[Tutorial 1](http://www.fpdf.org/en/tutorial/tuto1.htm): Minimal example

Site para mais tutoriais: http://www.fpdf.org/en/tutorial/index.php

Vamos começar com o exemplo clássico:

**<?php**

**require('fpdf.php');**

**$pdf = new FPDF();**

**$pdf->AddPage();**

**$pdf->SetFont('Arial','B',16);**

**$pdf->Cell(40,10,'Hello World!');**

**$pdf->Output();**

**?>**

Depois de incluir o arquivo de biblioteca, criamos um objeto FPDF. O construtor FPDF () é usado aqui com os valores padrão: as páginas estão em retrato A4 e a unidade de medida é o milímetro. Poderia ter sido especificado explicitamente com:

**$pdf = new FPDF('P','mm','A4');**

É possível a utilização de paisagem (L), outros tamanhos de página (como Carta e Ofício) e as unidades (pt, cm, IN).

Não há nenhuma página no momento, por isso temos de adicionar uma com AddPage (). A origem está no canto superior esquerdo e a posição atual é de estabelecida em 1 cm das bordas padrão; as margens podem ser alteradas com SetMargins ().  
  
Antes de podermos imprimir um texto, é obrigatória para selecionar uma fonte com SetFont (), caso contrário, o documento seria inválido. Nós escolhemos Arial negrito 16:

**$pdf->SetFont('Arial','B',16);**

Poderíamos ter especificado itálico com I, sublinhada com U ou uma fonte regular com uma seqüência vazia (ou qualquer combinação). Note-se que o tamanho da fonte é dado em pontos, não milímetros (ou outra unidade do usuário); é a única exceção. As outras fontes padrão são Times, Courier, Symbol e ZapfDingbats.  
  
Agora podemos imprimir um celular com celular (). Uma célula é uma área rectangular, possivelmente moldado, que contém uma linha de texto. É de saída na posição atual. Nós especificar as suas dimensões, o seu texto (centralizado ou alinhado), se as fronteiras deve ser elaborado, e onde a posição atual se move depois que ele (à direita, abaixo ou para o início da próxima linha). Para adicionar uma moldura, faríamos isso:

**$pdf->Cell(40,10,'Hello World !',1);**

Para adicionar uma nova célula ao lado dele com o texto centrado e ir para a próxima linha, faríamos:

**$pdf->Cell(60,10,'Powered by FPDF.',0,1,'C');**

Observação: a quebra de linha também pode ser feito com Ln (). Este método additionnaly permite especificar a altura do intervalo.  
  
Finalmente, o documento é fechado e enviado para o navegador com saída (). Poderíamos tê-la guardado em um arquivo, passando o nome do arquivo desejado.  
  
Atenção: em caso quando o PDF é enviado para o browser, nada mais deve ser emitido pelo roteiro, nem antes nem depois (sem HTML, nem mesmo um espaço ou um retorno de carro). Se você enviar algo antes, você receberá a mensagem de erro: "Alguns dados já tem sido de saída, não é possível enviar arquivos PDF". Se você enviar algo depois, o documento poderá não ser exibida.

[Tutorial 2](http://www.fpdf.org/en/tutorial/tuto2.htm): Header, footer, page break and image

Aqui está um exemplo de duas páginas com cabeçalho, rodapé e logotipo:

**<?php**

**require('fpdf.php');**

**class PDF extends FPDF**

**{**

**// Page header**

**function Header()**

**{**

**// Logo**

**$this->Image('logo.png',10,6,30);**

**// Arial bold 15**

**$this->SetFont('Arial','B',15);**

**// Move to the right**

**$this->Cell(80);**

**// Title**

**$this->Cell(30,10,'Title',1,0,'C');**

**// Line break**

**$this->Ln(20);**

**}**

**// Page footer**

**function Footer()**

**{**

**// Position at 1.5 cm from bottom**

**$this->SetY(-15);**

**// Arial italic 8**

**$this->SetFont('Arial','I',8);**

**// Page number**

**$this->Cell(0,10,'Page '.$this->PageNo().'/{nb}',0,0,'C');**

**}**

**}**

**// Instanciation of inherited class**

**$pdf = new PDF();**

**$pdf->AliasNbPages();**

**$pdf->AddPage();**

**$pdf->SetFont('Times','',12);**

**for($i=1;$i<=40;$i++)**

**$pdf->Cell(0,10,'Printing line number '.$i,0,1);**

**$pdf->Output();**

**?>**

This example makes use of the Header() and Footer() methods to process page headers and footers. They are called automatically. They already exist in the FPDF class but do nothing, therefore we have to extend the class and override them.

The logo is printed with the Image() method by specifying its upper-left corner and its width. The height is calculated automatically to respect the image proportions.

