

Classe Fpdf

Table des matières

1. Contenu	4
2. Définition.....	4
3. Constantes	4
4. Méthodes	4
__construct	4
AcceptPageBreak	5
AddFont	6
AddLink	7
AddPage.....	7
AliasNbPages	8
Cell	8
Close.....	9
Error	10
Footer	10
GetPageHeight.....	11
GetPageWidth.....	11
GetStringWidth	11
GetX.....	11
GetY.....	12
Header	12
Image	12
Line.....	14
Link.....	14
Ln	14
MultiCell	15
Output	16
PageNo	16
Rect	16
SetAuthor	17
SetAutoPageBreak.....	17
SetCompression	17
SetCreator	18
SetDisplayMode	18
SetDrawColor.....	19
SetFillColor	19
SetFont	19
SetFontSize	21
SetKeywords.....	21

Classe Fpdf

SetLeftMargin	21
SetLineWidth	21
SetLink	22
SetMargins	22
SetRightMargin	22
SetSubject	23
SetTextColor	23
SetTitle.....	23
SetTopMargin	24
SetX	24
SetXY	24
SetY	24
Text	25
Write.....	25

1. Contenu

Ce document à pour but d'expliquer et de définir l'usage de la classe **Fpdf** de l'outil **UniversalWeb**.

2. Définition

La classe **Fpdf** est une classe PHP écrite par Olivier PLATHEY disponible sur son site : <http://www.fpdf.org/>. Elle permet de générer des fichiers PDF en pur PHP, c'est-à-dire sans utiliser la librairie **PDFlib**.

Voici une liste de ses principales fonctionnalités :

- Choix des unités, du format des pages et des marges
- Gestion des en-têtes et pieds de page
- Saut de page automatique
- Saut de ligne automatique et justification
- Images (JPEG, PNG et GIF)
- Couleurs
- Liens
- Support des polices TrueType et Type1
- Compression des pages

La version mise à disposition pour UniversalWeb est la dernière en date, c'est-à-dire la V1.81 du 20 décembre 2015. A ce titre son code a été quelque peu adapté, voir amélioré pour quelques méthodes afin d'être compatible avec le code général produit par UniversalWeb, c'est-à-dire compatible avec la version de PHP 7.2

Vous trouverez sur le site d'Olivier PLATHEY plusieurs tutoriels très intéressant pour mettre en œuvre cette classe.

3. Constantes

Constantes disponibles :

Nom de la constante	Valeur
VERSION	Fournit la version de la classe. On peut utiliser cette constante via l'appel statique : <code>Fpdf::VERSION</code>

4. Méthodes

`__construct`

Constructeur de la classe. Il permet de fixer le format des pages, leur orientation par défaut ainsi que l'unité de mesure utilisée dans toutes les méthodes (sauf pour les tailles de police).

Description

```
__construct([string orientation [, string unit [, mixed size]])
```

Liste des paramètres

orientation : Orientation des pages par défaut. Les valeurs possibles sont (indépendamment de la casse) :

- P ou Portrait (valeur par défaut)
- L ou Landscape

unit : Unité de mesure utilisateur. Les valeurs possibles sont :

- pt : point
- mm : millimètre (valeur par défaut)
- cm : centimètre
- in : pouce

Un point est égal à 1/72ème de pouce, c'est-à-dire environ 0,35 millimètre. C'est une unité très courante en typographie; les tailles de police sont exprimées dans cette unité. Un pouce vaut 2,54 centimètres.

size : Le format utilisé pour les pages. Il peut s'agir d'une des valeurs ci-dessous (indépendamment de la casse) :

- A3
- A4 (valeur par défaut)
- A5
- Letter
- Legal

ou bien d'un tableau contenant la largeur et la hauteur (exprimées dans l'unité donnée par unit).

Valeurs de retour

Objet de la classe Fpdf.

Exemple

Exemple avec un format personnalisé de 100 x 150 mm :

```
$pdf = new Fpdf('P', 'mm', array(100,150));
```

