|  |  |
| --- | --- |
|  | **Tugas Praktikum Laravel**  Nama : Angela Fransisca Yulisiswati  NIM : 1204220007 |

**Praktek Modul**

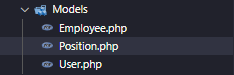
1. Pendahuluan

Tugas ini merupakan penerapan database tingkat lanjut dengan fokus materi pada Eloquent ORM dan Blade Templates Engine.

Eloquent ORM merupakan teknik pemetaan objek ke relasional yang memungkinkan berinteraksi dengan database menggunakan objek. Dengan ini tidak diperlukan lagi menulis query secara langsung.

Blade Template Engine merupakan mesin template laravel yang memungkinkan membuat tampilan dengna mudah, ringkas dan efisien.

1. Buat file model position dan employee dengan perintah php artisan make:models



1. Tambahkan relasi untuk postion dan employee

|  |
| --- |
| Relasi ini digunakan agar ketika pemanggilan data tidak perlu dilakukan secara manual dan juga dapat mempercepat load data. |
| <?php  namespace *App\Models*;  use *Illuminate\Database\Eloquent\Factories\HasFactory*;  use *Illuminate\Database\Eloquent\Model*;  class Position extends *Model*  {      use *HasFactory*;      public function employees()      {          return *$this*->hasMany(*Employee*::class);      }  } |

|  |
| --- |
|  |
| <?php  namespace *App\Models*;  use *Illuminate\Database\Eloquent\Factories\HasFactory*;  use *Illuminate\Database\Eloquent\Model*;  class Employee extends *Model*  {      use *HasFactory*;      public function position()      {          return *$this*->belongsTo(*Position*::class);      }  } |

1. Buat view pada EmployeeController

|  |
| --- |
| Controller ini bertipe resource yang mana untuk membuat get dapat dengan function index |
| <?php  namespace *App\Http\Controllers*;  use *App\Models\Employee*;  use *Illuminate\Http\Request*;  class EmployeeController extends *Controller*  {      public function index()      {          $pageTitle = 'Employee List';          $employees = *Employee*::all();          return view('employee.index', [              'pageTitle' => $pageTitle,              'employees' => $employees          ]);      }  } |

1. Buat view pada folder employee

|  |
| --- |
| View ini digunakan untuk menampilkan seluruh data employee dari database |
| <tbody>      @foreach ($employees as $employee)          <tr>              <td>{{ $employee->firstname }}</td>              <td>{{ $employee->lastname }}</td>              <td>{{ $employee->email }}</td>              <td>{{ $employee->age }}</td>              <td>{{ $employee->position->name }}</td>              <td>                  {{-- ACTIONS SECTION --}}                  <div *class*="d-flex">                      <a *href*="{{ route('employees.show', ['employee' => $employee->id]) }}"  *class*="btn btn-outline-dark btn-sm me-2"><i *class*="bi-person-lines-fill"></i></a>                      <a *href*="{{ route('employees.edit', ['employee' => $employee->id]) }}"  *class*="btn btn-outline-dark btn-sm me-2"><i *class*="bi-pencil-square"></i></a>                      <div>                          <form *action*="{{ route('employees.destroy', ['employee' => $employee->id]) }}" *method*="POST">                              @csrf                              @method('delete')                              <button *type*="submit" *class*="btn btn-outline-dark btn-sm me-2">                                  <i *class*="bi-trash"></i></button>                          </form>                      </div>                  </div>              </td>          </tr>      @endforeach  </tbody> |

1. Implementasikan validator

|  |
| --- |
| Fungsi store digunakan untuk menyimpan data. Validate akan memvalidasi data yang diterima pada variable request. Data ini akan divalidasi sesuai dengan kriteria yang telah ditulis |
| public function store(*Request* $request)      {          $messages = [              'required' => ':Attribute harus diisi.',              'email' => 'Isi :attribute dengan format yang benar',              'numeric' => 'Isi :attribute dengan angka'          ];          $validator = *Validator*::make($request->all(), [              'firstName' => 'required',              'lastName' => 'required',              'email' => 'required|email',              'age' => 'required|numeric',          ], $messages);          if ($validator->fails()) {              return redirect()->back()->withErrors($validator)->withInput();          }          // ELOQUENT          $employee = new *Employee*;          $employee->firstname = $request->firstName;          $employee->lastname = $request->lastName;          $employee->email = $request->email;          $employee->age = $request->age;          $employee->position\_id = $request->position;          $employee->save();          return redirect()->route('employees.index');      } |

