**Laravel Tutorial**

**Laravel** is today’s leading topic. In this example, we will use different input types like a radio button, select box, and checkbox. CRUD Operations are very important in Laravel but with varying types of input. We have used **Laravel 5.8** for this example. We are creating, reading and updating the data from the MySQL database. If you are new to Laravel, then it might be useful to you.

**Installation Requirements**

* X PHP >= 7.2.0
* BCMath PHP Extension
* Ctype PHP Extension
* JSON PHP Extension
* Mbstring PHP Extension
* OpenSSL PHP Extension
* PDO PHP Extension
* Tokenizer PHP Extension
* XML PHP Extension
* ML PHP Extension

**Step 0: Install composer**

Cd desktop

**Step 1: Installing The Laravel Framework.**

composer create-project --prefer-dist laravel/laravel coins

coins is the name of the project. cd desktop and then cd coins. Drag the folder to your sublime. Now, you can see your Laravel structure of your project.

It will create a Hierarchy and install all the Laravel dependencies.

**Step 2: Setup a MySQL database in .env file.**

Create a database in MySQL( Just create the database, no need to create tables) and then switch to your editor and open the **.env**file. Use the name of the database in your .env

// .env

DB\_CONNECTION=mysql

DB\_HOST=127.0.0.1

DB\_PORT=3306

DB\_DATABASE=laravel

DB\_USERNAME=root

DB\_PASSWORD=

Now, migrate two tables provided by Laravel. Switch to your terminal and type following command.

php artisan migrate

Above command will create two new tables in your database.

1. **users**
2. **password\_resets**

**Step 3: Create a model as well as migration file for our Forms table.**

Type the following command in your terminal.

php artisan make:model Form -m

It will create two files.

1. **Form.php** model.
2. **create\_forms\_table** migration file.

We need to create Schema for the forms table. So navigate to **coins  >>  database  >>  migrations  >>  create\_forms\_table.**

*// create\_forms\_table*

public function up()

{

Schema::create('forms', function (Blueprint $table) {

$table->increments('id');

$table->string('coinname');

$table->string('coinprice');

$table->string('radio');

$table->string('dropdown');

$table->string('checkbox');

$table->timestamps();

});

}

Now, migrate the table.

php artisan migrate

In the database, you can see the**forms**table.

**Step 4:  Make one view file to add the form data in the database.**

Make one folder called **forms**and then create a file in the **resources  >>  views  >>  forms  >>  create.blade.php**and add the following code to it.

*<!-- create.blade.php -->*

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Laravel 5 CRUD Tutorial With Example </title>

<link rel="stylesheet" href="{{asset('css/app.css')}}">

</head>

<body>

<div class="container">

<h2>Create A Form</h2><br />

<form method="post">

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<label for="CoinName">CoinName:</label>

<input type="text" class="form-control" name="coinname">

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<label for="CoinPrice">CoinPrice:</label>

<input type="text" class="form-control" name="coinprice">

</div>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<label>Keep</label>

<input type="radio" name="radio" value="keep">

<label>Port</label>

<input type="radio" name="radio" value="port">

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<label>Level</label>

<select name="dropdown">

<option value="beginner">Beginner</option>

<option value="intermediate">Intermediate</option>

<option value="advance">Advance</option>

</select>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<div class="checkbox">

<label><input type="checkbox" value="coindesk" name="option[]">Coindesk</label>

</div>

<div class="checkbox">

<label><input type="checkbox" value="coinbase" name="option[]">CoinBase</label>

</div>

<div class="checkbox">

<label><input type="checkbox" value="zebpay" name="option[]">Zebpay</label>

</div>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<button type="submit" class="btn btn-success" style="margin-left:38px">Submit</button>

</div>

</div>

</form>

</div>

</body>

</html>

**Step 5: Create one controller and route to display the Form**

Type the following command in your terminal.

php artisan make:controller FormController *--resource*

It will create one controller file called**FormController.php,**and It has all the CRUD Functions that needed in this example.

Here, we have used **resource**parameter, so by default, It provides us some routing patterns, but right now, It will not see until we register one route in **routes  >>  web.php**file. So let us do it.

*// web.php*

Route::resource('forms','FormController');

Type the following command in your terminal.

php artisan route:list

Next step would be to go to **FormController.php**file and add into **create()**function some code.

