

Веб-программирование электронных изданий

Щербакова Алина Николаевна

Лекция 1

Введение в РНР

Что такое PHP?

PHP – это широко используемый язык сценариев общего назначения с открытым исходным кодом.

Т.е., PHP это язык программирования, специально разработанный для написания web-приложений (сценариев), исполняющихся на Web-сервере.

Особенности языка PHP. Преимущества

Преимущества:

- Предоставляет web-разработчикам возможности быстрого создания динамически генерируемых web-страниц.
- Возможность создания HTML документов с внедренными командами PHP (в отличии от языков C и Perl).
- Поддержка широкого круга баз данных.

Особенности языка PHP. Особенности

Особенности:

- Отличием PHP от какого-либо кода, выполняющегося на стороне клиента, например, JavaScript, является то, что PHP-скрипты выполняются на стороне сервера. Соответственно, можно настроить так, что клиенты даже не смогут узнать, получают ли они обычный HTML-файл или результат выполнения скрипта.
- PHP доступен для большинства операционных систем, включая Linux, многие модификации Unix (такие, как Solaris и OpenBSD), Microsoft Windows, Mac OS X и многих других.
- В PHP включена поддержка большинства современных веб-серверов, таких, как Apache, Microsoft Internet Information Server, серверов Netscape и т.д.
- PHP поддерживает "общение" с различными сервисами с использованием таких протоколов, как LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (на платформах Windows) и многих других.

История PHP

1. Истоки PHP лежат в продукте, имевшем название PHP/FI (Personal Home Page / Forms Interpreter). PHP/FI был создан Расмусом Лерддорфом в 1995 году и представлял собой набор Perl-скриптов для ведения статистики посещений .
2. Очень скоро потребовалась большая функциональность и Расмус Лерддорф пишет новую, намного более обширную версию на C, работающую с базами данных и позволяющую пользователям разрабатывать простейшие web-приложения. Этот PHP/FI имел:
 - переменные в стиле Perl;
 - автоматическую интерпретацию форм;
 - возможность встраиваться в html-код;
 - синтаксис языка имел много общего с Perl, но был намного проще и ограниченнее.
3. В 1997 выходит PHP/FI 2.0 (разработкой занималось уже несколько людей).

История PHP

4. В конце 1997 года Энди Гутманс (Andi Gutmans) и Зив Сураски (Zeev Suraski) переписали код с начала. Объединившись с Расмусом Лердорфом, был разработан PHP (Hypertext Preprocessor) версии 3.0 (официально выпущен в 1998 году), который стал официальным преемником PHP/FI, разработка же PHP/FI была практически полностью прекращена.
5. Одной из сильнейших сторон PHP 3.0 была возможность расширения ядра. В последствии интерфейс написания расширений привлек к PHP множество сторонних разработчиков, работающих над своими модулями, что дало PHP возможность работать с огромным количеством баз данных, протоколов.
6. К концу 1998 года PHP использовался десятками тысяч пользователей. Сотни тысяч web-сайтов сообщали о том, что они работают с использованием языка. PHP 3.0 был установлен приблизительно на 10% серверах Интернета!

История PHP

7. В середине 1999 года был разработан новый движок, названный 'Zend Engine' (от имен создателей: Zeev и Andi), успешно справлялся с поставленными более сложными задачами. В мае 2000 года официально вышел PHP 4.0, основанный на этом движке и принесший с собой набор новых дополнительных функций .

8. В середине июля 2004 года был выпущен PHP 5. Он использует новую версию «движка» – Zend Engine 2. В PHP 5 объектная модель была значительно переработана. При этом было добавлено много новых возможностей, благодаря которым PHP 5 получил некоторые черты таких объектно-ориентированных языков, как C++ и Java. Поддержка XML в версии PHP 5 стала полной.

История PHP

9. Выпуск PHP 6 не состоялся. Главной особенностью версии 6 была поддержка символов Unicode, поскольку в основном PHP используется в веб-разработке, вебу же нужна поддержка Unicode, поэтому её реализация имела смысл.

Идея была в том, чтобы реализовать поддержку Unicode непосредственно в ядре. Такой подход должен был расширить возможности языка — от использования в качестве имен переменных и функций, до мощной функциональности интернациональных строк. PHP 6 в планах был амбициозным проектом. К сожалению, этот амбициозный план имел куда больше проблем, чем ожидалось. Большая часть кодовой базы должна была быть портирована для поддержки Unicode, как в случае ядра, так и в случае важных расширений, что оказалось затратным и непростым делом. Это замедлило процесс разработки других возможностей языка. Появились и другие барьеры, что привело к падению интереса к разработке встроенной поддержки Unicode, а со временем проект и вовсе оказался заброшенным.

История PHP

10. В 2015 году был выпущен PHP 7. Почти 10 лет отделяют его от выпуска первого релиза PHP 5. В PHP 7 появилась масса новшеств, увеличилась и скорость работы. Однако в седьмой версии была убрана устаревшая функциональность, что привело к некоторым проблемам с обратной совместимостью, затруднив перевод старых приложений на новую версию.

Рефакторинг внутренних структур данных и добавление дополнительного этапа перед компиляцией кода в виде абстрактного синтаксического дерева — Abstract Syntax Tree (AST), привели к превосходной производительности и более эффективному распределению памяти. Тесты, выполненные на реальных приложениях, показывают, что PHP 7 в среднем вдвое быстрее PHP 5.6, а также использует на 50% меньше памяти во время обработки запросов. **Необходимо отметить**, что в PHP 7 убраны или более некорректны открывающие / закрывающие тэги.

```
<% ... %>,  
<script language="php"> ... </script>
```

История PHP

10. 26 ноября 2020 года состоялся релиз языка программирования PHP 8.0.

Новая версия языка PHP содержит множество новых возможностей и оптимизаций (использование JIT-компилятора, добавление выражения `match`, появились именованные аргументы для использования в библиотеках или ассоциативных массивах, использование нового оператора `Nullsafe` и другие) , ряд новых функций.

Подходы в программировании

Процедурный подход

это монолитная программа с набором инструкций для выполнения, с ветвлениями и подпрограммами.

Объектно-ориентированное программирование

это одна из парадигм разработки, подразумевающая организацию программного кода, ориентируясь на данные и объекты, а не на функции и логические структуры.

Какой подход использовать: ООП или процедурный?

Все зависит от задачи, поставленной перед вами. Если вам нужно создать код для одноразового использования и этот код можно написать четырьмя или пяти строчками, тогда можно использовать процедурный подход программирования.

В том случае, если задача будет повторяться и постоянно меняться условия выполнения программы - лучше использовать объектно-ориентированное программирование.

Подходы в программировании

Процедурный подход

```
$user_input = $_POST['field'];
$filtered_content = filter($user_input);
mysql_connect («dbhost», «dbuser», «dbpassword»);
mysql_select_db («dbname»);
$sql = «some query»;
$result = mysql_query($sql);
while ($data = mysql_fetch_assoc())
{
    process ($data);
}
process_user_input($filtered_content);
```

Объектно-ориентированное программирование

```
$input_filter = new filter();
$input_filter->filter_user_input();
$db = new dal («mysql»);
$db->connect ($dbconfig);
$result = $db->execute ($sql);
ReportGenerator::makereport ($result);
$model = new Postmodel ($filter->
get_filtered_content());
$model->insert();
```

Подходы в программировании

Процедурный подход

```
$value = "Hi!";  
echo $value;
```

Объектно-ориентированное программирование

```
class Human {  
    private $words;  
    public function setWords($words) {  
        $this->words = $words;  
    }  
    public function getWords() {  
        return $this->words;  
    }  
    public function sayIt() {  
        return $this->getWords();  
    }  
}  
  
$human = new Human();  
$human->setWords("Hi!");  
echo $human->sayIt();
```

Выполнение программ PHP



Процесс выполнения php-сценария при обращении браузера к серверу:

1. Браузер запрашивает страницу с расширением .php .
2. После этого web-сервер пропускает программу через машину PHP и выдаёт результат в виде html-кода.
3. *Особенность.* Если взять стандартную страницу HTML, изменить расширение на .php и пропустить её через машину PHP, последняя просто перешлёт её пользователю без изменений.

