

PROIEKTU BAT SORTU

- AULA ADIBIDEA

[Alumnos erregistro guztiak erakutsi](#)

[Erregistro bat ikustaraztea](#)

[Ikasle berri bat sortu: Create eta store metodoak](#)

[Ikasle baten datuak aldatu: Edit eta update metodoak](#)

[Ikasle bat ezabatu: destroy\(\) metodoa](#)

[Ikasle bat ezabatu - Ezabatzeko konfirmazioa eskatu](#)

Alumnos erregistro guztiak erakutsi

Gure ikasleen zerrenda bistaratzeko hurrengo pausuak eman beharko ditugu:

- Ruta sortu
- AlumnoController kontroladorea sortu
- Kontroladorean beharrezkoan den metodoa sortu eta datu basean dauden ikasle guztiak bilatu all() metodoarekin
- Bista sortu aurkitutako ikasleak bistaratzeko

1. Ruta edo ibilbidea honako hau izango zen:

```
//Ikasle guztiak ikusteko
Route::get('alumnos', [AlumnoController::class, 'index'] )->name('alumnos.index');
```

2. AlumnoController sortu

3. **AlumnoController** kontrolatzalea irekiz hasiko gara. Index metodoa sortuko dugu eta horren barruan Eloquent all() metodoa erabiliko dugu. Jasotakoa \$alumnos aldagaien sartuko dugu eta hau da bista pasatuko duguna

```
public function index(){
    $alumnos = Alumno::all();
    return view('alumnos.index', compact('alumnos'));
}
```

4. Oraingoan bista sortu beharko genuke. Foreach batekin bilatuko dugu \$alumnos aldagaien dauden datu guztiak.

Laravelen, badaude beste kontbizio pare bat bistak sortzeko, alde batetik bistra deitzen duen metodoaren izen berdina jarri ohi zaio fitxategiari eta bestetik kontroladore batek deitzen dituen bista guztiak azpikarpetan sortzen dira, kontroladore horrek duen izen berdinarekin, kasu honetan "alumnos" karpeta izango da.

```
@foreach ($alumnos as $alumno)
    <label><strong>Izen abizenak:</strong> {{ $alumno->nombre }}</label><br>
    <label><strong>Adina:</strong> {{ $alumno->edad }}</label><br>
    <label><strong>Telefonoa:</strong> {{ $alumno->telefono }}</label><br>
    <label><strong>Helbidea:</strong> {{ $alumno->direccion }}</label>
@endforeach
```

Erregistro bat ikustaraztea

- Ruta sortu
- Kontroladorean beharrezkoan den metodoa sortu eta datu basean ikaslea ateru find() metodoarekin
- Bista sortu aurkitutako ikasleak bistaratzeko
- Aurreko bistan, ikasle bakoitzari gehitu ikasle zehatzta ikusteko esteka

Erregistro bakar bat erekusteko, show() metodoa erabiliko dugu:

1. **AlumnoController** kontrolatzalea irekiz hasiko gara. Show() metodoaren barruan, \$alumno motako objektu bat lortuko duen aldagai bat abiaraziko dugu. *find()* metodoari \$alumno parametro bat emango diogu erregistro jakin bat berreskuratzeko, eta, azkenik, ikaslea datuak berreskuratuta ikusiko dugu.

```
public function show($alumno){
    $alumno = Alumno::find($alumno);
    return view('alumnos.show', compact('alumno'));
}
```

2. Ibilbidea honako hau izango zen:

```
Route::get('alumnos/{alumno}', [AlumnoController::class, 'show'] )->name('alumnos.show');
```

3. Bista: **show.blade.php**

```
<label><strong>Izen abizenak:</strong> {{ $alumno->nombre }}</label><br>
<label><strong>Adina:</strong> {{ $alumno->edad }}</label><br>
<label><strong>Telefonoa:</strong> {{ $alumno->telefono }}</label><br>
<label><strong>Helbidea:</strong> {{ $alumno->direccion }}</label>
```

4. Aurreko bistan esteka gehitu:

```
<a href="{{ route('alumnos.show', $alumno->id) }}">Izen abizenak: {{ $alumno->nombre }}</a>
```

Ikasle berri bat sortu: Create eta store metodoak

- Ruta sortu
- Kontroladorean beharrezkoaren diren metodoak sortu eta datu basean ikaslea sortu: create() eta store() metodoak erabiliz.
- Bista sortu ikaslea sortzeko formularioarekin
- Menu bat sortu index eta ikaslea sortu aukererekin, horretarako layouts erabiliko ditugu

Oraingo honetan, ikasle berri bat sortuko dugu gure datu basean, horretarako metodo bi erabiltzen dira, create() eta store(): lehenengoan formulario sortuko dugu eta bigarrenean datuak gure datu basean sartuko ditugu.

