**The Code Learners**

Helpline for Software Development

Home Python Python Programs and Exercises Django Framework Javascript PHP Laravel Laravel LiveWire

**Search Here**

Search

Search

LARAVEL DRAG, DROP AND SORT
CATEGORIES SUBCATEGORIES USING JQUERY
thecodelearners.com

# Laravel Categories SubCategories Drag, Drop and Sort using JQuery

In this tutorial, You'll learn to create Drag, Drop and Sort Categories and SubCategories in Laravel using JQuery Library. Using this method you will be able to add unlimited nested SubCategories, loop through each category recursively and display in the template.

> **Note**
>
> Click here to skip content and directly jump to complete source codes section.

## Create Categories Table Migration

The first step is to create a migration class which contains table schema information which than Laravel uses to create a table in the database.

```
php artisan make:migration create_categories_table --create=catego
```

The table **categories** contain three columns **name**, **parent_id** and **sort_order** and **category_id** as primary key with three timestamp columns which are utilized by Laravel.

The column **parent_id**: In mapped to categories table primary key. And if its value is 0 then it is identified as the main parent category.
The **sort_order**: Represents arrangement of categories on after the other.

Next, migrate this class by executing below command in your terminal.

```
php artisan mirgate --path=database/migrations/2020_05_28_06405
```

Then switch your MySQL database you'll find category table over there.

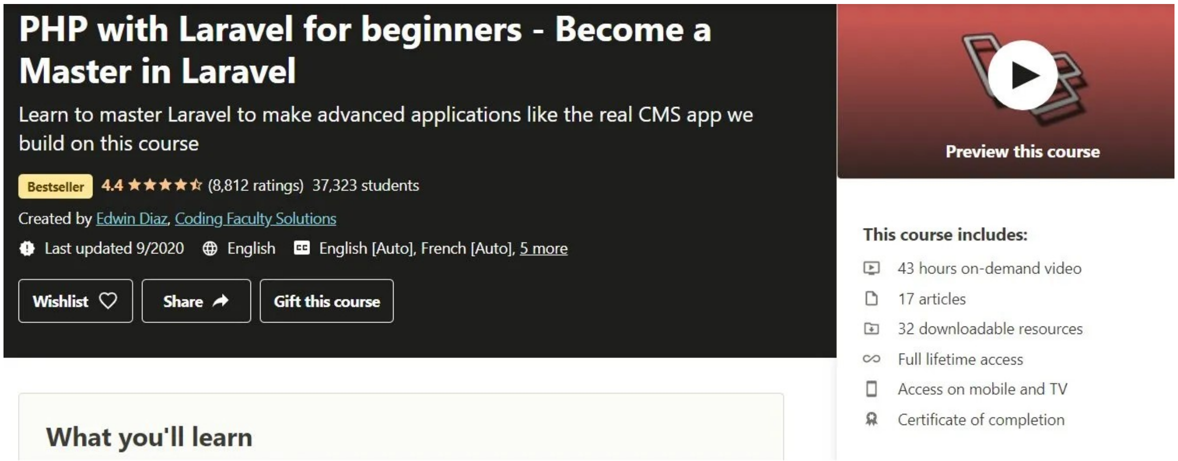
## Category Model and setting up a parent-child relationship

You can create a model for **categories** table using **php artisan make:model CategoryModel**.

In **CategoryModel** you first need to describe table columns and has the category have only one relationship that to only with itself than in this case the **hasMany** relationship is mapped to the same model (i.e CategoryModel) below is a snippet.

```
public function categories()
{
    return $this->hasMany(CategoryModel::class, 'parent_id', 'categor
}
```

The relationship method **categories** will return all child categories of than category or empty in case if a category has no child category.



PHP with Laravel for beginners - Become a Master in Laravel

Learn to master Laravel to make advanced applications like the real CMS app we build on this course

Bestseller 4.4 ★★★★☆ (8,812 ratings) 37,323 students

Created by Edwin Diaz, Coding Faculty Solutions

Last updated 9/2020 English English [Auto], French [Auto], 5 more

This course includes:

- 43 hours on-demand video
- 17 articles
- 32 downloadable resources
- Full lifetime access
- Access on mobile and TV
- Certificate of completion

What you'll learn

Udemy Laravel Course for Beginners

## Displaying Nested Categories and SubCategories in Template

First, you only have to select parent categories and then loop then inside template like below example.
Getting only parent categories in **CategoryController.php**.

```
$info = [
    'categories' => CategoryModel::where(['parent_id' => 0])->orderE
];

return view('category-subcategory.list', $info);
```

In template **category-subcategory.list** loop through all parent categories.

