

PDF? Pakai PHP saja!

Ivan Irawan

ivan@smg.linux.or.id

Lisensi Dokumen:

Copyright © 2003-2006 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Saya yakin jika Anda cukup bergaul dan *melek* internet, pasti pernah mendengar, membaca, juga barangkali membuat dokumen dalam bentuk format **PDF** (Portable Document Format). Ketika Anda membaca artikel ini secara *online*, saya semakin yakin bahwa tidak mungkin Anda tidak pernah mendengar mengenai PDF. Paling tidak Anda telah mendengar mengenai PDF judul dari judul artikel ini :)

Bisa beli PDF-nya, Pak?

Format dokumen PDF pertama kali dikenalkan oleh Adobe System (<http://www.adobe.com>), hasil dari pengembangan Dr. John Warnock di awal tahun 90-an. Proyek PDF dimaksudkan membuat format file untuk distribusi dokumen di dalam perusahaan yang dapat ditampilkan di berbagai platform komputasi.

PDF akhirnya menjadi format dokumen yang universal dan lintas platform, karena tersedia di berbagai platform komputasi. Program untuk membaca dokumen PDF dalam berbagai platform disediakan secara gratis untuk didownload oleh Adobe System. *Plug-in* program pembaca untuk *browser* populer juga tersedia, sehingga dokumen PDF dapat dibaca langsung secara *online* dari situs internet tanpa perlu mendownload dokumen dan membacanya dengan program yang terpisah dari *browser* web.

PDF memiliki beberapa fitur keunggulan antara lain:

- **Universal dan Lintas Platform.**

Dokumen PDF dapat dibaca di berbagai platform, bahkan dapat dibaca secara *online*

1

melalui *browser* yang telah dilengkapi dengan *plug-in* pembaca PDF. *Tools* dan *printer driver* untuk membuat dokumen PDF pun banyak tersedia. Keunggulan ini menyebabkan dokumen PDF dapat digunakan secara luas di seluruh dunia.

- **Yang Anda Lihat, Yang Anda Dapatkan.**
Berbeda dengan format HTML/SGML, PDF lebih dapat menjamin apa yang ditampilkan pada monitor, maka serupa itu pula yang akan kita dapatkan jika dokumen dicetak. Dengan format HTML/SGML, maka hasil cetakan bisa jadi akan berbeda dengan yang tampak di monitor, bahkan dokumen yang tampak di monitor pun bisa jadi berbeda-beda tergantung dari *browser* yang digunakan. Hasilnya, PDF sangat sesuai digunakan untuk menampilkan informasi kritis yang tidak memperbolehkan variasi dalam presentasinya, misalnya invoice, tanda terima, dan laporan-laporan baku lainnya.
- **Hemat Ukurannya.**
Secara *native* file PDF telah dikompresi sedemikian rupa, sehingga ukuran filenya akan lebih kecil daripada jika file tersebut dalam format HTML, misalnya. Kecilnya ukuran file PDF ini menyebabkan PDF populer digunakan untuk tukar-menukar dokumen lewat internet karena tidak memboroskan *bandwidth* data.
- **Pengamanan Dokumen Dapat Diandalkan.**
PDF juga memiliki fitur pengamanan dokumen dari kemungkinan pencurian isi, duplikasi, dan pencetakan dokumen oleh yang tidak berhak. Sebuah dokumen PDF dapat pula diberikan tanda tangan digital (*digital signature*) sehingga penyebaran dokumen dapat lebih dikendalikan. Fitur ini tidak dimiliki oleh format dokumen universal seperti HTML/SGML.

1.1. Benarkah Perlu PDF?

Kecuali jika Anda hanya ingin bekerja dengan dokumen-dokumen non presisi dan tidak menginginkan kemudahan bertukar dokumen, maka Anda kemungkinan besar membutuhkan format dokumen PDF.

Jika Anda ingin:

- membuat invoice, order pembelian, surat pengangkutan, tanda terima, dan dokumen komersial untuk situs e-commerce atau aplikasi berbasis web
- membuat laporan dari aplikasi berbasis web dengan format yang presisi yang tidak memperkenankan variasi hasil cetakan
- membuat cetakan form isian dari web dengan hasil yang seragam
- membuat dokumen lain yang membutuhkan pengendalian baik terhadap distribusi dokumen maupun presisi dan kualitas cetakan, serta dapat dipertukarkan secara mudah,

Saya berani memberi saran kepada Anda untuk menghindari kerja keras memformat dokumen dalam bentuk HTML dengan menggunakan format dokumen PDF.

Sebagai pecinta PHP, Anda benar-benar dimanjakan karena PHP memiliki kemampuan

untuk membuat dokumen PDF. Pada artikel ini kita akan coba mempelajarinya.

1.2. Pilih Yang Mana...

Cukup banyak ekstensi PHP tersedia yang memungkinkan Anda membuat dokumen PDF melalui PHP. Beberapa yang dapat Anda pilih:

- **PDFLib**. Aladdin Free Public License memperkenankan penggunaan PDFLib untuk penggunaan non komersial. Untuk lisensi komersial, termasuk penggunaan dalam layanan web komersial, dikenakan harga lisensi yang dapat dilihat pada <http://pdflib.com>.
- **FreePDFlib** oleh Thomas Szadel (*open source*).
- **ClibPDF** menggunakan model lisensi yang sama dengan PDFLib.

Dalam tulisan ini, kita akan menggunakan fungsi-fungsi yang ada pada PDFLib. Sebelum memulai semuanya, sebaiknya Anda pastikan dulu ekstensi PDFLib telah terpasang pada sistem PHP Anda. Jangan lupa juga, Anda juga harus memastikan *browser* Anda telah terinstalasi *plug-in* untuk membaca dokumen PDF.

