

2. Простые маршруты

Введение

Для реализации практических заданий в корневой директории Open Server **domains** создайте проект Laravel – **host**.

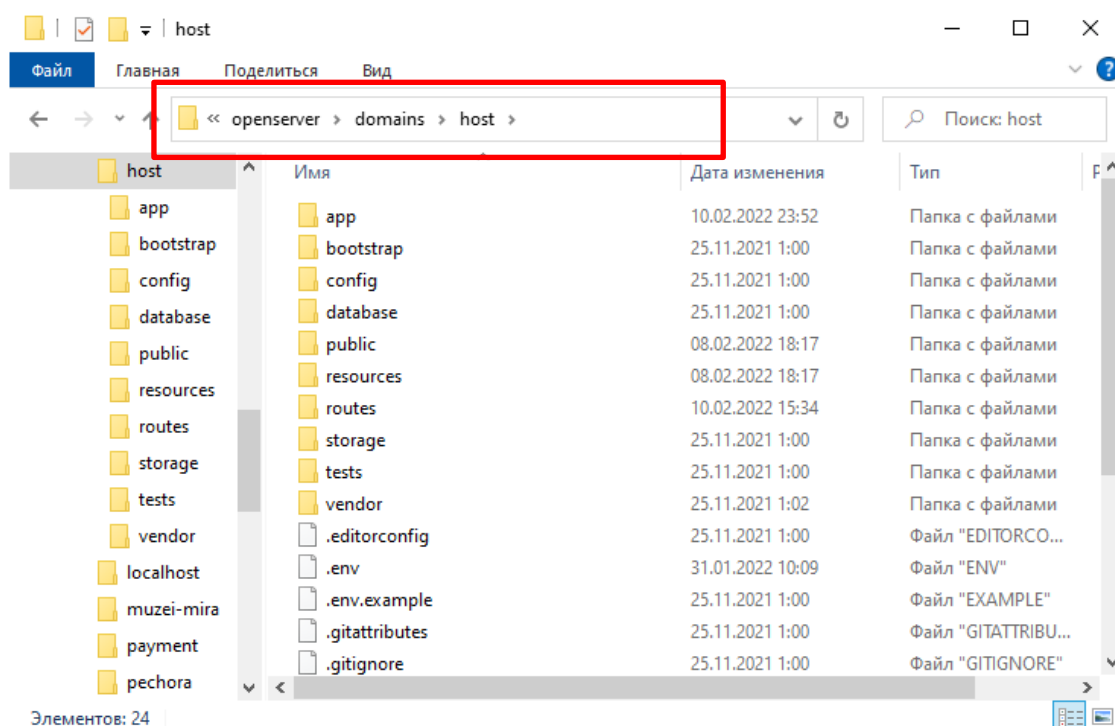


Рис.1. Проект Laravel

В настройках сервера Open Server создайте домен - **http://host** с папкой домена **\host\public**.

