

```
// create a laravel project by running the following command;
```

```
composer create-project laravel/laravel projectName
```

```
// run the project by running the following command;
```

```
Php artisan serve
```

```
// create database using phpMyAdmin / MySQL
```

```
RENAME THE DATABASE INSIDE THE. ENV FILE
```

```
→ .ENV stands for (environment specific variables for the app)
```

```
→ Php artisan migrate
```

```
//create migrations
```

```
php artisan make:migration create_students_table
```

```
// create model
```

```
php artisan make:model students
```

```
// create controller
```

```
php artisan make:controller StudentsController
```

**Inside the migration give columns inside the UP function;**

```
public function up(): void
{
    Schema::create('students', function (Blueprint $table) {
        $table->id();
        $table->string('Std_Name');
        $table->string('Std_Email');
        $table->timestamps();
    });
}
```

```
// run command php artisan migrate
```

**//Create a view by right clicking on the view folder and create a form**

```
<form action="/insert" method="post">
    @csrf
    <!-- (Cross-site Request Forgery) -->
    <input type="text" name="stdname" placeholder="Student Name">
    <input type="email" name="stdemail" placeholder="Student Email">
    <input type="submit" name="ADD RECORD">
</form>
```

## **CONTROLLER:**

### **Must import model in controller file**

```
use App\Models\students;
```

#### **insertion function:**

```
public function insert(Request $req){
    // Retrieve input values from the request
    $stdname = $req->input('stdname');
    $stdemail = $req->input('stdemail');

    // Insert the data into the students table
    students::create([
        'Std_Name' => $stdname,
        'Std_Email' => $stdemail
    ]);

    return redirect()->action([StudentsController::class,'getdata']);
}
```

#### **Fetching function:**

```
public function getdata(){
    $record = students::all();
    return view('display',[ 'record'=>$record]);
}
```

#### **Edit function:**

```
public function fetch_id($id)
{
    $editdata = students::findOrFail($id);
    return view('edit',[ 'editdata'=>$editdata]);
}
```

```
public function update_students(Request $req)
{
    $id = $req->input('id');
    $name = $req->input('stdname');
    $email = $req->input('stdemail');

    students::where('id', $id)->update(['Std_Name' => $name, 'Std_Email'
=> $email]);
}
```

```
        return redirect()->action([StudentsController::class, 'getdata']);
    }
```

## Delete Function:

```
public function del($id)
{
    students::destroy($id);
    return redirect()->action([StudentsController::class, 'getdata']);
}
```

## ROUTE

### Must import controller in web.php

```
use App\Http\Controllers\StudentsController;

Route::get('/', function () {
    return view('welcome');
});
Route::post('/insert',[StudentsController::class,'insert']);
Route::get('/display',[StudentsController::class,'getdata']);
Route::get('/edit/{id}', [StudentsController::class, 'fetch_id']);
Route::post('/update', [StudentsController::class, 'update_students']);
Route::get('/delete/{id}', [StudentsController::class, 'del']);
```

### Display.blade.php(view table)

```
<table>

<tr>
<th>Std Name:</th>
<th>Std Email:</th>
</tr>
@foreach($record as $pro)
<tr>
```

```

<td> {{$pro->Std_Name}} </td>
<td> {{$pro->Std_Email}} </td>
<td>
    <a href="/edit/{{ $pro->id }}">Edit</a>
    <a href="/delete/{{ $pro->id }}">Delete</a>
</td>
</tr>
@endforeach
</table>

```

## edit.blade.php(update form)

```

<h1>Update Data</h1>
<form action="/update" method="post">
    @csrf
    <input type="hidden" name="id" value="{{$editdata->id}}><br><br>
    <input type="text" name="stdname" value="{{$editdata-
>Std_Name}}><br><br>
    <input type="email" name="stdemail" value="{{$editdata-
>Std_Email}}><br><br>
    <input type="submit" value="Update Data">
</form>

```

## MODEL :

```

namespace App\Models;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class students extends Model
{
    use HasFactory;

    protected $fillable = [
        'Std_Name',
        'Std_Email',
    ];
}

```