

Experiment No: 07

Write a simple PHP program to create PDF document by using graphics concepts.

Resources required:

Hardware	Software
Computer System	Any database tools such as XAMPP

Practical Significance:














FPDF is a PHP class which allows to generate PDF files with PHP code. F from FPDF stands for Free: It is free to use and it does not require any API keys. you may use it for any kind of usage and modify it to user needs.

Theoretical Background:

- Advantages of FPDF :
 - Choice of measure unit, page format and margins
 - Allow to set Page header and footer management
 - It provides automatic line break, Page break and text justification
 - It supports Images in various formats (JPEG, PNG and GIF)
 - It allows to setup Colors, Links, TrueType, Type1 and encoding support
 - It allows Page compression
- Procedure to create a PDF in PHP:
 - Link to download latest version of FPDF class:
<http://www.fpdf.org/en/download.php>

Downloads

You can view the changelog [here](#).

-  An Arabic version of the 1.53 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Abdullah E.AL-Mehmadi](#))
-  A Catalan version of the 1.81 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [David Gimeno i Ayuso](#))
-  A Hungarian version of the 1.81 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Csaba Varga](#))
-  An Indonesian version of the 1.6 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Kang Oedin](#))
-  A Japanese version of the 1.52 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Matt Ataka](#))
-  A Korean version of the 1.6 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Kim Taegon](#))
-  A Dutch version of the 1.6 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Martijn Hogerheijde](#))
-  A Polish version of the 1.81 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Rafał Lysik](#))
-  A Brazilian Portuguese version of the 1.81 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Domingos Paraiso](#))
-  A Romanian version of the 1.52 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Cantemir Mihu](#))
-  A Russian version of the 1.53 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Olga Smila](#))
-  A Turkish version of the 1.7 manual is available in [ZIP](#) and [TGZ](#) formats (translation by [Caglar Ulkuderner](#))
-  A Chinese version of the 1.52 manual is available in simplified ([ZIP](#), [TGZ](#)) and traditional language ([ZIP](#), [TGZ](#)) (translation by [Andrew Choi](#))

v1.82 (2019-12-07)
[ZIP](#) | [TGZ](#)

- Download v1.82 and extract and place the folder names as “fpdf182” at C:\xampp\htdocs\test.

- Open fpdf182 folder and copy all sub-folders and files at C:\xampp\htdocs\test.
- Write the following script on notepad, save and execute.

Program Code:

```
<?php
require('fpdf.php');
$pdf=new FPDF();
$pdf->AddPage();
$pdf->SetFont('Arial','B',16);
$pdf->Cell(60,10,'Hello PHP World!',1,1,'C');
$pdf->Output();
?>
```

Cell ():

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.

Syntax:

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

Parameters:

Parameter name	Description
W	Cell width. If 0, the cell extends up to the right margin.
H	Cell height. Default value: 0
Txt	String to print. Default value: empty string.
border	Indicates if borders must be drawn around the cell. The value can be either a number: <ul style="list-style-type: none"> • 0: no border 1: frame <p>Default value: 0</p> <p>or</p> <p>a string containing some or all of the following characters (in any order):</p>

	<p>L: left T: top</p> <p>R: right B: bottom</p>
<u>ln</u>	<p>Indicates where the current position should go after the call. Possible values are:</p> <ul style="list-style-type: none"> • 0: to the right • 1: to the beginning of the next line • 2: below <p>Putting 1 is equivalent to putting 0 and calling Ln() just after. Default value: 0.</p>
<u>align</u>	<p>Allows to center or align the text. Possible values are:</p> <ul style="list-style-type: none"> • L or empty string: left align (default value) • C: center • R: right align
<u>fill</u>	<p>Indicates if the cell background must be painted (true) or transparent (false). Default value: false.</p>

Practical related questions:

1. What's the difference between include and require?

The include() statement will only generate a PHP warning but allow script execution to continue if the file to be included can't be found, whereas the require() statement will generate a fatal error and stops the script execution.