Чтобы включить в этот файл команды PHP, необходимо заключить команды PHP в специальные теги, которых различают 4 вида:

1. Инструкция обработки XML:

```
<?php  
...  
?>
```

2. Инструкция обработки SGML:

```
<?  
...  
?>
```

3. Инструкция обработки сценариев HTML:

```
<script language = "php">  
...  
</script>
```

4. Инструкция в стиле ASP:

```
<%  
...  
%>
```

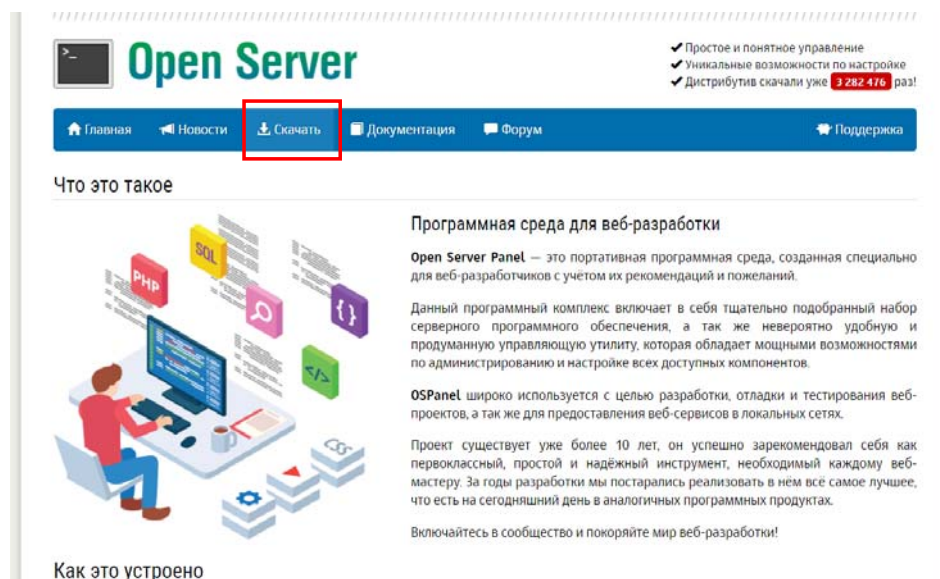
При дальнейшем изучении дисциплины будем придерживаться стиля XML.

Знакомство с языком PHP.

Установка OpenServer

Для использования языка PHP необходимо сначала установить один из web-серверов, например, Wamp (<http://www.wampserver.com/ru/>), OpenServer (<https://ospanel.io>), Denwer (<http://www.denwer.ru/>), XAMPP (<https://www.apachefriends.org/ru/index.html>) и т.д.


1. Заходим на официальный сайт, где нужно нажать по кнопке «Скачать» (она располагается на верхней панели).



Знакомство с языком PHP.

Установка OpenServer

2. Запускаем загрузку, нажав на кнопку «Нет, спасибо, хочу просто скачать»

 **Open Server**

- ✓ Простое и понятное управление
- ✓ Уникальные возможности по настройке
- ✓ Дистрибутив скачали уже **3 282 481** раз!

[Главная](#) [Новости](#) [Скачать](#) [Документация](#) [Форум](#) [Поддержка](#)


Загрузка Open Server Panel 5.4.3

Размер файла: 946 МБ

Поддержите наш проект

Open Server Panel — **некоммерческий** проект, это означает что команда разработчиков распространяет дистрибутив бесплатно и работает только на энтузиазме. Ваша финансовая поддержка очень важна для существования и успешного развития проекта.

Мы предлагаем вам воспользоваться услугой «Быстрая загрузка», которая позволит оперативно скачать дистрибутив Open Server Panel на скорости до 100 Мбит/с по специальной ссылке.

 **Услуга «Быстрая загрузка»**
Принимаем карты любого банка мира в любой валюте

Сумма... руб

Е-mail адрес...

Комментарий...

Способ оплаты

Банковские карты, Google Pay, Apple Pay

☐ Я принимаю условия Публичной оферты

VISA

Mastercard

БЕЛКАРТ

МИР

G Pay

SAMSUNG Pay

Apple Pay

Продолжить на высокой скорости

Нет, спасибо, хочу просто скачать

Комментарии пользователей

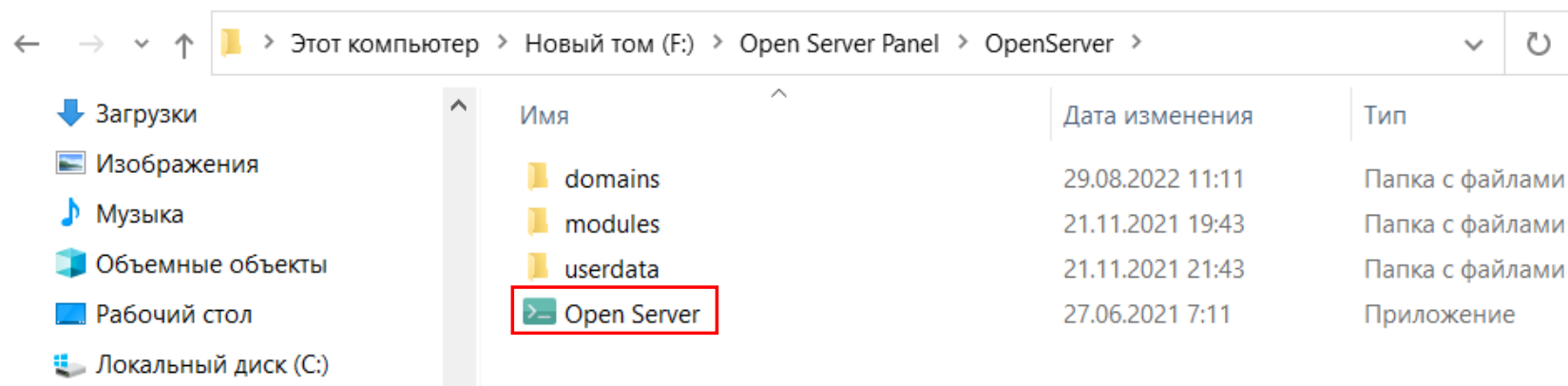
Помни, что краткость — сестра таланта :-)

29 августа	Баркал
29 августа	Спс
29 августа	Спс
29 августа	Всех благ!
29 августа	Кря
29 августа	Deus vult
29 августа	Спасибо из Твери!
29 августа	СПС из Владивостока
29 августа	Np
29 августа	Спасибо
29 августа	100 МБ В СЕКУНДУ ЗА 10 РУБЛЕЙ!
29 августа	На развитие проекта
29 августа	Спасибо!
29 августа	Спасибо!
29 августа	Спс
29 августа	Рахмат
29 августа	Thanks
28 августа	Спасибо
28 августа	Спасибо
28 августа	Спасибо!

Знакомство с языком PHP.