1. Lehenengo gauza beraz bista berri bat sortuko dugu, "create" izeneko eta horren ruta sortuko dugu jarraian.

2. **Bista**, "alumnos" karpetan sortuko dugu "create.blade.php" izenarekin

3. Hurrengo hau izango da formularioa izango duen bistaren kodea:

```
<form method="post" action="{{ route('alumnos.store') }}">
    @csrf
    Izen Abizena <input type="text" name="nombre" value=""><br>
    Adina <input type="text" name="edad" value=""><br>
    Telefonoa <input type="text" name="telefono" value=""><br>
    Helbidea<input type="text" name="direccion" value=""><br>
    <button type="submit">Bidali</button>
</form>
```

4. Fijatu, zelan formularioaren barruan, Laravelek **@csrf** tokena sartzea eskatzen digu, hau segurtasunagatik egiten da, atakeak sahesteko erabilitzen da. Horrelako token bat sortuko dugu formulario bakotzeko. Laravelek derrigorrez eskatzen digu token hau izatea

5. **Rutak**. Bi beharko ditugu, get bat formularioari deitzeko eta beste bat, post metodoarekin formularioaren datuak DB-an gordetzeko:

```
Route::get('alumnos/create', [AlumnoController::class, 'create']) ->name('alumnos.create');
Route::post('alumnos/store', [AlumnoController::class, 'store']) ->name('alumnos.store');
```

6. **AlumnoController**:

Create() metodoa ikasle berri bat sortzeko behar dugun formularioa erakusteko:

```
public function create(){
    return view('alumnos.create');
}
```

Store() metodoa formularioak bidalitako informazioa datu basean gordetzeko:

```
public function store(Request $request){
    $alumno = new Alumno();
    $alumno->nombre = $request->nombre;
    $alumno->edad = $request->edad;
    $alumno->telefono = $request->telefono;
    $alumno->direccion = $request->direccion;
    $alumno->save();
    return redirect()->route('alumnos.show',$alumno);
}
```

7. Index bistan ikasle berri bat sortzeko esteka jarriko dugu ere:

```
<a href="{{ route('alumnos.create') }}">Ikasle berria</a>
```

Ikasle baten datuak aldatu: Edit eta update metodoak

- Ruta sortu
- Kontroladorean beharrezkoaren diren metodoak sortu eta datu basean ikaslea sortu: edit() eta update() metodoak erabiliz.
- Bista sortu ikaslea sortzeko formularioarekin
- Ikaslea aldatzeko esteka jarri, index eta show aukeretan

1. Lehenengo **edit()** funtzioan aldatu nahi dugun ikaslea bilatuko dugu, horretarako **find(\$id)** metodoa erabiliko dugu. Datuak edit bistan kargatzeko ditugu. Formulario honek update funtziari deituko dio, datuak DBan aldatuko dituena.

AlumnoController

```
public function edit(Alumno $alumno){
    return view('alumnos.edit', compact('alumno'));
}
```

Ikusi dezakezueñez, orain, aldagai arrunta jaso beharrean, Alumno jarri dugu aurretik, horrela jartzeraoan Laravelek ulertzen du \$alumno Alumno klasearen instantzia bat dela.

