SIDEBAR

        <li class="{{($route == 'all.products') ?  'active' : '' }}"><a href="{{route('all.products')}}"><i class="ti-more"></i>Display Products</a></li>

Route

Route::get('/add/products',[ProductsController::class,'AddProducts'])->name('all.products');

Model

    protected $guarded = [];

TABLE

  $table->id();

            $table->integer('brand\_id');

            $table->integer('category\_id');

            $table->integer('subcategory\_id');

            $table->integer('subsubcategory\_id');

            $table->string('product\_name\_en');

            $table->string('product\_name\_hin');

            $table->string('product\_slug\_en');

            $table->string('product\_slug\_hin');

            $table->string('product\_code');

            $table->string('product\_qty');

            $table->string('product\_tags\_en');

            $table->string('product\_tags\_hin');

            $table->string('product\_size\_en')->nullable();

            $table->string('product\_size\_hin')->nullable();

            $table->string('product\_color\_en');

            $table->string('product\_color\_hin');

            $table->string('selling\_price');

            $table->string('discount\_price')->nullable();

            $table->string('short\_descp\_en');

            $table->string('short\_descp\_hin');

            $table->string('long\_descp\_en');

            $table->string('long\_descp\_hin');

            $table->string('product\_thambnail');

            $table->integer('hot\_deals')->nullable();

            $table->integer('featured')->nullable();

            $table->integer('special\_offer')->nullable();

            $table->integer('special\_deals')->nullable();

            $table->integer('status')->default(0);

            $table->timestamps();

**Php artisan migrate**

MODEL2

Php artisan make:model MultiImage -m

    protected $guarded = [];

blic function up()

    {

        Schema::create('multi\_images', function (Blueprint $table) {

            $table->id();

            $table->integer('product\_id');

            $table->string('photo\_name');

            $table->timestamps();

        });

    }

*/\*\**

Php artisan migrate

Controller

  public function AddProducts()

    {

        $brand = Brand::latest()->get();

        $categories = Category::latest()->get();

*return* view('admin.product.add\_product',compact('brand','categories'));

    }

Maintain the View

[C:\xampp\htdocs\faisal\resources\views\admin\product\add\_product.blade.php](file:///C:\xampp\htdocs\faisal\resources\views\admin\product\add_product.blade.php)

For making two ck editor we need to customize the js file



*//Add text editor*

$(function () {

  "use strict";

*// Replace the <textarea id="editor1"> with a CKEditor*

*// instance, using default configuration.*

CKEDITOR.replace('editor2')

*//bootstrap WYSIHTML5 - text editor*

$('.textarea').wysihtml5();

});

How to show live image USING javascipt

Use onchange function

     <input type="file" name="product\_thambnail" class="form-control" onchange="mainThumUrl(this)">

In the same col-md use image tag and give id after error

Ajax cdn

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.3/jquery.min.js"></script>

Script

<script type="text/javascript">

    function mainThumUrl(*input*){

*if* (input.files && input.files[0]) {

            var reader = new FileReader();

            reader.onload = function(*e*){

                $('#mainThum').attr('src',e.target.result).width(80).height(80);

            };

            reader.readAsDataURL(input.files[0]);

        }

    }

</script>

Multiimage preview

First give multiple in the input tag

     <input type="file" name="multi\_img[]" multiple="" class="form-control" >

Give id to the same input

     <input type="file" name="multi\_img[]" multiple="" id="multiImg" class="form-control" >

Create a div after the @error to show these images

     <div class="row" id="preview\_image"></div>

Write the script

<script>

    $(document).ready(function(){

     $('#multiImg').on('change', function(){ *//on file input change*

*if* (window.File && window.FileReader && window.FileList && window.Blob) *//check File API supported browser*

        {

            var data = $(this)[0].files; *//this file data*

            $.each(data, function(*index*, *file*){ *//loop though each file*

*if*(/(\.|\/)(gif|jpe?g|png)*$*/i.test(file.type)){ *//check supported file type*

                    var fRead = new FileReader(); *//new filereader*

                    fRead.onload = (function(*file*){ *//trigger function on successful read*

*return* function(*e*) {

                        var img = $('<img/>').addClass('thumb').attr('src', e.target.result) .width(80)

                    .height(80); *//create image element*

                        $('#preview\_img').append(img); *//append image to output element*

                    };

                    })(file);

                    fRead.readAsDataURL(file); *//URL representing the file's data.*

                }

            });

        }*else*{

            alert("Your browser doesn't support File API!"); *//if File API is absent*

        }

     });

    });

    </script>

ADD PRODUCT TO DB

                <form route="{{products.store}}" method="post" enctype="multipart/form-data">

Route

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

ProductController

 public function ProductStore(Request $request)

    {

        $request->validate(

            [

                'brand\_id' => 'required',

                'category\_id' => 'required',

                'subcategory\_id' => 'required',

                'subsubcategory\_id' => 'required',

                'product\_name\_en' => 'required',

                'product\_name\_hin' => 'required',

                'product\_code' => 'required',

                'product\_qty' => 'required',

                'product\_tags\_en' => 'required',

                'product\_tags\_hin' => 'required',

                'product\_size\_en' => 'required',

                'product\_size\_hin' => 'required',

                'product\_color\_en' => 'required',

                'product\_color\_hin' => 'required',

                'selling\_price' => 'required',

                'discount\_price' => 'required',

                'product\_thambnail' => 'required',

                'multi\_img' => 'required',

                'short\_descp\_en' => 'required',

                'short\_descp\_hin' => 'required',

                'long\_descp\_en' => 'required',

                'long\_descp\_hindi' => 'required',

                'hot\_deals' => 'required',

                'featured' => 'required',

                'special\_offer' => 'required',

                'special\_deals' => 'required',

            ],

        );

