

# **МДК 02.02. Web-программирование. Язык PHP**

**Работа с файловой системой**

## **Способы сохранения данных:**

1. сохранение в файл;
2. сохранение в базу данных.

## **Работа с файлами**

1. открытие файла;
2. обработка (чтение, запись);
3. закрытие файла.

## Пример создания файла

```
<?php
$fh = fopen("testfile.txt", 'w') or die("Создать файл не
удалось");
$text=<<<<EOD
Строка 1
Строка 2
Строка 3
EOD;
fwrite($fh, $text) or die("Сбой записи файла");
fclose($fh);
echo "Файл 'testfile.txt' записан успешно ";
?>
```

## Проверка существования файла

Функция `file_exists` позволяет проверить факт существования файла, возвращает либо TRUE, либо FALSE.

### Пример

```
if (file_exists("testfile.txt"))  
echo "Файл существует";  
else  
echo "Файл не существует";
```

## Создание файла

**Функция fopen(<Путь к файлу>, <Режим>)**  
открывает файл и возвращает дескриптор  
(идентификатор).

Если файл находится в текущей директории, то достаточно указать только его имя (без пути).

## Режимы работы, поддерживаемые функцией fopen

Режим	Действие	Описание
r	Чтение с начала файла	Открытие файла только для чтения; установка указателя файла на его начало.
r+	Чтение с начала файла с возможностью записи	Открытие файла для чтения и записи; установка указателя файла на его начало.
w	Запись с начала файла с усечением его размера	Открытие файла только для записи; установка указателя файла на его начало и сокращение размера файла до нуля
w+	Запись с начала файла с усечением его размера и возможностью чтения	Открытие файла для чтения и записи; установка указателя файла на его начало и сокращение его размера до нуля
a	Добавление к концу файла	Открытие файла только для записи; установка указателя файла на его конец.
a+	Добавление к концу файла с возможностью чтения	Открытие файла для чтения и записи; установка указателя файла на его конец.

**В режимах w, w+, a, a+ если файл не существует, то он будет создан.**

**Функция (конструкция) `exit()`, `die()` заканчивает выполнение скрипта и выдает пользователю сообщение об ошибке.**



## Запись в файл

1. **Функция fwrite(<Дескриптор>, <Строка>)** - записывает данные в файл, возвращает количество записанных байт.

### Пример

```
<?php  
$fp = fopen('testfile.txt', 'w');  
echo fwrite($fp, '234');  
fclose($fp);  
>
```

**2. Функция fputs()** – аналогична fwrite, первый параметр - указатель на ресурс, второй - строка, которую нужно записать.

**3. Функция file\_put\_contents()** принимает строку - имя файла и строку, которую необходимо записать.

Функция идентична последовательным успешным вызовам функций fopen(), fwrite() и fclose().

## Пример

```
<?php  
$file = "testfile.txt";  
$current = "John";  
echo file_put_contents($file, $current);  
?>
```

## Чтение файла

1. Функция `fgets(<Дескриптор>[, <Длина в байтах>])`  
– читает строку из файла.

### Пример

```
<?php
$fh = fopen("testfile.txt", 'r') or
die("Файл не существует или вы не обладаете правами на
его открытие");
$line = fgets($fh);
fclose($fh);
echo $line;
?>
```

**2. Функция `file_get_contents()` принимает имя файла и возвращает его содержимое одной строкой.**

### **Пример**

```
<?php  
echo file_get_contents("testfile.txt");  
?>
```

**3. Функция fread(<Дескриптор>, <Длина в байтах>)** позволяет прочитать из файла строку указанной длины.

### **Пример**

```
<?php
$filename = 'testfile.txt';
$handle = fopen($filename, "r");
echo $contents = fread($handle, filesize($filename));
fclose($handle);
?>
```

**Функция filesize()** – позволяет получить размер файла

**4. Функция readfile(<Путь к файлу>) – открывает файл и выводит все его содержимое в окно Web-браузера.**

### **Пример**

```
<?php  
echo $contents = readfile('testfile.txt');  
?>
```

## Копирование файлов

**Функция copy()** – копирует файл

### Пример

```
<?php  
copy('testfile.txt', 'testfile2.txt') or die("Копирование  
невозможно");  
echo "Файл успешно скопирован в 'testfile2.txt';  
?>
```



## Переименование файла

**Функция rename()** – служит для переименования файла

### Пример

```
<?php  
if (!rename('testfile2.txt', 'testfile2.new'))  
echo "Переименование невозможно";  
else echo "Файл успешно переименован в 'testfile2.new';  
?>
```

## Удаление файла

**Функция unlink ()** – служит для удаления файла

### Пример

```
<?php  
if (!unlink('testfile2.new')) echo "Удаление невозможно ";  
else echo "Файл 'testfile2.new' удален успешно";  
?>
```

## Заккрытие файла

**Функция fclose()**

## Функции для работы с каталогами

1. Функция `mkdir(<Имя каталога>)` создает **новый** каталог

### Пример

```
<?php  
$a='./11';  
mkdir($a);  
?>
```

2. **Функция `rmdir(<Имя каталога>)`** удаляет пустой каталог. Если в каталоге есть файлы, то каталог удален не будет.
3. **Функция `getcwd()`** возвращает текущий каталог.
4. **Функция `chdir(<Имя каталога>)`** делает указанный каталог текущим.
5. **Функция `opendir(<Имя каталога>)`** открывает каталог для чтения. Функция возвращает дескриптор, который указывается в других функциях.
6. **Функция `closedir(<Дескриптор>)`** закрывает каталог.