Установка OpenServer

3. Ожидайте завершения скачивания, а затем запустите полученный EXE-файл.
4. Задайте путь, куда будет установлен локальный сервер, и нажмите «Извлечь».
5. Перейдите по пути установки программы и запустите находящийся там исполняемый файл.

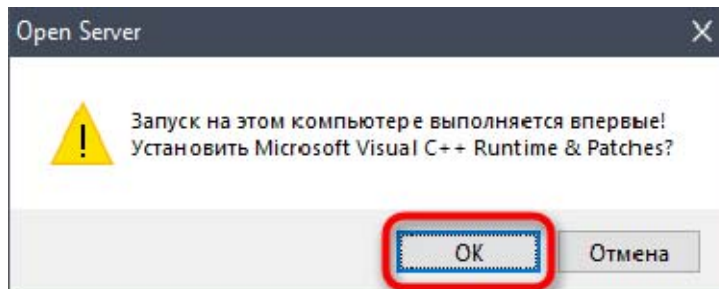


Знакомство с языком PHP.

Установка OpenServer

6. Выберите удобный язык интерфейса.

7. Вы будете уведомлены, что запуск осуществляется впервые, а это значит, что нужно установить патчи для Microsoft Visual C++. Обязательно подтвердите эту операцию.

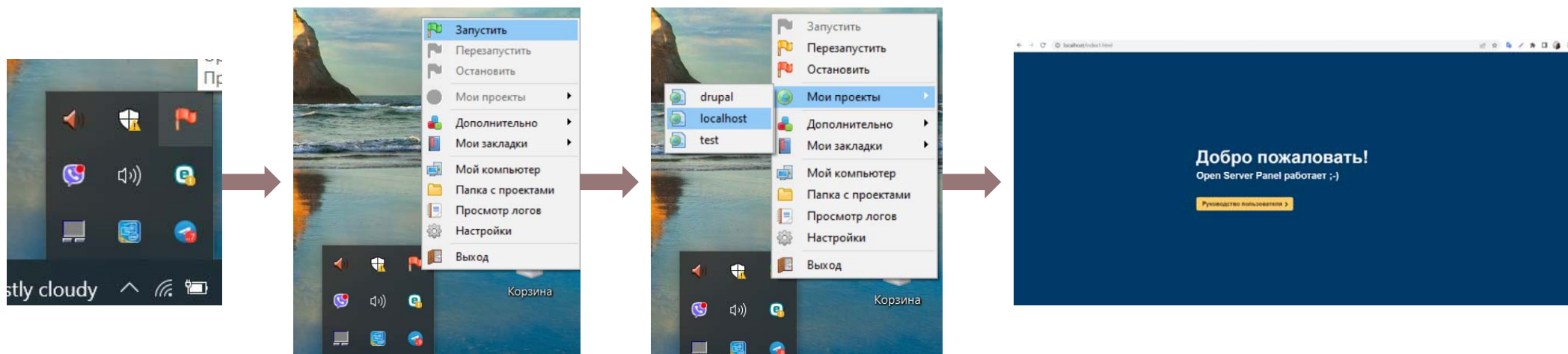


После установки желательно перезагрузить компьютер, и можно переходить к следующему этапу взаимодействия с OpenServer.

Знакомство с языком PHP.

Первый запуск OpenServer

1. После запуска сервера обратите внимание на панель задач. Там щелкните по значку OpenServer.
 2. В появившемся меню нажмите по кнопке «Запустить».
 3. После этого наведите курсор на «Мои проекты» и выберите там «localhost».
- Откроется новая страница в браузере по умолчанию, где вы получите уведомление о нормальной работе локального сервера.



Знакомство с языком PHP. Если Mac OS?

MAMP — это комплект софта, который устанавливается из одного пакета, сам настраивается и превращает ваш компьютер в локальный сервер для разработки и тестирования сайтов. В одном пакете уже собрано и настроено всё необходимое. Распаковывай и пользуйся.

MAMP — это аббревиатура:

Mac OS или Macintosh — операционная система, на которой будет работать софт;

Apache — веб-сервер;

MySQL — база данных;

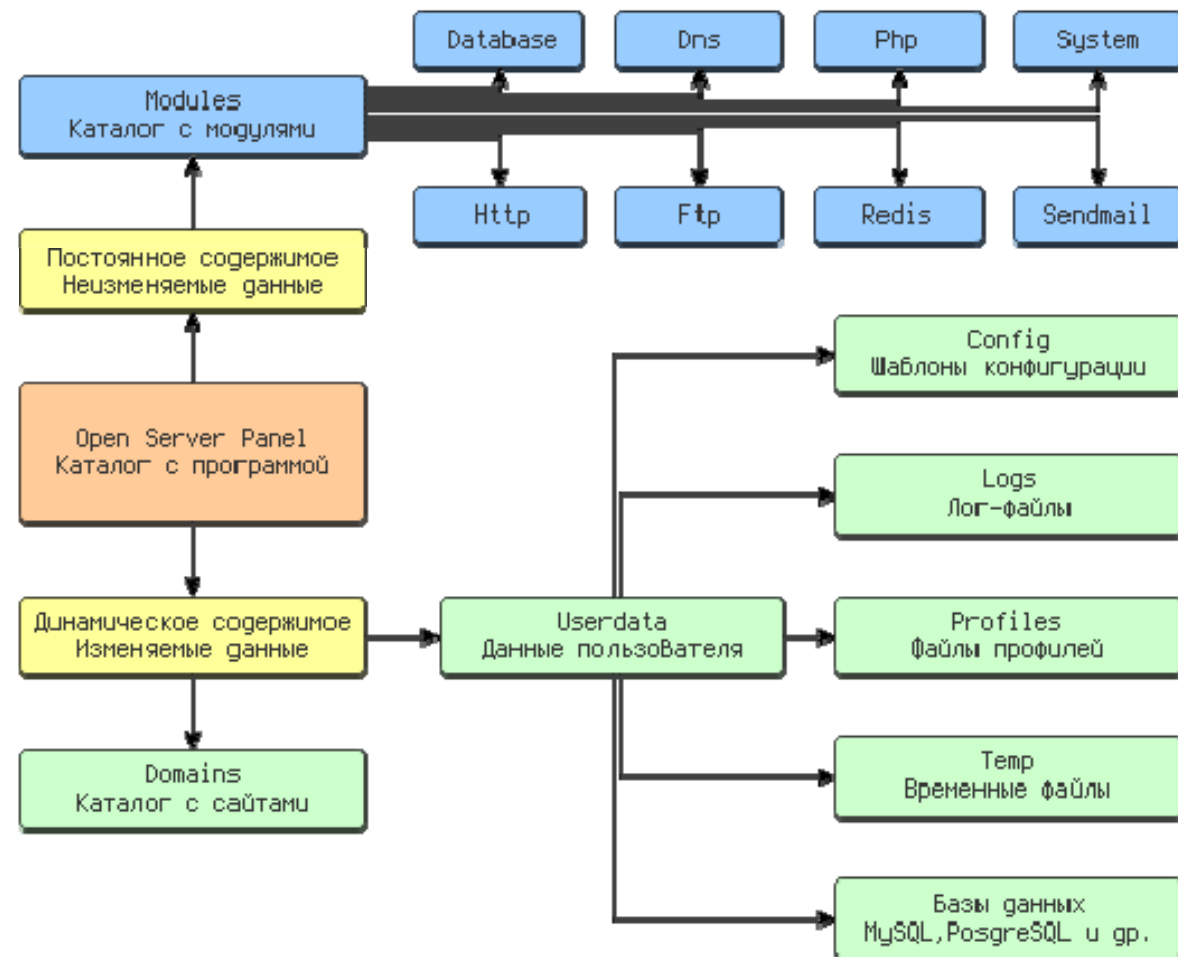
PHP — скриптовый язык для работы сайта.

