

L

A

M

P

I

R

A

N


```

@extends('layouts.master')

@section('title')
    Dashboard
@endsection

@section('breadcrumb')
    @parent
    <li
class="active">Dashboard</li>
@endsection

@section('content')
<!-- Small boxes (Stat box) -->
<div class="row">
    <div class="col-lg-3 col-xs-
6">

        <!-- small box -->
        <div class="small-box bg-
aqua">

            <div class="inner">
                <h3>{{ $kategori
}}</h3>

                <p>Total
Kategori</p>

                </div>
                <div class="icon">
                    <i class="fa fa-
cube"></i>

                </div>
                <a href="{{
route('kategori.index') }}"
class="small-box-footer">Lihat <i
class="fa fa-arrow-circle-
right"></i></a>

                </div>
            </div>
            <!-- ./col -->
            <div class="col-lg-3 col-xs-
6">

                <!-- small box -->
                <div class="small-box bg-
green">

                    <div class="inner">
                        <h3>{{ $produk
}}</h3>

                        <p>Total
Produk</p>

                        </div>
                        <div class="icon">
                            <i class="fa fa-
cubes"></i>

                        </div>

```

```

        <a href="{{
route('produk.index') }}"
class="small-box-footer">Lihat <i
class="fa fa-arrow-circle-
right"></i></a>

        </div>
    </div>
    <!-- ./col -->
    <div class="col-lg-3 col-xs-
6">

        <!-- small box -->
        <div class="small-box bg-
yellow">

            <div class="inner">
                <h3>{{ $member
}}</h3>

                <p>Total
Member</p>

                </div>
                <div class="icon">
                    <i class="fa fa-
id-card"></i>

                </div>
                <a href="{{
route('member.index') }}"
class="small-box-footer">Lihat <i
class="fa fa-arrow-circle-
right"></i></a>

                </div>
            </div>
            <!-- ./col -->
            <div class="col-lg-3 col-xs-
6">

                <!-- small box -->
                <div class="small-box bg-
red">

                    <div class="inner">
                        <h3>{{ $supplier
}}</h3>

                        <p>Total
Supplier</p>

                        </div>
                        <div class="icon">
                            <i class="fa fa-
truck"></i>

                        </div>
                        <a href="{{
route('supplier.index') }}"
class="small-box-footer">Lihat <i
class="fa fa-arrow-circle-
right"></i></a>

                        </div>
                    </div>

```

```

        <!-- ./col -->
    </div>
    <!-- /.row -->
    <!-- Main row -->
    <div class="row">
        <div class="col-lg-12">
            <div class="box">
                <div class="box-
header with-border">
                    <h3 class="box-
title">Grafik Pendapatan {{
tanggal_indonesia($tanggal_awal,
false) }} s/d {{
tanggal_indonesia($tanggal_akhir,
false) }}</h3>
                </div>
                <!-- /.box-header -->
                <div class="box-
body">
                    <div class="row">
                        <div
class="col-lg-12">
                            <div
class="chart">
                                <!--
Sales Chart Canvas -->
                                <canvas id="salesChart"
style="height: 180px;"></canvas>
                                </div>
                                <!--
/.chart-responsive -->
                                </div>
                            </div>
                        <!-- /.row -->
                    </div>
                </div>
            <!-- /.box -->
        </div>
    <!-- /.col -->
</div>
<!-- /.row (main row) -->
@endsection

@push('scripts')
<!-- ChartJS -->
<script src="{{ asset('AdminLTE-
2/bower_components/chart.js/Chart
.js') }}"></script>
<script>
$(function() {
    // Get context with jQuery -
using jQuery's .get() method.
    var salesChartCanvas =
$('#salesChart').get(0).getContext

```

```

t('2d');
    // This will get the first
returned node in the jQuery
collection.
    var salesChart = new
Chart(salesChartCanvas);

    var salesChartData = {
        labels: {{
json_encode($data_tanggal) }},
        datasets: [
            {
                label:
'Pendapatan',
                fillColor
: 'rgba(60,141,188,0.9)',
                strokeColor
: 'rgba(60,141,188,0.8)',
                pointColor
: '#3b8bba',
                pointStrokeColor
: 'rgba(60,141,188,1)',
                pointHighlightFill : '#fff',
                pointHighlightStroke:
'rgba(60,141,188,1)',
                data: {{
json_encode($data_pendapatan) }}
            }
        ]
    };

    var salesChartOptions = {
        pointDot : false,
        responsive : true
    };

    salesChart.Line(salesChartData,
salesChartOptions);
});
</script>
@endpush

<?php
use App\Http\Controllers\{
    DashboardController,
    KategoriController,
    LaporanController,
    ProdukController,
    MemberController,
    PengeluaranController,
    PembelianController,

```

```

        PembelianDetailController,
        PenjualanController,
        PenjualanDetailController,
        SettingController,
        SupplierController,
        UserController,
    };
    use
    Illuminate\Support\Facades\Route;

    /*
    |-----
    |-----
    | Web Routes
    |-----
    |-----
    |
    | Here is where you can register
    | web routes for your application.
    | These
    | routes are loaded by the
    | RouteServiceProvider within a
    | group which
    | contains the "web" middleware
    | group. Now create something
    | great!
    |
    */

    Route::get('/', function () {
        return redirect()-
    >route('login');
    });

    Route::group(['middleware' =>
    'auth'], function () {
        Route::get('/dashboard',
        [DashboardController::class,
        'index'])->name('dashboard');

        Route::group(['middleware' =>
        'level:1'], function () {

            Route::get('/kategori/data',
            [KategoriController::class,
            'data'])->name('kategori.data');

            Route::resource('/kategori',
            KategoriController::class);

            Route::get('/produk/data',
            [ProdukController::class,
            'data'])->name('produk.data');

            Route::post('/produk/delete-
            selected',
            [ProdukController::class,
            'deleteSelected'])-
            >name('produk.delete_selected');

            Route::post('/produk/cetak-
            barcode',
            [ProdukController::class,
            'cetakBarcode'])-
            >name('produk.cetak_barcode');

            Route::resource('/produk',
            ProdukController::class);
            Route::get('/produk-
            kritis',
            [ProdukController::class,
            'kritis'])-
            >name('produk.kritis');
            Route::get('/produk-
            riwayat',
            [ProdukController::class,
            'riwayatStok'])-
            >name('produk.riwayat');
            Route::get('/riwayat-
            data', [ProdukController::class,
            'riwayatData'])-
            >name('produk.riwayatData');

            Route::get('/member/data',
            [MemberController::class,
            'data'])->name('member.data');

            Route::post('/member/cetak-
            member',
            [MemberController::class,
            'cetakMember'])-
            >name('member.cetak_member');

            Route::resource('/member',
            MemberController::class);

            Route::get('/supplier/data',
            [SupplierController::class,
            'data'])->name('supplier.data');

            Route::resource('/supplier',
            SupplierController::class);

            Route::get('/pengeluaran/data',

```

```

[PengeluaranController::class,
'data'])-
>name('pengeluaran.data');

Route::resource('/pengeluaran',
PengeluaranController::class);

Route::get('/pembelian/data',
[PembelianController::class,
'data']->name('pembelian.data'));

Route::get('/pembelian/{id}/create', [PembelianController::class,
'create'])-
>name('pembelian.create');

Route::resource('/pembelian',
PembelianController::class)
->except('create');

Route::get('/pembelian_detail/{id}/data',
[PembelianDetailController::class,
'data'])-
>name('pembelian_detail.data');

Route::get('/pembelian_detail/loadform/{diskon}/{total}',
[PembelianDetailController::class,
'loadForm'])-
>name('pembelian_detail.load_form');

Route::resource('/pembelian_detail',
PembelianDetailController::class)
->except('create',
'show', 'edit');

Route::get('/penjualan/data',
[PenjualanController::class,
'data']->name('penjualan.data');
Route::get('/penjualan',
[PenjualanController::class,
'index'])-
>name('penjualan.index');

Route::get('/penjualan/{id}',
[PenjualanController::class,
'show']->name('penjualan.show');

Route::delete('/penjualan/{id}',

```

```

[PenjualanController::class,
'destroy'])-
>name('penjualan.destroy');
});

Route::group(['middleware' =>
'level:1,2'], function () {

Route::get('/transaksi/baru',
[PenjualanController::class,
'create'])-
>name('transaksi.baru');

Route::post('/transaksi/simpan',
[PenjualanController::class,
'store'])-
>name('transaksi.simpan');

Route::get('/transaksi/selesai',
[PenjualanController::class,
'seleasai'])-
>name('transaksi.seleasai');

Route::get('/transaksi/nota-kecil',
[PenjualanController::class,
'notaKecil'])-
>name('transaksi.nota_kecil');

Route::get('/transaksi/nota-besar',
[PenjualanController::class,
'notaBesar'])-
>name('transaksi.nota_besar');

Route::get('/transaksi/{id}/data',
[PenjualanDetailController::class,
'data'])-
>name('transaksi.data');

Route::get('/transaksi/loadform/{diskon}/{total}/{diterima}',
[PenjualanDetailController::class,
'loadForm'])-
>name('transaksi.load_form');

Route::resource('/transaksi',
PenjualanDetailController::class)
->except('create',
'show', 'edit');
});

Route::group(['middleware' =>