**Header ()**

Este método é usado para processar o cabeçalho da página. Ele é automaticamente chamado por AddPage () e não deve ser chamado diretamente pelo aplicativo. A implementação em FPDF está vazia, então você tem que subclasse-lo e substituir o método se você quiser um tratamento específico.  
exemplo:

**class PDF extends FPDF**

**{**

**function Header()**

**{**

**// Select Arial bold 15**

**$this->SetFont('Arial','B',15);**

**// Move to the right**

**$this->Cell(80);**

**// Framed title**

**$this->Cell(30,10,'Title',1,0,'C');**

**// Line break**

**$this->Ln(20);**

**}**

**}**

**Footer()**  
  
Este método é usado para processar rodapé da página. Ele é automaticamente chamado por AddPage () e Close () e não deve ser chamado diretamente pelo aplicativo. A implementação em FPDF está vazia, então você tem que subclasse-lo e substituir o método se você quiser um tratamento específico.  
exemplo:

**class PDF extends FPDF**

**{**

**function Footer()**

**{**

**// Go to 1.5 cm from bottom**

**$this->SetY(-15);**

**// Select Arial italic 8**

**$this->SetFont('Arial','I',8);**

**// Print centered page number**

**$this->Cell(0,10,'Page '.$this->PageNo(),0,0,'C');**

**}**

**}**

**Image()**

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

Coloca uma imagem. O tamanho vai demorar na página podem ser especificadas de diferentes maneiras:  
  
     largura explícita e altura (expressa na unidade do usuário ou dpi)  
     uma dimensão explícita, o outro que está sendo calculado automaticamente a fim de manter as proporções originais  
     nenhuma dimensão explícita, caso em que a imagem é colocada em 96 dpi  
  
Os formatos suportados são JPEG, PNG e GIF. A extensão GD é necessário para GIF.  
  
Para JPEGs, todos os sabores são permitidos:  
  
     escalas de cinza  
     verdadeiras cores (24 bits)  
     CMYK (32 bits)  
  
Para PNGs, são permitidos:  
  
     cinza escalas em, no máximo, 8 bits (256 níveis)  
     cores indexadas  
     verdadeiras cores (24 bits)  
  
Para GIFs: no caso de um GIF animado, somente o primeiro quadro é exibido.  
  
Transparência é suportado.  
  
O formato pode ser especificado explicitamente ou inferida a partir da extensão do arquivo.  
  
É possível colocar um link na imagem.  
  
Observação: se uma imagem é usada várias vezes, apenas uma cópia está embutido no arquivo.

**parâmetros**  
arquivo  
     Caminho ou URL da imagem.  
X  
     Abscissa do canto superior esquerdo. Se não for especificado ou igual a null, a abscissa atual será usado.  
y  
     Ordenada do canto superior esquerdo. Se não for especificado ou igual a null, a ordenada atual é usado; além disso, uma quebra de página é accionado pela primeira vez, se necessário (no caso da quebra de página automática está activada) e, após a chamada, a ordenada corrente é movida para a parte inferior da imagem.  
w  
     Largura da imagem na página. Há três casos:  
  
         Se o valor for positivo, ele representa a largura na unidade do usuário  
         Se o valor for negativo, o valor absoluto representa a resolução horizontal em dpi  
         Se o valor não for especificado ou igual a zero, ele é automaticamente calculada  
  
h  
     Altura da imagem na página. Há três casos:  
  
         Se o valor for positivo, ele representa a altura na unidade do usuário  
         Se o valor for negativa, o valor absoluto representa a resolução vertical em dpi  
         Se o valor não for especificado ou igual a zero, ele é automaticamente calculada  
  
tipo  
     Formato de imagem. Os valores possíveis são (case insensitive): JPG, JPEG, PNG e GIF. Se não for especificado, o tipo é inferida a partir da extensão do arquivo.  
link  
     URL ou identificador retornado por AddLink ().

**Example**

// Insert a logo in the top-left corner at 300 dpi

$pdf->Image('logo.png',10,10,-300);

// Insert a dynamic image from a URL

$pdf->Image('http://chart.googleapis.com/chart?cht=p3&chd=t:60,40&chs=250x100&chl=Hello|World',60,30,90,0,'PNG');

**AddLink**

**int AddLink ()**  
  
Cria um novo link interno e retorna seu identificador. Uma ligação interna é uma área clicável que direciona para outro lugar dentro do documento.  
O identificador pode, então, ser passado para celular (), write (), Imagem () ou Link (). O destino é definido com SetLink ().

Para imprimir o número da página, um valor nulo é passado como a largura da célula. Isso significa que a célula deve se estender até a margem direita da página; Isso é útil para centralizar o texto. O número da página atual é devolvido pelo método PáginaNenhuma (); como para o número total de páginas, que é obtido através do valor especial nb} {o qual é substituído quando o documento é terminado (contanto que primeiramente chamados AliasNbPages ()).  
Note-se a utilização do método setY () que permite ajustar a posição absoluta de uma localização na página, a partir da parte superior ou na parte inferior.

Outra característica interessante é usado aqui: Página da quebra automática. Assim que uma célula iria atravessar um limite na página (a dois centímetros da parte inferior, por padrão), uma pausa é emitido e a fonte restaurada. Embora o cabeçalho e rodapé selecionar sua própria fonte (Arial), o corpo continua com Times. Este mecanismo de restabelecimento automático também se aplica às cores e largura da linha. O limite que provoca quebras de página pode ser definida com SetAutoPageBreak ().