2. State the use of Cell().

Prints a cell (rectangular area) with optional borders, background color and character string.

3. State the use of AddPage();

Adds a new page to the document.

4. State the use of setFont().

Sets the font used to print character strings.

Exercise:

1. Write a script for setup Header and Footer along with line break and generate a pdf.

```
<?php

require('fpdf183/fpdf.php');

class PDF extends FPDF
{
    function Header()
    {
        $this->SetFont('Times','B',20);
        $this->Cell(80);
        $this->Cell(30, 20, 'HEADER', 0, 2, 'C');
        $this->Ln(5);
    }

    function Footer()
    {
        $this->SetY(-15);
        $this->SetFont('Arial', 'I', 8);
        $this->Cell(0, 10, 'FOOTER', 0, 0, 'C');
    }
}

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

$pdf->AddPage();

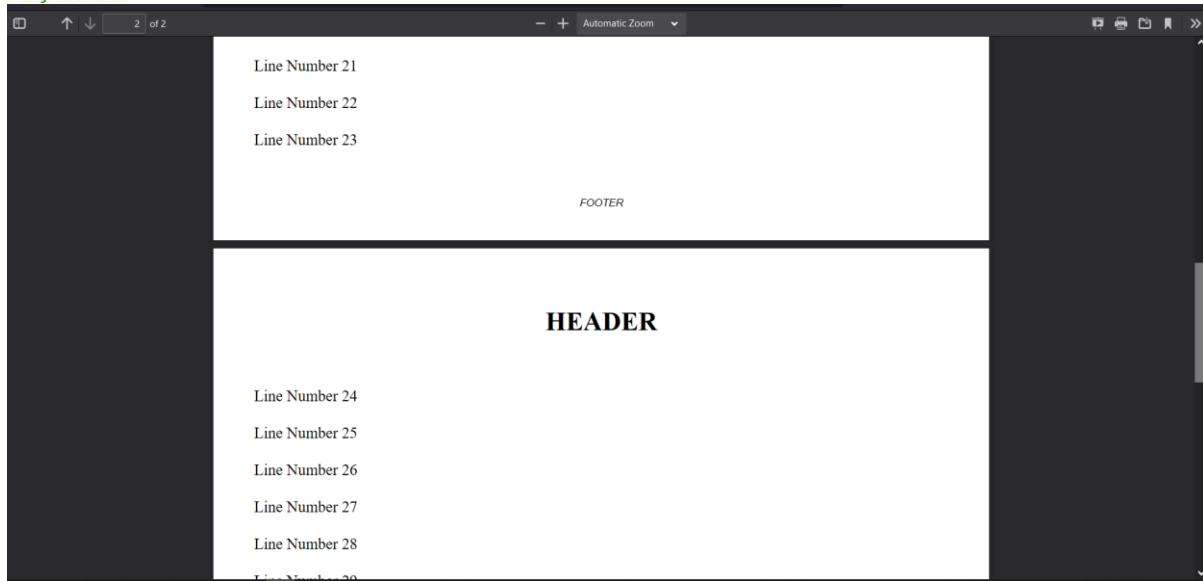
$pdf->SetFont('Times','',12);

for($i = 0; $i <= 30; $i++)
    $pdf->Cell(30, 10, 'Line Number ' . $i, 0, 2, 'L');

$pdf->Output();

?>
```

Output:



2. Write a code based on setFillColor() and setTextColor() and generate a pdf.

```
<?php

require('fpdf183/fpdf.php');

class PDF extends FPDF
{
    function AddString($string)
    {
        $this->AddPage();
        $this->SetFont('Times','',12);
        $this->setTextColor(239, 43, 43);
        $this->SetFillColor(64, 232, 46);
        $this->Cell(0,6,"$string",0,1,'C',true);
    }
}

$pdf = new PDF();
$pdf -> AddString("Hello World");
$pdf -> Output();

?>
```

Output:

