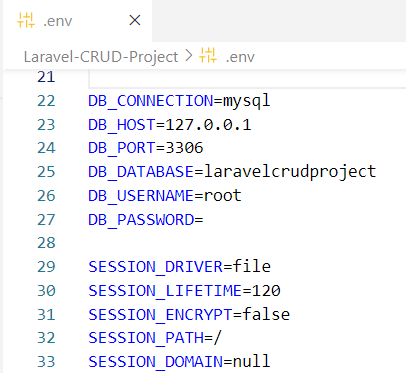
# Create project

Create a project composer create-project laravel/laravel Laravel-CRUD-Project

# Make db



# Run migration

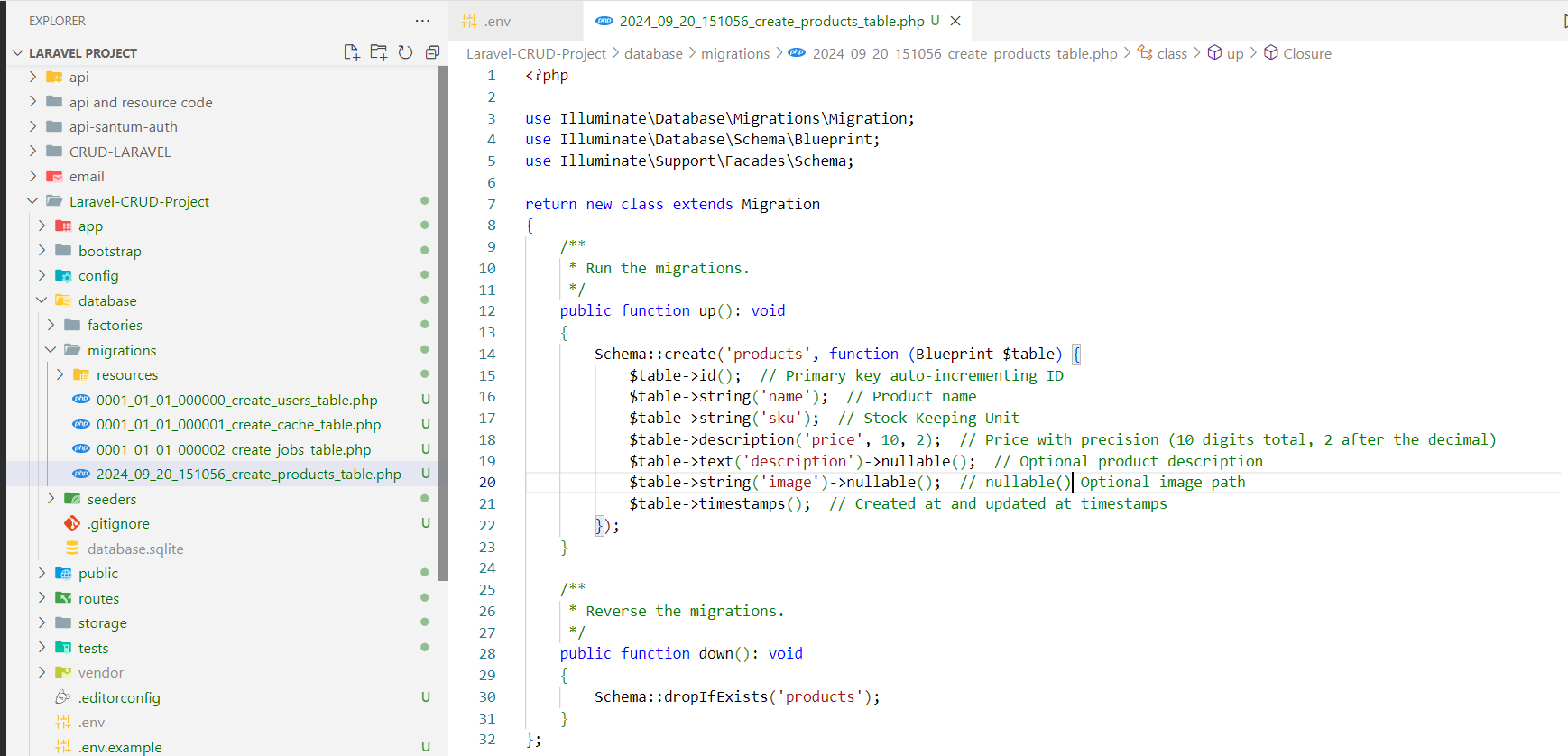
Now run command php artisan make:migration create\_products\_table

