

<pre> <?php namespace App; use Illuminate\Database\Eloquent\Model; use Illuminate\Support\Facades\DB; class Barang extends Model { protected \$table = 'barang'; public function getBarangTerjual() { \$data = \$this->transaksiBarang; \$jumlahBarang = 0; foreach(\$data as \$index => \$item) { \$jumlahBarang = \$jumlahBarang + (int) \$item->jumlah_barang; } return \$jumlahBarang; } public function transaksiBarang() { return \$this- >hasMany('App\Penjualan','idbarang'); } public function scopeGetIdsupplier(\$query, \$id) { return DB::table(\$this->table) ->where('id', \$id) ->value('idsupplier'); } </pre>	<pre> } public function scopeGetTotalByEtalase(\$query, \$idetalase) { return DB::table(\$this->table) ->where('idetalase', \$idetalase) ->count('id'); } public function scopeGetTotalByKategori(\$query, \$idetalase) { return DB::table(\$this->table) ->where('idetalase', \$idetalase) ->count('id'); } public function scopeGetTotal(\$query) { return DB::table(\$this->table) ->count('id'); } public function scopeGetBiayaPemesanan(\$query, \$id) { return DB::table(\$this->table) ->where('id', \$id) ->value('biaya_pemesanan'); } </pre>
---	--

```

public function
scopeGetHargaBarang($query, $id)
{
    return DB::table($this->table)
        ->where('id', $id)
        ->value('harga_barang');
}

```

```

public function
scopeGetBiayaPenyimpanan($query, $id)
{
    return DB::table($this->table)
        ->where('id', $id)
        ->value('biaya_penyimpanan');
}

```

```

public function
scopeGetAllWithoutLimit($query)
{
    return DB::table($this->table)
        ->select(
            'barang.id as idbarang',
            'barang.nama_barang',
            'barang.nama_brand',
            'barang.satuan_barang',
            'barang.ukuran_barang',
            'barang.stok',
            'barang.stok_pengaman',
            'barang.harga_barang',
            'barang.harga_jual',
            'barang.biaya_penyimpanan',
            'barang.tanggal_retur',

```

```

            'barang.idusers',
            'barang.idsupplier',
            'supplier.nama as nama_supplier',
            'supplier.leadtime',
            'supplier.waktu_operasional',
            'supplier.biaya_pemesanan',
            'etalase.etalase',
            'etalase.ukuran_etalase',
            'kategori.kategori',

            DB::raw('(select count(id) from diskons
where idbarang=barang.id) as jumlah_diskon')
        )
        ->join('supplier', 'supplier.id', '=',
            'barang.idsupplier')
        ->join('kategori', 'kategori.id', '=',
            'barang.idkategori')
        ->join('etalase', 'etalase.id', '=',
            'barang.idetalase')
        ->orderBy('barang.id', 'desc')
        ->get();
}

```

```

public function scopeGetAllWithID($query,
$data)
{
    return DB::table($this->table)
        ->select(
            'barang.id as idbarang',
            'barang.nama_barang',
            'barang.nama_brand',
            'barang.satuan_barang',
            'barang.ukuran_barang',
            'barang.stok',

```

```

'barang.stok_pengaman',
'barang.harga_barang',
'barang.harga_jual',
'barang.biaya_penyimpanan',
'barang.tanggal_retur',
'barang.idusers',
'barang.idsupplier',
'supplier.nama as nama_supplier',
'supplier.leadtime',
'supplier.waktu_operasional',
'supplier.biaya_pemesanan',
'etalase.etalase',
'etalase.ukuran_etalase',
'kategori.kategori',

DB::raw('(select count(id) from diskons
where idbarang=barang.id) as jumlah_diskon')

)

->join('supplier', 'supplier.id', '=',
'barang.idsupplier')

->join('kategori', 'kategori.id', '=',
'barang.idkategori')

->join('etalase', 'etalase.id', '=',
'barang.idetalase')

// ->where('barang.id', $data)

->where(function ($q) use ($data)
{
    foreach ($data as $value) {
        $q->orWhere('barang.id', $value-
>idbarang);
    }
})

->orderBy('barang.id', 'desc')
->get();