*// FormController.php*

*/\*\**

*\* Show the form for creating a new resource.*

*\**

*\* @return \Illuminate\Http\Response*

*\*/*

public function create()

{

return view('forms.create');

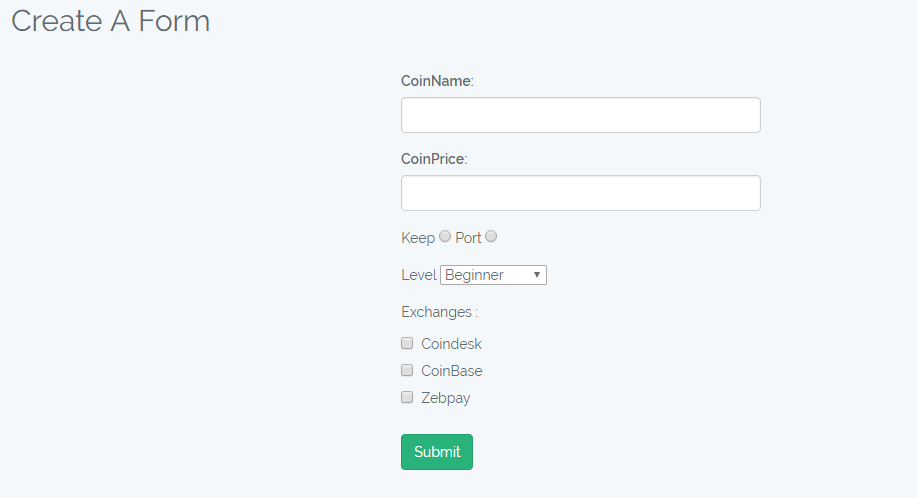
}

After that, we need to start a Development server. So in the terminal, type the following command.

php artisan serve

Switch to the browser and go to this URL: [http://localhost:8000/forms/create](http://localhost:8000/products/create)

You will see the following screen



**Step 6: Put the Laravel Validation in Form.**

Now we need to apply an action to our coin creation form.

*<!-- create.blade.php -->*

<form method="post" action="{{url('forms')}}">

Now, we need to handle **CSRF Token** issue. To get rid of this problem, add the following code in the form to remove the expiry page exception.

*<!-- create.blade.php -->*

{{csrf\_field()}}

One thing to keep in mind that, we need to include the **namespace**of **Form.php**model in the **FormController.php** file. So add the following line at the starting of **FormController**.**php** file.

use App\Form;

Also, we need to put the validation there.

*// FormController.php*

public function store(Request $request)

{

*//*

$request->validate([

'coinname' => 'required',

'coinprice'=> 'required|numeric',

]);

$form= new Form();

$form->coinname=$request->get('coinname');

$form->coinprice=$request->get('coinprice');

$checkbox = implode(",", $request->get('option'));

$form->dropdown=$request->get('dropdown');

$form->radio=$request->get('radio');

$form->checkbox = $checkbox;

$form->save();

return redirect('forms')->with('success', 'Coin has been added');

}

If validation fails, then we need to display an error. So go back to **create.blade.php**and put the following code after **h2**tag.

*<!-- create.blade.php -->*

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Laravel 5 CRUD Tutorial With Example From Scratch </title>

<link rel="stylesheet" href="{{asset('css/app.css')}}">

</head>

<body>

<div class="container">

<h2>Create A Form</h2><br />

@if ($errors->any())

<div class="alert alert-danger">

<ul>

@foreach ($errors->all() as $error)

<li>{{ $error }}</li>

@endforeach

</ul>

</div><br />

@endif

@if (\Session::has('success'))

<div class="alert alert-success">

<p>{{ \Session::get('success') }}</p>

</div><br />

@endif

<form method="post" action="{{url('forms')}}">

{{csrf\_field()}}

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<label for="CoinName">CoinName:</label>

<input type="text" class="form-control" name="coinname">

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<label for="CoinPrice">CoinPrice:</label>

<input type="text" class="form-control" name="coinprice">

</div>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<label>Keep</label>

<input type="radio" name="radio" value="keep">

<label>Port</label>

<input type="radio" name="radio" value="port">

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<label>Level</label>

<select name="dropdown">

<option value="beginner">Beginner</option>

<option value="intermediate">Intermediate</option>

<option value="advance">Advance</option>

</select>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<label>Exchanges :</label>

<div class="checkbox">

<label><input type="checkbox" value="coindesk" name="option[]">Coindesk</label>

</div>

<div class="checkbox">

<label><input type="checkbox" value="coinbase" name="option[]">CoinBase</label>

</div>

<div class="checkbox">

<label><input type="checkbox" value="zebpay" name="option[]">Zebpay</label>

</div>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<button type="submit" class="btn btn-success" style="margin-left:38px">Submit</button>