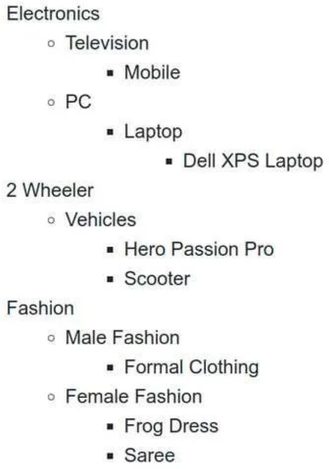
```
<ol>
    @foreach($categories as $k => $category)
        {{ $category['name'] }}

        <!-- If parent category has child categories then below templat
        @if(!empty($category->categories))
            @include('category-subcategory.child-includes.child-categor
        @endif
    @endforeach

</ol>
```

On each iteration of **$category** a conditional statement is used for checking whether **$category** has any child categories by calling **$category->categories**.

For recursively displaying subcategories call the sub-template **child-category-view** it’s snippet shown below.

```
@if(!empty($category->categories))
    <ol>
        @foreach($category->categories as $kk => $sub_category)
            <li>
                {{ $sub_category['name'] }}

                <!-- This will be recusively call itself and display subcatego
                @include('category-subcategory.child-includes.child-categ
            </li>
        @endforeach
    </ol>
@endif
```

The output will be like below.



```
Electronics
    ◦ Television
        ▪ Mobile
    ◦ PC
        ▪ Laptop
            ▪ Dell XPS Laptop
2 Wheeler
    ◦ Vehicles
        ▪ Hero Passion Pro
        ▪ Scooter
Fashion
    ◦ Male Fashion
        ▪ Formal Clothing
    ◦ Female Fashion
        ▪ Frog Dress
        ▪ Saree
```

## Saving Order of Categories after Drag and Drop Action

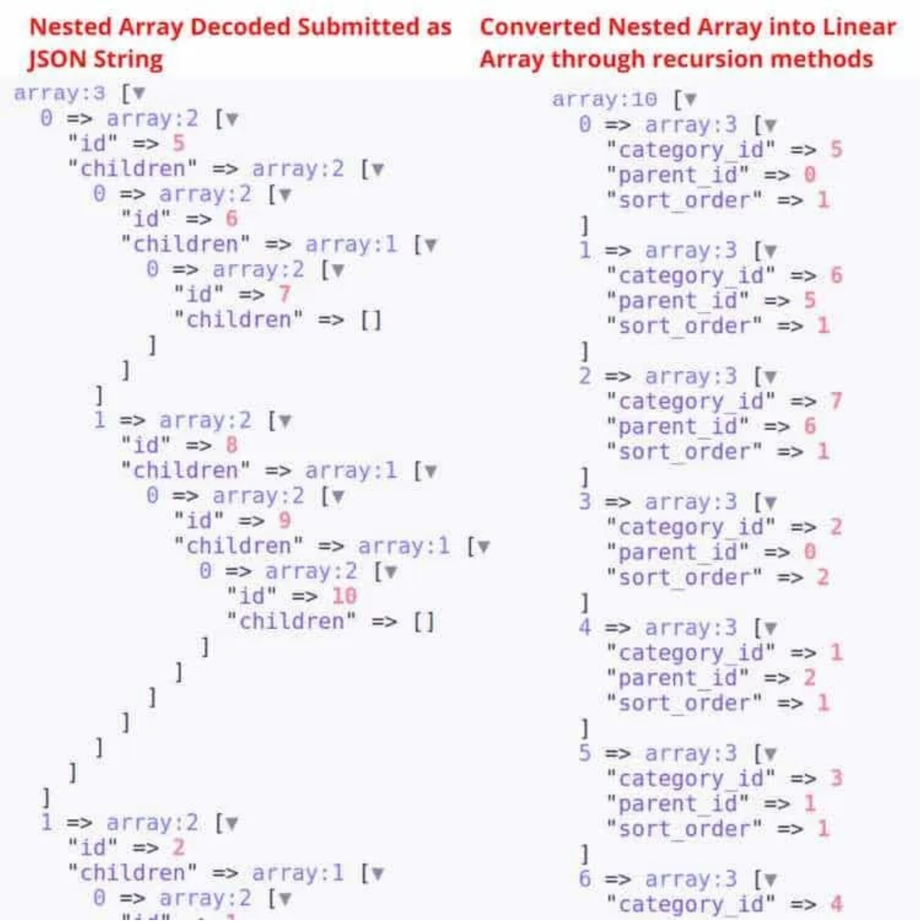
For Drag and Drop we have used dbushell/Nestable example and javascript logic.