```

```

}

public function
scopeGetAllByEtalase($query, $idetalase)
{
    return DB::table($this->table)
->select(
        'barang.id as idbarang',
        'barang.nama_barang',
        'barang.nama_brand',
        'barang.satuan_barang',
        'barang.ukuran_barang',
        'barang.stok',
        'barang.stok_pengaman',
        'barang.harga_barang',
        'barang.harga_jual',
        'barang.biaya_penyimpanan',
        'barang.tanggal_retur',
        'barang.idusers',
        'barang.idsupplier',
        'supplier.nama as nama_supplier',
        'supplier.leadtime',
        'supplier.waktu_operasional',
        'supplier.biaya_pemesanan',
        'etalase.etalase',
        'etalase.ukuran_etalase',
        'kategori.kategori',

        DB::raw('(select count(id) from diskons
where idbarang=barang.id) as jumlah_diskon')

    )

->join('supplier', 'supplier.id', '=',
'barang.idsupplier')

```

```

->join('kategori', 'kategori.id', '=',
'barang.idkategori')

->join('etalase', 'etalase.id', '=',
'barang.idetalase')

->where('barang.idetalase', $idetalse)

->orderBy('barang.id', 'desc')

->get();
}

```

public function scopeGetAll(\$query, \$limit)

```

{
    return DB::table($this->table)

->select(
    'barang.id',
    'barang.nama_barang',
    'barang.nama_brand',
    'barang.satuan_barang',
    'barang.ukuran_barang',
    'barang.stok',
    'barang.stok_pengaman',
    'barang.harga_barang',
    'barang.harga_jual',
    'barang.biaya_penyimpanan',
    'barang.tanggal_retur',
    'supplier.nama as nama_supplier',
    'supplier.leadtime',
    'supplier.waktu_operasional',
    'supplier.biaya_pemesanan',
    'etalase.etalase',
    'etalase.ukuran_etalase',
    'kategori.kategori',

```

```

DB::raw('(select count(id) from diskons
where idbarang=barang.id) as jumlah_diskon')
)

```

```

->join('supplier', 'supplier.id', '=',
'barang.idsupplier')

```

```

->join('kategori', 'kategori.id', '=',
'barang.idkategori')

```

```

->join('etalase', 'etalase.id', '=',
'barang.idetalase')

```

```

->orderBy('barang.id', 'desc')

```

```

->paginate($limit);

```

```

}

```

public function
scopeGetAllBySupplier(\$query, \$idsupplier)

```

{
    return DB::table($this->table)

->select(
    'barang.id',
    'barang.nama_barang',
    'barang.nama_brand',
    'barang.satuan_barang',
    'barang.ukuran_barang',
    'barang.stok',
    'barang.stok_pengaman',
    'barang.harga_barang',
    'barang.harga_jual',
    'barang.biaya_penyimpanan',
    'barang.tanggal_retur',
    'supplier.nama as nama_supplier',
    'supplier.leadtime',
    'supplier.waktu_operasional',
    'supplier.biaya_pemesanan',

```

```

        'etalase.etalase',

        'etalase.ukuran_etalase',

        'kategori.kategori',

        DB::raw('(select count(id) from diskons
where idbarang=barang.id) as jumlah_diskon')

    )

    ->join('supplier', 'supplier.id', '=',
'barang.idsupplier')

    ->join('kategori', 'kategori.id', '=',
'barang.idkategori')

    ->join('etalase', 'etalase.id', '=',
'barang.idetalase')

    ->where('barang.idsupplier', $idsupplier)

    ->orderBy('barang.id', 'desc')

    ->get();
}
}

```

<?php

```
namespace App;
```

```
use Illuminate\Database\Eloquent\Model;
```

```
use Illuminate\Support\Facades\DB;
```

```
class Diskon extends Model
```

```
{
```

```
    protected $table = 'diskons';
```

```
    public function
scopeCheckTypeDiscount($query, $idbarang,
$type)
```

```
{
```

```

        return DB::table($this->table)

        ->where('idbarang', $idbarang)

        ->where('tipe', $type)

        ->value('id');
    }

    public function scopeGetAll($query,
$idbarang, $limit)
    {
        return DB::table($this->table)

        ->select(

            'diskons.id',

            'diskons.diskon',

            'diskons.min',

            'diskons.max',

            'diskons.tipe',

            'diskons.idbarang',

            'diskons.created_at',

            'barang.harga_barang as harga'

        )

        ->join('barang', 'barang.id', '=',
'diskons.idbarang')

        ->where('diskons.idbarang', $idbarang)

        ->orderBy('id', 'desc')

        ->paginate($limit);
    }

    public function scopeGetAllNoLimit($query,
$idbarang)
    {
        return DB::table($this->table)

        ->select(

