageBuffer ($page)

setImageBuffer ($image, $data)

setImageSubBuffer ($image, $key, $data)

getImageBuffer ($image)

setFontBuffer ($font, $data)

setFontSubBuffer ($font, $key, $data)

getFontBuffer ($font)

hyphenateWord ($word, $patterns, $dictionary=array(), $leftmin=1, $rightmin=2, $charmin=1, $charmax=8)

checkPageRegions ($h, $x, $y)

convertSVGtMatrix ($tm)

SVGTransform ($tm)

setSVGStyles ($svgstyle, $prevsvgstyle, $x=0, $y=0, $w=1, $h=1, $clip_function='', $clip_params=array())

SVGPath ($d, $style='')

startSVGElementHandler ($parser, $name, $attribs, $ctm=array())

endSVGElementHandler ($parser, $name)

segSVGContentHandler ($parser, $data)

Protected Attributes

$page

$n

$offsets = array()

$pageobjects = array()

$buffer

$pages = array()

$state

$compress

$CurOrientation

$pagedim = array()

$k

$fwPt

$fhPt

$wPt

$hPt

$w

$h

$lMargin

$rMargin

$clMargin

$crMargin

$tMargin

$bMargin

$cell_padding = array('T' => 0, 'R' => 0, 'B' => 0, 'L' => 0)

$cell_margin = array('T' => 0, 'R' => 0, 'B' => 0, 'L' => 0)

$x

$y

$lasth

$LineWidth

$CoreFonts

$fonts = array()

$FontFiles = array()

$diffs = array()

$images = array()

$cached_files = array()

$PageAnnots = array()

$links = array()

$FontFamily

$FontStyle

$FontAscent

$FontDescent

$underline

$overline

$CurrentFont

$FontSizePt

$FontSize

$DrawColor

$FillColor

$TextColor

$ColorFlag

$AutoPageBreak

$PageBreakTrigger

$InHeader = false

$InFooter = false

$ZoomMode

$LayoutMode

$docinfounicode = true

$title = ''

$subject = ''

$author = ''

$keywords = ''

$creator = ''

$starting_page_number = 1

$img_rb_x

$img_rb_y

$imgscale = 1

$isunicode = false

$PDFVersion = '1.7'

$header_xobjid = false

$header_xobj_autoreset = false

$header_margin

$footer_margin

$original_lMargin

$original_rMargin

$header_font

$footer_font

$l

$barcode = false

$print_header = true

$print_footer = true

$header_logo = ''

$header_logo_width = 30

$header_title = ''

$header_string = ''

$header_text_color = array(0,0,0)

$header_line_color = array(0,0,0)

$footer_text_color = array(0,0,0)

$footer_line_color = array(0,0,0)

$txtshadow = array('enabled'=>false, 'depth_w'=>0, 'depth_h'=>0, 'color'=>false, 'opacity'=>1, 'blend_mode'=>'Normal')

$default_table_columns = 4

$HREF = array()

$fontlist = array()

$fgcolor

$listordered = array()

$listcount = array()

$listnum = 0

$listindent = 0

$listindentlevel = 0

$bgcolor

$tempfontsize = 10

$lispacer = ''

$encoding = 'UTF-8'

$internal_encoding

$rtl = false

$tmprtl = false

$encrypted

$encryptdata = array()

$last_enc_key

$last_enc_key_c

$file_id

$outlines = array()

$OutlineRoot

$javascript = ''

$n_js

$linethrough

$ur = array()

$dpi = 72

$newpagegroup = array()

$pagegroups = array()

$currpagegroup = 0

$extgstates

$jpeg_quality

$cell_height_ratio = **K_CELL_HEIGHT_RATIO**

$viewer_preferences

$PageMode

$gradients = array()

$intmrk = array()

$bordermrk = array()

$emptypagemrk = array()

$cntmrk = array()

$footerpos = array()

$footerlen = array()

$newline = true

$endlinex = 0

$linestyleWidth = ''

$linestyleCap = '0 J'

$linestyleJoin = '0 j'

$linestyleDash = '[] 0 d'

$openMarkedContent = false

$htmlvspace = 0

$spot_colors = array()

$lisymbol = ''

$epsmarker = 'x#!#EPS#!#x'

$transfmatrix = array()

$transfmatrix_key = 0

$booklet = false

$feps = 0.005

$tagvspaces = array()

$customlistindent = -1

$opencell = true

$embeddedfiles = array()

$premode = false

$transfmrk = array()

$htmlLinkColorArray = array(0, 0, 255)

$htmlLinkFontStyle = 'U'

$numpages = 0

$pagelen = array()

$numimages = 0

$imagekeys = array()

$bufferlen = 0

$diskcache = false

$numfonts = 0

$fontkeys = array()

$font_obj_ids = array()

$pageopen = array()

$default_monospaced_font = 'courier'

$objcopy

$cache_file_length = array()

$thead = ''

$theadMargins = array()

$sign = false

$signature_data = array()

$signature_max_length = 11742

$signature_appearance = array('page' => 1, 'rect' => '0 0 0 0')

$empty_signature_appearance = array()

$re_spaces = '/[^\S\xa0]/'

$re_space = array('p' => '[^\S\xa0]', 'm' => '')

$sig_obj_id = 0

$page_obj_id = array()

$form_obj_id = array()

$default_form_prop = array('lineWidth'=>1, 'borderStyle'=>'solid', 'fillColor'=>array(255, 255, 255), 'strokeColor'=>array(128, 128, 128))

$js_objects = array()

$form_action = ''

$form_enctype = 'application/x-www-form-urlencoded'

$form_mode = 'post'

$annotation_fonts = array()

$radiobutton_groups = array()

$radio_groups = array()

$textindent = 0

$start_transaction_page = 0

$start_transaction_y = 0

$inthead = false

$columns = array()

$num_columns = 1

$current_column = 0

$column_start_page = 0

$maxselcol = array('page' => 0, 'column' => 0)

$colxshift = array('x' => 0, 's' => array('H' => 0, 'V' => 0), 'p' => array('L' => 0, 'T' => 0, 'R' => 0, 'B' => 0))

$textrendermode = 0

$textstrokewidth = 0

$strokecolor

$pdfunit = 'mm'

$tocpage = false

$rasterize_vector_images = false

$font_subsetting = true

$default_graphic_vars = array()

$xobjects = array()

$inxobj = false

$xobjid = ''

$font_stretching = 100

$font_spacing = 0

$page_regions = array()

$check_page_regions = true

$pdflayers = array()

$dests = array()

$n_dests

$efnames = array()

$svgdir = ''

$svgunit = 'px'

$svggradients = array()

$svggradientid = 0

$svgdefsmode = false

$svgdefs = array()

$svgclipmode = false

$svgclippaths = array()

$svgcliptm = array()

$svgclipid = 0

$svgtext = ''

$svgtextmode = array()

$svgstyles

$force_srgb = false

$pdfa_mode = false

$doc_creation_timestamp

$doc_modification_timestamp

$custom_xmp = ''

$overprint = array('OP' => false, 'op' => false, 'OPM' => 0)

$alpha = array('CA' => 1, 'ca' => 1, 'BM' => '/Normal', 'AIS' => false)

$page_boxes = array('MediaBox', 'CropBox', 'BleedBox', 'TrimBox', 'ArtBox')

$tcpdflink = true

$gdgammacache = array()

Detailed Description

PHP class for generating PDF documents without requiring external extensions. **TCPDF** project (http://www.tcpdf.org) has been originally derived in 2002 from the Public Domain FPDF class by Olivier Plathey (http://www.fpdf.org), but now is almost entirely rewritten.

Constructor & Destructor Documentation

```
TCPDF::__construct ( $ orientation = 'P',
                     $ unit = 'mm',
                     $ format = 'A4',
                     $ unicode = true,
                     $ encoding = 'UTF-8',
                     $ diskcache = false,
                     $ pdfa = false
                   )
```

This is the class constructor. It allows to set up the page format, the orientation and the measure unit used in all the methods (except for the font sizes).

IMPORTANT: Please note that this method sets the mb_internal_encoding to ASCII, so if you are using the mbstring module functions with **TCPDF** you need to correctly set/unset the mb_internal_encoding when needed.

Parameters:

$orientation (string) page orientation. Possible values are (case insensitive):

- P or Portrait (default)
- L or Landscape
- '' (empty string) for automatic orientation

$unit (string) User measure unit. Possible values are:

- pt: point
- mm: millimeter (default)
- cm: centimeter
- in: inch

A point equals 1/72 of inch, that is to say about 0.35 mm (an inch being 2.54 cm). This is a very common unit in typography; font sizes are expressed in that unit.

$format (mixed) The format used for pages. It can be either: one of the string values specified at getPageSizeFromFormat() or an array of parameters specified at **setPageFormat()**.

$unicode (boolean) TRUE means that the input text is unicode (default = true)

$encoding (string) Charset encoding (used only when converting back html entities); default is UTF-8.

$diskcache (boolean) If TRUE reduce the RAM memory usage by caching temporary data on filesystem (slower).

$pdfa (boolean) If TRUE set the document to PDF/A mode.

See also:

getPageSizeFromFormat(), **setPageFormat()**

```
TCPDF::__destruct ( )
```

Default destructor.

Since:

1.53.0.TC016

Member Function Documentation

```
TCPDF::_addfield ( $ type,
                   $ name,
                   $ x,
                   $ y,
                   $ w,
                   $ h,
                   $ prop
                 )   [protected]
```

Adds a javascript form field.

Parameters:

$type (string) field type

$name (string) field name

$x (int) horizontal position

$y (int) vertical position

$w (int) width

$h (int) height

$prop (array) javascript field properties. Possible values are described on official Javascript for Acrobat API reference.

Author:

Denis Van Nuffelen, Nicola Asuni

Since:

2.1.002 (2008-02-12)

```
TCPDF::_beginpage ( $ orientation = '',
                    $ format = ''
                  )   [protected]
```

Initialize a new page.

Parameters:

$orientation (string) page orientation. Possible values are (case insensitive):

- P or PORTRAIT (default)
- L or LANDSCAPE

$format (mixed) The format used for pages. It can be either: one of the string values specified at getPageSizeFromFormat() or an array of parameters specified at **setPageFormat()**.

See also:

getPageSizeFromFormat(), **setPageFormat()**

```
TCPDF::_datastring ( $ s,
```

$ **n** = 0

) [protected]

Format a data string for meta information

Parameters:

$s (string) data string to escape.
$n (int) object ID

Returns:

string escaped string.

TCPDF::_datestring ($ **n** = 0,
$ **timestamp** = 0

) [protected]

Returns a formatted date for meta information

Parameters:

$n (int) Object ID.
$timestamp (int) Timestamp to convert.

Returns:

string escaped date string.

Since:

4.6.028 (2009-08-25)

TCPDF::_destroy ($ **destroyall** = false,
$ **preserve_objcopy** = false
)

Unset all class variables except the following critical variables.

Parameters:

$destroyall (boolean) if true destroys all class variables, otherwise preserves critical variables.
$preserve_objcopy (boolean) if true preserves the objcopy variable

Since:

4.5.016 (2009-02-24)

TCPDF::_dochecks ( ) [protected]

Check for locale-related bug

TCPDF::_dolinethrough ($ **x**,
$ **y**,
$ **txt**
) [protected]

Line through text.

Parameters:

$x (int) X coordinate
$y (int) Y coordinate
$txt (string) text to linethrough

TCPDF::_dolinethroughw ($ **x**,
$ **y**,
$ **w**
) [protected]

Line through for rectangular text area.

Parameters:

$x (int) X coordinate
$y (int) Y coordinate

$w (int) line length (width)

Since:
4.9.008 (2009-09-29)

TCPDF::_dooverline ($ **x,**
$ **y,**
$ **txt**
) [protected]

Overline text.

Parameters:
$x (int) X coordinate
$y (int) Y coordinate
$txt (string) text to overline

Since:
4.9.015 (2010-04-19)

TCPDF::_dooverlinew ($ **x,**
$ **y,**
$ **w**
) [protected]

Overline for rectangular text area.

Parameters:
$x (int) X coordinate
$y (int) Y coordinate

$w (int) width to overline

Since:
4.9.015 (2010-04-19)

TCPDF::_dounderline ($ **x,**
$ **y,**
$ **txt**
) [protected]

Underline text.

Parameters:
$x (int) X coordinate
$y (int) Y coordinate
$txt (string) text to underline

TCPDF::_dounderlinew ($ **x,**
$ **y,**
$ **w**
) [protected]

Underline for rectangular text area.

Parameters:
$x (int) X coordinate
$y (int) Y coordinate
$w (int) width to underline

Since:
4.8.008 (2009-09-29)

TCPDF::_encrypt_data ($ **n,**
$ **s**
) [protected]

Encrypt the input string.

Parameters:

$n (int) object number
$s (string) data string to encrypt

Returns:

encrypted string

Author:

Nicola Asuni

Since:

5.0.005 (2010-05-11)

TCPDF::_enddoc ( ) [protected]

Output end of document (EOF).

TCPDF::_endpage ( ) [protected]

Mark end of page.

TCPDF::_escapetext ($ s) [protected]

THIS METHOD IS DEPRECATED Format a text string

Parameters:

$s (string) string to escape.

Returns:

string escaped string.

Deprecated:

TCPDF::_fixAES256Password ($ password) [protected]

Convert password for AES-256 encryption mode

Parameters:

$password (string) password

Returns:

string password

Since:

5.9.006 (2010-10-19)

Author:

Nicola Asuni

TCPDF::_generateencryptionkey ( ) [protected]

Compute encryption key

Since:

2.0.000 (2008-01-02)

Author:

Nicola Asuni

TCPDF::_getannotsrefs ($ n) [protected]

Get references to page annotations.

Parameters:

$n (int) page number

Returns:

string

Author:
Nicola Asuni

Since:
5.0.010 (2010-05-17)

TCPDF::_getobj ($ objid = '') [protected]

Return the starting object string for the selected object ID.

Parameters:
$objid (int) Object ID (leave empty to get a new ID).

Returns:
string the starting object string

Since:
5.8.009 (2010-08-20)

TCPDF::_getrawstream ($ s, $ n = 0) [protected]

get raw output stream.

Parameters:
$s (string) string to output.
$n (int) object reference for encryption mode

Author:
Nicola Asuni

Since:
5.5.000 (2010-06-22)

TCPDF::_getstream ($ s, $ n = 0) [protected]

Format output stream (DEPRECATED).

Parameters:
$s (string) string to output.
$n (int) object reference for encryption mode

Deprecated:

TCPDF::_getxobjectdict ( ) [protected]

Return XObjects Dictionary.

Returns:
string XObjects dictionary

Since:
5.8.014 (2010-08-23)

TCPDF::_newobj ( ) [protected]

Begin a new object and return the object number.

Returns:
int object number

TCPDF::_objectkey ($ n) [protected]

Compute encryption key depending on object number where the encrypted data is stored. This is used for all strings and streams without crypt filter specifier.

Parameters:

$n (int) object number

Returns:
int object key

Author:
Nicola Asuni

Since:
2.0.000 (2008-01-02)

TCPDF::_OEvalue ( ) [protected]

Compute OE value (used for encryption)

Returns:
string OE value

Since:
5.9.006 (2010-10-19)

Author:
Nicola Asuni

TCPDF::_out ($ s) [protected]

Output a string to the document.

Parameters:
$s (string) string to output.

TCPDF::_outCurve ($ x1, $ y1, $ x2, $ y2, $ x3, $ y3) [protected]

Append a cubic Bezier curve to the current path. The curve shall extend from the current point to the point (x3, y3), using (x1, y1) and (x2, y2) as the Bezier control points. The new current point shall be (x3, y3).

Parameters:
$x1 (float) Abscissa of control point 1.
$y1 (float) Ordinate of control point 1.
$x2 (float) Abscissa of control point 2.
$y2 (float) Ordinate of control point 2.
$x3 (float) Abscissa of end point.
$y3 (float) Ordinate of end point.

Since:
2.1.000 (2008-01-08)

TCPDF::_outCurveV ($ x2, $ y2, $ x3, $ y3) [protected]

Append a cubic Bezier curve to the current path. The curve shall extend from the current point to the point (x3, y3), using the current point and (x2, y2) as the Bezier control points. The new current point shall be (x3, y3).

Parameters:
$x2 (float) Abscissa of control point 2.
$y2 (float) Ordinate of control point 2.
$x3 (float) Abscissa of end point.
$y3 (float) Ordinate of end point.

Since:

4.9.019 (2010-04-26)

TCPDF::_outCurveY ($ **x1**,
$ **y1**,
$ **x3**,
$ **y3**
) [protected]

Append a cubic Bezier curve to the current path. The curve shall extend from the current point to the point (x3, y3), using (x1, y1) and (x3, y3) as the Bezier control points. The new current point shall be (x3, y3).

Parameters:

$x1 (float) Abscissa of control point 1.
$y1 (float) Ordinate of control point 1.
$x3 (float) Abscissa of end point.
$y3 (float) Ordinate of end point.

Since:

2.1.000 (2008-01-08)

TCPDF::_outellipticalarc ($ **xc**,
$ **yc**,
$ **rx**,
$ **ry**,
$ **xang** = `0`,
$ **angs** = `0`,
$ **angf** = `360`,
$ **pie** = `false`,
$ **nc** = `2`,
$ **startpoint** = `true`,
$ **ccw** = `true`,
$ **svg** = `false`
) [protected]

Append an elliptical arc to the current path. An ellipse is formed from n Bezier curves.

Parameters:

$xc (float) Abscissa of center point.
$yc (float) Ordinate of center point.
$rx (float) Horizontal radius.
$ry (float) Vertical radius (if ry = 0 then is a circle, see **Circle()**). Default value: 0.
$xang,: (float) Angle between the X-axis and the major axis of the ellipse. Default value: 0.
$angs,: (float) Angle start of draw line. Default value: 0.
$angf,: (float) Angle finish of draw line. Default value: 360.
$pie (boolean) if true do not mark the border point (used to draw pie sectors).
$nc (integer) Number of curves used to draw a 90 degrees portion of ellipse.
$startpoint (boolean) if true output a starting point.
$ccw (boolean) if true draws in counter-clockwise.
$svg (boolean) if true the angles are in svg mode (already calculated).

Returns:

array bounding box coordinates (x min, y min, x max, y max)

Author:

Nicola Asuni

Since:

4.9.019 (2010-04-26)

TCPDF::_outLine ($ **x**,
$ **y**
) [protected]

Append a straight line segment from the current point to the point (x, y). The new current point shall be (x, y).

Parameters:

$x (float) Abscissa of end point.
$y (float) Ordinate of end point.

Since:
2.1.000 (2008-01-08)

TCPDF::_outPoint ($ x,
$ y
) [protected]

Begin a new subpath by moving the current point to coordinates (x, y), omitting any connecting line segment.

Parameters:
$x (float) Abscissa of point.
$y (float) Ordinate of point.

Since:
2.1.000 (2008-01-08)

TCPDF::_outRect ($ x,
$ y,
$ w,
$ h,
$ op
) [protected]

Append a rectangle to the current path as a complete subpath, with lower-left corner (x, y) and dimensions widthand height in user space.

Parameters:

$x	(float) Abscissa of upper-left corner.
$y	(float) Ordinate of upper-left corner.
$w	(float) Width.
$h	(float) Height.
$op	(string) options

Since:
2.1.000 (2008-01-08)

TCPDF::_outRestoreGraphicsState ( ) [protected]

Outputs the "restore graphics state" operator 'Q'

TCPDF::_outSaveGraphicsState ( ) [protected]

Outputs the "save graphics state" operator 'q'

TCPDF::_Ovalue ( ) [protected]

Compute O value (used for encryption)

Returns:
string O value

Since:
2.0.000 (2008-01-02)

Author:
Nicola Asuni

TCPDF::_putannotsobjs ( ) [protected]

Output annotations objects for all pages. !!! THIS METHOD IS NOT YET COMPLETED !!! See section 12.5 of PDF 32000_2008 reference.

Author:
Nicola Asuni

Since:
4.0.018 (2008-08-06)

TCPDF::_putannotsrefs ($ n) [protected]

Output references to page annotations

Parameters:

$n (int) page number

Author:

Nicola Asuni

Since:

4.7.000 (2008-08-29)

Deprecated:

TCPDF::_putAPXObject ($ w = 0, $ h = 0, $ stream = '') [protected]

Put appearance streams XObject used to define annotation's appearance states.

Parameters:

$w	(int) annotation width
$h	(int) annotation height
$stream	(string) appearance stream

Returns:

int object ID

Since:

4.8.001 (2009-09-09)

TCPDF::_putbookmarks ( ) [protected]

Create a bookmark PDF string.

Author:

Olivier Plathey, Nicola Asuni

Since:

2.1.002 (2008-02-12)

TCPDF::_putcatalog ( ) [protected]

Output Catalog.

Returns:

int object id

TCPDF::_putcidfont0 ($ font) [protected]

Output CID-0 fonts. A Type 0 CIDFont contains glyph descriptions based on the Adobe Type 1 font format

Parameters:

$font (array) font data

Author:

Andrew Whitehead, Nicola Asuni, Yukihiro Nakadaira

Since:

3.2.000 (2008-06-23)

TCPDF::_putdests ( ) [protected]

Insert Named Destinations.

Author:

Johannes G, Nicola Asuni

Since:

5.9.098 (2011-06-23)

TCPDF::_putEmbeddedFiles ( ) [protected]

Embedd the attached files.

Since:
4.4.000 (2008-12-07)

See also:
Annotation()

TCPDF::_putencryption ( ) [protected]

Put encryption on PDF document.

Author:
Nicola Asuni

Since:
2.0.000 (2008-01-02)

TCPDF::_putextgstates ( ) [protected]

Put extgstates for object transparency

Since:
3.0.000 (2008-03-27)

TCPDF::_putfonts ( ) [protected]

Output fonts.

Author:
Nicola Asuni

TCPDF::_putheader ( ) [protected]

Output PDF File Header (7.5.2).

TCPDF::_putimages ( ) [protected]

Output images.

TCPDF::_putinfo ( ) [protected]

Adds some Metadata information (Document Information Dictionary) (see Chapter 14.3.3 Document Information Dictionary of PDF32000_2008.pdf Reference)

Returns:
int object id

TCPDF::_putjavascript ( ) [protected]

Create a javascript PDF string.

Author:
Johannes G, Nicola Asuni

Since:
2.1.002 (2008-02-12)

TCPDF::_putocg ( ) [protected]

Put pdf layers.

Since:
3.0.000 (2008-03-27)

TCPDF::_putpages ( ) `[protected]`

Output pages (and replace page number aliases).

TCPDF::_putresourcedict ( ) `[protected]`

Output Resources Dictionary.

TCPDF::_putresources ( ) `[protected]`

Output Resources.

TCPDF::_putshaders ( ) `[protected]`

Output gradient shaders.

Author:
Nicola Asuni

Since:
3.1.000 (2008-06-09)

TCPDF::_putsignature ( ) `[protected]`

Add certification signature (DocMDP or UR3) You can set only one signature type

Author:
Nicola Asuni

Since:
4.6.008 (2009-05-07)

TCPDF::_putspotcolors ( ) `[protected]`

Output Spot Colors Resources.

Since:
4.0.024 (2008-09-12)

TCPDF::_putstream ($ s, $ n = 0) `[protected]`

Output a stream (DEPRECATED).

Parameters:
- $s (string) string to output.
- $n (int) object reference for encryption mode

Deprecated:

TCPDF::_puttruetypeunicode ($ font) `[protected]`

Adds unicode fonts.
Based on PDF Reference 1.3 (section 5)

Parameters:
- $font (array) font data

Author:
Nicola Asuni

Since:
1.52.0.TC005 (2005-01-05)

TCPDF::_putviewerpreferences ( ) `[protected]`

Output viewer preferences.

Returns:
string for viewer preferences

Author:
Nicola asuni

Since:
3.1.000 (2008-06-09)

TCPDF::_putXMP ( ) [protected]

Put XMP data object and return ID.

Returns:
(int) The object ID.

Since:
5.9.121 (2011-09-28)

TCPDF::_putxobjects ( ) [protected]

Output Form XObjects Templates.

Author:
Nicola Asuni

Since:
5.8.017 (2010-08-24)

See also:
startTemplate(), endTemplate(), printTemplate()

TCPDF::_textstring ($ s, $ n = 0) [protected]

Format a text string for meta information

Parameters:
$s (string) string to escape.
$n (int) object ID

Returns:
string escaped string.

TCPDF::_UEvalue ( ) [protected]

Compute UE value (used for encryption)

Returns:
string UE value

Since:
5.9.006 (2010-10-19)

Author:
Nicola Asuni

TCPDF::_Uvalue ( ) [protected]

Compute U value (used for encryption)

Returns:
string U value

Since:
2.0.000 (2008-01-02)

Author:
Nicola Asuni

TCPDF::AcceptPageBreak ( )

Whenever a page break condition is met, the method is called, and the break is issued or not depending on the returned value. The default implementation returns a value according to the mode selected by **SetAutoPageBreak()**.
This method is called automatically and should not be called directly by the application.

Returns:
boolean

Since:
1.4

See also:
SetAutoPageBreak()

TCPDF::addEmptySignatureAppearance

```
TCPDF::addEmptySignatureAppearance ( $ x = 0,
                                     $ y = 0,
                                     $ w = 0,
                                     $ h = 0,
                                     $ page = -1,
                                     $ name = ''
                                   )
```

Add an empty digital signature appearance (a cliccable rectangle area to get signature properties)

Parameters:

$x	(float) Abscissa of the upper-left corner.
$y	(float) Ordinate of the upper-left corner.
$w	(float) Width of the signature area.
$h	(float) Height of the signature area.
$page	(int) option page number (if < 0 the current page is used).
$name	(string) Name of the signature.

Author:
Nicola Asuni

Since:
5.9.101 (2011-07-06)

TCPDF::addExtGState ($ parms) [protected]

Add transparency parameters to the current extgstate

Parameters:

$parms	(array) parameters

Returns:
the number of extgstates

Since:
3.0.000 (2008-03-27)

TCPDF::AddFont

```
TCPDF::AddFont ( $ family,
                 $ style = '',
                 $ fontfile = '',
                 $ subset = 'default'
               )
```

Imports a TrueType, Type1, core, or CID0 font and makes it available. It is necessary to generate a font definition file first (read /fonts/utils/README.TXT). The definition file (and the font file itself when embedding) must be present either in the current directory or in the one indicated by K_PATH_FONTS if the constant is defined. If it could not be found, the error "Could not include font definition file" is generated.

Parameters:

$family	(string) Font family. The name can be chosen arbitrarily. If it is a standard family name, it will override the corresponding font.
$style	(string) Font style. Possible values are (case insensitive): • empty string: regular (default) • B: bold • I: italic • BI or IB: bold italic

$fontfile (string) The font definition file. By default, the name is built from the family and style, in lower case with no spaces.

Returns:

array containing the font data, or false in case of error.

Parameters:

$subset (mixed) if true embedd only a subset of the font (stores only the information related to the used characters); if false embedd full font; if 'default' uses the default value set using **setFontSubsetting()**. This option is valid only for TrueTypeUnicode fonts. If you want to enable users to change the document, set this parameter to false. If you subset the font, the person who receives your PDF would need to have your same font in order to make changes to your PDF. The file size of the PDF would also be smaller because you are embedding only part of a font.

Since:

1.5

See also:

SetFont(), **setFontSubsetting()**

```
TCPDF::addHtmlLink ( $ url,
                     $ name,
                     $ fill = false,
                     $ firstline = false,
                     $ color = '',
                     $ style = -1,
                     $ firstblock = false
                   )
```

Output anchor link.

Parameters:

$url (string) link URL or internal link (i.e.: <a href="#23,4.5">link to page 23 at 4.5 Y position</a>)

$name (string) link name

$fill (boolean) Indicates if the cell background must be painted (true) or transparent (false).

$firstline (boolean) if true prints only the first line and return the remaining string.

$color (array) array of RGB text color

$style (string) font style (U, D, B, I)

$firstblock (boolean) if true the string is the starting of a line.

Returns:

the number of cells used or the remaining text if $firstline = true;

```
TCPDF::addHTMLTOC ( $ page = '',
                    $ toc_name = 'TOC',
                    $ templates = array(),
                    $ correct_align = true,
                    $ style = '',
                    $ color = array(0,0,0)
                  )
```

Output a Table Of Content Index (TOC) using HTML templates. This method must be called after all Bookmarks were set. Before calling this method you have to open the page using the **addTOCPage()** method. After calling this method you have to call **endTOCPage()** to close the TOC page.

Parameters:

$page (int) page number where this TOC should be inserted (leave empty for current page).

$toc_name (string) name to use for TOC bookmark.

$templates (array) array of html templates. Use: "#TOC_DESCRIPTION#" for bookmark title, "#TOC_PAGE_NUMBER#" for page number.

$correct_align (boolean) if true correct the number alignment (numbers must be in monospaced font like courier and right aligned on LTR, or left aligned on RTL)

$style (string) Font style for title: B = Bold, I = Italic, BI = Bold + Italic.

$color (array) RGB color array for title (values from 0 to 255).

Author:

Nicola Asuni

Since:

5.0.001 (2010-05-06)

See also:

addTOCPage(), **endTOCPage()**, **addTOC()**

TCPDF::addHTMLVertSpace ($ hbz = 0,
$ hb = 0,
$ cell = false,
$ firsttag = false,
$ lasttag = false
) [protected]

Add vertical spaces if needed.

Parameters:

$hbz (string) Distance between current y and line bottom.
$hb (string) The height of the break.
$cell (boolean) if true add the default left (or right if RTL) padding to each new line (default false).
$firsttag (boolean) set to true when the tag is the first.
$lasttag (boolean) set to true when the tag is the last.