After each drag and drop event, the Javascript function will generate a multi-dimensional nested array further converted to JSON string is submitted as post data to controller method **saveNestedCategories(Request $request)**.
This **JSON** string is decoded into an array and converted from nested array to linear array.
In **CategoryController**.

```
$json = $request->nested_category_array;
```

```
$decoded_json = json_decode($json, TRUE);

$simplified_list = [];
$this->recur1($decoded_json, $simplified_list);

public function recur1($nested_array=[], &$simplified_list=[]){

    foreach($nested_array as $k => $v){

        $sort_order = $k+1;
        $simplified_list[] = [
            "category_id" => $v['id'],
            "parent_id" => 0,
            "sort_order" => $sort_order
        ];

        if(!empty($v["children"])){
            $this->recur2($v['children'], $simplified_list, $v['id']);
        }

    }
}

public function recur2($sub_nested_array=[], &$simplified_list=[], $p
    foreach($sub_nested_array as $k => $v){

        $sort_order = $k+1;
        $simplified_list[] = [
            "category_id" => $v['id'],
            "parent_id" => $parent_id,
            "sort_order" => $sort_order
        ];

        if(!empty($v["children"])){
            return $this->recur2($v['children'], $simplified_list, $v['id']);
        }
    }
}
```

There are two methods **recur1()** and **recur2()** converts nested array into a linear array and that linear array contains **category_id**, **parent_id** and **sort_order** which will be saved into the database.



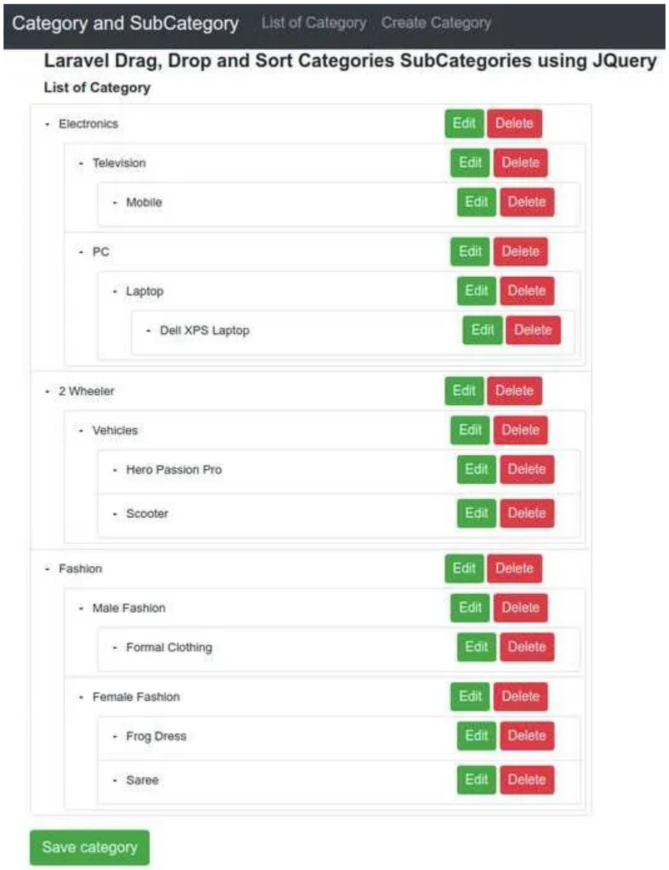
**Nested Array Decoded Submitted as JSON String**

```
array:3 [▼
  0 => array:2 [▼
    "id" => 5
    "children" => array:2 [▼
      0 => array:2 [▼
        "id" => 6
        "children" => array:1 [▼
          0 => array:2 [▼
            "id" => 7
            "children" => []
          ]
        ]
      ]
      1 => array:2 [▼
        "id" => 8
        "children" => array:1 [▼
          0 => array:2 [▼
            "id" => 9
            "children" => array:1 [▼
              0 => array:2 [▼
                "id" => 10
                "children" => []
              ]
            ]
          ]
        ]
      ]
    ]
  ]
  1 => array:2 [▼
    "id" => 2
    "children" => array:1 [▼
      0 => array:2 [▼
```