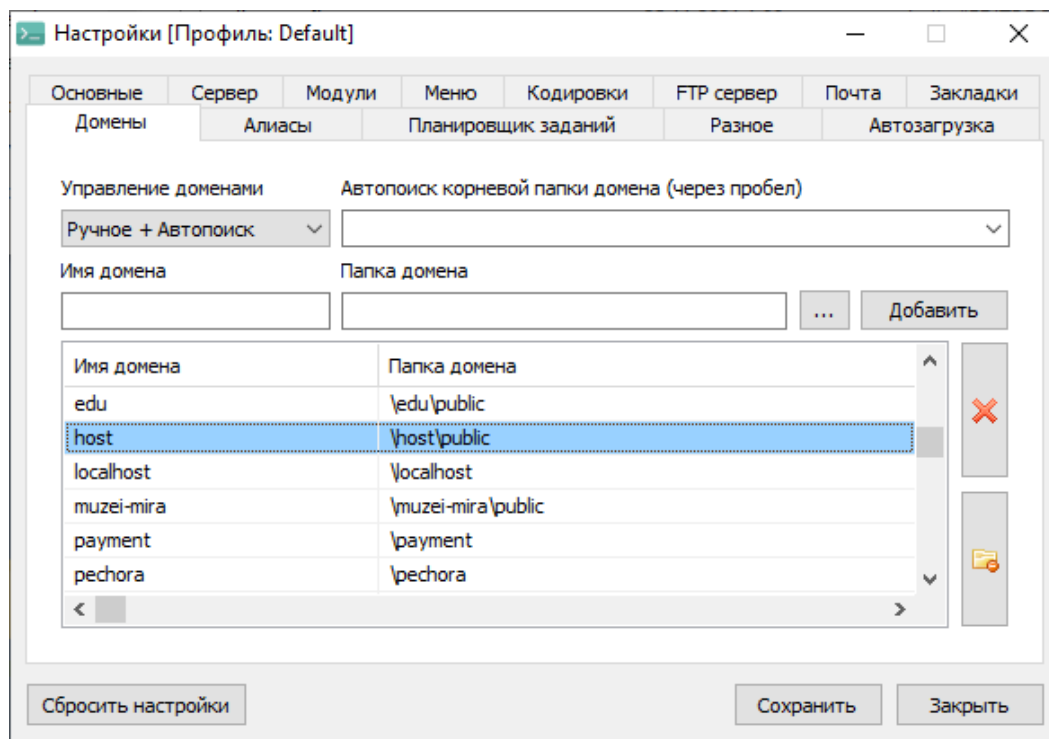


Рис. 2. Создание домена

Быстрый старт в фреймворке Laravel начнем с настройки **простых маршрутов** (маршруты еще называют **роутами**). Маршруты указывают фреймворку, что показывать при обращении к определенному URI в браузере.

Базовая структура **идентификатора ресурса URI** имеет вид:

`http://host/contacts/otdel-kadrov?sotr=Иванов&tel=2-12-85-06`

где:

- `http` - **протокол**
- `host` - **домен**, доменное имя
- `/contacts/otdel-kadrov` - **путь, маршрут, роут**
- `sotr=Иванов&tel=2-12-85-06` - **строка запроса**, get-параметры (параметр - **sotr** и параметр - **tel**)

Простые маршруты

Простейшие маршруты принимают **URI** (путь) и **функцию-замыкание** (callback), предоставляя простой метод определения маршрутов.

Маршруты настраиваются в файле **web.php**, расположенном в папке **routes**

Файл **ex_1/web.php**. Регистрируем маршрут

```
// регистрируем маршрут http://host/  
  
Route::get('/', function () {  
    echo "<h1>Главная страница</h1>";  
});
```

где:

- `'/'` - маршрут
- `function () { ... }` - функция обратного вызова (callback)

Теперь при обращении к URI `'/'` (то есть к главной странице сайта), в браузер будет отправлена строка:

```
"<h1>Главная страница</h1>".
```

В файле `web.php` можно писать **любое количество маршрутов**, указывая соответствующие адреса URI, тем самым создавая маршруты для вашего приложения.

Примеры **простых** маршрутов:

- `muzei-mira/piter`
- `distant/personnel/`
- `contacts/priemnaya/director`

Еще несколько примеров роутов из файла раздаточного материала **ex_1/web.php**:

Файл **ex_1/web.php**. Регистрируем маршрут

```
// регистрируем маршрут http://host/about
Route::get ('/about', function () {
    echo "<h1>Страница: О нас</h1>";
});

// регистрируем маршрут http://host/contacts
Route::get ('/contacts', function () {
    echo "<h1>Страница: Контакты</h1>";
});
```

Маршруты и строка запроса

В дальнейшем, во время разработки приложения, вы можете добавлять к маршруту какие угодно строки запросов, **маршрут от этого не меняется**.

С точки зрения Маршрутизатора приведенные ниже **роуты одинаковы**:

/contacts/otdel-kadrov

=

/contacts/otdel-kadrov?sotr=Иванов&tel=2-12-85-06

Пример регистрации маршрута и передачи в маршрут строки запроса:

Файл **ex_2/web.php**. Регистрируем маршрут

```
// регистрируем маршрут http://host
// набираем в адресной строке:
// http://host?user=master&role=admin
Route::get('/', function () {
```

```
echo "<h1>Главная страница сайта</h1>";

echo "<h1>Пользователь: {$_GET['user']}</h1>";

echo "<h1>Роль пользователя {$_GET['user']} -
{$_GET['role']}</h1>";

});
```

Несмотря на то, что запрашиваемые идентификаторы ресурсов разные, роутер одинаково сработает на запрашиваемый маршрут **http://host** в обоих случаях:

- **http://host/**
- **http://host/?user=master&role=admin**

Файлы раздаточного материала

ex_1/ - примеры простых маршрутов.

ex_2/ - примеры простых маршрутов используемых совместно с **get-параметрами**.