

Лабораторная работа № 4

Основы PHP

1. Изучить основные языковые конструкции PHP.
2. Изучить включаемые файлы PHP.
3. Изучить простые функции для работы со строками.

В процессе изучения языка PHP используйте официальное руководство, которое вы сможете найти по адресу:

<http://php.net/manual/ru/>

1. Когда PHP обрабатывает файл, он ищет один из специальных тегов, который сообщает ему о необходимости начать интерпретацию текста как кода PHP. Затем он выполняет весь найденный код до закрывающего тега, говорящего интерпретатору, что далее снова идет просто текст. Этот механизм позволяет вам встроить PHP-код в HTML - все за пределами тегов PHP остается неизменным, тогда как внутри - интерпретируется как код.

Для включения PHP-код нужно использовать теги **<?php. . .?>**.

Файл, внутри которого содержится код PHP (не имеет значения, есть ли там кроме него HTML или нет) ДОЛЖЕН иметь расширение .php.

Пример 1

PHP	Интерпретация HTML
<pre><!doctype html> <html> <head> <title>Пример 1</title> <meta charset="utf8"> </head> <body> <?php echo ("Привет!"); ?> </body> </html></pre>	<pre><!doctype html> <html> <head> <title>Пример 1</title> <meta charset="utf8"> </head> <body> Привет! </body> </html></pre>

Результат

Привет!

Основные правила:

- выражения заканчиваются точкой с запятой (;);

- имена переменных начинаются с символа (\$), первым символом после знака доллара должна быть буква или символ подчеркивания, регистр имеет значение;
- блоки кода заключаются в фигурные скобки ({});
- комментарии в коде начинаются с символов // (для однострочных комментариев) или начинаются с /* и заканчиваются */ (для многострочных комментариев).

Нет необходимости объявлять переменные перед их использованием. Как правило, программист не устанавливает тип переменной; предпочтительнее, чтобы это делал PHP во время выполнения программы в зависимости от контекста, в котором используется переменная.

Пример 2

```
<!doctype html>
<html>
<head>
  <title>Пример 2</title>
  <meta charset="utf-8">
</head>
<?php
$night = "Добрый вечер!";
$morning = "Доброе утро!";
?>

<body>
  <?php if (date("H") > 18) {
    echo ($night); } else {
    echo ($morning);
  }
  ?>
</body>

</html>

<!doctype html>
<html>

<head>
  <title>Пример 2</title>
  <meta charset="utf-8">
</head>
<?php
if (date("H") > 18) { ?>
Добрый вечер!
<?php
} else {
?>
Доброе утро!
```

```
<?php
} ?>
</body>

</html>
```

Циклы организуются с помощью конструкций while, do-while, for, foreach.

Пример 3. Печать квадратов натуральных чисел от 1 до 20

```
<!doctype html>
<html lang="ru">

<head>
  <title>Пример 3</title>
  <meta charset="utf-8">
  <style>
    table {
      border: 1px solid olive;
      width: 350px;
      border-collapse: collapse
    }

    td {
      padding: 3px;
      border: 1px solid olive;
    }

    tr:first-child td {
      font-weight: bold;
    }

    tr {
      text-align: center;
    }
  </style>
</head>

<body>
  <table>
    <tr>
      <td> Число</td>
      <td>Квадрат</td>
    </tr>
    <?php for ($i = 1; $i <= 20; $i++) {
      echo "<tr><td>" . $i . "</td><td>" . $i * $i . "</td></tr>";
    }
  <?>
  </table>
</body>

</html>
```

2. Мощным механизмом PHP является использование включаемых файлов. В общем виде включение файла выглядит так:

```
<?php
```

```
include "имя_файла.php";
?>
```

Более подробно про процедуру включения файлов вы можете прочитать в руководстве по языку PHP (или лекции по PHP).

Очень часто на страницах сайта имеются повторяющиеся участки кода, например, заголовки страниц, определения классов и функций, наборы ссылок и т.д. В этом случае имеет смысл разместить их во включаемом файле. Тогда для изменения на всех страницах сразу достаточно будет изменить только сам включаемый файл.

3. Строка в PHP представляет собой, фактически, массив символов; обращаться к каждому символу можно по его порядковому номеру. Вот некоторые функции для работы со строками.

Функция / операция	Действие	Пример	Результат
.	Конкатенация	<code>\$a = \$b . \$c;</code>	<code>\$a = "Примертекста";</code>
<code>strlen()</code> <code>mb_strlen()</code>	Длина строки	<code>\$l = strlen(\$a)</code> <code>\$l = mb_strlen(\$a, 'UTF-8')</code>	<code>\$l = 12, 24 для UTF-8</code> <code>\$l = 12</code>
<code>strpos()</code>	Позиция в строке	<code>\$s = strpos(\$a, 'м');</code>	<code>\$s = 3 (4 для UTF-8)</code>
<code>strstr()</code>	Подстрока	<code>\$s = strstr(\$a, 'и');</code> <code>\$s = strstr(\$a, 'и', true);</code>	<code>\$s = "имертекста"</code> <code>\$s = "Пр"</code>

* для всех примеров: `$b="Пример"`; `$c="текста"`

Пример 4. Подсчёт символов "а", "о", "е", стоящих на нечётных позициях в строке

```
<!doctype html>
<html lang="ru">

<head>
  <title>Пример 4</title>
  <meta charset="utf-8">
  <style>
    html {
      font-family: "helvetica, sans-serif";
      font-size: 1.2em;
    }

    p {
      line-height: 1em;
    }
  </style>
</head>

<body>
  <?php
```

```

    $u_string = "Когда браузер запрашивает у Web-сервера файл данной страницы, PHP-
код проходит синтаксический анализ и выполняется, а результат помещает-ся в HTML-
поток, возвращаемый браузеру.";
    $us_string = "aeo";
    $a_string = iconv("utf-8", "windows-1251", $u_string);
    $as_string = iconv("utf-8", "windows-1251", $us_string);
    $counter = 0;
    $len = strlen($a_string);
    for ($i = 1; $i < $len; $i = $i + 2)
        if (
            $as_string[0] == $a_string[$i] || $as_string[1] == $a_string[$i] ||
            $as_string[2] == $a_string[$i]
        ) {
            $counter++;
            printf("<p>Позиция: %d, символ %s</p>", $i, iconv(
                "windows-1251",
                "utf-8",
                $a_string[$i]
            )
        );
        }
    echo "<p>Строка: " . $u_string . "</p>";
    echo "<p><em>Всего найдено символов: " . $counter . "</em></p>";
    ?>
</body>

</html>

```

Задание.

1. Написать PHP-скрипт lab4_1.php, который выводит числа от 1 до 20, их квадраты, кубы и квадратные корни из них (вывод осуществлять в таблице, аналогично примеру 3).
2. Реализовать весь PHP код из задания номер 1 в виде включаемого файла, поместить код в файл tab.php, включить файл в главный файл lab4_2.php.
3. Дана строка символов S (профиль, фамилия, имя, отчество) и строка символов P (четыре первых символа месяца рождения). Определить (lab4_3.php):
 - 1) общее количество вхождений символов P в S;
 - 2) количество вхождений P в S для чётных символов в S;
 - 3) количество вхождений каждого символа отдельно из P в S.

Для каждого варианта вывести все номера вхождений символов и сами символы, найденные количества.
4. Дана произвольная строка, содержащая буквы и цифры. Определить и вывести все цифры (lab4_4.php).
5. Перепишите на php задачу 6 из лабораторной работы № 3 (lab4_5.php).
6. Перепишите на php задачу 7 из лабораторной работы № 3 (lab4_5.php).