TCPDF::addJavascriptObject ($ script,
$ onload = false
)

Adds a javascript object and return object ID

Parameters:

$script (string) Javascript code
$onload (boolean) if true executes this object when opening the document

Returns:

int internal object ID

Author:

Nicola Asuni

Since:

4.8.000 (2009-09-07)

TCPDF::AddLink ( )

Creates a new internal link and returns its identifier. An internal link is a clickable area which directs to another place within the document. The identifier can then be passed to **Cell()**, **Write()**, **Image()** or **Link()**. The destination is defined with **SetLink()**.

Since:

1.5

See also:

Cell(), **Write()**, **Image()**, **Link()**, **SetLink()**

TCPDF::AddPage ($ orientation = '',
$ format = '',
$ keepmargins = false,
$ tocpage = false
)

Adds a new page to the document. If a page is already present, the **Footer()** method is called first to output the footer (if enabled). Then the page is added, the current position set to the top-left corner according to the left and top margins (or top-right if in RTL mode), and **Header()** is called to display the header (if enabled). The origin of the coordinate system is at the top-left corner (or top-right for RTL) and increasing ordinates go downwards.

Parameters:

$orientation (string) page orientation. Possible values are (case insensitive):

- P or PORTRAIT (default)
- L or LANDSCAPE

$format (mixed) The format used for pages. It can be either: one of the string values specified at getPageSizeFromFormat() or an array of parameters specified at **setPageFormat()**.
$keepmargins (boolean) if true overwrites the default page margins with the current margins
$tocpage (boolean) if true set the tocpage state to true (the added page will be used to display Table Of Content).

Since:

1.0

See also:
startPage(), **endPage()**, **addTOCPage()**, **endTOCPage()**, getPageSizeFromFormat(), **setPageFormat()**

TCPDF::addPageRegion ($ region)

Add a single no-write region on selected page. A no-write region is a portion of the page with a rectangular or trapezium shape that will not be covered when writing text or html code. A region is always aligned on the left or right side of the page ad is defined using a vertical segment. You can set multiple regions for the same page.

Parameters:

$region	(array) array of a single no-write region array: ('page' => page number or empy for current page, 'xt' => X top, 'yt' => Y top, 'xb' => X bottom, 'yb' => Y bottom, 'side' => page side 'L' = left or 'R' = right).

Author:
Nicola Asuni

Since:
5.9.003 (2010-10-13)

See also:
setPageRegions(), **getPageRegions()**

TCPDF::AddSpotColor ($ name,
$ c,
$ m,
$ y,
$ k
)

Defines a new spot color. It can be expressed in RGB components or gray scale. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$name	(string) Full name of the spot color.
$c	(float) Cyan color for CMYK. Value between 0 and 100.
$m	(float) Magenta color for CMYK. Value between 0 and 100.
$y	(float) Yellow color for CMYK. Value between 0 and 100.
$k	(float) Key (Black) color for CMYK. Value between 0 and 100.

Since:
4.0.024 (2008-09-12)

See also:
SetDrawSpotColor(), **SetFillSpotColor()**, **SetTextSpotColor()**

TCPDF::addTOC ($ page = '',
$ numbersfont = '',
$ filler = '.',
$ toc_name = 'TOC',
$ style = '',
$ color = array(0,0,0)
)

Output a Table of Content Index (TOC). This method must be called after all Bookmarks were set. Before calling this method you have to open the page using the **addTOCPage()** method. After calling this method you have to call **endTOCPage()** to close the TOC page. You can override this method to achieve different styles.

Parameters:

$page	(int) page number where this TOC should be inserted (leave empty for current page).
$numbersfont	(string) set the font for page numbers (please use monospaced font for better alignment).
$filler	(string) string used to fill the space between text and page number.
$toc_name	(string) name to use for TOC bookmark.
$style	(string) Font style for title: B = Bold, I = Italic, BI = Bold + Italic.
$color	(array) RGB color array for bookmark title (values from 0 to 255).

Author:
Nicola Asuni

Since:
4.5.000 (2009-01-02)

See also:
addTOCPage(), endTOCPage(), addHTMLTOC()

```
TCPDF::addTOCPage ( $ orientation = '',
                    $ format = '',
                    $ keepmargins = false
                  )
```

Adds a new TOC (Table Of Content) page to the document.

Parameters:

$orientation	(string) page orientation.
$format	(mixed) The format used for pages. It can be either: one of the string values specified at getPageSizeFromFormat() or an array of parameters specified at setPageFormat().
$keepmargins	(boolean) if true overwrites the default page margins with the current margins

Since:
5.0.001 (2010-05-06)

See also:
AddPage(), startPage(), endPage(), endTOCPage()

```
TCPDF::addTTFfont ( $ fontfile,
                    $ fonttype = '',
                    $ enc = '',
                    $ flags = 32,
                    $ outpath = '',
                    $ platid = 3,
                    $ encid = 1,
                    $ addcbbox = false
                  )
```

Convert and add the selected TrueType or Type1 font to the fonts folder (that must be writeable).

Parameters:

$fontfile	(string) Font file (full path).
$fonttype	(string) Font type. Leave empty for autodetect mode. Valid values are: TrueTypeUnicode, TrueType, Type1, CID0JP = CID-0 Japanese, CID0KR = CID-0 Korean, CID0CS = CID-0 Chinese Simplified, CID0CT = CID-0 Chinese Traditional.
$enc	(string) Name of the encoding table to use. Leave empty for default mode. Omit this parameter for TrueType Unicode and symbolic fonts like Symbol or ZapfDingBats.
$flags	(int) Unsigned 32-bit integer containing flags specifying various characteristics of the font (PDF32000:2008 - 9.8.2 Font Descriptor Flags): +1 for fixed font; +4 for symbol or +32 for non-symbol; +64 for italic. Fixed and Italic mode are generally autodetected so you have to set it to 32 = non-symbolic font (default) or 4 = symbolic font.
$outpath	(string) Output path for generated font files (must be writeable by the web server). Leave empty for default font folder.
$platid	(int) Platform ID for CMAP table to extract (when building a Unicode font for Windows this value should be 3, for Macintosh should be 1).
$encid	(int) Encoding ID for CMAP table to extract (when building a Unicode font for Windows this value should be 1, for Macintosh should be 0). When Platform ID is 3, legal values for Encoding ID are: 0=Symbol, 1=Unicode, 2=ShiftJIS, 3=PRC, 4=Big5, 5=Wansung, 6=Johab, 7=Reserved, 8=Reserved, 9=Reserved, 10=UCS-4.
$addcbbox	(boolean) If true includes the character bounding box information on the php font file.

Returns:
(string) TCPDF font name.

Author:
Nicola Asuni

Since:
5.9.123 (2010-09-30)

Deprecated:

```
TCPDF::adjustCellPadding ( $ brd = 0 ) [protected]
```

Adjust the internal Cell padding array to take account of the line width.

Parameters:

$brd	(mixed) Indicates if borders must be drawn around the cell. The value can be a number: - 0: no border (default) - 1: frame or a string containing some or all of the following characters (in any order): - L: left - T: top - R: right - B: bottom or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

Returns:
array of adjustments

Since:
5.9.000 (2010-10-03)

```
TCPDF::Annotation ( $ x,
                    $ y,
                    $ w,
                    $ h,
                    $ text,
                    $ opt = array('Subtype'=>'Text'),
                    $ spaces = 0
                  )
```

Puts a markup annotation on a rectangular area of the page. !!!!THE ANNOTATION SUPPORT IS NOT YET FULLY IMPLEMENTED !!!!

Parameters:

$x	(float) Abscissa of the upper-left corner of the rectangle
$y	(float) Ordinate of the upper-left corner of the rectangle
$w	(float) Width of the rectangle
$h	(float) Height of the rectangle
$text	(string) annotation text or alternate content
$opt	(array) array of options (see section 8.4 of PDF reference 1.7).
$spaces	(int) number of spaces on the text to link

Since:
4.0.018 (2008-08-06)

```
TCPDF::Arrow ( $ x0,
               $ y0,
               $ x1,
               $ y1,
               $ head_style = 0,
               $ arm_size = 5,
               $ arm_angle = 15
             )
```

Draws a grahic arrow.

Parameters:

$x0	(float) Abscissa of first point.
$y0	(float) Ordinate of first point.
$x1	(float) Abscissa of second point.
$y1	(float) Ordinate of second point.
$head_style	(int) (0 = draw only arrowhead arms, 1 = draw closed arrowhead, but no fill, 2 = closed and filled arrowhead, 3 = filled arrowhead)
$arm_size	(float) length of arrowhead arms
$arm_angle	(int) angle between an arm and the shaft

Author:
Piotr Galecki, Nicola Asuni, Andy Meier

Since:
4.6.018 (2009-07-10)

```
TCPDF::Bookmark ( $ txt,
                  $ level = 0,
                  $ y = -1,
                  $ page = '',
                  $ style = '',
                  $ color = array(0,0,0),
                  $ x = -1,
                  $ link = ''
                )
```

Adds a bookmark.

Parameters:

$txt	(string) Bookmark description.
$level	(int) Bookmark level (minimum value is 0).
$y	(float) Y position in user units of the bookmark on the selected page (default = -1 = current position; 0 = page start;).
$page	(int) Target page number (leave empty for current page).
$style	(string) Font style: B = Bold, I = Italic, BI = Bold + Italic.
$color	(array) RGB color array (values from 0 to 255).
$x	(float) X position in user units of the bookmark on the selected page (default = -1 = current position;).
$link	(mixed) URL, or numerical link ID, or named destination (# character followed by the destination name), or embedded file (* character followed by the file name).

Since:

2.1.002 (2008-02-12)

```
TCPDF::Button ( $ name,
                $ w,
                $ h,
                $ caption,
                $ action,
                $ prop = array(),
                $ opt = array(),
                $ x = '',
                $ y = '',
                $ js = false
              )
```

Creates a button field

Parameters:

$name	(string) field name
$w	(int) width
$h	(int) height
$caption	(string) caption.
$action	(mixed) action triggered by pressing the button. Use a string to specify a javascript action. Use an array to specify a form action options as on section 12.7.5 of PDF32000_2008.
$prop	(array) javascript field properties. Possible values are described on official Javascript for Acrobat API reference.
$opt	(array) annotation parameters. Possible values are described on official PDF32000_2008 reference.
$x	(float) Abscissa of the upper-left corner of the rectangle
$y	(float) Ordinate of the upper-left corner of the rectangle
$js	(boolean) if true put the field using JavaScript (requires Acrobat Writer to be rendered).

Author:

Nicola Asuni

Since:

4.8.000 (2009-09-07)

```
TCPDF::Cell ( $ w,
              $ h = 0,
              $ txt = '',
              $ border = 0,
              $ ln = 0,
              $ align = '',
              $ fill = false,
              $ link = '',
              $ stretch = 0,
              $ ignore_min_height = false,
              $ calign = 'T',
              $ valign = 'M'
          )
```

Prints a cell (rectangular area) with optional borders, background color and character string. The upper-left corner of the cell corresponds to the current position. The text can be aligned or centered. After the call, the current position moves to the right or to the next line. It is possible to put a link on the text. If automatic page breaking is enabled and the cell goes beyond the limit, a page break is done before outputting.

Parameters:

$w (float) Cell width. If 0, the cell extends up to the right margin.

$h (float) Cell height. Default value: 0.

$txt (string) String to print. Default value: empty string.

$border (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top
- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

$ln (int) Indicates where the current position should go after the call. Possible values are:

- 0: to the right (or left for RTL languages)
- 1: to the beginning of the next line
- 2: below

Putting 1 is equivalent to putting 0 and calling **Ln()** just after. Default value: 0.

$align (string) Allows to center or align the text. Possible values are:

- L or empty string: left align (default value)
- C: center
- R: right align
- J: justify

$fill (boolean) Indicates if the cell background must be painted (true) or transparent (false).

$link (mixed) URL or identifier returned by **AddLink()**.

$stretch (int) font stretch mode:

- 0 = disabled
- 1 = horizontal scaling only if text is larger than cell width
- 2 = forced horizontal scaling to fit cell width
- 3 = character spacing only if text is larger than cell width
- 4 = forced character spacing to fit cell width

General font stretching and scaling values will be preserved when possible.

$ignore_min_height (boolean) if true ignore automatic minimum height value.

$calign (string) cell vertical alignment relative to the specified Y value. Possible values are:

- T : cell top
- C : center
- B : cell bottom
- A : font top
- L : font baseline
- D : font bottom

$valign (string) text vertical alignment inside the cell. Possible values are:

- T : top
- C : center
- B : bottom

Since:

1.0

See also:

SetFont(), SetDrawColor(), SetFillColor(), SetTextColor(), SetLineWidth(), AddLink(), Ln(), MultiCell(), Write(), SetAutoPageBreak()

```
TCPDF::CheckBox ( $ name,
                  $ w,
                  $ checked = false,
                  $ prop = array(),
                  $ opt = array(),
                  $ onvalue = 'Yes',
                  $ x = '',
                  $ y = '',
                  $ js = false
                )
```

Creates a CheckBox field

Parameters:

$name (string) field name
$w (int) width
$checked (boolean) define the initial state.
$prop (array) javascript field properties. Possible values are described on official Javascript for Acrobat API reference.
$opt (array) annotation parameters. Possible values are described on official PDF32000_2008 reference.
$onvalue (string) value to be returned if selected.
$x (float) Abscissa of the upper-left corner of the rectangle
$y (float) Ordinate of the upper-left corner of the rectangle
$js (boolean) if true put the field using JavaScript (requires Acrobat Writer to be rendered).

Author:

Nicola Asuni

Since:

4.8.000 (2009-09-07)

```
TCPDF::checkPageBreak ( $ h = 0,
                        $ y = '',
                        $ addpage = true

                      )   [protected]
```

Add page if needed.

Parameters:

$h (float) Cell height. Default value: 0.
$y (mixed) starting y position, leave empty for current position.
$addpage (boolean) if true add a page, otherwise only return the true/false state

Returns:

boolean true in case of page break, false otherwise.

Since:

3.2.000 (2008-07-01)

```
TCPDF::checkPageRegions ( $ h,
                          $ x,
                          $ y
                        )   [protected]
```

Check page for no-write regions and adapt current coordinates and page margins if necessary. A no-write region is a portion of the page with a rectangular or trapezium shape that will not be covered when writing text or html code. A region is always aligned on the left or right side of the page ad is defined using a vertical segment.

Parameters:

$h (float) height of the text/image/object to print in user units
$x (float) current X coordinate in user units
$y (float) current Y coordinate in user units

Returns:
array($x, $y)

Author:
Nicola Asuni

Since:
5.9.003 (2010-10-13)

```
TCPDF::Circle ( $ x0,
                $ y0,
                $ r,
                $ angstr = 0,

                $ angend = 360,
                $ style = '',
                $ line_style = array(),
                $ fill_color = array(),
                $ nc = 2
              )
```

Draws a circle. A circle is formed from n Bezier curves.

Parameters:

$x0	(float) Abscissa of center point.
$y0	(float) Ordinate of center point.
$r	(float) Radius.
$angstr,:	(float) Angle start of draw line. Default value: 0.
$angend,:	(float) Angle finish of draw line. Default value: 360.
$style	(string) Style of rendering. See the getPathPaintOperator() function for more information.
$line_style	(array) Line style of circle. Array like for **SetLineStyle()**. Default value: default line style (empty array).
$fill_color	(array) Fill color. Format: array(red, green, blue). Default value: default color (empty array).
$nc	(integer) Number of curves used to draw a 90 degrees portion of circle.

Since:
2.1.000 (2008-01-08)

```
TCPDF::Clip ( $ x,
              $ y,
              $ w,
              $ h
            )   [protected]
```

Set a rectangular clipping area.

Parameters:

$x	(float) abscissa of the top left corner of the rectangle (or top right corner for RTL mode).
$y	(float) ordinate of the top left corner of the rectangle.
$w	(float) width of the rectangle.
$h	(float) height of the rectangle.

Author:
Andreas W, Nicola Asuni

Since:
3.1.000 (2008-06-09)

```
TCPDF::Close ( )
```

Terminates the PDF document. It is not necessary to call this method explicitly because **Output()** does it automatically. If the document contains no page, **AddPage()** is called to prevent from getting an invalid document.

Since:
1.0

See also:
Open(), **Output()**

```
TCPDF::closeHTMLTagHandler ( $  dom,
                             $  key,
                             $  cell,
                             $  maxbottomliney = 0
                           )    [protected]
```

Process closing tags.

Parameters:

$dom	(array) html dom array
$key	(int) current element id
$cell	(boolean) if true add the default left (or right if RTL) padding to each new line (default false).
$maxbottomliney	(int) maximum y value of current line

Returns:
$dom array

```
TCPDF::colorRegistrationBar ( $  x,
                              $  y,
                              $  w,
                              $  h,
                              $  transition = true,
                              $  vertical = false,
                              $  colors = 'A,
                              R  ,
                              G  ,
                              B  ,
                              C  ,
                              M  ,
                              Y  ,
                              K'
                            )
```

Paints color transition registration bars

Parameters:

$x	(float) abscissa of the top left corner of the rectangle.
$y	(float) ordinate of the top left corner of the rectangle.
$w	(float) width of the rectangle.
$h	(float) height of the rectangle.
$transition	(boolean) if true prints tcolor transitions to white.
$vertical	(boolean) if true prints bar vertically.
$colors	(string) colors to print separated by comma. Valid values are: A,W,R,G,B,C,M,Y,K,RGB,CMYK,ALL,ALLSPOT,<SPOT_COLOR_NAME>. Where: A = grayscale black, W = grayscale white, R = RGB red, G RGB green, B RGB blue, C = CMYK cyan, M = CMYK magenta, Y = CMYK yellow, K = CMYK key/black, RGB = RGB registration color, CMYK = CMYK registration color, ALL = Spot registration color, ALLSPOT = print all defined spot colors, <SPOT_COLOR_NAME> = name of the spot color to print.

Author:
Nicola Asuni

Since:
4.9.000 (2010-03-26)

```
TCPDF::ComboBox ( $  name,
                  $  w,
                  $  h,
                  $  values,
                  $  prop = array(),
                  $  opt = array(),
                  $  x = '',
                  $  y = '',
                  $  js = false
                )
```

Creates a Combo-box field

Parameters:

$name (string) field name
$w (int) width
$h (int) height
$values (array) array containing the list of values.
$prop (array) javascript field properties. Possible values are described on official Javascript for Acrobat API reference.
$opt (array) annotation parameters. Possible values are described on official PDF32000_2008 reference.
$x (float) Abscissa of the upper-left corner of the rectangle
$y (float) Ordinate of the upper-left corner of the rectangle
$js (boolean) if true put the field using JavaScript (requires Acrobat Writer to be rendered).

Author:

Nicola Asuni

Since:

4.8.000 (2009-09-07)

TCPDF::commitTransaction ( )

Delete the copy of the current **TCPDF** object used for undo operation.

Since:

4.5.029 (2009-03-19)

TCPDF::convertSVGtMatrix ($ tm) `[protected]`

Convert SVG transformation matrix to PDF.

Parameters:

$tm (array) original SVG transformation matrix

Returns:

array transformation matrix

Since:

5.0.000 (2010-05-02)

TCPDF::CoonsPatchMesh ($ x,
$ y,
$ w,
$ h,
$ col1 = `array(),`
$ col2 = `array(),`
$ col3 = `array(),`
$ col4 = `array(),`
$ coords = `array(0.00,0.0,0.33,0.00,0.67,0.00,1.00,0.00,1.00,0.33,1.00,0.67,1.00,1.00,0.67,1.00,0.33,1.00,0.00`
$ coords_min = `0,`
$ coords_max = `1,`
$ antialias = `false`
)

Paints a coons patch mesh.

Parameters:

$x (float) abscissa of the top left corner of the rectangle.
$y (float) ordinate of the top left corner of the rectangle.
$w (float) width of the rectangle.
$h (float) height of the rectangle.
$col1 (array) first color (lower left corner) (RGB components).
$col2 (array) second color (lower right corner) (RGB components).
$col3 (array) third color (upper right corner) (RGB components).
$col4 (array) fourth color (upper left corner) (RGB components).
(array)

- for one patch mesh: array(float x1, float y1, ... float x12, float y12): 12 pairs of coordinates (normally from 0 to 1) which specify the Bezier control points that define the patch. First pair is the lower left edge point, next is its right control point (control point 2). Then

$coords
the other points are defined in the order: control point 1, edge point, control point 2 going counter-clockwise around the patch. Last (x12, y12) is the first edge point's left control point (control point 1).

- for two or more patch meshes: array[number of patches]: arrays with the following keys for each patch: f: where to put that patch (0 = first patch, 1, 2, 3 = right, top and left of precedent patch - I didn't figure this out completely - just try and error ;-) points: 12 pairs of coordinates of the Bezier control points as above for the first patch, 8 pairs of coordinates for the following patches, ignoring the coordinates already defined by the precedent patch (I also didn't figure out the order of these - also: try and see what's happening) colors: must be 4 colors for the first patch, 2 colors for the following patches

$coords_min (array) minimum value used by the coordinates. If a coordinate's value is smaller than this it will be cut to coords_min. default: 0
$coords_max (array) maximum value used by the coordinates. If a coordinate's value is greater than this it will be cut to coords_max. default: 1
$antialias (boolean) A flag indicating whether to filter the shading function to prevent aliasing artifacts.

Author:
Andreas W, Nicola Asuni

Since:
3.1.000 (2008-06-09)

TCPDF::copyPage ($ page = 0)

Clone the specified page to a new page.

Parameters:
$page (int) number of page to copy (0 = current page)

Returns:
true in case of success, false in case of error.

Since:
4.9.015 (2010-04-20)

```
TCPDF::cropMark ( $  x,
                  $  y,
                  $  w,
                  $  h,
                  $  type = 'T,
                  R  ,
                  B  ,
                  L' ,
                  $  color = array(100,100,100,100,'All')
                )
```

Paints crop marks.

Parameters:
$x (float) abscissa of the crop mark center.
$y (float) ordinate of the crop mark center.
$w (float) width of the crop mark.
$h (float) height of the crop mark.
$type (string) type of crop mark, one symbol per type separated by comma: T = TOP, F = BOTTOM, L = LEFT, R = RIGHT, TL = A = TOP-LEFT, TR = B = TOP-RIGHT, BL = C = BOTTOM-LEFT, BR = D = BOTTOM-RIGHT.
$color (array) crop mark color (default spot registration color).

Author:
Nicola Asuni

Since:
4.9.000 (2010-03-26)

```
TCPDF::Curve ( $  x0,
               $  y0,
               $  x1,
               $  y1,
               $  x2,
               $  y2,
               $  x3,
               $  y3,
               $  style = '',
               $ line_style = array(),
               $ fill_color = array()
              )
```

Draws a Bezier curve. The Bezier curve is a tangent to the line between the control points at either end of the curve.

Parameters:

$x0	(float) Abscissa of start point.
$y0	(float) Ordinate of start point.
$x1	(float) Abscissa of control point 1.
$y1	(float) Ordinate of control point 1.
$x2	(float) Abscissa of control point 2.
$y2	(float) Ordinate of control point 2.
$x3	(float) Abscissa of end point.
$y3	(float) Ordinate of end point.
$style	(string) Style of rendering. See the getPathPaintOperator() function for more information.
$line_style	(array) Line style of curve. Array like for **SetLineStyle()**. Default value: default line style (empty array).
$fill_color	(array) Fill color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName). Default value: default color (empty array).

See also:
SetLineStyle()

Since:
2.1.000 (2008-01-08)

```
TCPDF::deletePage ( $ page )
```

Remove the specified page.

Parameters:

$page	(int) page to remove

Returns:
true in case of success, false in case of error.

Since:
4.6.004 (2009-04-23)

```
TCPDF::drawHTMLTagBorder ( $ tag,
                           $ xmax
                         )   [protected]
```

Draw an HTML block border and fill

Parameters:

$tag	(array) array of tag properties.
$xmax	(int) end X coordinate for border.

Since:
5.7.000 (2010-08-03)

```
TCPDF::Ellipse ( $ x0,
                 $ y0,
                 $ rx,
                 $ ry = '',
                 $ angle = 0,
                 $ astart = 0,
                 $ afinish = 360,
                 $ style = '',
                 $ line_style = array(),

                 $ fill_color = array(),
                 $ nc = 2
               )
```

Draws an ellipse. An ellipse is formed from n Bezier curves.

Parameters:

$x0 (float) Abscissa of center point.

$y0 (float) Ordinate of center point.

$rx (float) Horizontal radius.

$ry (float) Vertical radius (if ry = 0 then is a circle, see **Circle()**). Default value: 0.

$angle,: (float) Angle oriented (anti-clockwise). Default value: 0.

$astart,: (float) Angle start of draw line. Default value: 0.

$afinish,: (float) Angle finish of draw line. Default value: 360.

$style (string) Style of rendering. See the getPathPaintOperator() function for more information.

$line_style (array) Line style of ellipse. Array like for **SetLineStyle()**. Default value: default line style (empty array).

$fill_color (array) Fill color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName). Default value: default color (empty array).

$nc (integer) Number of curves used to draw a 90 degrees portion of ellipse.

Author:
Nicola Asuni

Since:
2.1.000 (2008-01-08)

TCPDF::endLayer ( )

End the current PDF layer.

Since:
5.9.102 (2011-07-13)

TCPDF::endPage ($ tocpage = `false`)

Terminate the current page

Parameters:

$tocpage (boolean) if true set the tocpage state to false (end the page used to display Table Of Content).

Since:
4.2.010 (2008-11-14)

See also:
AddPage(), **startPage()**, **addTOCPage()**, **endTOCPage()**

TCPDF::endSVGElementHandler ($ parser, $ name) `[protected]`

Sets the closing SVG element handler function for the XML parser.

Parameters:

$parser (resource) The first parameter, parser, is a reference to the XML parser calling the handler.

$name (string) The second parameter, name, contains the name of the element for which this handler is called. If case-folding is in effect for this parser, the element name will be in uppercase letters.

Author:
Nicola Asuni

Since:
5.0.000 (2010-05-02)

TCPDF::endTemplate ( )

End the current XObject Template started with **startTemplate()** and restore the previous graphic state. An XObject Template is a PDF block that is a self-contained description of any sequence of graphics objects (including path objects, text objects, and sampled images). An XObject Template may be painted multiple times, either on several pages or at several locations on the same page and produces the same results each time, subject only to the graphics state at the time it is invoked.

Returns:
int the XObject Template ID in case of success or false in case of error.

Author:
Nicola Asuni

Since:
5.8.017 (2010-08-24)

See also:
startTemplate(), printTemplate()

TCPDF::endTOCPage ( )

Terminate the current TOC (Table Of Content) page

Since:
5.0.001 (2010-05-06)

See also:
AddPage(), startPage(), endPage(), addTOCPage()

TCPDF::Error ($ msg)

Throw an exception or print an error message and die if the K_TCPDF_PARSER_THROW_EXCEPTION_ERROR constant is set to true.

Parameters:

$msg	(string) The error message

Since:
1.0

TCPDF::fitBlock ($ w,
$ h,
$ x,
$ y,
$ fitonpage = false
) [protected]

Set the block dimensions accounting for page breaks and page/column fitting

Parameters:

$w	(float) width
$h	(float) height
$x	(float) X coordinate
$y	(float) Y coodiante
$fitonpage	(boolean) if true the block is resized to not exceed page dimensions.

Returns:
array($w, $h, $x, $y)

Since:
5.5.009 (2010-07-05)

TCPDF::fixHTMLCode ($ html,
$ default_css = '',
$ tagvs = '',
$ tidy_options = ''
)

Cleanup HTML code (requires HTML Tidy library).

Parameters:

$html	(string) htmlcode to fix
$default_css	(string) CSS commands to add
$tagvs	(array) parameters for setHtmlVSpace method
$tidy_options	(array) options for tidy_parse_string function

Returns:
string XHTML code cleaned up

Author:
Nicola Asuni

Since:

5.9.017 (2010-11-16)

See also:
setHtmlVSpace()

TCPDF::Footer ( )

This method is used to render the page footer. It is automatically called by **AddPage()** and could be overwritten in your own inherited class.

Reimplemented in **TOC_TCPDF**, and **MYPDF**.

TCPDF::getAbsFontMeasure ($ s)

Convert a relative font measure into absolute value.

Parameters:
$s (int) Font measure.

Returns:
float Absolute measure.

Since:
5.9.186 (2012-09-13)

TCPDF::GetAbsX ( )

Returns the absolute X value of current position.

Returns:
float

Since:
1.2

See also:
SetX(), GetY(), SetY()

TCPDF::getAliasNbPages ( )

Returns the string alias used for the total number of pages. If the current font is unicode type, the returned string is surrounded by additional curly braces. This alias will be replaced by the total number of pages in the document.

Returns:
string

Since:
4.0.018 (2008-08-08)

TCPDF::getAliasNumPage ( )

Returns the string alias used for the page number. If the current font is unicode type, the returned string is surrounded by additional curly braces. This alias will be replaced by the page number.

Returns:
string

Since:
4.5.000 (2009-01-02)

TCPDF::getAliasRightShift ( )

Returns the string alias used right align page numbers. If the current font is unicode type, the returned string wil contain an additional open curly brace.

Returns:
string

Since:
5.9.099 (2011-06-27)

TCPDF::getAllInternalPageNumberAliases ( ) `[protected]`

Return an array containing all internal page aliases.

Returns:
array of page number aliases

TCPDF::getAllSpotColors ( )

Returns the array of spot colors.

Returns:
(array) Spot colors array.

Since:
6.0.038 (2013-09-30)

TCPDF::getAlpha ( )

Get the alpha mode array (CA, ca, BM, AIS). (Check the "Entries in a Graphics State Parameter Dictionary" on PDF 32000-1:2008).