AcceptPageBreak

Accepte ou non un saut de page automatique

Description

```
boolean AcceptPageBreak()
```

Liste des paramètres

Aucun

Valeurs de retour

true / false

Exemple

La méthode est redéfinie dans une classe dérivée afin d'obtenir un formatage sur 3 colonnes :

```
class PDF extends FPDF {  
    var $col = 0;
```

```
public function SetCol($col) {
    // Positionnement sur une colonne
    $this->col = $col;
    $x = 10+$col*65;
    $this->SetLeftMargin($x);
    $this->SetX($x);
}

public function AcceptPageBreak() {
    if($this->col<2) {
        // Aller à la colonne suivante
        $this->SetCol($this->col+1);
        $this->SetY(10);
        return false;
    }
    else {
        // Retour en première colonne et saut de page
        $this->SetCol(0);
        return true;
    }
}

$pdf = new PDF();
$pdf->AddPage();
$pdf->SetFont('Arial','',12);
for($i=1;$i<=300;$i++)
    $pdf->Cell(0,5,"Ligne $i",0,1);
$pdf->Output();
```

Note

Lorsqu'une condition de saut de page est remplie, la méthode est appelée, et en fonction de la valeur de retour, le saut est effectué ou non. L'implémentation par défaut renvoie une valeur selon le mode sélectionné par **SetAutoPageBreak**.

Cette méthode est appelée automatiquement et ne devrait donc pas être appelée directement par l'application.

AddFont

Ajoute une nouvelle police. Importe une police TrueType, OpenType ou Type1 et la rend disponible.

Description

AddFont(**string** \$family [, **string** \$style [, **string** \$file])

Liste des paramètres

family : Famille de la police. Le nom peut être choisi arbitrairement. S'il est celui d'une famille standard, la police correspondante sera masquée.

style : Style de la police. Les valeurs possibles sont (indépendamment de la casse) :

- chaîne vide : normal (valeur par défaut)
- B : gras
- I : italique
- BI ou IB : gras italique

file : Le fichier de définition de police. Par défaut, le nom est construit à partir du nom de la famille et du style, en minuscules et sans espace.

Valeurs de retour

Aucune

Exemple

```
$pdf->AddFont('Comic','I');
```

est équivalent à :

```
$pdf->AddFont('Comic','I','comici.php');
```

Note

Il faut au préalable avoir généré un fichier de définition de police avec l'utilitaire MakeFont. Le fichier de définition (ainsi que le fichier de police en cas d'incorporation) doit être présent dans le répertoire des polices. S'il n'est pas trouvé, l'erreur *"Could not include font definition file"* est renvoyée.

AddLink

Crée un lien interne. Crée un nouveau lien interne et renvoie son identifiant. Un lien interne est une zone cliquable qui amène à un autre endroit dans le document.

Description

```
int AddLink()
```

Liste des paramètres

Aucun

Valeurs de retour

Identifiant du lien

Note

L'identifiant pourra être transmis aux méthodes `Cell`, `Write`, `Image` ou `Link`. La destination est définie à l'aide de `SetLink`.

AddPage

Ajoute une nouvelle page au document.

Si une page était en cours, la méthode `Footer` est appelée pour traiter le pied de page. Puis la page est ajoutée, la position courante mise en haut à gauche en fonction des marges gauche et haute, et `Header` est appelée pour afficher l'en-tête.

Description

```
AddPage([string $orientation [, mixed $size [, int $rotation]])
```

Liste des paramètres

orientation : Orientation de la page. Les valeurs possibles sont (indépendamment de la casse) :

- P ou Portrait
- L ou Landscape

La valeur par défaut est celle indiquée dans le constructeur.

size : Format de la page. Il peut s'agir d'une des valeurs ci-dessous (indépendamment de la casse) ou bien d'un tableau contenant la largeur et la hauteur (exprimées en unité utilisateur) :

- A3
- A4
- A5
- Letter
- Legal

La valeur par défaut est celle indiquée dans le constructeur.

Classe Fpdf

rotation : Angle de rotation de la page. La valeur doit être un multiple de 90 ; la rotation se fait dans le sens horaire. La valeur par défaut est 0.

Valeurs de retour

Aucune

Note

La police qui était en cours au moment de l'appel est automatiquement restaurée. Il n'est donc pas nécessaire d'appeler à nouveau **setFont** si vous souhaitez continuer avec la même police. Même chose pour les couleurs et l'épaisseur du trait.

L'origine du système de coordonnées est en haut à gauche et les ordonnées croissantes vont vers le bas.

AliasNbPages

Définit un alias pour le nombre de pages. Cet alias sera substitué lors de la fermeture du document.

Description

AliasNbPages([string \$alias])

Liste des paramètres

alias : L'alias (valeur par défaut {nb})

Valeurs de retour

Aucune

Exemple

```
class PDF extends FPDF {
    function Footer() {
        // Positionnement à 1,5 cm du bas
        $this->SetY(-15);
        // Police Arial italique 8
        $this->SetFont('Arial','I',8);
        // Numéro et nombre de pages
        $this->Cell(0,10,'Page '.$this->PageNo().'/{nb}',0,0,'C');
    }
}

$pdf = new PDF(); $pdf->AliasNbPages();
```

Cell

Imprime une cellule (zone rectangulaire) avec éventuellement des bords, un fond et une chaîne de caractères.