By this we can make our table

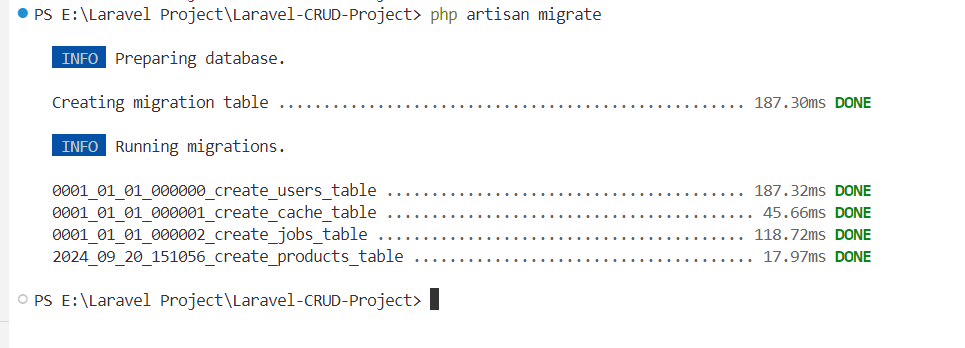


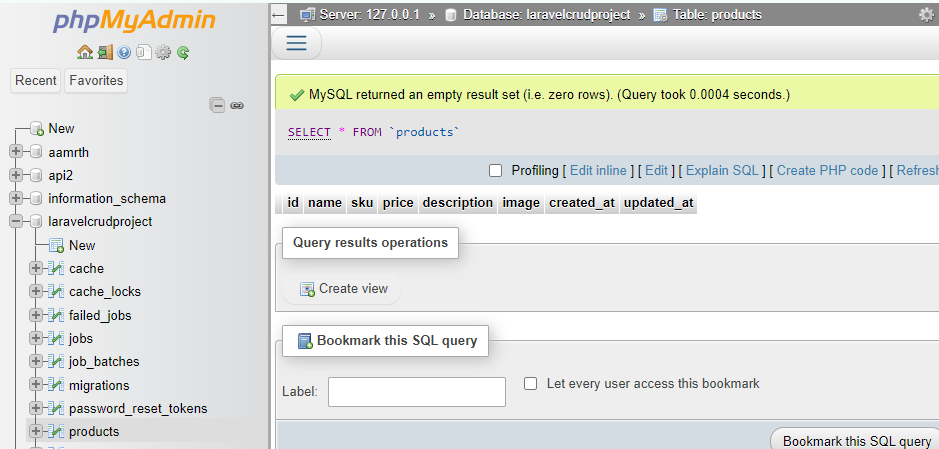
# Make product table structure

We write code in create\_products\_table



And run migrate command so the table will create on db



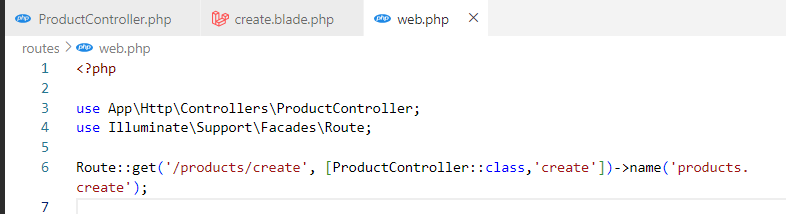


# Make controller

php artisan make:controller ProductController

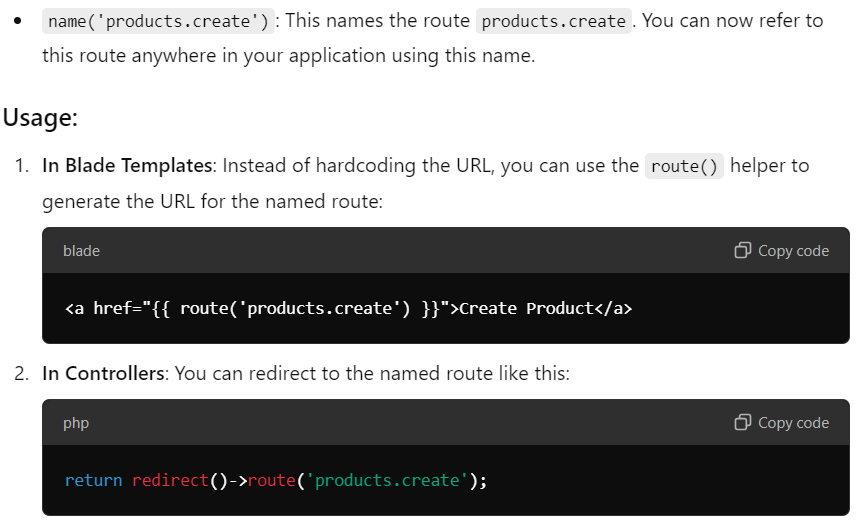
# create route

now we create a route in web.php file



**->name('products.create')**: Assigns a route name (products.create) that you can refer to in views or controllers using the route() helper function.

In Laravel, name('products.create') is used to assign a name to the route. This allows you to reference the route using the route name instead of the URL, which makes it easier to maintain your code. If you ever change the URL, you don't need to update it everywhere in your codebase—just update the route definition.

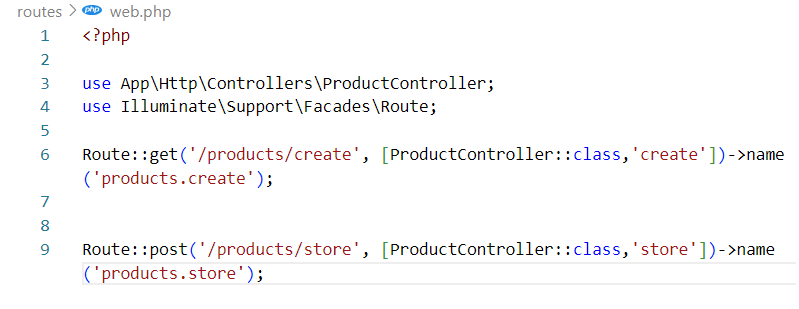


# Make form

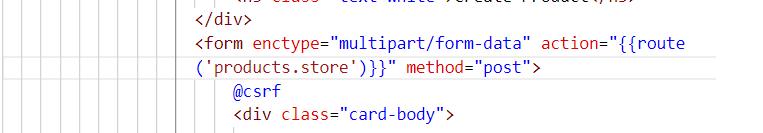


# Make products/store' route for form submit

Route::post('/ , [ProductController::class,'store'])->name('products.store');