Returns:
array.

Since:
5.9.152 (2012-03-23)

TCPDF::GetArrStringWidth ($ sa,

```
$ fontname = '',
$ fontstyle = '',
$ fontsize = 0,
$ getarray = false
)
```

Returns the string length of an array of chars in user unit or an array of characters widths. A font must be selected.

Parameters:

$sa	(string) The array of chars whose total length is to be computed
$fontname	(string) Family font. It can be either a name defined by AddFont() or one of the standard families. It is also possible to pass an empty string, in that case, the current family is retained.
$fontstyle	(string) Font style. Possible values are (case insensitive): • empty string: regular • B: bold • I: italic • U: underline • D: line through • O: overline or any combination. The default value is regular.
$fontsize	(float) Font size in points. The default value is the current size.
$getarray	(boolean) if true returns an array of characters widths, if false returns the total length.

Returns:
mixed int total string length or array of characted widths

Author:
Nicola Asuni

Since:
2.4.000 (2008-03-06)

TCPDF::getAutoPageBreak ( )

Return the auto-page-break mode (true or false).

Returns:
boolean auto-page-break mode

Since:
5.9.088

TCPDF::getBarcode ( )

Get current barcode.

Returns:
string

Since:
4.0.012 (2008-07-24)

TCPDF::getBorderStartPosition ( ) [protected]

Return the starting coordinates to draw an html border

Returns:
array containing top-left border coordinates

Since:
5.7.000 (2010-08-03)

TCPDF::getBreakMargin ($ pagenum = '')

Returns the page break margin.

Parameters:
$pagenum (int) page number (empty = current page)

Returns:
int page break margin.

Author:
Nicola Asuni

Since:
1.5.2

See also:
getPageDimensions()

TCPDF::getBuffer ( ) [protected]

Get buffer content.

Returns:
string buffer content

Since:
4.5.000 (2009-01-02)

TCPDF::getCellBorder ($ x, $ y, $ w, $ h, $ brd) [protected]

```
TCPDF::getCellBorder ( $ x,
                       $ y,
                       $ w,
                       $ h,
                       $ brd
                     )   [protected]
```

Returns the code to draw the cell border

Parameters:

$x	(float) X coordinate.
$y	(float) Y coordinate.
$w	(float) Cell width.
$h	(float) Cell height.

$brd (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top

- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

Returns:

string containing cell border code

See also:

SetLineStyle()

Since:

5.7.000 (2010-08-02)

```
TCPDF::getCellCode ( $ w,
                     $ h = 0,
                     $ txt = '',
                     $ border = 0,
                     $ ln = 0,
                     $ align = '',
                     $ fill = false,
                     $ link = '',
                     $ stretch = 0,
                     $ ignore_min_height = false,
                     $ calign = 'T',
                     $ valign = 'M'
                   )  [protected]
```

Returns the PDF string code to print a cell (rectangular area) with optional borders, background color and character string. The upper-left corner of the cell corresponds to the current position. The text can be aligned or centered. After the call, the current position moves to the right or to the next line. It is possible to put a link on the text.
If automatic page breaking is enabled and the cell goes beyond the limit, a page break is done before outputting.

Parameters:

$w (float) Cell width. If 0, the cell extends up to the right margin.

$h (float) Cell height. Default value: 0.

$txt (string) String to print. Default value: empty string.

$border (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top
- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

$ln (int) Indicates where the current position should go after the call. Possible values are:

- 0: to the right (or left for RTL languages)
- 1: to the beginning of the next line
- 2: below

Putting 1 is equivalent to putting 0 and calling **Ln()** just after. Default value: 0.

$align (string) Allows to center or align the text. Possible values are:

- L or empty string: left align (default value)
- C: center
- R: right align
- J: justify

$fill (boolean) Indicates if the cell background must be painted (true) or transparent (false).

$link (mixed) URL or identifier returned by **AddLink()**.

(int) font stretch mode:

- 0 = disabled
- 1 = horizontal scaling only if text is larger than cell width
- 2 = forced horizontal scaling to fit cell width

$stretch	• 3 = character spacing only if text is larger than cell width • 4 = forced character spacing to fit cell width General font stretching and scaling values will be preserved when possible.
$ignore_min_height	(boolean) if true ignore automatic minimum height value.
$calign	(string) cell vertical alignment relative to the specified Y value. Possible values are: • T : cell top • C : center • B : cell bottom • A : font top • L : font baseline • D : font bottom
$valign	(string) text vertical alignment inside the cell. Possible values are: • T : top • M : middle • B : bottom

Returns:
string containing cell code

Since:
1.0

See also:
Cell()

TCPDF::getCellHeight ($ fontsize, $ padding = TRUE)

Return the cell height

Parameters:

$fontsize	(int) Font size in internal units
$padding	(boolean) If true add cell padding

TCPDF::getCellHeightRatio ( )

return the height of cell repect font height.

Since:
4.0.012 (2008-07-24)

TCPDF::getCellMargins ( )

Get the internal Cell margin array.

Returns:
array of margin values

Since:
5.9.000 (2010-10-03)

See also:
setCellMargins()

TCPDF::getCellPaddings ( )

Get the internal Cell padding array.

Returns:
array of padding values

Since:
5.9.000 (2010-10-03)

See also:
setCellPaddings(), **SetCellPadding()**

TCPDF::getCharBBox ($ char)

Returns the glyph bounding box of the specified character in the current font in user units.

Parameters:

$char (int) Input character code.

Returns:

mixed array(xMin, yMin, xMax, yMax) or FALSE if not defined.

Since:

5.9.186 (2012-09-13)

**TCPDF::GetCharWidth ($ char,
$ notlast = `true`
)**

Returns the length of the char in user unit for the current font considering current stretching and spacing (tracking).

Parameters:

$char	(int) The char code whose length is to be returned
$notlast	(boolean) If false ignore the font-spacing.

Returns:

float char width

Author:

Nicola Asuni

Since:

2.4.000 (2008-03-06)

TCPDF::getColumn ( )

Return the current column number

Returns:

int current column number

Since:

5.5.011 (2010-07-08)

TCPDF::getCSSBorderDashStyle ($ style) `[protected]`

Returns the border dash style from CSS property

Parameters:

$style (string) border style to convert

Returns:

int sash style (return -1 in case of none or hidden border)

Since:

5.7.000 (2010-08-02)

**TCPDF::getCSSBorderMargin ($ cssbspace,
$ width = `0`
)**

Get the border-spacing from CSS attribute.

Parameters:

$cssbspace	(string) border-spacing CSS properties
$width	(float) width of the containing element

Returns:

array of border spacings

Since:

5.9.010 (2010-10-27)

TCPDF::getCSSBorderStyle ($ cssborder) [protected]

Returns the border style array from CSS border properties

Parameters:

$cssborder (string) border properties

Returns:

array containing border properties

Since:

5.7.000 (2010-08-02)

TCPDF::getCSSBorderWidth ($ width) [protected]

Returns the border width from CSS property

Parameters:

$width (string) border width

Returns:

int with in user units

Since:

5.7.000 (2010-08-02)

TCPDF::getCSSFontSpacing ($ spacing, $ parent = 0) [protected]

Returns the letter-spacing value from CSS value

Parameters:

$spacing (string) letter-spacing value
$parent (float) font spacing (tracking) value of the parent element

Returns:

float quantity to increases or decreases the space between characters in a text.

Since:

5.9.000 (2010-10-02)

TCPDF::getCSSFontStretching ($ stretch, $ parent = 100) [protected]

Returns the percentage of font stretching from CSS value

Parameters:

$stretch (string) stretch mode
$parent (float) stretch value of the parent element

Returns:

float font stretching percentage

Since:

5.9.000 (2010-10-02)

TCPDF::getCSSMargin ($ cssmargin, $ width = 0)

Get the internal Cell margin from CSS attribute.

Parameters:

$cssmargin (string) margin properties
$width (float) width of the containing element

Returns:
array of cell margins

Since:
5.9.000 (2010-10-04)

TCPDF::getCSSPadding ($ csspadding, $ width = 0)

Get the internal Cell padding from CSS attribute.

Parameters:

$csspadding	(string) padding properties
$width	(float) width of the containing element

Returns:
array of cell paddings

Since:
5.9.000 (2010-10-04)

TCPDF::getDestination ( )

Return the Named Destination array.

Returns:
(array) Named Destination array.

Author:
Nicola Asuni

Since:
5.9.097 (2011-06-23)

TCPDF::getDocCreationTimestamp ( )

Returns document creation timestamp in seconds.

Returns:
(int) Creation timestamp in seconds.

Since:
5.9.152 (2012-03-23)

TCPDF::getDocModificationTimestamp ( )

Returns document modification timestamp in seconds.

Returns:
(int) Modfication timestamp in seconds.

Since:
5.9.152 (2012-03-23)

TCPDF::getFontAscent ($ font, $ style = '', $ size = 0)

Return the font ascent value.

Parameters:

$font	(string) font name
$style	(string) font style
$size	(float) The size (in points)

Returns:
int font ascent

Author:

Nicola Asuni

Since:
4.9.003 (2010-03-30)

TCPDF::getFontBBox ( )

Returns the bounding box of the current font in user units.

Returns:
array

Since:
5.9.152 (2012-03-23)

TCPDF::getFontBuffer ($ font) [protected]

Get font buffer content.

Parameters:
$font (string) font key

Returns:
string font buffer content or false in case of error

Since:
4.5.000 (2009-01-02)

TCPDF::getFontDescent ($ font, $ style = '', $ size = 0)

Return the font descent value

Parameters:
$font (string) font name
$style (string) font style
$size (float) The size (in points)

Returns:
int font descent

Author:
Nicola Asuni

Since:
4.9.003 (2010-03-30)

TCPDF::getFontFamily ( )

Returns the current font family name.

Returns:
string current font family name

Since:
4.3.008 (2008-12-05)

TCPDF::getFontFamilyName ($ fontfamily)

Return normalized font name

Parameters:
$fontfamily (string) property string containing font family names

Returns:
string normalized font name

Author:
Nicola Asuni

Since:
5.8.004 (2010-08-17)

TCPDF::getFontSize ( )

Returns the current font size.

Returns:
current font size

Since:
3.2.000 (2008-06-23)

TCPDF::getFontSizePt ( )

Returns the current font size in points unit.

Returns:
current font size in points unit

Since:
3.2.000 (2008-06-23)

TCPDF::getFontsList ( ) [protected]

Fill the list of available fonts ($this->fontlist).

Since:
4.0.013 (2008-07-28)

TCPDF::getFontSpacing ( )

Get the amount to increase or decrease the space between characters in a text.

Returns:
int font spacing (tracking) value

Author:
Nicola Asuni

Since:
5.9.000 (2010-09-29)

TCPDF::getFontStretching ( )

Get the percentage of character stretching.

Returns:
float stretching value

Author:
Nicola Asuni

Since:
5.9.000 (2010-09-29)

TCPDF::getFontStyle ( )

Returns the current font style.

Returns:
string current font style

Since:
4.3.008 (2008-12-05)

TCPDF::getFontSubsetting ( )

Return the default option for font subsetting.

Returns:
boolean default font subsetting state.

Author:
Nicola Asuni

Since:
5.3.002 (2010-06-07)

TCPDF::getFooterFont ( )

Get Footer font.

Returns:
array() Array describing the basic font parameters: (family, style, size).

Since:
4.0.012 (2008-07-24)

TCPDF::getFooterMargin ( )

Returns footer margin in user units.

Returns:
float

Since:
4.0.012 (2008-07-24)

TCPDF::getFormDefaultProp ( )

Return the default properties for form fields.

Returns:
array $prop javascript field properties. Possible values are described on official Javascript for Acrobat API reference.

Author:
Nicola Asuni

Since:
4.8.000 (2009-09-06)

TCPDF::getGDgamma ($ img, $ c) [protected]

Get the GD-corrected PNG gamma value from alpha color

Parameters:

$img (int) GD image Resource ID.
$c (int) alpha color

Since:
4.3.007 (2008-12-04)

TCPDF::getGraphicVars ( ) [protected]

Returns current graphic variables as array.

Returns:
array of graphic variables

Since:
4.2.010 (2008-11-14)

TCPDF::getGroupPageNo ( )

Return the current page in the group.

Returns:
current page in the group

Since:
3.0.000 (2008-03-27)

TCPDF::getGroupPageNoFormatted ( )

Returns the current group page number formatted as a string.

Since:
4.3.003 (2008-11-18)

See also:
PaneNo(), formatPageNumber()

TCPDF::getHeaderData ( )

Returns header data:

- $ret['logo'] = logo image
- $ret['logo_width'] = width of the image logo in user units
- $ret['title'] = header title
- $ret['string'] = header description string

Returns:
array()

Since:
4.0.012 (2008-07-24)

TCPDF::getHeaderFont ( )

Get header font.

Returns:
array() Array describing the basic font parameters: (family, style, size).

Since:
4.0.012 (2008-07-24)

TCPDF::getHeaderMargin ( )

Returns header margin in user units.

Returns:
float

Since:
4.0.012 (2008-07-24)

TCPDF::getHtmlDomArray ($ html) [protected]

Returns the HTML DOM array.

Parameters:
$html (string) html code

Returns:
array

Since:
3.2.000 (2008-06-20)

TCPDF::getHTMLFontUnits ($ val,
$ refsize = 12,
$ parent_size = 12,
$ defaultunit = 'pt'
)

Convert HTML string containing font size value to points

Parameters:

$val	(string) String containing font size value and unit.
$refsize	(float) Reference font size in points.
$parent_size	(float) Parent font size in points.
$defaultunit	(string) Default unit (can be one of the following: %, em, ex, px, in, mm, pc, pt).

Returns:
float value in points

TCPDF::getHTMLUnitToUnits ($ htmlval, $ refsize = 1, $ defaultunit = 'px', $ points = false)

Convert HTML string containing value and unit of measure to user's units or points.

Parameters:

$htmlval	(string) String containing values and unit.
$refsize	(string) Reference value in points.
$defaultunit	(string) Default unit (can be one of the following: %, em, ex, px, in, mm, pc, pt).
$points	(boolean) If true returns points, otherwise returns value in user's units.

Returns:
float value in user's unit or point if $points=true

Since:
4.4.004 (2008-12-10)

TCPDF::getImageBuffer ($ image) [protected]

Get image buffer content.

Parameters:
$image (string) image key

Returns:
string image buffer content or false in case of error

Since:
4.5.000 (2008-12-31)

TCPDF::getImageRBX ( )

Return the right-bottom (or left-bottom for RTL) corner X coordinate of last inserted image

Returns:
float

TCPDF::getImageRBY ( )

Return the right-bottom (or left-bottom for RTL) corner Y coordinate of last inserted image

Returns:
float

TCPDF::getImageScale ( )

Returns the adjusting factor to convert pixels to user units.

Returns:
float adjusting factor to convert pixels to user units.

Author:
Nicola Asuni

Since:
1.5.2

TCPDF::getInternalPageNumberAliases ($ a = '') [protected]

Return an array containing variations for the basic page number alias.

Parameters:
$a (string) Base alias.

Returns:
array of page number aliases

TCPDF::getLastH ( )

Get the last cell height.

Returns:
last cell height

Since:
4.0.017 (2008-08-05)

TCPDF::GetLineWidth ( )

Returns the current the line width.

Returns:
int Line width

Since:
2.1.000 (2008-01-07)

See also:
Line(), SetLineWidth()

TCPDF::getMargins ( )

Returns an array containing current margins:

- $ret['left'] = left margin
- $ret['right'] = right margin
- $ret['top'] = top margin
- $ret['bottom'] = bottom margin
- $ret['header'] = header margin
- $ret['footer'] = footer margin
- $ret['cell'] = cell padding array
- $ret['padding_left'] = cell left padding
- $ret['padding_top'] = cell top padding
- $ret['padding_right'] = cell right padding
- $ret['padding_bottom'] = cell bottom padding

Returns:
array containing all margins measures

Since:
3.2.000 (2008-06-23)

TCPDF::getNumberOfColumns ( )

Return the current number of columns.

Returns:
int number of columns

Since:
5.8.018 (2010-08-25)

TCPDF::GetNumChars ($ s)

Returns the numbero of characters in a string.

Parameters:
$s (string) The input string.

Returns:
int number of characters

Since:
2.0.0001 (2008-01-07)

TCPDF::getNumLines ($ txt,

```
$ w = 0,
$ reseth = false,
$ autopadding = true,
$ cellpadding = '',
$ border = 0
)
```

This method return the estimated number of lines for print a simple text string using Multicell() method.

Parameters:

$txt	(string) String for calculating his height
$w	(float) Width of cells. If 0, they extend up to the right margin of the page.
$reseth	(boolean) if true reset the last cell height (default false).
$autopadding	(boolean) if true, uses internal padding and automatically adjust it to account for line width (default true).
$cellpadding	(float) Internal cell padding, if empty uses default cell padding.
$border	(mixed) Indicates if borders must be drawn around the cell. The value can be a number: • 0: no border (default) • 1: frame or a string containing some or all of the following characters (in any order): • L: left • T: top • R: right • B: bottom or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

Returns:
float Return the minimal height needed for multicell method for printing the $txt param.

Author:
Alexander Escalona Fern, Nicola Asuni

Since:
4.5.011

TCPDF::getNumPages ( )

Get the total number of insered pages.

Returns:
int number of pages

Since:
2.1.000 (2008-01-07)

See also:
setPage(), getPage(), lastpage()

TCPDF::getOriginalMargins ( )

Returns an array containing original margins:

- $ret['left'] = left margin
- $ret['right'] = right margin

Returns:
array containing all margins measures

Since:
4.0.012 (2008-07-24)

TCPDF::getOverprint ( )

Get the overprint mode array (OP, op, OPM). (Check the "Entries in a Graphics State Parameter Dictionary" on PDF 32000-1:2008).

Returns:
array.

Since:

5.9.152 (2012-03-23)

TCPDF::getPage ( )

Get current document page number.

Returns:
int page number

Since:
2.1.000 (2008-01-07)

See also:
setPage(), lastpage(), getNumPages()

TCPDF::getPageBuffer ($ page) [protected]

Get page buffer content.

Parameters:
$page (int) page number

Returns:
string page buffer content or false in case of error

Since:
4.5.000 (2008-12-31)

TCPDF::getPageDimensions ($ pagenum = '')

Returns an array of page dimensions:

- $this->pagedim[$this->page]['w'] = page width in points
- $this->pagedim[$this->page]['h'] = height in points
- $this->pagedim[$this->page]['wk'] = page width in user units
- $this->pagedim[$this->page]['hk'] = page height in user units
- $this->pagedim[$this->page]['tm'] = top margin
- $this->pagedim[$this->page]['bm'] = bottom margin
- $this->pagedim[$this->page]['lm'] = left margin
- $this->pagedim[$this->page]['rm'] = right margin
- $this->pagedim[$this->page]['pb'] = auto page break
- $this->pagedim[$this->page]['or'] = page orientation
- $this->pagedim[$this->page]['olm'] = original left margin
- $this->pagedim[$this->page]['orm'] = original right margin
- $this->pagedim[$this->page]['Rotate'] = The number of degrees by which the page shall be rotated clockwise when displayed or printed. The value shall be a multiple of 90.
- $this->pagedim[$this->page]['PZ'] = The page's preferred zoom (magnification) factor.
- $this->pagedim[$this->page]['trans'] : the style and duration of the visual transition to use when moving from another page to the given page during a presentation
  - $this->pagedim[$this->page]['trans']['Dur'] = The page's display duration (also called its advance timing): the maximum length of time, in seconds, that the page shall be displayed during presentations before the viewer application shall automatically advance to the next page.
  - $this->pagedim[$this->page]['trans']['S'] = transition style : Split, Blinds, Box, Wipe, Dissolve, Glitter, R, Fly, Push, Cover, Uncover, Fade
  - $this->pagedim[$this->page]['trans']['D'] = The duration of the transition effect, in seconds.
  - $this->pagedim[$this->page]['trans']['Dm'] = (Split and Blinds transition styles only) The dimension in which the specified transition effect shall occur: H = Horizontal, V = Vertical. Default value: H.
  - $this->pagedim[$this->page]['trans']['M'] = (Split, Box and Fly transition styles only) The direction of motion for the specified transition effect: I = Inward from the edges of the page, O = Outward from the center of the pageDefault value: I.
  - $this->pagedim[$this->page]['trans']['Di'] = (Wipe, Glitter, Fly, Cover, Uncover and Push transition styles only) The direction in which the specified transition effect shall moves, expressed in degrees counterclockwise starting from a left-to-right direction. If the value is a number, it shall be one of: 0 = Left to right, 90 = Bottom to top (Wipe only), 180 = Right to left (Wipe only), 270 = Top to bottom, 315 = Top-left to bottom-right (Glitter only). If the value is a name, it shall be None, which is relevant only for the Fly transition when the value of SS is not 1.0. Default value: 0.
  - $this->pagedim[$this->page]['trans']['SS'] = (Fly transition style only) The starting or ending scale at which the changes shall be drawn. If M specifies an inward transition, the scale of the changes drawn shall progress from SS to 1.0 over the course of the transition. If M specifies an outward transition, the scale of the changes drawn shall progress from 1.0 to SS over the course of the transition. Default: 1.0.
  - $this->pagedim[$this->page]['trans']['B'] = (Fly transition style only) If true, the area that shall be flown in is rectangular and opaque. Default: false.
- $this->pagedim[$this->page]['MediaBox'] : the boundaries of the physical medium on which the page shall be displayed or printed
  - $this->pagedim[$this->page]['MediaBox']['llx'] = lower-left x coordinate in points
  - $this->pagedim[$this->page]['MediaBox']['lly'] = lower-left y coordinate in points
  - $this->pagedim[$this->page]['MediaBox']['urx'] = upper-right x coordinate in points
  - $this->pagedim[$this->page]['MediaBox']['ury'] = upper-right y coordinate in points
- $this->pagedim[$this->page]['CropBox'] : the visible region of default user space
  - $this->pagedim[$this->page]['CropBox']['llx'] = lower-left x coordinate in points
  - $this->pagedim[$this->page]['CropBox']['lly'] = lower-left y coordinate in points
  - $this->pagedim[$this->page]['CropBox']['urx'] = upper-right x coordinate in points

    - $this->pagedim[$this->page]['CropBox']['ury'] = upper-right y coordinate in points
- $this->pagedim[$this->page]['BleedBox'] : the region to which the contents of the page shall be clipped when output in a production environment
    - $this->pagedim[$this->page]['BleedBox']['llx'] = lower-left x coordinate in points
    - $this->pagedim[$this->page]['BleedBox']['lly'] = lower-left y coordinate in points
    - $this->pagedim[$this->page]['BleedBox']['urx'] = upper-right x coordinate in points
    - $this->pagedim[$this->page]['BleedBox']['ury'] = upper-right y coordinate in points
- $this->pagedim[$this->page]['TrimBox'] : the intended dimensions of the finished page after trimming
    - $this->pagedim[$this->page]['TrimBox']['llx'] = lower-left x coordinate in points
    - $this->pagedim[$this->page]['TrimBox']['lly'] = lower-left y coordinate in points
    - $this->pagedim[$this->page]['TrimBox']['urx'] = upper-right x coordinate in points
    - $this->pagedim[$this->page]['TrimBox']['ury'] = upper-right y coordinate in points
- $this->pagedim[$this->page]['ArtBox'] : the extent of the page's meaningful content
    - $this->pagedim[$this->page]['ArtBox']['llx'] = lower-left x coordinate in points
    - $this->pagedim[$this->page]['ArtBox']['lly'] = lower-left y coordinate in points
    - $this->pagedim[$this->page]['ArtBox']['urx'] = upper-right x coordinate in points
    - $this->pagedim[$this->page]['ArtBox']['ury'] = upper-right y coordinate in points

Parameters:

$pagenum (int) page number (empty = current page)

Returns:

array of page dimensions.

Author:

Nicola Asuni

Since:

4.5.027 (2009-03-16)

TCPDF::getPageGroupAlias ( )

Return the alias for the total number of pages in the current page group. If the current font is unicode type, the returned string is surrounded by additional curly braces. This alias will be replaced by the total number of pages in this group.

Returns:

alias of the current page group

Since:

3.0.000 (2008-03-27)

TCPDF::getPageHeight ($ pagenum = '')

Returns the page height in units.

Parameters:

$pagenum (int) page number (empty = current page)

Returns:

int page height.

Author:

Nicola Asuni

Since:

1.5.2

See also:

getPageDimensions()

TCPDF::getPageNumGroupAlias ( )

Return the alias for the page number on the current page group. If the current font is unicode type, the returned string is surrounded by additional curly braces. This alias will be replaced by the page number (relative to the belonging group).

Returns:

alias of the current page group

Since:

4.5.000 (2009-01-02)

TCPDF::getPageRegions ( )

Return an array of no-write page regions

Returns:
array of no-write page regions

Author:
Nicola Asuni

Since:
5.9.003 (2010-10-13)

See also:
setPageRegions(), **addPageRegion()**

TCPDF::getPageWidth ($ pagenum = '')

Returns the page width in units.

Parameters:
$pagenum (int) page number (empty = current page)

Returns:
int page width.

Author:
Nicola Asuni

Since:
1.5.2

See also:
getPageDimensions()

TCPDF::getPDFData ( )

Returns the PDF data.

TCPDF::getRawCharWidth ($ char)

Returns the length of the char in user unit for the current font.

Parameters:
$char (int) The char code whose length is to be returned

Returns:
float char width

Author:
Nicola Asuni

Since:
5.9.000 (2010-09-28)

TCPDF::getRemainingWidth ( ) `[protected]`

Returns the remaining width between the current position and margins.

Returns:
int Return the remaining width

TCPDF::getRTL ( )

Return the RTL status

Returns:
boolean

Since:
4.0.012 (2008-07-24)

TCPDF::getScaleFactor ( )

Returns the scale factor (number of points in user unit).

Returns:
int scale factor.

Author:
Nicola Asuni

Since:
1.5.2

TCPDF::getSignatureAppearanceArray ($ **x** = 0,
$ **y** = 0,
$ **w** = 0,
$ **h** = 0,
$ **page** = -1,
$ **name** = ''
) [protected]

Get the array that defines the signature appearance (page and rectangle coordinates).

Parameters:

$x (float) Abscissa of the upper-left corner.
$y (float) Ordinate of the upper-left corner.
$w (float) Width of the signature area.
$h (float) Height of the signature area.
$page (int) option page number (if < 0 the current page is used).
$name (string) Name of the signature.

Returns:
(array) Array defining page and rectangle coordinates of signature appearance.

Author:
Nicola Asuni

Since:
5.9.101 (2011-07-06)

TCPDF::getSpaceString ( ) [protected]

Returns the string used to find spaces

Returns:
string

Author:
Nicola Asuni

Since:
4.8.024 (2010-01-15)

TCPDF::getStringHeight ($ **w**,
$ **txt**,
$ **reseth** = false,
$ **autopadding** = true,
$ **cellpadding** = '',
$ **border** = 0
)

This method return the estimated height needed for printing a simple text string using the Multicell() method. Generally, if you want to know the exact height for a block of content you can use the following alternative technique:

Precondition:
// store current object $pdf->**startTransaction()**; // store starting values $start_y = $pdf->**GetY()**; $start_page = $pdf->**getPage()**; // call your printing functions with your parameters

- $pdf->MultiCell($w=0, $h=0, $txt, $border=1, $align='L', $fill=false, $ln=1, $x='', $y='', $reseth=true, $stretch=0, $ishtml=false, $autopadding=true, $maxh=0);
- // get the new Y $end_y = $pdf->**GetY()**; $end_page = $pdf->**getPage()**; // calculate height $height = 0; if ($end_page == $start_page) { $height = $end_y - $start_y; } else { for ($page=$start_page; $page <= $end_page; ++$page) { $this->setPage($page); if ($page == $start_page) { // first page $height = $this->h - $start_y - $this->bMargin; } elseif ($page == $end_page) { // last page $height = $end_y - $this->tMargin; } else { $height = $this->h - $this->tMargin - $this->bMargin; } } } // restore previous object $pdf = $pdf->**rollbackTransaction()**;

Parameters:

$w (float) Width of cells. If 0, they extend up to the right margin of the page.

$txt (string) String for calculating his height

$reseth (boolean) if true reset the last cell height (default false).

$autopadding (boolean) if true, uses internal padding and automatically adjust it to account for line width (default true).

$cellpadding (float) Internal cell padding, if empty uses default cell padding.

$border (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top
- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

Returns:

float Return the minimal height needed for multicell method for printing the $txt param.

Author:

Nicola Asuni, Alexander Escalona Fern

TCPDF::GetStringWidth ($ **s,**
$ **fontname** = `''`,
$ **fontstyle** = `''`,
$ **fontsize** = `0`,
$ **getarray** = `false`
)

Returns the length of a string in user unit. A font must be selected.

Parameters:

$s (string) The string whose length is to be computed

$fontname (string) Family font. It can be either a name defined by **AddFont()** or one of the standard families. It is also possible to pass an empty string, in that case, the current family is retained.

$fontstyle (string) Font style. Possible values are (case insensitive):

- empty string: regular
- B: bold
- I: italic
- U: underline
- D: line-through
- O: overline

or any combination. The default value is regular.

$fontsize (float) Font size in points. The default value is the current size.

$getarray (boolean) if true returns an array of characters widths, if false returns the total length.

Returns:

mixed int total string length or array of characted widths

Author:

Nicola Asuni

Since:

1.2

TCPDF::getTextShadow ( )

Return the text shadow parameters array.

Returns:

Array of parameters.

Since:

5.9.174 (2012-07-25)

TCPDF::GetX ( )

Returns the relative X value of current position. The value is relative to the left border for LTR languages and to the right border for RTL languages.

