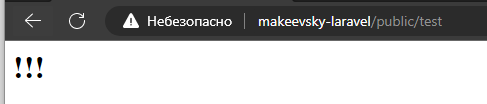
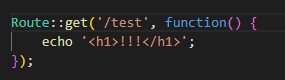
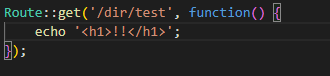
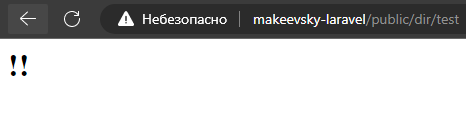
**Практическая работа №1**

1. Сделайте так, чтобы при обращении на адрес /test/ в браузер выводилось сообщение '!'.

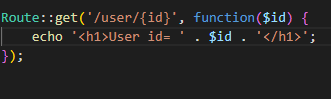


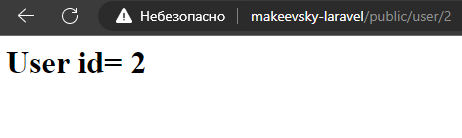
1. Сделайте так, чтобы при обращении на адрес /dir/test/ в браузер выводилось сообщение '!!'.



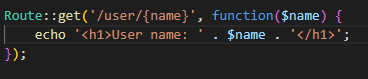


1. Пусть дан адрес вида /user/:id/, где вместо :id может быть любое число. Сделайте так, чтобы при обращении на адрес такого вида в браузер выводилось содержимое переданного параметра.



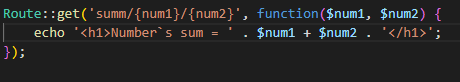


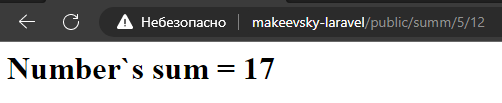
1. Пусть дан адрес вида /user/:name/, где вместо :name может быть любая строка. Сделайте так, чтобы при обращении на адрес такого вида в браузер выводилась строка 'имя юзера :name', где вместо :name будет переданная строка.



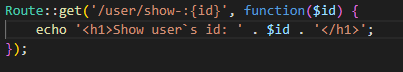


1. Пусть дан адрес вида /sum/:num1/:num2/, где вместо :num1 и :num2 могут быть любые числа. Сделайте так, чтобы при обращении на адрес такого вида в браузер выводилось сумма переданных чисел.



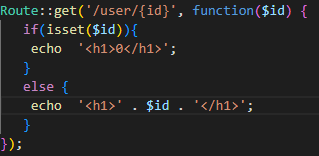


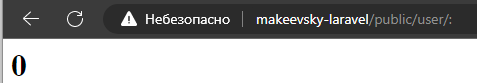
1. Пусть дан адрес вида /user/show-:id/, где вместо :id может быть любое число. Сделайте так, чтобы при обращении на адрес такого вида в браузер выводилось содержимое переданного параметра.



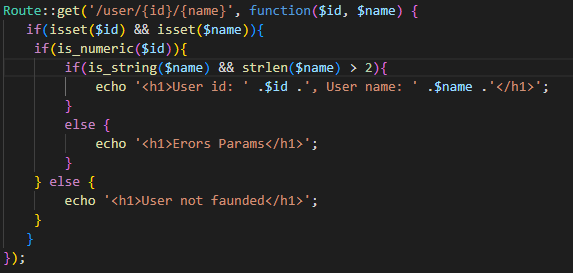


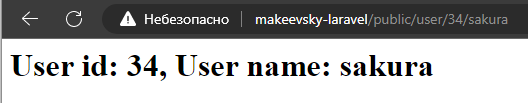
1. Пусть дан адрес вида /user/:id/, где вместо :id может быть любое число. Сделайте так, чтобы параметр id был не обязательный и, если он не передан, вместо него будет передаваться 0.

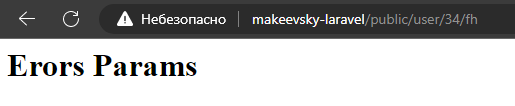




1. Сделайте маршрут вида /user/:id/:name, где вместо :id должно быть число, а вместо :name - строка, состоящая из маленьких латинских букв количеством более 2-х. Добавьте проверку параметров регулярными выражениями.

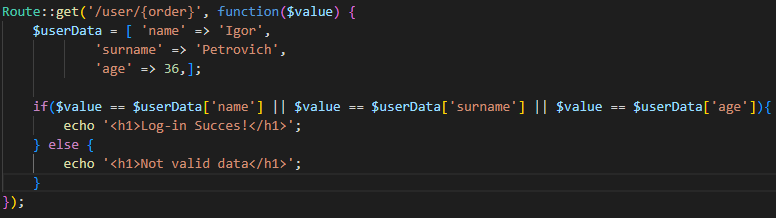


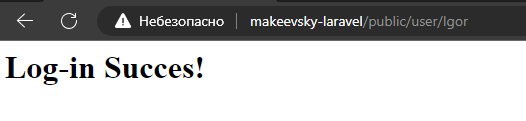


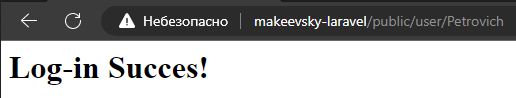


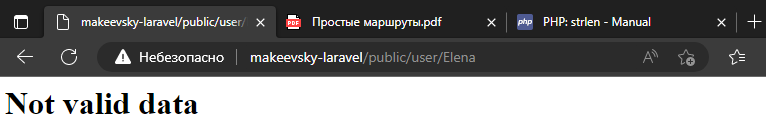


1. Сделайте маршрут вида /users/:order/, где вместо :order должно быть одно из значений: 'name', 'surname' или 'age'.