```

```

'level:1'], function () {
    Route::get('/laporan',
[LaporanController::class,
'index'])->name('laporan.index');

Route::get('/laporan/data/{awal}/{
akhir}',
[LaporanController::class,
'data'])->name('laporan.data');

Route::get('/laporan/pdf/{awal}/{
akhir}',
[LaporanController::class,
'exportPDF'])->
name('laporan.export_pdf');

Route::get('/user/data',
[UserController::class, 'data'])->
name('user.data');
Route::resource('/user',
UserController::class);

Route::get('/setting',
[SettingController::class,
'index'])->name('setting.index');

Route::get('/setting/first',
[SettingController::class,
'show'])->name('setting.show');
Route::post('/setting',
[SettingController::class,
'update'])->
name('setting.update');
});

Route::group(['middleware' =>
'level:1,2'], function () {
    Route::get('/profil',
[UserController::class,
'profil'])->name('user.profil');
Route::post('/profil',
[UserController::class,
'updateProfil'])->
name('user.update_profil');
});
});

```

```
<?php
```

```

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\User;
use
Illuminate\Support\Facades\Hash;

```

```

class UserController extends
Controller
{
    public function index()
    {
        return
view('user.index');
    }

    public function data()
    {
        $user =
User::isNotAdmin()->orderBy('id',
'desc')->get();

        return datatables()
->of($user)
->addIndexColumn()
->addColumn('aksi',
function ($user) {
            return '
<div class="btn-
group">

                <button
type="button"
onclick="editForm(`'.
route('user.update', $user->id)
.'`)" class="btn btn-xs btn-info
btn-flat"><i class="fa fa-
pencil"></i></button>

                <button
type="button"
onclick="deleteData(`'.
route('user.destroy', $user->id)
.'`)" class="btn btn-xs btn-
danger btn-flat"><i class="fa fa-
trash"></i></button>

            </div>
';
        })
        -
>rawColumns(['aksi'])
->make(true);
    }

    /**
     * Show the form for creating
     a new resource.
     *
     * @return
     \Illuminate\Http\Response
     */
    public function create()
    {

```

```

        //
    }

    /**
     * Store a newly created
    resource in storage.
     *
     * @param
    \Illuminate\Http\Request
    $request
     * @return
    \Illuminate\Http\Response
    */
    public function store(Request
    $request)
    {
        $user = new User();
        $user->name = $request-
    >name;
        $user->email = $request-
    >email;
        $user->password =
    bcrypt($request->password);
        $user->level = 2;
        $user->foto =
    '/img/user.jpg';
        $user->save();

        return response()-
    >json('Data berhasil disimpan',
    200);
    }

    /**
     * Display the specified
    resource.
     *
     * @param int $id
     * @return
    \Illuminate\Http\Response
    */
    public function show($id)
    {
        $user = User::find($id);

        return response()-
    >json($user);
    }

    /**
     * Show the form for editing
    the specified resource.
     *
     * @param int $id
     * @return

```

```

    \Illuminate\Http\Response
    */
    public function edit($id)
    {
        //
    }

    /**
     * Update the specified
    resource in storage.
     *
     * @param
    \Illuminate\Http\Request
    $request
     * @param int $id
     * @return
    \Illuminate\Http\Response
    */
    public function
    update(Request $request, $id)
    {
        $user = User::find($id);
        $user->name = $request-
    >name;
        $user->email = $request-
    >email;
        if ($request-
    >has('password') && $request-
    >password != "")
            $user->password =
    bcrypt($request->password);
        $user->update();

        return response()-
    >json('Data berhasil disimpan',
    200);
    }

    /**
     * Remove the specified
    resource from storage.
     *
     * @param int $id
     * @return
    \Illuminate\Http\Response
    */
    public function destroy($id)
    {
        $user = User::find($id)-
    >delete();

        return response(null,
    204);
    }

```

```

        public function profil()
        {
            $profil = auth()->user();
            return
            view('user.profil',
            compact('profil'));
        }

        public function
        updateProfil(Request $request)
        {
            $user = auth()->user();

            $user->name = $request-
            >name;

            if ($request-
            >has('password') && $request-
            >password != "") {
                if
                (Hash::check($request-
                >old_password, $user->password))
                {
                    if ($request-
                    >password == $request-
                    >password_confirmation) {
                        $user-
                        >password = bcrypt($request-
                        >password);
                    } else {
                        return
                        response()->json('Konfirmasi
                        password tidak sesuai', 422);
                    }
                } else {
                    return
                    response()->json('Password lama
                    tidak sesuai', 422);
                }
            }

            if ($request-
            >hasFile('foto')) {
                $file = $request-
                >file('foto');
                $nama = 'logo-' .
                date('YmdHis') . '.' . $file-
                >getClientOriginalExtension();
                $file-
                >move(public_path('/img'),
                $nama);

                $user->foto =
                "/img/$nama";
            }
        }
    }

```

```

        $user->update();

        return response()-
        >json($user, 200);
    }
}

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Supplier;

class SupplierController extends
Controller
{
    public function index()
    {
        return
        view('supplier.index');
    }

    public function data()
    {
        $supplier =
        Supplier::orderBy('id_supplier',
        'desc')->get();

        return datatables()-
        >of($supplier)
        >addIndexColumn()
        >addColumn('aksi',
        function ($supplier) {
            return '
            <div class="btn-
            group">

                <button
                type="button"
                onclick="editForm(`'.
                route('supplier.update',
                $supplier->id_supplier) .`')"
                class="btn btn-xs btn-info btn-
                flat"><i class="fa fa-
                pencil"></i></button>

                <button
                type="button"
                onclick="deleteData(`'.
                route('supplier.destroy',
                $supplier->id_supplier) .`')"
                class="btn btn-xs btn-danger btn-
                flat"><i class="fa fa-
                trash"></i></button>
            </div>
            '
            ;
        }
    }
}

```

```

        })
        -
>rawColumns(['aksi'])
        ->make(true);
    }

    /**
     * Show the form for creating
     a new resource.
     *
     * @return
     \Illuminate\Http\Response
     */
    public function create()
    {
        //
    }

    /**
     * Store a newly created
     resource in storage.
     *
     * @param
     \Illuminate\Http\Request
     $request
     * @return
     \Illuminate\Http\Response
     */
    public function store(Request
    $request)
    {
        $supplier =
        Supplier::create($request-
        >all());

        return response()-
        >json('Data berhasil disimpan',
        200);
    }

    /**
     * Display the specified
     resource.
     *
     * @param int $id
     * @return
     \Illuminate\Http\Response
     */
    public function show($id)
    {
        $supplier =
        Supplier::find($id);

        return response()-
        >json($supplier);
    }

```

```

    }

    /**
     * Show the form for editing
     the specified resource.
     *
     * @param int $id
     * @return
     \Illuminate\Http\Response
     */
    public function edit($id)
    {
        //
    }

    /**
     * Update the specified
     resource in storage.
     *
     * @param
     \Illuminate\Http\Request
     $request
     * @param int $id
     * @return
     \Illuminate\Http\Response
     */
    public function
    update(Request $request, $id)
    {
        $supplier =
        Supplier::find($id)-
        >update($request->all());

        return response()-
        >json('Data berhasil disimpan',
        200);
    }

    /**
     * Remove the specified
     resource from storage.
     *
     * @param int $id
     * @return
     \Illuminate\Http\Response
     */
    public function destroy($id)
    {
        $supplier =
        Supplier::find($id)->delete();

        return response(null,
        204);
    }
}

```

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Setting;

class SettingController extends Controller
{
    public function index()
    {
        return
view('setting.index');
    }

    public function show()
    {
        return Setting::first();
    }

    public function
update(Request $request)
    {
        $setting =
Setting::first();
        $setting->nama_perusahaan =
$request->nama_perusahaan;
        $setting->telepon =
$request->telepon;
        $setting->alamat =
$request->alamat;
        $setting->diskon =
$request->diskon;
        $setting->tipe_nota =
$request->tipe_nota;

        if ($request-
>hasFile('path_logo')) {
            $file = $request-
>file('path_logo');
            $nama = 'logo-' .
date('YmdHis') . '.' . $file-
>getClientOriginalExtension();
            $file-
>move(public_path('/img'),
$nama);

            $setting->path_logo =
"/img/$nama";
        }

        if ($request-
>hasFile('path_kartu_member')) {

```

```

            $file = $request-
>file('path_kartu_member');
            $nama = 'logo-' .
date('Y-m-dHis') . '.' . $file-
>getClientOriginalExtension();
            $file-
>move(public_path('/img'),
$nama);

            $setting-
>path_kartu_member =
"/img/$nama";
        }

        $setting->update();

        return response()-
>json('Data berhasil disimpan',
200);
    }
}

<?php

namespace App\Http\Controllers;

use App\Models\Kategori;
use Illuminate\Http\Request;
use App\Models\Produk;
use App\Models\RiwayatStok;
use PDF;

class ProdukController extends Controller
{
    /**
     * Display a listing of the
     * resource.
     *
     * @return
     * \Illuminate\Http\Response
     */
    public function index()
    {
        $kategori =
Kategori::all()-
>pluck('nama_kategori',
'id_kategori');

        return
view('produk.index',
compact('kategori'));
    }

    public function kritis()

```

```

        {
            $kategori =
            Kategori::all()-
            >pluck('nama_kategori',
            'id_kategori');

            return
            view('produk.index',
            compact('kategori'));
        }

        public function data(Request
        $request)
        {

            $query =
            Produk::leftJoin('kategori',
            'kategori.id_kategori',
            'produk.id_kategori')
            ->select('produk.*',
            'nama_kategori')
            -
            >orderBy('kode_produk', 'asc');

            if($request->status ==
            'kritis'){
                $query->where('stok'
                , '<', 5);
            }

            $produk = $query->get();

            return datatables()
            ->of($produk)
            ->addIndexColumn()
            -
            >addColumn('select_all', function
            ($produk) {

                return '
                <input

                type="checkbox"
                name="id_produk[]" value="'.
                $produk->id_produk .' ">
                '
                ;

            })
            -
            >addColumn('kode_produk',
            function ($produk) {

                return '<span
                class="label label-success">'.
                $produk->kode_produk .'</span>';