Untuk lebih memudahkan Anda, maka telah tersedia *source code* dan hasilnya (dalam dokumen PDF) dalam bentuk file *.zip*. Anda tinggal download saja file [artpdf.zip](#) ini.

Jika Anda telah siap, mari kita lanjutkan dengan membuat dokumen PDF paling sederhana.

1.3. PDF Pertamaku, Tak Terlupakan...

Sesungguhnya saya tidak menyukai memulai sesuatu dengan rutinitas. Ketika Anda belajar bahasa pemrograman apa pun, selalu kita dipaksa untuk mengucapkan, "Hello, World!" dalam bahasa pemrograman baru. Saya juga akan mengajak Anda untuk melakukannya lagi saat ini, namun untuk sedikit mengurangi kejenuhan, mari kita ubah program ini menjadi program "PDF Pertamaku".

Coba Anda buat skrip seperti di bawah ini, dan coba jalankan di PHP melalui *browser* favorit Anda.

```
<?php
    $halaman = pdf_new();
    pdf_open_file($halaman);
    pdf_set_info($halaman,"Creator","pdf-ku.php");
    pdf_set_info($halaman,"Author","Mr. Dodol");
    pdf_set_info($halaman,"Title","PDF Pertamaku");
    pdf_begin_page($halaman,595,842);
```

3

```
$huruf = pdf_findfont($halaman,"Helvetica-Bold","host",0);
pdf_setfont($halaman,$huruf,38.0);
pdf_show_xy($halaman,"Inilah PDF Pertamaku!",50,700);
pdf_end_page($halaman);
pdf_close($halaman);
$buf = pdf_get_buffer($halaman);
$panjangbuffer = strlen($buf);
Header("Content-type: application/pdf");
Header("Content-Length: $panjangbuffer");
Header("Content-Disposition: inline; filename=pdf-ku.pdf");
echo $buf;
pdf_delete($halaman);

?>
```

1.4. Tentukan Letak Bintang di Langit

Membuat dokumen PDF, sebenarnya mirip dengan proses menggambar pada kanvas. Agar Anda mampu menempatkan gambar maupun tulisan di tempat yang sesuai dengan keinginan Anda, maka Anda perlu mengetahui cara penentuan letak pada dokumen PDF.

Pada dokumen PHP berlaku sistem koordinat dua dimensi (x,y), dimana titik asal atau koordinat (0,0) ada di pojok kiri bawah dokumen. Arah koordinat x adalah arah horisontal (dari kiri ke kanan) dan arah koordinat y adalah arah vertikal (dari bawah ke atas). Sistem ukuran dasar/skala yang digunakan adalah **point** atau disingkat **pt**, dengan konversinya:

1 pt = 1/72 inch = 0.35277777778 mm, atau dengan kata lain
 1 inch = 72 pt.

Pembuatan dokumen PHP, dikerjakan per halaman. Pada saat awal membuat halaman, Anda harus menentukan terlebih dahulu lebar dan panjang kertas yang akan digunakan dalam satuan pt. Tabel berikut ini akan memberikan informasi ukuran jenis kertas dalam satuan pt.

Jenis Ukuran Kertas	Horisontal	Vertikal
US Letter	612	792
US Legal	612	1008
US Ledger	1224	792
11x17	792	1224
A0	2380	3368
A1	1684	2380
A2	1190	1648

A3	842	1190
A4	595	842
A5	421	595
A6	297	421
B5	501	709

Mari Anda coba skrip di bawah ini untuk belajar bermain dalam koordinat halaman berukuran A4.

```
<?php
    $halaman = pdf_new();
    pdf_open_file($halaman);
    pdf_set_info($halaman,"Creator","koord.php");
    pdf_set_info($halaman,"Author","Mr. Dodol");
    pdf_set_info($halaman,"Title","Test Koordinat PDF");
    pdf_begin_page($halaman,595,842); //ukuran kertas A4

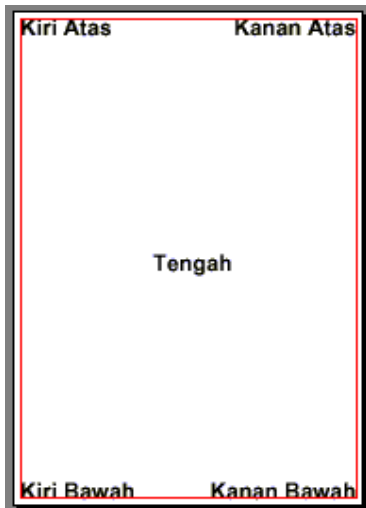
    //membuat tulisan pada halaman
    $huruf = pdf_findfont($halaman,"Helvetica-Bold","host",0);
    pdf_setfont($halaman,$huruf,38.0);
    pdf_show_xy($halaman, "Kiri Bawah", 10, 10);
    pdf_show_xy($halaman, "Kanan Bawah", 335, 10);
    pdf_show_xy($halaman, "Kiri Atas", 10, 802);
    pdf_show_xy($halaman, "Kanan Atas", 375, 802);
    pdf_show_xy($halaman, "Tengah", 595/2-60, 842/2-20);

    // membuat garis di pinggir halaman
    pdf_setrgbcolor_stroke($halaman,1,0,0);
    pdf_moveto($halaman,10,10);
    pdf_lineto($halaman,10,832);
    pdf_lineto($halaman,585,832);
    pdf_lineto($halaman,585,10);
    pdf_lineto($halaman,10,10);
    pdf_stroke($halaman);
    pdf_setrgbcolor_stroke($halaman,0,0,0);

    pdf_end_page($halaman);
    pdf_set_parameter($halaman, "openaction", "fitpage");
    pdf_close($halaman);
    $buf = pdf_get_buffer($halaman);
    $panjangbuffer = strlen($buf);
    Header("Content-type:application/pdf");
    Header("Content-Length:$panjangbuffer");
    Header("Content-Disposition:inline; filename=koord.pdf");
    echo $buf;
```

```
pdf_delete($halaman);  
?>
```

Skrip di atas akan menghasilkan dokumen PDF serupa ini.