Le coin supérieur gauche de la cellule correspond à la position courante. Le texte peut être aligné ou centré.

Description

Cell(float \$w [, float \$h [, string \$txt [, mixed \$border [, int \$ln [, string \$align [, boolean \$fill [, mixed \$link]]]]]])

Liste des paramètres

w : Largeur de la cellule. Si elle vaut 0, la cellule s'étend jusqu'à la marge droite de la page.

h : hauteur de la cellule. Valeur par défaut : 0.

txt : Chaîne à imprimer (Valeur par défaut : chaîne vide).

border : Indique si des bords doivent être tracés autour de la cellule. La valeur peut être soit un nombre soit une chaîne contenant certains ou tous les caractères suivants (dans un ordre quelconque) :

- 0 : aucun bord (valeur par défaut)
- 1 : cadre
- L : gauche
- T : haut
- R : droit
- B : bas

ln : Indique où se déplace la position courante après l'appel à la méthode. Valeurs possibles :

- 0 : à droite (valeur par défaut)
- 1 : au début de la ligne suivante
- 2 : en dessous

Mettre 1 est équivalent à mettre 0 et appeler la méthode **Ln** juste après.

align : Permet de centrer ou d'aligner le texte. Les valeurs possibles sont :

1. L ou chaîne vide : alignement à gauche (valeur par défaut)
2. C : centrage
3. R : alignement à droite

fill : Indique si le fond de la cellule doit être coloré (true) ou transparent (false). Valeur par défaut : false.

link : URL ou identifiant retourné par **AddLink**.

Valeurs de retour

Aucune

Exemple

```
// Sélection de la police
$pdf->SetFont('Arial','B',16);
// Décalage de 8 cm à droite
$pdf->Cell(80);
// Texte centré dans une cellule 20*10 mm encadrée et retour à la ligne
$pdf->Cell(20,10,'Titre',1,1,'C');
```

Notes

Après l'appel, la position courante se déplace à droite ou un retour à la ligne est effectué.

Il est possible de mettre un lien sur le texte.

Si le saut de page automatique est activé et que la cellule dépasse le seuil de déclenchement, un saut de page est effectué avant de l'imprimer.

Close

Termine le document.

Description

Close()

Liste des paramètres

Aucun

Valeurs de retour

Aucune

Notes

Il n'est pas nécessaire de l'appeler explicitement car `Output` le fait automatiquement.

Si le document ne contient aucune page, `AddPage` est appelé pour éviter d'obtenir un document invalide.

Error

Erreur fatale.

Description

`Error(string $msg)`

Liste des paramètres

`msg` : le message d'erreur

Valeurs de retour

Aucune

Notes

Cette méthode est appelée automatiquement en cas d'erreur fatale ; elle se contente de lancer une exception avec le message fourni.

Une classe dérivée peut redéfinir la méthode afin de personnaliser le traitement des erreurs, mais elle doit impérativement interrompre le traitement, sinon le fichier PDF résultant serait probablement invalide.

Footer

Cette méthode permet de définir le pied de page.

Description

`Footer()`

Liste des paramètres

Aucun

Valeurs de retour

Aucune

Exemple

```
class PDF extends FPDF {
    function Footer() {
        // Positionnement à 1,5 cm du bas
        $this->SetY(-15);
        // Police Arial italique 8
        $this->SetFont('Arial','I',8);
        // Numéro de page centré
        $this->Cell(0,10,'Page '.$this->PageNo(),0,0,'C');
    }
}
```

Notes

Classe Fpdf

Cette méthode est appelée automatiquement par `AddPage` et `Close` et ne devrait donc pas être appelée explicitement par l'application.

L'implémentation de `Footer` dans FPDF est vide, donc vous devez dériver la classe et redéfinir la méthode si vous voulez un traitement particulier pour vos pieds de page.

GetPageHeight

Renvoie la hauteur de la page courante.

Description

```
float GetPageHeight()
```

Liste des paramètres

Aucun

Valeurs de retour

Aucune

GetPageWidth

Renvoie la largeur de la page courante.

Description

```
float GetPageWidth()
```

Liste des paramètres

Aucun

Valeurs de retour

Aucune

GetStringWidth

Calcule et renvoie la longueur d'une chaîne en unité utilisateur. Une police doit être sélectionnée.

