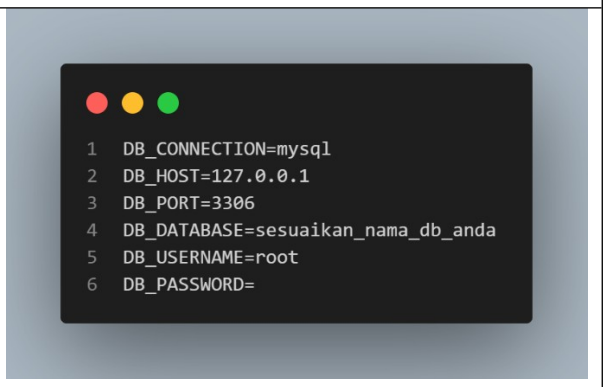
TATA CARA PENGERJAAN URUTAN LARAVEL – SERKOM

1. Copy file Master Laravel (Minimum php required 8.2) – ubah nama file dan sesuaikan
2. Buka file dengan Text editor vs code -> open folder
3. Buka phpMyAdmin dan buat database portal_berita_nama
4. Buka VS Code dan buka file .env ubah code seperti dibawah.

SEBELUM	SESUDAH
	

5. Setelah itu jalankan php artisan migrate di terminal
6. Setelah migrate berhasil silahkan buka phpMyAdmin dan sesuaikan database anda dengan soal project serkom
7. Sesuaikan table users (Ubah name menjadi nama , ubah atribut dari unsigned menjadi kosong)
8. Buat table kategori (sesuaikan field dengan soal)
9. Buat table berita (sesuaikan field dengan soal)
    a. id_berita (int 11 auto increment) PK
    b. id_kategori (int 11) FK
    c. id_user (bigint samakan dengan users) FK
10. Relasikan table nya

11. Buat table view untuk berita agar nama kategori dan nama user bisa tampil juga
12. Sesuaikan model user untuk persiapan seeder
13. Buat seeder untuk users

```
php artisan make:seeder UserSeeder
```

14. Gunakan Use DB dan Hash

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

15. Untuk seeder 2 data secara langsung kodenya seperti dibawah ini

```
public function run(): void
{
    DB::table('users')->insert([
        [
            'nama'=>'Nama user 1',
            'email'=>'emailuser1@gmail',
            'password'=>Hash::make('bebas')
        ],
        [
            'nama'=>'Nama User 2',
            'email'=>'emailuser2@gmail.com',
            'password'=>Hash::make('bebas')
        ],
    ]);
}
```

16. Setelah selesai silahkan seeder ke database dengan ketikkan perintah di terminal

```
php artisan db:seed UserSeeder
```

17. Buat model untuk berita `php artisan make:model Berita`
18. Sesuaikan Nama Field nya. Agar tidak eror matikan created_at dan update nya

```php
class Berita extends Model
{
    const CREATED_AT = null;
    const UPDATED_AT = null;

    protected $table = 'berita';
    protected $primaryKey = 'id_berita';
    protected $fillable = [
        'id_kategori', 'id_user', 'judul', 'ringkasan', 'isi', 'gambar', 'tgl_publish', 'utama'
    ];
}
```

19. Silahkan atur template admin dan login nya . selanjutnya adalah contoh kode yang mungkin bisa di implementasikan ke dalam program anda
20. Contoh kode login

```php
public function authlogin(Request $request)
{
    $data = [
        'email' => $request->input('email'),
        'password' => $request->input('password'),
    ];

    if (Auth::Attempt($data))
        {
            // Sesuikan sendiri namanya
            return redirect('dashboard');
        }
    else
        {
            Session::flash('error', 'Mohon periksa lagi email dan password anda!');
            return redirect('/login');
        }
}
```

21. Contoh kode pemisahan route middleware

```php
Route::middleware(['auth'])->group(function () {
    // masukkan semua route admin anda disini , contoh
    // Route::get('dashboard', [DashboardControllers::class, 'index'])->name('index');
});
```

22. Contoh kode untuk menampilkan data di administrator bisa juga di gunakan untuk menampilkan data di websitenya tinggal sesuaikan function dan arah viewnya

```php
public function berita(){

    $berita = DB::table('v_berita')->get();
    return view('admin.pages.berita', compact('berita'));
}
```

23. contoh kode tambah data berita

```php
public function berita_store(Request $request){

    // tambah data pegawai dengan gambar
    $request->validate([
        'gambar' => 'required|image|mimes:jpeg,png,jpg,webp|max:5048',
    ]);

    $data = $request->all();

    if ($request->hasFile('gambar')) {
        $file = $request->file('gambar');
        $filename = time().'_'.Str::slug($request->nama).'.'.$file->getClientOriginalExtension();
        // pindahkan foto ke direktori public -> folder gambar
        $path = $request->gambar->move(public_path('gambar'), $filename);
        $data['gambar'] = $filename;
    }

    BeritaModel::create($data);

     return redirect()->route('berita')
        ->with('berhasil', 'Data berhasil di tambahkan!');

}
```

24. contoh update data untuk menghandle gambar

```php
public function berita_update(Request $request, $id){

    $request->validate([
        'gambar' => 'nullable|image|mimes:jpg,jpeg,png,gif|max:5048',
    ]);

    $berita = BeritaModel::findOrFail($id);
    $data = $request->except('gambar');
    $data['utama'] = $request->has('utama') ? 'Y' : 'N';

    // Handle gambar
    if ($request->hasFile('gambar')) {
        if ($berita->gambar && Storage::disk('public')->exists($berita->gambar)) {

            unlink(public_path('gambar/'.$berita->gambar));
        }

        $file = $request->file('gambar');
        $filename = time() . '_' . Str::slug($request->nama) . '.' . $file->getClientOriginalExtension();
        $path = $request->gambar->move(public_path('gambar'), $filename);
        $data['gambar'] = $filename;
    }

    $berita->update($data);

    return redirect()->route('berita')->with('success', 'Berita berhasil diperbarui.');

}
```

25. Contoh kode hapus data

```php
public function berita_delete($id){
    $berita = BeritaModel::find($id);

    unlink(public_path('gambar/'.$berita->gambar));

    $berita->delete();
    return redirect()->route('berita')
        ->with('berhasil', 'Data berhasil di hapus!');
}
```

26. penggunaan use di controller (gunakan sesuai kebutuhan)

```php
namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\BeritaModel;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\DB;
use Illuminate\Support\Str;
use Illuminate\Support\Facades\Storage;
```