And pass in form



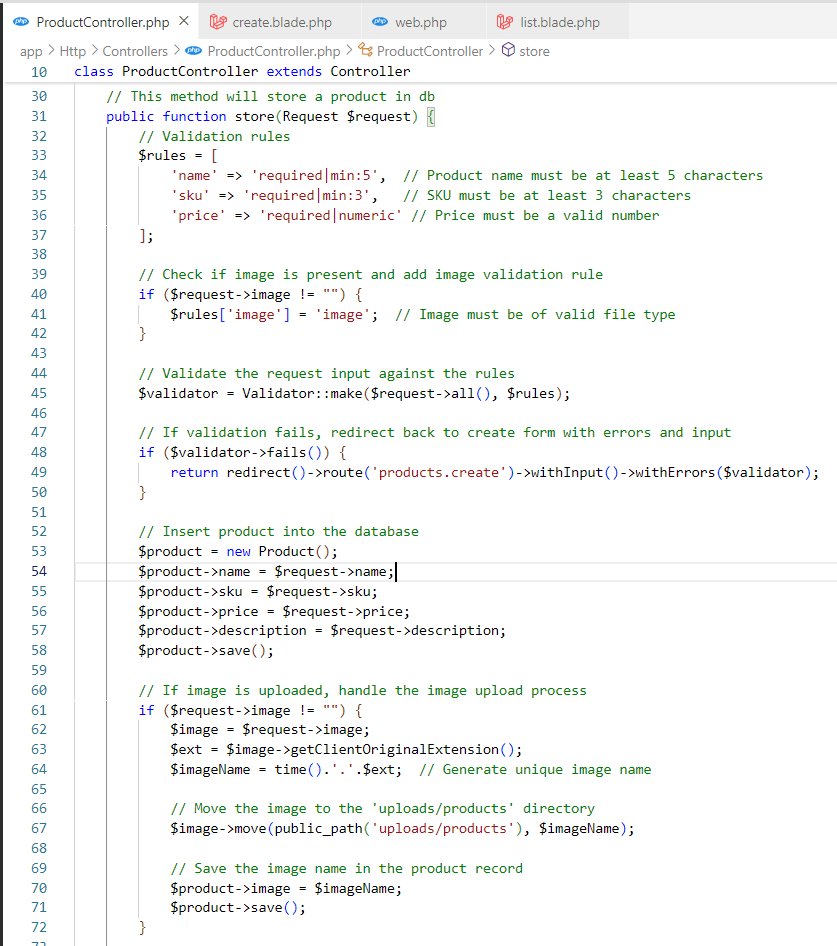
When we submit the url will go to products.store and store functions will run

# Make products model

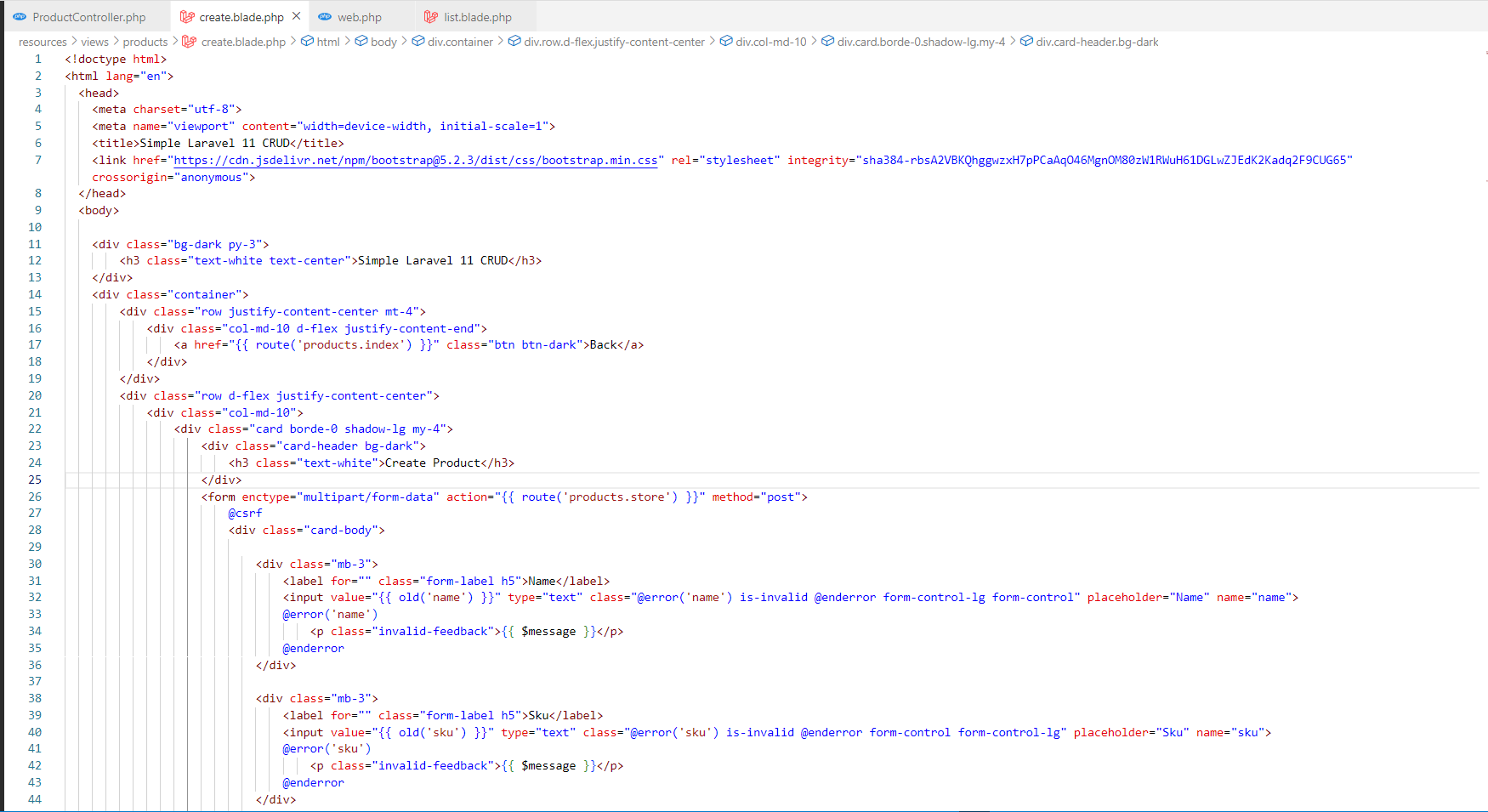
Php artisan make: model Product

# Create full code .

## Create controller



## form





### @error for validation

If you this for error messages

 @error('name')

         <p class="invalid-feedback">{{ $message }}</p>

  @enderror

 @error('sku')

         <p class="invalid-feedback">{{ $message }}</p>

  @enderror

 @error('price')

         <p class="invalid-feedback">{{ $message }}</p>

  @enderror

### value="{{ old('name') }}"

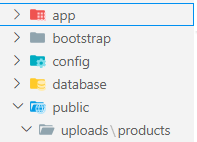
to preserve old value

when errors comes so the correct value will stay .