2. Metodo honi deitzen dion ruta beharko dugu:

```
Route::get('alumnos/edit/{alumno}', [AlumnoController::class, 'edit']) ->name('alumnos.edit');
```

3. Ikaslearen datuak aldatzeko beharrezkoaren diren formularioa bertsua baten sortuko dugu ere, baina formulario honetan gauza pare bat izango ditugu, alde batetik, oraingoan valuen ikasle horrek dituen datuak erakutsiko ditugu, eta bestetik metodoa aldatu behar izango dugu, Laravelek updateak egiterakoan rutak POST metodoa izan beharrean nahiago du PUT metodoa izatea, eta hori horrela izatean formularioan adierazi behar diogu ere:

```
<form method="post" action="{{ route('alumnos.update', $alumno) }}">
```

```

@csrf
@method('PUT')
Izen Abizena <input type="text" name="nombre" value="{{ $alumno->nombre }}><br>
Adina <input type="text" name="edad" value="{{ $alumno->edad }}><br>
Telefonoa <input type="text" name="telefono" value="{{ $alumno->telefono }}><br>
Helbidea<input type="text" name="direccion" value="{{ $alumno->direccion }}><br>
<button type="submit">Bidali aldaketak</button>
</form>

```

4. Updatea egiteko ruta horrelako izango litzateke, put metodoa erabiliz:

```
Route::put('alumnos/update/{alumno}', [AlumnoController::class, 'update']) ->name('alumnos.update');
```

5. Kontroladorean, prestatuko dugu ere **update** funtzia, honek, DBa bidalitako datuekin aldatuko du, hau da edit bistak hurrengo funtzio honi deituko dio, kasu honetan metodoak alde batetik formularioak bidalitakoa jasoko du, eta bestetik aldatu behar dugun objektua ere.

```

public function update(Request $request, Alumno $alumno){
    $alumno->nombre = $request->nombre;
    $alumno->edad = $request->edad;
    $alumno->telefono = $request->telefono;
    $alumno->direccion = $request->direccion;
    $alumno->save();
    return redirect()->route('alumnos.index');
}

```

Ikasle bat ezabatu: destroy() metodoa

- Ruta sortu
- Kontroladorean beharrezko den metodoa sortu, destroy() metodoa izago da hau
- Ikaslea ezabatzeko esteka jarri, index eta show aukeretan

1. Ruta sortu:

```
Route::delete('alumnos/{alumno}', [AlumnoController::class, 'destroy']) ->name('alumnos.destroy');
```

Ikusten dugunez, ruta hau delete metodoari deitzen dio, hau da ezabatzeko Laravelek erabiltzen duen metodoa

2. **AlumnoController** barruan sortuko dugu erregistro bat ezabatzeko metodoa, *destroy()* metodoa izango da hau, honen barruan *delete()* metodoari deituko diogu.

```

public function destroy(Alumno $alumno){
    $alumno->delete();
    return redirect()->route('alumnos.index');
}

```

3. Ikaslea ezabatzeko esteka. Ruta egiterakoan ikusi dugunez, Laravelek delete metodoa erabiltze du datu baseko erregistro bat ezabatzeko, beraz, ruta hori deitzerakoan, html-an, metodo hori erabiliko dugula adierazi beharko dugu, eta hori formulario baten barruan soilik egin dezakegu, beraz, destroy ruta bat beti formulario baten bidez egin behar dugu.

```

<form action="{{ route('alumnos.destroy', $alumno) }}" method="POST">
    @csrf
    @method('DELETE')
    <button type="submit">EZABATU</button>
</form>

```

4. Ikusten dugunez, honek zuzenean ezabatzen du erregistroa.

Ikasle bat ezabatu - Ezabatzeko konfirmazioa eskatu

Baina, normalean badakigu zeozer ezabatu baino lehen aukera eman behar dugula ziurtatzeko erregistro hori ezabatu nahi dugula, zelan egindo genuke hori ordun?

- Ruta berria sortu
- Kontroladorean konfirmazio bistari deitu
- Konfirmazio bista sortu

1. Ruta sortu:

```
Route::get('alumnos/confirm/{alumno}', [AlumnoController::class, 'confirm']) ->name('alumnos.confirm');
```

2. **AlumnoController** barruan sortuko dugu erregistro bat konfirmatzeko metodoa.

```

public function confirm(Alumno $alumno){
    return view('alumnos.confirm', compact('alumno'));
}

```

3. **confirm.blade.php** - Bista bat sortuko dugu formulario batekin, honek ezabatzeko botoia izango du eta Cancel esteka ere.

```

<form action="{{ route('alumnos.destroy', $alumno) }}" method="POST">
    @csrf
    @method('DELETE')
    <button type="submit">EZABATU</button>
    <a href="{{ route('alumnos.index') }}">
        Cancel
    </a>
</form>

```

