# Multi-columns

This example is a variant of the previous one showing how to lay the text across multiple columns.

<?php  
require('fpdf.php');  
  
class PDF extends FPDF  
{  
protected $col = 0; // Current column  
protected $y0; // Ordinate of column start  
  
function Header()  
{  
 // Page header  
 global $title;  
  
 $this->SetFont('Arial','B',15);  
 $w = $this->GetStringWidth($title)+6;  
 $this->SetX((210-$w)/2);  
 $this->SetDrawColor(0,80,180);  
 $this->SetFillColor(230,230,0);  
 $this->SetTextColor(220,50,50);  
 $this->SetLineWidth(1);  
 $this->Cell($w,9,$title,1,1,'C',true);  
 $this->Ln(10);  
 // Save ordinate  
 $this->y0 = $this->GetY();  
}  
  
function Footer()  
{  
 // Page footer  
 $this->SetY(-15);  
 $this->SetFont('Arial','I',8);  
 $this->SetTextColor(128);  
 $this->Cell(0,10,'Page '.$this->PageNo(),0,0,'C');  
}  
  
function SetCol($col)  
{  
 // Set position at a given column  
 $this->col = $col;  
 $x = 10+$col\*65;  
 $this->SetLeftMargin($x);  
 $this->SetX($x);  
}  
  
function AcceptPageBreak()  
{  
 // Method accepting or not automatic page break  
 if($this->col<2)  
 {  
 // Go to next column  
 $this->SetCol($this->col+1);  
 // Set ordinate to top  
 $this->SetY($this->y0);  
 // Keep on page  
 return false;  
 }  
 else  
 {  
 // Go back to first column  
 $this->SetCol(0);  
 // Page break  
 return true;  
 }  
}  
  
function ChapterTitle($num, $label)  
{  
 // Title  
 $this->SetFont('Arial','',12);  
 $this->SetFillColor(200,220,255);  
 $this->Cell(0,6,"Chapter $num : $label",0,1,'L',true);  
 $this->Ln(4);  
 // Save ordinate  
 $this->y0 = $this->GetY();  
}  
  
function ChapterBody($file)  
{  
 // Read text file  
 $txt = file\_get\_contents($file);  
 // Font  
 $this->SetFont('Times','',12);  
 // Output text in a 6 cm width column  
 $this->MultiCell(60,5,$txt);  
 $this->Ln();  
 // Mention  
 $this->SetFont('','I');  
 $this->Cell(0,5,'(end of excerpt)');  
 // Go back to first column  
 $this->SetCol(0);  
}  
  
function PrintChapter($num, $title, $file)  
{  
 // Add chapter  
 $this->AddPage();  
 $this->ChapterTitle($num,$title);  
 $this->ChapterBody($file);  
}  
}  
  
$pdf = new PDF();  
$title = '20000 Leagues Under the Seas';  
$pdf->SetTitle($title);  
$pdf->SetAuthor('Jules Verne');  
$pdf->PrintChapter(1,'A RUNAWAY REEF','20k\_c1.txt');  
$pdf->PrintChapter(2,'THE PROS AND CONS','20k\_c2.txt');  
$pdf->Output();  
?>

[[Run]](http://docs.google.com/tuto4.php)

The key method used is [AcceptPageBreak()](http://docs.google.com/doc/acceptpagebreak.htm). It allows to accept or not an automatic page break. By refusing it and altering the margin and current position, the desired column layout is achieved.

For the rest, not many changes; two properties have been added to the class to save the current column number and the position where columns begin, and the MultiCell() call specifies a 6 centimeter width.