Returns:
float

Since:
1.2

See also:
SetX(), GetY(), SetY()

TCPDF::GetY ( )

Returns the ordinate of the current position.

Returns:
float

Since:
1.0

See also:
SetY(), GetX(), SetX()

TCPDF::Gradient

```
TCPDF::Gradient ( $ type,
                  $ coords,
                  $ stops,
                  $ background = array(),
                  $ antialias = false
                )
```

Output gradient.

Parameters:

$type (int) type of gradient (1 Function-based shading; 2 Axial shading; 3 Radial shading; 4 Free-form Gouraud-shaded triangle mesh; 5 Lattice-form Gouraud-shaded triangle mesh; 6 Coons patch mesh; 7 Tensor-product patch mesh). (Not all types are currently supported)

$coords (array) array of coordinates.

$stops (array) array gradient color components: color = array of GRAY, RGB or CMYK color components; offset = (0 to 1) represents a location along the gradient vector; exponent = exponent of the exponential interpolation function (default = 1).

$background (array) An array of colour components appropriate to the colour space, specifying a single background colour value.

$antialias (boolean) A flag indicating whether to filter the shading function to prevent aliasing artifacts.

Author:
Nicola Asuni

Since:
3.1.000 (2008-06-09)

TCPDF::Header ( )

This method is used to render the page header. It is automatically called by AddPage() and could be overwritten in your own inherited class.

Reimplemented in TOC_TCPDF, MYPDF, and MYPDF.

TCPDF::hyphenateText

```
TCPDF::hyphenateText ( $ text,
                       $ patterns,
                       $ dictionary = array(),
                       $ leftmin = 1,
                       $ rightmin = 2,
                       $ charmin = 1,
                       $ charmax = 8
                     )
```

Returns text with soft hyphens.

Parameters:

$text (string) text to process

$patterns	(mixed) Array of hypenation patterns or a TEX file containing hypenation patterns. TEX patterns can be downloaded from http://www.ctan.org/tex-archive/language/hyph-utf8/tex/generic/hyph-utf8/patterns/
$dictionary	(array) Array of words to be returned without applying the hyphenation algoritm.
$leftmin	(int) Minimum number of character to leave on the left of the word without applying the hyphens.
$rightmin	(int) Minimum number of character to leave on the right of the word without applying the hyphens.
$charmin	(int) Minimum word length to apply the hyphenation algoritm.
$charmax	(int) Maximum length of broken piece of word.

Returns:

array text with soft hyphens

Author:

Nicola Asuni

Since:

4.9.012 (2010-04-12)

```
TCPDF::hyphenateWord ( $ word,
                       $ patterns,
                       $ dictionary = array(),
                       $ leftmin = 1,
                       $ rightmin = 2,
                       $ charmin = 1,
                       $ charmax = 8
                     )   [protected]
```

Returns an array of chars containing soft hyphens.

Parameters:

$word	(array) array of chars
$patterns	(array) Array of hypenation patterns.
$dictionary	(array) Array of words to be returned without applying the hyphenation algoritm.
$leftmin	(int) Minimum number of character to leave on the left of the word without applying the hyphens.
$rightmin	(int) Minimum number of character to leave on the right of the word without applying the hyphens.
$charmin	(int) Minimum word length to apply the hyphenation algoritm.
$charmax	(int) Maximum length of broken piece of word.

Returns:

array text with soft hyphens

Author:

Nicola Asuni

Since:

4.9.012 (2010-04-12)

```
TCPDF::Image ( $ file,
               $ x = '',
               $ y = '',
               $ w = 0,
               $ h = 0,
               $ type = '',
               $ link = '',
               $ align = '',
               $ resize = false,
               $ dpi = 300,
               $ palign = '',
               $ ismask = false,
               $ imgmask = false,
               $ border = 0,
               $ fitbox = false,
               $ hidden = false,
               $ fitonpage = false,
               $ alt = false,
               $ altimgs = array()
             )
```

Puts an image in the page. The upper-left corner must be given. The dimensions can be specified in different ways:

- explicit width and height (expressed in user unit)
- one explicit dimension, the other being calculated automatically in order to keep the original proportions
- no explicit dimension, in which case the image is put at 72 dpi

Supported formats are JPEG and PNG images whitout GD library and all images supported by GD: GD, GD2, GD2PART, GIF, JPEG, PNG, BMP, XBM, XPM; The format can be specified explicitly or inferred from the file extension.
It is possible to put a link on the image.
Remark: if an image is used several times, only one copy will be embedded in the file.

Parameters:

$file (string) Name of the file containing the image or a '@' character followed by the image data string. To link an image without embedding it on the document, set an asterisk character before the URL (i.e.: '*http://www.example.com/image.jpg').

$x (float) Abscissa of the upper-left corner (LTR) or upper-right corner (RTL).

$y (float) Ordinate of the upper-left corner (LTR) or upper-right corner (RTL).

$w (float) Width of the image in the page. If not specified or equal to zero, it is automatically calculated.

$h (float) Height of the image in the page. If not specified or equal to zero, it is automatically calculated.

$type (string) Image format. Possible values are (case insensitive): JPEG and PNG (whitout GD library) and all images supported by GD: GD, GD2, GD2PART, GIF, JPEG, PNG, BMP, XBM, XPM;. If not specified, the type is inferred from the file extension.

$link (mixed) URL or identifier returned by **AddLink()**.

$align (string) Indicates the alignment of the pointer next to image insertion relative to image height. The value can be:

- T: top-right for LTR or top-left for RTL
- M: middle-right for LTR or middle-left for RTL
- B: bottom-right for LTR or bottom-left for RTL
- N: next line

$resize (mixed) If true resize (reduce) the image to fit $w and $h (requires GD or ImageMagick library); if false do not resize; if 2 force resize in all cases (upscaling and downscaling).

$dpi (int) dot-per-inch resolution used on resize

$palign (string) Allows to center or align the image on the current line. Possible values are:

- L : left align
- C : center
- R : right align
- '' : empty string : left for LTR or right for RTL

$ismask (boolean) true if this image is a mask, false otherwise

$imgmask (mixed) image object returned by this function or false

$border (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top
- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

$fitbox (mixed) If not false scale image dimensions proportionally to fit within the ($w, $h) box. $fitbox can be true or a 2 characters string indicating the image alignment inside the box. The first character indicate the horizontal alignment (L = left, C = center, R = right) the second character indicate the vertical alignment (T = top, M = middle, B = bottom).

$hidden (boolean) If true do not display the image.

$fitonpage (boolean) If true the image is resized to not exceed page dimensions.

$alt (boolean) If true the image will be added as alternative and not directly printed (the ID of the image will be returned).

$altimgs (array) Array of alternate images IDs. Each alternative image must be an array with two values: an integer representing the image ID (the value returned by the Image method) and a boolean value to indicate if the image is the default for printing.

Returns:

image information

Since:

1.1

```
TCPDF::ImageEps ( $ file,
                  $ x = '',
                  $ y = '',
                  $ w = 0,
                  $ h = 0,
                  $ link = '',
                  $ useBoundingBox = true,
                  $ align = '',
                  $ palign = '',
                  $ border = 0,
                  $ fitonpage = false,
                  $ fixoutvals = false
                  )
```

Embed vector-based Adobe Illustrator (AI) or AI-compatible EPS files. NOTE: EPS is not yet fully implemented, use the **setRasterizeVectorImages()** method to enable/disable rasterization of vector images using ImageMagick library. Only vector drawing is supported, not text or bitmap. Although the script was successfully tested with various AI format versions, best results are probably achieved with files that were exported in the AI3 format (tested with Illustrator CS2, Freehand MX and Photoshop CS2).

Parameters:

$file	(string) Name of the file containing the image or a '@' character followed by the EPS/AI data string.
$x	(float) Abscissa of the upper-left corner.
$y	(float) Ordinate of the upper-left corner.
$w	(float) Width of the image in the page. If not specified or equal to zero, it is automatically calculated.
$h	(float) Height of the image in the page. If not specified or equal to zero, it is automatically calculated.
$link	(mixed) URL or identifier returned by **AddLink()**.
$useBoundingBox	(boolean) specifies whether to position the bounding box (true) or the complete canvas (false) at location (x,y). Default value is true.
$align	(string) Indicates the alignment of the pointer next to image insertion relative to image height. The value can be: • T: top-right for LTR or top-left for RTL • M: middle-right for LTR or middle-left for RTL • B: bottom-right for LTR or bottom-left for RTL • N: next line
$palign	(string) Allows to center or align the image on the current line. Possible values are: • L : left align • C : center • R : right align • '' : empty string : left for LTR or right for RTL
$border	(mixed) Indicates if borders must be drawn around the cell. The value can be a number: • 0: no border (default) • 1: frame or a string containing some or all of the following characters (in any order): • L: left • T: top • R: right • B: bottom or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))
$fitonpage	(boolean) if true the image is resized to not exceed page dimensions.
$fixoutvals	(boolean) if true remove values outside the bounding box.

Author:

Valentin Schmidt, Nicola Asuni

Since:

3.1.000 (2008-06-09)

```
TCPDF::ImagePngAlpha ( $ file,
                       $ x,
                       $ y,
                       $ wpx,
                       $ hpx,
                       $ w,
                       $ h,
                       $ type,
                       $ link,
                       $ align,
                       $ resize,
                       $ dpi,
                       $ palign,
                       $ filehash = ''
                     )   [protected]
```

Extract info from a PNG image with alpha channel using the Imagick or GD library.

Parameters:

$file (string) Name of the file containing the image.

$x (float) Abscissa of the upper-left corner.

$y (float) Ordinate of the upper-left corner.

$wpx (float) Original width of the image in pixels.

$hpx (float) original height of the image in pixels.

$w (float) Width of the image in the page. If not specified or equal to zero, it is automatically calculated.

$h (float) Height of the image in the page. If not specified or equal to zero, it is automatically calculated.

$type (string) Image format. Possible values are (case insensitive): JPEG and PNG (whitout GD library) and all images supported by GD: GD, GD2, GD2PART, GIF, JPEG, PNG, BMP, XBM, XPM;. If not specified, the type is inferred from the file extension.

$link (mixed) URL or identifier returned by **AddLink()**.

$align (string) Indicates the alignment of the pointer next to image insertion relative to image height. The value can be:

- T: top-right for LTR or top-left for RTL
- M: middle-right for LTR or middle-left for RTL
- B: bottom-right for LTR or bottom-left for RTL
- N: next line

$resize (boolean) If true resize (reduce) the image to fit $w and $h (requires GD library).

$dpi (int) dot-per-inch resolution used on resize

$palign (string) Allows to center or align the image on the current line. Possible values are:

- L : left align
- C : center
- R : right align
- '' : empty string : left for LTR or right for RTL

$filehash (string) File hash used to build unique file names.

Author:

Nicola Asuni

Since:

4.3.007 (2008-12-04)

See also:

Image()

```
TCPDF::ImageSVG ( $ file,
                  $ x = '',
                  $ y = '',
                  $ w = 0,
                  $ h = 0,
                  $ link = '',
                  $ align = '',
                  $ palign = '',
                  $ border = 0,
                  $ fitonpage = false
                )
```

Embedd a Scalable Vector Graphics (SVG) image. NOTE: SVG standard is not yet fully implemented, use the **setRasterizeVectorImages()** method to enable/disable rasterization of vector images using ImageMagick library.

Parameters:

$file (string) Name of the SVG file or a '@' character followed by the SVG data string.

$x (float) Abscissa of the upper-left corner.

$y (float) Ordinate of the upper-left corner.

$w (float) Width of the image in the page. If not specified or equal to zero, it is automatically calculated.

$h (float) Height of the image in the page. If not specified or equal to zero, it is automatically calculated.

$link (mixed) URL or identifier returned by **AddLink()**.

$align (string) Indicates the alignment of the pointer next to image insertion relative to image height. The value can be:

- T: top-right for LTR or top-left for RTL
- M: middle-right for LTR or middle-left for RTL
- B: bottom-right for LTR or bottom-left for RTL
- N: next line

If the alignment is an empty string, then the pointer will be restored on the starting SVG position.

$palign (string) Allows to center or align the image on the current line. Possible values are:

- L : left align
- C : center
- R : right align
- '' : empty string : left for LTR or right for RTL

$border (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top
- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

$fitonpage (boolean) if true the image is resized to not exceed page dimensions.

Author:

Nicola Asuni

Since:

5.0.000 (2010-05-02)

TCPDF::IncludeJS ($ script)

Adds a javascript

Parameters:

$script (string) Javascript code

Author:

Johannes G, Nicola Asuni

Since:

2.1.002 (2008-02-12)

TCPDF::inPageBody ( ) [protected]

Check if we are on the page body (excluding page header and footer).

Returns:

true if we are not in page header nor in page footer, false otherwise.

Since:

5.9.091 (2011-06-15)

TCPDF::isCharDefined ($ char, $ font = '', $ style = '')

Return true in the character is present in the specified font.

Parameters:

$char (mixed) Character to check (integer value or string)

$font (string) Font name (family name).

$style (string) Font style.

Returns:

(boolean) true if the char is defined, false otherwise.

Since:
5.9.153 (2012-03-28)

TCPDF::isRTLTextDir ( )

Return the current temporary RTL status

Returns:
boolean

Since:
4.8.014 (2009-11-04)

TCPDF::isUnicodeFont ( )

Return true if the current font is unicode type.

Returns:
true for unicode font, false otherwise.

Author:
Nicola Asuni

Since:
5.8.002 (2010-08-14)

TCPDF::lastPage ($ resetmargins = `false`)

Reset pointer to the last document page.

Parameters:
$resetmargins (boolean) if true reset left, right, top margins and Y position.

Since:
2.0.000 (2008-01-04)

See also:
setPage(), getPage(), getNumPages()

TCPDF::Line ($ x1, $ y1, $ x2, $ y2, $ style = `array()`)

Draws a line between two points.

Parameters:

$x1	(float) Abscissa of first point.
$y1	(float) Ordinate of first point.
$x2	(float) Abscissa of second point.
$y2	(float) Ordinate of second point.
$style	(array) Line style. Array like for SetLineStyle(). Default value: default line style (empty array).

Since:
1.0

See also:
SetLineWidth(), SetDrawColor(), SetLineStyle()

TCPDF::LinearGradient ($ x, $ y, $ w, $ h, $ col1 = `array()`, $ col2 = `array()`, $ coords = `array(0,0,1,0)`)

Paints a linear colour gradient.

Parameters:

$x (float) abscissa of the top left corner of the rectangle.
$y (float) ordinate of the top left corner of the rectangle.
$w (float) width of the rectangle.
$h (float) height of the rectangle.
$col1 (array) first color (Grayscale, RGB or CMYK components).
$col2 (array) second color (Grayscale, RGB or CMYK components).
$coords (array) array of the form (x1, y1, x2, y2) which defines the gradient vector (see linear_gradient_coords.jpg). The default value is from left to right (x1=0, y1=0, x2=1, y2=0).

Author:

Andreas W, Nicola Asuni

Since:

3.1.000 (2008-06-09)

```
TCPDF::Link ( $ x,
              $ y,
              $ w,
              $ h,
              $ link,

              $ spaces = 0
            )
```

Puts a link on a rectangular area of the page. Text or image links are generally put via **Cell()**, **Write()** or **Image()**, but this method can be useful for instance to define a clickable area inside an image.

Parameters:

$x (float) Abscissa of the upper-left corner of the rectangle
$y (float) Ordinate of the upper-left corner of the rectangle
$w (float) Width of the rectangle
$h (float) Height of the rectangle
$link (mixed) URL or identifier returned by **AddLink()**
$spaces (int) number of spaces on the text to link

Since:

1.5

See also:

AddLink(), **Annotation()**, **Cell()**, **Write()**, **Image()**

```
TCPDF::ListBox ( $ name,
                 $ w,
                 $ h,
                 $ values,
                 $ prop = array(),
                 $ opt = array(),
                 $ x = '',
                 $ y = '',
                 $ js = false
               )
```

Creates a List-box field

Parameters:

$name (string) field name
$w (int) width
$h (int) height
$values (array) array containing the list of values.
$prop (array) javascript field properties. Possible values are described on official Javascript for Acrobat API reference.
$opt (array) annotation parameters. Possible values are described on official PDF32000_2008 reference.
$x (float) Abscissa of the upper-left corner of the rectangle
$y (float) Ordinate of the upper-left corner of the rectangle

$js (boolean) if true put the field using JavaScript (requires Acrobat Writer to be rendered).

Author:
Nicola Asuni

Since:
4.8.000 (2009-09-07)

TCPDF::Ln ($ **h** = '',
$ **cell** = false
)

Performs a line break. The current abscissa goes back to the left margin and the ordinate increases by the amount passed in parameter.

Parameters:

$h	(float) The height of the break. By default, the value equals the height of the last printed cell.
$cell	(boolean) if true add the current left (or right o for RTL) padding to the X coordinate

Since:
1.0

See also:
Cell()

TCPDF::MirrorH ($ **x** = '')

Horizontal Mirroring.

Parameters:
$x (int) abscissa of the point. Default is current x position

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), **StopTransform()**

TCPDF::MirrorL ($ **angle** = 0,
$ **x** = '',
$ **y** = ''
)

Reflection against a straight line through point (x, y) with the gradient angle (angle).

Parameters:

$angle	(float) gradient angle of the straight line. Default is 0 (horizontal line).
$x	(int) abscissa of the point. Default is current x position
$y	(int) ordinate of the point. Default is current y position

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), **StopTransform()**

TCPDF::MirrorP ($ **x** = '',
$ **y** = ''
)

Point reflection mirroring.

Parameters:

$x	(int) abscissa of the point. Default is current x position
$y	(int) ordinate of the point. Default is current y position

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), **StopTransform()**

```
TCPDF::MirrorV ( $ y = '' )
```

Verical Mirroring.

Parameters:

$y (int) ordinate of the point. Default is current y position

Since:

2.1.000 (2008-01-07)

See also:

StartTransform(), StopTransform()

```
TCPDF::movePage ( $ frompage,
                  $ topage
                )
```

Move a page to a previous position.

Parameters:

$frompage (int) number of the source page

$topage (int) number of the destination page (must be less than $frompage)

Returns:

true in case of success, false in case of error.

Since:

4.5.000 (2009-01-02)

```
TCPDF::MultiCell ( $ w,
                   $ h,
                   $ txt,
                   $ border = 0,
                   $ align = 'J',
                   $ fill = false,
                   $ ln = 1,
                   $ x = '',
                   $ y = '',
                   $ reseth = true,
                   $ stretch = 0,
                   $ ishtml = false,
                   $ autopadding = true,
                   $ maxh = 0,
                   $ valign = 'T',
                   $ fitcell = false
                 )
```

This method allows printing text with line breaks. They can be automatic (as soon as the text reaches the right border of the cell) or explicit (via the character). As many cells as necessary are output, one below the other.
Text can be aligned, centered or justified. The cell block can be framed and the background painted.

Parameters:

$w (float) Width of cells. If 0, they extend up to the right margin of the page.

$h (float) Cell minimum height. The cell extends automatically if needed.

$txt (string) String to print

$border (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top
- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

$align (string) Allows to center or align the text. Possible values are:

- L or empty string: left align
- C: center
- R: right align
- J: justification (default value when $ishtml=false)

$fill (boolean) Indicates if the cell background must be painted (true) or transparent (false).

$ln (int) Indicates where the current position should go after the call. Possible values are:

- 0: to the right
- 1: to the beginning of the next line [DEFAULT]
- 2: below

$x (float) x position in user units

$y (float) y position in user units

$reseth (boolean) if true reset the last cell height (default true).

$stretch (int) font stretch mode:

- 0 = disabled
- 1 = horizontal scaling only if text is larger than cell width
- 2 = forced horizontal scaling to fit cell width
- 3 = character spacing only if text is larger than cell width
- 4 = forced character spacing to fit cell width

General font stretching and scaling values will be preserved when possible.

$ishtml (boolean) INTERNAL USE ONLY -- set to true if $txt is HTML content (default = false). Never set this parameter to true, use instead **writeHTMLCell()** or **writeHTML()** methods.

$autopadding (boolean) if true, uses internal padding and automatically adjust it to account for line width.

$maxh (float) maximum height. It should be >= $h and less then remaining space to the bottom of the page, or 0 for disable this feature. This feature works only when $ishtml=false.

$valign (string) Vertical alignment of text (requires $maxh = $h > 0). Possible values are:

- T: TOP
- M: middle
- B: bottom

. This feature works only when $ishtml=false and the cell must fit in a single page.

$fitcell (boolean) if true attempt to fit all the text within the cell by reducing the font size (do not work in HTML mode).

Returns:
int Return the number of cells or 1 for html mode.

Since:
1.3

See also:
SetFont(), **SetDrawColor()**, **SetFillColor()**, **SetTextColor()**, **SetLineWidth()**, **Cell()**, **Write()**, **SetAutoPageBreak()**

TCPDF::Open ( )

This method begins the generation of the PDF document. It is not necessary to call it explicitly because **AddPage()** does it automatically. Note: no page is created by this method

Since:
1.0

See also:
AddPage(), **Close()**

TCPDF::openHTMLTagHandler ($ dom, $ key, $ cell) [protected]

Process opening tags.

Parameters:

$dom (array) html dom array

$key (int) current element id

$cell (boolean) if true add the default left (or right if RTL) padding to each new line (default false).

Returns:
$dom array

TCPDF::Output ($ **name** = 'doc.pdf',
$ **dest** = 'I'
)

Send the document to a given destination: string, local file or browser. In the last case, the plug-in may be used (if present) or a download ("Save as" dialog box) may be forced.
The method first calls **Close()** if necessary to terminate the document.

Parameters:

$name (string) The name of the file when saved. Note that special characters are removed and blanks characters are replaced with the underscore character.

$dest (string) Destination where to send the document. It can take one of the following values:

- I: send the file inline to the browser (default). The plug-in is used if available. The name given by name is used when one selects the "Save as" option on the link generating the PDF.
- D: send to the browser and force a file download with the name given by name.
- F: save to a local server file with the name given by name.
- S: return the document as a string (name is ignored).
- FI: equivalent to F + I option
- FD: equivalent to F + D option
- E: return the document as base64 mime multi-part email attachment (RFC 2045)

Since:
1.0

See also:
Close()

TCPDF::PageNo ( )

Returns the current page number.

Returns:
int page number

Since:
1.0

See also:
getAliasNbPages()

TCPDF::PageNoFormatted ( )

Returns the current page number formatted as a string.

Since:
4.2.005 (2008-11-06)

See also:
PaneNo(), formatPageNumber()

TCPDF::PieSector ($ **xc**,
$ **yc**,
$ **r**,
$ **a**,
$ **b**,
$ **style** = 'FD',
$ **cw** = true,
$ **o** = 90
)

Draw the sector of a circle. It can be used for instance to render pie charts.

Parameters:

$xc (float) abscissa of the center.

$yc (float) ordinate of the center.

$r (float) radius.
$a (float) start angle (in degrees).
$b (float) end angle (in degrees).
$style (string) Style of rendering. See the getPathPaintOperator() function for more information.
$cw,: (float) indicates whether to go clockwise (default: true).
$o,: (float) origin of angles (0 for 3 o'clock, 90 for noon, 180 for 9 o'clock, 270 for 6 o'clock). Default: 90.

Author:
Maxime Delorme, Nicola Asuni

Since:
3.1.000 (2008-06-09)

TCPDF::PieSectorXY ($ xc,
$ yc,
$ rx,
$ ry,
$ a,
$ b,
$ style = `'FD'`,
$ cw = `false`,
$ o = `0`,
$ nc = `2`
)

Draw the sector of an ellipse. It can be used for instance to render pie charts.

Parameters:
$xc (float) abscissa of the center.
$yc (float) ordinate of the center.
$rx (float) the x-axis radius.
$ry (float) the y-axis radius.
$a (float) start angle (in degrees).
$b (float) end angle (in degrees).
$style (string) Style of rendering. See the getPathPaintOperator() function for more information.
$cw,: (float) indicates whether to go clockwise.
$o,: (float) origin of angles (0 for 3 o'clock, 90 for noon, 180 for 9 o'clock, 270 for 6 o'clock).
$nc (integer) Number of curves used to draw a 90 degrees portion of arc.

Author:
Maxime Delorme, Nicola Asuni

Since:
3.1.000 (2008-06-09)

TCPDF::pixelsToUnits ($ px)

Converts pixels to User's Units.

Parameters:
$px (int) pixels

Returns:
float value in user's unit

See also:
setImageScale(), getImageScale()

TCPDF::Polycurve ($ x0,
$ y0,
$ segments,
$ style = `''`,
$ line_style = `array()`,
$ fill_color = `array()`
)

Draws a poly-Bezier curve. Each Bezier curve segment is a tangent to the line between the control points at either end of the curve.

Parameters:

$x0	(float) Abscissa of start point.
$y0	(float) Ordinate of start point.
$segments	(float) An array of bezier descriptions. Format: array(x1, y1, x2, y2, x3, y3).
$style	(string) Style of rendering. See the getPathPaintOperator() function for more information.
$line_style	(array) Line style of curve. Array like for **SetLineStyle()**. Default value: default line style (empty array).
$fill_color	(array) Fill color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName). Default value: default color (empty array).

See also:
SetLineStyle()

Since:
3.0008 (2008-05-12)

TCPDF::Polygon ($ **p**,
$ **style** = '',
$ **line_style** = array(),
$ **fill_color** = array(),
$ **closed** = true
)

Draws a polygon.

Parameters:

$p	(array) Points 0 to ($np - 1). Array with values (x0, y0, x1, y1,..., x(np-1), y(np - 1))
$style	(string) Style of rendering. See the getPathPaintOperator() function for more information.
$line_style	(array) Line style of polygon. Array with keys among the following: • all: Line style of all lines. Array like for **SetLineStyle()**. • 0 to ($np - 1): Line style of each line. Array like for **SetLineStyle()**. If a key is not present or is null, not draws the line. Default value is default line style (empty array).
$fill_color	(array) Fill color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName). Default value: default color (empty array).
$closed	(boolean) if true the polygon is closes, otherwise will remain open

Since:
2.1.000 (2008-01-08)

TCPDF::PolyLine ($ **p**,
$ **style** = '',
$ **line_style** = array(),
$ **fill_color** = array()
)

Draws a polygonal line

Parameters:

$p	(array) Points 0 to ($np - 1). Array with values (x0, y0, x1, y1,..., x(np-1), y(np - 1))
$style	(string) Style of rendering. See the getPathPaintOperator() function for more information.
$line_style	(array) Line style of polygon. Array with keys among the following: • all: Line style of all lines. Array like for **SetLineStyle()**. • 0 to ($np - 1): Line style of each line. Array like for **SetLineStyle()**. If a key is not present or is null, not draws the line. Default value is default line style (empty array).
$fill_color	(array) Fill color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName). Default value: default color (empty array).

Since:
4.8.003 (2009-09-15)

TCPDF::printTemplate ($ **id**,
$ **x** = '',
$ **y** = '',
$ **w** = 0,
$ **h** = 0,

```
                              $ align = '',
                              $ palign = '',
                              $ fitonpage = false
                             )
```

Print an XObject Template. You can print an XObject Template inside the currently opened Template. An XObject Template is a PDF block that is a self-contained description of any sequence of graphics objects (including path objects, text objects, and sampled images). An XObject Template may be painted multiple times, either on several pages or at several locations on the same page and produces the same results each time, subject only to the graphics state at the time it is invoked.

Parameters:

$id (string) The ID of XObject Template to print.

$x (int) X position in user units (empty string = current x position)

$y (int) Y position in user units (empty string = current y position)

$w (int) Width in user units (zero = remaining page width)

$h (int) Height in user units (zero = remaining page height)

$align (string) Indicates the alignment of the pointer next to template insertion relative to template height. The value can be:

- T: top-right for LTR or top-left for RTL
- M: middle-right for LTR or middle-left for RTL
- B: bottom-right for LTR or bottom-left for RTL
- N: next line

$palign (string) Allows to center or align the template on the current line. Possible values are:

- L : left align
- C : center
- R : right align
- '' : empty string : left for LTR or right for RTL

$fitonpage (boolean) If true the template is resized to not exceed page dimensions.

Author:

Nicola Asuni

Since:

5.8.017 (2010-08-24)

See also:

startTemplate(), endTemplate()

```
TCPDF::putHtmlListBullet ( $ listdepth,
                           $ listtype = '',
                           $ size = 10
                         )   [protected]
```

Output an HTML list bullet or ordered item symbol

Parameters:

$listdepth (int) list nesting level

$listtype (string) type of list

$size (float) current font size

Since:

4.4.004 (2008-12-10)

```
TCPDF::RadialGradient ( $ x,
                        $ y,
                        $ w,
                        $ h,
                        $ col1 = array(),
                        $ col2 = array(),
                        $ coords = array(0.5,0.5,0.5,0.5,1)
                      )
```

Paints a radial colour gradient.

Parameters:

$x (float) abscissa of the top left corner of the rectangle.

$y (float) ordinate of the top left corner of the rectangle.

$w	(float) width of the rectangle.
$h	(float) height of the rectangle.
$col1	(array) first color (Grayscale, RGB or CMYK components).
$col2	(array) second color (Grayscale, RGB or CMYK components).
$coords	(array) array of the form (fx, fy, cx, cy, r) where (fx, fy) is the starting point of the gradient with color1, (cx, cy) is the center of the circle with color2, and r is the radius of the circle (see radial_gradient_coords.jpg). (fx, fy) should be inside the circle, otherwise some areas will not be defined.

