- CRUD PEMINJAMAN

CONTROLLER

```
public function store(Request $request){
      $validator=Validator::make($request->all(),
          'id anggota'=>'required',
          'id petugas'=>'required',
          'tgl pinjam'=>'required',
          'deadline'=>'required',
          'denda'=>'required'
      );
      if($validator->fails()){
        return Response()->json($validator->errors());
      $simpan=Pinjam::create([
        'id anggota'=>$request->id anggota,
        'id petugas'=>$request->id petugas,
        'tgl_pinjam'=>$request->tgl_pinjam,
        'deadline'=>$request->deadline,
        'denda'=>$request->denda
      ]);
      $status=1;
      $message="Peminjaman Berhasil Ditambahkan";
      if($simpan){
        return Response()->json(compact('status', 'message'));
      }else {
        return Response()->json(['status'=>0]);
    public function update($id, Request $request){
      $validator=Validator::make($request->all(),
          'id_anggota'=>'required',
          'id petugas'=>'required',
          'tgl pinjam'=>'required',
          'deadline'=>'required',
          'denda'=>'required'
    );
```

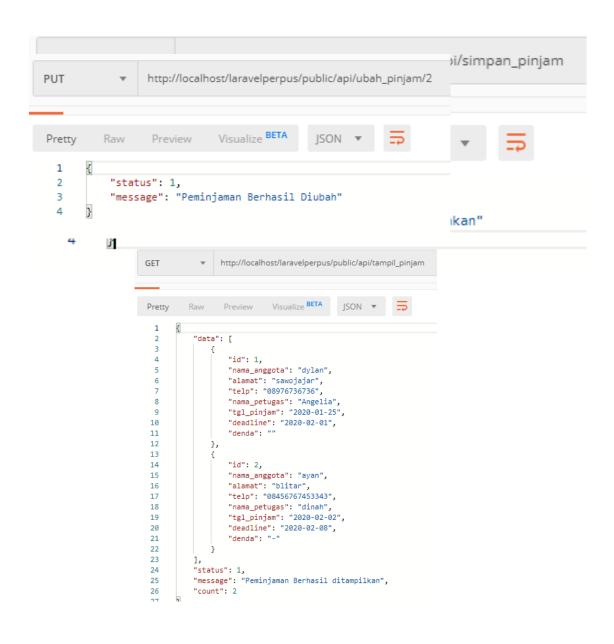
```
if($validator->fails()){
      return Response()->json($validator->errors());
    $ubah=Pinjam::where('id',$id)->update([
      'id anggota'=>$request->id anggota,
      'id petugas'=>$request->id petugas,
      'tgl_pinjam'=>$request->tgl_pinjam,
      'deadline'=>$request->deadline,
      'denda'=>$request->denda
    ]);
    $status=1;
    $message="Peminjaman Berhasil Diubah";
    if($ubah){
      return Response()->json(compact('status', 'message'));
    }else {
      return Response()->json(['status'=>0]);
 public function tampil pinjam(){
    $data=DB::table('peminjaman')
    ->join('anggota','anggota.id','=','peminjaman.id_anggota')
    ->join('petugas','petugas.id','=','peminjaman.id petugas')
>select('peminjaman.id', 'anggota.nama_anggota', 'anggota.alamat', 'anggot
a.telp','petugas.nama_petugas','peminjaman.tgl_pinjam','peminjaman.dead
line','peminjaman.denda')
    ->get();
   $count=$data->count();
    $status=1;
    $message="Peminjaman Berhasil ditampilkan";
    return response()-
>json(compact('data','status','message','count'));
 }
 public function destroy($id){
    $hapus=Pinjam::where('id',$id)->delete();
    $status=1;
    $message="Peminjaman Berhasil Dihapus";
   if($hapus){
      return Response()->json(compact('status', 'message'));
    }else {
      return Response()->json(['status'=>0]);
```

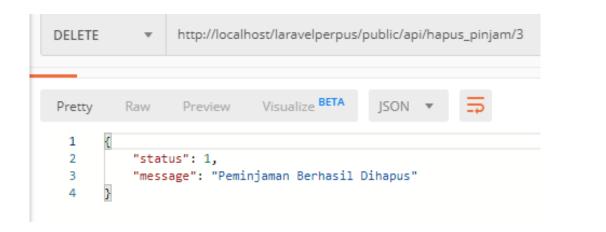
}

MODEL

```
class Pinjam extends Model
 protected $table="peminjaman";
 protected $primaryKey="id";
 protected $fillable = [
    'id_anggota',
    'id_petugas',
    'tgl_pinjam',
    'deadline',
    'denda'
 ];
 public function anggota() {
    return $this->belongsTo('App/Anggota', 'id_anggota');
 public function petugas() {
    return $this->belongsTo('App/Petugas', 'id_petugas');
 public function detail(){
    return $this->hasMany('App\Detail_pinjam','id');
```