Знакомство с языком PHP.

Настройка OpenServer

Архитектура расположения каталогов программного комплекса OpenServer подразумевает под собой чёткое разделение двух различных типов данных: изменяемых данных пользователя (настройки, временные файлы, логи т.д.) и неизменяемых данных (модули, программы, служебные файлы).

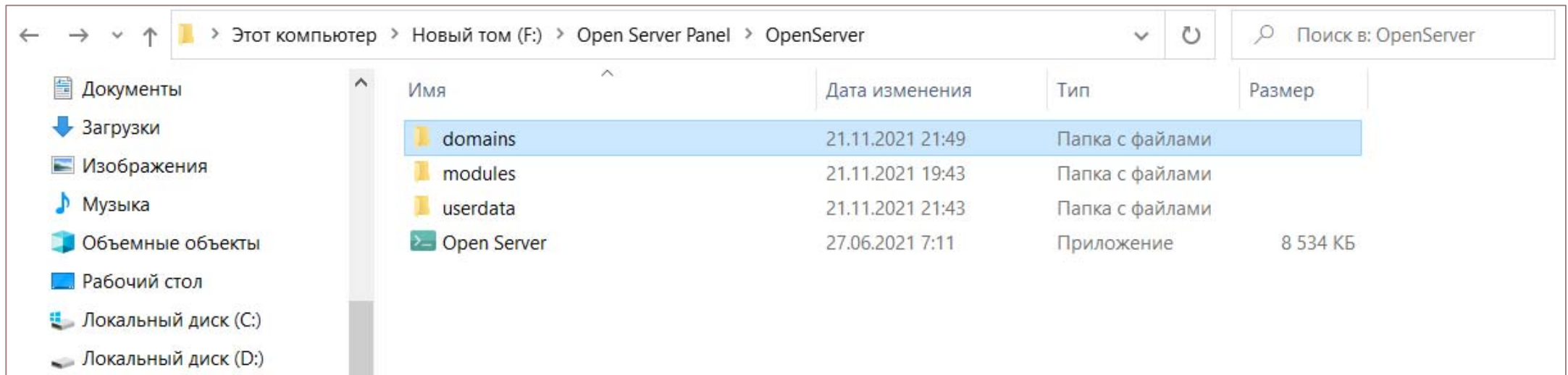
Фактически в папке modules никакие файлы никогда не изменяются, не создаются и не удаляются, а в папках domains и userdata напротив, данные постоянно меняются. Такая архитектура создана специально для упрощения синхронизации данных между различными копиями комплекса и экономии места при резервном копировании.



Знакомство с языком PHP.

Настройка OpenServer

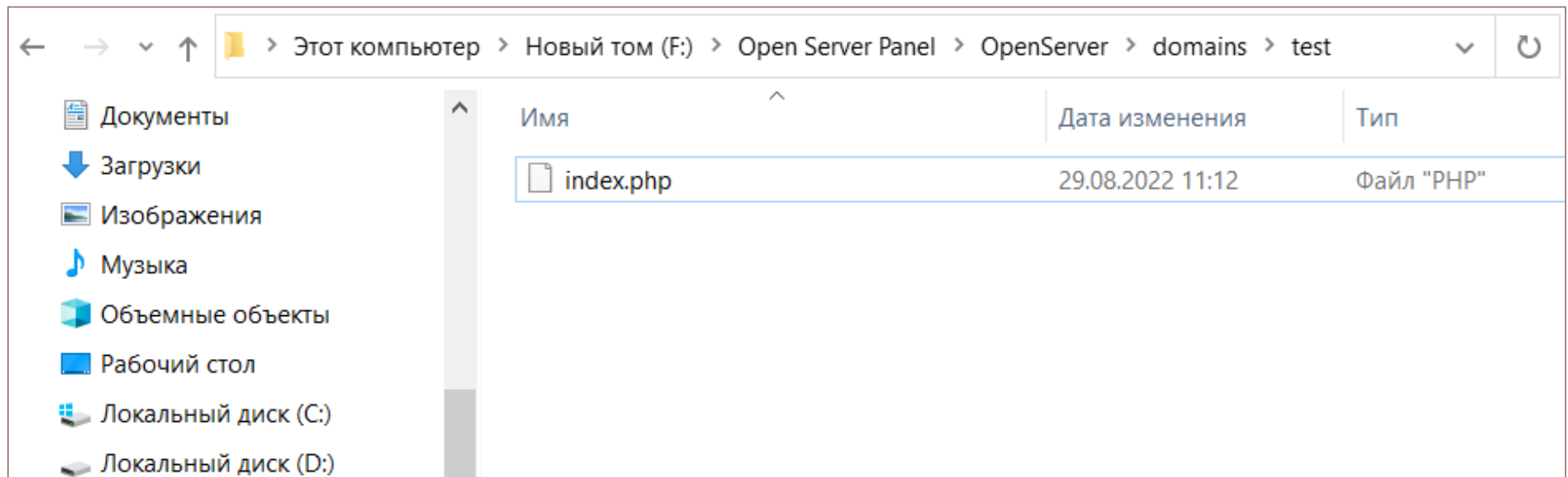
Все веб-серверы (программные комплексы) имеют выделенный каталог для размещения проектов, написанных на языке PHP. Так, например для веб-сервера Wamp по умолчанию это C:\wamp\www, а для OpenServer – C:\OpenServer\OSPanel\domains.



Знакомство с языком PHP.

Настройка OpenServer



В данном каталоге необходимо создать каталог проекта, например, test.dev или просто test (отметим, что некоторые веб-серверы не допускают использования многоуровневого вложения доменных имен). По умолчанию стартовым файлом будет index.php



Знакомство с языком PHP.

Возьмем в качестве основного OpenServer. Выведем в данном файле информацию об используемой версии PHP с помощью команды `phpinfo()`.

```
<?php
    phpinfo();
?>
```

<div> <div>PHP Version 8.0.8</div> <div>  </div> </div>	
System	Windows NT DESKTOP-VUNIJM1 10.0 build 19044 (Windows 10) AMD64
Build Date	Jun 29 2021 15:59:22
Build System	Microsoft Windows Server 2019 Datacenter [10.0.17763]
Compiler	Visual C++ 2019
Architecture	x64
Configure Command	cscript /nologo /e:javascript configure.js "--enable-snapshot-build" "--enable-debug-pack" "--with-pdo-oci=.\\..\\..\\instantclient\\sdk\\shared" "--with-oci8-19=.\\..\\..\\instantclient\\sdk\\shared" "--enable-object-orient" "--enable-com-dotnet=shared" "--without-analyzer" "--with-pgo"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	no value
Loaded Configuration File	W:\modules\php\PHP_8.0\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20200930
PHP Extension	20200930
Zend Extension	420200930
Zend Extension Build	API420200930,TS,VS16
PHP Extension Build	API20200930,TS,VS16
Debug Build	no
Thread Safety	enabled
Thread API	Windows Threads
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	php, file, glob, data, http, ftp, zip, compress, zlib, compress, bzip2, https, ftps, phar
Registered Stream Socket Transports	tcp, udp, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	convert.iconv.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, zlib.*, bzip2.*
<div> <div> This program makes use of the Zend Scripting Language Engine: Zend Engine v4.0.8, Copyright (c) Zend Technologies </div> <div>  </div> </div>	

Знакомство с языком PHP.

