

Загрузка файлов на сервер с помощью PHP

Кочитов Михаил Евгеньевич

Приамурский государственный университет им. Шолом-Алейхема

студент

Аннотация

В данной статье рассматривается загрузка различных файлов на сервер, используя язык программирования PHP и язык разметки HTML. В статье будет разработан собственный пример с загрузкой файлов на локальный сервер.

Ключевые слова: загрузка файлов, сервер, HTML, PHP

Upload files to the server using PHP

Kochitov Mikhail Evgenevich

Sholom-Aleichem Priamursky State University

student

Abstract

This article discusses uploading various files to the server using the PHP programming language and the HTML markup language. The article will develop its own example with uploading files to the local server.

Keywords: upload files, server, HTML, PHP

Пользователи сайтов каждый день загружают на различные сайты файлы, например фотографии, видеофайлы, аудиофайлы, статьи, научные работы и т.д. Эти сайты имеют свои сервера для хранения загружаемых от пользователей файлов, чтобы можно было их отображать на определенных веб-страницах сайтов, а также предоставлять возможность пользователям скачивать их себе на устройство или компьютер. В языке программирования PHP имеется встроенный функционал для загрузки файлов любого типа и дальнейшей их обработки и хранению на сервере.

Целью данной статьи является загрузка различных файлов на сервер, используя язык программирования PHP и язык разметки HTML. Также в статье будет разработан собственный пример с загрузкой файлов на локальный сервер.

В статье Р.П. Калашникова рассматривается разработка информационной системы генерирования XML-файлов и загрузка информации в базу данных [1]. А.О. Репкин и А.Н. Тихомирова рассмотрели реализацию непрерывной загрузки файлов больших размеров на удаленное хранилище в WEB приложении [2]. Рассматривая книгу Р.Р. Лукманова можно заметить автоматизацию загрузки изображений и других файлов на

хостинг [3]. Я.А. Береза, А.Ю. Барсуков и А.Г. Пимонов рассмотрели программные средства для настройки файла загрузки рекламных компаний Яндекс.Директ и Google AdWords [4].

Теперь приступим к разработке собственного примера загрузки различных файлов на локальный сервер с помощью языка программирования PHP и языка разметки HTML. Для начала на веб сервере создадим файл под названием “index.php” в нем напишем сперва HTML код, представленный на рисунке ниже

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Загрузка файлов на сервер</title>
</head>
<body>
  <form method="post" action="index.php" enctype="multipart/form-data">
    <input type="file" name="myFile">
    <input type="submit" value="Загрузить файл">
  </form>
</body>
</html>
```

Рис. 1. HTML код формы загрузки файла

На рисунке 1 представлен HTML код формы загрузки файла, теперь рассмотрим его более детально. Глядя на сам код, то можно увидеть стандартную структуру построения любой HTML страницы из тегов: <html>, <head> и <body>. В теге <head> указывается два тега <meta>, который дает веб странице с помощью параметра “charset” установить кодировку UTF-8 во избежание неизвестных символов на самой веб странице “index.php”. Последний тег <title> указывает название самой веб странице, которая в основном пишется на вкладке любого браузера. Далее перейдем к тегу <body> в нем видно тег самой формы <form> он настроен с помощью параметра “enctype”, что дает самой форме принимать выбираемый файл пользователем и загружать его непосредственно на сервер. Главный тег, который отображает выбор файла на странице является <input>, где в параметре “type” пишется значение “file”, что дает браузеру понять, что нужно отобразить кнопку “Выбрать файл”. Параметр “name” задает уникальное имя, чтобы можно было обращаться к нему в дальнейшем на языке PHP, в нашем случае уникальное имя - “myFile” Следующий тег <input> у которого параметр “type” теперь имеет значение “submit”, который дает браузеру отобразить на странице кнопку отправки данных и загрузки файла на локальный сервер, последний параметр “value” означает название самой кнопки.

Теперь глянем в самом браузере как будет выглядеть форма загрузки файла (Рис. 2)

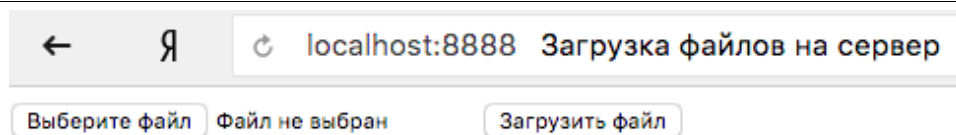


Рис. 2. Форма загрузки файла в браузере

Далее идет уже главный PHP код получения и обработки файла на сервере, он представлен на рисунке ниже