Author:
Andreas W, Nicola Asuni

Since:
3.1.000 (2008-06-09)

```
TCPDF::RadioButton ( $ name,
                     $ w,
                     $ prop = array(),
                     $ opt = array(),
                     $ onvalue = 'On',
                     $ checked = false,
                     $ x = '',
                     $ y = '',
                     $ js = false
                   )
```

Creates a RadioButton field.

Parameters:

$name	(string) Field name.
$w	(int) Width of the radio button.
$prop	(array) Javascript field properties. Possible values are described on official Javascript for Acrobat API reference.
$opt	(array) Annotation parameters. Possible values are described on official PDF32000_2008 reference.
$onvalue	(string) Value to be returned if selected.
$checked	(boolean) Define the initial state.
$x	(float) Abscissa of the upper-left corner of the rectangle
$y	(float) Ordinate of the upper-left corner of the rectangle
$js	(boolean) If true put the field using JavaScript (requires Acrobat Writer to be rendered).

Author:
Nicola Asuni

Since:
4.8.000 (2009-09-07)

```
TCPDF::readDiskCache ( $ filename ) [protected]
```

Read data from a temporary file on filesystem.

Parameters:

$filename	(string) file name

Returns:
mixed retrieved data

Since:
4.5.000 (2008-12-31)

```
TCPDF::Rect ( $ x,
              $ y,
              $ w,
              $ h,
              $ style = '',
              $ border_style = array(),
              $ fill_color = array()
            )
```

Draws a rectangle.

Parameters:

$x	(float) Abscissa of upper-left corner.
$y	(float) Ordinate of upper-left corner.
$w	(float) Width.
$h	(float) Height.
$style	(string) Style of rendering. See the getPathPaintOperator() function for more information.
$border_style	(array) Border style of rectangle. Array with keys among the following: • all: Line style of all borders. Array like for **SetLineStyle()**. • L, T, R, B or combinations: Line style of left, top, right or bottom border. Array like for **SetLineStyle()**. If a key is not present or is null, the correspondent border is not drawn. Default value: default line style (empty array).
$fill_color	(array) Fill color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName). Default value: default color (empty array).

Since:
1.0

See also:
SetLineStyle()

TCPDF::registrationMark ($ **x**,
$ **y**,
$ **r**,
$ **double** = `false`,
$ **cola** = `array(100,100,100,100,'All')`,
$ **colb** = `array(0,0,0,0,'None')`
)

Paints a registration mark

Parameters:

$x	(float) abscissa of the registration mark center.
$y	(float) ordinate of the registration mark center.
$r	(float) radius of the crop mark.
$double	(boolean) if true print two concentric crop marks.
$cola	(array) crop mark color (default spot registration color 'All').
$colb	(array) second crop mark color (default spot registration color 'None').

Author:
Nicola Asuni

Since:
4.9.000 (2010-03-26)

TCPDF::registrationMarkCMYK ($ **x**,
$ **y**,
$ **r**
)

Paints a CMYK registration mark

Parameters:

$x	(float) abscissa of the registration mark center.
$y	(float) ordinate of the registration mark center.
$r	(float) radius of the crop mark.

Author:
Nicola Asuni

Since:
6.0.038 (2013-09-30)

TCPDF::RegularPolygon ($ **x0**,
$ **y0**,
$ **r**,
$ **ns**,

$ **angle** = `0`,
$ **draw_circle** = `false`,
$ **style** = `''`,
$ **line_style** = `array()`,
$ **fill_color** = `array()`,
$ **circle_style** = `''`,
$ **circle_outLine_style** = `array()`,
$ **circle_fill_color** = `array()`
)

Draws a regular polygon.

Parameters:

$x0 (float) Abscissa of center point.

$y0 (float) Ordinate of center point.

$r,: (float) Radius of inscribed circle.

$ns (integer) Number of sides.

$angle (float) Angle oriented (anti-clockwise). Default value: 0.

$draw_circle (boolean) Draw inscribed circle or not. Default value: false.

$style (string) Style of rendering. See the getPathPaintOperator() function for more information.

$line_style (array) Line style of polygon sides. Array with keys among the following:

- all: Line style of all sides. Array like for **SetLineStyle()**.
- 0 to ($ns - 1): Line style of each side. Array like for **SetLineStyle()**.

If a key is not present or is null, not draws the side. Default value is default line style (empty array).

$fill_color (array) Fill color. Format: array(red, green, blue). Default value: default color (empty array).

$circle_style (string) Style of rendering of inscribed circle (if draws). Possible values are:

- D or empty string: Draw (default).
- F: Fill.
- DF or FD: Draw and fill.
- CNZ: Clipping mode (using the even-odd rule to determine which regions lie inside the clipping path).
- CEO: Clipping mode (using the nonzero winding number rule to determine which regions lie inside the clipping path).

$circle_outLine_style (array) Line style of inscribed circle (if draws). Array like for **SetLineStyle()**. Default value: default line style (empty array).

$circle_fill_color (array) Fill color of inscribed circle (if draws). Format: array(red, green, blue). Default value: default color (empty array).

Since:

2.1.000 (2008-01-08)

TCPDF::removePageRegion ($ **key**)

Remove a single no-write region.

Parameters:

$key (int) region key

Author:

Nicola Asuni

Since:

5.9.003 (2010-10-13)

See also:

setPageRegions(), **getPageRegions()**

TCPDF::replaceBuffer ($ **data**) `[protected]`

Replace the buffer content

Parameters:

$data (string) data

Since:

5.5.000 (2010-06-22)

TCPDF::replaceChar ($ **oldchar**,
$ **newchar**

) [protected]

Replace a char if is defined on the current font.

Parameters:

$oldchar (int) Integer code (unicode) of the character to replace.
$newchar (int) Integer code (unicode) of the new character.

Returns:

int the replaced char or the old char in case the new char i not defined

Since:

5.9.167 (2012-06-22)

TCPDF::replaceMissingChars ($ **text,**
$ **font =** '',
$ **style =** '',
$ **subs =** array()
)

Replace missing font characters on selected font with specified substitutions.

Parameters:

$text (string) Text to process.
$font (string) Font name (family name).
$style (string) Font style.
$subs (array) Array of possible character substitutions. The key is the character to check (integer value) and the value is a single intege value or an array of possible substitutes.

Returns:

(string) Processed text.

Since:

5.9.153 (2012-03-28)

TCPDF::replaceRightShiftPageNumAliases ($ **page,**
$ **aliases,**
$ **diff**
) [protected]

Replace right shift page number aliases with spaces to correct right alignment. This works perfectly only when using monospaced fonts.

Parameters:

$page (string) Page content.
$aliases (array) Array of page aliases.
$diff (int) initial difference to add.

Returns:

replaced page content.

TCPDF::resetColumns ( )

Remove columns and reset page margins.

Since:

5.9.072 (2011-04-26)

TCPDF::resetHeaderTemplate ( )

Reset the xobject template used by **Header()** method.

TCPDF::resetLastH ( )

Reset the last cell height.

Since:

5.9.000 (2010-10-03)

```
TCPDF::rollbackTransaction ( $ self = false )
```

This method allows to undo the latest transaction by returning the latest saved **TCPDF** object with **startTransaction()**.

Parameters:

$self (boolean) if true restores current class object to previous state without the need of reassignment via the returned value.

Returns:

TCPDF object.

Since:

4.5.029 (2009-03-19)

```
TCPDF::Rotate ( $ angle,
                $ x = '',
                $ y = ''
              )
```

Rotate object.

Parameters:

$angle (float) angle in degrees for counter-clockwise rotation
$x (int) abscissa of the rotation center. Default is current x position
$y (int) ordinate of the rotation center. Default is current y position

Since:

2.1.000 (2008-01-07)

See also:

StartTransform(), **StopTransform()**

```
TCPDF::RoundedRect ( $ x,
                     $ y,
                     $ w,
                     $ h,
                     $ r,
                     $ round_corner = '1111',
                     $ style = '',
                     $ border_style = array(),
                     $ fill_color = array()
                   )
```

Draws a rounded rectangle.

Parameters:

$x (float) Abscissa of upper-left corner.
$y (float) Ordinate of upper-left corner.
$w (float) Width.
$h (float) Height.
$r (float) the radius of the circle used to round off the corners of the rectangle.
$round_corner (string) Draws rounded corner or not. String with a 0 (not rounded i-corner) or 1 (rounded i-corner) in i-position. Positions are, in order and begin to 0: top right, bottom right, bottom left and top left. Default value: all rounded corner ("1111").
$style (string) Style of rendering. See the getPathPaintOperator() function for more information.
$border_style (array) Border style of rectangle. Array like for **SetLineStyle()**. Default value: default line style (empty array).
$fill_color (array) Fill color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName). Default value: default color (empty array).

Since:

2.1.000 (2008-01-08)

```
TCPDF::RoundedRectXY ( $ x,
                       $ y,
                       $ w,
                       $ h,

                       $ rx,
                       $ ry,
                       $ round_corner = '1111',
                       $ style = '',
                       $ border_style = array(),
                       $ fill_color = array()
                      )
```

Draws a rounded rectangle.

Parameters:

$x	(float) Abscissa of upper-left corner.
$y	(float) Ordinate of upper-left corner.
$w	(float) Width.
$h	(float) Height.
$rx	(float) the x-axis radius of the ellipse used to round off the corners of the rectangle.
$ry	(float) the y-axis radius of the ellipse used to round off the corners of the rectangle.
$round_corner	(string) Draws rounded corner or not. String with a 0 (not rounded i-corner) or 1 (rounded i-corner) in i-position. Positions are, in order and begin to 0: top right, bottom right, bottom left and top left. Default value: all rounded corner ("1111").
$style	(string) Style of rendering. See the getPathPaintOperator() function for more information.
$border_style	(array) Border style of rectangle. Array like for **SetLineStyle()**. Default value: default line style (empty array).
$fill_color	(array) Fill color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName). Default value: default color (empty array).

Since:
4.9.019 (2010-04-22)

```
TCPDF::Scale ( $ s_x,
               $ s_y,
               $ x = '',
               $ y = ''
             )
```

Vertical and horizontal non-proportional Scaling.

Parameters:

$s_x	(float) scaling factor for width as percent. 0 is not allowed.
$s_y	(float) scaling factor for height as percent. 0 is not allowed.
$x	(int) abscissa of the scaling center. Default is current x position
$y	(int) ordinate of the scaling center. Default is current y position

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), **StopTransform()**

```
TCPDF::ScaleX ( $ s_x,
                $ x = '',
                $ y = ''
              )
```

Horizontal Scaling.

Parameters:

$s_x	(float) scaling factor for width as percent. 0 is not allowed.
$x	(int) abscissa of the scaling center. Default is current x position
$y	(int) ordinate of the scaling center. Default is current y position

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), **StopTransform()**

```
TCPDF::ScaleXY ( $ s,
                 $ x = '',
                 $ y = ''
               )
```

Vertical and horizontal proportional Scaling.

Parameters:

$s (float) scaling factor for width and height as percent. 0 is not allowed.
$x (int) abscissa of the scaling center. Default is current x position
$y (int) ordinate of the scaling center. Default is current y position

Since:

2.1.000 (2008-01-07)

See also:

StartTransform(), StopTransform()

```
TCPDF::ScaleY ( $ s_y,
                $ x = '',
                $ y = ''
              )
```

Vertical Scaling.

Parameters:

$s_y (float) scaling factor for height as percent. 0 is not allowed.
$x (int) abscissa of the scaling center. Default is current x position
$y (int) ordinate of the scaling center. Default is current y position

Since:

2.1.000 (2008-01-07)

See also:

StartTransform(), StopTransform()

```
TCPDF::segSVGContentHandler ( $ parser,
                              $ data
                            )   [protected]
```

Sets the character data handler function for the XML parser.

Parameters:

$parser (resource) The first parameter, parser, is a reference to the XML parser calling the handler.
$data (string) The second parameter, data, contains the character data as a string.

Author:

Nicola Asuni

Since:

5.0.000 (2010-05-02)

```
TCPDF::selectColumn ( $ col = '' )
```

Set position at a given column

Parameters:

$col (int) column number (from 0 to getNumberOfColumns()-1); empty string = current column.

Since:

4.9.001 (2010-03-28)

```
TCPDF::serializeTCPDFtagParameters ( $ pararray )
```

Serialize an array of parameters to be used with TCPDF tag in HTML code.

Parameters:

$pararray (array) parameters array

Returns:

sting containing serialized data

Since:

4.9.006 (2010-04-02)

Deprecated:

TCPDF::SetAbsX ($ x)

Set the absolute X coordinate of the current pointer.

Parameters:

$x (float) The value of the abscissa in user units.

Since:

5.9.186 (2012-09-13)

See also:

setAbsX(), setAbsY(), SetAbsXY()

TCPDF::SetAbsXY ($ x,
$ y
)

Set the absolute X and Y coordinates of the current pointer.

Parameters:

$x (float) The value of the abscissa in user units.

$y (float) (float) The value of the ordinate in user units.

Since:

5.9.186 (2012-09-13)

See also:

setAbsX(), setAbsY(), SetAbsXY()

TCPDF::SetAbsY ($ y)

Set the absolute Y coordinate of the current pointer.

Parameters:

$y (float) (float) The value of the ordinate in user units.

Since:

5.9.186 (2012-09-13)

See also:

setAbsX(), setAbsY(), SetAbsXY()

TCPDF::setAlpha ($ stroking = `1`,
$ bm = `'Normal'`,
$ nonstroking = `''`,
$ ais = `false`
)

Set alpha for stroking (CA) and non-stroking (ca) operations.

Parameters:

$stroking	(float) Alpha value for stroking operations: real value from 0 (transparent) to 1 (opaque).
$bm	(string) blend mode, one of the following: Normal, Multiply, Screen, Overlay, Darken, Lighten, ColorDodge, ColorBurn, HardLight, SoftLight, Difference, Exclusion, Hue, Saturation, Color, Luminosity
$nonstroking	(float) Alpha value for non-stroking operations: real value from 0 (transparent) to 1 (opaque).
$ais	(boolean)

Since:

3.0.000 (2008-03-27)

TCPDF::SetAuthor ($ author)

Defines the author of the document.

Parameters:

$author (string) The name of the author.

Since:

1.2

See also:
SetCreator(), SetKeywords(), SetSubject(), SetTitle()

TCPDF::SetAutoPageBreak ($ auto, $ margin = 0)

Enables or disables the automatic page breaking mode. When enabling, the second parameter is the distance from the bottom of the page that defines the triggering limit. By default, the mode is on and the margin is 2 cm.

Parameters:

$auto (boolean) Boolean indicating if mode should be on or off.
$margin (float) Distance from the bottom of the page.

Since:
1.0

See also:
Cell(), MultiCell(), AcceptPageBreak()

TCPDF::setBarcode ($ bc = '')

Set document barcode.

Parameters:

$bc (string) barcode

TCPDF::SetBooklet ($ booklet = true, $ inner = -1, $ outer = -1)

Set the booklet mode for double-sided pages.

Parameters:

$booklet (boolean) true set the booklet mode on, false otherwise.
$inner (float) Inner page margin.
$outer (float) Outer page margin.

Since:
4.2.000 (2008-10-29)

TCPDF::setBookmark ($ txt, $ level = 0, $ y = -1, $ page = '', $ style = '', $ color = array(0,0,0), $ x = -1, $ link = '')

Adds a bookmark - alias for Bookmark().

Parameters:

$txt (string) Bookmark description.
$level (int) Bookmark level (minimum value is 0).
$y (float) Y position in user units of the bookmark on the selected page (default = -1 = current position; 0 = page start;).
$page (int) Target page number (leave empty for current page).
$style (string) Font style: B = Bold, I = Italic, BI = Bold + Italic.
$color (array) RGB color array (values from 0 to 255).
$x (float) X position in user units of the bookmark on the selected page (default = -1 = current position;).
$link (mixed) URL, or numerical link ID, or named destination (# character followed by the destination name), or embedded file (* character followed by the file name).

TCPDF::setBuffer ($ data) [protected]

Set buffer content (always append data).

Parameters:

$data (string) data

Since:

4.5.000 (2009-01-02)

TCPDF::setCellHeightRatio ($ h)

Set the height of the cell (line height) respect the font height.

Parameters:

$h (int) cell proportion respect font height (typical value = 1.25).

Since:

3.0.014 (2008-06-04)

```
TCPDF::setCellMargins ( $ left = '',
                        $ top = '',
                        $ right = '',
                        $ bottom = ''
                      )
```

Set the internal Cell margins.

Parameters:

$left	(float) left margin
$top	(float) top margin
$right	(float) right margin
$bottom	(float) bottom margin

Since:

5.9.000 (2010-10-03)

See also:

getCellMargins()

TCPDF::SetCellPadding ($ pad)

Set the same internal Cell padding for top, right, bottom, left-

Parameters:

$pad (float) internal padding.

Since:

2.1.000 (2008-01-09)

See also:

getCellPaddings(), **setCellPaddings()**

```
TCPDF::setCellPaddings ( $ left = '',
                         $ top = '',
                         $ right = '',
                         $ bottom = ''

                       )
```

Set the internal Cell paddings.

Parameters:

$left	(float) left padding
$top	(float) top padding
$right	(float) right padding
$bottom	(float) bottom padding

Since:

5.9.000 (2010-10-03)

See also:

getCellPaddings(), SetCellPadding()

TCPDF::setColor ($ **type**,
$ **col1** = `0`,
$ **col2** = `-1`,
$ **col3** = `-1`,
$ **col4** = `-1`,
$ **ret** = `false`,
$ **name** = `''`
)

Defines the color used by the specified type ('draw', 'fill', 'text').

Parameters:

$type (string) Type of object affected by this color: ('draw', 'fill', 'text').
$col1 (float) GRAY level for single color, or Red color for RGB (0-255), or CYAN color for CMYK (0-100).
$col2 (float) GREEN color for RGB (0-255), or MAGENTA color for CMYK (0-100).
$col3 (float) BLUE color for RGB (0-255), or YELLOW color for CMYK (0-100).
$col4 (float) KEY (BLACK) color for CMYK (0-100).
$ret (boolean) If true do not send the command.
$name (string) spot color name (if any)

Returns:

(string) The PDF command or empty string.

Since:

5.9.125 (2011-10-03)

TCPDF::setColorArray ($ **type**,
$ **color**,
$ **ret** = `false`
)

Set the color array for the specified type ('draw', 'fill', 'text'). It can be expressed in RGB, CMYK or GRAY SCALE components. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$type (string) Type of object affected by this color: ('draw', 'fill', 'text').
$color (array) Array of colors (1=gray, 3=RGB, 4=CMYK or 5=spotcolor=CMYK+name values).
$ret (boolean) If true do not send the PDF command.

Returns:

(string) The PDF command or empty string.

Since:

3.1.000 (2008-06-11)

TCPDF::setColumnsArray ($ **columns**)

Set columns array. Each column is represented by an array of arrays with the following keys: (w = width, s = space between columns, y = column top position).

Parameters:

$columns (array)

Since:

4.9.001 (2010-03-28)

TCPDF::SetCompression ($ **compress** = `true`)

Activates or deactivates page compression. When activated, the internal representation of each page is compressed, which leads to a compression ratio of about 2 for the resulting document. Compression is on by default. Note: the Zlib extension is required for this feature. If not present, compression will be turned off.

Parameters:

$compress (boolean) Boolean indicating if compression must be enabled.

Since:
1.4

TCPDF::setContentMark ($ page = 0) [protected]

Set start-writing mark on selected page. Borders and fills are always created after content and inserted on the position marked by this method.

Parameters:
$page (int) page number (default is the current page)

Since:
4.6.021 (2009-07-20)

TCPDF::SetCreator ($ creator)

Defines the creator of the document. This is typically the name of the application that generates the PDF.

Parameters:
$creator (string) The name of the creator.

Since:
1.2

See also:
SetAuthor(), **SetKeywords()**, **SetSubject()**, **SetTitle()**

TCPDF::SetDefaultMonospacedFont ($ font)

Defines the default monospaced font.

Parameters:
$font (string) Font name.

Since:
4.5.025

TCPDF::setDefaultTableColumns ($ cols = 4)

Set the default number of columns in a row for HTML tables.

Parameters:
$cols (int) number of columns

Since:
3.0.014 (2008-06-04)

TCPDF::setDestination ($ name, $ y = -1, $ page = '', $ x = -1)

Add a Named Destination. NOTE: destination names are unique, so only last entry will be saved.

Parameters:

$name	(string) Destination name.
$y	(float) Y position in user units of the destiantion on the selected page (default = -1 = current position; 0 = page start;).
$page	(int) Target page number (leave empty for current page).
$x	(float) X position in user units of the destiantion on the selected page (default = -1 = current position;).

Returns:
(string) Stripped named destination identifier or false in case of error.

Author:
Christian Deligant, Nicola Asuni

Since:
5.9.097 (2011-06-23)

```
TCPDF::SetDisplayMode ( $ zoom,
                        $ layout = 'SinglePage',
                        $ mode = 'UseNone'
                      )
```

Defines the way the document is to be displayed by the viewer.

Parameters:

$zoom (mixed) The zoom to use. It can be one of the following string values or a number indicating the zooming factor to use.

- fullpage: displays the entire page on screen
- fullwidth: uses maximum width of window
- real: uses real size (equivalent to 100% zoom)
- default: uses viewer default mode

$layout (string) The page layout. Possible values are:

- SinglePage Display one page at a time
- OneColumn Display the pages in one column
- TwoColumnLeft Display the pages in two columns, with odd-numbered pages on the left
- TwoColumnRight Display the pages in two columns, with odd-numbered pages on the right
- TwoPageLeft (PDF 1.5) Display the pages two at a time, with odd-numbered pages on the left
- TwoPageRight (PDF 1.5) Display the pages two at a time, with odd-numbered pages on the right

$mode (string) A name object specifying how the document should be displayed when opened:

- UseNone Neither document outline nor thumbnail images visible
- UseOutlines Document outline visible
- UseThumbs Thumbnail images visible
- FullScreen Full-screen mode, with no menu bar, window controls, or any other window visible
- UseOC (PDF 1.5) Optional content group panel visible
- UseAttachments (PDF 1.6) Attachments panel visible

Since:
1.2

```
TCPDF::setDocCreationTimestamp ( $ time )
```

Set the document creation timestamp

Parameters:
$time (mixed) Document creation timestamp in seconds or date-time string.

Since:
5.9.152 (2012-03-23)

```
TCPDF::SetDocInfoUnicode ( $ unicode = true )
```

Turn on/off Unicode mode for document information dictionary (meta tags). This has effect only when unicode mode is set to false.

Parameters:
$unicode (boolean) if true set the meta information in Unicode

Since:
5.9.027 (2010-12-01)

```
TCPDF::setDocModificationTimestamp ( $ time )
```

Set the document modification timestamp

Parameters:
$time (mixed) Document modification timestamp in seconds or date-time string.

Since:
5.9.152 (2012-03-23)

```
TCPDF::SetDrawColor ( $ col1 = 0,
                      $ col2 = -1,
                      $ col3 = -1,
                      $ col4 = -1,
```

$ **ret** = false,
$ **name** = ''
)

Defines the color used for all drawing operations (lines, rectangles and cell borders). It can be expressed in RGB components or gray scale. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$col1	(float) GRAY level for single color, or Red color for RGB (0-255), or CYAN color for CMYK (0-100).
$col2	(float) GREEN color for RGB (0-255), or MAGENTA color for CMYK (0-100).
$col3	(float) BLUE color for RGB (0-255), or YELLOW color for CMYK (0-100).
$col4	(float) KEY (BLACK) color for CMYK (0-100).
$ret	(boolean) If true do not send the command.
$name	(string) spot color name (if any)

Returns:
string the PDF command

Since:
1.3

See also:
SetDrawColorArray(), **SetFillColor()**, **SetTextColor()**, **Line()**, **Rect()**, **Cell()**, **MultiCell()**

TCPDF::SetDrawColorArray ($ **color**,
$ **ret** = false
)

Defines the color used for all drawing operations (lines, rectangles and cell borders). It can be expressed in RGB, CMYK or GRAY SCALE components. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$color	(array) Array of colors (1, 3 or 4 values).
$ret	(boolean) If true do not send the PDF command.

Returns:
string the PDF command

Since:
3.1.000 (2008-06-11)

See also:
SetDrawColor()

TCPDF::SetDrawSpotColor ($ **name**,
$ **tint** = 100
)

Defines the spot color used for all drawing operations (lines, rectangles and cell borders).

Parameters:

$name	(string) Name of the spot color.
$tint	(float) Intensity of the color (from 0 to 100 ; 100 = full intensity by default).

Since:
4.0.024 (2008-09-12)

See also:
AddSpotColor(), **SetFillSpotColor()**, **SetTextSpotColor()**

TCPDF::setEqualColumns ($ **numcols** = 0,
$ **width** = 0,
$ **y** = ''
)

Set multiple columns of the same size

Parameters:

$numcols	(int) number of columns (set to zero to disable columns mode)
$width	(int) column width

$y (int) column starting Y position (leave empty for current Y position)

Since:
4.9.001 (2010-03-28)

TCPDF::setExtGState ($ gs) [protected]

Add an extgstate

Parameters:
$gs (array) extgstate

Since:
3.0.000 (2008-03-27)

TCPDF::setExtraXMP ($ xmp)

Set additional XMP data to be added on the default XMP data just before the end of "x:xmpmeta" tag. IMPORTANT: This data is added as-is without controls, so you have to validate your data before using this method!

Parameters:
$xmp (string) Custom XMP data.

Since:
5.9.128 (2011-10-06)

TCPDF::SetFillColor ($ col1 = 0, $ col2 = -1, $ col3 = -1, $ col4 = -1, $ ret = false, $ name = '')

Defines the color used for all filling operations (filled rectangles and cell backgrounds). It can be expressed in RGB components or gray scale. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$col1	(float) GRAY level for single color, or Red color for RGB (0-255), or CYAN color for CMYK (0-100).
$col2	(float) GREEN color for RGB (0-255), or MAGENTA color for CMYK (0-100).
$col3	(float) BLUE color for RGB (0-255), or YELLOW color for CMYK (0-100).
$col4	(float) KEY (BLACK) color for CMYK (0-100).
$ret	(boolean) If true do not send the command.
$name	(string) Spot color name (if any).

Returns:
(string) The PDF command.

Since:
1.3

See also:
SetFillColorArray(), SetDrawColor(), SetTextColor(), Rect(), Cell(), MultiCell()

TCPDF::SetFillColorArray ($ color, $ ret = false)

Defines the color used for all filling operations (filled rectangles and cell backgrounds). It can be expressed in RGB, CMYK or GRAY SCALE components. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$color	(array) Array of colors (1, 3 or 4 values).
$ret	(boolean) If true do not send the PDF command.

Since:
3.1.000 (2008-6-11)

See also:

SetFillColor()

TCPDF::SetFillSpotColor ($ name,
$ tint = `100`
)

Defines the spot color used for all filling operations (filled rectangles and cell backgrounds).

Parameters:

$name (string) Name of the spot color.
$tint (float) Intensity of the color (from 0 to 100 ; 100 = full intensity by default).

Since:

4.0.024 (2008-09-12)

See also:

AddSpotColor(), SetDrawSpotColor(), SetTextSpotColor()

TCPDF::SetFont ($ family,
$ style = `''`,
$ size = `null`,
$ fontfile = `''`,
$ subset = `'default'`,
$ out = `true`
)

Sets the font used to print character strings. The font can be either a standard one or a font added via the AddFont() method. Standard fonts use Windows encoding cp1252 (Western Europe). The method can be called before the first page is created and the font is retained from page to page. If you just wish to change the current font size, it is simpler to call SetFontSize(). Note: for the standard fonts, the font metric files must be accessible. There are three possibilities for this:

- They are in the current directory (the one where the running script lies)
- They are in one of the directories defined by the include_path parameter
- They are in the directory defined by the K_PATH_FONTS constant

Parameters:

$family (string) Family font. It can be either a name defined by AddFont() or one of the standard Type1 families (case insensitive):

- times (Times-Roman)
- timesb (Times-Bold)
- timesi (Times-Italic)
- timesbi (Times-BoldItalic)
- helvetica (Helvetica)
- helveticab (Helvetica-Bold)
- helveticai (Helvetica-Oblique)
- helveticabi (Helvetica-BoldOblique)
- courier (Courier)
- courierb (Courier-Bold)
- courieri (Courier-Oblique)
- courierbi (Courier-BoldOblique)
- symbol (Symbol)
- zapfdingbats (ZapfDingbats)

It is also possible to pass an empty string. In that case, the current family is retained.

$style (string) Font style. Possible values are (case insensitive):

- empty string: regular
- B: bold
- I: italic
- U: underline
- D: line through
- O: overline

or any combination. The default value is regular. Bold and italic styles do not apply to Symbol and ZapfDingbats basic fonts or other fonts when not defined.

$size (float) Font size in points. The default value is the current size. If no size has been specified since the beginning of the document, the value taken is 12

$fontfile (string) The font definition file. By default, the name is built from the family and style, in lower case with no spaces.

(mixed) if true embedd only a subset of the font (stores only the information related to the used characters); if false embedd full font; if 'default' uses the default value set using setFontSubsetting(). This option is valid only for TrueTypeUnicode fonts. If you want to enable users to

$subset change the document, set this parameter to false. If you subset the font, the person who receives your PDF would need to have your same font in order to make changes to your PDF. The file size of the PDF would also be smaller because you are embedding only part of a font.

$out (boolean) if true output the font size command, otherwise only set the font properties.

Author:
Nicola Asuni

Since:
1.0

See also:
AddFont(), SetFontSize()

TCPDF::setFontBuffer ($ font, $ data) [protected]

Set font buffer content.

Parameters:

$font (string) font key

$data (array) font data

Since:
4.5.000 (2009-01-02)

TCPDF::SetFontSize ($ size, $ out = true)

Defines the size of the current font.