</div>

</div>

</form>

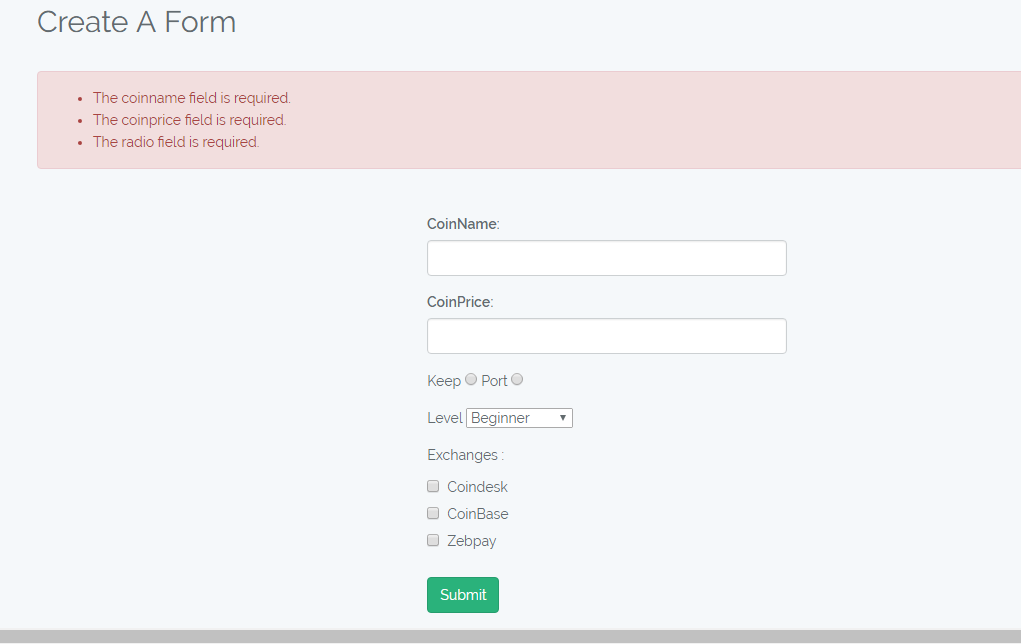
</div>

</body>

</html>

Switch to the browser and go to the following link.  
[http://localhost:8000/forms/create](http://localhost:8000/products/create)

If you submit the form without any value then, you can see the errors like below image.



**Step 7: Make an index page to list the coins.**

To fetch the data from the database, we need to query the database and fetch the collection and then pass that collection to the **index.blade.php**file**.**So, in **FormController.php**file, we need to write the code to bring the data and return it to the **index view.**

*// FormController.php*

*/\*\**

*\* Display a listing of the resource.*

*\**

*\* @return \Illuminate\Http\Response*

*\*/*

public function index()

{

$forms = Form::all()->toArray();

return view('forms.index', compact('forms'));

}

In **resources  >>  views  >>  forms**folder, create one new file called **index.blade.php**file.

*<!-- index.blade.php -->*

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Index Page</title>

<link rel="stylesheet" href="{{asset('css/app.css')}}">

</head>

<body>

<div class="container">

<br />

@if (\Session::has('success'))

<div class="alert alert-success">

<p>{{ \Session::get('success') }}</p>

</div><br />

@endif

<table class="table table-striped">

<thead>

<tr>

<th>ID</th>

<th>CoinName</th>

<th>CoinPrice</th>

<th colspan="2">Action</th>

</tr>

</thead>

<tbody>

@foreach($forms as $form)

<tr>

<td>{{$form['id']}}</td>

<td>{{$form['coinname']}}</td>

<td>{{$form['coinprice']}}</td>

<td><a href="{{action('FormController@edit', $form['id'])}}" class="btn btn-warning">Edit</a></td>

<td>

<form action="{{action('FormController@destroy', $form['id'])}}" method="post">

{{csrf\_field()}}

<input name="\_method" type="hidden" value="DELETE">

<button class="btn btn-danger" type="submit">Delete</button>

</form>

</td>

</tr>

@endforeach

</tbody>

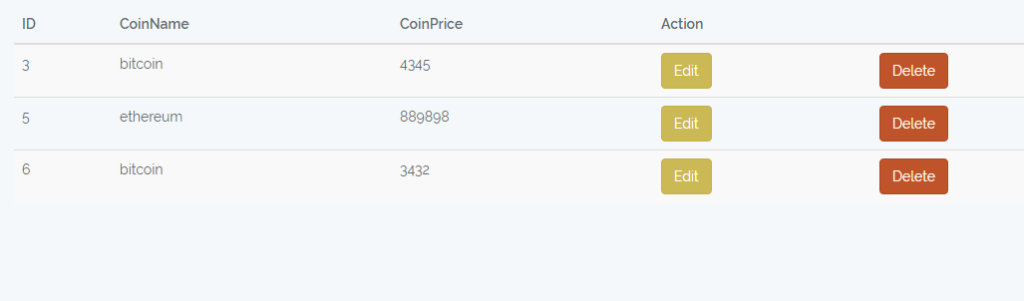
</table>

</div>

</body>

</html>

So, when you go to this link: [**http://localhost:8000/forms**](http://localhost:8000/forms)