            })
            -
            >addColumn('harga_beli', function
            ($produk) {

```

```

                return
                format_uang($produk->harga_beli);
            })
            -
            >addColumn('harga_jual', function
            ($produk) {

                return
                format_uang($produk->harga_jual);
            })
            ->addColumn('stok',
            function ($produk) {

                return
                format_uang($produk->stok);
            })
            ->addColumn('aksi',
            function ($produk) {

                return '
                <div class="btn-
                group">

                    <button

                    type="button"
                    onclick="editForm(`'.
                    route('produk.update', $produk-
                    >id_produk) .'`)" class="btn btn-
                    xs btn-info btn-flat"><i
                    class="fa fa-
                    pencil"></i></button>

                    <button

                    type="button"
                    onclick="deleteData(`'.
                    route('produk.destroy', $produk-
                    >id_produk) .'`)" class="btn btn-
                    xs btn-danger btn-flat"><i
                    class="fa fa-trash"></i></button>
                    </div>
                    '
                    ;

                })
                ->rawColumns(['aksi',
                'kode_produk', 'select_all'])
                ->make(true);

            }

            /**
             * Show the form for creating
             a new resource.
             *
             * @return
             \Illuminate\Http\Response
             */
            public function create()
            {

                //

            }

            /**

```

```

        * Store a newly created
resource in storage.
    *
    * @param
\Illuminate\Http\Request
$request
    * @return
\Illuminate\Http\Response
    */
    public function store(Request
$request)
    {
        $produk =
Produk::latest()->first() ?? new
Produk();
        $request['kode_produk'] =
'P'.
        tambah_nol_didepan((int)$produk-
>id_produk +1, 6);

        $produk =
Produk::create($request->all());
        RiwayatStok::create([
            'body' => 'Produk ' .
$request->nama_produk . ' telah
ditambahkan ke etalase sebanyak '
. $request->stok . ' buah'
        ]);
        return response()-
>json('Data berhasil disimpan',
200);
    }

    /**
     * Display the specified
resource.
     *
     * @param int $id
     * @return
\Illuminate\Http\Response
     */
    public function show($id)
    {
        $produk =
Produk::find($id);

        return response()-
>json($produk);
    }

    /**
     * Show the form for editing
the specified resource.
     *
     * @param int $id

```

```

        * @return
\Illuminate\Http\Response
    */
    public function edit($id)
    {
        //
    }

    /**
     * Update the specified
resource in storage.
     *
     * @param
\Illuminate\Http\Request
$request
     * @param int $id
     * @return
\Illuminate\Http\Response
     */
    public function
update(Request $request, $id)
    {
        $produk =
Produk::find($id);
        $produk->update($request-
>all());

        return response()-
>json('Data berhasil disimpan',
200);
    }
<?php

namespace App\Http\Controllers;

use App\Models\Member;
use App\Models\Penjualan;
use App\Models\PenjualanDetail;
use App\Models\Produk;
use App\Models\Setting;
use Illuminate\Http\Request;

class PenjualanDetailController
extends Controller
{
    public function index()
    {
        $produk =
Produk::orderBy('nama_produk')-
>get();
        $member =
Member::orderBy('nama')->get();
        $diskon =
Setting::first()->diskon ?? 0;

```

```

        // Cek apakah ada
        transaksi yang sedang berjalan
        if ($id_penjualan =
        session('id_penjualan')) {
            $penjualan =
            Penjualan::find($id_penjualan);
            $memberSelected =
            $penjualan->member ?? new
            Member();

            return
            view('penjualan_detail.index',
            compact('produk', 'member',
            'diskon', 'id_penjualan',
            'penjualan', 'memberSelected'));
        } else {
            if (auth()->user()-
            >level == 1) {
                return
                redirect()-
                >route('transaksi.baru');
            } else {
                return
                redirect()->route('home');
            }
        }

        public function data($id)
        {
            $detail =
            PenjualanDetail::with('produk')
            -
            >where('id_penjualan', $id)
            ->get();

            $data = array();
            $total = 0;
            $total_item = 0;

            foreach ($detail as
            $item) {
                $row = array();
                $row['kode_produk'] =
                '<span class="label label-
                success">'. $item-
                >produk['kode_produk'] .</span';
                $row['nama_produk'] =
                $item->produk['nama_produk'];
                $row['harga_jual'] =
                'Rp. '. format_uang($item-
                >harga_jual);
                $row['jumlah'] =
                '<input type="number"
                class="form-control input-sm

```

```

                quantity" data-id="'. $item-
                >id_penjualan_detail .' "
                value="'. $item->jumlah .' ">';
                $row['diskon'] =
                $item->diskon . '%';
                $row['subtotal'] =
                'Rp. '. format_uang($item-
                >subtotal);
                $row['aksi'] =
                '<div class="btn-group">

                <button onclick="deleteData(`'.
                route('transaksi.destroy', $item-
                >id_penjualan_detail) .'`)"
                class="btn btn-xs btn-danger btn-
                flat"><i class="fa fa-
                trash"></i></button>

                </div>';

                $data[] = $row;

                $total += $item-
                >harga_jual * $item->jumlah;
                $total_item += $item-
                >jumlah;
            }
            $data[] = [
                'kode_produk' => '
                <div class="total
                hide">'. $total .</div>
                <div
                class="total_item hide">'.
                $total_item .</div>',
                'nama_produk' => '',
                'harga_jual' => '',
                'jumlah' => '',
                'diskon' => '',
                'subtotal' => '',
                'aksi' => '',
            ];

            return datatables()
            ->of($data)
            ->addIndexColumn()
            ->rawColumns(['aksi',
            'kode_produk', 'jumlah'])
            ->make(true);
        }

        public function store(Request
        $request)
        {
            $produk =
            Produk::where('id_produk',
            $request->id_produk)->first();

```

```

        if (! $produk) {
            return response()-
>json('Data gagal disimpan',
400);
        }

        $detail = new
PenjualanDetail();
        $detail->id_penjualan =
$request->id_penjualan;
        $detail->id_produk =
$produk->id_produk;
        $detail->harga_jual =
$produk->harga_jual;
        $detail->jumlah = 1;
        $detail->diskon = 0;
        $detail->subtotal =
$produk->harga_jual;
        $detail->save();

        return response()-
>json('Data berhasil disimpan',
200);
    }

    public function
update(Request $request, $id)
    {
        $detail =
PenjualanDetail::find($id);
        $detail->jumlah =
$request->jumlah;
        $detail->subtotal =
$detail->harga_jual * $request-
>jumlah;
        $detail->update();
    }

    public function destroy($id)
    {
        $detail =
PenjualanDetail::find($id);
        $detail->delete();

        return response(null,
204);
    }

    public function
loadForm($diskon = 0, $total = 0,
$diterima = 0)
    {
        $bayar = $total -
($diskon / 100 * $total);
        $kembali = ($diterima !=

```

```

0) ? $diterima - $bayar : 0;
        $data = [
            'totalrp' =>
format_uang($total),
            'bayar' => $bayar,
            'bayarrp' =>
format_uang($bayar),
            'terbilang' =>
ucwords(terbilang($bayar). '
Rupiah'),
            'kembalirp' =>
format_uang($kembali),
            'kembali_terbilang'
=> ucwords(terbilang($kembali). '
Rupiah'),
        ];

        return response()-
>json($data);
    }
}

/**
 * Remove the specified
resource from storage.
 *
 * @param int $id
 * @return
\Illuminate\Http\Response
 */
    public function destroy($id)
    {
        $produk =
Produk::find($id);
        $produk->delete();

        RiwayatStok::create([
            'body' => 'Produk ' .
$produk->nama_produk . ' Telah
dihapus di etalase'
        ]);

        return response(null,
204);
    }

    public function
deleteSelected(Request $request)
    {
        foreach ($request-
>id_produk as $id) {
            $produk =
Produk::find($id);
            $produk->delete();
        }
    }
}

```

```

    }

    return response(null,
204);
    }

    public function
cetakBarcode(Request $request)
    {
        $dataproduk = array();
        foreach ($request-
>id_produk as $id) {
            $produk =
Produk::find($id);
            $dataproduk[] =
$produk;
        }

        $no = 1;
        $pdf =
PDF::loadView('produk.barcode',
compact('dataproduk', 'no'));
        $pdf->setPaper('a4',
'potrait');
        return $pdf-
>stream('produk.pdf');
    }

    public function riwayatStok()
    {
        $kategori =
Kategori::all()-
>pluck('nama_kategori',
'id_kategori');
        return
view('produk.riwayat_produk',
compact('kategori'));
    }

    public function
riwayatData(Request $request)
    {
        $query =
RiwayatStok::query()-
>orderBy('id','desc');

        $produk = $query->get();

        return datatables()
            ->of($produk)
            ->addIndexColumn()
            -
            >addColumn('select_all', function
($produk) {

```

```

            return '
                <input
                    type="checkbox"
                    name="id_produk[]" value="'.
                    $produk->id_produk .' ">
                ';
            })
            -
            >addColumn('kode_produk',
            function ($produk) {
                return $produk-
            >body;
            })

            ->rawColumns(['aksi',
            'kode_produk', 'select_all'])
            ->make(true);
        }
    }

    <?php

    namespace App\Http\Controllers;

    use App\Models\Penjualan;
    use App\Models\PenjualanDetail;
    use App\Models\Produk;
    use App\Models\RiwayatStok;
    use App\Models\Setting;
    use Illuminate\Http\Request;
    use PDF;

    class PenjualanController extends
Controller
    {
        public function index()
        {
            return
view('penjualan.index');
        }

        public function data()
        {
            $penjualan =
Penjualan::with('member')-
            >orderBy('id_penjualan', 'desc')-
            >get();

            return datatables()
                ->of($penjualan)
                ->addIndexColumn()
                -
                >addColumn('total_item', function
($penjualan) {
                    return

```