Anda memiliki kemungkinan untuk mengubah titik asal (0,0) dan membalik arah koordinat, sebagai contoh, Anda dapat membuat pojok kiri atas sebagai titik asal koordinat dan memiliki nilai positif untuk arah dari atas ke bawah dan dari kiri ke kanan. Anda dapat mencoba skrip berikut ini.

```
<?php  
$halaman = pdf_new();  
pdf_open_file($halaman);  
pdf_set_info($halaman,"Creator","koordbalik.php");  
pdf_set_info($halaman,"Author","Mr. Dodol");  
pdf_set_info($halaman,"Title","Mengubah Titik Asal dan Arah  
Koordinat (PHP)");  
pdf_begin_page($halaman,595,842);  
  
// Mengubah Titik Asal  
pdf_translate($halaman,0,842);  
  
// Membalik Arah Sumbu Mendatar  
pdf_scale($halaman, 1, -1);  
  
// Mencerminkan skala horisontal  
pdf_set_value($halaman,"horizscaling",-100);  
  
$huruf = pdf_findfont($halaman,"Helvetica-Bold","host",0);
```

6

```
pdf_setfont($halaman,$huruf,-38.0);  
pdf_show_xy($halaman, "Top Left", 10, 40);  
  
pdf_end_page($halaman);  
pdf_set_parameter($halaman, "openaction", "fitpage");  
pdf_close($halaman);  
$buf = pdf_get_buffer($halaman);  
$panjangbuffer = strlen($buf);  
Header("Content-type:application/pdf");  
Header("Content-Length:$panjangbuffer");  
Header("Content-Disposition:inline;  
filename=koordbalik.pdf");  
echo $buf;  
pdf_delete($halaman);  
  
?>
```

1.5. Bicara dengan Tulisan

Dalam contoh-contoh skrip di atas, Anda telah mencoba menulis pada halaman PDF. Kali ini Anda akan mendapatkan penjelasan mengenai fungsi-fungsi dasar yang digunakan untuk membuat tulisan pada dokumen PDF. Sebagian dari fungsi-fungsi ini telah pernah Anda gunakan.

pdf_show_xy()

Fungsi ini digunakan untuk menuliskan *text* pada posisi tertentu yang diberikan. Contoh:

```
<? pdf_show_xy($halaman,"Tulisan Saya",50,100); ?>
```

Contoh di atas akan menuliskan text "Tulisan Saya" mulai pada koordinat (50,100) pada halaman PDF yang didefinisikan oleh variabel \$halaman.

pdf_show()

Fungsi ini digunakan untuk menuliskan *text* pada posisi tertentu yang telah diset terlebih dahulu dengan fungsi pdf_set_text_pos(). Contoh:

```
<? pdf_set_text_pos($halaman, 50,100);  
pdf_show($halaman,"Text"); ?>
```

pdf_continue_text()

Fungsi ini digunakan untuk menuliskan *text* pada posisi baris berikutnya.

pdf_show_boxed()

Fungsi ini digunakan untuk memformat *text* dalam suatu box/kotak tertentu. Contoh skrip berikut ini akan memberi kesempatan bagi Anda untuk mengerti maksud penggunaan fungsi `pdf_show_boxed()`.

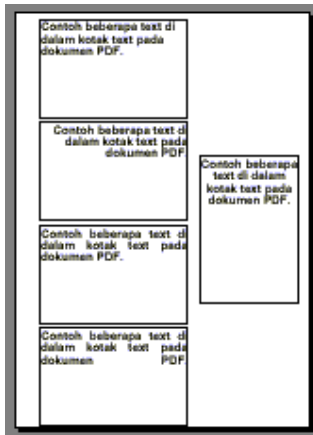
```
<?php
    $halaman = pdf_new();
    pdf_open_file($halaman);
    pdf_begin_page($halaman,595,842);
    $font = pdf_findfont($halaman,"Helvetica-Bold","host",0);
    pdf_setfont($halaman,$font,24.0);

    $text = <<<TEKS
    Contoh beberapa text di dalam kotak text pada dokumen PDF.
    TEKS;

    pdf_show_boxed($halaman, $text, 50, 630, 300, 200, "left");
    pdf_rect($halaman,50,630,300,200); pdf_stroke($halaman);
    pdf_show_boxed($halaman, $text, 50, 420, 300, 200, "right");
    pdf_rect($halaman,50,420,300,200); pdf_stroke($halaman);
    pdf_show_boxed($halaman, $text, 50, 210, 300, 200, "justify");
    pdf_rect($halaman,50,210,300,200);
    pdf_stroke($halaman);
    pdf_show_boxed($halaman, $text, 50, 0, 300, 200,
    "fulljustify");
    pdf_rect($halaman,50,0,300,200);
    pdf_stroke($halaman);
    pdf_show_boxed($halaman, $text, 375, 250, 200, 300, "center");
    pdf_rect($halaman,375,250,200,300);
    pdf_stroke($halaman);

    pdf_end_page($halaman);
    pdf_set_parameter($halaman, "openaction", "fitpage");
    pdf_close($halaman);
    $buf = pdf_get_buffer($halaman);
    $panjangbuffer = strlen($buf);
    Header("Content-type:application/pdf");
    Header("Content-Length:$panjangbuffer");
    Header("Content-Disposition:inline;
    filename=kotakteks.pdf");
    echo $buf;
    pdf_delete($halaman);
?>
```

Skrip di atas jika dijalankan, akan menghasilkan dokumen PDF seperti gambar berikut ini.



