Web.php:

```php
<?php

use Illuminate\Support\Facades\Route;
use Inertia\Inertia;
use App\Http\Controllers\con1;
Route::get('/', function () {
    return Inertia::render('welcome');
})->name('home');

Route::middleware(['auth', 'verified'])->group(function () {
    Route::get('dashboard', function () {
        return Inertia::render('dashboard');
    })->name('dashboard');
});

require __DIR__.'/settings.php';
require __DIR__.'/auth.php';


Route::view('productD' ,'form');
Route::post('data' ,[con1::class,'data']);

Route::view('datafile' , 'datafile');
Route::get('datafile' ,[con1::class,'getdata']);

Route::get('Trushdata/{id}' ,[con1::class,'Trushdata'])->name('Trush.data');
Route::get('restore/{id}' ,[con1::class, 'restore'])->name('restore.data');

Route::view('temp-delete' , 'temp-delete');
Route::get('temp-delete' ,[con1::class,'tempdelete']);


Route::get('deletedata/{id}' , [con1::class,'permanatedelete'])->name('per.delete');
```

Controller: con1.php: `<?php`

```php
namespace App\Http\Controllers;
use App\Models\Product;
use Illuminate\Http\Request;

class con1 extends Controller
```

```php
{
    function data(Request $req){
    $obj = new Product;
    $obj->name = $req->name;
    $obj->price = $req->price;
    $obj->description = $req->description;
    $obj->save();
    return redirect('productD');
    }


    function getdata() {
    $obj = new Product;
    $data = $obj->all();
    $finaldata = compact('data');
    return view('datafile')->with($finaldata);
    }




    function Trushdata($id){
    $obj = new Product;
    $obj->find($id)->delete();
    return redirect('datafile');

    }




    function tempdelete() {
    $data1 = Product::onlyTrashed()->get();
    $data = compact('data1');
    return view('temp-delete')->with($data);
    }






    function restore($id){

    $data1 = Product::onlyTrashed()->find($id);
    if($data1){
    $data1->restore();
```

```php
    }

    return redirect('temp-delete');
}

 function permanatedelete($id){
  $data1 = Product::onlyTrashed()->find($id);
  if($data1){

   $data1->forceDelete();

  }

    return redirect('temp-delete');

  }

}
```

Table:
```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::connection('mysql2')->create('product', function (Blueprint $table) {
            $table->id();
            $table->string('name', 100);
```

```php
        $table->float('price');
        $table->string('description');


        $table->timestamps();
    });
}

/**
 * Reverse the migrations.
 */
public function down(): void
{
    Schema::dropIfExists('product');
}
};
```

Model:
```php
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\SoftDeletes;
class Product extends Model
{
    use SoftDeletes;
    protected $table = "Product";
    protected $id = "id";


}
```

Delete_column:
```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
```

```php
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::table('Product', function (Blueprint $table) {
            $table->SoftDeletes();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::table('Product', function (Blueprint $table) {
            $table->dropSoftDeletes();
        });
    }
};
```

Veiw files:

Form.blade.php:
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>

</head>
<body>
    <form action="data" method="post">
    @csrf
    <label>
        Name:
        <input type = "text" name = "name">
</label>


<label>
        Email:
        <input type = "text" name = "price">
```

```html
            </label>
            <label>
                <div>Message</div>
                <textarea class="fixed-textarea" placeholder="Message" name = "description"></textarea>
                </label>

            <button type = "submit">SUBMIT</button>
            </form>
            </body>
            </html>
```

Datafile.blade.php:
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <table border=2>
     <tr>
    <th>id</th>
    <th>name</th>
    <th>price</th>
    <th>description</th>
    <th></th>

</tr>
@for($i = 0; $i < count($data); $i++)
    <tr>
    <td>{{ $data[$i]->id }}</td>
    <td>{{ $data[$i]->name }}</td>
    <td>{{ $data[$i]->price}}</td>
    <td>{{ $data[$i]->description}}</td>
<td>
 <a href = "{{route('Trush.data',$data[$i]->id)}}">
 <button type ="submit">Trush</button>
</a>
</td>

    @endfor
</tr>
</tabel>
```

```
</body>
</html>
```

Temp.blade.php:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <table border=2>
     <tr>
    <th>id</th>
    <th>name</th>
    <th>price</th>
    <th>description</th>
    <th></th>

</tr>
    @for($i = 0; $i < count($data1); $i++)
    <tr>
    <td>{{ $data1[$i]->id }}</td>
    <td>{{ $data1[$i]->name }}</td>
    <td>{{ $data1[$i]->price }}</td>
    <td>{{ $data1[$i]->description }}</td>
<td>
<a href = "{{route('restore.data',$data1[$i]->id)}}">
 <button type ="submit">Restore</button></a>


 <a href = "{{route('per.delete',$data1[$i]->id)}}">
 <button type ="submit">Delete</button></a>




</td>
```

```
    @endfor
</tr>
</tabel>



</body>
</html>
```

M Gmail   ▶ YouTube   📍 Maps   📰 News   🔤 Translate   🔶 Tenda WiFi   🌐 Image   🌐 https://result.msbte...

Name: [Mouse]   Price: [2000]
decription
```
fsdlkfjrklkfsdlkfxcnv,xmcnvmn
sdfsweordifsfvxc,nvmxncz,vmnz
xdnfkl;sdjaflk;asdjrfdkejfclx
zncv,mnxc,m.vnzfisd;rjsdkljfk
lxcvmcnx,m.nmd,cf.xgnvlkdjgpo
erdjflkds,fc.mnx,mvnxzm,ncsd,
mnfc,msndm,lasnm,nas,mdnz,mxn
c,mxnv,mcxnvbncxm,vnx,cmv
```
[SUBMIT]

id	name	price	description	
7	keybord	2000	dfsdfsd	Trush
8	Mouse	2000	fsdfasdf	Trush
9	speakers	500	sdfsdf	Trush
10	Headphone	1200	fdfsdf	Trush

id	name	price	description		
8	Mouse	2000	fsdfasdf	Restore	Delete
10	Headphone	1200	fdfsdf	Restore	Delete