Попробуем написать первый скрипт, выводящий какую-либо информацию, например, 'Привет, я - скрипт PHP!'. Для этого в C:\OpenServer\OSPanel\domains создадим каталог test и в нем создадим файл index.php со следующим содержимым (оператор *echo* позволяет выводить на экран).

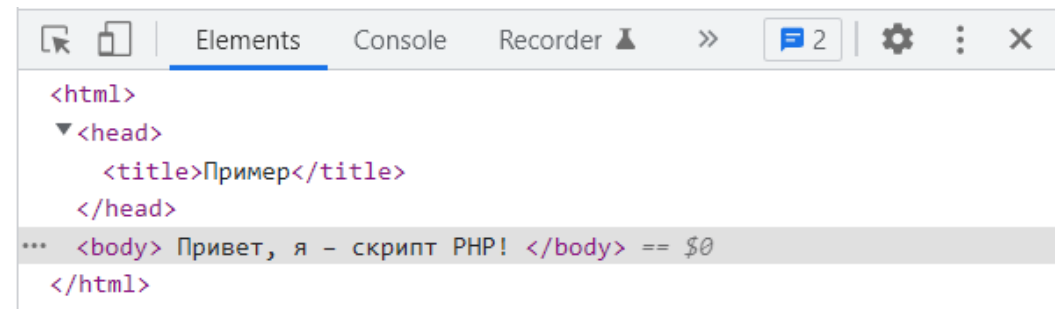
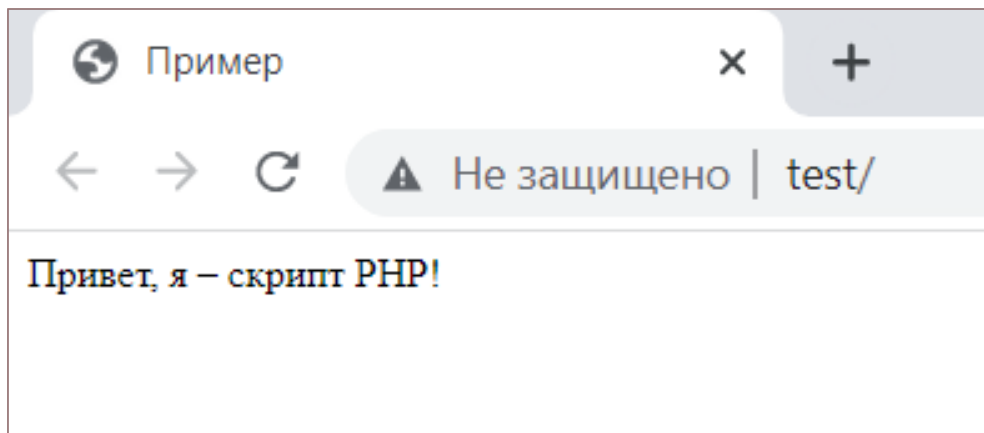
```
<html>
<head>
    <title>Пример</title>
</head>
<body>

<?php
echo ('Привет, я - скрипт PHP!');
?>

</body>
</html>
```

Знакомство с языком PHP.

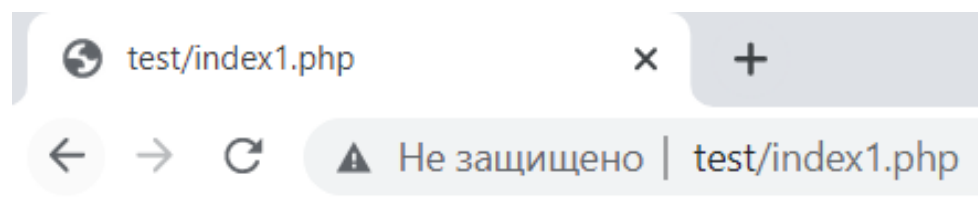
Если изучить HTML-вывод приведенного примера, можно заметить, что PHP-код отсутствует в файле, отправленном с сервера в браузер. Весь PHP, присутствующий на веб-странице, обрабатывается и удаляется со страницы. Единственное, что возвращается клиенту с веб-сервера, — это чистый вывод HTML.



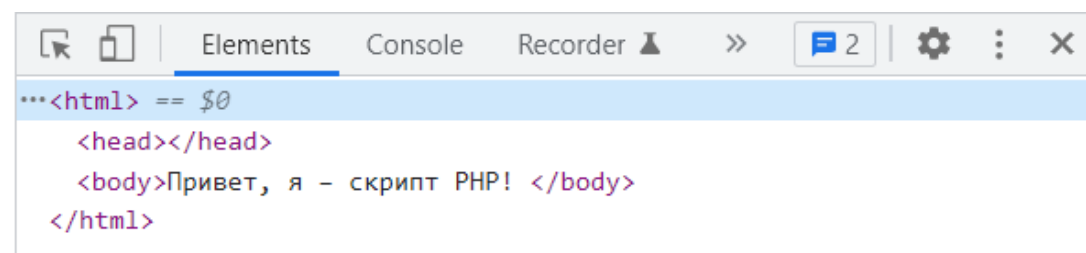
Знакомство с языком PHP.

```
<?php  
echo ('Привет, я - скрипт PHP!');  
?>
```

Знакомство с языком PHP.

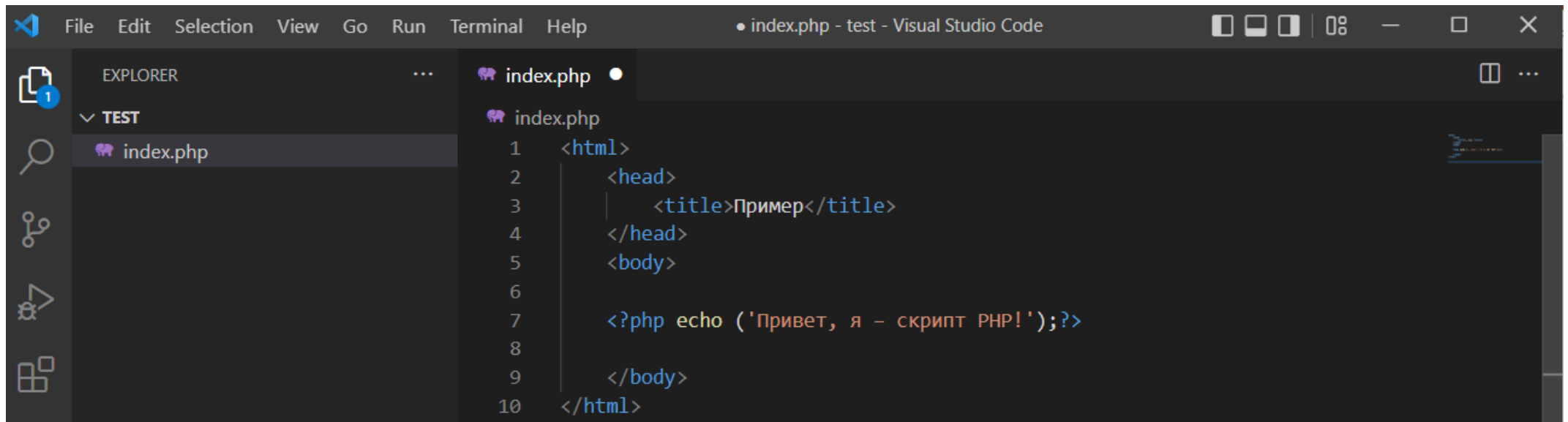


Привет, я – скрипт PHP!



Знакомство с языком PHP.

