**Практическая работа PHP и MySql**

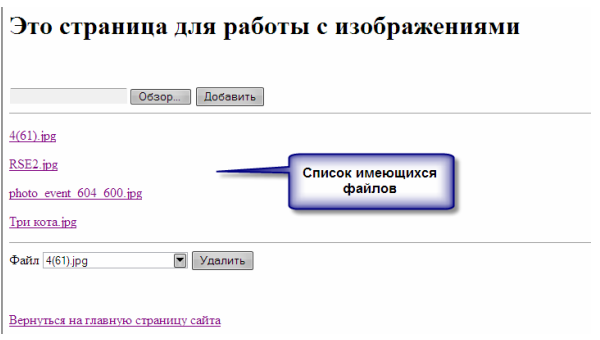
**Работа с файлами**

1. Изучить материал: <https://metanit.com/php/tutorial/5.4.php>
2. Выполнить все примеры из  теории.
3. Выполнить следующие упражнения:

**Упражнение 1. Вывод списка файлов**

1. Создайте на локальном узле (в нашем случае в вашей личной папке на сервере)  папки photo и files для размещения изображений и файлов соответственно. Поместите в эти папки несколько изображений и документов.

2. Создайте страницу photo.php, разместите на ней поясняющий текст, ниже поясняющего текста – две горизонтальные линии (между этими линиями будет выводиться список имеющихся на web-узле изображений и ссылками на эти изображениями), см. рис.



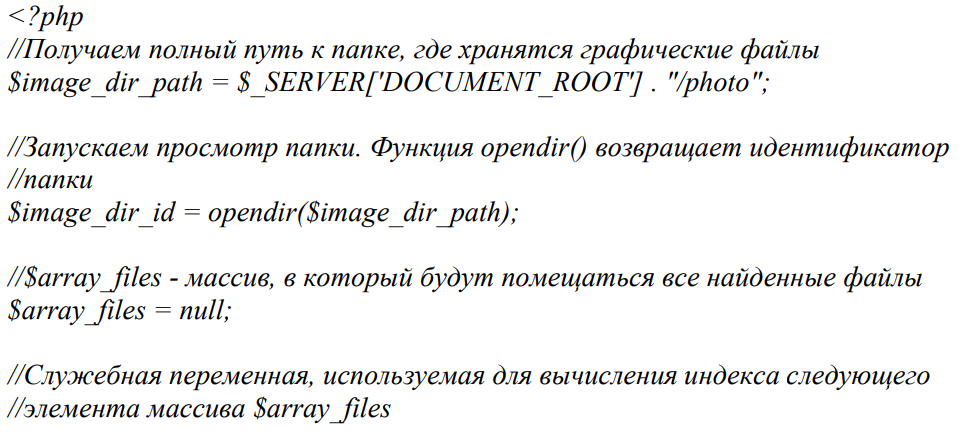
3. Для вывода списка файлов, уже имеющихся на сервере, необходимо создать два сценария:

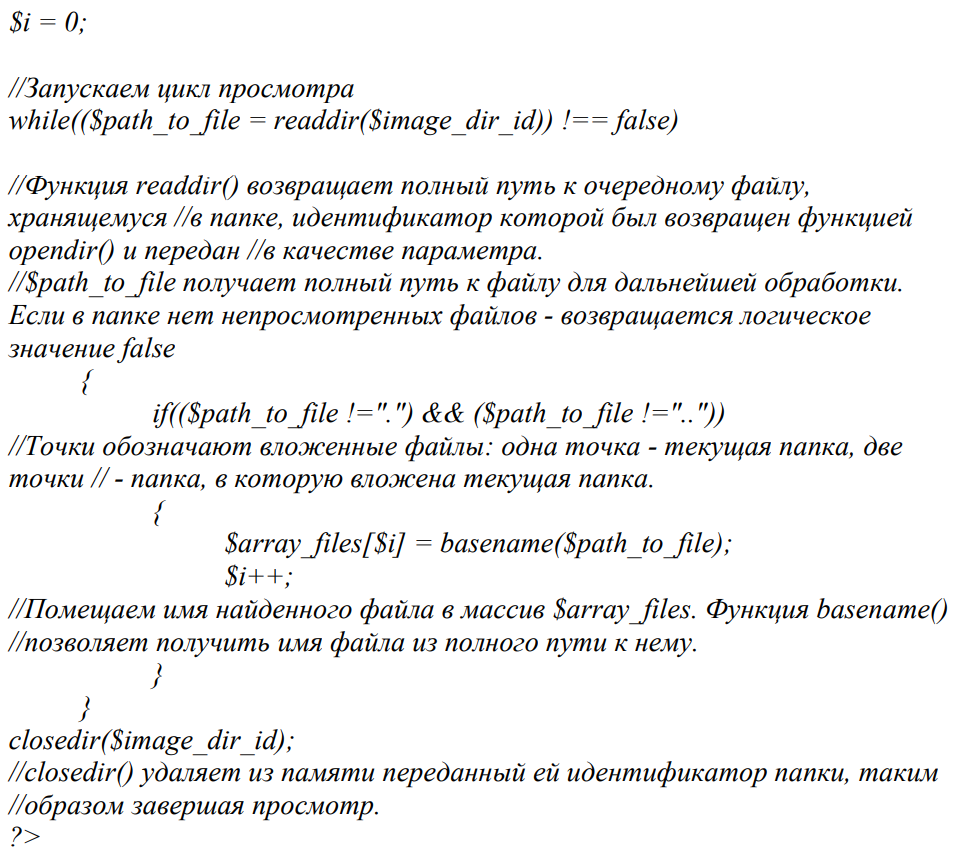
• первый будет формировать список файлов с изображениями,

• второй - выводить этот список на страницу.

3.1. Формирование списка файлов. Для формирования списка файлов необходимо Получить путь к целевой папке, где хранятся необходимые файлы, а затем создать массив, в который будет помещаться необходимая информация о файлах (имена, пути к ним);

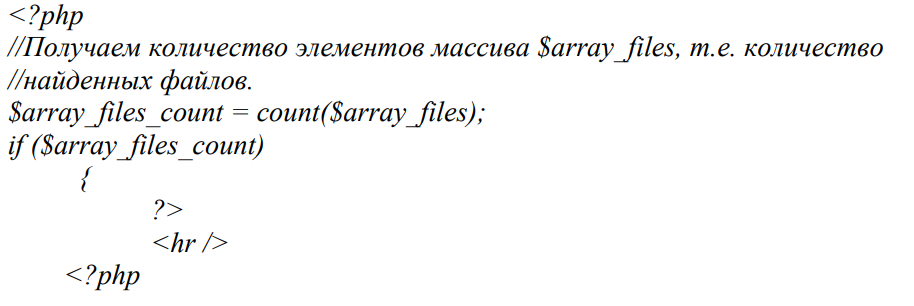
Вариант реализации кода

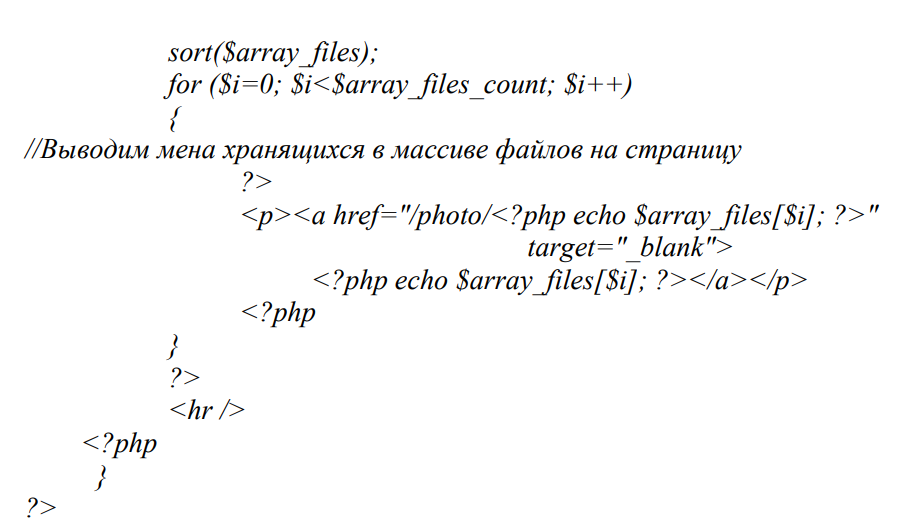




3.2. Вывод списка файлов на страницу.