        $image = $request->file('product\_thambnail');

        $name\_gen = hexdec(uniqid()).'.'.$image->getClientOriginalExtension();

        Image::make($image)->resize(917,1000)->save('upload/products/thambnail/'.$name\_gen);

        $save\_url = 'upload/products/thambnail/'.$name\_gen;

      Product::insert([

        'brand\_id' => $request->brand\_id,

        'category\_id' => $request->category\_id,

        'subcategory\_id' => $request->subcategory\_id,

        'subsubcategory\_id' => $request->subsubcategory\_id,

        'product\_name\_en' => $request->product\_name\_en,

        'product\_name\_hin' => $request->product\_name\_hin,

        'product\_slug\_en' =>  strtolower(str\_replace(' ', '-', $request->product\_name\_en)),

        'product\_slug\_hin' => str\_replace(' ', '-', $request->product\_name\_hin),

        'product\_code' => $request->product\_code,

        'product\_qty' => $request->product\_qty,

        'product\_tags\_en' => $request->product\_tags\_en,

        'product\_tags\_hin' => $request->product\_tags\_hin,

        'product\_size\_en' => $request->product\_size\_en,

        'product\_size\_hin' => $request->product\_size\_hin,

        'product\_color\_en' => $request->product\_color\_en,

        'product\_color\_hin' => $request->product\_color\_hin,

        'selling\_price' => $request->selling\_price,

        'discount\_price' => $request->discount\_price,

        'short\_descp\_en' => $request->short\_descp\_en,

        'short\_descp\_hin' => $request->short\_descp\_hin,

        'long\_descp\_en' => $request->long\_descp\_en,

        'long\_descp\_hin' => $request->long\_descp\_hin,

        'hot\_deals' => $request->hot\_deals,

        'featured' => $request->featured,

        'special\_offer' => $request->special\_offer,

        'special\_deals' => $request->special\_deals,

        'product\_thambnail' => $save\_url,

        'status' => 1,

        'created\_at' => Carbon::now(),

      ]);

    }

//working code

  $image = $request->file('product\_thambnail');

        $name\_gen = hexdec(uniqid()).'.'.$image->getClientOriginalExtension();

        Image::make($image)->resize(917,1000)->save('upload/products/thambnail/'.$name\_gen);

        $save\_url = 'upload/products/thambnail/'.$name\_gen;

        $product\_id = Product::insertGetId([

        'brand\_id' => $request->brand\_id,

        'category\_id' => $request->category\_id,

        'subcategory\_id' => $request->subcategory\_id,

        'subsubcategory\_id' => $request->subsubcategory\_id,

        'product\_name\_en' => $request->product\_name\_en,

        'product\_name\_hin' => $request->product\_name\_hin,

        'product\_slug\_en' =>  strtolower(str\_replace(' ', '-', $request->product\_name\_en)),

        'product\_slug\_hin' => str\_replace(' ', '-', $request->product\_name\_hin),

        'product\_code' => $request->product\_code,

        'product\_qty' => $request->product\_qty,

        'product\_tags\_en' => $request->product\_tags\_en,

        'product\_tags\_hin' => $request->product\_tags\_hin,

        'product\_size\_en' => $request->product\_size\_en,

        'product\_size\_hin' => $request->product\_size\_hin,

        'product\_color\_en' => $request->product\_color\_en,

        'product\_color\_hin' => $request->product\_color\_hin,

        'selling\_price' => $request->selling\_price,

        'discount\_price' => $request->discount\_price,

        'short\_descp\_en' => $request->short\_descp\_en,

        'short\_descp\_hin' => $request->short\_descp\_hin,

        'long\_descp\_en' => $request->long\_descp\_en,

        'long\_descp\_hin' => $request->long\_descp\_hindi,

        'hot\_deals' => $request->hot\_deals,

        'featured' => $request->featured,

        'special\_offer' => $request->special\_offer,

        'special\_deals' => $request->special\_deals,

        'product\_thambnail' => $save\_url,

        'status' => 1,

        'created\_at' => Carbon::now(),

      ]);

*////////// Multiple Image Upload Start ///////////*

        $images = $request->file('multi\_img');

*foreach* ($images as $img)

        {

            $make\_name = hexdec(uniqid()).'.'.$img->getClientOriginalExtension();

            Image::make($img)->resize(917,1000)->save('upload/products/multi\_image/'.$make\_name);

            $uploadPath = 'upload/products/multi\_image/'.$make\_name;

          MultiImage::insert([

              'product\_id' => $product\_id,

              'photo\_name' => $uploadPath,

              'created\_at' => Carbon::now(),

          ]);

        }

*////////// Een Multiple Image Upload Start ///////////*

        $notification = array(

            'message' => 'Product Inserted Successfully',

            'alert-type' => 'success'

        );

*return* redirect()->back()->with($notification);