### enctype="multipart/form-data"

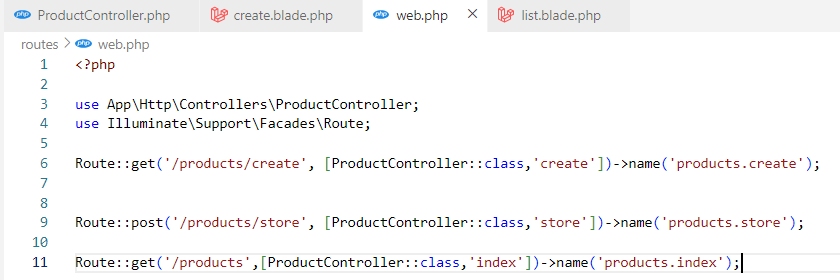
this is used for file upload

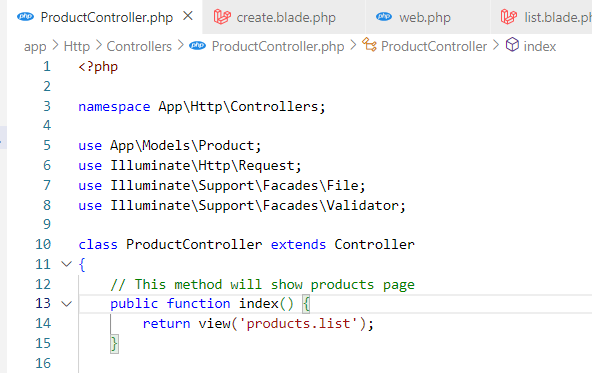
### create folders to store image

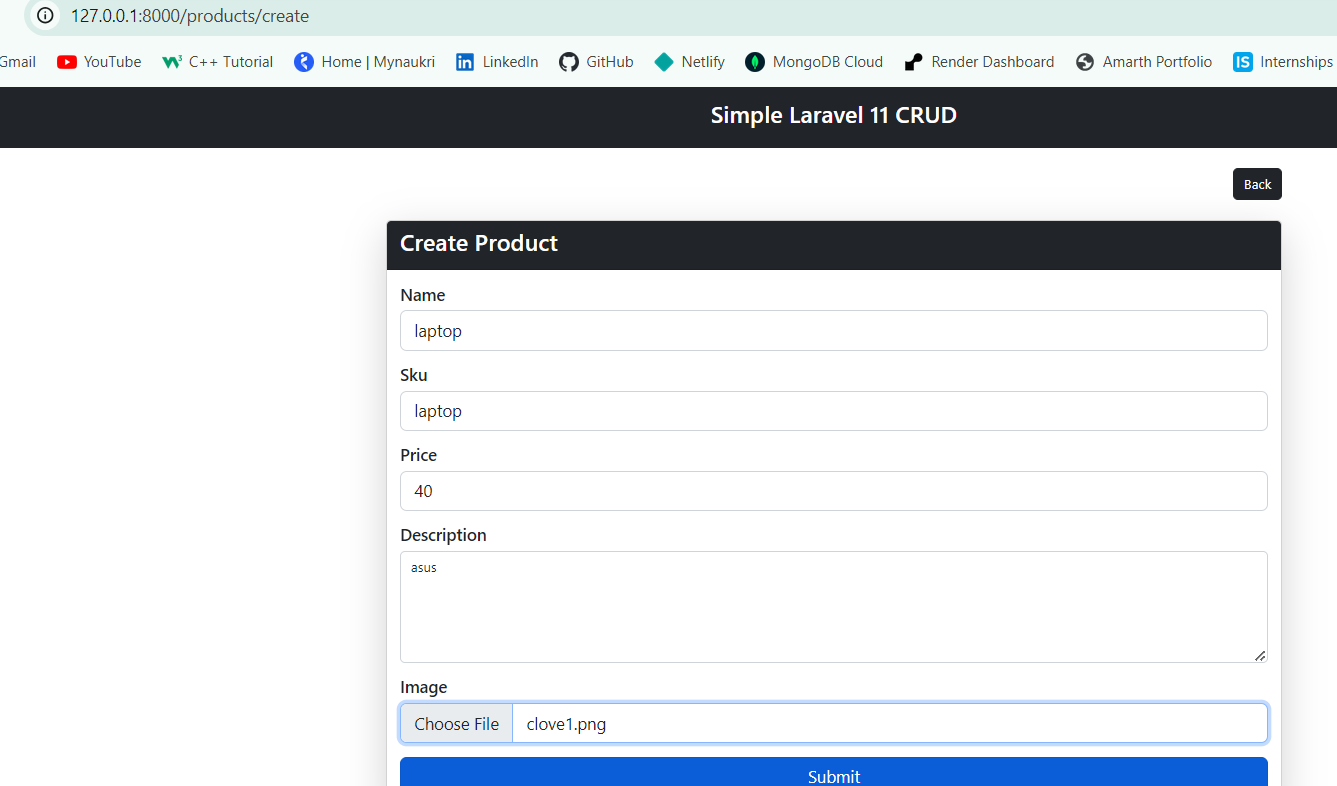


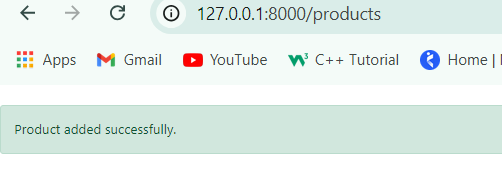
Public 🡪 uploads folder 🡪 products folder .