```

| | |
|---|---|
| 'diskons.id', | ->where('diskons.idbarang', \$idbarang) |
| 'diskons.diskon', | ->where('diskons.tipe', \$type) |
| 'diskons.min', | ->orderBy('id', 'asc') |
| 'diskons.max', | ->get(); |
| 'diskons.tipe', | } |
| 'diskons.idbarang', | |
| 'diskons.created_at', | } |
| 'barang.harga_barang as harga' | |
|) | <?php |
| ->join('barang', 'barang.id', '=',
'diskons.idbarang') | namespace App; |
| ->where('diskons.idbarang', \$idbarang) | |
| ->orderBy('id', 'asc') | use Illuminate\Database\Eloquent\Model; |
| ->get(); | |
| } | class Etalase extends Model |
| | { |
| public function scopeGetAllByType(\$query,
\$idbarang, \$type) | protected \$table = 'etalase'; |
| { | } |
| return DB::table(\$this->table) | |
| ->select(| <?php |
| 'diskons.id', | namespace App; |
| 'diskons.diskon', | |
| 'diskons.min', | use Illuminate\Database\Eloquent\Model; |
| 'diskons.max', | |
| 'diskons.tipe', | class Kategori extends Model |
| 'diskons.idbarang', | { |
| 'diskons.created_at', | protected \$table = 'kategori'; |
| 'barang.harga_barang as harga' | } |
|) | |
| ->join('barang', 'barang.id', '=',
'diskons.idbarang') | <?php |

```

namespace App;

use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\DB;

class Pembelian extends Model
{
    protected $table = 'pembelian';

    public function scopeGetAll($query, $limit)
    {
        return DB::table($this->table)
            ->select(
                $this->table.'.id',
                $this->table.'.idsupplier',
                'barang.id as id_barang',
                'barang.nama_barang',
                'barang.satuan_barang',
                'supplier.nama as nama_supplier',
                DB::raw('(select count(id) from
pembelian where
idsupplier=pembelian.idsupplier) as
total_barang')
            )
            ->leftJoin('barang', 'barang.id', '=', $this->table.'.idbarang')
            ->leftJoin('supplier', 'supplier.id', '=',
$this->table.'.idsupplier')
            ->orderBy($this->table.'.status', 'asc')
            ->groupBy($this->table.'.idsupplier')
            ->paginate($limit);
    }
}

```

```

public function
scopeJumlahPembelian($query, $idsupplier)
{
    return DB::table($this->table)
        ->where('idsupplier', $idsupplier)
        ->count('id');
}

public function
scopeTotalPembelian($query, $idsupplier)
{
    return DB::table($this->table)
        ->where('idsupplier', $idsupplier)
        ->sum('total_biaya');
}

public function scopeGetBySupplier($query,
$idsupplier, $limit)
{
    return DB::table($this->table)
        ->select(
            $this->table.'.id',
            $this->table.'.kode_transaksi',
            $this->table.'.jumlah_pembelian',
            $this->table.'.harga_barang',
            $this->table.'.biaya_penyimpanan',
            $this->table.'.diskon',
            $this->table.'.tanggal_pembelian',
            $this->table.'.status',
            'barang.id as id_barang',
            'barang.nama_barang',

```

```

        'barang.satuan_barang',
        'supplier.nama as nama_supplier'
    )
    ->leftJoin('barang', 'barang.id', '=', $this->table.'.idbarang')
    ->leftJoin('supplier', 'supplier.id', '=', $this->table.'.idsupplier')
    ->where($this->table.'.idsupplier', $idsupplier)
    ->orderBy($this->table.'.status', 'asc')
    ->paginate($limit);
}

```

```

public function
scopeGetAllForLaporan($query, $tglAwal, $tglAkhir, $order)
{
    return DB::table($this->table)
    ->select(
        $this->table.'.id',
        $this->table.'.kode_transaksi',
        $this->table.'.jumlah_pembelian',
        $this->table.'.harga_barang',
        $this->table.'.biaya_penyimpanan',
        $this->table.'.diskon',
        $this->table.'.tanggal_pembelian',
        $this->table.'.status',
        'barang.id as id_barang',
        'barang.nama_barang',
        'barang.satuan_barang',
        'supplier.nama as nama_supplier'
    )
}

```

```

    ->leftJoin('barang', 'barang.id', '=', $this->table.'.idbarang')
    ->leftJoin('supplier', 'supplier.id', '=', $this->table.'.idsupplier')
    ->whereBetween($this->table.'.created_at', [$tglAwal, $tglAkhir])
    ->orderBy($this->table.'.status', 'asc')
    ->get();
}

