Crud Laravel : (BLADES&CONTROLLER)

-

BLADES:

CreateElv.blade.php:
```
@section('content')  <h1>Nouveau Eleve :</h1>
<form action="{{route(eleve.store')}}" method="POST">
  @csrf  <div class="">
    <label for="nom">Nom</label><br>
    <input  id="nom" type="text" name="nom"
value="{{old('nom')}}"> <br> @error('nom')
    <div class="">{{$message}}</div>  @enderror </div>
  <div class="">
    <label for="prenom">Prenom</label><br>
    <input id="prenom" type="text" name="adresse"
value="{{old(prenom)}}"> <br>  @error(prenom)
    <div class="">{{$message}}</div>  @enderror</div>
<input type="submit" value="Ajouter"></form>
@endsection
```

Index.Blade.php:
```
@section('content')
  <h1>Listes Des Elevs : </h1>
  <a href="{{route(elevs.create')}}">Nouveau
Eleve : :</a> <br> <table> <thead>  <tr> <th> id</th>
<th>Nom</th> <th>Prenom</th> </tr> </thead>
<tbody>
    @forelse ($elevs as $elv)  <tr>
    <td >{{$elv->id}}</td> <td >{{$elv->nom}}</td>
    <td >{{$elv->adresse}}</td>  <td>
    <a href="{{ url(elevs/'.$elvs->id)}}" ">View</a>
    <a href="{{ url(elevs/'.$elvs->id.'/edit')}}
>Edit</a> </td><td>
      <button url="{{elevs/'.$elvs->id}}" ><button/>
</td> </tr>  @empty  Aucun Eleve @endforelse
</tbody> </table> @endsection
```

Show.blade.php:
```
@section('content')
<h1>{{$elev ->nom}}</h1> <p>{{$elev ->prenom}}</p>
<ul>  <li> Nom : {{$elev ->nom}} </li>
    <li>Prenom : {{$elev ->adresse}}</li> </ul>
@endsection
```

Delete.blade.php:
```
@section('content')
<form action="{{route('elv.destroy',$elv->id)}}"
method="POST"> @csrf @methode("DELETE")
<p>Etes Vous sure de supprimer  {{$elv->nom}} ????</p>
</form>
@endsection
```

Controller(eleves are in Club):
```
public function index(){
  $elevs = Eleve::find()->all()->paginate->latest(5)->get() ;
    return view('elvs.index',['elevs=>$elevs]);}
public function create(){
  $elevs = Eleve::find()->all()->latest()->get()
    return view('elvs.create,['elevs=>$elevs]);}

public function show(String $id){
  $elevs = Eleve::findOrFail($id):
    return view('elvs.show,['elevs=>$elevs]);}
public function edit(String $id){
  $elevs = Eleve::findOrFail($id):
    return view('elvs.edit,['elevs=>$elevs]);}

public function store(Request $request){
  $validated =$request->validate([
      "nom"=>'required|max:255'
      "prenom"=>'required|max:255']) ;
    Eleve::create($validated)
    return redirect('/elevs')

public function update(Request $request,  string $id) {
    $elev = Eleve::findOrFail($id);
    $validated = $request->validate([
      'nom'=>'required|min:3',
      prenom => 'required|min:3', ]);
    $elev ->update($validated);
    return redirect('/elevs); }

  public function destroy(string $id) {
    $elev = Eleve::findOrFail($id);
    $elev->delete();
    return redirect('/elevs);
```

Model (Les relations) : (Eleve in Club)

Club.php:
```
class club extens model {
    use HasFactory;
    protected $fillable=[
      'nom', 'adresse','ville' ];
    public function eleves(){
      return $this->HasMany(Eleve::class);}}
```

Eleve.php:
```
class eleve extens model {
    use HasFacory ;
    protected $fillable=[
      'nom', 'prenom']
  Public function clubs(){
    Return $this->BelongsTo(Club::class)}}
```

Migration:
```
public function up() :void {
  Schema::create('eleves' function(Bluprint
$table){
      $table->id() , $table->string('nom'),
      $table->string('prnom')} ); }
public function down(): void {
  Schema::dropIfExists('eleves') }
```

Routes:
```
 la vue index Route::get('/index',
[nom_controller::class,'index'])->name('index')
Pour la vue show Route::get('/show/{test},
[nom_controller::class,'show'])->name('show')
 Pour ajouter la vue create Route::get('/create',
[nom_controller::class,'create'])->name('create')
Pour modifier la vue edite Route::get('/edit/{test}',
[nom_controller::class,'edit'])->name('edit')
Pour souvgarder Route::post('/store',
[nom_controller::class::class,'store'])->name('store')
Pour update Route::update('/update/{test}',
[nom_controller::class,'update'])->name('update')
Pour delete Route::delete('/destroy/{test},
[nom_controller::class,'destroy'])->name('destroy')
 Pour voire tout ca dans une route
Route::resources('/test,nom_controller::class)
```

Factory :
```
class EleveFactory extends Factory{
   public function definition(): array{
     return [
      'nom'=>fake()->word(),
      prenom=>fake()->word() ]; }}
```

Seeder :
```
class DatabaseSeeder extends Seeder {
   public function run(): void {
     Eleve::factory(5)->create();
     Club::factory(15)->create(); } }
Or :
DB::table('test')-
>insert([ 'nom'=>'moussan', 'prenom=>'flan '])
```

Laravel Breeze=>
```
 1 : composer require laravel/breeze
 2 : php artisan breeze:install
 3 : php artisan migrate
 4 : npm install
 5 : npm run dev
```

Midlleware: un middleware pour empêcher l'accès aux utilisateurs qui ne sont pas autorisés :
```
class CheckAuthorization {
   public function handle($request, Closure $next) {
     // Vérification de l'autorisation ici
     if (!Auth::check() || !Auth::user()->isAdmin()) {
        return abort(403, 'Accès interdit'); }
     return $next($request);   } }
```

Les Commandes :
create projet => composer create-project laravel/laravel example-app
lancer projet => php artisan serve
create model => php artisan make:model Test
create controller => php artisan make:controller TestController -r
create migration => php artisan make:migration create_test_table

.env:

DB_CONNECTION=mysql	DB_HOST=127.0.0.1
DB_PORT=3306	DB_DATABASE=db-name
DB_USERNAME=root	DB_PASSWORD=