```

format_uang($penjualan->total_item);
    })
    -
    >addColumn('total_harga',
function ($penjualan) {
    return 'Rp. '.
format_uang($penjualan->total_harga);
    })
    ->addColumn('bayar',
function ($penjualan) {
    return 'Rp. '.
format_uang($penjualan->bayar);
    })
    -
    >addColumn('tanggal', function
($penjualan) {
    return
tanggal_indonesia($penjualan->created_at, false);
    })
    -
    >addColumn('kode_member',
function ($penjualan) {
    $member =
$penjualan->member->kode_member
?? '';
    return '<span
class="label label-success">'.
$member .'</spa>';
    })
    -
    >editColumn('diskon', function
($penjualan) {
    return
$penjualan->diskon . '%';
    })
    ->editColumn('kasir',
function ($penjualan) {
    return
$penjualan->user->name ?? '';
    })
    ->addColumn('aksi',
function ($penjualan) {
    return '
<div class="btn-
group">
        <button
onclick="showDetail(`'.
route('penjualan.show',
$penjualan->id_penjualan) .` `)"
class="btn btn-xs btn-info btn-
flat"><i class="fa fa-
eye"></i></button>

```

```

        <button
onclick="deleteData(`'.
route('penjualan.destroy',
$penjualan->id_penjualan) .` `)"
class="btn btn-xs btn-danger btn-
flat"><i class="fa fa-
trash"></i></button>
        </div>
    ');
    })
    ->rawColumns(['aksi',
'kode_member'])
    ->make(true);
    }

    public function create()
    {
        $penjualan = new
Penjualan();
        $penjualan->id_member =
null;
        $penjualan->total_item =
0;
        $penjualan->total_harga =
0;
        $penjualan->diskon = 0;
        $penjualan->bayar = 0;
        $penjualan->diterima = 0;
        $penjualan->id_user =
auth()->id();
        $penjualan->save();

        session(['id_penjualan'
=> $penjualan->id_penjualan]);
        return redirect()-
>route('transaksi.index');
    }

    public function store(Request
$request)
    {
        $penjualan =
Penjualan::findOrFail($request->id_penjualan);
        $penjualan->id_member =
$request->id_member;
        $penjualan->total_item =
$request->total_item;
        $penjualan->total_harga =
$request->total;
        $penjualan->diskon =
$request->diskon;
        $penjualan->bayar =
$request->bayar;
        $penjualan->diterima =

```

```

$request->diterima;
    $penjualan->update();

    $detail =
    PenjualanDetail::where('id_penjualan', $penjualan->id_penjualan)->get();
    foreach ($detail as $item) {
        $item->diskon = $request->diskon;
        $item->update();

        $produk =
        Produk::find($item->id_produk);
        $produk->stok -= $item->jumlah;
        $produk->update();

        RiwayatStok::create([
            'body' => 'Produk ' . $produk->nama_produk . ' telah terjual sebanyak ' . $item->jumlah
        ]);
    }

    return redirect()->route('transaksi.selesai');
}

public function show($id)
{
    $detail =
    PenjualanDetail::with('produk')->where('id_penjualan', $id)->get();

    return datatables()
        ->of($detail)
        ->addIndexColumn()
        -
        >addColumn('kode_produk',
        function ($detail) {
            return '<span class="label label-success">'. $detail->produk->kode_produk . '</span>';
        })
        -
        >addColumn('nama_produk',
        function ($detail) {
            return $detail->produk->nama_produk;
        })
        -
        >addColumn('harga_jual', function ($detail) {
            return 'Rp. '. format_uang($detail->harga_jual);
        })
        ->addColumn('jumlah', function ($detail) {
            return format_uang($detail->jumlah);
        })
        -
        >addColumn('subtotal', function ($detail) {
            return 'Rp. '. format_uang($detail->subtotal);
        })
        -
        >rawColumns(['kode_produk'])
        ->make(true);
}

public function destroy($id)
{
    $penjualan =
    Penjualan::find($id);
    $detail =
    PenjualanDetail::where('id_penjualan', $penjualan->id_penjualan)->get();
    foreach ($detail as $item) {
        $produk =
        Produk::find($item->id_produk);
        if ($produk) {
            $produk->stok += $item->jumlah;
            $produk->update();
        }

        $item->delete();
    }

    $penjualan->delete();

    return response(null, 204);
}

public function selesai()
{
    $setting =
    Setting::first();

```

```

        return
    view('penjualan.selesai',
    compact('setting'));
    }

    public function notaKecil()
    {
        $setting =
    Setting::first();
        $penjualan =
    Penjualan::find(session('id_penju
    alan'));
        if (! $penjualan) {
            abort(404);
        }
        $detail =
    PenjualanDetail::with('produk')
        -
    >where('id_penjualan',
    session('id_penjualan'))
        ->get();

        return
    view('penjualan.nota_kecil',
    compact('setting', 'penjualan',
    'detail'));
    }

    public function notaBesar()
    {
        $setting =
    Setting::first();
        $penjualan =
    Penjualan::find(session('id_penju
    alan'));
        if (! $penjualan) {
            abort(404);
        }
        $detail =
    PenjualanDetail::with('produk')
        -
    >where('id_penjualan',
    session('id_penjualan'))
        ->get();

        $pdf =
    PDF::loadView('penjualan.nota_bes
    ar', compact('setting',
    'penjualan', 'detail'));
        $pdf-
    >setPaper(0,0,609,440,
    'potrait');
        return $pdf-
    >stream('Transaksi-'. date('Y-m-

```

```

d-his') .'.pdf');
    }
}

<?php

namespace App\Http\Controllers;

use App\Models\Pembelian;
use App\Models\PembelianDetail;
use App\Models\Produk;
use App\Models\Supplier;
use Illuminate\Http\Request;

class PembelianDetailController
extends Controller
{
    public function index()
    {
        $id_pembelian =
    session('id_pembelian');
        $produk =
    Produk::orderBy('nama_produk')-
    >get();

        $supplier =
    Supplier::find(session('id_suppli
    er'));
        $diskon =
    Pembelian::find($id_pembelian)-
    >diskon ?? 0;

        if (! $supplier) {
            abort(404);
        }

        return
    view('pembelian_detail.index',
    compact('id_pembelian', 'produk',
    'supplier', 'diskon'));
    }

    public function data($id)
    {
        $detail =
    PembelianDetail::with('produk')
        -
    >where('id_pembelian', $id)
        ->get();
        $data = array();
        $total = 0;
        $total_item = 0;

        foreach ($detail as
    $item) {
            $row = array();

```

```

        $row['kode_produk'] =
'<span class="label label-
success">'. $item-
>produk['kode_produk'] . '</span';
        $row['nama_produk'] =
$item->produk['nama_produk'];
        $row['harga_beli'] =
'Rp. '. format_uang($item-
>harga_beli);
        $row['jumlah'] =
'<input type="number"
class="form-control input-sm
quantity" data-id="'. $item-
>id_pembelian_detail .'"
value="'. $item->jumlah .'">';
        $row['subtotal'] =
'Rp. '. format_uang($item-
>subtotal);
        $row['aksi'] =
'<div class="btn-group">

<button onclick="deleteData(`'.
route('pembelian_detail.destroy',
$item->id_pembelian_detail) .`)"
class="btn btn-xs btn-danger btn-
flat"><i class="fa fa-
trash"></i></button>

</div>';

        $data[] = $row;

        $total += $item-
>harga_beli * $item->jumlah;
        $total_item += $item-
>jumlah;
    }
    $data[] = [
        'kode_produk' => '
        <div class="total
hide">'. $total . '</div>
        <div
class="total_item hide">'.
$total_item . '</div>',
        'nama_produk' => '',
        'harga_beli' => '',
        'jumlah' => '',
        'subtotal' => '',
        'aksi' => ''
    ];

    return datatables()
->of($data)
->addIndexColumn()
->rawColumns(['aksi',
'kode_produk', 'jumlah'])

```

```

->make(true);
    }

    public function store(Request
$request)
    {
        $produk =
Produk::where('id_produk',
$request->id_produk)->first();
        if (! $produk) {
            return response()-
>json('Data gagal disimpan',
400);
        }

        $detail = new
PembelianDetail();
        $detail->id_pembelian =
$request->id_pembelian;
        $detail->id_produk =
$produk->id_produk;
        $detail->harga_beli =
$produk->harga_beli;
        $detail->jumlah = 1;
        $detail->subtotal =
$produk->harga_beli;
        $detail->save();

        return response()-
>json('Data berhasil disimpan',
200);
    }

    public function
update(Request $request, $id)
    {
        $detail =
PembelianDetail::find($id);
        $detail->jumlah =
$request->jumlah;
        $detail->subtotal =
$detail->harga_beli * $request-
>jumlah;
        $detail->update();
    }

    public function destroy($id)
    {
        $detail =
PembelianDetail::find($id);
        $detail->delete();

        return response(null,
204);
    }

```

```

        public function
loadForm($diskon, $total)
    {
        $bayar = $total -
($diskon / 100 * $total);
        $data = [
            'totalrp' =>
format_uang($total),
            'bayar' => $bayar,
            'bayarrp' =>
format_uang($bayar),
            'terbilang' =>
ucwords(terbilang($bayar). '
Rupiah')
        ];

        return response()-
>json($data);
    }
}

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Pembelian;
use App\Models\PembelianDetail;
use App\Models\Produk;
use App\Models\RiwayatStok;
use App\Models\Supplier;

class PembelianController extends
Controller
{
    public function index()
    {
        $supplier =
Supplier::orderBy('nama')->get();

        return
view('pembelian.index',
compact('supplier'));
    }

    public function data()
    {
        $pembelian =
Pembelian::orderBy('id_pembelian'
, 'desc')->get();

        return datatables()
->of($pembelian)
->addIndexColumn()

```

```

-
>addColumn('total_item', function
($pembelian) {
    return
format_uang($pembelian->
total_item);
})
-
>addColumn('total_harga',
function ($pembelian) {
    return 'Rp. '.
format_uang($pembelian->
total_harga);
})
->addColumn('bayar',
function ($pembelian) {
    return 'Rp. '.
format_uang($pembelian->bayar);
})
-
>addColumn('tanggal', function
($pembelian) {
    return
tanggal_indonesia($pembelian->
created_at, false);
})
-
>addColumn('supplier', function
($pembelian) {
    return
$pembelian->supplier->nama;
})
-
>editColumn('diskon', function
($pembelian) {
    return
$pembelian->diskon . '%';
})
->addColumn('aksi',
function ($pembelian) {
    return '
<div class="btn-
group">
        <button
onclick="showDetail(`'.
route('pembelian.show',
$pembelian->id_pembelian) .`)"
class="btn btn-xs btn-info btn-
flat"><i class="fa fa-
eye"></i></button>
        <button
onclick="deleteData(`'.
route('pembelian.destroy',
$pembelian->id_pembelian) .`)"
class="btn btn-xs btn-danger btn-

```