HASIL





CRUD DETAIL PEMINJAMAN

CONTROLLER

```
public function simpan(Request $request){
    $validator=Validator::make($request->all(),
        'id pinjam'=>'required',
        'id buku'=>'required',
        'qty'=>'required'
    );
    if($validator->fails()){
      return Response()->json($validator->errors());
    $simpan=Detail pinjam::create([
      'id_pinjam'=>$request->id_pinjam,
      'id buku'=>$request->id buku,
      'qty'=>$request->qty
    ]);
    $status=1;
    $message="Detail Peminjaman Berhasil Ditambahkan";
    if($simpan){
      return Response()->json(compact('status', 'message'));
    }else {
      return Response()->json(['status'=>0]);
  public function ubah($id, Request $request){
    $validator=Validator::make($request->all(),
        'id_pinjam'=>'required',
        'id_buku'=>'required',
        'qty'=>'required'
  );
  if($validator->fails()){
    return Response()->json($validator->errors());
```

```
$ubah=Detail_pinjam::where('id',$id)->update([
    'id pinjam'=>$request->id pinjam,
    'id_buku'=>$request->id_buku,
    'qty'=>$request->qty
 ]);
 $status=1;
 $message="Detail Peminjaman Berhasil Diubah";
 if($ubah){
    return Response()->json(compact('status', 'message'));
 }else {
    return Response()->json(['status'=>0]);
public function hapus($id){
 $hapus=Detail_pinjam::where('id',$id)->delete();
 $status=1;
 $message="Detail Peminjaman Berhasil Dihapus";
 if($hapus){
    return Response()->json(compact('status', 'message'));
 }else {
    return Response()->json(['status'=>0]);
```

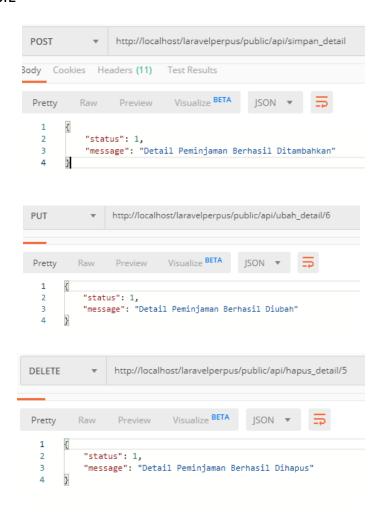
MODEL

```
class Detail_pinjam extends Model
{
  protected $table="detail_pinjam";
  protected $primaryKey="id";
  protected $fillable = [
    'id_pinjam',
    'id_buku',
    'qty'
];

public function buku() {
    return $this->belongsTo('App/Buku', 'id_buku');
}

public function peminjaman() {
    return $this->belongsTo('App/Pinjam', 'id_pinjam');
}
}
```

HASIL



- JOIN PEMINJAMAN DAN DETAIL PEMINJAMAN (TAMPIL DETAIL)
 - CONTROLLER

```
public function index(){
    if(Auth::user()->level=="petugas"){
        $peminjaman=DB::table('peminjaman')
        ->join('anggota','anggota.id','=','peminjaman.id_anggota')
        ->join('petugas','petugas.id','=','peminjaman.id_petugas')
    -
>select('peminjaman.id','peminjaman.id_anggota','anggota.nama_anggota','peminjaman.tgl_pinjam','peminjaman.deadline','peminjaman.denda')
    ->get();
```

```
$data=array();
 foreach ($peminjaman as $dt_pj){
   $ok=Detail_pinjam::where('id_pinjam',$dt_pj->id)->get();
   $arr detail=array();
   foreach ($ok as $ok) {
      $arr_detail[]=array(
      'id pinjam' =>$ok->id pinjam,
      'id_buku' => $ok->id_buku,
      'qty' => $ok->qty
     );
   $data[]=array(
      'id' => $dt_pj->id,
      'id_anggota' => $dt_pj->id_anggota,
      'nama_anggota' => $dt_pj->nama_anggota,
      'tgl_pinjam' => $dt_pj->tgl_pinjam,
      'deadline' => $dt_pj->deadline,
      'denda' => $dt_pj->denda,
      'detail_pinjam' => $arr_detail
   );
 return response()->json(compact("data"));
} else{
 return response()->json(['status'=>'anda bukan admin']);
```

• HASIL

```
GET
                 http://localhost/laravelperpus/public/api/peminjaman
                             Visualize BETA
Pretty
                                             JSON ▼
         Raw
                  Preview
2
           "data": [
 3
                   "id": 1,
  4
                   "id_anggota": 3,
  5
                   "nama_anggota": "dylan",
  6
                   "id_petugas": 9,
  8
                   "tgl_pinjam": "2020-01-25",
                   "deadline": "2020-02-01",
 9
                   "denda": "",
 10
                   "detail_pinjam": [
 11
 12
                       {
 13
                           "id_pinjam": 1,
 14
                           "id_buku": 5,
                           "qty": 1
 15
16
17
                   1
 18
               },
 19
 20
                   "id": 2,
                   "id_anggota": 5,
"nama_anggota": "ayan",
 21
 22
                   "id_petugas": 8,
23
                   "tgl_pinjam": "2020-02-02",
24
                   "deadline": "2020-02-08",
 25
 26
                   "denda": "-",
 27
                   "detail_pinjam": [
                    {
 28
```