

# Janaan Laporan

cara nak guna plugin untuk jana laporan dalam bentuk pdf dll

- [PDF](#)
- [barryvdh/laravel-dompdf](#)
- [setasign/fpdf](#)

# PDF

untuk menjana cetakan laporan dalam bentuk PDF, ada beberapa plugin yang boleh digunakan. dalam laravel, boleh gunakan package `barryvdh/laravel-dompdf` atau `setasign/fpdf`. Jika prefer javascript, boleh juga gunakan [PDFMake](#).

Perbezaan ketiga-tiga package ini adalah:

DOMPDF : pdf dijana daripada blade(html). walau bagaimanapun, hanya simple css sahaja yang boleh digunakan untuk susunan output.

FPDF : output perlu disusun menggunakan fungsi.

PDFMake : output diletakkan dalam blade sebagai Javascript.

# barryvdh/laravel-dompdf

install package menggunakan command ini:

```
composer require barryvdh/laravel-dompdf
```

dalam controller, jangan lupa include DomPDF. kemudian load data macam kita biasa buat untuk view. Perbedaannya kat sini, kita panggil PDF::loadView('nama\_blade', [data]) instead of view().

```
use Barryvdh\DomPDF\Facade\Pdf;
```

```
class ReportController extends Controller
```

```
{
```

```
    public function download()
```

```
    {
```

```
        $data = [
```

```
            'title' => 'Laporan Bulanan',
```

```
            'date' => date('d/m/Y'),
```

```
            'content' => 'Ini contoh report PDF.'
```

```
        ];
```

```
        $pdf = Pdf::loadView('pdf.report', $data);
```

```
        return $pdf->download('laporan.pdf'); // download
```

```
        // return $pdf->stream('laporan.pdf'); // terus buka dalam browser
```

```
    }
```

```
}
```

untuk gambar dalam output tu, guna public\_path('img/logo.png');

tapi kalau kita nak guna asset, kena enable dulu option isRemoteEnabled untuk remote. kat bawah ni contoh code untuk enable gambar guna asset();

```
// apply option masa create instance
```

```
$pdf = \PDF::setOptions([
```

```
    'isRemoteEnabled' => true,
```

```
    'defaultFont' => 'sans-serif'
```

```
])->loadHTML($pdfView)
```

```
->setPaper('a4', 'portrait');
```

# setasign/fpdf

install package guna composer:

```
composer require setasign/fpdf
```

dalam controller, kena buat satu-satu line output macam kat bawah

```
<?php
```

```
namespace App\Http\Controllers;
```

```
use Illuminate\Http\Request;
```

```
use setasign\Fpdf\Fpdf; // kalau nak guna FPDF untuk import pdf lain
```

```
use FPDF;
```

```
class PdfController extends Controller
```

```
{
```

```
    public function generate()
```

```
    {
```

```
        $pdf = new \FPDF();
```

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

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

```
        // Title
```

```
        $pdf->Cell(0, 10, 'Hello dari Laravel + FPDF!', 0, 1, 'C');
```

```
        // Normal text
```

```
        $pdf->SetFont('Arial', '', 12);
```

```
        $pdf->Cell(0, 10, 'Ini contoh content dalam PDF.', 0, 1);
```

```
        // Output to browser
```

```
        $pdf->Output('I', 'contoh.pdf');
```

```
        exit;
```

```
    }
```

```
}
```

lepas create objek FPDF(), tambah page dulu guna AddPage(). untuk next page, kita kene panggil AddPage() lagi.

panggil Cell() untuk add baris dalam pdf tu.

untuk output, panggil Output().