1.6. Kusuka Bentuknya

PDFLib yang Anda gunakan menyediakan 14 pilihan huruf *built-in*, yaitu:

- Courier
- Courier-Bold
- Courier-Oblique
- Courier-BoldOblique
- Helvetica
- Helvetica-Bold
- Helvetica-Oblique
- Helvetica-BoldOblique
- Times-Roman
- Times-Bold
- Times-Italic
- Times-BoldItalic
- Symbol
- ZapfDingbats.

Berikut beberapa contoh dari jenis huruf standar tersebut di atas.

Courier
Courier-Bold
Courier-Oblique
Courier-BoldOblique
Helvetica
Helvetica-Bold
Helvetica-Oblique
Helvetica-BoldOblique
Times-Roman
Times-Bold
Times-Italic
Times-BoldItalic

Anda juga dapat menggunakan jenis huruf AFM, Postscript Type-1, dan TTF. Mari kita coba lihat cara menggunakan jenis huruf TTF atau *True Type Font*.

```
<?php
    pdf_set_parameter($halaman,"FontOutline",
                      "Arial==/usr/fonts/arial.ttf");
    $font = pdf_findfont($halaman,"Arial","host",1);
?>
```

Jenis huruf (*font*) dapat pula didefinisikan pada file `pdflib.upr`. Contoh berikut ini menunjukkan caranya.

```
<?php
// Ubah nilai variabel ini sesuai dengan path file pdflib.upr
// di sistem PHP Anda.
$file_upr = "/usr/share/fonts/pdflib/pdflib.upr";

$halaman = pdf_new();
pdf_open_file($halaman);
pdf_set_info($halaman,"Creator","hurufttf.php");
pdf_set_info($halaman,"Author","Mr. Dodol");
pdf_set_info($halaman,"Title","Test Jenis Huruf (PHP)");
pdf_set_parameter($halaman,"resourcefile",$file_upr);
pdf_begin_page($halaman,595,842);
pdf_set_text_pos($halaman,25,800);
```

```
// Buat Array Nama Font dan jenisnya
$fonts = array('Arial'=>1,'Comic Sans MS'=>1,'Impact'=>1);

// Cetak Font ke Dokumen PDF
foreach($fonts as $f=>$embed) {
    $font = pdf_findfont($halaman,$f,"host",$embed);
    pdf_setfont($halaman,$font,25.0);
    pdf_continue_text($halaman,"$f (".chr(128)." Ç à á â ã
ç è é ê)");
}

pdf_end_page($halaman);
pdf_close($halaman);
$buf = pdf_get_buffer($halaman);
$len = strlen($buf);
Header("Content-type:application/pdf");
Header("Content-Length:$len");
Header("Content-Disposition:inline;
filename=huruf.ttf.pdf");
echo $buf;
pdf_delete($halaman);

?>
```

File `pdflib.upr` harus terisi dengan entri sebagai berikut.

```
FontOutline
Arial=arial.ttf
Comic Sans MS=comic.ttf
Impact=IMPACT.ttf
.
```

Jika muncul pesan kesalahan, kemungkinan Anda harus mengisi entri nama file pada `FontOutline` lengkap dengan path dari file `.ttf`. Jika berhasil, Anda akan memperoleh hasil sebagai berikut.

Arial (€ Ç à á â ã ç è é ê)
Comic Sans MS (€ Ç à á â ã ç è é ê)
Impact (€ Ç à á â ã ç è é ê)

Enkoding karakter yang tersedia pada PDFLib yaitu

- **winansi** (superset dari ISO 8859-1)
- **macroman** (Enkoding standar Macintosh)

- **ebcdic** (Digunakan pada IBM AS/400 dan S/390)
- **biltin** (Enkoding asli yang digunakan oleh huruf teks non latin)
- **host** (macroman di Mac, ebcdic di sistem EBCDIC, dan winansi di windows)

1.7. Gambar Bermakna Seribu Kata...

Anda telah mendapatkan dasar yang cukup mengenai cara menulis teks pada halaman dokumen PDF. Kali ini kita akan mempelajari cara penempatan gambar pada dokumen PHP. PDFLib mampu menangani beberapa format gambar untuk dijadikan dokumen PDF, antara lain:

- **PNG** (tanpa *alpha-channel*)
- **JPEG** (Progressive jpegs didukung mulai versi Acrobat 4)
- **GIF** (non-interlacing diperbolehkan, untuk animasi GIF hanya gambar pertama yang ditampilkan)
- **TIFF**
- **CCITT compressed image data**
- **Raw image data**