**Converted Nested Array into Linear Array through recursion methods**

```
array:10 [▼
  0 => array:3 [▼
    "category_id" => 5
    "parent_id" => 0
    "sort_order" => 1
  ]
  1 => array:3 [▼
    "category_id" => 6
    "parent_id" => 5
    "sort_order" => 1
  ]
  2 => array:3 [▼
    "category_id" => 7
    "parent_id" => 6
    "sort_order" => 1
  ]
  3 => array:3 [▼
    "category_id" => 2
    "parent_id" => 0
    "sort_order" => 2
  ]
  4 => array:3 [▼
    "category_id" => 1
    "parent_id" => 2
    "sort_order" => 1
  ]
  5 => array:3 [▼
    "category_id" => 3
    "parent_id" => 1
    "sort_order" => 1
  ]
  6 => array:3 [▼
    "category_id" => 4
```

## Final Output Preview

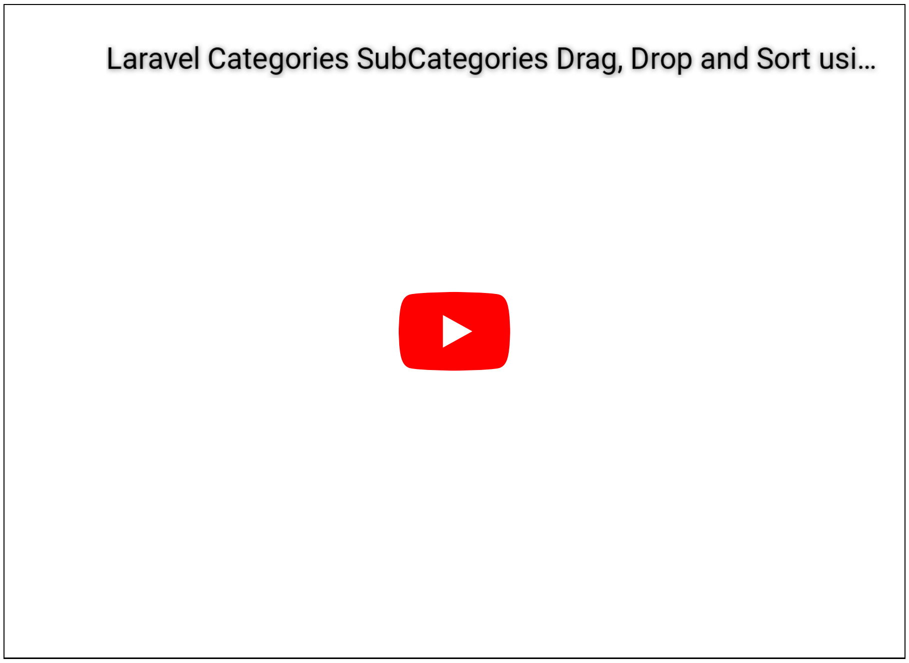


**Laravel Drag, Drop and Sort Categories SubCategories using JQuery**

**List of Category**

- Electronics
  - Television
    - Mobile
  - PC
    - Laptop
      - Dell XPS Laptop
- 2 Wheeler
  - Vehicles
    - Hero Passion Pro
    - Scooter
- Fashion
  - Male Fashion
    - Formal Clothing
  - Female Fashion
    - Frog Dress
    - Saree

## Video Preview



## Complete Source Codes:

**Categories Migration Class**

In **database/migrations /2020_05_28_064056_create_categories_table.php**.

```
<?php

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

class CreateCategoriesTable extends Migration
{
    /**
        * Run the migrations.
        *
        * @return void
        */
    public function up()
    {
        Schema::create('categories', function (Blueprint $table) {
            $table->bigIncrements('category_id');
            $table->string('name', 255);
            $table->bigInteger('parent_id')->default(0);
            $table->integer('sort_order')->default(0);
            $table->timestamps();
            $table->softDeletes();
        });
    }

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

### Category Model

In **app/CategoryModel.php**.

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;
use Illuminate\Support\Facades\Log;

class CategoryModel extends Model
```

```
{
    protected $table="categories";
    protected $primaryKey = 'category_id';
    protected $fillable = ['name', 'parent_id', 'sort_order'];


    public function categories()
    {
        return $this->hasMany(CategoryModel::class, 'parent_id', 'categ
    }
}
```

### Web Routes

In **routes/web.php**.

```
// Category and Subcategory Routes
Route::get('category-subcategory/list', 'CategoryController@index')-
Route::post('category-subcategory/save-nested-categories', 'Categor

Route::get('category-subcategory/create', 'CategoryController@creat
Route::post('category-subcategory/save', 'CategoryController@store
Route::get('category-subcategory/edit/{category_id}', 'CategoryContr
Route::get('category-subcategory/remove/{category_id}', 'CategoryC
```