```
<?php
if (isset($_GET['file'])) {
    unlink("files/".$_GET['file']);
    echo "<script>location.assign(\"index.php\");</script>";
}

if (is_uploaded_file($_FILES['myFile']['tmp_name'])) {
    echo "<br>Вы загрузили файл: <br>";
    echo "Название файла: " . $_FILES['myFile']['name'] . "<br>";
    echo "Тип файла: " . $_FILES['myFile']['type'] . "<br>";
    echo "Размер файла: " . floor($_FILES['myFile']['size'] / 1024) . " килобайт <br>";

    move_uploaded_file($_FILES['myFile']['tmp_name'], "files/" . $_FILES['myFile']['name']);
}

echo "<br>Загруженные файлы:<br>";
$dir = scandir("files");
foreach ($dir as $num_file => $name_file) {
    if (($name_file != ".") and ($name_file != "..") and ($name_file != ".DS_Store"))
    {
        echo "<a href=\"files/".$name_file."\">".$name_file."</a> <a href=\"index.php?file=".$name_file."\">
        Удалить</a><br>";
    }
}
?>
```

Рис. 3. PHP код обработки загружаемого файла на сервер

На рисунке 3 представлен PHP код обработки загружаемого файла на сервер. Теперь глянем этот весь код немного подробнее. В начале видно условие “isset(\$_GET[‘file’])” оно выполняется если существует эта переменная \$_GET[‘file’], в которой содержится имя файла, который надо удалить из сервера. Эту переменную можно получить GET запросом через адресную строку в браузере. Далее идет функция “unlink”, которая удаляет сам файл. После функция “echo” имеет в значении функцию JavaScript - “location.assign”, эта функция отправляет команду браузеру немедленно обновить страницу без вмешательства пользователя, это сделано для того, чтобы очистить содержимое переменной \$_GET[‘file’].

Далее идет условие “is_uploaded_file” в нем пишется глобальный массив загружаемых файлов на сервер - \$_FILES. Первый индекс означает уникальное имя, которое было задано в HTML форме в теге <input>, второй индекс - ключевой индекс, в котором содержится временной путь хранения загруженного файла на сервере. Само условие “is_uploaded_file” выполняется если файл был успешно загружен на сервер. Теперь перейдем дальше, как видно идут одни функции “echo” для отображения информации загруженного файла на страницу. Теперь рассмотрим подробнее. В массиве

`$_FILES` для каждого загружаемого файла выделяются стандартные параметры (индексы массива): `name` - название загруженного файла с расширением, `type` - тип файла и `size` - размер файла в байтах. На рисунке 3, где отображается в функции “echo” размер файла было произведено математическое деление текущего размера в байтах на 1024, чтобы получить килобайты и функция “floor”, которая округляет полученный размер вниз до целого числа. Последняя функция в этом условии - “move_uploaded_file” перемещает загруженный файл из временного хранилища в само хранилище сервера в папку “files”.

