

## Лабораторная работа № 5

### Работа со строками и заголовком HTML документа

#### Задание

- В Вашей папке находящейся в корневой директории web-сервера создайте документ с названием l5.php.
- Задайте ему кодировку UTF-8.

Все следующие операции производятся в этом документе

#### Работа со строками

- Определить строку с использованием синтаксиса одинарных кавычек;
- Определить строку с использованием синтаксиса двойных кавычек;
- Определить строку с использованием HEREDOC синтаксиса;
- Создать массив из минимум трёх элементов и вывести его с использованием echo, print, print\_r, serialize пояснить полученные результаты;
- Создать строку `$message` следующего содержания:

```
<b>
  <div style='cursor:pointer;' onclick='javascript:alert(window.location);'>
    <font color='red'>
      Форум не поддерживает фильтрование даже простейших опасных конструкций
    </font>
  </div>
</b>
```

- Вывести эту строку с использованием `echo()`;
  - Вывести эту строку с использованием `htmlspecialchars()`;
- Вычислить контрольную сумму для строки используя функцию `CRC32()`;
- Вычислить хэш строки по алгоритму md5 используя функцию `md5()`;
- Вычислить хэш строки по алгоритму sha1 используя функцию `sha1()`;
- Создать строку следующего содержания: `"Толстой|Горький|Блок|Бунин|Державин";` С этой строкой:
  - Разбить в массив используя функцию `explode()`;
  - Вывести полученный массив в браузер.
- Применить к любой строке функцию `str_shuffle()` и вывести результат.
- Создать строку “Карл у Клары украл кораллы” и с использованием функции `substr()` вывести слово “украл”;

#### Работа с файлами

- Используя функцию  `fopen()` создать файл `file.txt` и записать в него строку “File was created”.
- Используя функцию `copy()` скопировать только что созданный файл, задав ему имя `file_copy.txt`.
- Проверить существование файла `file_copy.txt` функцией `file_exists()`.
- Дописать в файл `file_copy.txt` строку “Это вторая строка”.
- Удалить файл `file_copy.txt` используя функцию `unlink()`.



## Работа с заголовком HTML документа

### Функция header()

- Продемонстрировать возможность редиректа с использованием  
`header('Location: http://www.example.com/');`
- Вызвать формирование статусного кода 404  
`header("HTTP/1.0 404 Not Found");`
- Вызвать стандартный диалог требования аутентификации  
`header('WWW-Authenticate: Negotiate');`  
`header('WWW-Authenticate: NTLM', false);`
- Экспорт данных в Excel
  - Создать несколько строк в следующем формате:  
Identifier;Goodname;Price;Description (**Важно:** разделитель символ точки с запятой). **Пример:**  
`$outStr = "784;Flash drive;75Ls;Very cool thing\n";`  
`$outStr .= "545;CD-ROM drive;15Ls;Very cool CD thing\n";`
  - Выводить все строки, указав тип документа  
`header('Content-Type: text/csv');`  
и имя файла  
`header("Content-Disposition: attachment; filename=search_results.csv");`



## Справочная информация

### Строки

#### Пример: HEREDOC синтаксис

```
<?php
$str = <<<EOD
Пример строки,
охватывающей несколько строчек,
с использованием heredoc-синтаксиса.
EOD;
?>
```

#### Функции `serialize()` / `unserialize()`

Функции предназначены для создания строкового представления текущего состояния массива или объекта, которое затем можно передать куда-либо или сохранить. Соответственно `unserialize()` восстанавливает состояние массива / объекта из строки.

#### Пример: функции `serialize()` / `unserialize()`

```
$arr = array("a1", "a3", "a13");
print_r($arr);

$arrString = serialize($arr);
echo $arrString . "<br />";

$arr2 = unserialize($arrString);
print_r($arr2);
```

Функция `htmlspecialchars()` преобразует специальные символы в HTML сущности.

#### Пример: использование функции `htmlspecialchars()`

```
<?php
$new = htmlspecialchars("<a href='www.hackerssite.com/bad_script.php'>Страничка моей бабушки</a>");
echo $new; // &lt;a href='www.hackerssite.com/bad_script.php'&gt;Страничка моей бабушки&lt;/a&gt;?>
```

Функция `explode()` разбивает строку полученную во втором параметре по разделяющему символу полученному в первом параметре в простой индексированный массив.

#### Пример: Использование функции `explode()`

```
$params = "aa|ss|dd|ff|gg";
$arr = explode("|", $params);
print_r($arr);
```

Функция `string substr ( string $string, int $start [, int $length] )` возвращает подстроку строки `$string` начиная с `$start` элемента и до конца, либо `$length` символов если задано значение третьего, необязательного параметра.

#### Пример: использование функции `substr()`

```
<?php
$rest = substr("йцукен", 1); // возвращает " цукен "
$rest = substr("йцукен", 1, 3); // возвращает " цук "
?>
```



## Файлы

Функция `resource fopen ( string $filename, string $mode [, bool $use_include_path [, resource $context]] )` открывает файл `$filename`, т.е. возвращает указатель на него, в помощью которого потом с файлом можно производить нужные действия.

Может работать в нескольких режимах определяемых параметром `$mode`.

- 'r' Открывает файл только для чтения; помещает указатель в начало файла.
- 'w' Открывает файл только для записи; помещает указатель в начало файла и обрезает файл до нулевой длины. Если файл не существует - пробует его создать.
- 'w+' Открывает файл для чтения и записи; помещает указатель в начало файла и обрезает файл до нулевой длины. Если файл не существует - пробует его создать.
- 'a' Открывает файл только для записи; помещает указатель в конец файла. Если файл не существует - пытается его создать.

**Пример:** простая запись в файл строки

```
$outStr      = «Some string to write»;  
$outFile     = fopen("data.js","w+");  
fwrite($outFile,$outStr);  
fclose($outFile);
```

Функция `bool copy ( string $source, string $dest )` копирует `$source` файл и задаёт ему новое название `$dest`. Возвращает FALSE в случае ошибки.

Функция `bool file_exists ( string $filename )` проверяет существование файла `$filename`, возвращает TRUE, если файл существует.

Функция `bool unlink ( string $filename )` пытается удалить файл `$filename`.

**Внимание,** для удаления директорий не подходит!