1. Sesuaikan view pada bagian edit

|  |
| --- |
|  |
| <form *action*="{{ route('employees.update', ['employee' => $employee->id]) }}" *method*="POST">      @csrf      @method('put')      <div *class*="row justify-content-center">          <div *class*="p-5 bg-light rounded-3 col-xl-6">              <div *class*="mb-3 text-center">                  <i *class*="bi-person-circle fs-1"></i>                  <h4>Edit Employee</h4>              </div>              <hr>              <div *class*="row">                  <div *class*="col-md-6 mb-3">                      <label *for*="firstName" *class*="form-                          label">First Name</label>                      <input *class*="form-control                          @error('firstName') is-invalid @enderror"  *type*="text" *name*="firstName" *id*="firstName"  *value*="{{ $errors->any() ? old('firstName') : $employee->firstname }}"  *placeholder*="Enter First Name">                      @error('firstName')                          <div *class*="text-danger">                              <small>{{ $message }}</small>                          </div>                      @enderror                  </div>                  <div *class*="col-md-6 mb-3">                      <label *for*="lastName" *class*="form-label">Last Name</label>                      <input *class*="form-control @error('lastName')                          is-invalid @enderror"  *type*="text" *name*="lastName" *id*="lastName"  *value*="{{ $errors->any() ? old('lastName') : $employee->lastname }}"  *placeholder*="Enter Last Name">                      @error('lastName')                          <div *class*="text-danger">                              <small>{{ $message }}</small>                          </div>                      @enderror                  </div>                  <div *class*="col-md-6 mb-3">                      <label *for*="email" *class*="form-                          label">Email</label>                      <input *class*="form-control @error('email')                          is-invalid @enderror"  *type*="text" *name*="email" *id*="email"  *value*="{{ $errors->any() ? old('email') : $employee->email }}" *placeholder*="Enter Email">                      @error('email')                          <div *class*="text-danger">                              <small>{{ $message }}</small>                          </div>                      @enderror                  </div>                  <div *class*="col-md-6 mb-3">                      <label *for*="age" *class*="form-                          label">Age</label>                      <input *class*="form-control @error('age') is-                          invalid @enderror"  *type*="text" *name*="age" *id*="age"  *value*="{{ $errors->any() ? old('age') : $employee->age }}" *placeholder*="Enter Age">                      @error('age')                          <div *class*="text-danger">                              <small>{{ $message }}</small>                          </div>                      @enderror                  </div>                  <div *class*="col-md-12 mb-3">                      <label *for*="position" *class*="form-                          label">Position</label>                      <select *name*="position" *id*="position" *class*="form-select">                          @php                              $selected = '';                              if ($errors->any()) {                                  $selected = old('position');                              } else {                                  $selected = $employee->position\_id;                              }                          @endphp                          @foreach ($positions as $position)                              <option *value*="{{ $position->id }}" {{ $selected == $position->id ? 'selected' : '' }}>                                  {{ $position->code .                                      ' -                                                                                                                          ' .                                      $position->name }}                              </option>                          @endforeach                      </select>                      @error('position')                          <div *class*="text-danger">                              <small>{{ $message }}</small>                          </div>                      @enderror                  </div>              </div>              <hr>              <div *class*="row">                  <div *class*="col-md-6 d-grid">                      <a *href*="{{ route('employees.index') }}" *class*="btn btn-outline-dark btn-lg mt-3"><i  *class*="bi-arrow-left-circle                          me-2"></i> Cancel</a>                  </div>                  <div *class*="col-md-6 d-grid">                      <button *type*="submit" *class*="btn btn-dark btn-lg mt-3"><i *class*="bi-check-circle me-2"></i>                          Edit</button>                  </div>              </div>          </div>      </div>  </form> |

1. Implementasikan destroy pada EmployeeController

|  |
| --- |
| Destroy digunakan untuk melakukan penghapusan data. Untuk menghapus data dibutuhkan id |
| public function destroy(string $id)      {          // ELOQUENT  *Employee*::find($id)->delete();          return redirect()->route('employees.index');      } |

