# Tables

This tutorial shows different ways to make tables.

<?php  
require('fpdf.php');  
  
class PDF extends FPDF  
{  
// Load data  
function LoadData($file)  
{  
 // Read file lines  
 $lines = file($file);  
 $data = array();  
 foreach($lines as $line)  
 $data[] = explode(';',trim($line));  
 return $data;  
}  
  
// Simple table  
function BasicTable($header, $data)  
{  
 // Header  
 foreach($header as $col)  
 $this->Cell(40,7,$col,1);  
 $this->Ln();  
 // Data  
 foreach($data as $row)  
 {  
 foreach($row as $col)  
 $this->Cell(40,6,$col,1);  
 $this->Ln();  
 }  
}  
  
// Better table  
function ImprovedTable($header, $data)  
{  
 // Column widths  
 $w = array(40, 35, 40, 45);  
 // Header  
 for($i=0;$i<count($header);$i++)  
 $this->Cell($w[$i],7,$header[$i],1,0,'C');  
 $this->Ln();  
 // Data  
 foreach($data as $row)  
 {  
 $this->Cell($w[0],6,$row[0],'LR');  
 $this->Cell($w[1],6,$row[1],'LR');  
 $this->Cell($w[2],6,number\_format($row[2]),'LR',0,'R');  
 $this->Cell($w[3],6,number\_format($row[3]),'LR',0,'R');  
 $this->Ln();  
 }  
 // Closing line  
 $this->Cell(array\_sum($w),0,'','T');  
}  
  
// Colored table  
function FancyTable($header, $data)  
{  
 // Colors, line width and bold font  
 $this->SetFillColor(255,0,0);  
 $this->SetTextColor(255);  
 $this->SetDrawColor(128,0,0);  
 $this->SetLineWidth(.3);  
 $this->SetFont('','B');  
 // Header  
 $w = array(40, 35, 40, 45);  
 for($i=0;$i<count($header);$i++)  
 $this->Cell($w[$i],7,$header[$i],1,0,'C',true);  
 $this->Ln();  
 // Color and font restoration  
 $this->SetFillColor(224,235,255);  
 $this->SetTextColor(0);  
 $this->SetFont('');  
 // Data  
 $fill = false;  
 foreach($data as $row)  
 {  
 $this->Cell($w[0],6,$row[0],'LR',0,'L',$fill);  
 $this->Cell($w[1],6,$row[1],'LR',0,'L',$fill);  
 $this->Cell($w[2],6,number\_format($row[2]),'LR',0,'R',$fill);  
 $this->Cell($w[3],6,number\_format($row[3]),'LR',0,'R',$fill);  
 $this->Ln();  
 $fill = !$fill;  
 }  
 // Closing line  
 $this->Cell(array\_sum($w),0,'','T');  
}  
}  
  
$pdf = new PDF();  
// Column headings  
$header = array('Country', 'Capital', 'Area (sq km)', 'Pop. (thousands)');  
// Data loading  
$data = $pdf->LoadData('countries.txt');  
$pdf->SetFont('Arial','',14);  
$pdf->AddPage();  
$pdf->BasicTable($header,$data);  
$pdf->AddPage();  
$pdf->ImprovedTable($header,$data);  
$pdf->AddPage();  
$pdf->FancyTable($header,$data);  
$pdf->Output();  
?>

[[Run]](http://docs.google.com/tuto5.php)

A table being just a collection of cells, it's natural to build one from them. The first example is achieved in the most basic way possible: simple framed cells, all of the same size and left aligned. The result is rudimentary but very quick to obtain.

The second table brings some improvements: each column has its own width, headings are centered, and numbers right aligned. Moreover, horizontal lines have been removed. This is done by means of the border parameter of the [Cell()](http://docs.google.com/doc/cell.htm) method, which specifies which sides of the cell must be drawn. Here we want the left (L) and right (R) ones. It remains the problem of the horizontal line to finish the table. There are two possibilities: either check for the last line in the loop, in which case we use LRB for the border parameter; or, as done here, add the line once the loop is over.

The third table is similar to the second one but uses colors. Fill, text and line colors are simply specified. Alternate coloring for rows is obtained by using alternatively transparent and filled cells.