## Route

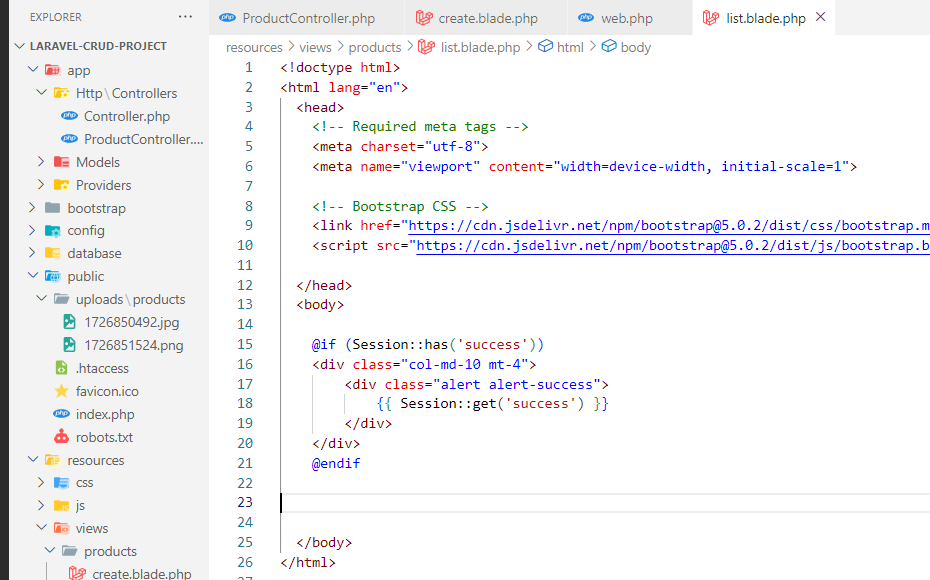




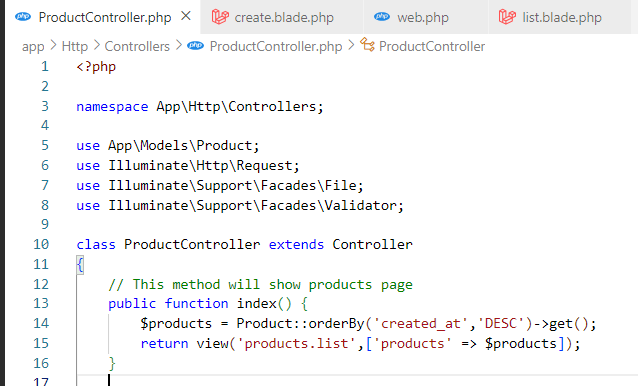


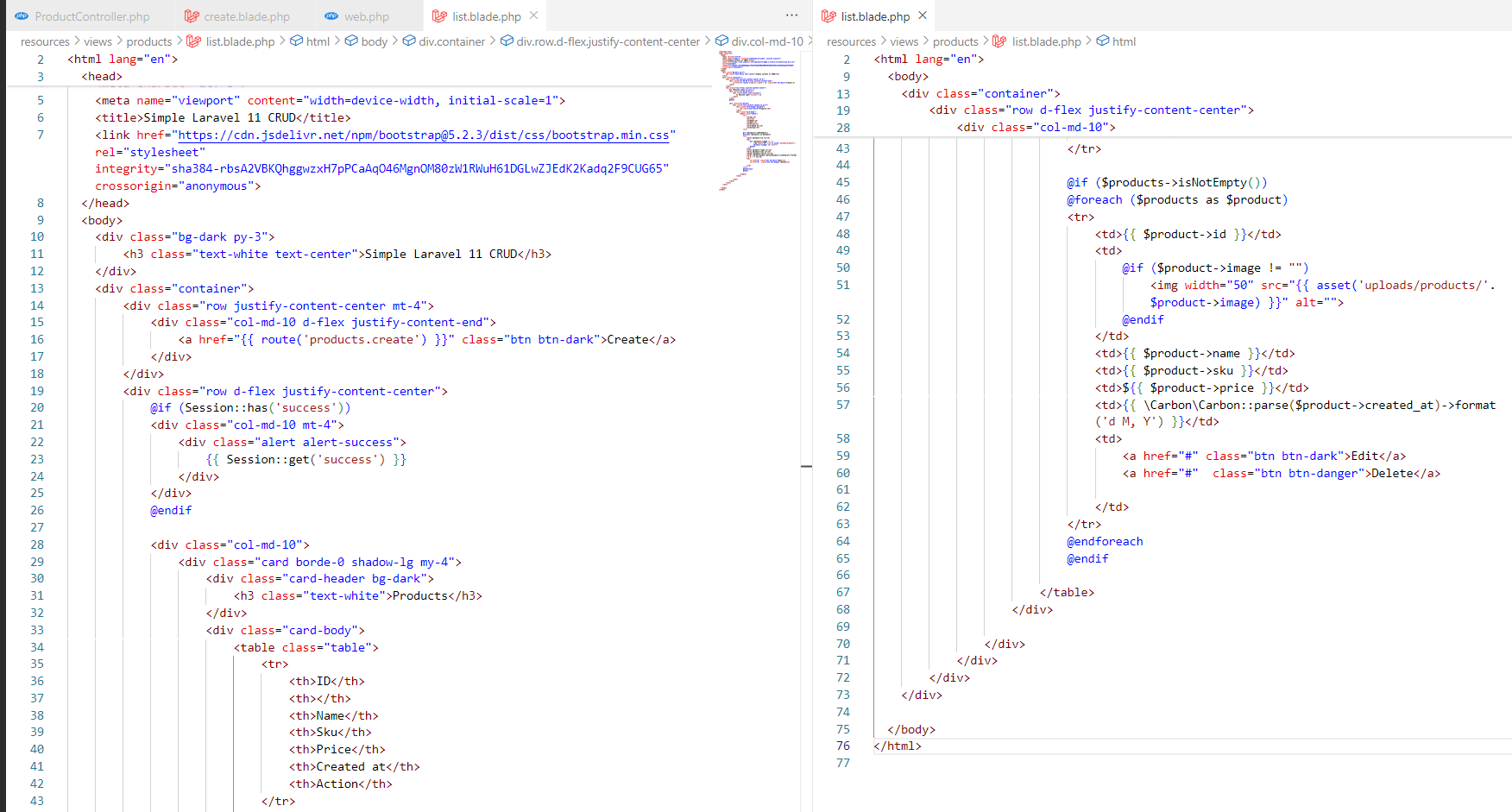


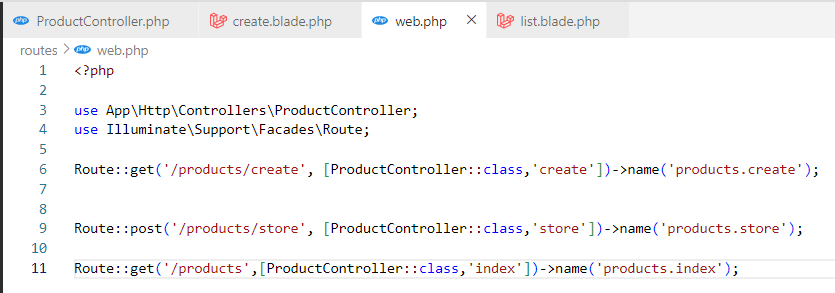
When you submit it will call products.store route and after data save it redirect to products.index