### Category Controller

In **app/Http/Controllers/CategoryController.php**.

```
<?php

namespace App\Http\Controllers;


use Illuminate\Http\Request;
use Illuminate\Support\Facades\Validator;
use App\CategoryModel;
use DB;

class CategoryController extends Controller
{

    public function index(Request $request){
        $info = [
            'categories' => CategoryModel::where(['parent_id' => 0])->o
        ];

        return view('category-subcategory.list', $info);
```

```
}

public function saveNestedCategories(Request $request){

    $json = $request->nested_category_array;
    $decoded_json = json_decode($json, TRUE);

    $simplified_list = [];
    $this->recur1($decoded_json, $simplified_list);

    DB::beginTransaction();
    try {
        $info = [
            "success" => FALSE,
        ];

        foreach($simplified_list as $k => $v){
            $category = CategoryModel::find($v['category_id']);
            $category->fill([
                "parent_id" => $v['parent_id'],
                "sort_order" => $v['sort_order'],
            ]);

            $category->save();
        }

        DB::commit();
        $info['success'] = TRUE;
    } catch (\Exception $e) {
        DB::rollback();
        $info['success'] = FALSE;
    }

    if($info['success']){
        $request->session()->flash('success', "All Categories updated
    }else{
        $request->session()->flash('error', "Something went wrong w
    }

    return redirect(route('category-subcategory.list'));
}

public function recur1($nested_array=[], &$simplified_list=[]){

    static $counter = 0;

    foreach($nested_array as $k => $v){
        $sort_order = $k+1;
        $simplified_list[] = [
            "category_id" => $v['id'],
            "parent_id" => 0,
            "sort_order" => $sort_order
        ];

        if(!empty($v["children"])){
            $counter+=1;
            $this->recur2($v['children'], $simplified_list, $v['id']);
        }

    }
}

public function recur2($sub_nested_array=[], &$simplified_list=[],

    static $counter = 0;

    foreach($sub_nested_array as $k => $v){

        $sort_order = $k+1;
        $simplified_list[] = [
            "category_id" => $v['id'],
            "parent_id" => $parent_id,
            "sort_order" => $sort_order
        ];

        if(!empty($v["children"])){
            $counter+=1;
            return $this->recur2($v['children'], $simplified_list, $v['id']);
        }
    }
}

public function create(Request $request){
    $info = [
        'categories' => CategoryModel::orderBy('name', 'ASC')->get(
    ];

    return view('category-subcategory.create', $info);
}

public function store(Request $request){
    $rules=[
        'name' => 'required',
        'parent_id' => 'nullable',
    ];
    $messages = [
        "name.required" => "Category name is required.",
    ];

    $validator = Validator::make($request->all(), $rules, $messages)

    if( $validator->fails() ){
        return back()->withErrors($validator)->withInput();
    } else {
        DB::beginTransaction();
        try {
            $info = [
                "success" => FALSE,
            ];
            $query = [
                'name' => $request->name,
                'parent_id' => (!empty($request->parent_id))? $request-
            ];

            $category = CategoryModel::updateOrCreate(['category_i

            DB::commit();
            $info['success'] = TRUE;
        } catch (\Exception $e) {
            DB::rollback();
            $info['success'] = FALSE;
        }

        if(!$info['success']){
            return redirect(route('category-subcategory.create'))->with
        }

        return redirect(route('category-subcategory.edit', ['category_i
    }
}

public function edit(Request $request){
    $info = [
        'categories' => CategoryModel::orderBy('name', 'ASC')->get(
        'category' => CategoryModel::find($request->category_id),
    ];

    return view('category-subcategory.create', $info);
}

public function remove(Request $request){
    $category = CategoryModel::find($request->category_id);
```

```
        $category->delete();

        return redirect(route('category-subcategory.list'))->with('succes
    }
}
```

### List Categories and SubCategories Template

In **resources/views/category-subcategory/list.blade.php**.