```

flat"><i class="fa fa-
trash"></i></button>
        </div>
        ';
    })
    -
>rawColumns(['aksi'])
    ->make(true);
}

public function create($id)
{
    $pembelian = new
Pembelian();
    $pembelian->id_supplier =
$id;
    $pembelian->total_item =
0;
    $pembelian->total_harga =
0;
    $pembelian->diskon =
0;
    $pembelian->bayar =
0;
    $pembelian->save();

    session(['id_pembelian'
=> $pembelian->id_pembelian]);
    session(['id_supplier' =>
$pembelian->id_supplier]);

    return redirect()-
>route('pembelian_detail.index');
}

public function store(Request
$request)
{
    $pembelian =
Pembelian::findOrFail($request-
>id_pembelian);
    $pembelian->total_item =
$request->total_item;
    $pembelian->total_harga =
$request->total;
    $pembelian->diskon =
$request->diskon;
    $pembelian->bayar =
$request->bayar;
    $pembelian->update();

    $detail =
PembelianDetail::where('id_pembel
ian', $pembelian->id_pembelian)-
>get();

```

```

        foreach ($detail as
$item) {
            $produk =
Produk::find($item->id_produk);
            $produk->stok +=
$item->jumlah;
            $produk->update();

            RiwayatStok::create([
                'body' => 'Produk
' . $produk->nama_produk . '
telah ditambahkan ke etalase
sebanyak ' . $item->jumlah . '
buah melalui supplier'
            ]);
        }

        return redirect()-
>route('pembelian.index');
    }

    public function show($id)
    {
        $detail =
PembelianDetail::with('produk')-
>where('id_pembelian', $id)-
>get();

        return datatables()-
->of($detail)
->addIndexColumn()
-
>addColumn('kode_produk',
function ($detail) {
            return '<span
class="label label-success">'.
$detail->produk->kode_produk
.'</span>';
        })
        -
>addColumn('nama_produk',
function ($detail) {
            return $detail-
>produk->nama_produk;
        })
        -
>addColumn('harga_beli', function
($detail) {
            return 'Rp. '.
format_uang($detail->harga_beli);
        })
        ->addColumn('jumlah',
function ($detail) {

```

```

        return
        format_uang($detail->jumlah);
    })
    -
    >addColumn('subtotal', function
    ($detail) {
        return 'Rp. '.
        format_uang($detail->subtotal);
    })
    -
    >rawColumns(['kode_produk'])
    ->make(true);
    }

    public function destroy($id)
    {
        $pembelian =
        Pembelian::find($id);
        $detail =
        PembelianDetail::where('id_pembel
        ian', $pembelian->id_pembelian)-
        >get();
        foreach ($detail as
        $item) {
            $produk =
            Produk::find($item->id_produk);
            if ($produk) {
                $produk->stok -=
                $item->jumlah;
                $produk-
                >update();
            }
            $item->delete();
        }

        $pembelian->delete();

        return response(null,
        204);
    }
}

<?php

namespace App\Http\Controllers;

use App\Models\Member;
use App\Models\Setting;
use Illuminate\Http\Request;
use PDF;

class MemberController extends
Controller
{
    /**

```

```

        * Display a listing of the
        resource.
        *
        * @return
        \Illuminate\Http\Response
        */
        public function index()
        {
            return
            view('member.index');
        }

        public function data()
        {
            $member =
            Member::orderBy('kode_member')-
            >get();

            return datatables()
            ->of($member)
            ->addIndexColumn()
            -
            >addColumn('select_all', function
            ($produk) {
                return '
                <input
                type="checkbox"
                name="id_member[]" value="'.
                $produk->id_member .' ">
                ';
            })
            -
            >addColumn('kode_member',
            function ($member) {
                return '<span
                class="label label-success">'.
                $member->kode_member .'<span>';
            })
            ->addColumn('aksi',
            function ($member) {
                return '
                <div class="btn-
                group">

                <button
                type="button"
                onclick="editForm(`'.
                route('member.update', $member-
                >id_member) .'`)" class="btn btn-
                xs btn-info btn-flat"><i
                class="fa fa-
                pencil"></i></button>

                <button
                type="button"
                onclick="deleteData(`'.
                route('member.destroy', $member-

```

```

>id_member) .'`)" class="btn btn-
xs btn-danger btn-flat"><i
class="fa fa-trash"></i></button>
</div>
';
    })
    ->rawColumns(['aksi',
'select_all', 'kode_member'])
    ->make(true);
}

/**
 * Show the form for creating
a new resource.
 *
 * @return
\Illuminate\Http\Response
 */
public function create()
{
    //
}

/**
 * Store a newly created
resource in storage.
 *
 * @param
\Illuminate\Http\Request
$request
 * @return
\Illuminate\Http\Response
 */
public function store(Request
$request)
{
    $member =
Member::latest()->first() ?? new
Member();
    $kode_member = (int)
$member->kode_member +1;

    $member = new Member();
    $member->kode_member =
tambah_nol_didepan($kode_member,
5);
    $member->nama = $request-
>nama;
    $member->telepon =
$request->telepon;
    $member->alamat =
$request->alamat;
    $member->save();

    return response()-

```

```

>json('Data berhasil disimpan',
200);
}

/**
 * Display the specified
resource.
 *
 * @param int $id
 * @return
\Illuminate\Http\Response
 */
public function show($id)
{
    $member =
Member::find($id);

    return response()-
>json($member);
}

/**
 * Show the form for editing
the specified resource.
 *
 * @param int $id
 * @return
\Illuminate\Http\Response
 */
public function edit($id)
{
    //
}

/**
 * Update the specified
resource in storage.
 *
 * @param
\Illuminate\Http\Request
$request
 * @param int $id
 * @return
\Illuminate\Http\Response
 */
public function
update(Request $request, $id)
{
    $member =
Member::find($id)-
>update($request->all());

    return response()-
>json('Data berhasil disimpan',
200);
}

```

```

    }

    /**
     * Remove the specified
resource from storage.
     *
     * @param int $id
     * @return
\Illuminate\Http\Response
     */
    public function destroy($id)
    {
        $member =
Member::find($id);
        $member->delete();

        return response(null,
204);
    }

    public function
cetakMember(Request $request)
    {
        $datamember =
collect(array());
        foreach ($request-
>id_member as $id) {
            $member =
Member::find($id);
            $datamember[] =
$member;
        }

        $datamember =
$datamember->chunk(2);
        $setting =
Setting::first();

        $no = 1;
        $pdf =
PDF::loadView('member.cetak',
compact('datamember', 'no',
'setting'));
        $pdf->setPaper(array(0,
0, 566.93, 850.39), 'potrait');
        return $pdf-
>stream('member.pdf');
    }
}

<?php

namespace App\Http\Controllers;

use App\Models\Kategori;

```

```

use App\Models\Member;
use App\Models\Pembelian;
use App\Models\Pengeluaran;
use App\Models\Penjualan;
use App\Models\Produk;
use App\Models\Supplier;
use Illuminate\Http\Request;

class DashboardController extends
Controller
{
    public function index()
    {
        $kategori =
Kategori::count();
        $produk =
Produk::count();
        $supplier =
Supplier::count();
        $member =
Member::count();

        $tanggal_awal = date('Y-
m-01');
        $tanggal_akhir = date('Y-
m-d');

        $data_tanggal = array();
        $data_pendapatan =
array();

        while
(strtotime($tanggal_awal) <=
strtotime($tanggal_akhir)) {
            $data_tanggal[] =
(int) substr($tanggal_awal, 8,
2);

            $total_penjualan =
Penjualan::where('created_at',
'LIKE', "%$tanggal_awal%")-
>sum('bayar');
            $total_pembelian =
Pembelian::where('created_at',
'LIKE', "%$tanggal_awal%")-
>sum('bayar');
            $total_pengeluaran =
Pengeluaran::where('created_at',
'LIKE', "%$tanggal_awal%")-
>sum('nominal');

            $pendapatan =
$total_penjualan -
$total_pembelian -
$total_pengeluaran;

```

```

                $data_pendapatan[] +=
$pendapatan;

                $tanggal_awal =
date('Y-m-d', strtotime("+1 day",
strtotime($tanggal_awal)));
            }

            if (auth()->user()->level
== 1) {
                return
view('admin.dashboard',
compact('kategori', 'produk',
'supplier', 'member',
'tanggal_awal', 'tanggal_akhir',
'data_tanggal',
'data_pendapatan'));
            } else {
                return
view('kasir.dashboard');
            }
        }
    }
}

```

<?php