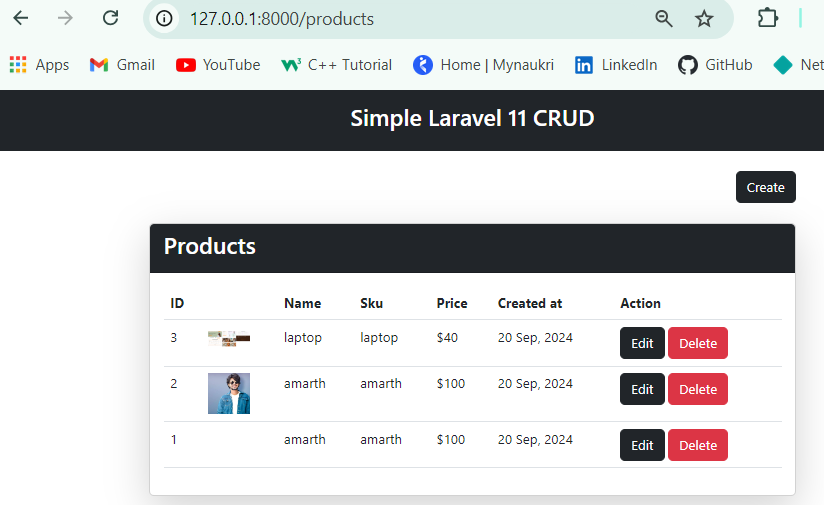


# Show List full code

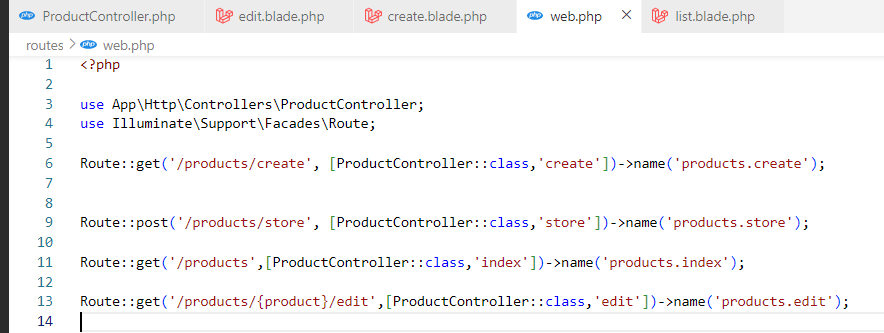


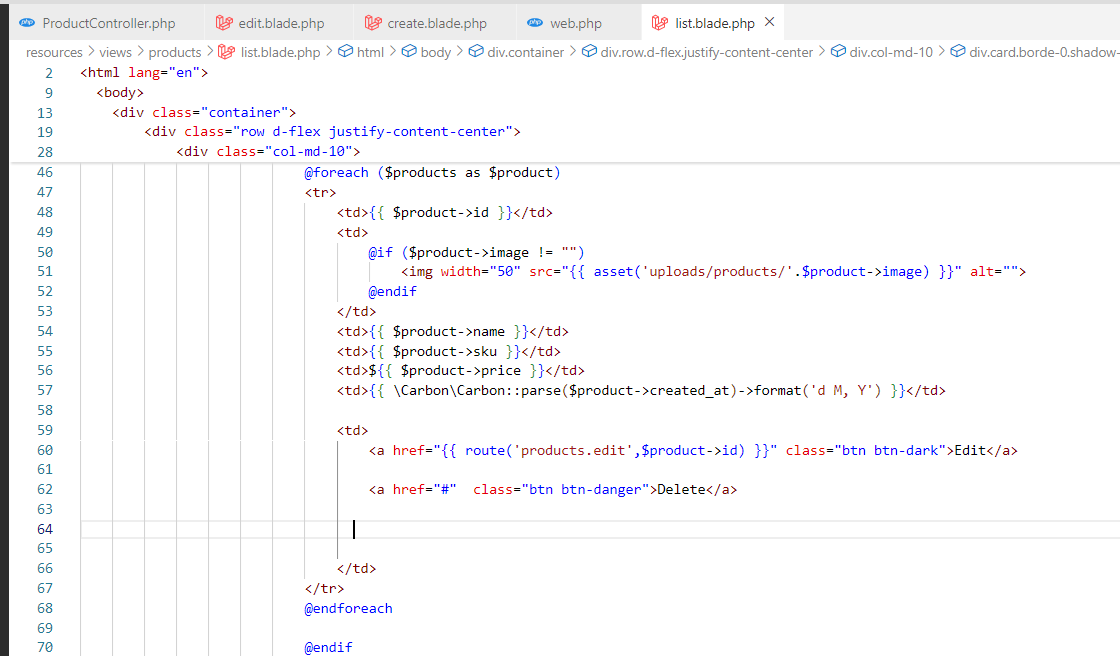