1. Pisahkan navbar menjadi file tersendiri

|  |
| --- |
| Navbar akan dipisahkan guna mempermudah pemanggilan navbar nantinya. Dengan cara ini kode lebih reusable |
| @php      $currentRouteName = Route::currentRouteName();  @endphp  <nav *class*="navbar navbar-expand-md navbar-dark bg-primary">      <div *class*="container">          <a *href*="{{ route('home') }}" *class*="navbar-brand mb-0 h1"><i *class*="bi-hexagon-fill me-2"></i> Data Master</a>          <button *type*="button" *class*="navbar-toggler" *data-bs-* *toggle*="collapse" *data-bs-target*="#navbarSupportedContent">              <span *class*="navbar-toggler-icon"></span>          </button>          <div *class*="collapse navbar-collapse" *id*="navbarSupportedContent">              <hr *class*="d-md-none text-white-50">              <ul *class*="navbar-nav flex-row flex-wrap">                  <li *class*="nav-item col-2 col-md-auto"><a *href*="{{ route('home') }}"  *class*="nav-link @if ($currentRouteName == 'home') active @endif">Home</a></li>                  <li *class*="nav-item col-2 col-md-auto"><a *href*="{{ route('employees.index') }}"  *class*="nav-link  @if ($currentRouteName == 'employees.index') active @endif">Employee</a></li>              </ul>              <hr *class*="d-md-none text-white-50">              <a *href*="{{ route('profile') }}" *class*="btn btn-outline-light  my-2 ms-md-auto"><i  *class*="bi-person-circle me-1"></i> My Profile</a>          </div>      </div>  </nav> |

1. Definisikan layout utama aplikasi

|  |
| --- |
| Layout utama dapat dibuat dengan @yeild. Fungsi ini nantinya akan diisi dengan section file yang meng-extends file dengan @yeild |
| <!DOCTYPE *html*>  <html *lang*="en">  <head>      <meta *charset*="UTF-8">      <meta *http-equiv*="X-UA-Compatible" *content*="IE=edge">      <meta *name*="viewport" *content*="width=device-width, initial-  scale=1.0">      <title>{{ $pageTitle }}</title>      @vite('resources/sass/app.scss')  </head>  <body>      @include('layouts.nav')      @yield('content')      @vite('resources/js/app.js')  </body>  </html> |

1. Buat template untuk default konten

|  |
| --- |
|  |
| <div *class*="container mt-4">      <h4>{{ $pageTitle }}</h4>      <hr>      <div *class*="d-flex align-items-center py-2 px-4 bg-light rounded-3      border">          <div *class*="bi-house-fill me-3 fs-1"></div>          <h4 *class*="mb-0">Well done! this is {{ $pageTitle }}.</h4>      </div>  </div> |

1. Pisahkan actions section

|  |
| --- |
|  |
| <div *class*="d-flex">      <a *href*="{{ route('employees.show', ['employee' => $employee->id]) }}" *class*="btn btn-outline-dark btn-sm me-2"><i  *class*="bi-person-      lines-fill"></i></a>      <a *href*="{{ route('employees.edit', ['employee' => $employee->id]) }}" *class*="btn btn-outline-dark btn-sm me-2"><i  *class*="bi-pencil-      square"></i></a>      <div>          <form *action*="{{ route('employees.destroy', ['employee' => $employee->id]) }}" *method*="POST">              @csrf              @method('delete')              <button *type*="submit" *class*="btn btn-outline-dark btn-sm me-      2"><i *class*="bi-trash"></i></button>          </form>      </div>  </div> |

1. Sesuaikan file view untuk menampilkan data pegawai

|  |
| --- |
| File layout utama yang sudah dibuat tadi dapat dipanggil. Sehingga nantinya tampilan pada index akan menyatu dengan app |
| @extends('layouts.app')  @section('content')      <div *class*="container mt-4">          <div *class*="row mb-0">              <div *class*="col-lg-9 col-xl-10">                  <h4 *class*="mb-3">{{ $pageTitle }}</h4>              </div>              <div *class*="col-lg-3 col-xl-2">                  <div *class*="d-grid gap-2">                      <a *href*="{{ route('employees.create') }}" *class*="btn  btn-primary">Create Employee</a>                  </div>              </div>          </div>          <hr>          <div *class*="table-responsive border p-3 rounded-3">              <table *class*="table table-bordered table-hover table-striped  mb-0 bg-white">                  <thead>                      <tr>                          <th>First Name</th>                          <th>Last Name</th>                          <th>Email</th>                          <th>Age</th>                          <th>Position</th>                          <th></th>                      </tr>                  </thead>                  <tbody>                      @foreach ($employees as $employee)                          <tr>                              <td>{{ $employee->firstname }}</td>                              <td>{{ $employee->lastname }}</td>                              <td>{{ $employee->email }}</td>                              <td>{{ $employee->age }}</td>                              <td>{{ $employee->position->name }}</td>                              <td>@include('employee.actions')</td>                          </tr>                      @endforeach                  </tbody>              </table>          </div>      </div>  @endsection |