Для дальнейшей работы данную страницу можно не закрывать, а при необходимости обновлять. Для разработки приложений на языке PHP удобно пользоваться какой-либо популярной IDE-программой, например, PhpStorm, Sublime Text, Visual Studio Code, Eclipse PDT, Zend Studio и т.д.



```
File Edit Selection View Go Run Terminal Help
• index.php - test - Visual Studio Code

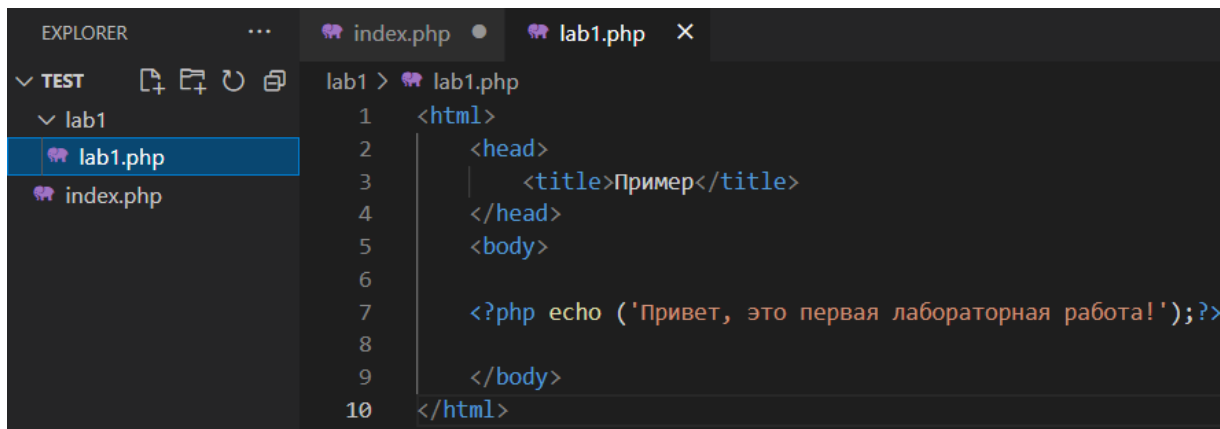
EXPLORER
TEST
  index.php

index.php
1 <html>
2   <head>
3     <title>Пример</title>
4   </head>
5   <body>
6
7     <?php echo ('Привет, я – скрипт PHP!');?>
8
9   </body>
10  </html>
```

Пример использования Visual Studio Code.

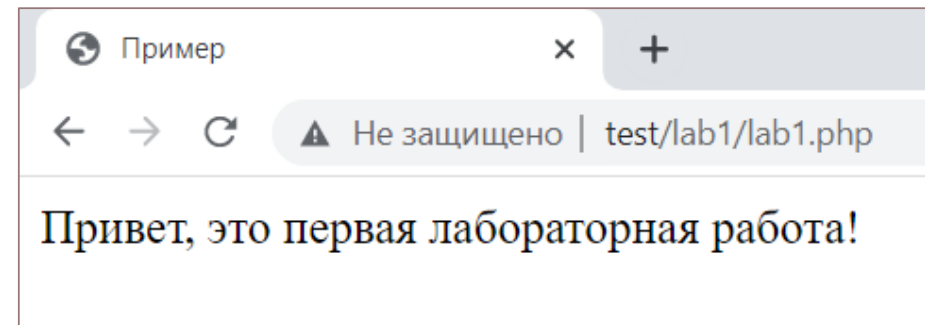
Знакомство с языком PHP.

Если необходимо посмотреть выполнение какого-либо php-файла (не index.php), то необходимо в URL-строке указать путь к нему относительно каталога проекта, например, `http://test/lab1/lab1.php`.



The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project named 'TEST' with a subdirectory 'lab1' containing two files: 'lab1.php' and 'index.php'. The 'lab1.php' file is selected. The code editor shows the following PHP code:

```
1 <html>
2   <head>
3     <title>Пример</title>
4   </head>
5   <body>
6
7     <?php echo ('Привет, это первая лабораторная работа!');?>
8
9   </body>
10 </html>
```



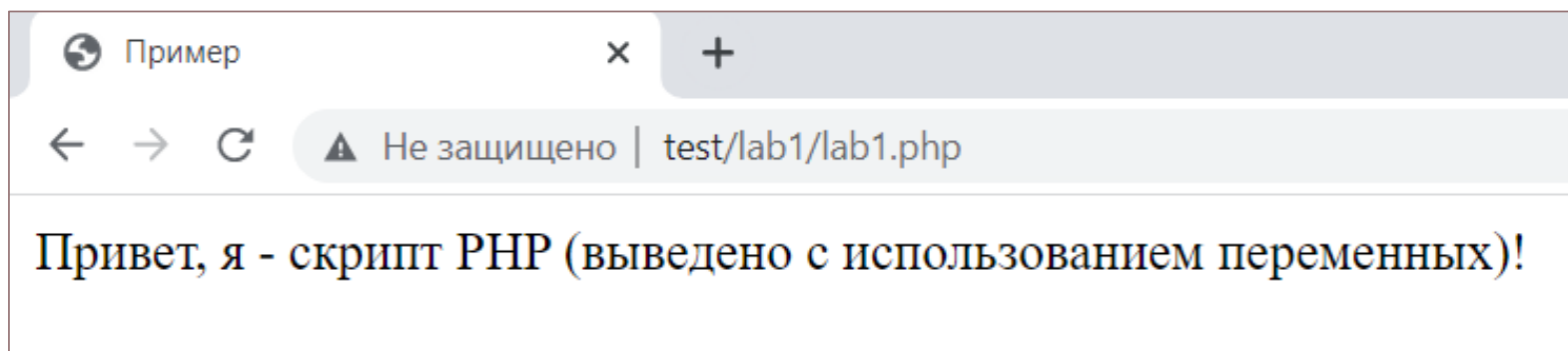
Комментарии в PHP скриптах

```
<?php
echo "Это тест"; // Это однострочный комментарий в стиле c++
/* Это многострочный комментарий
   еще одна строка комментария */
echo "Это еще один тест";
echo "Последний тест"; # Это комментарий в стиле оболочки Unix
?>
```

Однострочные комментарии идут только до конца строки или текущего блока PHP-кода, в зависимости от того, что идет перед ними.

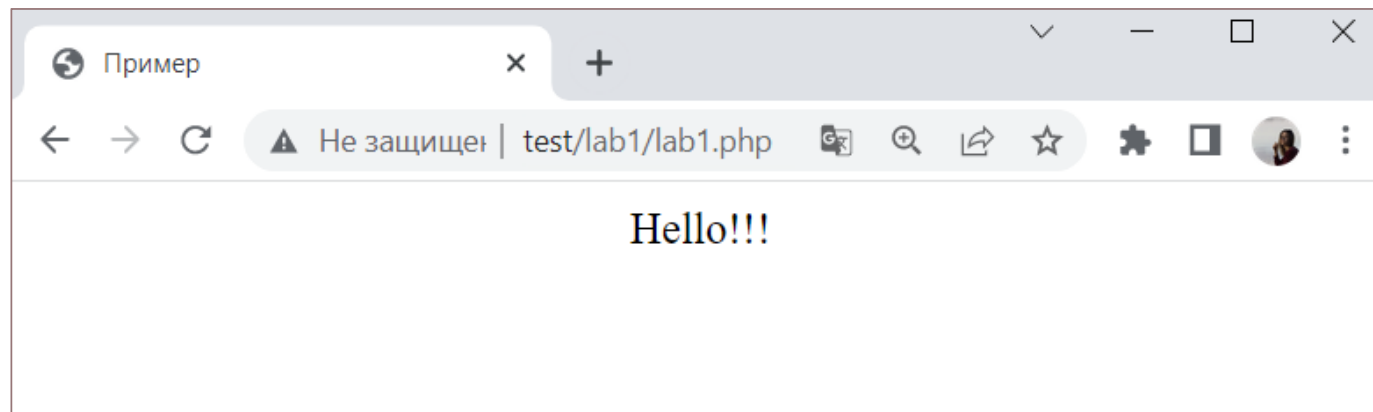
Переменные в PHP

```
<?php
$message = "Привет, я - скрипт PHP (выведено с использованием переменных)";
echo $message;
?>
```