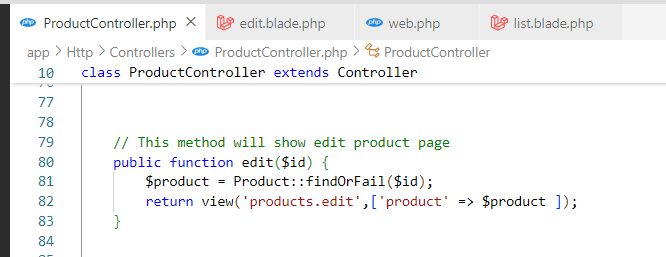


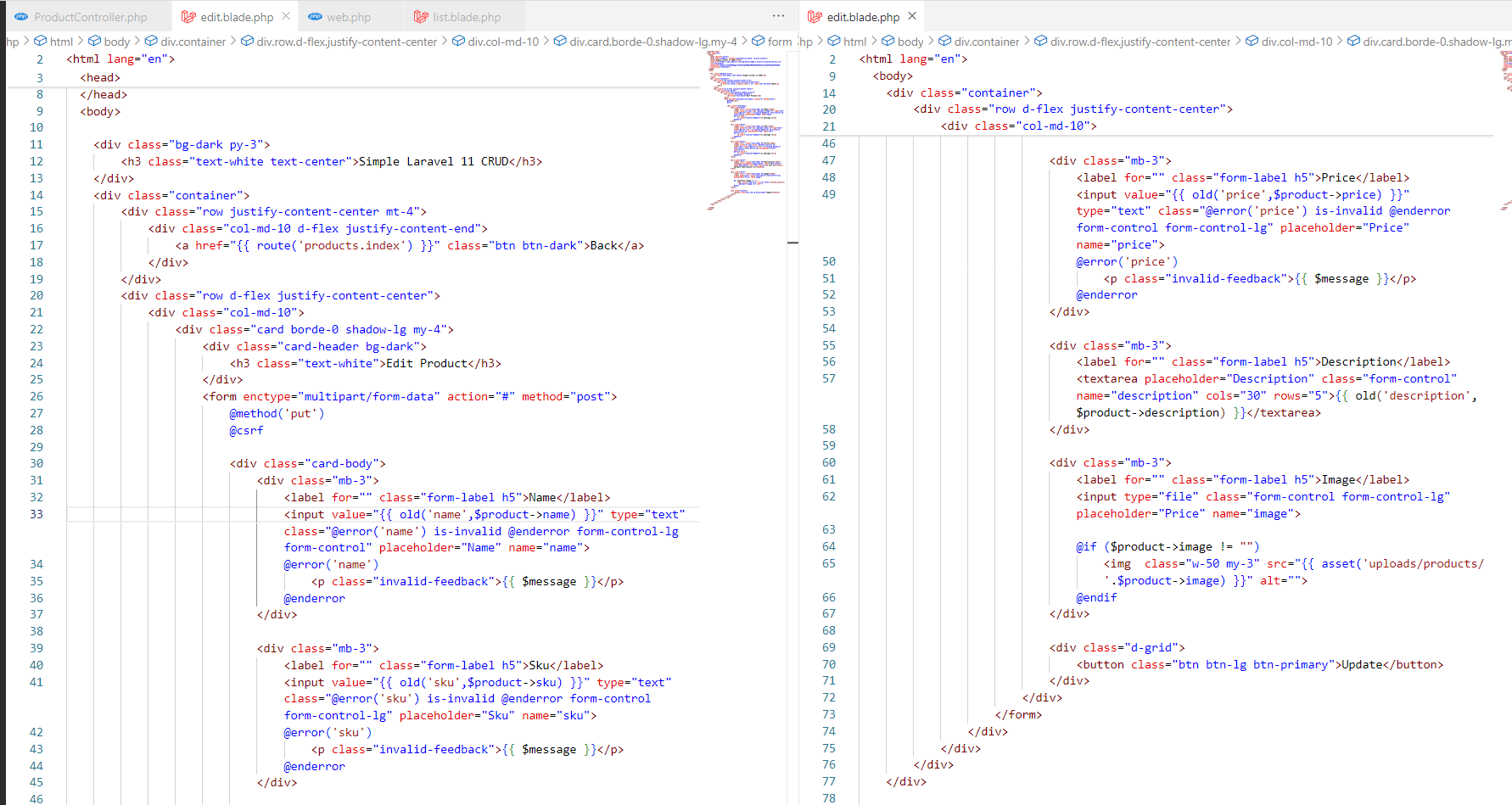


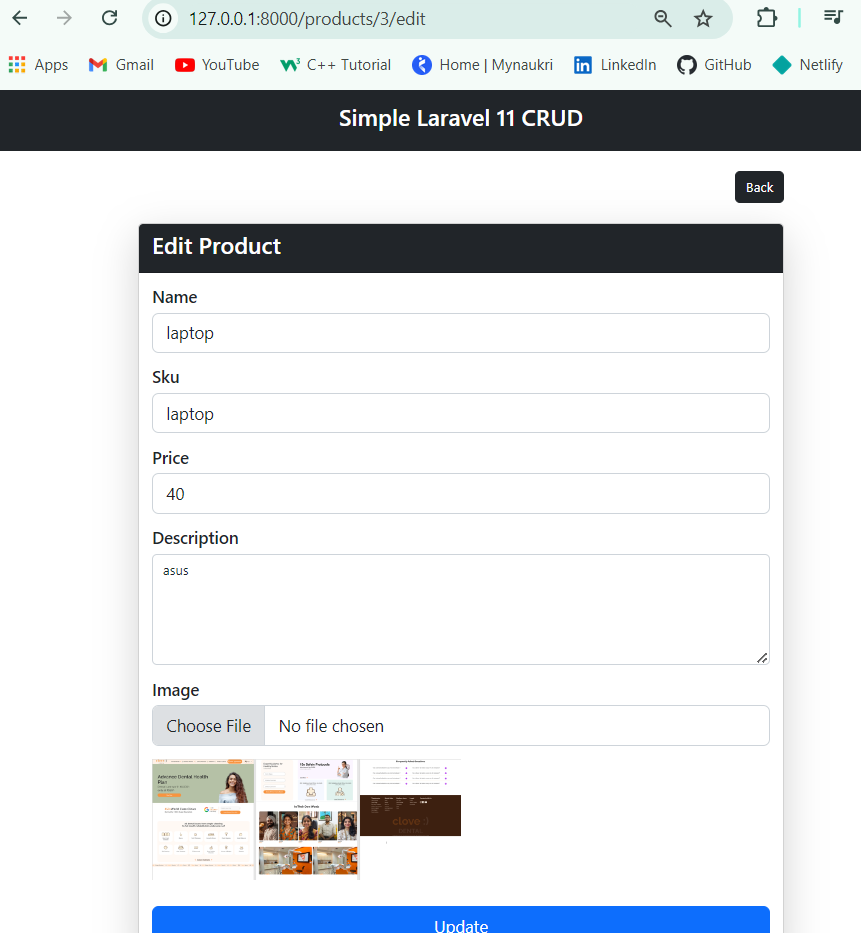