Description

```
float GetStringWidth(string $s)
```

Liste des paramètres

`s` : La chaîne dont on veut déterminer la longueur.

Valeurs de retour

Aucune

GetX

Renvoie l'abscisse de la position courante.

Description

```
float GetX()
```

Liste des paramètres

Aucun

Valeurs de retour

Aucune

GetY

Renvoie l'ordonnée de la position courante.

Description

```
float GetY()
```

Liste des paramètres

Aucun

Valeurs de retour

Aucune

Header

Cette méthode permet de définir l'en-tête de page.

Description

```
Header()
```

Liste des paramètres

Aucun

Valeurs de retour

Aucune

Exemple

```
class PDF extends FPDF {  
    function Header() {  
        // Police Arial gras 15  
        $this->SetFont('Arial','B',15);  
        // Décalage  
        $this->Cell(80);  
        // Titre encadré  
        $this->Cell(30,10,'Titre',1,0,'C');  
        // Saut de ligne  
        $this->Ln(20);  
    }  
}
```

Notes

Elle est appelée automatiquement par `AddPage` et ne devrait donc pas être appelée explicitement par l'application. L'implémentation de `Header` dans FPDF est vide, donc vous devez dériver la classe et redéfinir la méthode si vous voulez un traitement particulier pour vos en-têtes.

Image

Place une image.

Les dimensions occupées dans la page peuvent être indiquées de plusieurs manières :

- largeur et hauteur explicites (exprimées dans l'unité utilisateur ou en dpi)
- une dimension explicite, l'autre étant calculée automatiquement afin de respecter les proportions de l'image originale
- aucune dimension explicite, auquel cas l'image est dimensionnée en 96 dpi

Les formats supportés sont le JPEG, le PNG et le GIF. Le GIF nécessite l'extension GD.

Pour le JPEG, toutes les variantes sont autorisées :

- niveaux de gris
- couleurs vraies (24 bits)
- CMYK (32 bits)

Pour le PNG, sont autorisées :

- les images en niveaux de gris sur 8 bits au plus (256 teintes)
- les images en couleurs indexées
- les images en couleurs vraies (24 bits)

Pour le GIF : en cas de GIF animé, seule la première image est affichée.

La transparence est gérée.

Le format peut être spécifié explicitement ou bien déduit de l'extension du fichier.

Il est possible de mettre un lien sur l'image.

Description

```
Image(string $file [, float $x [, float $y [, float $w [, float $h [,  
string $type [, mixed $link]]]]])
```

Liste des paramètres

file : Chemin ou URL de l'image.

x : Abscisse du coin supérieur gauche. Si non précisée ou égale à null, l'abscisse courante est utilisée.

y : Ordonnée du coin supérieur gauche. Si non précisée ou égale à null, l'ordonnée courante est utilisée ; de plus, un saut de page est d'abord effectué si nécessaire (en cas de saut de page automatique) ; puis, après l'appel, l'ordonnée courante est positionnée en bas de l'image.

w : Largeur de l'image dans la page. Il y a trois cas possibles :

- Si la valeur est positive, elle représente la largeur en unité utilisateur
- Si la valeur est négative, sa valeur absolue représente la résolution horizontale en dpi
- Si elle n'est pas indiquée ou vaut zéro, elle est calculée automatiquement

h : Hauteur de l'image dans la page. Il y a trois cas possibles :

- Si la valeur est positive, elle représente la hauteur en unité utilisateur
- Si la valeur est négative, sa valeur absolue représente la résolution verticale en dpi
- Si elle n'est pas indiquée ou vaut zéro, elle est calculée automatiquement

type : Format de l'image. Les valeurs possibles sont (indépendamment de la casse) : JPG, JPEG, PNG et GIF. S'il n'est pas précisé, le type est déduit de l'extension du fichier.

link : URL ou identifiant retourné par `AddLink`.

Valeurs de retour

Aucune

Exemple

```
// Insère un logo en haut à gauche à 300 dpi
$pdf->Image('logo.png', 10, 10, -300);
// Insère une image dynamique à partir d'une URL
$pdf->Image('http://chart.googleapis.com/chart?cht=p3&chd=t:60,40&chs=250x100&chl=Hello|World', 60, 30, 90, 0, 'PNG');
```

Notes

Si une image est utilisée plusieurs fois, une seule copie est intégrée au document.