Parameters:

$size (float) The font size in points.

$out (boolean) if true output the font size command, otherwise only set the font properties.

Since:
1.0

See also:
SetFont()

TCPDF::setFontSpacing ($ spacing = 0)

Set the amount to increase or decrease the space between characters in a text.

Parameters:

$spacing (float) amount to increase or decrease the space between characters in a text (0 = default spacing)

Author:
Nicola Asuni

Since:
5.9.000 (2010-09-29)

TCPDF::setFontStretching ($ perc = 100)

Set the percentage of character stretching.

Parameters:

$perc (int) percentage of stretching (100 = no stretching)

Author:
Nicola Asuni

Since:
5.9.000 (2010-09-29)

TCPDF::setFontSubBuffer ($ font,

```
                                    $ key,
                                    $ data
                                  ) [protected]
```

Set font buffer content.

Parameters:

$font (string) font key
$key (string) font sub-key
$data (array) font data

Since:

4.5.000 (2009-01-02)

TCPDF::setFontSubsetting ($ enable = true)

Enable or disable default option for font subsetting.

Parameters:

$enable (boolean) if true enable font subsetting by default.

Author:

Nicola Asuni

Since:

5.3.002 (2010-06-07)

TCPDF::setFooter ( ) [protected]

This method is used to render the page footer.

Since:

4.0.012 (2008-07-24)

TCPDF::setFooterData ($ tc = array(0,0,0), $ lc = array(0,0,0))

Set footer data.

Parameters:

$tc (array) RGB array color for text.
$lc (array) RGB array color for line.

TCPDF::setFooterFont ($ font)

Set footer font.

Parameters:

$font (array) Array describing the basic font parameters: (family, style, size).

Since:

1.1

TCPDF::setFooterMargin ($ fm = 10)

Set footer margin. (minimum distance between footer and bottom page margin)

Parameters:

$fm (int) distance in user units

TCPDF::setFormDefaultProp ($ prop = array())

Set default properties for form fields.

Parameters:

$prop (array) javascript field properties. Possible values are described on official Javascript for Acrobat API reference.

Author:
Nicola Asuni

Since:
4.8.000 (2009-09-06)

TCPDF::setGraphicVars ($ gvars, $ extended = false) [protected]

Set graphic variables.

Parameters:

$gvars	(array) array of graphic variablesto restore
$extended	(boolean) if true restore extended graphic variables

Since:
4.2.010 (2008-11-14)

TCPDF::setHeader ( ) [protected]

This method is used to render the page header.

Since:
4.0.012 (2008-07-24)

TCPDF::setHeaderData ($ ln = '', $ lw = 0, $ ht = '', $ hs = '', $ tc = array(0,0,0), $ lc = array(0,0,0))

Set header data.

Parameters:

$ln	(string) header image logo
$lw	(string) header image logo width in mm
$ht	(string) string to print as title on document header
$hs	(string) string to print on document header
$tc	(array) RGB array color for text.
$lc	(array) RGB array color for line.

TCPDF::setHeaderFont ($ font)

Set header font.

Parameters:
$font (array) Array describing the basic font parameters: (family, style, size).

Since:
1.1

TCPDF::setHeaderMargin ($ hm = 10)

Set header margin. (minimum distance between header and top page margin)

Parameters:
$hm (int) distance in user units

TCPDF::setHeaderTemplateAutoreset ($ val = true)

Set a flag to automatically reset the xobject template used by **Header()** method at each page.

Parameters:
$val (boolean) set to true to reset Header xobject template at each page, false otherwise.

TCPDF::setHtmlLinksStyle ($ color = array(0,0,255), $ fontstyle = 'U')

Set the color and font style for HTML links.

Parameters:

$color (array) RGB array of colors
$fontstyle (string) additional font styles to add

Since:

4.4.003 (2008-12-09)

TCPDF::setHtmlVSpace ($ tagvs)

Set the vertical spaces for HTML tags. The array must have the following structure (example): $tagvs = array('h1' => array(0 => array('h' => '', 'n' => 2), 1 => array('h' => 1.3, 'n' => 1))); The first array level contains the tag names, the second level contains 0 for opening tags or 1 for closing tags, the third level contains the vertical space unit (h) and the number spaces to add (n). If the h parameter is not specified, default values are used.

Parameters:

$tagvs (array) array of tags and relative vertical spaces.

Since:

4.2.001 (2008-10-30)

TCPDF::setImageBuffer ($ image, $ data) [protected]

Set image buffer content.

Parameters:

$image (string) image key
$data (array) image data

Returns:

int image index number

Since:

4.5.000 (2008-12-31)

TCPDF::setImageScale ($ scale)

Set the adjusting factor to convert pixels to user units.

Parameters:

$scale (float) adjusting factor to convert pixels to user units.

Author:

Nicola Asuni

Since:

1.5.2

TCPDF::setImageSubBuffer ($ image, $ key, $ data) [protected]

Set image buffer content for a specified sub-key.

Parameters:

$image (string) image key
$key (string) image sub-key
$data (array) image data

Since:

4.5.000 (2008-12-31)

TCPDF::setJPEGQuality ($ quality)

Set the default JPEG compression quality (1-100)

Parameters:

$quality (int) JPEG quality, integer between 1 and 100

Since:

3.0.000 (2008-03-27)

TCPDF::SetKeywords ($ keywords)

Associates keywords with the document, generally in the form 'keyword1 keyword2 ...'.

Parameters:

$keywords (string) The list of keywords.

Since:

1.2

See also:

SetAuthor(), **SetCreator()**, **SetSubject()**, **SetTitle()**

TCPDF::setLanguageArray ($ language)

Set language array.

Parameters:

$language (array)

Since:

1.1

TCPDF::setLastH ($ h)

Set the last cell height.

Parameters:

$h (float) cell height.

Author:

Nicola Asuni

Since:

1.53.0.TC034

TCPDF::SetLeftMargin ($ margin)

Defines the left margin. The method can be called before creating the first page. If the current abscissa gets out of page, it is brought back to the margin.

Parameters:

$margin (float) The margin.

Since:

1.4

See also:

SetTopMargin(), **SetRightMargin()**, **SetAutoPageBreak()**, **SetMargins()**

TCPDF::SetLineStyle ($ style,
$ ret = `false`
)

Set line style.

Parameters:

(array) Line style. Array with keys among the following:

- width (float): Width of the line in user units.
- cap (string): Type of cap to put on the line. Possible values are: butt, round, square. The difference between "square" and "butt" is that "square" projects a flat end past the end of the line.

$style	• join (string): Type of join. Possible values are: miter, round, bevel. • dash (mixed): Dash pattern. Is 0 (without dash) or string with series of length values, which are the lengths of the on and off dashes. For example: "2" represents 2 on, 2 off, 2 on, 2 off, ...; "2,1" is 2 on, 1 off, 2 on, 1 off, ... • phase (integer): Modifier on the dash pattern which is used to shift the point at which the pattern starts. • color (array): Draw color. Format: array(GREY) or array(R,G,B) or array(C,M,Y,K) or array(C,M,Y,K,SpotColorName).
$ret	(boolean) if true do not send the command.

Returns:
string the PDF command

Since:
2.1.000 (2008-01-08)

TCPDF::SetLineWidth ($ width)

Defines the line width. By default, the value equals 0.2 mm. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$width	(float) The width.

Since:
1.0

See also:
Line(), Rect(), Cell(), MultiCell()

TCPDF::SetLink ($ link, $ y = 0, $ page = -1)

Defines the page and position a link points to.

Parameters:

$link	(int) The link identifier returned by AddLink()
$y	(float) Ordinate of target position; -1 indicates the current position. The default value is 0 (top of page)
$page	(int) Number of target page; -1 indicates the current page. This is the default value

Since:
1.5

See also:
AddLink()

TCPDF::setListIndentWidth ($ width)

Set custom width for list indentation.

Parameters:

$width	(float) width of the indentation. Use negative value to disable it.

Since:
4.2.007 (2008-11-12)

TCPDF::setLIsymbol ($ symbol = '!')

Set the default bullet to be used as LI bullet symbol

Parameters:

$symbol	(string) character or string to be used (legal values are: '' = automatic, '!' = auto bullet, '#' = auto numbering, 'disc', 'disc', 'circle', 'square', '1', 'decimal', 'decimal-leading-zero', 'i', 'lower-roman', 'I', 'upper-roman', 'a', 'lower-alpha', 'lower-latin', 'A', 'upper-alpha', 'upper-latin', 'lower-greek', 'img|type|width|height|image.ext')

Since:
4.0.028 (2008-09-26)

TCPDF::SetMargins ($ left, $ top,

```
                    $ right = -1,
                    $ keepmargins = false
                   )
```

Defines the left, top and right margins.

Parameters:

$left (float) Left margin.
$top (float) Top margin.
$right (float) Right margin. Default value is the left one.
$keepmargins (boolean) if true overwrites the default page margins

Since:
1.0

See also:
SetLeftMargin(), SetTopMargin(), SetRightMargin(), SetAutoPageBreak()

TCPDF::setOpenCell ($ isopen)

Set the top/bottom cell sides to be open or closed when the cell cross the page.

Parameters:

$isopen (boolean) if true keeps the top/bottom border open for the cell sides that cross the page.

Since:
4.2.010 (2008-11-14)

TCPDF::setOverprint ($ stroking = true, $ nonstroking = '', $ mode = 0)

Set overprint mode for stroking (OP) and non-stroking (op) painting operations. (Check the "Entries in a Graphics State Parameter Dictionary" on PDF 32000-1:2008).

Parameters:

$stroking (boolean) If true apply overprint for stroking operations.
$nonstroking (boolean) If true apply overprint for painting operations other than stroking.
$mode (integer) Overprint mode: (0 = each source colour component value replaces the value previously painted for the corresponding device colorant; 1 = a tint value of 0.0 for a source colour component shall leave the corresponding component of the previously painted colour unchanged).

Since:
5.9.152 (2012-03-23)

TCPDF::setPage ($ pnum, $ resetmargins = false)

Move pointer at the specified document page and update page dimensions.

Parameters:

$pnum (int) page number (1 ... numpages)
$resetmargins (boolean) if true reset left, right, top margins and Y position.

Since:
2.1.000 (2008-01-07)

See also:
getPage(), lastpage(), getNumPages()

TCPDF::setPageBoxTypes ($ boxes) [protected]

Set page boxes to be included on page descriptions.

Parameters:

$boxes (array) Array of page boxes to set on document: ('MediaBox', 'CropBox', 'BleedBox', 'TrimBox', 'ArtBox').

TCPDF::setPageBuffer ($ page,
$ data,
$ append = false
) [protected]

Set page buffer content.

Parameters:

$page (int) page number
$data (string) page data
$append (boolean) if true append data, false replace.

Since:

4.5.000 (2008-12-31)

TCPDF::setPageFormat ($ format,
$ orientation = 'P'
) [protected]

Change the format of the current page

Parameters:

$format

(mixed) The format used for pages. It can be either: one of the string values specified at getPageSizeFromFormat() documentation or an array of two numners (width, height) or an array containing the following measures and options:

- ['format'] = page format name (one of the above);
- ['Rotate'] : The number of degrees by which the page shall be rotated clockwise when displayed or printed. The value shall be a multiple of 90.
- ['PZ'] : The page's preferred zoom (magnification) factor.
- ['MediaBox'] : the boundaries of the physical medium on which the page shall be displayed or printed:
- ['MediaBox']['llx'] : lower-left x coordinate in points
- ['MediaBox']['lly'] : lower-left y coordinate in points
- ['MediaBox']['urx'] : upper-right x coordinate in points
- ['MediaBox']['ury'] : upper-right y coordinate in points
- ['CropBox'] : the visible region of default user space:
- ['CropBox']['llx'] : lower-left x coordinate in points
- ['CropBox']['lly'] : lower-left y coordinate in points
- ['CropBox']['urx'] : upper-right x coordinate in points
- ['CropBox']['ury'] : upper-right y coordinate in points
- ['BleedBox'] : the region to which the contents of the page shall be clipped when output in a production environment:
- ['BleedBox']['llx'] : lower-left x coordinate in points
- ['BleedBox']['lly'] : lower-left y coordinate in points
- ['BleedBox']['urx'] : upper-right x coordinate in points
- ['BleedBox']['ury'] : upper-right y coordinate in points
- ['TrimBox'] : the intended dimensions of the finished page after trimming:
- ['TrimBox']['llx'] : lower-left x coordinate in points
- ['TrimBox']['lly'] : lower-left y coordinate in points
- ['TrimBox']['urx'] : upper-right x coordinate in points
- ['TrimBox']['ury'] : upper-right y coordinate in points
- ['ArtBox'] : the extent of the page's meaningful content:
- ['ArtBox']['llx'] : lower-left x coordinate in points
- ['ArtBox']['lly'] : lower-left y coordinate in points
- ['ArtBox']['urx'] : upper-right x coordinate in points
- ['ArtBox']['ury'] : upper-right y coordinate in points
- ['BoxColorInfo'] :specify the colours and other visual characteristics that should be used in displaying guidelines on the screen for each of the possible page boundaries other than the MediaBox:
- ['BoxColorInfo'][BOXTYPE]['C'] : an array of three numbers in the range 0-255, representing the components in the DeviceRGB colour space.
- ['BoxColorInfo'][BOXTYPE]['W'] : the guideline width in default user units
- ['BoxColorInfo'][BOXTYPE]['S'] : the guideline style: S = Solid; D = Dashed
- ['BoxColorInfo'][BOXTYPE]['D'] : dash array defining a pattern of dashes and gaps to be used in drawing dashed guidelines
- ['trans'] : the style and duration of the visual transition to use when moving from another page to the given page during a presentation
- ['trans']['Dur'] : The page's display duration (also called its advance timing): the maximum length of time, in seconds, that the page shall be displayed during presentations before the viewer application shall automatically advance to the next page.
- ['trans']['S'] : transition style : Split, Blinds, Box, Wipe, Dissolve, Glitter, R, Fly, Push, Cover, Uncover, Fade
- ['trans']['D'] : The duration of the transition effect, in seconds.
- ['trans']['Dm'] : (Split and Blinds transition styles only) The dimension in which the specified transition effect shall occur: H = Horizontal, V = Vertical. Default value: H.
- ['trans']['M'] : (Split, Box and Fly transition styles only) The direction of motion for the specified transition effect: I = Inward from the edges of the page, O = Outward from the center of the pageDefault value: I.
- ['trans']['Di'] : (Wipe, Glitter, Fly, Cover, Uncover and Push transition styles only) The direction in which the specified transition effect shall moves, expressed in degrees counterclockwise starting from a left-to-right direction. If the value is a number, it shall be one of: 0 = Left to right, 90 = Bottom to top (Wipe only), 180 = Right to left (Wipe only), 270 = Top to bottom, 315 = Top-left to bottom-right

(Glitter only). If the value is a name, it shall be None, which is relevant only for the Fly transition when the value of SS is not 1.0. Default value: 0.

- ['trans']['SS'] : (Fly transition style only) The starting or ending scale at which the changes shall be drawn. If M specifies an inward transition, the scale of the changes drawn shall progress from SS to 1.0 over the course of the transition. If M specifies an outward transition, the scale of the changes drawn shall progress from 1.0 to SS over the course of the transition. Default: 1.0.
- ['trans']['B'] : (Fly transition style only) If true, the area that shall be flown in is rectangular and opaque. Default: false.

$orientation (string) page orientation. Possible values are (case insensitive):

- P or Portrait (default)
- L or Landscape
- '' (empty string) for automatic orientation

Since:
3.0.015 (2008-06-06)

See also:
getPageSizeFromFormat()

TCPDF::setPageMark ( )

Set start-writing mark on current page stream used to put borders and fills. Borders and fills are always created after content and inserted on the position marked by this method. This function must be called after calling **Image()** function for a background image. Background images must be always inserted before calling Multicell() or WriteHTMLCell() or WriteHTML() functions.

Since:
4.0.016 (2008-07-30)

TCPDF::setPageOrientation ($ orientation, $ autopagebreak = '', $ bottommargin = '')

Set page orientation.

Parameters:

$orientation (string) page orientation. Possible values are (case insensitive):

- P or Portrait (default)
- L or Landscape
- '' (empty string) for automatic orientation

$autopagebreak (boolean) Boolean indicating if auto-page-break mode should be on or off.

$bottommargin (float) bottom margin of the page.

Since:
3.0.015 (2008-06-06)

TCPDF::setPageRegions ($ regions = array())

Set no-write regions on page. A no-write region is a portion of the page with a rectangular or trapezium shape that will not be covered when writing text or html code. A region is always aligned on the left or right side of the page ad is defined using a vertical segment. You can set multiple regions for the same page.

Parameters:

$regions (array) array of no-write regions. For each region you can define an array as follow: ('page' => page number or empy for current page, 'xt' => X top, 'yt' => Y top, 'xb' => X bottom, 'yb' => Y bottom, 'side' => page side 'L' = left or 'R' = right). Omit this parameter to remove all regions.

Author:
Nicola Asuni

Since:
5.9.003 (2010-10-13)

See also:
addPageRegion(), **getPageRegions()**

TCPDF::setPageUnit ($ unit)

Set the units of measure for the document.

Parameters:

$unit	(string) User measure unit. Possible values are: • pt: point • mm: millimeter (default) • cm: centimeter • in: inch A point equals 1/72 of inch, that is to say about 0.35 mm (an inch being 2.54 cm). This is a very common unit in typography; font sizes are expressed in that unit.

Since:

3.0.015 (2008-06-06)

TCPDF::setPDFVersion ($ version = '1.7')

Set the PDF version (check PDF reference for valid values).

Parameters:

$version	(string) PDF document version.

Since:

3.1.000 (2008-06-09)

TCPDF::setPrintFooter ($ val = true)

Set a flag to print page footer.

Parameters:

$val	(boolean) set to true to print the page footer (default), false otherwise.

TCPDF::setPrintHeader ($ val = true)

Set a flag to print page header.

Parameters:

$val	(boolean) set to true to print the page header (default), false otherwise.

TCPDF::SetProtection

```
TCPDF::SetProtection ( $ permissions = array('print', 'modify', 'copy', 'annot-forms', 'fill-forms', 'extract', 'assemble', 'print-high')
                       $ user_pass = '',
                       $ owner_pass = null,
                       $ mode = 0,
                       $ pubkeys = null
                     )
```

Set document protection Remark: the protection against modification is for people who have the full Acrobat product. If you don't set any password, the document will open as usual. If you set a user password, the PDF viewer will ask for it before displaying the document. The master password, if different from the user one, can be used to get full access. Note: protecting a document requires to encrypt it, which increases the processing time a lot. This can cause a PHP time-out in some cases, especially if the document contains images or fonts.

Parameters:

$permissions	(Array) the set of permissions (specify the ones you want to block): • print : Print the document; • modify : Modify the contents of the document by operations other than those controlled by 'fill-forms', 'extract' and 'assemble'; • copy : Copy or otherwise extract text and graphics from the document; • annot-forms : Add or modify text annotations, fill in interactive form fields, and, if 'modify' is also set, create or modify interactive form fields (including signature fields); • fill-forms : Fill in existing interactive form fields (including signature fields), even if 'annot-forms' is not specified; • extract : Extract text and graphics (in support of accessibility to users with disabilities or for other purposes); • assemble : Assemble the document (insert, rotate, or delete pages and create bookmarks or thumbnail images), even if 'modify' is not set; • print-high : Print the document to a representation from which a faithful digital copy of the PDF content could be generated. When this is not set, printing is limited to a low-level representation of the appearance, possibly of degraded quality. • owner : (inverted logic - only for public-key) when set permits change of encryption and enables all other permissions.
$user_pass	(String) user password. Empty by default.
$owner_pass	(String) owner password. If not specified, a random value is used.
$mode	(int) encryption strength: 0 = RC4 40 bit; 1 = RC4 128 bit; 2 = AES 128 bit; 3 = AES 256 bit.
$pubkeys	(String) array of recipients containing public-key certificates ('c') and permissions ('p'). For example: array(array('c' => 'file://../examples/data/cert/tcpdf.crt', 'p' => array('print')))

Since:
2.0.000 (2008-01-02)

Author:
Nicola Asuni

TCPDF::setRasterizeVectorImages ($ mode)

Enable/disable rasterization of vector images using ImageMagick library.

Parameters:

$mode	(boolean) if true enable rasterization, false otherwise.

Since:
5.0.000 (2010-04-27)

TCPDF::SetRightMargin ($ margin)

Defines the right margin. The method can be called before creating the first page.

Parameters:

$margin	(float) The margin.

Since:
1.5

See also:
SetLeftMargin(), SetTopMargin(), SetAutoPageBreak(), SetMargins()

TCPDF::setRTL ($ enable,
$ resetx = true
)

Enable or disable Right-To-Left language mode

Parameters:

$enable	(Boolean) if true enable Right-To-Left language mode.
$resetx	(Boolean) if true reset the X position on direction change.

Since:
2.0.000 (2008-01-03)

TCPDF::setSignature ($ signing_cert = '',
$ private_key = '',
$ private_key_password = '',
$ extracerts = '',
$ cert_type = 2,
$ info = array()
)

Enable document signature (requires the OpenSSL Library). The digital signature improve document authenticity and integrity and allows o enable extra features on Acrobat Reader. To create self-signed signature: openssl req -x509 -nodes -days 365000 -newkey rsa:1024 -keyout tcpdf.crt -out tcpdf.crt To export crt to p12: openssl pkcs12 -export -in tcpdf.crt -out tcpdf.p12 To convert pfx certificate to pem: openssl pkcs12 -in tcpdf.pfx -out tcpdf.crt -nodes

Parameters:

$signing_cert	(mixed) signing certificate (string or filename prefixed with 'file://')
$private_key	(mixed) private key (string or filename prefixed with 'file://')
$private_key_password	(string) password
$extracerts	(string) specifies the name of a file containing a bunch of extra certificates to include in the signature which can for example be used to help the recipient to verify the certificate that you used.
$cert_type	(int) The access permissions granted for this document. Valid values shall be: 1 = No changes to the document shall be permitted; any change to the document shall invalidate the signature; 2 = Permitted changes shall be filling in forms, instantiating page templates, and signing; other changes shall invalidate the signature; 3 = Permitted changes shall be the same as for 2, as well as annotation creation, deletion, and modification; other changes shall invalidate the signature.
$info	(array) array of option information: Name, Location, Reason, ContactInfo.

Author:
Nicola Asuni

Since:

4.6.005 (2009-04-24)

TCPDF::setSignatureAppearance ($ x = 0, $ y = 0, $ w = 0, $ h = 0, $ page = -1, $ name = '')

Set the digital signature appearance (a cliccable rectangle area to get signature properties)

Parameters:

$x (float) Abscissa of the upper-left corner.
$y (float) Ordinate of the upper-left corner.
$w (float) Width of the signature area.
$h (float) Height of the signature area.
$page (int) option page number (if < 0 the current page is used).
$name (string) Name of the signature.

Author:

Nicola Asuni

Since:

5.3.011 (2010-06-17)

TCPDF::setSpacesRE ($ re = '/[^\s\xa0]/')

Set regular expression to detect withespaces or word separators. The pattern delimiter must be the forward-slash character "/". Some example patterns are:

```
Non-Unicode or missing PCRE unicode support: "/[^\S\xa0]/"
Unicode and PCRE unicode support: "/(?!\xa0)[\s\p{Z}]/u"
Unicode and PCRE unicode support in Chinese mode: "/(?!\xa0)[\s\p{Z}\p{Lo}]/u"
if PCRE unicode support is turned ON ("\P" is the negate class of "\p"):
          : any whitespace character
     {Z}  : any separator
     {Lo} : Unicode letter or ideograph that does not have lowercase and uppercase variants. Is used to chunk chinese words.
        : Unicode Character 'NO-BREAK SPACE' (U+00A0)
```

Parameters:

$re (string) regular expression (leave empty for default).

Since:

4.6.016 (2009-06-15)

TCPDF::setSpotColor ($ type, $ name, $ tint = 100)

Set the spot color for the specified type ('draw', 'fill', 'text').

Parameters:

$type (string) Type of object affected by this color: ('draw', 'fill', 'text').
$name (string) Name of the spot color.
$tint (float) Intensity of the color (from 0 to 100 ; 100 = full intensity by default).

Returns:

(string) PDF color command.

Since:

5.9.125 (2011-10-03)

TCPDF::setSRGBmode ($ mode = false)

Set flag to force sRGB_IEC61966-2.1 black scaled ICC color profile for the whole document.

Parameters:

$mode (boolean) If true force sRGB output intent.

Since:
5.9.121 (2011-09-28)

TCPDF::setStartingPageNumber ($ num = 1)

Set the starting page number.

Parameters:
$num (int) Starting page number.

Since:
5.9.093 (2011-06-16)

TCPDF::SetSubject ($ subject)

Defines the subject of the document.

Parameters:
$subject (string) The subject.

Since:
1.2

See also:
SetAuthor(), SetCreator(), SetKeywords(), SetTitle()

TCPDF::setSVGStyles ($ svgstyle, $ prevsvgstyle, $ x = 0, $ y = 0, $ w = 1, $ h = 1, $ clip_function = '', $ clip_params = array()) [protected]

Apply the requested SVG styles (*** TO BE COMPLETED ***)

Parameters:

$svgstyle	(array) array of SVG styles to apply
$prevsvgstyle	(array) array of previous SVG style
$x	(int) X origin of the bounding box
$y	(int) Y origin of the bounding box
$w	(int) width of the bounding box
$h	(int) height of the bounding box
$clip_function	(string) clip function
$clip_params	(array) array of parameters for clipping function

Returns:
object style

Author:
Nicola Asuni

Since:
5.0.000 (2010-05-02)

TCPDF::setTableHeader ( ) [protected]

This method is used to render the table header on new page (if any).

Since:
4.5.030 (2009-03-25)

TCPDF::setTempRTL ($ mode)

Force temporary RTL language direction

Parameters:

$mode (mixed) can be false, 'L' for LTR or 'R' for RTL

Since:
2.1.000 (2008-01-09)

```
TCPDF::SetTextColor ( $ col1 = 0,
                      $ col2 = -1,
                      $ col3 = -1,
                      $ col4 = -1,
                      $ ret = false,
                      $ name = ''
                    )
```

Defines the color used for text. It can be expressed in RGB components or gray scale. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$col1	(float) GRAY level for single color, or Red color for RGB (0-255), or CYAN color for CMYK (0-100).
$col2	(float) GREEN color for RGB (0-255), or MAGENTA color for CMYK (0-100).
$col3	(float) BLUE color for RGB (0-255), or YELLOW color for CMYK (0-100).
$col4	(float) KEY (BLACK) color for CMYK (0-100).
$ret	(boolean) If true do not send the command.
$name	(string) Spot color name (if any).

Returns:
(string) Empty string.

Since:
1.3

See also:
SetTextColorArray(), **SetDrawColor()**, **SetFillColor()**, **Text()**, **Cell()**, **MultiCell()**

```
TCPDF::SetTextColorArray ( $ color,
                           $ ret = false
                         )
```

Defines the color used for text. It can be expressed in RGB components or gray scale. The method can be called before the first page is created and the value is retained from page to page.

Parameters:

$color	(array) Array of colors (1, 3 or 4 values).
$ret	(boolean) If true do not send the PDF command.

Since:
3.1.000 (2008-6-11)

See also:
SetFillColor()

```
TCPDF::setTextRenderingMode ( $ stroke = 0,
                              $ fill = true,
                              $ clip = false
                            )
```

Set Text rendering mode.

Parameters:

$stroke	(int) outline size in user units (0 = disable).
$fill	(boolean) if true fills the text (default).
$clip	(boolean) if true activate clipping mode

Since:
4.9.008 (2009-04-02)

```
TCPDF::setTextShadow ( $ params = array('enabled'=>false, 'depth_w'=>0, 'depth_h'=>0, 'color'=>false, 'opacity'=>1, 'blend_mode'=>'Nor
```

Set parameters for drop shadow effect for text.

Parameters:

$params (array) Array of parameters: enabled (boolean) set to true to enable shadow; depth_w (float) shadow width in user units; depth_h (float) shadow height in user units; color (array) shadow color or false to use the stroke color; opacity (float) Alpha value: real value from 0 (transparent) to 1 (opaque); blend_mode (string) blend mode, one of the following: Normal, Multiply, Screen, Overlay, Darken, Lighten, ColorDodge, ColorBurn, HardLight, SoftLight, Difference, Exclusion, Hue, Saturation, Color, Luminosity.

Since:

5.9.174 (2012-07-25)

TCPDF::SetTextSpotColor ($ **name**,
$ **tint** = `100`
)

Defines the spot color used for text.

Parameters:

$name (string) Name of the spot color.

$tint (int) Intensity of the color (from 0 to 100 ; 100 = full intensity by default).

Since:

4.0.024 (2008-09-12)

See also:

AddSpotColor(), **SetDrawSpotColor()**, **SetFillSpotColor()**

TCPDF::SetTitle ($ **title**)

Defines the title of the document.

Parameters:

$title (string) The title.

Since:

1.2

See also:

SetAuthor(), **SetCreator()**, **SetKeywords()**, **SetSubject()**

TCPDF::SetTopMargin ($ **margin**)

Defines the top margin. The method can be called before creating the first page.

Parameters:

$margin (float) The margin.