```

```

public function
scopeGetTotalCostMultiItemByIdsupplier($query, $idsupplier)
{
    return DB::table($this->table)
    ->where('idsupplier', $idsupplier)
    ->sum('total_cost');
}
}

```

<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

use Illuminate\Support\Facades\DB;

class Pemesanan extends Model

{

protected \$table = 'pemesanan';

public function getNamaBarang() {


```

        return $this->hasOne('App\Barang','id','idbarang');
    }

    public function scopeGetTotal($query)
    {
        return DB::table($this->table)
            ->count('id');
    }

    public function
    scopeGetAllReportPerItem($query, $tglAwal,
    $tglAkhir, $order)
    {
        return DB::table($this->table)
            ->select(
                'pemesanan.id',
                'pemesanan.harga_barang',
                'pemesanan.jumlah_unit',
                'pemesanan.total_cost',
                'pemesanan.reorder_point',
                'pemesanan.frekuensi_pembelian',
                'barang.id as id_barang',
                'barang.nama_barang',
                'barang.satuan_barang',
                'barang.idsupplier',
                'barang.biaya_penyimpanan',
                'supplier.biaya_pemesanan',
                'supplier.nama as nama_supplier'
            )
            ->leftJoin('barang', 'barang.id', '=',
            'pemesanan.idbarang')

```

```

            ->leftJoin('supplier', 'supplier.id', '=',
            'pemesanan.idsupplier')
        }

        -
        >whereBetween('pemesanan.created_at',
        [$tglAwal, $tglAkhir])

        ->orderBy('pemesanan.id', $order)

        ->get();
    }

    public function
    scopeGetAllReportPerItemBySupplier($query,
    $tglAwal, $tglAkhir, $order, $idsupplier)
    {
        return DB::table($this->table)
            ->select(
                'pemesanan.id',
                'pemesanan.harga_barang',
                'pemesanan.jumlah_unit',
                'pemesanan.total_cost',
                'pemesanan.reorder_point',
                'pemesanan.frekuensi_pembelian',
                'barang.id as id_barang',
                'barang.nama_barang',
                'barang.satuan_barang',
                'barang.idsupplier',
                'barang.biaya_penyimpanan',
                'supplier.biaya_pemesanan',
                'supplier.nama as nama_supplier'
            )
            ->leftJoin('barang', 'barang.id', '=',
            'pemesanan.idbarang')

            ->leftJoin('supplier', 'supplier.id', '=',
            'pemesanan.idsupplier')

```

```

-
>whereBetween('pemesanan.created_at',
[$tglAwal, $tglAkhir])

->where('pemesanan.idsupplier',
$idsupplier)

->orderBy('pemesanan.id', $order)

->get();
}

```

```

public function
scopeGetAllItemByType($query, $type)
{
    return DB::table($this->table)
    ->select(
        'pemesanan.id',
        'pemesanan.harga_barang',
        'pemesanan.jumlah_unit',
        'pemesanan.total_cost',
        'pemesanan.reorder_point',
        'pemesanan.frekuensi_pembelian',
        'pemesanan.tipe',
        'barang.id as id_barang',
        'barang.nama_barang',
        'barang.satuan_barang',
        'barang.idsupplier',
        'barang.biaya_penyimpanan',
        'barang.stok',
        'supplier.biaya_pemesanan',
        'supplier.nama as nama_supplier'
    )

    ->leftJoin('barang', 'barang.id', '=',
'pemesanan.idbarang')

```

```

->leftJoin('supplier', 'supplier.id', '=',
'pemesanan.idsupplier')

->orderBy('pemesanan.id', 'desc')

->where('pemesanan.tipe', $type)

->get();
}

```

```

public function
scopeGetAllSingleItem($query, $limit)
{
    return DB::table($this->table)
    ->select(
        'pemesanan.id',
        'pemesanan.harga_barang',
        'pemesanan.jumlah_unit',
        'pemesanan.total_cost',
        'pemesanan.reorder_point',
        'pemesanan.frekuensi_pembelian',
        'barang.id as id_barang',
        'barang.nama_barang',
        'barang.satuan_barang',
        'barang.idsupplier',
        'barang.biaya_penyimpanan',
        'supplier.biaya_pemesanan',
        'supplier.nama as nama_supplier'
    )

    ->leftJoin('barang', 'barang.id', '=',
'pemesanan.idbarang')

    ->leftJoin('supplier', 'supplier.id', '=',
'pemesanan.idsupplier')

    ->orderBy('pemesanan.id', 'desc')

    ->paginate($limit);
}

```