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-
        <title>Category Form</title>
        <link rel="stylesheet" href="{{ url('category-subcategory-assets
        <link rel="stylesheet" href="{{ url('category-subcategory-assets
    </head>
    <body>

        @include('category-subcategory.includes.menu')

        <div class="container">

            @include('category-subcategory.includes.notification')

            <h2 class="text-center">Laravel Drag, Drop and Sort Catego

            <h4> List of Category</h4>

            <div class="row">
                <div class="col-md-12 dd" id="nestable-wrapper">
                    <ol class="dd-list list-group">
                        @foreach($categories as $k => $category)
                            <li class="dd-item list-group-item" data-id="{{ $ca
                                <div class="dd-handle" >{{ $category['name'] }}
                                <div class="dd-option-handle">
                                    <a href="{{ route('category-subcategory.edit',
                                    <a href="{{ route('category-subcategory.remo
                                </div>

                                @if(!empty($category->categories))
                                    @include('category-subcategory.child-include
                                @endif
                            </li>
                        @endforeach
                    </ol>
                </div>
            </div>

            <div class="row">
                <form action="{{ route('category-subcategory.save-nested
                    @csrf
                    <textarea style="display: none;" name="nested_categor
                    <button type="submit" class="btn btn-success" style="
                </form>
            </div>


        </div>


        <script src="{{ url('category-subcategory-assets/js/jquery-3.5.1.
        <script src="{{ url('category-subcategory-assets/js/bootstrap.mi
        <script src="{{ url('category-subcategory-assets/js/popper.min.j


        <script src="{{ url('category-subcategory-assets/js/jquery.nestal

        <script src="{{ url('category-subcategory-assets/js/style.js') }}">
    </body>
</html>
```

### Child Categories Template

## In **resources/views/category-subcategory/child-includes/child-category-view.blade.php**.

```
@if(!empty($category->categories))
    <ol class="dd-list list-group">
        @foreach($category->categories as $kk => $sub_category)
            <li class="dd-item list-group-item" data-id="{{ $sub_categor
                <div class="dd-handle" >{{ $sub_category['name'] }}</div>
                <div class="dd-option-handle">
                    <a href="{{ route('category-subcategory.edit', ['category
                    <a href="{{ route('category-subcategory.remove', ['categ
                </div>

                @include('category-subcategory.child-includes.child-categ
            </li>
        @endforeach
    </ol>
@endif
```

### CSS File for List Styling

In **public/category-subcategory-assets/css/style.css**.

```
html {
    margin: 0;
    padding: 0;
}

body {
    font-size: 100%;
    margin: 0;
    padding: 0;
    font-family: 'Helvetica Neue', Arial, sans-serif;
}

h1 {
    font-size: 1.75em;
    margin: 0 0 0.6em 0;
}

a {
    color: #2996cc;
}

a:hover {
    text-decoration: none;
}

p {
    line-height: 1.5em;
}

.small {
    color: #666;
    font-size: 0.875em;
}

.large {
    font-size: 1.25em;
}

/** Nestable */

.list-group-item{
    padding: 5px!important;
}

.dd {
    position: relative;
    display: block;
    margin: 0;
    padding: 0;
    max-width: 600px;
    list-style: none;
    font-size: 13px;
    line-height: 20px;
}

.dd-list {
    display: block;
    position: relative;
    margin: 0;
    padding: 0;
    list-style: none;
}

.dd-list .dd-list {
    padding-left: 30px;
}

.dd-collapsed .dd-list {
    display: none;
}

.dd-item > .dd-handle {
    width: 70%;
    float: left;
    padding: 6px 0px;
}

.dd-item > .dd-option-handle {
    white-space: nowrap;
    margin-bottom: 5px;
}

.dd-item,
.dd-empty,
.dd-placeholder {
    display: block;
    position: relative;
    margin: 0;
    padding: 0;
    min-height: 20px;
    font-size: 13px;
    line-height: 20px;
}
```

```
.dd-handle:hover {
    color: #2ea8e5;
    background: #fff;
}

.dd-item>button {
    display: block;
    position: relative;
    cursor: pointer;
    float: left;
    width: 25px;
    height: 20px;
    margin: 5px 0;
    padding: 0;
    text-indent: 100%;
    white-space: nowrap;
    overflow: hidden;
    border: 0;
    background: transparent;
    font-size: 12px;
    line-height: 1;
    text-align: center;
    font-weight: bold;
}

.dd-item>button:before {
    content: '+';
    display: block;
    position: absolute;
    width: 100%;
    text-align: center;
    text-indent: 0;
}

.dd-item>button[data-action="collapse"]:before {
    content: '-';
}

.dd-placeholder,
.dd-empty {
    margin: 5px 0;
    padding: 0;
    min-height: 30px;
    background: #f2fbff;
    border: 1px dashed #b6bcbf;
    box-sizing: border-box;
    -moz-box-sizing: border-box;
}
```