```

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Kategori;

class KategoriController extends
Controller
{
    /**
     * Display a listing of the
resource.
     *
     * @return
\Illuminate\Http\Response
     */
    public function index()
    {
        return
view('kategori.index');
    }

    public function data()
    {
        $kategori =
Kategori::orderBy('id_kategori',
'desc')->get();

        return datatables()
->of($kategori)

```

```

->addIndexColumn()
->addColumn('aksi',
function ($kategori) {
    return '
<div class="btn-
group">
        <button
onclick="editForm(`'.
route('kategori.update',
$kategori->id_kategori) .`)"
class="btn btn-xs btn-info btn-
flat"><i class="fa fa-
pencil"></i></button>
        <button
onclick="deleteData(`'.
route('kategori.destroy',
$kategori->id_kategori) .`)"
class="btn btn-xs btn-danger btn-
flat"><i class="fa fa-
trash"></i></button>
    </div>
';
    })
-

```

>rawColumns(['aksi'])

```

->make(true);
    }

    /**
     * Show the form for creating
a new resource.
     *
     * @return
\Illuminate\Http\Response
     */
    public function create()
    {
        //
    }

    /**
     * Store a newly created
resource in storage.
     *
     * @param
\Illuminate\Http\Request
$request
     * @return
\Illuminate\Http\Response
     */
    public function store(Request
$request)
    {
        $kategori = new
Kategori();

```

```

        $kategori->nama_kategori
= $request->nama_kategori;
        $kategori->save();

        return response()-
>json('Data berhasil disimpan',
200);
    }

    /**
     * Display the specified
resource.
     *
     * @param int $id
     * @return
\Illuminate\Http\Response
     */
    public function show($id)
    {
        $kategori =
Kategori::find($id);

        return response()-
>json($kategori);
    }

    /**
     * Show the form for editing
the specified resource.
     *
     * @param int $id
     * @return
\Illuminate\Http\Response
     */
    public function edit($id)
    {
        //
    }

    /**
     * Update the specified
resource in storage.
     *
     * @param
\Illuminate\Http\Request
$request
     * @param int $id
     * @return
\Illuminate\Http\Response
     */
    public function
update(Request $request, $id)
    {
        $kategori =
Kategori::find($id);

```

```

        $kategori->nama_kategori
= $request->nama_kategori;
        $kategori->update();

        return response()-
>json('Data berhasil disimpan',
200);
    }

    /**
     * Remove the specified
resource from storage.
     *
     * @param int $id
     * @return
\Illuminate\Http\Response
     */
    public function destroy($id)
    {
        $kategori =
Kategori::find($id);
        $kategori->delete();

        return response(null,
204);
    }
}

<?php

namespace App\Http\Controllers;

use App\Models\Pembelian;
use App\Models\Pengeluaran;
use App\Models\Penjualan;
use Illuminate\Http\Request;
use PDF;

class LaporanController extends
Controller
{
    public function index(Request
$request)
    {
        $tanggalAwal = date('Y-m-
d', mktime(0, 0, 0, date('m'), 1,
date('Y')));
        $tanggalAkhir = date('Y-
m-d');

        if ($request-
>has('tanggal_awal') && $request-
>tanggal_awal != "" && $request-
>has('tanggal_akhir') &&
$request->tanggal_akhir) {

```

```

        $tanggalAwal =
$request->tanggal_awal;
        $tanggalAkhir =
$request->tanggal_akhir;
    }

    return
view('laporan.index',
compact('tanggalAwal',
'tanggalAkhir'));
}

    public function
getData($awal, $akhir)
    {
        $no = 1;
        $data = array();
        $pendapatan = 0;
        $total_pendapatan = 0;

        while (strtotime($awal)
<= strtotime($akhir)) {
            $tanggal = $awal;
            $awal = date('Y-m-d',
strtotime("+1 day",
strtotime($awal)));

            $total_penjualan =
Penjualan::where('created_at',
'LIKE', "%$tanggal%")-
>sum('bayar');
            $total_pembelian =
Pembelian::where('created_at',
'LIKE', "%$tanggal%")-
>sum('bayar');
            $total_pengeluaran =
Pengeluaran::where('created_at',
'LIKE', "%$tanggal%")-
>sum('nominal');

            $pendapatan =
$total_penjualan -
$total_pembelian -
$total_pengeluaran;
            $total_pendapatan +=
$pendapatan;

            $row = array();
            $row['DT_RowIndex'] =
$no++;
            $row['tanggal'] =
tanggal_indonesia($tanggal,
false);
            $row['penjualan'] =
format_uang($total_penjualan);

```

```

            $row['pembelian'] =
format_uang($total_pembelian);
            $row['pengeluaran'] =
format_uang($total_pengeluaran);
            $row['pendapatan'] =
format_uang($pendapatan);

            $data[] = $row;
        }

        $data[] = [
            'DT_RowIndex' => '',
            'tanggal' => '',
            'penjualan' => '',
            'pembelian' => '',
            'pengeluaran' =>
'Total Pendapatan',
            'pendapatan' =>
format_uang($total_pendapatan),
        ];

        return $data;
    }

    public function data($awal,
$akhir)
    {
        $data = $this-
>getData($awal, $akhir);

        return datatables()
->of($data)
->make(true);
    }

    public function
exportPDF($awal, $akhir)
    {
        $data = $this-
>getData($awal, $akhir);
        $pdf =
PDF::loadView('laporan.pdf',
compact('awal', 'akhir',
'data'));
        $pdf->setPaper('a4',
'potrait');

        return $pdf-
>stream('Laporan-pendapatan-' .
date('Y-m-d-his') . '.pdf');
    }
}

<?php

```

```

namespace App\Http\Controllers;

use
Illuminate\Foundation\Auth\Access\
    AuthorizesRequests;
use
Illuminate\Foundation\Bus\DispatchesJobs;
use
Illuminate\Foundation\Validation\
    ValidatesRequests;
use Illuminate\Routing\Controller
    as BaseController;

class Controller extends
    BaseController
{
    use AuthorizesRequests,
        DispatchesJobs,
        ValidatesRequests;
}

@extends('layouts.master')

@section('title')
    Edit Profil
@endsection

@section('breadcrumb')
    @parent
    <li class="active">Edit
        Profil</li>
@endsection

@section('content')
<div class="row">
    <div class="col-lg-12">
        <div class="box">
            <form action="{{
                route('user.update_profil') }}"
                method="post" class="form-profil"
                data-toggle="validator"
                enctype="multipart/form-data">
                @csrf
                <div class="box-
                    body">
                        <div
                            class="alert alert-info alert-
                                dismissible" style="display:
                                    none;">
                                <button
                                    type="button" class="close" data-
                                        dismiss="alert" aria-
                                            hidden="true">&times;</button>
                                <i

```

```

class="icon fa fa-check"></i>
Perubahan berhasil disimpan
        </div>
    </div>
    <div
        class="form-group row">
            <label
                for="name" class="col-lg-2
                    control-label">Nama</label>
            <div
                class="col-lg-6">
                <input type="text" name="name"
                    class="form-control" id="name"
                    required autofocus value="{{
                        $profil->name }}">
            <span
                class="help-block with-
                    errors"></span>
            </div>
        </div>
    <div
        class="form-group row">
            <label
                for="foto" class="col-lg-2
                    control-label">Profil</label>
            <div
                class="col-lg-4">
                <input type="file" name="foto"
                    class="form-control" id="foto"

                onchange="preview('.tampil-foto',
                    this.files[0])">
            <span
                class="help-block with-
                    errors"></span>
            <br>
            <div
                class="tampil-foto">

                

            </div>
        </div>
    </div>
    <div
        class="form-group row">
            <label
                for="old_password" class="col-lg-
                    2 control-label">Password
                Lama</label>
            <div
                class="col-lg-6">

```

```

<input type="password"
name="old_password"
id="old_password" class="form-
control"

minlength="6">

<span
class="help-block with-
errors"></span>

</div>
</div>
<div
class="form-group row">
<label
for="password" class="col-lg-2
control-label">Password</label>
<div
class="col-lg-6">

<input type="password"
name="password" id="password"
class="form-control"

minlength="6">

<span

class="help-block with-
errors"></span>

</div>
</div>
<div
class="form-group row">
<label
for="password_confirmation"
class="col-lg-2 control-
label">Konfirmasi
Password</label>
<div
class="col-lg-6">

<input type="password"
name="password_confirmation"
id="password_confirmation"
class="form-control"

data-match="#password">

<span

class="help-block with-
errors"></span>

</div>
</div>
</div>
<div class="box-
footer text-right">

<button
class="btn btn-sm btn-flat btn-

```

```

primary"><i class="fa fa-
save"></i> Simpan
Perubahan</button>

</div>
</form>
</div>
</div>
<endsection

@push('scripts')
<script>
$(function () {

$('#old_password').on('keyup',
function () {
if ($(this).val() !=
"") $('#password,
#password_confirmation').attr('re
quired', true);
else $('#password,
#password_confirmation').attr('re
quired', false);
});

$('.form-
profil').validator().on('submit',
function (e) {
if (!
e.preventDefault()) {
$.ajax({
url:
$('#form-profil').attr('action'),
type:
$('#form-profil').attr('method'),
data: new
FormData($('#form-profil')[0]),
async: false,
processData:
false,
contentType:
false

})
.done(response =>
{
$('#[name=name]').val(response.nam
e);

$('.tampil-
foto').html('');

$('.img-
profil').attr('src', `{{ url('/')
}}/${response.foto}`);

```

```

$( '.alert' ).fadeIn();
                                setTimeout( ()
=> {
                                }, 3000);
                                })
                                .fail( errors => {
                                if
                                ( errors.status == 422 ) {