Далее последний кусок кода отображает на странице браузера все загруженные файлы на сервере для их просмотра или удаления их из сервера. Функция “scandir” получает массив всех файлов и каталогов в указанном ее параметре. В нашем случае необходимо получить список всех файлов в папке “files”, которая находится на локальном сервере. Функция “foreach” используется только для массивов и предназначена в полном отображении всех значений по каждому индексу массива. Иными словами, для отображения всего содержимого массива. В нашем случае будут отображены все файлы в папке “files”. Также внутри “foreach” есть условие которое проверяет, чтобы не отобразить в браузере страницы несуществующие файлы на локальном сервере в папке “files”. К каждому файлу приписывается ссылка “Удалить”, в которой передается браузеру в адресную строку параметр “file”, чтобы исполнить первое условие в данном PHP коде и удалить выбранный файл функцией “unlink”.

Напоследок осталось проверить результат работы загрузки файлов на сервер в самом браузере (Рис. 4)

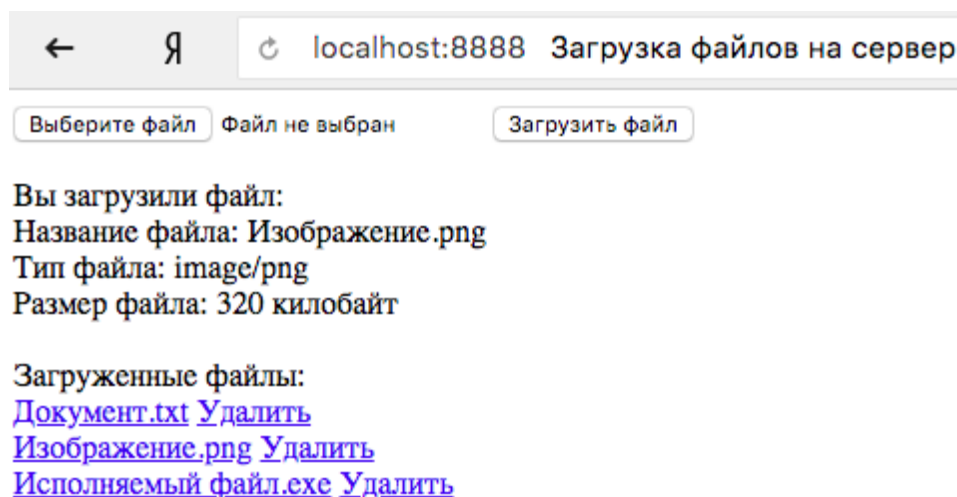


Рис. 4. Результат работы загрузки файлов на сервер в браузере

Как видно на рисунке 4 изображен результат работы загрузки файлов на сервер на самой странице браузера. Сверху видно саму HTML форму загрузки файла, ниже информацию о последнем загруженном файле на

сервер и в конце список всех загруженных файлов на сервер с возможностью их скачивания, просмотра и удаления из локального сервера.

Таким образом была рассмотрена загрузка файлов на сервер с помощью языка программирования PHP, а также был разработан собственный пример с загрузкой файлов на локальный сервер. Можно в заключении сказать, что данная возможность необходима тем сайтам и социальным сетям, которые имеют большой функционал для зарегистрированных пользователей - изменить аватарку, загружая свою новую на сервер, добавление собственных медиа-файлов либо различных работ и т.д.

Библиографический список

1. Калашников Р.П. Разработка информационной системы генерирования XML-файлов и загрузка информации в базу данных // В сборнике: Информационные и измерительные системы и технологии Сборник научных статей по материалам Международной научно-технической конференции. 2016. С. 356-358.
2. Репкин А.О., Тихомирова А.Н. Реализация непрерывной загрузки файлов больших размеров на удаленное хранилище в WEB приложении // В сборнике: Вопросы образования и науки: теоретический и методический аспекты сборник научных трудов по материалам Международной заочной научно-практической конференции: в 7 частях. 2012. С. 95-96.
3. Лукманов Р.Р. Автоматизация загрузки изображений и других файлов на хостинг // В книге: Студент и аграрная наука Материалы VI Всероссийской студенческой конференции. Министерство сельского хозяйства Российской Федерации, Министерство образования Республики Башкортостан, ФГБОУ ВПО Башкирский государственный аграрный университет, Совет молодых ученых университета. 2012. С. 265.
4. Берёза Я.А., Барсуков А.Ю., Пимонов А.Г. Программные средства для настройки файла загрузки рекламных кампаний Яндекс.Директ и Google AdWords // В сборнике: Сборник материалов IX Всероссийской научно-практической конференции молодых ученых с международным участием "Россия молодая" Конференция проходит при поддержке Российского фонда фундаментальных исследований. Ответственный редактор Костюк Светлана Георгиевна . 2017. С. 41003.
5. Загрузка файлов на сервер в PHP URL: http://komotoz.ru/uroki/php/zagruzka_fajlov_na_server.php (дата обращения 09.08.2018)
6. Загрузка файлов на сервер PHP URL: <http://snipp.ru/view/115> (дата обращения 09.08.2018)