Anda dapat menyisipkan gambar pada dokumen PDF dengan perintah `pdf_open_jpeg()` untuk membuka file gambar dan diikuti dengan perintah `pdf_place_image()` untuk meletakkan gambarnya pada dokumen PDF. Untuk menutup file gambar, digunakan perintah `pdf_close_image()`. Kode berikut ini adalah contohnya.

```
<?php
    $halaman = pdf_new();
    pdf_open_file($halaman);
    pdf_begin_page($halaman,595,842);

    // Load file gambar php-big.jpg
    $im = pdf_open_jpeg($halaman, "php-big.jpg");
    pdf_place_image($halaman, $im, 200, 700, 1.0);
    pdf_place_image($halaman, $im, 200, 600, 0.75);
    pdf_place_image($halaman, $im, 200, 535, 0.50);
    pdf_place_image($halaman, $im, 200, 501, 0.25);
    pdf_place_image($halaman, $im, 200, 486, 0.10);
    $x = pdf_get_value($halaman, "imagewidth", $im);
    $y = pdf_get_value($halaman, "imageheight", $im);
    pdf_close_image ($halaman,$im);

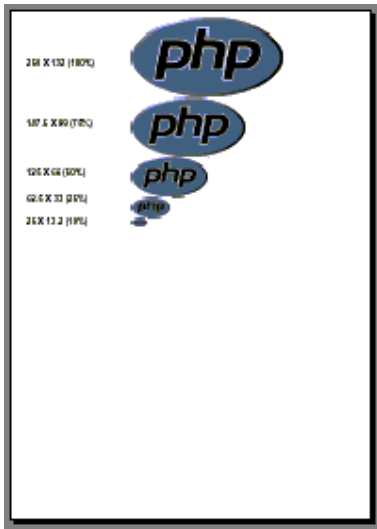
    $huruf = pdf_findfont($halaman,"Helvetica-Bold","host",0);
    pdf_setfont($halaman,$huruf,14.0);
    pdf_show_xy($halaman,"$x X $y"." (100%)",25,750);
    pdf_show_xy($halaman, $x*0.75 . " X " . $y*0.75 . "
    (75%)",25,650);
    pdf_show_xy($halaman, $x*0.50 . " X " . $y*0.50 . "12
```

```
(50%)" ,25,570);
pdf_show_xy($halaman, $x*0.25 . " X " . $y*0.25 . "
(25%)" ,25,525);
pdf_show_xy($halaman, $x*0.10 . " X " . $y*0.10 . "
(10%)" ,25,490);

pdf_end_page($halaman);
pdf_set_parameter($halaman, "openaction", "fitpage");
pdf_close($halaman);
$buf = pdf_get_buffer($halaman);
$panjangbuffer = strlen($buf);
Header("Content-type:application/pdf");
Header("Content-Length:$panjangbuffer");
Header("Content-Disposition:inline;
filename=sisipgambar.pdf");
echo $buf;
pdf_delete($halaman);

?>
```

Hasil tampilan program di atas kurang lebih seperti gambar berikut.

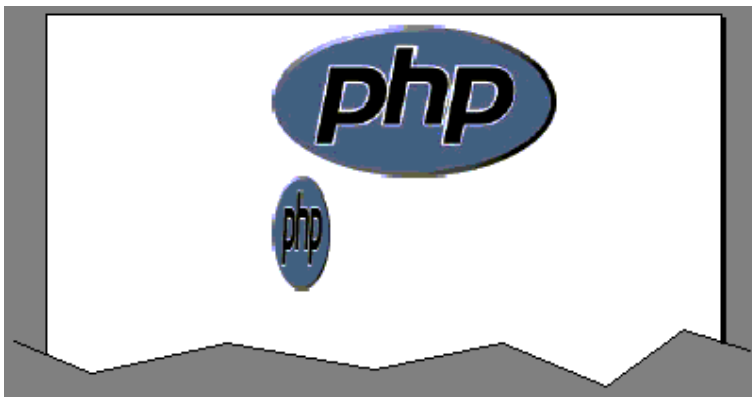


Kita dapat pula menerapkan skala non linier kepada gambar yang akan kita sisipkan pada dokumen PDF. Untuk melakukan itu, kita harus mengatur skala koordinat.

```
<?php
$halaman = pdf_new();
pdf_open_file($halaman);
pdf_begin_page($halaman,595,842);
```

```
$im = pdf_open_jpeg($halaman, "php-big.jpg");  
pdf_place_image($halaman, $im, 200, 700, 1.0);  
pdf_save($halaman); // Simpan Setting Sistem Koordinat yang  
ada  
$nx = 50/pdf_get_value($halaman,"imagewidth",$im);  
$ny = 100/pdf_get_value($halaman,"imageheight",$im);  
pdf_scale($halaman, $nx, $ny);  
pdf_place_image($halaman, $im, 200/$nx, 600/$ny, 1.0);  
pdf_restore($halaman); // Kembalikan ke sebelumnya  
pdf_close_image ($halaman,$im);  
  
pdf_end_page($halaman);  
pdf_set_parameter($halaman, "openaction", "fitpage");  
pdf_close($halaman);  
$buf = pdf_get_buffer($halaman);  
$panjangbuffer = strlen($buf);  
Header("Content-type:application/pdf");  
Header("Content-Length:$panjangbuffer");  
Header("Content-Disposition:inline;  
filename=sisipgambar2.pdf");  
echo $buf;  
pdf_delete($halaman);  
  
?>
```

Program di atas akan memberikan hasil kurang lebih seperti gambar berikut.



1.8. Mari Menggambar Bersama

Selain menggunakan gambar yang telah ada untuk disisipkan ke dokumen PDF, Anda dapat juga langsung menggambar pada dokumen PDF dengan beberapa perintah gambar sederhana. Pada dasarnya, menggambar di dokumen PDF adalah menentukan atau mendefinisikan *path* untuk kemudian ditampilkan pada dokumen PDF. Sebuah *path* terdiri dari bentuk-bentuk dasar grafis seperti garis, kurva, persegi panjang, lingkaran,