                                alert( errors.responseJSON );
                                } else {

                                alert( 'Tidak dapat menyimpan
                                data' );

                                }
                                return;
                                });
                                }
                                });
                                });
</script>
@endpush

@extends( 'layouts.master' )

@section( 'title' )
    Daftar User
@endsection

@section( 'breadcrumb' )
    @parent
    <li class="active">Daftar
    User</li>
@endsection

@section( 'content' )
<div class="row">
    <div class="col-lg-12">
        <div class="box">
            <div class="box-
            header with-border">
                <button
                onclick="addForm( '{{
                route( 'user.store' ) }}' )"
                class="btn btn-success btn-xs
                btn-flat"><i class="fa fa-plus-
                circle"></i> Tambah</button>
            </div>
            <div class="box-body
            table-responsive">
                <table
                class="table table-stiped table-
                bordered">
                    <thead>
                        <th
                        width="5%">No</th>
                        <th>Nama</th>
                        <th>Email</th>
                    </thead>
                    <tbody>
                        <tr>
                            <th
                            width="15%"><i class="fa fa-
                            cog"></i></th>
                        </tr>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</div>

@includeIf( 'user.form' )
@endsection

@push( 'scripts' )
<script>
    let table;

    $( function () {
        table =
        $( '.table' ).DataTable( {
            processing: true,
            autoWidth: false,
            ajax: {
                url: '{{
                route( 'user.data' ) }}',
            },
            columns: [
                { data:
                'DT_RowIndex', searchable: false,
                sortable: false },
                { data: 'name',
                searchable: false, sortable:
                false },
                { data: 'email',
                searchable: false, sortable:
                false },
                { data: 'aksi',
                searchable: false, sortable:
                false }
            ]
        } );

        $( '#modal-
        form' ).validator().on( 'submit',
        function ( e ) {
            if ( !
            e.preventDefault() ) {
                $.post( $( '#modal-
                form form' ).attr( 'action' ),
                $( '#modal-form

```

```

form').serialize())

.done((response) => {

$('#modal-form').modal('hide');

table.ajax.reload();
        })

.fail((errors) => {

alert('Tidak dapat menyimpan
data');

                                return;
                                });
        });

function addForm(url) {
    $('#modal-
form').modal('show');
    $('#modal-form .modal-
title').text('Tambah User');

    $('#modal-form
form')[0].reset();
    $('#modal-form
form').attr('action', url);
    $('#modal-form
[name=_method]').val('post');
    $('#modal-form
[name=name]').focus();

    $('#password,
#password_confirmation').attr('re
quired', true);
    }

function editForm(url) {
    $('#modal-
form').modal('show');
    $('#modal-form .modal-
title').text('Edit User');

    $('#modal-form
form')[0].reset();
    $('#modal-form
form').attr('action', url);
    $('#modal-form
[name=_method]').val('put');
    $('#modal-form
[name=name]').focus();

    $('#password,

```

```

#password_confirmation').attr('re
quired', false);

        $.get(url)
            .done((response) => {
                $('#modal-form
[name=name]').val(response.name);
                $('#modal-form
[name=email]').val(response.email
);
            })
            .fail((errors) => {
                alert('Tidak
dapat menampilkan data');
                return;
            });
    }

function deleteData(url) {
    if (confirm('Yakin ingin
menghapus data terpilih?')) {
        $.post(url, {
            '_token':
$('#[name=csrf-
token]').attr('content'),
            '_method':
'delete'
        })
        .done((response)
=> {
            table.ajax.reload();
        })
        .fail((errors) =>
{
            alert('Tidak
dapat menghapus data');
            return;
        });
    }
}
</script>
@endpush

<div class="modal fade"
id="modal-form" tabindex="-1"
role="dialog" aria-
labelledby="modal-form">
    <div class="modal-dialog
modal-lg" role="document">
        <form action=""
method="post" class="form-
horizontal">
            @csrf
            @method('post')

```

```

        <div class="modal-
content">
            <div
class="modal-header">
                <button
type="button" class="close" data-
dismiss="modal" aria-
label="Close"><span
                                aria-
hidden="true">&times;</span></but
ton>
                <h4
class="modal-title"></h4>
            </div>
            <div
class="modal-body">
                <div
class="form-group row">
                    <label
for="name" class="col-lg-3 col-
lg-offset-1 control-
label">Nama</label>
                    <div
class="col-lg-6">
                        <input type="text" name="name"
id="name" class="form-control"
required autofocus>
                    <span
class="help-block with-
errors"></span>
                </div>
            </div>
            <div
class="form-group row">
                <label
for="email" class="col-lg-3 col-
lg-offset-1 control-
label">Email</label>
                <div
class="col-lg-6">
                    <input type="email" name="email"
id="email" class="form-control"
required>
                <span
class="help-block with-
errors"></span>
            </div>
        </div>
        <div
class="form-group row">
            <label
for="password" class="col-lg-3

```

```

col-lg-offset-1 control-
label">Password</label>
            <div
class="col-lg-6">
                <input type="password"
name="password" id="password"
class="form-control"
required
minlength="6">
            <span
class="help-block with-
errors"></span>
        </div>
    </div>
    <div
class="form-group row">
        <label
for="password_confirmation"
class="col-lg-3 col-lg-offset-1
control-label">Konfirmasi
Password</label>
        <div
class="col-lg-6">
            <input type="password"
name="password_confirmation"
id="password_confirmation"
class="form-control"
required
data-match="#password">
        <span
class="help-block with-
errors"></span>
    </div>
</div>
<div
class="modal-footer">
    <button
class="btn btn-sm btn-flat btn-
primary"><i class="fa fa-
save"></i> Simpan</button>
    <button
type="button" class="btn btn-sm
btn-flat btn-warning" data-
dismiss="modal"><i class="fa fa-
arrow-circle-left"></i>
Batal</button>
</div>
</div>

```

```

        </form>
    </div>
</div>

@extends('layouts.master')

@section('title')
    @if (request()->is('produk-
kritis'))
        Produk Kritis
    @php
        $status = 'kritis'
    @endphp
    @else
    @php
        $status = 'normal'
    @endphp
        Daftar Produk
    @endif
@endsection

@section('breadcrumb')
    @parent
    <li class="active">Daftar
Produk</li>
@endsection

@section('content')
<div class="row">
    <div class="col-lg-12">
        <div class="box">
            <div class="box-
header with-border">
                <div class="btn-
group">
                    <button
onclick="addForm('{{
route('produk.store') }}')"
class="btn btn-success btn-xs
btn-flat"><i class="fa fa-plus-
circle"></i> Tambah</button>
                    <button
onclick="deleteSelected('{{
route('produk.delete_selected')
}}')" class="btn btn-danger btn-
xs btn-flat"><i class="fa fa-
trash"></i> Hapus</button>
                    <button
onclick="cetakBarcode('{{
route('produk.cetak_barcode')
}}')" class="btn btn-info btn-xs
btn-flat"><i class="fa fa-
barcode"></i> Cetak
Barcode</button>
                </div>
            </div>
        </div>
    </div>
</div>

```

```

    </div>
    <div class="box-body
table-responsive">
        <form action=""
method="post" class="form-
produk">
            @csrf
            <table
class="table table-stiped table-
bordered">
                <thead>
                    <th
width="5%">
                        <input type="checkbox"
name="select_all"
id="select_all">
                    </th>
                    <th
width="5%">No</th>
                    <th>Kode</th>
                    <th>Nama</th>
                    <th>Kategori</th>
                    <th>Merk</th>
                    <th>Harga Beli</th>
                    <th>Harga Jual</th>
                    <th>Diskon</th>
                    <th>Stok</th>
                <th
width="15%"><i class="fa fa-
cog"></i></th>
            </thead>
        </table>
    </form>
    </div>
</div>
</div>

@includeIf('produk.form')
@endsection

@push('scripts')
<script>
    let table;

    $(function () {

```

```

        table =
$($('.table').DataTable({
    processing: true,
    autoWidth: false,
    ajax: {
        url: '{{
route('produk.data') }}',
        data:{status: '{{
$status }}'}}
    },
    columns: [
        {data:
'select_all', searchable: false,
sortable: false},
        {data:
'DT_RowIndex', searchable: false,
sortable: false},
        {data:
'kode_produk'},
        {data:
'nama_produk'},
        {data:
'nama_kategori'},
        {data: 'merk'},
        {data:
'harga_beli'},
        {data:
'harga_jual'},
        {data: 'diskon'},
        {data: 'stok'},
        {data: 'aksi'},
searchable: false, sortable:
false},
    ]
});

$('#modal-
form').validator().on('submit',
function (e) {
    if (!
e.preventDefault()) {
        $.post($('#modal-
form form').attr('action'),
$('#modal-form
form').serialize())

.done((response) => {
$('#modal-form').modal('hide');

table.ajax.reload();
})

.fail((errors) => {

```

```

alert('Tidak dapat menyimpan
data');
return;
});
});

$('[name=select_all]').on('click'
, function () {
$(':checkbox').prop('checked',
this.checked);
});

function addForm(url) {
    $('#modal-
form').modal('show');
    $('#modal-form .modal-
title').text('Tambah Produk');

    $('#modal-form
form')[0].reset();
    $('#modal-form
form').attr('action', url);
    $('#modal-form
[name=_method]').val('post');
    $('#modal-form
[name=nama_produk]').focus();
}

function editForm(url) {
    $('#modal-
form').modal('show');
    $('#modal-form .modal-
title').text('Edit Produk');

    $('#modal-form
form')[0].reset();
    $('#modal-form
form').attr('action', url);
    $('#modal-form
[name=_method]').val('put');
    $('#modal-form
[name=nama_produk]').focus();