Since:

1.5

See also:

SetLeftMargin(), **SetRightMargin()**, **SetAutoPageBreak()**, **SetMargins()**

TCPDF::setUserRights ($ **enable** = `true`,
$ **document** = `'/FullSave'`,
$ **annots** = `'/Create/Delete/Modify/Copy/Import/Export'`,
$ **form** = `'/Add/Delete/FillIn/Import/Export/SubmitStandalone/SpawnTemplate'`,
$ **signature** = `'/Modify'`,
$ **ef** = `'/Create/Delete/Modify/Import'`,
$ **formex** = `''`
)

Set User's Rights for PDF Reader WARNING: This is experimental and currently do not work. Check the PDF Reference 8.7.1 Transform Methods, Table 8.105 Entries in the UR transform parameters dictionary

Parameters:

$enable (boolean) if true enable user's rights on PDF reader

$document (string) Names specifying additional document-wide usage rights for the document. The only defined value is "/FullSave", which permits a user to save the document along with modified form and/or annotation data.

$annots (string) Names specifying additional annotation-related usage rights for the document. Valid names in PDF 1.5 and later are /Create/Delete/Modify/Copy/Import/Export, which permit the user to perform the named operation on annotations.

$form	(string) Names specifying additional form-field-related usage rights for the document. Valid names are: /Add/Delete/FillIn/Import/Export/SubmitStandalone/SpawnTemplate
$signature	(string) Names specifying additional signature-related usage rights for the document. The only defined value is /Modify, which permits a user to apply a digital signature to an existing signature form field or clear a signed signature form field.
$ef	(string) Names specifying additional usage rights for named embedded files in the document. Valid names are /Create/Delete/Modify/Import, which permit the user to perform the named operation on named embedded files Names specifying additional embedded-files-related usage rights for the document.
$formex	(string) Names specifying additional form-field-related usage rights. The only valid name is BarcodePlaintext, which permits text form field data to be encoded as a plaintext two-dimensional barcode.

Author:

Nicola Asuni

Since:

2.9.000 (2008-03-26)

TCPDF::setViewerPreferences ($ preferences)

Set the viewer preferences dictionary controlling the way the document is to be presented on the screen or in print. (see Section 8.1 of PDF reference, "Viewer Preferences").

- HideToolbar boolean (Optional) A flag specifying whether to hide the viewer application's tool bars when the document is active. Default value: false.
- HideMenubar boolean (Optional) A flag specifying whether to hide the viewer application's menu bar when the document is active. Default value: false.
- HideWindowUI boolean (Optional) A flag specifying whether to hide user interface elements in the document's window (such as scroll bars and navigation controls), leaving only the document's contents displayed. Default value: false.
- FitWindow boolean (Optional) A flag specifying whether to resize the document's window to fit the size of the first displayed page. Default value: false.
- CenterWindow boolean (Optional) A flag specifying whether to position the document's window in the center of the screen. Default value: false.
- DisplayDocTitle boolean (Optional; PDF 1.4) A flag specifying whether the window's title bar should display the document title taken from the Title entry of the document information dictionary (see Section 10.2.1, "Document Information Dictionary"). If false, the title bar should instead display the name of the PDF file containing the document. Default value: false.
- NonFullScreenPageMode name (Optional) The document's page mode, specifying how to display the document on exiting full-screen mode:
  - UseNone Neither document outline nor thumbnail images visible
  - UseOutlines Document outline visible
  - UseThumbs Thumbnail images visible
  - UseOC Optional content group panel visible

  This entry is meaningful only if the value of the PageMode entry in the catalog dictionary (see Section 3.6.1, "Document Catalog") is FullScreen; it is ignored otherwise. Default value: UseNone.
- ViewArea name (Optional; PDF 1.4) The name of the page boundary representing the area of a page to be displayed when viewing the document on the screen. Valid values are (see Section 10.10.1, "Page Boundaries").:
  - MediaBox
  - CropBox (default)
  - BleedBox
  - TrimBox
  - ArtBox
- ViewClip name (Optional; PDF 1.4) The name of the page boundary to which the contents of a page are to be clipped when viewing the document on the screen. Valid values are (see Section 10.10.1, "Page Boundaries").:
  - MediaBox
  - CropBox (default)
  - BleedBox
  - TrimBox
  - ArtBox
- PrintArea name (Optional; PDF 1.4) The name of the page boundary representing the area of a page to be rendered when printing the document. Valid values are (see Section 10.10.1, "Page Boundaries").:
  - MediaBox
  - CropBox (default)
  - BleedBox
  - TrimBox
  - ArtBox
- PrintClip name (Optional; PDF 1.4) The name of the page boundary to which the contents of a page are to be clipped when printing the document. Valid values are (see Section 10.10.1, "Page Boundaries").:
  - MediaBox
  - CropBox (default)
  - BleedBox
  - TrimBox
  - ArtBox
- PrintScaling name (Optional; PDF 1.6) The page scaling option to be selected when a print dialog is displayed for this document. Valid values are:
  - None, which indicates that the print dialog should reflect no page scaling
  - AppDefault (default), which indicates that applications should use the current print scaling
- Duplex name (Optional; PDF 1.7) The paper handling option to use when printing the file from the print dialog. The following values are valid:
  - Simplex - Print single-sided
  - DuplexFlipShortEdge - Duplex and flip on the short edge of the sheet
  - DuplexFlipLongEdge - Duplex and flip on the long edge of the sheet

  Default value: none
- PickTrayByPDFSize boolean (Optional; PDF 1.7) A flag specifying whether the PDF page size is used to select the input paper tray. This setting influences only the preset values used to populate the print dialog presented by a PDF viewer application. If PickTrayByPDFSize is true, the check box

in the print dialog associated with input paper tray is checked. Note: This setting has no effect on Mac OS systems, which do not provide the ability to pick the input tray by size.

- PrintPageRange array (Optional; PDF 1.7) The page numbers used to initialize the print dialog box when the file is printed. The first page of the PDF file is denoted by 1. Each pair consists of the first and last pages in the sub-range. An odd number of integers causes this entry to be ignored. Negative numbers cause the entire array to be ignored. Default value: as defined by PDF viewer application
- NumCopies integer (Optional; PDF 1.7) The number of copies to be printed when the print dialog is opened for this file. Supported values are the integers 2 through 5. Values outside this range are ignored. Default value: as defined by PDF viewer application, but typically 1

Parameters:

$preferences (array) array of options.

Author:

Nicola Asuni

Since:

3.1.000 (2008-06-09)

TCPDF::setVisibility ($ v)

Set the visibility of the successive elements. This can be useful, for instance, to put a background image or color that will show on screen but won't print.

Parameters:

$v (string) visibility mode. Legal values are: all, print, screen or view.

Since:

3.0.000 (2008-03-27)

TCPDF::SetX ($ x, $ rtloff = false)

Defines the abscissa of the current position. If the passed value is negative, it is relative to the right of the page (or left if language is RTL).

Parameters:

$x	(float) The value of the abscissa in user units.
$rtloff	(boolean) if true always uses the page top-left corner as origin of axis.

Since:

1.2

See also:

GetX(), **GetY()**, **SetY()**, **SetXY()**

TCPDF::SetXY ($ x, $ y, $ rtloff = false)

Defines the abscissa and ordinate of the current position. If the passed values are negative, they are relative respectively to the right and bottom of the page.

Parameters:

$x	(float) The value of the abscissa.
$y	(float) The value of the ordinate.
$rtloff	(boolean) if true always uses the page top-left corner as origin of axis.

Since:

1.2

See also:

SetX(), **SetY()**

TCPDF::SetY ($ y, $ resetx = true, $ rtloff = false)

Moves the current abscissa back to the left margin and sets the ordinate. If the passed value is negative, it is relative to the bottom of the page.

Parameters:

$y	(float) The value of the ordinate in user units.

$resetx (bool) if true (default) reset the X position.
$rtloff (boolean) if true always uses the page top-left corner as origin of axis.

Since:
1.0

See also:
GetX(), GetY(), SetY(), SetXY()

TCPDF::Skew ($ **angle_x,**
$ **angle_y,**
$ **x = '',**
$ **y = ''**
)

Skew.

Parameters:
$angle_x (float) angle in degrees between -90 (skew to the left) and 90 (skew to the right)
$angle_y (float) angle in degrees between -90 (skew to the bottom) and 90 (skew to the top)
$x (int) abscissa of the skewing center. default is current x position
$y (int) ordinate of the skewing center. default is current y position

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), StopTransform()

TCPDF::SkewX ($ **angle_x,**
$ **x = '',**
$ **y = ''**
)

Skew horizontally.

Parameters:
$angle_x (float) angle in degrees between -90 (skew to the left) and 90 (skew to the right)
$x (int) abscissa of the skewing center. default is current x position
$y (int) ordinate of the skewing center. default is current y position

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), StopTransform()

TCPDF::SkewY ($ **angle_y,**
$ **x = '',**
$ **y = ''**
)

Skew vertically.

Parameters:
$angle_y (float) angle in degrees between -90 (skew to the bottom) and 90 (skew to the top)
$x (int) abscissa of the skewing center. default is current x position
$y (int) ordinate of the skewing center. default is current y position

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), StopTransform()

TCPDF::sortBookmarks ( ) `[protected]`

Sort bookmarks for page and key.

Since:

5.9.119 (2011-09-19)

```
TCPDF::StarPolygon ( $ x0,
                     $ y0,
                     $ r,
                     $ nv,
                     $ ng,
                     $ angle = 0,
                     $ draw_circle = false,
                     $ style = '',
                     $ line_style = array(),
                     $ fill_color = array(),
                     $ circle_style = '',
                     $ circle_outLine_style = array(),
                     $ circle_fill_color = array()
                   )
```

Draws a star polygon

Parameters:

$x0	(float) Abscissa of center point.
$y0	(float) Ordinate of center point.
$r	(float) Radius of inscribed circle.
$nv	(integer) Number of vertices.
$ng	(integer) Number of gap (if ($ng % $nv = 1) then is a regular polygon).
$angle,:	(float) Angle oriented (anti-clockwise). Default value: 0.
$draw_circle,:	(boolean) Draw inscribed circle or not. Default value is false.
$style	(string) Style of rendering. See the getPathPaintOperator() function for more information.
$line_style	(array) Line style of polygon sides. Array with keys among the following: • all: Line style of all sides. Array like for **SetLineStyle()**. • 0 to (n - 1): Line style of each side. Array like for **SetLineStyle()**. If a key is not present or is null, not draws the side. Default value is default line style (empty array).
$fill_color	(array) Fill color. Format: array(red, green, blue). Default value: default color (empty array).
$circle_style	(string) Style of rendering of inscribed circle (if draws). Possible values are: • D or empty string: Draw (default). • F: Fill. • DF or FD: Draw and fill. • CNZ: Clipping mode (using the even-odd rule to determine which regions lie inside the clipping path). • CEO: Clipping mode (using the nonzero winding number rule to determine which regions lie inside the clipping path).
$circle_outLine_style	(array) Line style of inscribed circle (if draws). Array like for **SetLineStyle()**. Default value: default line style (empty array).
$circle_fill_color	(array) Fill color of inscribed circle (if draws). Format: array(red, green, blue). Default value: default color (empty array).

Since:

2.1.000 (2008-01-08)

```
TCPDF::startLayer ( $ name = '',
                    $ print = true,
                    $ view = true,
                    $ lock = true
                  )
```

Start a new pdf layer.

Parameters:

$name	(string) Layer name (only a-z letters and numbers). Leave empty for automatic name.
$print	(boolean|null) Set to TRUE to print this layer, FALSE to not print and NULL to not set this option
$view	(boolean) Set to true to view this layer.
$lock	(boolean) If true lock the layer

Since:

5.9.102 (2011-07-13)

TCPDF::startPage ($ **orientation** = `''`,
$ **format** = `''`,
$ **tocpage** = `false`
)

Starts a new page to the document. The page must be closed using the **endPage()** function. The origin of the coordinate system is at the top-left corner and increasing ordinates go downwards.

Parameters:

$orientation (string) page orientation. Possible values are (case insensitive):

- P or PORTRAIT (default)
- L or LANDSCAPE

$format (mixed) The format used for pages. It can be either: one of the string values specified at getPageSizeFromFormat() or an array of parameters specified at **setPageFormat()**.

$tocpage (boolean) if true the page is designated to contain the Table-Of-Content.

Since:
4.2.010 (2008-11-14)

See also:
AddPage(), **endPage()**, **addTOCPage()**, **endTOCPage()**, getPageSizeFromFormat(), **setPageFormat()**

TCPDF::startPageGroup ($ **page** = `''`)

Create a new page group. NOTE: call this function before calling **AddPage()**

Parameters:
$page (int) starting group page (leave empty for next page).

Since:
3.0.000 (2008-03-27)

TCPDF::startSVGElementHandler ($ **parser**,
$ **name**,
$ **attribs**,
$ **ctm** = `array()`
) `[protected]`

Sets the opening SVG element handler function for the XML parser. (*** TO BE COMPLETED ***)

Parameters:

$parser (resource) The first parameter, parser, is a reference to the XML parser calling the handler.

$name (string) The second parameter, name, contains the name of the element for which this handler is called. If case-folding is in effect for this parser, the element name will be in uppercase letters.

$attribs (array) The third parameter, attribs, contains an associative array with the element's attributes (if any). The keys of this array are the attribute names, the values are the attribute values. Attribute names are case-folded on the same criteria as element names. Attribute values are not case-folded. The original order of the attributes can be retrieved by walking through attribs the normal way, using each(). The first key in the array was the first attribute, and so on.

$ctm (array) tranformation matrix for clipping mode (starting transformation matrix).

Author:
Nicola Asuni

Since:
5.0.000 (2010-05-02)

TCPDF::startTemplate ($ **w** = `0`,
$ **h** = `0`,
$ **group** = `false`
)

Start a new XObject Template. An XObject Template is a PDF block that is a self-contained description of any sequence of graphics objects (including path objects, text objects, and sampled images). An XObject Template may be painted multiple times, either on several pages or at several locations on the same page and produces the same results each time, subject only to the graphics state at the time it is invoked. Note: X,Y coordinates will be reset to 0,0.

Parameters:

$w (int) Template width in user units (empty string or zero = page width less margins).

$h (int) Template height in user units (empty string or zero = page height less margins).

$group (mixed) Set transparency group. Can be a boolean value or an array specifying optional parameters: 'CS' (solour space name), 'I' (boolean flag to indicate isolated group) and 'K' (boolean flag to indicate knockout group).

Returns:
int the XObject Template ID in case of success or false in case of error.

Author:
Nicola Asuni

Since:
5.8.017 (2010-08-24)

See also:
endTemplate(), **printTemplate()**

TCPDF::startTransaction ( )

Stores a copy of the current **TCPDF** object used for undo operation.

Since:
4.5.029 (2009-03-19)

TCPDF::StartTransform ( )

Starts a 2D tranformation saving current graphic state. This function must be called before scaling, mirroring, translation, rotation and skewing. Use **StartTransform()** before, and **StopTransform()** after the transformations to restore the normal behavior.

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), **StopTransform()**

TCPDF::StopTransform ( )

Stops a 2D tranformation restoring previous graphic state. This function must be called after scaling, mirroring, translation, rotation and skewing. Use **StartTransform()** before, and **StopTransform()** after the transformations to restore the normal behavior.

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), **StopTransform()**

TCPDF::stringLeftTrim ($ str,
$ replace = ' '
)

Left trim the input string

Parameters:

$str (string) string to trim
$replace (string) string that replace spaces.

Returns:
left trimmed string

Author:
Nicola Asuni

Since:
5.8.000 (2010-08-11)

TCPDF::stringRightTrim ($ str,
$ replace = ' '
)

Right trim the input string

Parameters:

$str (string) string to trim

$replace (string) string that replace spaces.

Returns:
right trimmed string

Author:
Nicola Asuni

Since:
5.8.000 (2010-08-11)

TCPDF::stringTrim ($ str, $ replace = '')

Trim the input string

Parameters:

$str	(string) string to trim
$replace	(string) string that replace spaces.

Returns:
trimmed string

Author:
Nicola Asuni

Since:
5.8.000 (2010-08-11)

TCPDF::SVGPath ($ d, $ style = '') [protected]

Draws an SVG path

Parameters:

$d (string) attribute d of the path SVG element

$style (string) Style of rendering. Possible values are:

- D or empty string: Draw (default).
- F: Fill.
- F*: Fill using the even-odd rule to determine which regions lie inside the clipping path.
- DF or FD: Draw and fill.
- DF* or FD*: Draw and fill using the even-odd rule to determine which regions lie inside the clipping path.
- CNZ: Clipping mode (using the even-odd rule to determine which regions lie inside the clipping path).
- CEO: Clipping mode (using the nonzero winding number rule to determine which regions lie inside the clipping path).

Returns:
array of container box measures (x, y, w, h)

Author:
Nicola Asuni

Since:
5.0.000 (2010-05-02)

TCPDF::SVGTransform ($ tm) [protected]

Apply SVG graphic transformation matrix.

Parameters:
$tm (array) original SVG transformation matrix

Since:
5.0.000 (2010-05-02)

TCPDF::swapMargins ($ reverse = true) [protected]

Swap the left and right margins.

Parameters:

$reverse (boolean) if true swap left and right margins.

Since:

4.2.000 (2008-10-29)

```
TCPDF::Text ( $ x,
              $ y,
              $ txt,
              $ fstroke = false,
              $ fclip = false,
              $ ffill = true,
              $ border = 0,
              $ ln = 0,
              $ align = '',
              $ fill = false,
              $ link = '',
              $ stretch = 0,
              $ ignore_min_height = false,
              $ calign = 'T',
              $ valign = 'M',
              $ rtloff = false
            )
```

Prints a text cell at the specified position. This method allows to place a string precisely on the page.

Parameters:

$x (float) Abscissa of the cell origin

$y (float) Ordinate of the cell origin

$txt (string) String to print

$fstroke (int) outline size in user units (false = disable)

$fclip (boolean) if true activate clipping mode (you must call **StartTransform()** before this function and **StopTransform()** to stop the clipping tranformation).

$ffill (boolean) if true fills the text

$border (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top
- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

$ln (int) Indicates where the current position should go after the call. Possible values are:

- 0: to the right (or left for RTL languages)
- 1: to the beginning of the next line
- 2: below

Putting 1 is equivalent to putting 0 and calling **Ln()** just after. Default value: 0.

$align (string) Allows to center or align the text. Possible values are:

- L or empty string: left align (default value)
- C: center
- R: right align
- J: justify

$fill (boolean) Indicates if the cell background must be painted (true) or transparent (false).

$link (mixed) URL or identifier returned by **AddLink()**.

$stretch (int) font stretch mode:

- 0 = disabled
- 1 = horizontal scaling only if text is larger than cell width
- 2 = forced horizontal scaling to fit cell width
- 3 = character spacing only if text is larger than cell width

- 4 = forced character spacing to fit cell width

General font stretching and scaling values will be preserved when possible.

$ignore_min_height (boolean) if true ignore automatic minimum height value.

$calign (string) cell vertical alignment relative to the specified Y value. Possible values are:

- T : cell top
- A : font top
- L : font baseline
- D : font bottom
- B : cell bottom

$valign (string) text vertical alignment inside the cell. Possible values are:

- T : top
- C : center
- B : bottom

$rtloff (boolean) if true uses the page top-left corner as origin of axis for $x and $y initial position.

Since:
1.0

See also:
Cell(), **Write()**, **MultiCell()**, WriteHTML(), WriteHTMLCell()

```
TCPDF::TextField ( $ name,
                   $ w,
                   $ h,
                   $ prop = array(),
                   $ opt = array(),
                   $ x = '',
                   $ y = '',
                   $ js = false
                 )
```

Creates a text field

Parameters:

$name (string) field name
$w (float) Width of the rectangle
$h (float) Height of the rectangle
$prop (array) javascript field properties. Possible values are described on official Javascript for Acrobat API reference.
$opt (array) annotation parameters. Possible values are described on official PDF32000_2008 reference.
$x (float) Abscissa of the upper-left corner of the rectangle
$y (float) Ordinate of the upper-left corner of the rectangle
$js (boolean) if true put the field using JavaScript (requires Acrobat Writer to be rendered).

Author:
Nicola Asuni

Since:
4.8.000 (2009-09-07)

```
TCPDF::Transform ( $ tm ) [protected]
```

Apply graphic transformations.

Parameters:

$tm (array) transformation matrix

Since:
2.1.000 (2008-01-07)

See also:
StartTransform(), **StopTransform()**

```
TCPDF::Translate ( $ t_x,
                   $ t_y
                 )
```

Translate graphic object horizontally and vertically.

Parameters:

$t_x (int) movement to the right
$t_y (int) movement to the bottom

Since:

2.1.000 (2008-01-07)

See also:

StartTransform(), StopTransform()

TCPDF::TranslateX ($ t_x)

Translate graphic object horizontally.

Parameters:

$t_x (int) movement to the right (or left for RTL)

Since:

2.1.000 (2008-01-07)

See also:

StartTransform(), StopTransform()

TCPDF::TranslateY ($ t_y)

Translate graphic object vertically.

Parameters:

$t_y (int) movement to the bottom

Since:

2.1.000 (2008-01-07)

See also:

StartTransform(), StopTransform()

TCPDF::unhtmlentities ($ text_to_convert)

Reverse function for htmlentities. Convert entities in UTF-8.

Parameters:

$text_to_convert (string) Text to convert.

Returns:

string converted text string

TCPDF::unichr ($ c)

Returns the unicode caracter specified by the value

Parameters:

$c (int) UTF-8 value

Returns:

Returns the specified character.

Since:

2.3.000 (2008-03-05)

Deprecated:

TCPDF::Write ($ h,
$ txt,
$ link = '',
$ fill = false,
$ align = '',
$ ln = false,
$ stretch = 0,

```
        $ firstline = false,
        $ firstblock = false,
        $ maxh = 0,
        $ wadj = 0,
        $ margin = ''
       )
```

This method prints text from the current position.

Parameters:

$h (float) Line height

$txt (string) String to print

$link (mixed) URL or identifier returned by **AddLink()**

$fill (boolean) Indicates if the cell background must be painted (true) or transparent (false).

$align (string) Allows to center or align the text. Possible values are:

- L or empty string: left align (default value)
- C: center
- R: right align
- J: justify

$ln (boolean) if true set cursor at the bottom of the line, otherwise set cursor at the top of the line.

$stretch (int) font stretch mode:

- 0 = disabled
- 1 = horizontal scaling only if text is larger than cell width
- 2 = forced horizontal scaling to fit cell width
- 3 = character spacing only if text is larger than cell width
- 4 = forced character spacing to fit cell width

General font stretching and scaling values will be preserved when possible.

$firstline (boolean) if true prints only the first line and return the remaining string.

$firstblock (boolean) if true the string is the starting of a line.

$maxh (float) maximum height. It should be >= $h and less then remaining space to the bottom of the page, or 0 for disable this feature.

$wadj (float) first line width will be reduced by this amount (used in HTML mode).

$margin (array) margin array of the parent container

Returns:

mixed Return the number of cells or the remaining string if $firstline = true.

Since:

1.5

```
TCPDF::write1DBarcode ( $ code,
                        $ type,
                        $ x = '',
                        $ y = '',
                        $ w = '',
                        $ h = '',
                        $ xres = '',
                        $ style = '',
                        $ align = ''
                      )
```

Print a Linear Barcode.

Parameters:

$code (string) code to print

$type (string) type of barcode (see **tcpdf_barcodes_1d.php** for supported formats).

$x (int) x position in user units (empty string = current x position)

$y (int) y position in user units (empty string = current y position)

$w (int) width in user units (empty string = remaining page width)

$h (int) height in user units (empty string = remaining page height)

$xres (float) width of the smallest bar in user units (empty string = default value = 0.4mm)

(array) array of options:

- boolean $style['border'] if true prints a border
- int $style['padding'] padding to leave around the barcode in user units (set to 'auto' for automatic padding)
- int $style['hpadding'] horizontal padding in user units (set to 'auto' for automatic padding)

$style

- int $style['vpadding'] vertical padding in user units (set to 'auto' for automatic padding)
- array $style['fgcolor'] color array for bars and text
- mixed $style['bgcolor'] color array for background (set to false for transparent)
- boolean $style['text'] if true prints text below the barcode
- string $style['label'] override default label
- string $style['font'] font name for text
- int $style['fontsize'] font size for text
- int $style['stretchtext']: 0 = disabled; 1 = horizontal scaling only if necessary; 2 = forced horizontal scaling; 3 = character spacing only if necessary; 4 = forced character spacing.
- string $style['position'] horizontal position of the containing barcode cell on the page: L = left margin; C = center; R = right margin.
- string $style['align'] horizontal position of the barcode on the containing rectangle: L = left; C = center; R = right.
- string $style['stretch'] if true stretch the barcode to best fit the available width, otherwise uses $xres resolution for a single bar.
- string $style['fitwidth'] if true reduce the width to fit the barcode width + padding. When this option is enabled the 'stretch' option is automatically disabled.
- string $style['cellfitalign'] this option works only when 'fitwidth' is true and 'position' is unset or empty. Set the horizontal position of the containing barcode cell inside the specified rectangle: L = left; C = center; R = right.

$align (string) Indicates the alignment of the pointer next to barcode insertion relative to barcode height. The value can be:

- T: top-right for LTR or top-left for RTL
- M: middle-right for LTR or middle-left for RTL
- B: bottom-right for LTR or bottom-left for RTL
- N: next line

Author:
Nicola Asuni

Since:
3.1.000 (2008-06-09)

```
TCPDF::write2DBarcode ( $ code,
                        $ type,
                        $ x = '',
                        $ y = '',
                        $ w = '',
                        $ h = '',
                        $ style = '',
                        $ align = '',
                        $ distort = false
                      )
```

Print 2D Barcode.

Parameters:

$code	(string) code to print
$type	(string) type of barcode (see **tcpdf_barcodes_2d.php** for supported formats).
$x	(int) x position in user units
$y	(int) y position in user units
$w	(int) width in user units
$h	(int) height in user units

$style (array) array of options:

- boolean $style['border'] if true prints a border around the barcode
- int $style['padding'] padding to leave around the barcode in barcode units (set to 'auto' for automatic padding)
- int $style['hpadding'] horizontal padding in barcode units (set to 'auto' for automatic padding)
- int $style['vpadding'] vertical padding in barcode units (set to 'auto' for automatic padding)
- int $style['module_width'] width of a single module in points
- int $style['module_height'] height of a single module in points
- array $style['fgcolor'] color array for bars and text
- mixed $style['bgcolor'] color array for background or false for transparent
- string $style['position'] barcode position on the page: L = left margin; C = center; R = right margin; S = stretch
- $style['module_width'] width of a single module in points
- $style['module_height'] height of a single module in points

$align (string) Indicates the alignment of the pointer next to barcode insertion relative to barcode height. The value can be:

- T: top-right for LTR or top-left for RTL
- M: middle-right for LTR or middle-left for RTL
- B: bottom-right for LTR or bottom-left for RTL
- N: next line

$distort	(boolean) if true distort the barcode to fit width and height, otherwise preserve aspect ratio

Author:
Nicola Asuni

Since:
4.5.037 (2009-04-07)

TCPDF::writeDiskCache ($ filename,
$ data,
$ append = `false`
) `[protected]`

Writes data to a temporary file on filesystem.

Parameters:

$filename	(string) file name
$data	(mixed) data to write on file
$append	(boolean) if true append data, false replace.

Since:
4.5.000 (2008-12-31)

TCPDF::writeHTML ($ html,
$ ln = `true`,
$ fill = `false`,
$ reseth = `false`,
$ cell = `false`,
$ align = `''`
)

Allows to preserve some HTML formatting (limited support).
IMPORTANT: The HTML must be well formatted - try to clean-up it using an application like HTML-Tidy before submitting. Supported tags are: a, b, blockquote, br, dd, del, div, dl, dt, em, font, h1, h2, h3, h4, h5, h6, hr, i, img, li, ol, p, pre, small, span, strong, sub, sup, table, tcpdf, td, th, thead, tr, tt, u, ul
NOTE: all the HTML attributes must be enclosed in double-quote.

Parameters:

$html	(string) text to display
$ln	(boolean) if true add a new line after text (default = true)
$fill	(boolean) Indicates if the background must be painted (true) or transparent (false).
$reseth	(boolean) if true reset the last cell height (default false).
$cell	(boolean) if true add the current left (or right for RTL) padding to each Write (default false).
$align	(string) Allows to center or align the text. Possible values are: • L : left align • C : center • R : right align • '' : empty string : left for LTR or right for RTL

TCPDF::writeHTMLCell ($ w,
$ h,
$ x,
$ y,
$ html = `''`,
$ border = `0`,
$ ln = `0`,
$ fill = `false`,
$ reseth = `true`,
$ align = `''`,
$ autopadding = `true`
)

Prints a cell (rectangular area) with optional borders, background color and html text string. The upper-left corner of the cell corresponds to the current position. After the call, the current position moves to the right or to the next line.
If automatic page breaking is enabled and the cell goes beyond the limit, a page break is done before outputting. IMPORTANT: The HTML must be well formatted - try to clean-up it using an application like HTML-Tidy before submitting. Supported tags are: a, b, blockquote, br, dd, del, div, dl, dt, em, font, h1,

h2, h3, h4, h5, h6, hr, i, img, li, ol, p, pre, small, span, strong, sub, sup, table, tcpdf, td, th, thead, tr, tt, u, ul NOTE: all the HTML attributes must be enclosed in double-quote.

Parameters:

$w (float) Cell width. If 0, the cell extends up to the right margin.

$h (float) Cell minimum height. The cell extends automatically if needed.

$x (float) upper-left corner X coordinate

$y (float) upper-left corner Y coordinate

$html (string) html text to print. Default value: empty string.