# Edit Data For Update full code



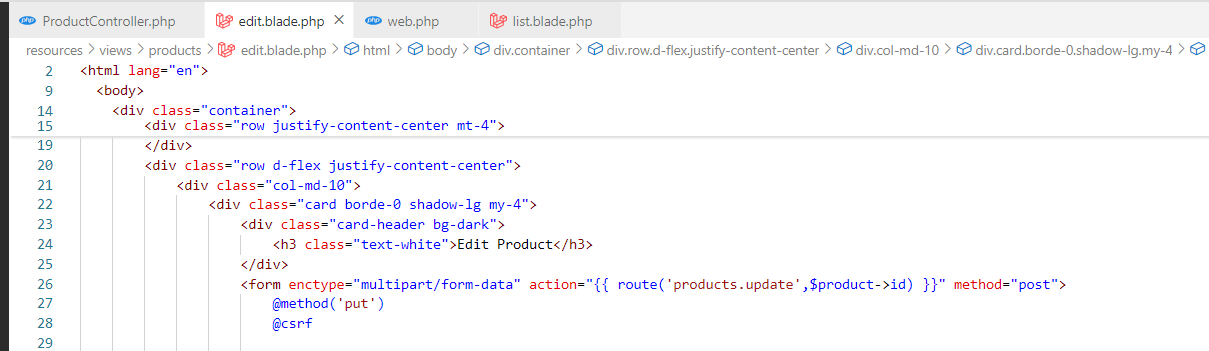


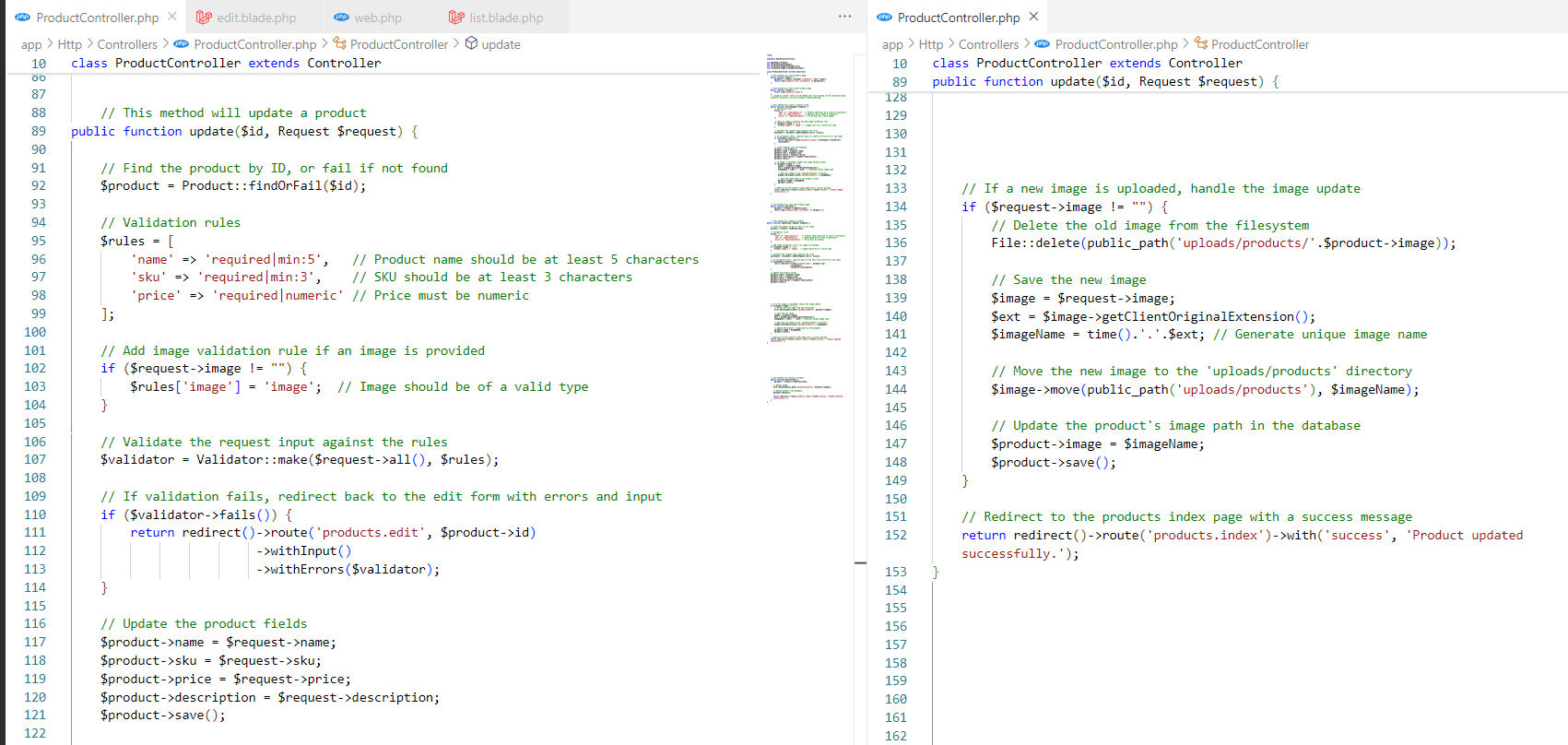






# Update Full code







# Delete