**Step 8: Make an edit view for update the forms.**

Next step will be to add the edit function in **FormController.php**file and put the following code in it.

*// FormController.php*

*/\*\**

*\* Show the form for editing the specified resource.*

*\**

*\* @param int $id*

*\* @return \Illuminate\Http\Response*

*\*/*

public function edit($id)

{

$form = Form::find($id);

return view('forms.edit',compact('form','id'));

}

Now, make an **edit.blade.php**file inside **resources  >>  views  >>  forms**

*<!-- edit.blade.php -->*

@php

$values = explode(",", $form->checkbox);

@endphp

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Laravel 5 CRUD Tutorial With Example From Scratch </title>

<link rel="stylesheet" href="{{asset('css/app.css')}}">

</head>

<body>

<div class="container">

<h2>Edit A Form</h2><br />

@if ($errors->any())

<div class="alert alert-danger">

<ul>

@foreach ($errors->all() as $error)

<li>{{ $error }}</li>

@endforeach

</ul>

</div><br />

@endif

<form method="post" action="{{action('FormController@update', $id)}}">

{{csrf\_field()}}

<input name="\_method" type="hidden" value="PATCH">

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<label for="name">CoinName:</label>

<input type="text" class="form-control" name="coinname" value="{{$form->coinname}}">

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<label for="price">CoinPrice:</label>

<input type="text" class="form-control" name="coinprice" value="{{$form->coinprice}}">

</div>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<label>Keep</label>

<input type="radio" name="radio" value="keep" @if($form->radio == 'keep') checked @endif>

<label>Port</label>

<input type="radio" name="radio" value="port" @if($form->radio == 'port') checked @endif>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<label>Level</label>

<select name="dropdown">

<option value="beginner" @if($form->dropdown=="beginner") selected @endif>Beginner</option>

<option value="intermediate" @if($form->dropdown=="intermediate") selected @endif>Intermediate</option>

<option value="advance" @if($form->dropdown=="advance") selected @endif>Advance</option>

</select>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4" style="margin-left:38px">

<lable>Exchanges :</lable>

<div class="checkbox">

<label><input type="checkbox" value="coindesk" name="option[]" @if(in\_array("coindesk", $values)) checked @endif>Coindesk</label>

</div>

<div class="checkbox">

<label><input type="checkbox" value="coinbase" name="option[]" @if(in\_array("coinbase", $values)) checked @endif>CoinBase</label>

</div>

<div class="checkbox">

<label><input type="checkbox" value="zebpay" name="option[]" @if(in\_array("zebpay", $values)) checked @endif>Zebpay</label>

</div>

</div>

</div>

<div class="row">

<div class="col-md-4"></div>

<div class="form-group col-md-4">

<button type="submit" class="btn btn-success" style="margin-left:38px">Update</button>

</div>

</div>

</form>

</div>

</body>

</html>

Next step would be to code the update function to update the values in the database.

*// ProductController.php*

*/\*\**

*\* Update the specified resource in storage.*

*\**

*\* @param \Illuminate\Http\Request $request*

*\* @param int $id*

*\* @return \Illuminate\Http\Response*

*\*/*

$request->validate([

'coinname' => 'required',

'coinprice'=> 'required|numeric',

]);

$form = Form::find($id);

$form->coinname=$request->get('coinname');

$form->coinprice=$request->get('coinprice');

$checkbox = implode(",", $request->get('option'));

$form->dropdown=$request->get('dropdown');

$form->radio=$request->get('radio');

$form->checkbox = $checkbox;

$form->save();

return redirect('forms');

**Step 9: Delete the product.**

*// FormController.php*

*/\*\**

*\* Remove the specified resource from storage.*

*\**

*\* @param int $id*

*\* @return \Illuminate\Http\Response*

*\*/*

public function destroy($id)

{

$form = form::find($id);

$form->delete();

return redirect('forms')->with('success','Coin has been deleted');

}

Finally, our **Laravel Tutorial Example From** is over.