1. Sesuaikan file view untuk fitur form input pegawai

|  |
| --- |
| Form input pegawai juga akan menerapkan blade template engine |
| @extends('layouts.app')  @section('content')      <form *action*="{{ route('employees.store') }}" *method*="POST">          @csrf          <div *class*="row justify-content-center">              <div *class*="p-5 bg-light rounded-3 border col-xl-6">                  <div *class*="mb-3 text-center">                      <i *class*="bi-person-circle fs-1"></i> |

1. Sesuaikan file view untuk fitur detail data pegawai

|  |
| --- |
| Begitu pula pada view |
| @extends('layouts.app')  @section('content')      <div *class*="container-sm my-5">          <div *class*="row justify-content-center">              <div *class*="p-5 bg-light rounded-3 col-xl-4 border">                  <div *class*="mb-3 text-center">                      <i *class*="bi-person-circle fs-1"></i>                      <h4>Detail Employee</h4> |

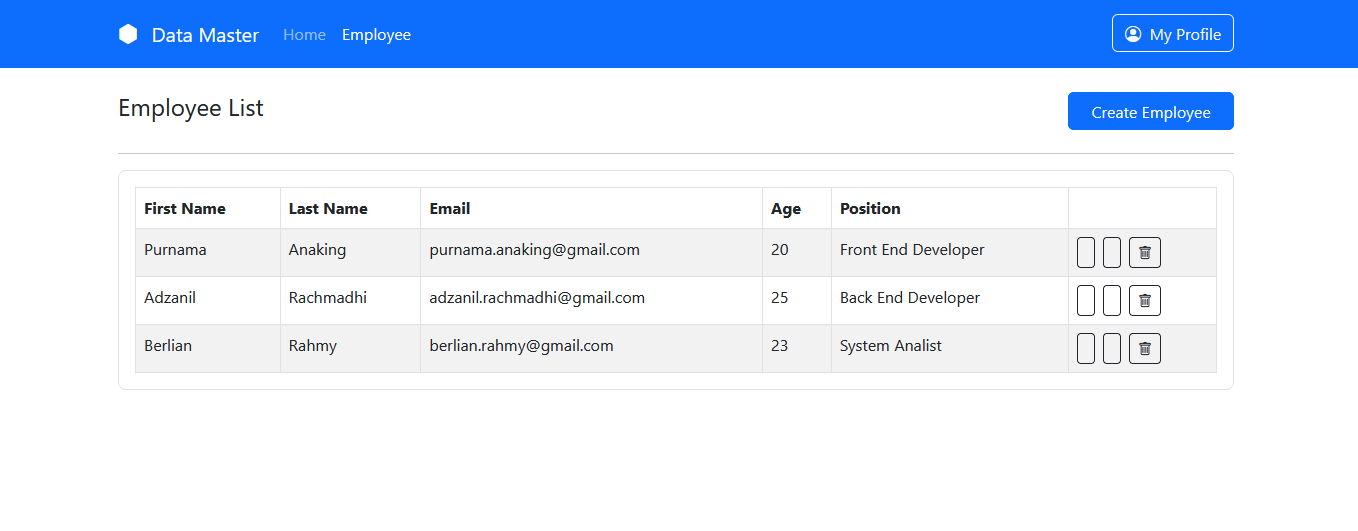
1. Penyesuaian file view untuk fitur form edit pegawai

|  |
| --- |
| Dan file-file default lainya |
| @extends('layouts.app')  @section('content')      <form *action*="{{ route('employees.update', ['employee' => $employee->id]) }}" *method*="POST">          @csrf          @method('put')          <div *class*="row justify-content-center">              <div *class*="p-5 bg-light rounded-3 col-xl-6">                  <div *class*="mb-3 text-center">                      <i *class*="bi-person-circle fs-1"> |

1. Sesuaikan file file untuk halaman home dan profile

|  |
| --- |
|  |
| @extends('layouts.app')  @section('content')      @include('default')  @endsection |

1. Hasil Output



**Link Github**