Примеры

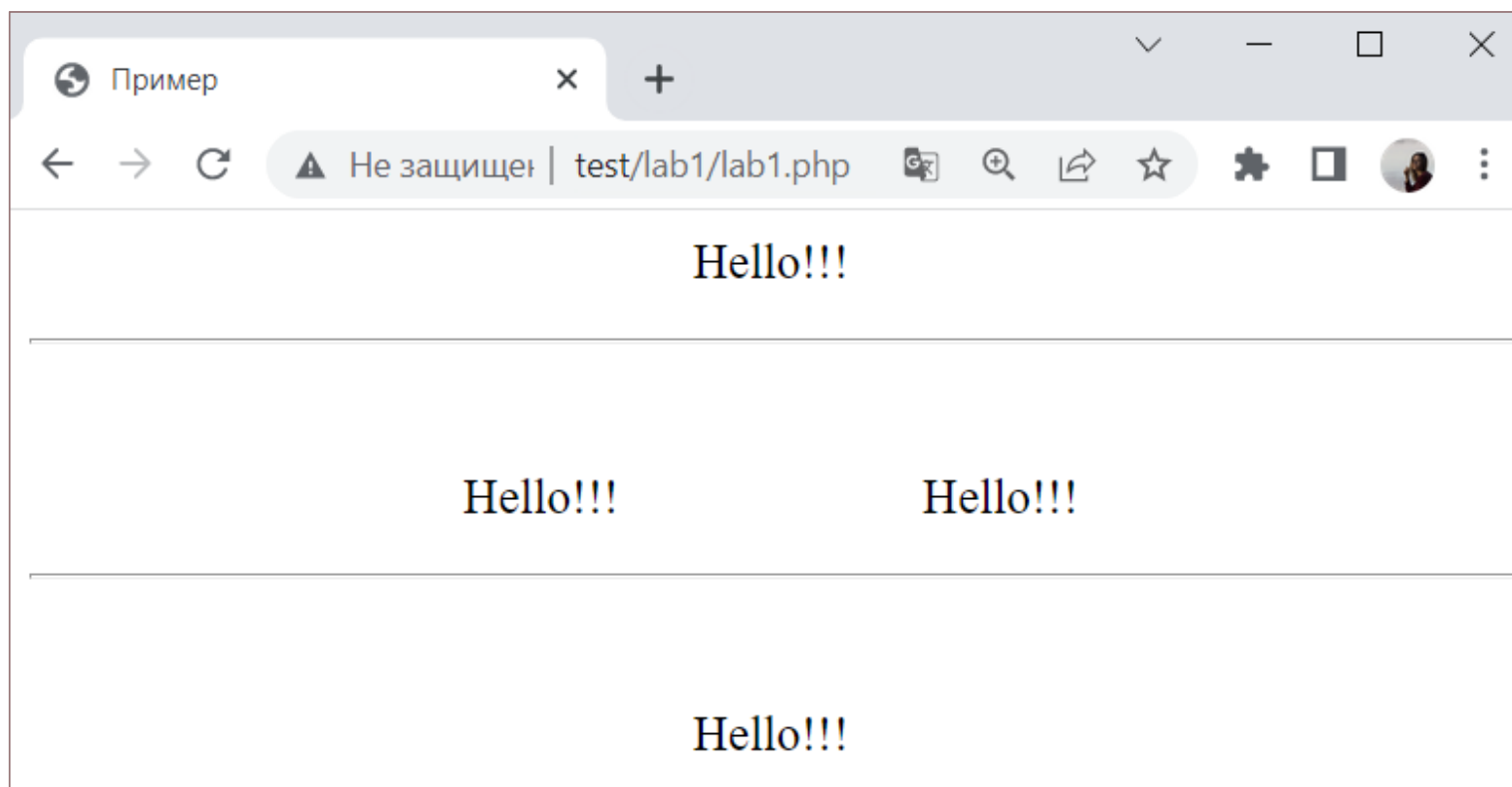
```
<p align="center">
  <?php
    echo "Hello!!!";
  ?>
</p>
```



Примеры

```
<p align="center"> <?php echo "Hello!!!" ?> <p>  
<hr>  
<br>  
<p align="center"> <?php echo "Hello!!!" ?>  
&emsp;&emsp;&emsp;&emsp;&emsp;&emsp;&emsp;  
<?php echo "Hello!!!" ?> <p>  
<hr>  
<br>  
<p align="center"> <?php echo "Hello!!!" ?> <p>
```

