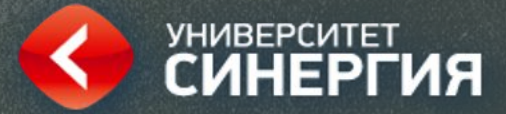


Работа с файлами

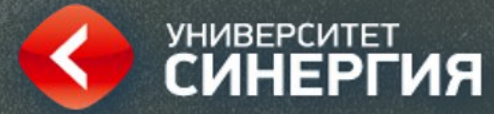
Преподаватель – Борисова Татьяна Андреевна

Сегодня на уроке:



- 1) Запись в файл;
- 2) Добавление в файл;
- 3) Чтение файла;
- 4) Обсуждение домашнего задания;

Запись в файл

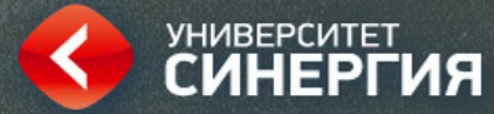


PHP предлагает ряд функций, которые можно использовать при создании, чтении, загрузке и редактировании файлов.

Функция `fopen()` создает или открывает файл.

Если вы используете функцию `fopen()` с файлом, которого не существует, файл будет создан с учетом того, что файл был открыт для записи (**w**) или добавления (**a**).

Запись в файл



Используйте один из следующих режимов для открытия файла:

r: открывает файл только для чтения.

w: открывает файл только для записи. Стирает содержимое файла или создает новый файл, если он не существует.

a: открывает файл только для записи.

x: создает новый файл только для записи.

r+: открывает файл для чтения/записи.

w+: открывает файл для чтения/записи. Стирает содержимое файла или создает новый файл, если его нет.

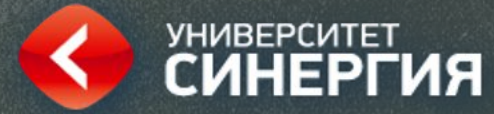
a+: открывает файл для чтения/записи. Создает новый файл, если он не существует.

x+: создает новый файл для чтения/записи.

В примере ниже создается новый файл **"file.txt"**, который будет создан в той же директории, где находится PHP-код:

```
$myfile = fopen("file.txt", "w");
```

Запись в файл



При записи в файл используйте функцию `fwrite()`.
Первый параметр функции `fwrite()` - это файл, в который нужно записывать; второй параметр - это строка, которую нужно записать.

В примере на следующем слайде пара имен в новый файл под названием **"names.txt"**:

Обратите внимание, что мы дважды писали в файл **"names.txt"**, а затем использовали функцию `fclose()` для закрытия файла.

Примечание: Символ `\n` используется для перехода на новую строку.

Запись в файл

```
<?php  
  
$myfile = fopen("names.txt", "w");  
  
$txt = "John\n";  
  
fwrite($myfile, $txt);  
  
$txt = "David\n";  
  
fwrite($myfile, $txt);  
  
fclose($myfile); /* Файл содержит: John David */  
  
?>
```

`fclose()`

Функция `fclose()` закрывает открытый файл и возвращает `TRUE` в случае успеха или `FALSE` в случае неудачи.

Примечание :

Хорошей практикой является закрытие всех файлов после того, как вы закончили работу с ними.

Добавление в файл

Если вы хотите добавить данные в файл, вам нужно открыть файл в режиме добавления.

Например:

```
$myFile = "test.txt";  
  
$fh = fopen($myFile, 'a');  
  
fwrite($fh, "Some text");  
  
fclose($fh);
```

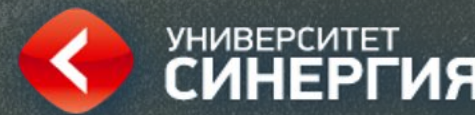
При добавлении в файл с помощью режима 'a' указатель на файл помещается в конец файла, обеспечивая добавление всех новых данных в конец файла.

Добавление в файл

Создадим пример формы, добавляющей в файл внесенные в нее данные:

```
<?php
if(isset($_POST['text'])) {
    $name = $_POST['text'];
    $handle = fopen('names.txt', 'a');
    fwrite($handle, $name."\n");
    fclose($handle);
}
?>
<form method="post">
    Name: <input type="text" name="text" />
    <input type="submit" name="submit" />
</form>
```

Добавление в файл



Теперь каждый раз, когда имя вводится в форму и отправляется, оно добавляется в конец файла "**names.txt**" в виде новой строки.

Функция `isset()` определяет, была ли форма отправлена, а также содержит ли она текст.

Мы не указывали атрибут **action** для формы, поэтому она будет отправлена сама себе.

Чтение файла

Функция `file()` считывает весь файл в массив. Каждый элемент внутри массива соответствует строке в файле:

```
$read = file('names.txt');  
  
foreach ($read as $line) {  
  
    echo $line .", ";  
  
}
```

В этом примере выводятся все строки из файла, разделенные запятыми.

Мы использовали цикл **foreach**, потому что переменная **\$read** - это массив.

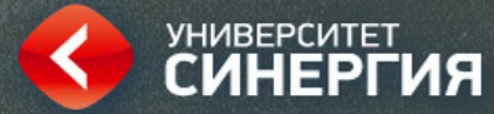
Чтение файла

В конце вывода в предыдущем примере у нас была бы запятая, так как мы выводим ее после каждого элемента массива.

Код на следующем слайде позволяет избежать печати этой последней запятой:

```
$read = file('names.txt');  
$count = count($read);  
$i = 1;  
foreach ($read as $line) {  
    echo $line;  
    if($i < $count) {  
        echo ', '  
    }  
    $i++;  
}
```


Чтение файла



Переменная **\$count** использует функцию `count()` для получения количества элементов в массиве **\$read**.

Затем, в цикле **foreach**, после каждой распечатки строк, мы определяем, меньше ли номер текущей строки, чем общее количество строк, и выводим запятую, если это так.

Примечание:

Это позволяет избежать вывода последней запятой, так как для последней строки **\$i** равен **\$count**.

Домашнее задание:



Выполнить тест по пройденному материалу.

Результаты прислать в личные сообщения в тг.

*СПАСИБО ЗА
ВНИМАНИЕ!*