```
.dd-empty {
    border: 1px dashed #bbb;
    min-height: 100px;
    background-color: #e5e5e5;
    background-image: -webkit-linear-gradient(45deg, #fff 25%, trans
        -webkit-linear-gradient(45deg, #fff 25%, transparent 25%, trans
    background-image: -moz-linear-gradient(45deg, #fff 25%, transpa
        -moz-linear-gradient(45deg, #fff 25%, transparent 25%, transpa
    background-image: linear-gradient(45deg, #fff 25%, transparent 2
        linear-gradient(45deg, #fff 25%, transparent 25%, transparent 7
    background-size: 60px 60px;
    background-position: 0 0, 30px 30px;
}

.dd-dragel {
    position: absolute;
    pointer-events: none;
    z-index: 9999;
}

.dd-dragel>.dd-item .dd-handle {
    margin-top: 0;
}

.dd-dragel .dd-handle {
    -webkit-box-shadow: 2px 4px 6px 0 rgba(0, 0, 0, .1);
    box-shadow: 2px 4px 6px 0 rgba(0, 0, 0, .1);
}

/**
    * Nestable Extras
    */

.nestable-lists {
    display: block;
    clear: both;
    padding: 30px 0;
    width: 100%;
    border: 0;
    border-top: 2px solid #ddd;
    border-bottom: 2px solid #ddd;
}

#nestable-menu {
    padding: 0;
    margin: 20px 0;
}
```

```
#nestable-output,
#nestable2-output {
    width: 100%;
    height: 7em;
    font-size: 0.75em;
    line-height: 1.333333em;
    font-family: Consolas, monospace;
    padding: 5px;
    box-sizing: border-box;
    -moz-box-sizing: border-box;
}

#nestable2 .dd-handle {
    color: #fff;
    border: 1px solid #999;
    background: #bbb;
    background: -webkit-linear-gradient(top, #bbb 0%, #999 100%);
    background: -moz-linear-gradient(top, #bbb 0%, #999 100%);
    background: linear-gradient(top, #bbb 0%, #999 100%);
}

#nestable2 .dd-handle:hover {
    background: #bbb;
}

#nestable2 .dd-item>button:before {
    color: #fff;
}

@media only screen and (min-width: 700px) {

    .dd {
        float: left;
        width: 48%;
    }

    .dd+.dd {
        margin-left: 2%;
    }

}

.dd-hover>.dd-handle {
    background: #2ea8e5 !important;
}

/** Nestable Draggable Handles */
```

```
.dd3-content {
    display: block;
    height: 30px;
    margin: 5px 0;
    padding: 5px 10px 5px 40px;
    color: #333;
    text-decoration: none;
    font-weight: bold;
    border: 1px solid #ccc;
    background: #fafafa;
    background: -webkit-linear-gradient(top, #fafafa 0%, #eee 100%);
    background: -moz-linear-gradient(top, #fafafa 0%, #eee 100%);
    background: linear-gradient(top, #fafafa 0%, #eee 100%);
    -webkit-border-radius: 3px;
    border-radius: 3px;
    box-sizing: border-box;
    -moz-box-sizing: border-box;
}

.dd3-content:hover {
    color: #2ea8e5;
    background: #fff;
}

.dd-dragel>.dd3-item>.dd3-content {
    margin: 0;
}

.dd3-item>button {
    margin-left: 30px;
}

.dd3-handle {
    position: absolute;
    margin: 0;
    left: 0;
    top: 0;
    cursor: pointer;
    width: 30px;
    text-indent: 100%;
    white-space: nowrap;
    overflow: hidden;
    border: 1px solid #aaa;
    background: #ddd;
    background: -webkit-linear-gradient(top, #ddd 0%, #bbb 100%);
    background: -moz-linear-gradient(top, #ddd 0%, #bbb 100%);
    background: linear-gradient(top, #ddd 0%, #bbb 100%);
    border-top-right-radius: 0;
    border-bottom-right-radius: 0;
}

.dd3-handle:before {
    content: '≡';
    display: block;
    position: absolute;
    left: 0;
    top: 3px;
    width: 100%;
    text-align: center;
    text-indent: 0;
    color: #fff;
    font-size: 20px;
    font-weight: normal;
}

.dd3-handle:hover {
    background: #ddd;
}
```

**Javascript Files for Drag, Drop and Sort Funtionality**

In **public/category-subcategory-assets/js/jquery.nestable.js**.

**Note**

Find the link for the script at Github dbushell/Nestable.

In **public/category-subcategory-assets/js/style.js**