3.1.1. Найдите код, создающий две горизонтальные линии. Создайте следующий сценарий вывода списка файлов:





3.1.2. Создайте гиперссылку возврата на главную страницу и гиперссылку с главной страницы на страницу photo.php («Фото» в наборе гиперссылок).

4. Проверьте работу сценариев.

**Упражнение 2. Отправка файлов на сервер**

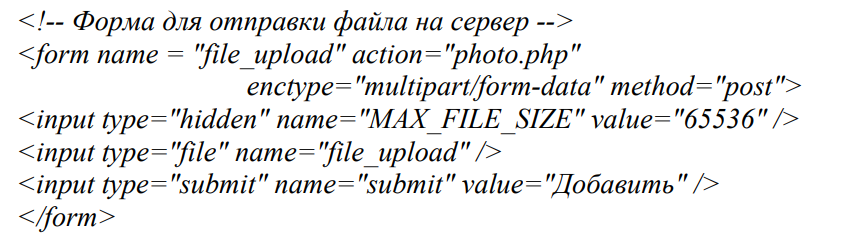
Отправка файлов на web-сайт состоит из двух этапов:

• создание необходимой формы для отправки файлов на странице сайта;

• написание сценария PHP для получения отправленного посетителем сайта файла.

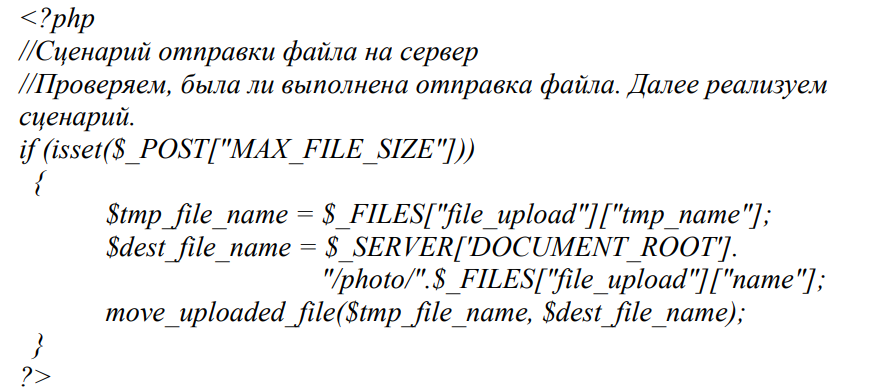
1*. Создание формы.* Стандарт HTML предусматривает т.н. поле выбора файла, в которое посетитель должен будет ввести имя отправляемого файла. От обычного поля ввода оно отличается тем, что позволяет работать с окном открытия файлов Windows, а также оно отправляет серверной программе не введенное в него имя файла, а сам файл. Поле выбора файла создается с помощью тэга INPUT, атрибут type которого имеет значение “file”. Также в форму с данным полем необходимо добавить скрытое поле, задающее максимальный размер отправляемого файла, с именем MAX\_FILE\_SIZE. Значение этого поля (т.е. максимально возможный размер файла) определяется в байтах.

Форма, из которой будет отправляться файл, должна кодировать данные по методу multipart/form-data и передавать данные только методом POST.



В данном случае максимальный размер файла равен 65536 байта, т.е. 64 Кбайта. Данный размер при необходимости может быть увеличен.

*2. Получение отправленного файла.* Все принятые файлы помещаются интерпретатором PHP в особую служебную папку, которая не является частью сайта (т.н. «буферная» папка). Сведения обо всех принятых и помещенных в буферную папку файлах хранятся во встроенном массиве PHP $\_FILES. Каждый элемент массива соответствует принятому файлу и представляет собой вложенный массив, содержащий различные сведения о файле. В начало страницы поместите следующий код:



Функция **move\_uploaded\_file ( string filename, string destination )** проверяет, является ли файл filename загруженным на сервер (переданным по протоколу HTTP POST). Если файл действительно загружен на сервер, он будет перемещён в место, указанное в аргументе destination.