Line

Trace une ligne entre deux points

Description

`Line(float $x1, float $y1, float $x2, float $y2)`

Liste des paramètres

x1 : Abscisse du premier point.
y1 : Ordonnée du premier point.
x2 : Abscisse du second point.
y2 : Ordonnée du second point.

Valeurs de retour

Aucune

Link

Place un lien sur une zone rectangulaire de la page.

Description

`Link(float $x, float $y, float $w, float $h, mixed $link)`

Liste des paramètres

x : Abscisse du coin supérieur gauche du rectangle.
y : Ordonnée du coin supérieur gauche du rectangle.
w : Largeur du rectangle.
h : Hauteur du rectangle.
link : URL ou identifiant retourné par `AddLink`.

Valeurs de retour

Aucune

Note

Les liens textes ou images se posent généralement via `Cell`, `Write` ou `Image`, mais cette méthode peut être utile par exemple pour placer une zone cliquable à l'intérieur d'une image.

Ln

Effectue un saut de ligne. L'abscisse courante est ramenée à la valeur de la marge gauche et

l'ordonnée augmente de la valeur indiquée en paramètre.

Description

`Ln([float $h])`

Liste des paramètres

h : L'amplitude du saut de ligne. Par défaut, la valeur est égale à la hauteur de la dernière cellule imprimée.

Valeurs de retour

Aucune

MultiCell

Cette méthode permet d'imprimer du texte avec des retours à la ligne. Ceux-ci peuvent être automatiques (dès que le texte atteint le bord droit de la cellule) ou explicites (via le caractère \n).

Autant de cellules que nécessaire sont imprimées, les unes en dessous des autres.

Le texte peut être aligné, centré ou justifié. Le bloc de cellules peut être encadré et le fond coloré.

Description

`int MultiCell(float $w, float $h, string $txt [, mixed $border [, string $align [, boolean $fill]]])`

Liste des paramètres

w : Largeur des cellules. Si elle vaut 0, elles s'étendent jusqu'à la marge droite de la page.

h : Hauteur des cellules.

txt : Chaîne à imprimer.

border : Indique si des bords doivent être tracés autour du bloc de cellules. La valeur peut être soit un nombre soit une chaîne contenant certains ou tous les caractères suivants (dans un ordre quelconque) :

- 0 : aucun bord (valeur par défaut)
- 1 : cadre
- L : gauche
- T : haut
- R : droit
- B : bas

align : Contrôle l'alignement du texte. Les valeurs possibles sont :

- L : alignement à gauche
- C : centrage
- R : alignement à droite
- J : justification (valeur par défaut)

fill : Indique si le fond des cellules doit être coloré (`true`) ou transparent (`false`). Valeur par défaut : `false`.

Valeurs de retour

Nombre de lignes écrites dans la cellule

Output

Envoie le document vers une destination donnée : navigateur, fichier ou chaîne de caractères. Dans le cas du navigateur, on peut utiliser le visualiseur PDF ou bien forcer le téléchargement.

Description

```
string Output([string $dest [, string $name [, boolean $isUTF8]])
```

Liste des paramètres

dest : Destination où envoyer le document. Le paramètre peut prendre les valeurs suivantes :

- I : envoyer en inline au navigateur. Le visualiseur PDF est utilisé. (valeur par défaut)
- D : envoyer au navigateur en forçant le téléchargement, avec le nom indiqué dans **name**.
- F : sauver dans un fichier local, avec le nom indiqué dans **name** (peut inclure un répertoire).
- S : renvoyer le document sous forme de chaîne.

name : Le nom du fichier. Il est ignoré dans le cas de la destination S. La valeur par défaut est doc.pdf.

isUTF8 : Indique si **name** est encodé en ISO-8859-1 (**false**) ou en UTF-8 (**true**).

Ce paramètre est utilisé uniquement pour les destinations I et D. La valeur par défaut est false.

Valeurs de retour

Envoi

Note

La méthode commence par appeler **close** si nécessaire pour terminer le document.

PageNo

Renvoie le numéro de page courant.

Description

```
int PageNo()
```

Liste des paramètres

Aucun.

Valeurs de retour

Le numéro de la page courante

Rect

Dessine un rectangle à partir de son coin supérieur gauche. Il peut être tracé, rempli ou les deux à la fois.

Description

```
Rect(float $x, float $y, float $w, float $h [, string $style])
```

Liste des paramètres

x : Abscisse du coin supérieur gauche.

y : Ordonnée du coin supérieur gauche.

w : Largeur.

h : Hauteur.

style : Manière de tracer le rectangle. Les valeurs possibles sont :

- D ou chaîne vide : contour (draw). Valeur par défaut.
- F : remplissage (fill)
- DF ou FD : contour et remplissage

Valeurs de retour

Aucune

SetAuthor

Dessine l'auteur du document.