    $.get(url)
        .done((response) => {
            $('#modal-form
[name=nama_produk]').val(response
.nama_produk);

            $('#modal-form
[name=id_kategori]').val(response
.id_kategori);

```

```

                $('#modal-form
[name=merk]').val(response.merk);
                $('#modal-form
[name=harga_beli]').val(response.
harga_beli);
                $('#modal-form
[name=harga_jual]').val(response.
harga_jual);
                $('#modal-form
[name=diskon]').val(response.disk
on);
                $('#modal-form
[name=stok]').val(response.stok);
            })
            .fail((errors) => {
                alert('Tidak
dapat menampilkan data');
                return;
            });
        }

        function deleteData(url) {
            if (confirm('Yakin ingin
menghapus data terpilih?')) {
                $.post(url, {
                    '_token':
$('[name=csrf-
token]').attr('content'),
                    '_method':
'delete'
                })
                .done((response)
=> {
                    table.ajax.reload();
                })
                .fail((errors) =>
{
                    alert('Tidak
dapat menghapus data');
                    return;
                });
            }

            function deleteSelected(url)
{
                if
($('input:checked').length > 1) {
                    if (confirm('Yakin
ingin menghapus data terpilih?'))
{
                        $.post(url,
$('.form-produk').serialize())

```

```

                .done((response) => {
                    table.ajax.reload();
                })

                .fail((errors) => {
                    alert('Tidak dapat menghapus
data');
                    return;
                });
            }
        } else {
            alert('Pilih data
yang akan dihapus');
            return;
        }

        function cetakBarcode(url) {
            if
($('input:checked').length < 1) {
                alert('Pilih data
yang akan dicetak');
                return;
            } else if
($('input:checked').length < 3) {
                alert('Pilih minimal
3 data untuk dicetak');
                return;
            } else {
                $('.form-produk')
                .attr('target',
'_blank')
                .attr('action',
url)
                .submit();
            }
        }
    </script>
    @endpush

    <div class="modal fade"
id="modal-form" tabindex="-1"
role="dialog" aria-
labelledby="modal-form">
        <div class="modal-dialog
modal-lg" role="document">
            <form action=""
method="post" class="form-
horizontal">
                @csrf
                @method('post')

                <div class="modal-

```

```

content">
    <div
class="modal-header">
        <button
type="button" class="close" data-
dismiss="modal" aria-
label="Close"><span
                                aria-
hidden="true">&times;</span></but
ton>
        <h4
class="modal-title"></h4>
    </div>
    <div
class="modal-body">
        <div
class="form-group row">
            <label
for="deskripsi" class="col-lg-2
col-lg-offset-1 control-
label">Deskripsi</label>
            <div
class="col-lg-6">
                <input type="text"
name="deskripsi" id="deskripsi"
class="form-control" required
autofocus>
                <span
class="help-block with-
errors"></span>
            </div>
        </div>
        <div
class="form-group row">
            <label
for="nominal" class="col-lg-2
col-lg-offset-1 control-
label">Nominal</label>
            <div
class="col-lg-6">
                <input type="number"
name="nominal" id="nominal"
class="form-control" required>
                <span
class="help-block with-
errors"></span>
            </div>
        </div>
    </div>
    <div
class="modal-footer">
        <button
class="btn btn-sm btn-flat btn-

```

```

primary"><i class="fa fa-
save"></i> Simpan</button>
        <button
type="button" class="btn btn-sm
btn-flat btn-warning" data-
dismiss="modal"><i class="fa fa-
arrow-circle-left"></i>
Batal</button>
    </div>
</div>
</form>
</div>
</div>
<div class="modal fade"
id="modal-supplier" tabindex="-1"
role="dialog" aria-
labelledby="modal-supplier">
    <div class="modal-dialog
modal-lg" role="document">
        <div class="modal-
content">
            <div class="modal-
header">
                <button
type="button" class="close" data-
dismiss="modal" aria-
label="Close"><span
                                aria-
hidden="true">&times;</span></but
ton>
                <h4 class="modal-
title">Pilih Supplier</h4>
            </div>
            <div class="modal-
body">
                <table
class="table table-striped table-
bordered table-supplier">
                    <thead>
                        <th
width="5%">No</th>
                        <th>Nama</th>
                        <th>Telepon</th>
                        <th>Alamat/Wilayah</th>
                    </thead>
                    <tbody>
                        <@foreach
($supplier as $key => $item)
                        <tr>

```

```

<td width="5%">{{ $key+1 }}</td>

<td>{{ $item->nama }}</td>

<td>{{ $item->telepon }}</td>

<td>{{ $item->alamat }}</td>

<td>

<a href="{{
route('pembelian.create', $item-
>id_supplier) }}" class="btn btn-
primary btn-xs btn-flat">

<i class="fa fa-check-
circle"></i>

Pilih

</a>

</td>

</tr>

@endforeach

</tbody>
</table>
</div>
</div>
</div>
</div>

<div class="modal fade"
id="modal-detail" tabindex="-1"
role="dialog" aria-
labelledby="modal-detail">
  <div class="modal-dialog
modal-lg" role="document">
    <div class="modal-
content">
      <div class="modal-
header">
        <button
type="button" class="close" data-
dismiss="modal" aria-
label="Close"><span
aria-
hidden="true">&times;</span></but-
ton>
        <h4 class="modal-
title">Detail Pembelian</h4>
      </div>
      <div class="modal-

```

```

body">
      <table
class="table table-striped table-
bordered table-detail">
        <thead>
          <th
width="5%">No</th>

          <th>Kode</th>

          <th>Nama</th>

          <th>Harga</th>

          <th>Jumlah</th>

          <th>Subtotal</th>
        </thead>
      </table>
    </div>
  </div>
</div>

@extends('layouts.master')

@section('title')
  Laporan Pendapatan {{
tanggal_indonesia($tanggalAwal,
false) }} s/d {{
tanggal_indonesia($tanggalAkhir,
false) }}
@endsection

@push('css')
  <link rel="stylesheet" href="{{
asset('/AdminLTE-
2/bower_components/bootstrap-
datepicker/dist/css/bootstrap-
datepicker.min.css') }}">
@endpush

@section('breadcrumb')
  @parent
  <li
class="active">Laporan</li>
@endsection

@section('content')
  <div class="row">
    <div class="col-lg-12">
      <div class="box">
        <div class="box-
header with-border">
          <button

```

```

onclick="updatePeriode()"
class="btn btn-info btn-xs btn-
flat"><i class="fa fa-plus-
circle"></i> Ubah
Periode</button>
<a href="{{
route('laporan.export_pdf',
[$tanggalAwal, $tanggalAkhir])
}}" target="_blank" class="btn
btn-success btn-xs btn-flat"><i
class="fa fa-file-excel-o"></i>
Export PDF</a>
</div>
<div class="box-body
table-responsive">
<table
class="table table-stiped table-
bordered">
<thead>
<th
width="5%">No</th>
<th>Tanggal</th>
<th>Penjualan</th>
<th>Pembelian</th>
<th>Pengeluaran</th>
<th>Pendapatan</th>
</thead>
</table>
</div>
</div>
</div>
</div>
@includeIf('laporan.form')
@endsection

@push('scripts')
<script src="{{ asset('/AdminLTE-
2/bower_components/bootstrap-
datepicker/dist/js/bootstrap-
datepicker.min.js') }}"></script>
<script>
let table;

$(function () {
table =
$('.table').DataTable({
processing: true,
autoWidth: false,
ajax: {

```

```

url: '{{
route('laporan.data',
[$tanggalAwal, $tanggalAkhir])
}}',
},
columns: [
{data:
'DT_RowIndex', searchable: false,
sortable: false},
{data:
'tanggal'},
{data:
'penjualan'},
{data:
'pembelian'},
{data:
'pengeluaran'},
{data:
'pendapatan'}
],
dom: 'Brt',
bSort: false,
bPaginate: false,
});

$('.datepicker').datepicker({
format: 'yyyy-mm-dd',
autoclose: true
});

function updatePeriode() {
$('#modal-
form').modal('show');
}
</script>
@endpush

@extends('layouts.auth')

@section('login')
<div class="login-box">
<!-- /.login-logo -->
<div class="login-box-body">
<div class="login-logo">
<a href="{{ url('/')
}}">

</a>
</div>

```

```

        <form action="{{
route('login') }}" method="post"
class="form-login">
        @csrf
        <div class="form-
group has-feedback
@error('email') has-error
@enderror">
                <input
type="email" name="email"
class="form-control"
placeholder="Email" required
value="{{ old('email') }}"
autofocus>
                <span
class="glyphicon glyphicon-
envelope form-control-
feedback"></span>
                @error('email')
                <span
class="help-block">{{ $message
}}</span>
                @else
                <span
class="help-block with-
errors"></span>
                @enderror
        </div>
        <div class="form-
group has-feedback
@error('password') has-error
@enderror">
                <input
type="password" name="password"
class="form-control"
placeholder="Password" required>
                <span
class="glyphicon glyphicon-lock
form-control-feedback"></span>
                @error('password')
                <span
class="help-block">{{ $message
}}</span>
                @else
                <span
class="help-block with-
errors"></span>
                @enderror
        </div>
        <div class="row">
                <div class="col-
xs-8">
                        <div
class="checkbox icheck">

```

```

                                <label>
                                <input type="checkbox"> Remember
                                Me
                                </label>
                                </div>
                                </div>
                                <!-- /.col -->
                                <div class="col-
xs-4">
                                        <button
type="submit" class="btn btn-
primary btn-block btn-flat">Sign
In</button>
                                        </div>
                                        <!-- /.col -->
                                </div>
                                </form>
                                </div>
                                <!-- /.login-box-body -->
                                </div>
                                <!-- /.login-box -->
                                @endsection

```