elips, dan lain-lain. Sebagai contoh:

```
<?php
$halaman = pdf_new();
pdf_open_file($halaman);
pdf_begin_page($halaman,595,342);

// Menggambar path garis diikuti kurva
pdf_moveto($halaman,150,250);
pdf_lineto($halaman,450,250);
pdf_lineto($halaman,100,300);
pdf_curveto($halaman,80,50,70,50,250,150);

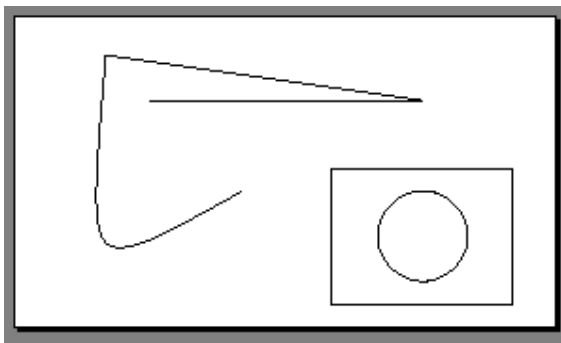
// Menggambar lingkaran
pdf_circle($halaman,450,100,50);

// Menggambar persegi panjang
pdf_rect($halaman,350,25,200,150);

// Memplot semua path/kurva yang telah dibuat
// pada dokumen PDF
pdf_stroke($halaman);

pdf_end_page($halaman);
pdf_close($halaman);
$buf = pdf_get_buffer($halaman);
$panjangbuffer = strlen($buf);
Header("Content-type:application/pdf");
Header("Content-Length:$panjangbuffer");
Header("Content-Disposition:inline;
filename=menggambar1.pdf");
echo $buf;
pdf_delete($halaman);
?>
```

Jika skrip di atas dijalankan, maka akan terlihat hasil sebagai berikut.



Anda dapat menggunakan perintah `pdf_closepath()` untuk menutup *path* secara otomatis dan `pdf_fill_stroke()` untuk mengisi kurva tertutup tersebut dengan warna isian.

```
<?php
    $halaman = pdf_new();
    pdf_open_file($halaman);
    pdf_begin_page($halaman,595,342);

    // Set warna isian
    pdf_setcolor($halaman,"fill","rgb", 1.0, 0.8, 0.1);

    // Menggambar path garis diikuti kurva
    pdf_moveto($halaman,150,250);
    pdf_lineto($halaman,450,250);
    pdf_lineto($halaman,100,300);
    pdf_curveto($halaman,80,50,70,50,250,150);

    // Menutup kurva/path
    pdf_closepath($halaman);

    // Memplot dengan isian warna
    pdf_fill_stroke($halaman);

    // Set warna isian
    pdf_setcolor($halaman,"fill","rgb", 1.0, 0.0, 0.0);

    // Menggambar lingkaran
    pdf_circle($halaman,450,100,50);

    // Memplot dengan isian warna
    pdf_fill_stroke($halaman);

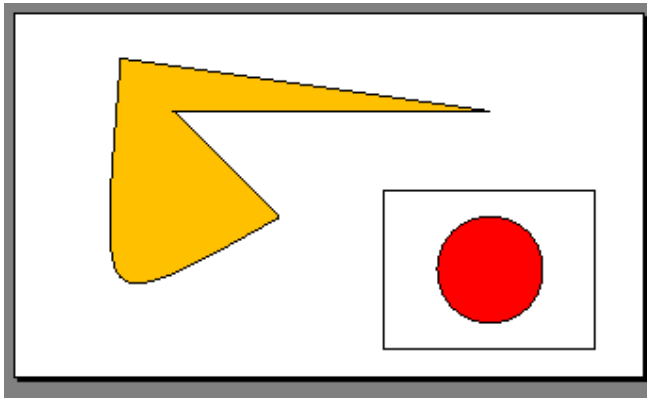
    // Menggambar persegi panjang
    pdf_rect($halaman,350,25,200,150);

    // Memplot tanpa isian warna
    pdf_stroke($halaman);

    pdf_end_page($halaman);
    pdf_close($halaman);
    $buf = pdf_get_buffer($halaman);
    $panjangbuffer = strlen($buf);
    Header("Content-type:application/pdf");
    Header("Content-Length:$panjangbuffer");
    Header("Content-Disposition:inline;
    filename=menggambar2.pdf");
    echo $buf;
    pdf_delete($halaman);
```


?>

Hasil dari skrip di atas adalah dokumen PDF seperti berikut ini.



Mari kita coba contoh menggambar berikutnya. Anda dapat membuat garis putus-putus dengan perintah `pdf_setdash()`.

```
<?php
$halaman = pdf_new();
pdf_open_file($halaman);
pdf_begin_page($halaman,595,342);

// Lingkaran
pdf_setcolor($halaman,"fill","rgb", 0.8, 0.5, 0.8);
pdf_circle($halaman,400,150,75);
pdf_fill_stroke($halaman);

// Funky Arc
pdf_setcolor($halaman,"fill","rgb", 0.8, 0.5, 0.5);
pdf_moveto($halaman,200,150);
pdf_arc($halaman,300,150,50,0,120);
pdf_closepath($halaman);
pdf_fill_stroke($halaman);

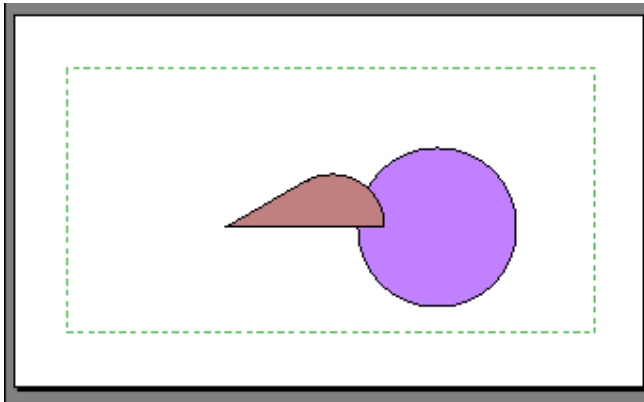
// kotak dengan garis putus-putus
pdf_setcolor($halaman,"stroke","rgb", 0.3, 0.8, 0.3);
pdf_setdash($halaman,4,6);
pdf_rect($halaman,50,50,500,250);
pdf_stroke($halaman);

pdf_end_page($halaman);
pdf_close($halaman);
$buf = pdf_get_buffer($halaman);
$panjangbuffer = strlen($buf);
```

17

```
Header("Content-type:application/pdf");  
Header("Content-Length:$panjangbuffer");  
Header("Content-Disposition:inline;  
filename=menggambar3.pdf");  
echo $buf;  
pdf_delete($halaman);  
?>
```

Inilah hasil gambar dari skrip di atas.