Description

`SetAuthor(string $author [, boolean $isUTF8])`

Liste des paramètres

author : Le nom de l'auteur.

isUTF8 : Indique si la chaîne est encodée en ISO-8859-1 (`false`) ou en UTF-8 (`true`). Valeur par défaut : `false`.

Valeurs de retour

Aucune

SetAutoPageBreak

Active ou désactive le mode saut de page automatique. En cas d'activation, le second paramètre représente la distance par rapport au bas de la page qui déclenche le saut. Par défaut, le mode est activé et la marge vaut 2 cm.

Description

`SetAutoPageBreak(boolean $auto [, float $margin])`

Liste des paramètres

auto : Booléen indiquant si le mode doit être activé.

margin : Distance par rapport au bas de la page.

Valeurs de retour

Aucune

SetCompression

Active ou désactive la compression des pages. Lorsqu'elle est activée, la représentation interne de chaque page est compressée, ce qui donne généralement un taux de compression de l'ordre de 2 pour le document résultant.

Description

`SetCompression(boolean $compress)`

Liste des paramètres

Classe Fpdf

compress : Booléen indiquant si la compression doit être activée.

Valeurs de retour

Aucune

Notes

La compression est activée par défaut.

l'extension `zlib` est requise pour cette fonctionnalité. Si elle n'est pas disponible, la compression sera désactivée.

SetCreator

Définit le créateur du document. Il s'agit typiquement du nom de l'application qui génère le PDF.

Description

`SetCreator(string $creator [, boolean $isUTF8])`

Liste des paramètres

creator : Le nom du créateur.

isUTF8 : Indique si la chaîne est encodée en ISO-8859-1 (`false`) ou en UTF-8 (`true`). Valeur par défaut : `false`.

Valeurs de retour

Aucune

SetDisplayMode

Contrôle la manière dont le document sera affiché par le lecteur.

On peut régler le niveau de zoom : afficher la page en entier, occuper toute la largeur de la fenêtre, utiliser la taille réelle, choisir un facteur de zoom particulier ou encore utiliser la valeur par défaut de l'utilisateur (celle paramétrée dans le menu Préférences d'Adobe Reader).

On peut également spécifier la disposition des pages : une seule à la fois, affichage continu, pages sur deux colonnes ou réglage par défaut.

Description

`SetDisplayMode(mixed $zoom [, string $layout])`

Liste des paramètres

zoom : Le zoom à utiliser. Il peut prendre l'une des valeurs chaînes suivantes ou bien un entier indiquant le facteur de zoom à utiliser :

- **fullpage** : affiche entièrement les pages à l'écran
- **fullwidth** : maximise la largeur des pages
- **real** : affiche la taille réelle (équivalent à un zoom de 100%)
- **default** : utilise le mode d'affichage par défaut du lecteur

layout : La disposition des pages. Les valeurs possibles sont :

- **single** : affiche une seule page à la fois
- **continuous** : affichage continu d'une page à l'autre
- **two** : affiche deux pages sur deux colonnes
- **default** : utilise le mode d'affichage par défaut du lecteur (valeur par défaut)

Valeurs de retour

Aucune

SetDrawColor

Fixe la couleur pour toutes les opérations de tracé (lignes, rectangles et contours de cellules). Elle peut être indiquée en composantes RGB ou en niveau de gris.

Description

```
SetDrawColor(int $r [, int $g, int $b])
```

Liste des paramètres

r : Si g et b sont renseignés, composante de rouge; sinon, indique le niveau de gris. Valeur comprise entre 0 et 255.

g : Composante de vert (entre 0 et 255).

b : Composante de bleu (entre 0 et 255).

Valeurs de retour

Aucune

Note

La méthode peut être appelée avant que la première page ne soit créée et la valeur est conservée de page en page.

SetFillColor

Fixe la couleur pour toutes les opérations de remplissage (rectangles pleins et fonds de cellules). Elle peut être indiquée en composantes RGB ou en niveau de gris.

Description

```
SetFillColor(int $r [, int $g, int $b])
```

Liste des paramètres

r : Si g et b sont renseignés, composante de rouge; sinon, indique le niveau de gris. Valeur comprise entre 0 et 255.

g : Composante de vert (entre 0 et 255).

b : Composante de bleu (entre 0 et 255).

Valeurs de retour

Aucune

Note

La méthode peut être appelée avant que la première page ne soit créée et la valeur est conservée de page en page.

SetFont

Fixe la police utilisée pour imprimer les chaînes de caractères. Il est obligatoire d'appeler cette méthode au moins une fois avant d'imprimer du texte, sinon le document résultant ne sera pas valide.

La police peut être soit une police standard, soit une police ajoutée à l'aide de la méthode **AddFont**.

Les polices standard utilisent l'encodage Windows cp1252 (Europe de l'ouest).

La méthode peut être appelée avant que la première page ne soit créée et la police est conservée de page en page.

Description

SetFont(**string** \$family [, **string** \$style [, **float** \$size]])

Liste des paramètres

family : Famille de la police. Il peut s'agir d'un nom défini par **AddFont** ou bien d'une des familles standard :

- Courier (caractères de largeur fixe)
- Helvetica ou Arial (synonymes; sans serif)
- Times (avec serif)
- Symbol (symboles)
- ZapfDingbats (symboles)

Le nom n'est pas sensible à la casse.

Il est également possible de passer une chaîne vide, auquel cas la famille courante est conservée.

style : Style de la police. Les valeurs possibles sont (indépendamment de la casse) :

- chaîne vide : normal (valeur par défaut)
- B : gras
- I : italique
- U : souligné
- ou une combinaison quelconque.

Les styles gras et italique ne s'appliquent pas aux familles Symbol et ZapfDingbats.

size : Taille de la police en points. La valeur par défaut est la taille courante. Si aucune taille n'a encore été spécifiée depuis le début du document, la valeur prise est 12.

Valeurs de retour

Aucune

Exemple

```
// Times normal 12
$pdf->SetFont('Times');
// Arial gras 14
$pdf->SetFont('Arial','B',14);
// Enlève le gras
$pdf->SetFont('');
// Times gras, italique et souligné 14
$pdf->SetFont('Times','BIU');
```

Note

- Si vous souhaitez juste changer la taille courante, il est plus simple d'appeler **setFontSize**.
- Il est nécessaire que les fichiers de définition soient accessibles. Il sont recherchés successivement dans :
 - Le répertoire défini par la constante **FPDF_FONTPATH** (si cette constante est définie)
 - Le répertoire **font** situé dans le répertoire de **fpdf.php** (s'il existe)
 - Les répertoires accessibles par **include()**

Exemple d'utilisation de **FPDF_FONTPATH** :

```
define('FPDF_FONTPATH','/home/www/font');
require('fpdf.php');
```

Si le fichier correspondant à la police demandée n'est pas trouvé, l'erreur "*Could not include font definition file*" est générée.

SetFontSize

Fixe la taille de la police courante.

Description

`SetFontSize(float $size)`

Liste des paramètres

`size` : taille en points

Valeurs de retour

Aucune

SetKeywords

Associe des mots-clés au document, généralement sous la forme 'mot-clé1 mot-clé2 ...'.

Description

`SetKeywords(string $keywords [, boolean $isUTF8])`

Liste des paramètres

`keywords` : La liste de mots-clés.

`isUTF8` : Indique si la chaîne est encodée en ISO-8859-1 (false) ou en UTF-8 (true).

Valeur par défaut : false.

Valeurs de retour

Aucune

SetLeftMargin

Fixe la marge gauche. La méthode peut être appelée avant de créer la première page.

Description

`SetLeftMargin(float $margin)`

Liste des paramètres

`margin` : La marge

Valeurs de retour

Aucune

Note

Si l'abscisse courante se retrouve hors page, elle est ramenée à la marge.

SetLineWidth

Fixe l'épaisseur des traits. La valeur est par défaut égale à 0,2 mm. La méthode peut être appelée avant que la première page ne soit créée et la valeur est conservée de page en page.

Description

`SetLineWidth(float $width)`

Liste des paramètres

`width` : L'épaisseur

Valeurs de retour

Aucune

SetLink

Indique la page et la position vers lesquelles pointe un lien interne.

Description

`SetLink(int $link [, float $y [, int $page]])`

Liste des paramètres

`link` : Identifiant du lien retourné par `AddLink()`.

`y` : Ordonnée de la position. -1 indique la position courante. La valeur par défaut est 0 (haut de la page).

`page` : Numéro de la page. -1 indique la page courante. C'est la valeur par défaut.

Valeurs de retour

Aucune

SetMargins

Fixe les marges gauche, haute et droite. Par défaut, elles valent 1 cm. Appelez cette méthode si vous désirez les changer.