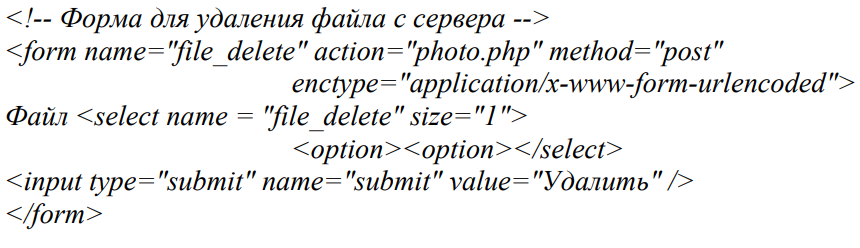
Упражнение 3. Удаление файла с сервера

Для удаления файла надо выбрать необходимый файл. Это делается с помощью формы, в которую помещается список для выбора удаляемого файла из имеющихся на сервере файлов. Для заполнения этого списка необходимо использовать уже созданный массив $array\_files.  Реализация удаления происходит в два этапа:

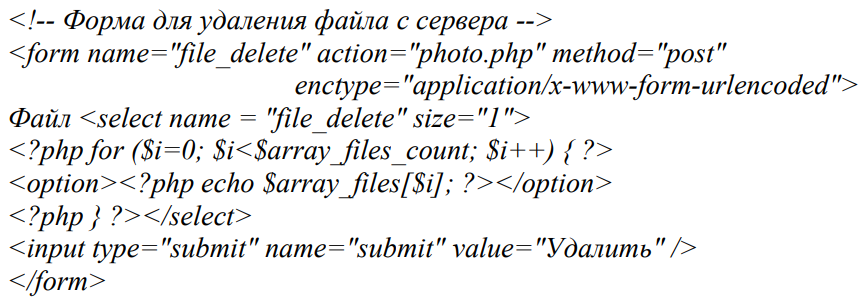
• Создание формы для удаления файла;

• Создание сценария удаления файла.

1. Создание формы для выбора удаляемого файла

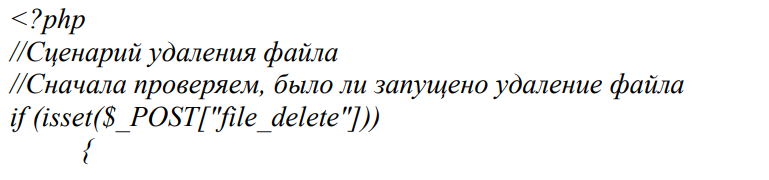


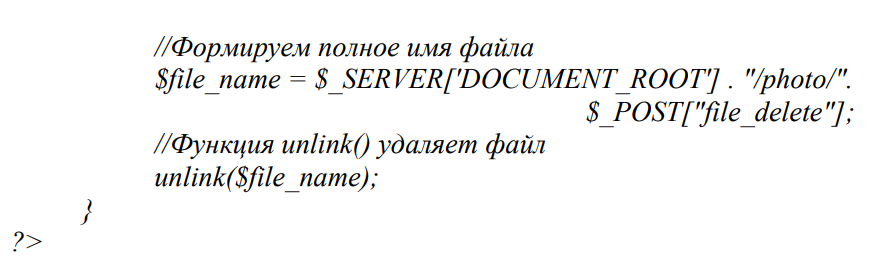
Парные тэги SELECT создает список. Пункты списка создаются тэгами OPTION. Сам список имеющихся на сервере файлов необходимо получить из созданного ранее массива $array\_files



Во включенном в форму цикле создается столько пунктов (тэгов), сколько элементов присутствует в массиве $array\_files.

2. Создание сценария удаления файла.





3. Сохраните изменения, проверьте корректность работы.

**Упражнение 4. Создание страницы для работы с файлами**

Самостоятельно создайте страницу для управления прочими файлами files.php. Она во всем аналогична странице photo.php, кроме:

1. Она должна обрабатывать файлы, находящиеся в папке files.

2. Просмотр файлов делать необязательно (это не изображения), следовательно не надо делать гиперссылки из имени каждого файла.

3. Необходимо увеличить максимальный размер файла хотя бы до 1 Мбайта.

4. Сделайте гиперссылки, связывающие данную страницу с главной страницей.