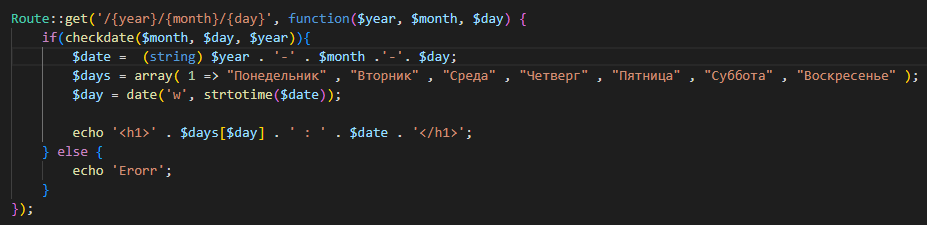


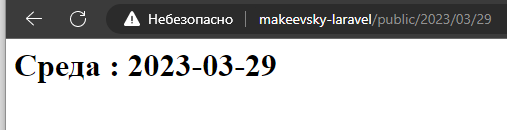






1. Сделайте маршрут вида /:year/:month/:day/, где вместо :year должен быть год, вместо :month - месяц, вместо :day - день. Выведите на экран день недели, соответствующий этой дате.

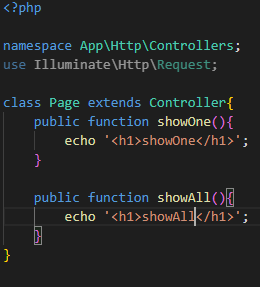




**Практическая работа 2 по Laravel. Контроллеры**

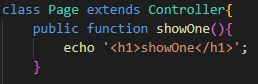
**Задание**

1. Сделайте контроллер **Page.**

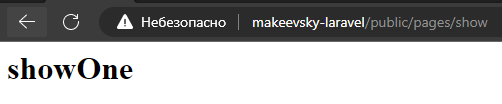


1. В контроллере **Page** сделайте действие **showOne**. Сделайте так, чтобы данное действие было доступно по адресу **/pages/show/**. По обращению к данному действию выведите на экран какую-нибудь строку.

В Controllers/Page.php

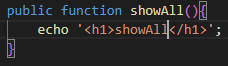


В routes/Web.php

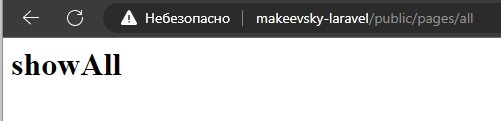
1. В контроллере Page сделайте действие showAll. Сделайте так, чтобы данное действие было доступно по адресу /pages/all/. По обращению к данному действию выведите на экран какую-нибудь строку.

В Controllers/Page.php



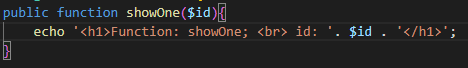
В routes/Web.php





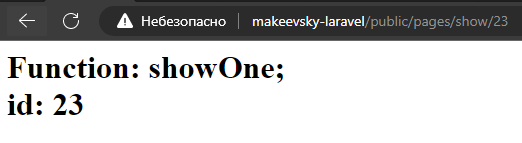
1. Модифицируйте действие showOne контроллера Page. Сделайте так, чтобы данное действие было доступно по адресу /pages/show/:id/, где вместо :id будет какое-нибудь число. По обращению к данному действию выведите на экран переданное число.

В Controllers/Page.php



В routes/Web.php





1. Модифицируйте предыдущую задачу так, чтобы на экран выводился квадрат переданного числа.
2. Сделайте в действии **showOne** следующий массив:

$pages = [

   1 => **'страница 1'**,

   2 => **'страница 2'**,

   3 => **'страница 3'**,

   4 => **'страница 4'**,

   5 => **'страница 5'**,

];

Выведите на экран элемент массива, номер которого соответствует переданному параметру. Если страницы с таким номером нет - выведите сообщение об этом.

1. Создайте контроллер **Test**, а в нем действие **sum**. Сделайте так, чтобы данное действие было доступно по адресу **/test/sum/:num1/:num2/**, где где вместо **:num1** и **:num2** будут какие-нибудь числа. Пусть по обращению к данному действию на экран выводится сумма переданных чисел.
2. Создайте контроллер **Employee** (работник). В конструкторе класса запишите в свойство **$this->employees** массив с работниками:

**namespace** App\Http\Controllers;

**use** App\Http\Controllers\Controller;

**class** TestController **extends** Controller

{

**private $employees**;

**public function** \_\_construct()

   {

       $this->**employees** = [

           1 => [

**'name'** => **'user1'**,

**'surname'** => **'surname1'**,

**'salary'** => 1000,

           ],

           2 => [

**'name'** => **'user2'**,

**'surname'** => **'surname2'**,

**'salary'** => 2000,

           ],

           3 => [

**'name'** => **'user3'**,

**'surname'** => **'surname3'**,

**'salary'** => 3000,

           ],

           4 => [

**'name'** => **'user4'**,

**'surname'** => **'surname4'**,

**'salary'** => 4000,

           ],

           5 => [

**'name'** => **'user5'**,

**'surname'** => **'surname5'**,

**'salary'** => 5000,

           ],

       ];

   }

}

1. В контроллере **Employee** сделайте действие showOne. Параметром оно должно принимать номер работника в массиве $this->employees и выводить на экран строку с именем, фамилией и зарплатой работника.
2. В контроллере **Employee** сделайте действие **showField**. Параметрами оно должно принимать номер работника в массиве **$this->employees** и название одного из полей ('name', 'surname' или 'salary'). Действие выводить на экран значение соответствующего поля соответствующего работника.