```
$(document).ready(function()
{

    var updateOutput = function(e)
    {
        var list   = e.length ? e : $(e.target),
            output = list.data('output');
        if (window.JSON) {
            output.val(window.JSON.stringify(list.nestable('serialize')));//,
        } else {
            output.val('JSON browser support required for this demo.');
        }
    };
    // activate Nestable for list 1
    $('#nestable-wrapper').nestable({
        group: 1,
        maxDepth : 10,
    })
    .on('change', updateOutput);

    // output initial serialised data
    updateOutput($('#nestable-wrapper').data('output', $('#nestable-

    $('#nestable-menu').on('click', function(e)
    {
        var target = $(e.target),
            action = target.data('action');
        if (action === 'expand-all') {
            $('.dd').nestable('expandAll');
        }
        if (action === 'collapse-all') {
            $('.dd').nestable('collapseAll');
        }
    });


});
```

### Create Category Form

In `resources/views/category-subcategory/create.blade.php`.

```
<!DOCTYPE html>
    <html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-
        <title>Category Form</title>
        <link rel="stylesheet" href="{{ url('assets/css/bootstrap.min.css'
    </head>
    <body>

        @include('category-subcategory.includes.menu')

        <div class="container">

            @include('category-subcategory.includes.notification')

            <div class="row">
                <div class="col-md-6">
                    <h2> {{ (isset($category['category_id']))? "Edit" : "Create
                </div>
            </div>

            <form action="{{ route('category-subcategory.store') }}" meth
                @csrf
                @if(isset($category['category_id']))
                    <input type="hidden" name="category_id" value="{{ $c
                @endif
                <div class="row">
                    <div class="col-md-6">
                        <label for="">Name</label>
                        <input type="text" name="name" class="form-contro
                        @if($errors->first('name'))
                            <label for="" style="color:red;">{{ $errors->first('na
                        @endif
                    </div>
                </div>

                <br>

                <div class="row">
                    <div class="col-md-6">
                        <label for="">Parent Category ID</label>
                        <select name="parent_id" class="form-control">
                            <option value="">Choose One</option>
                            @foreach($categories as $k => $v)
                                <option value="{{ $v['category_id'] }}" {{ (isset($c
                            @endforeach
                        </select>
                    </div>
                </div>

                <br>

                <div class="row">
                    <div class="col-md-6">
                        <input type="submit" class="btn btn-success" value=
                    </div>
                </div>

            </form>
        </div>



        <script src="{{ url('assets/js/jquery.min.js') }}"></script>
        <script src="{{ url('assets/js/bootstrap.min.js') }}"></script>
    </body>
</html>
```

### Menu Template

In **resources/views/category-subcategory/includes/menu.blade.php**.

```
<nav class="navbar navbar-expand-sm bg-dark navbar-dark">
    <a class="navbar-brand" href="{{ route('category-subcategory.list

    <ul class="navbar-nav">
        <li class="nav-item">
        <a class="nav-link" href="{{ route('category-subcategory.list') }}
        </li>
        <li class="nav-item">
        <a class="nav-link" href="{{ route('category-subcategory.create
        </li>
    </ul>

</nav>
```

### Notification Template

In **resources/views/category-subcategory/includes/notification.blade.php**.

```
@if(session()->get('success'))
    <div class="row">
        <div class="col-md-6">
            <div class="alert alert-success">
            {{ session()->get('success') }}
            </div>
        </div>
    </div>
@endif

@if(session()->get('error'))
    <div class="row">
        <div class="col-md-6">
            <div class="alert alert-danger">
            {{ session()->get('error') }}
            </div>
        </div>
    </div>
@endif
```

## Credits

The Drag, Drop and Sort functionality as possible by the help of David Bushell Jquery script and we thank you for your wonderful contributions.

## Conclusion

In Conclusion, In the end, you have learnt to handle nested subcategories in **Laravel Categories SubCategories Drag, Drop and Sort using JQuery**. Motivate us by sharing this post and for suggestions and queries comment in below post.

## Related Posts

- The Fastest Way of Debugging Laravel Application using Laravel Debugbar
- PHP Laravel Restful API Authentication with JWT Token using Postman

**Summary**

**Reviewer**

The Code Learners

**Review Date**

2020-06-01

**Reviewed Item**

Laravel Categories SubCategories Drag, Drop and Sort using JQuery

**Author Rating**

★★★★★

**Software Name**

Laravel Framework

**Software Name**

Windows Os, Mac Os, Ubuntu Os

**Software Category**

Web Development

**Previous**
Python Django Projects | Create a Todo Web Application

**Next**
How to Send Emails and Attachments using Django

About Disclaimer Privacy Policy Terms and Conditions Contact