Description

`SetMargins(float $left, float $top [, float $right])`

Liste des paramètres

`left` : Marge gauche.

`top` : Marge haute.

`right` : Marge droite. Par défaut, est égale à la gauche.

Valeurs de retour

Aucune

SetRightMargin

Fixe la marge droite. La méthode peut être appelée avant de créer la première page.

Description

`SetRightMargin(float $margin)`

Liste des paramètres

`margin` : La marge.

Valeurs de retour

Aucune

SetSubject

Définit le sujet du document.

Description

`SetSubject(string $subject [, boolean $isUTF8])`

Liste des paramètres

subject : L'intitulé du sujet.

isUTF8 : Indique si la chaîne est encodée en ISO-8859-1 (false) ou en UTF-8 (true).

Valeur par défaut : false.

Valeurs de retour

Aucune

SetTextColor

Fixe la couleur pour le texte. Elle peut être indiquée en composantes RGB ou en niveau de gris. La méthode peut être appelée avant que la première page ne soit créée et la valeur est conservée de page en page.

Description

`SetTextColor(int $r [, int $g, int $b])`

Liste des paramètres

r : Si g et b sont renseignés, composante de rouge; sinon, indique le niveau de gris. Valeur comprise entre 0 et 255.

g : Composante de vert (entre 0 et 255).

b : Composante de bleu (entre 0 et 255).

Valeurs de retour

Aucune

SetTitle

Définit le titre du document.

Description

`SetTitle(string $title [, boolean $isUTF8])`

Liste des paramètres

title : Le titre.

isUTF8 : Indique si la chaîne est encodée en ISO-8859-1 (false) ou en UTF-8 (true).

Valeur par défaut : false.

Valeurs de retour

Aucune

SetTopMargin

Fixe la marge haute. La méthode peut être appelée avant de créer la première page.

Description

`SetTopMargin(float $margin)`

Liste des paramètres

`margin` : La marge.

Valeurs de retour

Aucune

SetX

Fixe l'abscisse de la position courante. Si la valeur transmise est négative, elle est relative à l'extrémité droite de la page.

Description

`SetX(float $x)`

Liste des paramètres

`x` : La valeur de l'abscisse.

Valeurs de retour

Aucune

SetXY

Fixe l'abscisse et l'ordonnée de la position courante. Si les valeurs transmises sont négatives, elles sont relatives respectivement aux extrémités droite et basse de la page.

Description

`SetXY(float $x, float $y)`

Liste des paramètres

`x` : La valeur de l'abscisse.

`y` : La valeur de l'ordonnée.

Valeurs de retour

Aucune

SetY

Fixe l'ordonnée et ramène éventuellement l'abscisse courante à la marge gauche. Si la valeur est négative, elle est relative au bas de la page.

Description

`SetY(float $y [, boolean $resetX])`

Liste des paramètres

y : La valeur de l'ordonnée.

resetx : Indique si l'abscisse doit être réinitialisée. Valeur par défaut : **true**.

Valeurs de retour

Aucune

Text

Imprime une chaîne de caractères. L'origine est à gauche du premier caractère, sur la ligne de base. Cette méthode permet de positionner précisément une chaîne dans la page, mais il est généralement plus simple d'utiliser **Cell**, **MultiCell** ou **Write** qui sont les méthodes standard pour imprimer du texte.

Description

Text(**float** \$x, **float** \$y, **string** \$txt)

Liste des paramètres

x : Abscisse de l'origine.

y : Ordonnée de l'origine.

txt : Chaîne à imprimer.

Valeurs de retour

Aucune

Write

Cette méthode imprime du texte à partir de la position courante. Lorsque la marge droite est atteinte (ou que le caractère \n est rencontré), un saut de ligne est effectué et le texte continue à partir de la marge gauche. Au retour de la méthode, la position courante est située juste à la fin du texte. Il est possible de mettre un lien sur le texte.

Description

Write(**float** \$h, **string** \$txt [, **mixed** \$link])

Liste des paramètres

h : Hauteur de la ligne.

txt : Chaîne à imprimer.

link : Url ou identifiant retourné par **Addlink**.

Valeurs de retour

Aucune

Exemple

```
// Début en police normale
$pdf->SetFont('Arial','',14);
$pdf->Write(5,'Visitez ');
// Lien en bleu souligné
$pdf->SetTextColor(0,0,255);
$pdf->SetFont('','U');
$pdf->Write(5,'www.fpdf.org','http://www.fpdf.org');
```