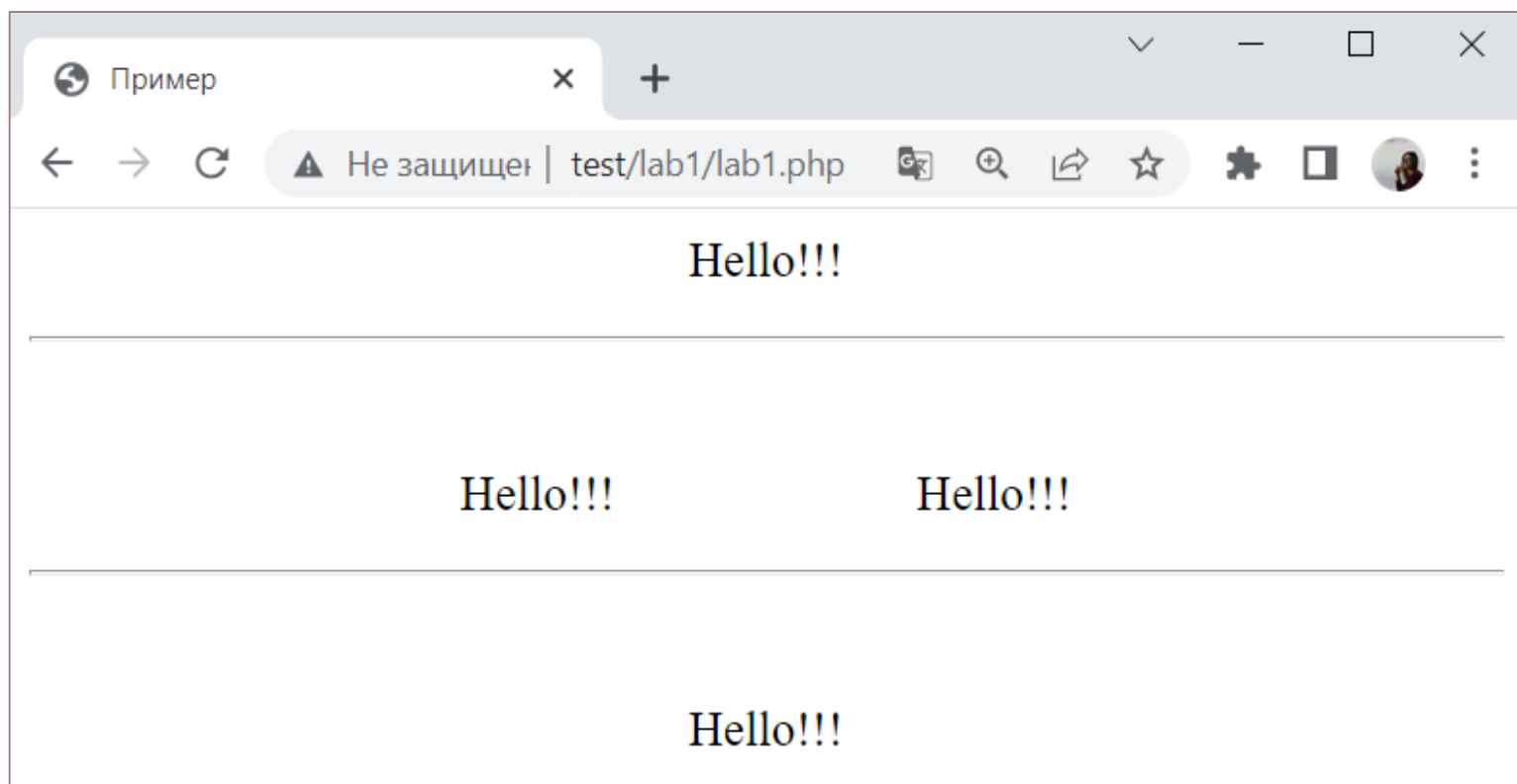
Примеры



Примеры

```
<p align="center">
    <?php
        echo "Hello!!!";
    ?>
<p>
<hr>
<br>
<p align="center">
    <?php
        echo "Hello!!!";
    ?>
    &emsp;&emsp;&emsp;&emsp;&emsp;&emsp;
    <?php
        echo "Hello!!!";
    ?>
<p>
<hr>
<br>
<p align="center">
    <?php
        echo "Hello!!!";
    ?>
<p>
```

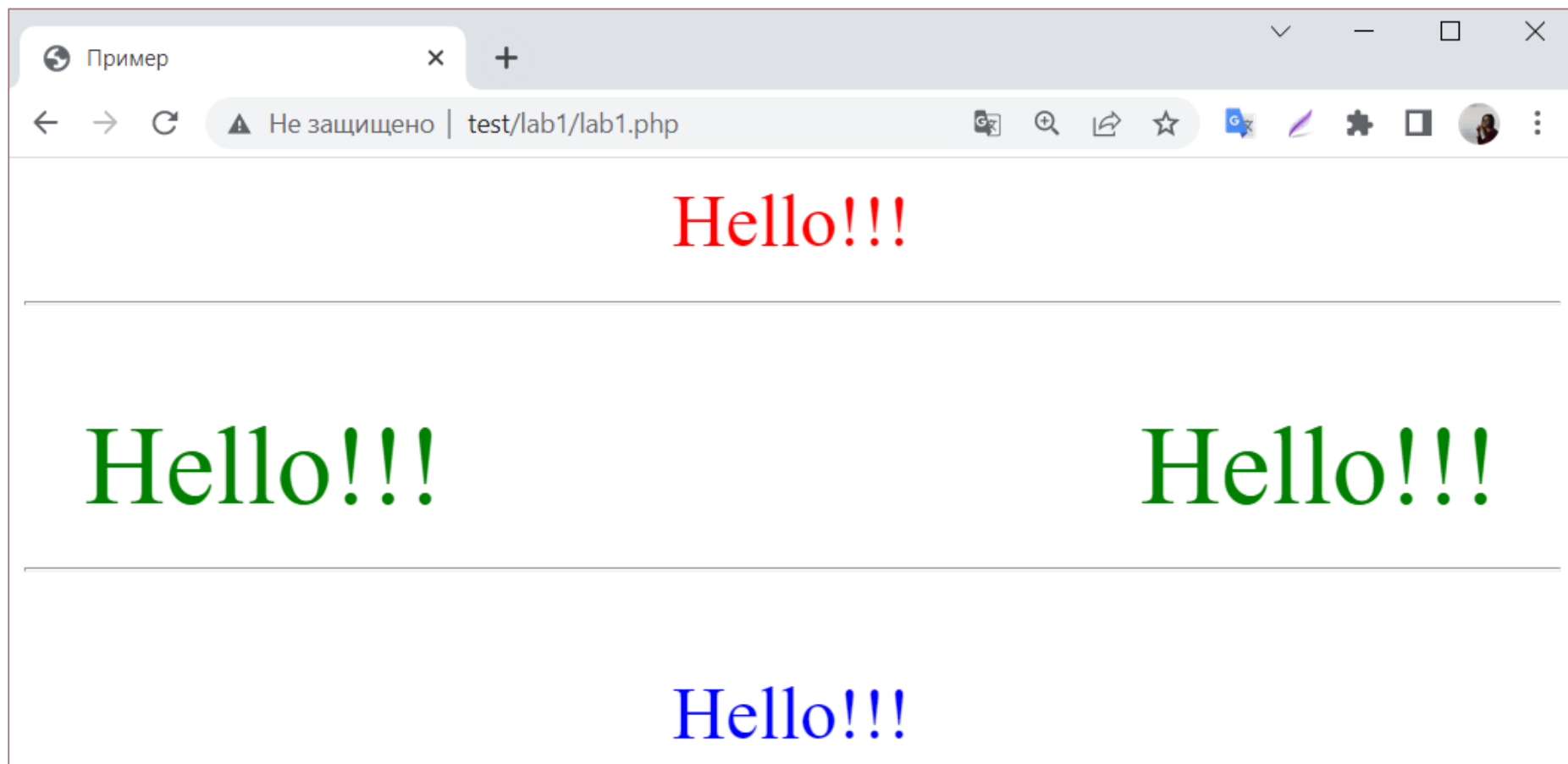
Примеры



Примеры

```
<p align="center">
    <span style="color: red; font-size: 2em">
        <?php echo "Hello!!!"; ?>
    </span>
<p>
<hr>
<br>
<p align="center">
    <span style="color: green; font-size: 3em">
        <?php echo "Hello!!!"; ?>
        &emsp;&emsp;&emsp;&emsp;&emsp;&emsp;
        <?php echo "Hello!!!"; ?>
    </span>
<p>
<hr>
<br>
<p align="center">
    <span style="color: blue; font-size: 2em">
        <?php echo "Hello!!!"; ?>
    </span>
<p>
```


Примеры



Примеры

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <title>Стили</title>
  <style type="text/css">
    H1 {
      font-size: 120%; /* Размер шрифта */
      font-family: Verdana, Arial, Helvetica, sans-serif; /* Семейство шрифта */
      color: #FF0000; /* Цвет текста */
      text-align: left;
    }
  </style>
  <style type="text/css">
    H2 {
      font-size: 160%; /* Размер шрифта */
      font-family: Verdana, Arial, Helvetica, sans-serif; /* Семейство шрифта */
      color: #00FF00; /* Цвет текста */
      text-align: center;
    }
  </style>
```

```
H3 {
    font-size: 200%; /* Размер шрифта */
    font-family: Verdana, Arial, Helvetica, sans-serif; /* Семейство шрифта */
    color: #0000FF; /* Цвет текста */
    text-align: right;
}
</style>
</head>
<body>
    <?php
        echo "<H1>";
        echo "HELLO WORLD!";
        echo "</H1>";

        echo "<H2>";
        echo "HELLO WORLD!";
        echo "</H2>";

        echo "<H3>";
        echo "HELLO WORLD!";
        echo "</H3>";
    ?>
</body>
</html>
```

Примеры