$border (mixed) Indicates if borders must be drawn around the cell. The value can be a number:

- 0: no border (default)
- 1: frame

or a string containing some or all of the following characters (in any order):

- L: left
- T: top
- R: right
- B: bottom

or an array of line styles for each border group - for example: array('LTRB' => array('width' => 2, 'cap' => 'butt', 'join' => 'miter', 'dash' => 0, 'color' => array(0, 0, 0)))

$ln (int) Indicates where the current position should go after the call. Possible values are:

- 0: to the right (or left for RTL language)
- 1: to the beginning of the next line
- 2: below

Putting 1 is equivalent to putting 0 and calling **Ln()** just after. Default value: 0.

$fill (boolean) Indicates if the cell background must be painted (true) or transparent (false).

$reseth (boolean) if true reset the last cell height (default true).

$align (string) Allows to center or align the text. Possible values are:

- L : left align
- C : center
- R : right align
- '' : empty string : left for LTR or right for RTL

$autopadding (boolean) if true, uses internal padding and automatically adjust it to account for line width.

See also:

Multicell(), **writeHTML()**

Member Data Documentation

TCPDF::$alpha = array('CA' => 1, 'ca' => 1, 'BM' => '/Normal', 'AIS' => false) [protected]

Alpha mode array. (Check the "Entries in a Graphics State Parameter Dictionary" on PDF 32000-1:2008).

Since:

5.9.152 (2012-03-23)

TCPDF::$annotation_fonts = array() [protected]

List of fonts used on form fields (fontname => fontkey).

Since:

4.8.001 (2009-09-09)

TCPDF::$author = '' [protected]

Document author.

TCPDF::$AutoPageBreak [protected]

Automatic page breaking.

TCPDF::$barcode = false [protected]

Barcode to print on page footer (only if set).

TCPDF::$bgcolor `[protected]`

Current background color.

TCPDF::$bMargin `[protected]`

Page break margin.

TCPDF::$booklet = false `[protected]`

Booklet mode for double-sided pages.

Since:
4.2.000 (2008-10-29)

TCPDF::$bordermrk = array() `[protected]`

Array used to store positions inside the pages buffer (keys are the page numbers).

Since:
5.7.000 (2010-08-03)

TCPDF::$buffer `[protected]`

Buffer holding in-memory PDF.

TCPDF::$bufferlen = 0 `[protected]`

Length of the buffer in bytes.

Since:
4.5.000 (2008-12-31)

TCPDF::$cache_file_length = array() `[protected]`

Array used to store the lengths of cache files.

Since:
4.5.029 (2009-03-19)

TCPDF::$cached_files = array() `[protected]`

Array of cached files.

TCPDF::$cell_height_ratio = K_CELL_HEIGHT_RATIO `[protected]`

Default cell height ratio.

Since:
3.0.014 (2008-05-23)

TCPDF::$cell_margin = array('T' => 0, 'R' => 0, 'B' => 0, 'L' => 0) `[protected]`

Array of cell margins ('T' => top, 'R' => right, 'B' => bottom, 'L' => left).

Since:
5.9.000 (2010-10-04)

TCPDF::$cell_padding = array('T' => 0, 'R' => 0, 'B' => 0, 'L' => 0) `[protected]`

Array of cell internal paddings ('T' => top, 'R' => right, 'B' => bottom, 'L' => left).

Since:
5.9.000 (2010-10-03)

TCPDF::$check_page_regions = true [protected]

Boolean value true when page region check is active.

TCPDF::$clMargin [protected]

Cell left margin (used by regions).

TCPDF::$cntmrk = array() [protected]

Array used to store content positions inside the pages buffer (keys are the page numbers).

Since:
4.6.021 (2009-07-20)

TCPDF::$ColorFlag [protected]

Indicates whether fill and text colors are different.

TCPDF::$column_start_page = 0 [protected]

Starting page for columns.

Since:
4.9.001 (2010-03-28)

TCPDF::$columns = array() [protected]

Array of column measures (width, space, starting Y position).

Since:
4.9.001 (2010-03-28)

TCPDF::$colxshift = array('x' => 0, 's' => array('H' => 0, 'V' => 0), 'p' => array('L' => 0, 'T' => 0, 'R' => 0, 'B' => 0)) [protected]

Array of: X difference between table cell x start and starting page margin, cellspacing, cellpadding.

Since:
5.8.000 (2010-08-11)

TCPDF::$compress [protected]

Compression flag.

TCPDF::$CoreFonts [protected]

Array of standard font names.

TCPDF::$creator = '' [protected]

Document creator.

TCPDF::$crMargin [protected]

Cell right margin (used by regions).

TCPDF::$CurOrientation [protected]

Current page orientation (P = Portrait, L = Landscape).

TCPDF::$current_column = 0 [protected]

Current column number.

Since:
4.9.001 (2010-03-28)

TCPDF::$CurrentFont `[protected]`

Current font info.

TCPDF::$currpagegroup = 0 `[protected]`

Current page group number.

Since:
3.0.000 (2008-03-27)

TCPDF::$custom_xmp = '' `[protected]`

Custom XMP data.

Since:
5.9.128 (2011-10-06)

TCPDF::$customlistindent = -1 `[protected]`

HTML PARSER: custom indent amount for lists. Negative value means disabled.

Since:
4.2.007 (2008-11-12)

TCPDF::$default_form_prop = array('lineWidth'=>1, 'borderStyle'=>'solid', 'fillColor'=>array(255, 255, 255), 'strokeColor'=>array(128, 128, 128)) `p`

Deafult Javascript field properties. Possible values are described on official Javascript for Acrobat API reference. Annotation options can be directly specified using the 'aopt' entry.

Since:
4.8.000 (2009-09-07)

TCPDF::$default_graphic_vars = array() `[protected]`

Array of default graphic settings.

Since:
5.5.008 (2010-07-02)

TCPDF::$default_monospaced_font = 'courier' `[protected]`

Default monospace font.

Since:
4.5.025 (2009-03-10)

TCPDF::$default_table_columns = 4 `[protected]`

Default number of columns for html table.

TCPDF::$dests = array() `[protected]`

A dictionary of names and corresponding destinations (Dests key on document Catalog).

Since:
5.9.097 (2011-06-23)

TCPDF::$diffs = array() `[protected]`

Array of encoding differences.

TCPDF::$diskcache = false `[protected]`

If true enables disk caching.

Since:

4.5.000 (2008-12-31)

TCPDF::$doc_creation_timestamp [protected]

Document creation date-time

Since:
5.9.152 (2012-03-22)

TCPDF::$doc_modification_timestamp [protected]

Document modification date-time

Since:
5.9.152 (2012-03-22)

TCPDF::$docinfounicode = true [protected]

If true set the document information dictionary in Unicode.

TCPDF::$dpi = 72 [protected]

DPI (Dot Per Inch) Document Resolution (do not change).

Since:
3.0.000 (2008-03-27)

TCPDF::$DrawColor [protected]

Commands for drawing color.

TCPDF::$efnames = array() [protected]

Embedded Files Names

Since:
5.9.204 (2013-01-23)

TCPDF::$embeddedfiles = array() [protected]

Array of files to embedd.

Since:
4.4.000 (2008-12-07)

TCPDF::$empty_signature_appearance = array() [protected]

Array of empty digital signature appearances.

Since:
5.9.101 (2011-07-06)

TCPDF::$emptypagemrk = array() [protected]

Array used to store page positions to track empty pages (keys are the page numbers).

Since:
5.8.007 (2010-08-18)

TCPDF::$encoding = 'UTF-8' [protected]

Default encoding.

Since:
1.53.0.TC010

TCPDF::$encryptdata = array() [protected]

Array containing encryption settings.

Since:
5.0.005 (2010-05-11)

TCPDF::$encrypted `[protected]`

IBoolean flag indicating whether document is protected.

Since:
2.0.000 (2008-01-02)

TCPDF::$endlinex = 0 `[protected]`

End position of the latest inserted line.

Since:
3.2.000 (2008-07-01)

TCPDF::$epsmarker = 'x#!#EPS#!#x' `[protected]`

String used to mark the beginning and end of EPS image blocks.

Since:
4.1.000 (2008-10-18)

TCPDF::$extgstates `[protected]`

Array of transparency objects and parameters.

Since:
3.0.000 (2008-03-27)

TCPDF::$feps = 0.005 `[protected]`

Epsilon value used for float calculations.

Since:
4.2.000 (2008-10-29)

TCPDF::$fgcolor `[protected]`

Current foreground color.

TCPDF::$fhPt `[protected]`

Height of page format in points.

TCPDF::$file_id `[protected]`

File ID (used on document trailer).

Since:
5.0.005 (2010-05-12)

TCPDF::$FillColor `[protected]`

Commands for filling color.

TCPDF::$font_obj_ids = array() `[protected]`

Store the font object IDs.

Since:
4.8.001 (2009-09-09)

TCPDF::$font_spacing = 0 `[protected]`

Increases or decreases the space between characters in a text by the specified amount (tracking).

Since:
5.9.000 (2010-09-29)

TCPDF::$font_stretching = 100 `[protected]`

Percentage of character stretching.

Since:
5.9.000 (2010-09-29)

TCPDF::$font_subsetting = true `[protected]`

Boolean flag: if true enables font subsetting by default.

Since:
5.3.002 (2010-06-07)

TCPDF::$FontAscent `[protected]`

Current font ascent (distance between font top and baseline).

Since:
2.8.000 (2007-03-29)

TCPDF::$FontDescent `[protected]`

Current font descent (distance between font bottom and baseline).

Since:
2.8.000 (2007-03-29)

TCPDF::$FontFamily `[protected]`

Current font family.

TCPDF::$FontFiles = array() `[protected]`

Array of font files.

TCPDF::$fontkeys = array() `[protected]`

Store the font keys.

Since:
4.5.000 (2009-01-02)

TCPDF::$fontlist = array() `[protected]`

List of available fonts on filesystem.

TCPDF::$fonts = array() `[protected]`

Array of used fonts.

TCPDF::$FontSize `[protected]`

Current font size in user unit.

TCPDF::$FontSizePt `[protected]`

Current font size in points.

TCPDF::$FontStyle `[protected]`

Current font style.

TCPDF::$footer_font `[protected]`

Default font used on page footer.

TCPDF::$footer_line_color = array(0,0,0) `[protected]`

Color for footer line (RGB array).

Since:
5.9.174 (2012-07-25)

TCPDF::$footer_margin `[protected]`

Minimum distance between footer and bottom page margin.

TCPDF::$footer_text_color = array(0,0,0) `[protected]`

Color for footer text (RGB array).

Since:
5.9.174 (2012-07-25)

TCPDF::$footerlen = array() `[protected]`

Array used to store footer length of each page.

Since:
4.0.014 (2008-07-29)

TCPDF::$footerpos = array() `[protected]`

Array used to store footer positions of each page.

Since:
3.2.000 (2008-07-01)

TCPDF::$force_srgb = false `[protected]`

If true force sRGB color profile for all document.

Since:
5.9.121 (2011-09-28)

TCPDF::$form_action = '' `[protected]`

Current form action (used during XHTML rendering).

Since:
4.8.000 (2009-09-07)

TCPDF::$form_enctype = 'application/x-www-form-urlencoded' `[protected]`

Current form encryption type (used during XHTML rendering).

Since:
4.8.000 (2009-09-07)

TCPDF::$form_mode = 'post' `[protected]`

Current method to submit forms.

Since:
4.8.000 (2009-09-07)

TCPDF::$form_obj_id = array() `[protected]`

List of form annotations IDs.

Since:
4.8.000 (2009-09-07)

TCPDF::$fwPt `[protected]`

Width of page format in points.

TCPDF::$gdgammacache = array() `[protected]`

Cache array for computed GD gamma values.

Since:
5.9.1632 (2012-06-05)

TCPDF::$gradients = array() `[protected]`

Array for storing gradient information.

Since:
3.1.000 (2008-06-09)

TCPDF::$h `[protected]`

Current height of page in user unit.

TCPDF::$header_font `[protected]`

Default font used on page header.

TCPDF::$header_line_color = array(0,0,0) `[protected]`

Color for header line (RGB array).

Since:
5.9.174 (2012-07-25)

TCPDF::$header_logo = '' `[protected]`

Header image logo.

TCPDF::$header_logo_width = 30 `[protected]`

Width of header image logo in user units.

TCPDF::$header_margin `[protected]`

Minimum distance between header and top page margin.

TCPDF::$header_string = '' `[protected]`

String to pring on page header after title.

TCPDF::$header_text_color = array(0,0,0) `[protected]`

Color for header text (RGB array).

Since:
5.9.174 (2012-07-25)

TCPDF::$header_title = '' `[protected]`

Title to be printed on default page header.

TCPDF::$header_xobj_autoreset = false `[protected]`

If true reset the Header Xobject template at each page

TCPDF::$header_xobjid = false `[protected]`

ID of the stored default header template (-1 = not set).

TCPDF::$hPt `[protected]`

Current height of page in points.

TCPDF::$HREF = array() `[protected]`

HTML PARSER: array to store current link and rendering styles.

TCPDF::$htmlLinkColorArray = array(0, 0, 255) `[protected]`

Default color for html links.

Since:
4.4.003 (2008-12-09)

TCPDF::$htmlLinkFontStyle = 'U' `[protected]`

Default font style to add to html links.

Since:
4.4.003 (2008-12-09)

TCPDF::$htmlvspace = 0 `[protected]`

Count the latest inserted vertical spaces on HTML.

Since:
4.0.021 (2008-08-24)

TCPDF::$imagekeys = array() `[protected]`

Store the image keys.

Since:
4.5.000 (2008-12-31)

TCPDF::$images = array() `[protected]`

Array of used images.

TCPDF::$img_rb_x `[protected]`

The right-bottom (or left-bottom for RTL) corner X coordinate of last inserted image.

Since:
2002-07-31

Author:
Nicola Asuni

TCPDF::$img_rb_y `[protected]`

The right-bottom corner Y coordinate of last inserted image.

Since:
2002-07-31

Author:
Nicola Asuni

TCPDF::$imgscale = 1 `[protected]`

Adjusting factor to convert pixels to user units.

Since:
2004-06-14

Author:
Nicola Asuni

TCPDF::$InFooter = false `[protected]`

Flag set when processing page footer.

TCPDF::$InHeader = false `[protected]`

Flag set when processing page header.

TCPDF::$internal_encoding `[protected]`

PHP internal encoding.

Since:
1.53.0.TC016

TCPDF::$inthead = false `[protected]`

True when we are printing the thead section on a new page.

Since:
4.8.027 (2010-01-25)

TCPDF::$intmrk = array() `[protected]`

Array used to store positions inside the pages buffer (keys are the page numbers).

Since:
3.2.000 (2008-06-26)

TCPDF::$inxobj = false `[protected]`

Boolean value true when we are inside an XObject.

Since:
5.8.017 (2010-08-24)

TCPDF::$isunicode = false `[protected]`

Boolean flag set to true when the input text is unicode (require unicode fonts).

Since:
2005-01-02

Author:
Nicola Asuni

TCPDF::$javascript = '' `[protected]`

Javascript code.

Since:
2.1.002 (2008-02-12)

TCPDF::$jpeg_quality `[protected]`

Set the default JPEG compression quality (1-100).

Since:
3.0.000 (2008-03-27)

TCPDF::$js_objects = array() [protected]

Javascript objects array.

Since:
4.8.000 (2009-09-07)

TCPDF::$k [protected]

Scale factor (number of points in user unit).

TCPDF::$keywords = '' [protected]

Document keywords.

TCPDF::$l [protected]

Language templates.

TCPDF::$last_enc_key [protected]

Last RC4 key encrypted (cached for optimisation).

Since:
2.0.000 (2008-01-02)

TCPDF::$last_enc_key_c [protected]

Last RC4 computed key.

Since:
2.0.000 (2008-01-02)

TCPDF::$lasth [protected]

Height of last cell printed.

TCPDF::$LayoutMode [protected]

Layout display mode.

TCPDF::$linestyleCap = '0 J' [protected]

PDF string for CAP value of the last line.

Since:
4.0.006 (2008-07-16)

TCPDF::$linestyleDash = '[] 0 d' [protected]

PDF string for dash value of the last line.

Since:
4.0.006 (2008-07-16)

TCPDF::$linestyleJoin = '0 j' [protected]

PDF string for join value of the last line.

Since:
4.0.006 (2008-07-16)

TCPDF::$linestyleWidth = '' [protected]

PDF string for width value of the last line.

Since:

4.0.006 (2008-07-16)

TCPDF::$linethrough `[protected]`

line through state

Since:
2.8.000 (2008-03-19)

TCPDF::$LineWidth `[protected]`

Line width in user unit.

TCPDF::$links = array() `[protected]`

Array of internal links.

TCPDF::$lispacer = '' `[protected]`

Spacer string for LI tags.

TCPDF::$listcount = array() `[protected]`

HTML PARSER: array count list items on nested lists.

TCPDF::$listindent = 0 `[protected]`

HTML PARSER: indent amount for lists.

TCPDF::$listindentlevel = 0 `[protected]`

HTML PARSER: current list indententation level.

TCPDF::$listnum = 0 `[protected]`

HTML PARSER: current list nesting level.

TCPDF::$listordered = array() `[protected]`

HTML PARSER: array of boolean values, true in case of ordered list (OL), false otherwise.

TCPDF::$lisymbol = '' `[protected]`

Symbol used for HTML unordered list items.

Since:
4.0.028 (2008-09-26)

TCPDF::$lMargin `[protected]`

Left margin.

TCPDF::$maxselcol = array('page' => 0, 'column' => 0) `[protected]`

Maximum page and column selected.

Since:
5.8.000 (2010-08-11)

TCPDF::$n `[protected]`

Current object number.

TCPDF::$n_dests `[protected]`

Object ID for Named Destinations

Since:
5.9.097 (2011-06-23)

TCPDF::$n_js `[protected]`

Javascript counter.

Since:
2.1.002 (2008-02-12)

TCPDF::$newline = true `[protected]`

Boolean flag to indicate if a new line is created.

Since:
3.2.000 (2008-07-01)

TCPDF::$newpagegroup = array() `[protected]`

Array of page numbers were a new page group was started (the page numbers are the keys of the array).

Since:
3.0.000 (2008-03-27)

TCPDF::$num_columns = 1 `[protected]`

Number of colums.

Since:
4.9.001 (2010-03-28)

TCPDF::$numfonts = 0 `[protected]`

Counts the number of fonts.

Since:
4.5.000 (2009-01-02)

TCPDF::$numimages = 0 `[protected]`

Counts the number of pages.

Since:
4.5.000 (2008-12-31)

TCPDF::$numpages = 0 `[protected]`

Counts the number of pages.

Since:
4.5.000 (2008-12-31)

TCPDF::$objcopy `[protected]`

Cloned copy of the current class object.

Since:
4.5.029 (2009-03-19)

TCPDF::$offsets = array() `[protected]`

Array of object offsets.

TCPDF::$opencell = true `[protected]`

Boolean flag to indicate if the border of the cell sides that cross the page should be removed.

Since:
4.2.010 (2008-11-14)

TCPDF::$openMarkedContent = false `[protected]`

Boolean flag to indicate if marked-content sequence is open.

Since:
4.0.013 (2008-07-28)

TCPDF::$original_lMargin `[protected]`

Original left margin value.

Since:
1.53.0.TC013

TCPDF::$original_rMargin `[protected]`

Original right margin value.

Since:
1.53.0.TC013

TCPDF::$OutlineRoot `[protected]`

Outline root for bookmark.

Since:
2.1.002 (2008-02-12)

TCPDF::$outlines = array() `[protected]`

Outlines for bookmark.

Since:
2.1.002 (2008-02-12)

TCPDF::$overline `[protected]`

Overlining flag.

TCPDF::$overprint = array('OP' => false, 'op' => false, 'OPM' => 0) `[protected]`

Overprint mode array. (Check the "Entries in a Graphics State Parameter Dictionary" on PDF 32000-1:2008).

Since:
5.9.152 (2012-03-23)

TCPDF::$page `[protected]`

Current page number.

TCPDF::$page_boxes = array('MediaBox', 'CropBox', 'BleedBox', 'TrimBox', 'ArtBox') `[protected]`

Define the page boundaries boxes to be set on document.

Since:
5.9.152 (2012-03-23)

TCPDF::$page_obj_id = array() `[protected]`

ID of page objects.

Since:
4.7.000 (2009-08-29)

TCPDF::$page_regions = array() `[protected]`

Array of no-write regions. ('page' => page number or empy for current page, 'xt' => X top, 'yt' => Y top, 'xb' => X bottom, 'yb' => Y bottom, 'side' => page side 'L' = left or 'R' = right)

Since:
5.9.003 (2010-10-14)

TCPDF::$PageAnnots = array() `[protected]`

Array of Annotations in pages.

TCPDF::$PageBreakTrigger `[protected]`

Threshold used to trigger page breaks.

TCPDF::$pagedim = array() `[protected]`

Page dimensions.

TCPDF::$pagegroups = array() `[protected]`

Array that contains the number of pages in each page group.

Since:
3.0.000 (2008-03-27)

TCPDF::$pagelen = array() `[protected]`

Array containing page lengths in bytes.

Since:
4.5.000 (2008-12-31)

TCPDF::$PageMode `[protected]`

A name object specifying how the document should be displayed when opened.

Since:
3.1.000 (2008-06-09)

TCPDF::$pageobjects = array() `[protected]`

Array of object IDs for each page.

TCPDF::$pageopen = array() `[protected]`

Store the fage status (true when opened, false when closed).

Since:
4.5.000 (2009-01-02)

TCPDF::$pages = array() `[protected]`

Array containing pages.

TCPDF::$pdfa_mode = false `[protected]`

If true set the document to PDF/A mode.

Since:
5.9.121 (2011-09-27)

TCPDF::$pdflayers = array() `[protected]`

Array of PDF layers data.

Since:
5.9.102 (2011-07-13)

TCPDF::$pdfunit = 'mm' `[protected]`

Default unit of measure for document.

Since:
5.0.000 (2010-04-22)

TCPDF::$PDFVersion = '1.7' `[protected]`

PDF version.

Since:
1.5.3

TCPDF::$premode = false `[protected]`

Boolean flag to indicate if we are inside a PRE tag.

Since:
4.4.001 (2008-12-08)

TCPDF::$print_footer = true `[protected]`

Boolean flag to print/hide page footer.

TCPDF::$print_header = true `[protected]`

Boolean flag to print/hide page header.

TCPDF::$radio_groups = array() `[protected]`

List of radio group objects IDs.

Since:
4.8.001 (2009-09-09)

TCPDF::$radiobutton_groups = array() `[protected]`

List of radio buttons parent objects.

Since:
4.8.001 (2009-09-09)

TCPDF::$rasterize_vector_images = false `[protected]`

Boolean flag: if true convert vector images (SVG, EPS) to raster image using GD or ImageMagick library.

Since:
5.0.000 (2010-04-26)

TCPDF::$re_space = array('p' => '[^\S\xa0]', 'm' => '') `[protected]`

Array of $re_spaces parts.

Since:
5.5.011 (2010-07-09)

TCPDF::$re_spaces = '/[^\S\xa0]/' `[protected]`

Regular expression used to find blank characters (required for word-wrapping).

Since:
4.6.006 (2009-04-28)

TCPDF::$rMargin `[protected]`

Right margin.

TCPDF::$rtl = false [protected]

Boolean flag to indicate if the document language is Right-To-Left.

Since:
2.0.000

TCPDF::$sig_obj_id = 0 [protected]

Digital signature object ID.

Since:
4.6.022 (2009-06-23)

TCPDF::$sign = false [protected]

Boolean flag to enable document digital signature.

Since:
4.6.005 (2009-04-24)

TCPDF::$signature_appearance = array('page' => 1, 'rect' => '0 0 0 0') [protected]

Data for digital signature appearance.

Since:
5.3.011 (2010-06-16)

TCPDF::$signature_data = array() [protected]

Digital signature data.

Since:
4.6.005 (2009-04-24)

TCPDF::$signature_max_length = 11742 [protected]

Digital signature max length.

Since:
4.6.005 (2009-04-24)

TCPDF::$spot_colors = array() [protected]

Array of Spot colors.

Since:
4.0.024 (2008-09-12)

TCPDF::$start_transaction_page = 0 [protected]

Store page number when **startTransaction()** is called.

Since:
4.8.006 (2009-09-23)

TCPDF::$start_transaction_y = 0 [protected]

Store Y position when **startTransaction()** is called.

Since:
4.9.001 (2010-03-28)

TCPDF::$starting_page_number = 1 [protected]

Starting page number.

TCPDF::$state `[protected]`

Current document state.

TCPDF::$strokecolor `[protected]`

Current stroke color.

Since:
4.9.008 (2010-04-03)

TCPDF::$subject = '' `[protected]`

Document subject.

TCPDF::$svgclipid = 0 `[protected]`

ID of last SVG clipPath.

Since:
5.0.000 (2010-05-02)

TCPDF::$svgclipmode = false `[protected]`

Boolean value true when in SVG clipPath tag.

Since:
5.0.000 (2010-04-26)

TCPDF::$svgclippaths = array() `[protected]`

Array of SVG clipPath commands.

Since:
5.0.000 (2010-05-02)

TCPDF::$svgcliptm = array() `[protected]`

Array of SVG clipPath tranformation matrix.

Since:
5.8.022 (2010-08-31)

TCPDF::$svgdefs = array() `[protected]`

Array of SVG defs.

Since:
5.0.000 (2010-05-02)

TCPDF::$svgdefsmode = false `[protected]`

Boolean value true when in SVG defs group.

Since:
5.0.000 (2010-05-02)

TCPDF::$svgdir = '' `[protected]`

Directory used for the last SVG image.

Since:
5.0.000 (2010-05-05)

TCPDF::$svggradientid = 0 `[protected]`

ID of last SVG gradient.

Since:

5.0.000 (2010-05-02)

TCPDF::$svggradients = array() `[protected]`

Array of SVG gradients.

Since:
5.0.000 (2010-05-02)

TCPDF::$svgstyles `[protected]`

Array of SVG properties.

Since:
5.0.000 (2010-05-02)

TCPDF::$svgtext = '' `[protected]`

SVG text.

Since:
5.0.000 (2010-05-02)

TCPDF::$svgtextmode = array() `[protected]`

SVG text properties.

Since:
5.8.013 (2010-08-23)

TCPDF::$svgunit = 'px' `[protected]`

Deafult unit of measure for SVG.

Since:
5.0.000 (2010-05-02)

TCPDF::$tagvspaces = array() `[protected]`

Array used for custom vertical spaces for HTML tags.

Since:
4.2.001 (2008-10-30)

TCPDF::$tcpdflink = true `[protected]`

If true print **TCPDF** meta link.

Since:
5.9.152 (2012-03-23)

TCPDF::$tempfontsize = 10 `[protected]`

Temporary font size in points.

TCPDF::$TextColor `[protected]`

Commands for text color.

TCPDF::$textindent = 0 `[protected]`

Text indentation value (used for text-indent CSS attribute).

Since:
4.8.006 (2009-09-23)

TCPDF::$textrendermode = 0 `[protected]`

Text rendering mode: 0 = Fill text; 1 = Stroke text; 2 = Fill, then stroke text; 3 = Neither fill nor stroke text (invisible); 4 = Fill text and add to path for clipping; 5 = Stroke text and add to path for clipping; 6 = Fill, then stroke text and add to path for clipping; 7 = Add text to path for clipping.

Since:
4.9.008 (2010-04-03)

TCPDF::$textstrokewidth = 0 `[protected]`

Text stroke width in doc units.

Since:
4.9.008 (2010-04-03)

TCPDF::$thead = '' `[protected]`

Table header content to be repeated on each new page.

Since:
4.5.030 (2009-03-20)

TCPDF::$theadMargins = array() `[protected]`

Margins used for table header.

Since:
4.5.030 (2009-03-20)

TCPDF::$title = '' `[protected]`

Document title.

TCPDF::$tMargin `[protected]`

Top margin.

TCPDF::$tmprtl = false `[protected]`

Boolean flag used to force RTL or LTR string direction.

Since:
2.0.000

TCPDF::$tocpage = false `[protected]`

Boolean flag true when we are on TOC (Table Of Content) page.

TCPDF::$transfmatrix = array() `[protected]`

Array of transformation matrix.

Since:
4.2.000 (2008-10-29)

TCPDF::$transfmatrix_key = 0 `[protected]`

Current key for transformation matrix.

Since:
4.8.005 (2009-09-17)

TCPDF::$transfmrk = array() `[protected]`

Array used to store positions of graphics transformation blocks inside the page buffer. keys are the page numbers

Since:
4.4.002 (2008-12-09)

TCPDF::$txtshadow = array('enabled'=>false, 'depth_w'=>0, 'depth_h'=>0, 'color'=>false, 'opacity'=>1, 'blend_mode'=>'Normal') `[protected]`

Text shadow data array.

Since:
5.9.174 (2012-07-25)

TCPDF::$underline `[protected]`

Underlining flag.

TCPDF::$ur = array() `[protected]`

Array with additional document-wide usage rights for the document.

Since:
5.8.014 (2010-08-23)

TCPDF::$viewer_preferences `[protected]`

PDF viewer preferences.

Since:
3.1.000 (2008-06-09)

TCPDF::$w `[protected]`

Current width of page in user unit.

TCPDF::$wPt `[protected]`

Current width of page in points.

TCPDF::$x `[protected]`

Current horizontal position in user unit for cell positioning.

TCPDF::$xobjects = array() `[protected]`

Array of XObjects.

Since:
5.8.014 (2010-08-23)

TCPDF::$xobjid = '' `[protected]`

Current XObject ID.

Since:
5.8.017 (2010-08-24)

TCPDF::$y `[protected]`

Current vertical position in user unit for cell positioning.

TCPDF::$ZoomMode `[protected]`

Zoom display mode.

The documentation for this class was generated from the following file:

- /home/nick/WWW/_OPENSOURCE/tcpdf/**tcpdf.php**

EN IT

W3C XHTML 1.0 W3C CSS 2.0