1.9. Satu Untuk Semua, Semuanya Dari Satu

Adakalanya kita ingin mencetak bentuk, gambar atau tulisan secara berulang pada beberapa halaman PDF yang kita akan buat. Naluri kepemalasan kita akan berontak. Bisa tidak semua perulangan ini dibuat lebih ringkas dan mudah?

Pada kondisi ini maka kita membutuhkan pola baku alias *template*. Dalam dunia PDF, *template* dikenal sebagai **form XObjects**. Dokumen PDF yang dibuat akan menjadi lebih kecil ukurannya dengan memanfaatkan *template*. Contoh skrip berikut ini dapat coba Anda pahami.

```
<?php  
$halaman = pdf_new();  
pdf_open_file($halaman);  
  
// Muat Gambar/Logo PHP  
$gambar = pdf_open_jpeg($halaman, "php-big.jpg");  
  
// Memulai Template  
$template = pdf_begin_template($halaman,595,442);  
pdf_save($halaman);  
  
pdf_place_image($halaman, $gambar, 4, 403, 0.25);
```

18

```
pdf_place_image($halaman, $gambar, 525, 403, 0.25);

pdf_moveto($halaman,10,395);
pdf_lineto($halaman,585,395);
pdf_lineto($halaman,585,10);
pdf_lineto($halaman,10,10);
pdf_closepath($halaman);
pdf_stroke($halaman);
pdf_moveto($halaman,10,375);
pdf_lineto($halaman,585,375);
pdf_stroke($halaman);

$font = pdf_findfont($halaman,"Times-Bold","host",0);
pdf_setfont($halaman,$font,38.0);
pdf_show_xy($halaman,"Contoh Template PDF",100,407);
pdf_restore($halaman);
pdf_end_template($halaman);

// Tutup Gambar/Logo PHP
pdf_close_image ($halaman,$gambar);

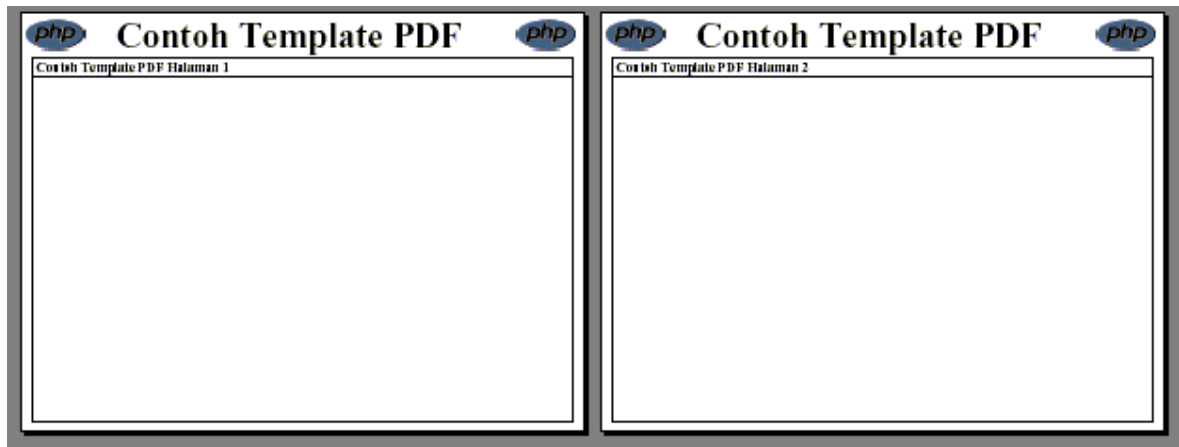
// Halaman Pertama
pdf_begin_page($halaman,595,442);
pdf_place_image($halaman, $template, 0, 0, 1.0);
pdf_setfont($halaman,$font,14.0);
pdf_show_xy($halaman,"Contoh Template PDF Halaman 1",15,380);
pdf_end_page($halaman);

// Halaman Kedua
pdf_begin_page($halaman,595,442);
pdf_place_image($halaman, $template, 0, 0, 1.0);
pdf_setfont($halaman,$font,14.0);
pdf_show_xy($halaman,"Contoh Template PDF Halaman 2",15,380);
pdf_end_page($halaman);

pdf_close($halaman);
$buf = pdf_get_buffer($halaman);
$panjangbuffer = strlen($buf);
Header("Content-type:application/pdf");
Header("Content-Length:$panjangbuffer");
Header("Content-Disposition:inline;
filename=template.pdf");
echo $buf;
pdf_delete($halaman);

?>
```

Dokumen PDF dua halaman akan terbentuk sebagai berikut.



Contoh aplikasi dalam dunia nyata adalah untuk dokumen atau pencetakan *invoice*, kuitansi, *packing list*, dan dokumen-dokumen lain yang berbentuk form baku.

1.10. Mewarnai dengan Pola

Pola isian mirip dengan pola baku/*templates*, hanya saja pola isian digunakan untuk pengganti warna isian. Anda dapat menggambar kurva/path, garis atau bentuk primitif kurva lainnya (*stroke*), dan mengisi warna kurva/path tersebut dengan sebuah pola isian. Contoh berikut ini akan mengobarkan kembali semangat PHP kita.

```
<?php
    $halaman = pdf_new();
    pdf_open_file($halaman);

    // Muat gambar untuk pola isian

    $gambar = pdf_open_jpeg($halaman, "php-big.jpg");

    // Membuat pola
    $pola = pdf_begin_pattern($halaman,21,14,25,18,1);
    pdf_save($halaman);
    pdf_place_image($halaman, $gambar, 0,0,0.08);
    pdf_restore($halaman);
    pdf_end_pattern($halaman);

    // Tutup gambar
    pdf_close_image ($halaman,$gambar);

    pdf_begin_page($halaman,595,842);

    // Gunakan pola untuk isian dan garis gambar
```

```
pdf_setcolor($halaman, "fill", "pattern", $pola);  
pdf_setcolor($halaman, "stroke", "pattern", $pola);  
  
pdf_setlinewidth($halaman, 60.0);  
pdf_circle($halaman, 200, 680, 100);  
pdf_stroke($halaman);  
  
pdf_end_page($halaman);  
  
pdf_close($halaman);  
$buf = pdf_get_buffer($halaman);  
$panjangbuffer = strlen($buf);  
Header("Content-type:application/pdf");  
Header("Content-Length:$panjangbuffer");  
Header("Content-Disposition:inline;  
filename=polaisian.pdf");  
echo $buf;  
pdf_delete($halaman);  
  
?>
```

1.11. Tandai Yang Penting

Anda dapat menandai bagian-bagian yang penting dalam dokumen PDF dengan *bookmark*, sehingga navigasi dokumen menjadi lebih mudah. *Bookmark* ini bisa berbentuk struktur pohon (*tree structure*) menyerupai daftar isi. Selain berupa *bookmark*, Anda dapat juga mempermudah navigasi dokumen dengan membuat daftar *thumbnail* halaman-halaman yang ada pada dokumen PDF. Contoh berikut ini akan membantu untuk mempelajari caranya.

```
<?php  
$halaman = pdf_new();  
pdf_open_file($halaman);  
  
// Halaman Pertama  
pdf_begin_page($halaman, 595, 842);  
  
// Set Bookmark Awal dan Jenis Huruf  
$top = pdf_add_bookmark($halaman, "Sistem Operasi");  
$font = pdf_findfont($halaman, "Helvetica-Bold", "host", 0);  
  
$gambar = pdf_open_jpeg($halaman, "freebsd.jpg");  
pdf_add_thumbnail($halaman, $gambar);  
pdf_setfont($halaman, $font, 20);  
pdf_add_bookmark($halaman, "FreeBSD", $top);  
pdf_show_xy($halaman, "Ini adalah halaman tentang FreeBSD",  
50, 670);  
pdf_place_image($halaman, $gambar, 50, 700, 1);  
pdf_close_image($halaman, $gambar);  
pdf_end_page($halaman);
```

```
// Halaman Kedua
pdf_begin_page($halaman,595,842);
$gambar = pdf_open_jpeg($halaman, "linux.jpg");
pdf_add_thumbnail($halaman, $gambar);
pdf_setfont($halaman, $font, 20);
pdf_add_bookmark($halaman, "Linux", $top);
pdf_show_xy($halaman, "Ini adalah halaman tentang Linux", 50,
670);
pdf_place_image($halaman, $gambar, 50, 700, 1);
pdf_close_image($halaman,$gambar);
pdf_end_page($halaman);

// Halaman Ketiga
pdf_begin_page($halaman,595,842);
$gambar = pdf_open_jpeg($halaman, "mac.jpg");
pdf_add_thumbnail($halaman, $gambar);
pdf_setfont($halaman, $font, 20);
pdf_add_bookmark($halaman, "Mac", $top);
pdf_show_xy($halaman, "Ini adalah halaman tentang Mac", 50,
670);
pdf_place_image($halaman, $gambar, 50, 700, 1);
pdf_close_image($halaman,$gambar);
pdf_end_page($halaman);

pdf_close($halaman);
$buf = pdf_get_buffer($halaman);
$panjangbuffer = strlen($buf);
Header("Content-type:application/pdf");
Header("Content-Length:$panjangbuffer");
Header("Content-Disposition:inline;
filename=tandabuku.pdf");
echo $buf;
pdf_delete($halaman);

?>
```

1.12. Habis Ini, Terus...

Jika Anda telah mempelajari artikel ini sampai pada tahap ini, maka Anda telah memiliki dasar yang cukup untuk membuat dokumen PDF dengan PHP dan PDFLib. Keterangan mengenai fungsi-fungsi PDF yang digunakan dalam skrip contoh di artikel ini dapat Anda pelajari lebih detil pada Manual PHP yang tersedia di <http://www.php.net>. Selamat menjadi PDF Maker!

Biografi dan Profil



Ivan Irawan. Lulus Cum Laude dari Teknik Nuklir Universitas Gadjah Mada tahun 1995. Tinggal di Semarang, memiliki pengalaman lebih dari 7 tahun pada operasional bank swasta nasional. Saat ini mendirikan usaha konsultan sistem informasi [AtlantisIndonesia](http://AtlantisIndonesia.com) yang berbasis di Semarang. Konsultan Teknologi Informasi dan Manajemen Proses Operasi pada beberapa bank, institusi pembiayaan/finansial, dan perusahaan manufaktur. Mendalami berbagai bahasa pemrograman Visual Basic, Delphi/Kylix, C/C++, Java, dan PHP.

Ivan Irawan adalah programmer AI-Billinet, sistem billing dan administrasi warnet yang telah digunakan di lebih dari 100 warung internet di seluruh Indonesia. Aktif di Kelompok Pengguna Linux Semarang (KPLi Semarang) dan telah menulis beberapa buku komputer dan artikel-artikel pemrograman.

Informasi lebih lanjut tentang penulis ini bisa didapat melalui:

URL: <http://skrip.php.xoasis.com>

Email: rii@ai.co.id