PHP File Handling

PHP File System allows us to create file, read file line by line, read file character by character, write file, append file, delete file and close file.

PHP Open File - fopen()

The PHP fopen() function is used to open a file.

Syntax

```
resource fopen ( string $filename , string $mode [, bool $use_include_path = false [, resource $context ]] )
```

Example

Skip Ad

```
<?php
$handle = fopen("c:\\folder\\file.txt", "r");
?>
```

Click me for more details...

PHP Close File - fclose()

The PHP fclose() function is used to close an open file pointer.

Syntax

```
ool fclose ( resource $handle )
```

Example

```
<?php
fclose($handle);
?>
```

PHP Read File - fread()

The PHP fread() function is used to read the content of the file. It accepts two arguments: resource and file size.

Syntax

```
string fread ( resource $handle , int $length )
```

Example

```
<?php
$filename = "c:\\myfile.txt";
$handle = fopen($filename, "r");//open file in read mode

$contents = fread($handle, filesize($filename));//read file

echo $contents;//printing data of file
fclose($handle);//close file
?>
```

Output

```
hello php file
```

Click me for more details...

PHP Write File - fwrite()

The PHP fwrite() function is used to write content of the string into file.

Syntax

```
int fwrite ( resource $handle , string $string [, int $length ] )
```

Example

```
<?php
$fp = fopen('data.txt', 'w');//open file in write mode
fwrite($fp, 'hello ');
fwrite($fp, 'php file');
fclose($fp);

echo "File written successfully";
?>
```

Output

```
File written successfully
```

Click me for more details...

PHP Delete File - unlink()

The PHP unlink() function is used to delete file.

Syntax

```
bool unlink ( string $filename [, resource $context ] )
```

Example

```
<?php
unlink('data.txt');

echo "File deleted successfully";
?>
```

Click me for more details...

PHP Open File

PHP fopen() function is used to open file or URL and returns resource. The fopen() function accepts two arguments: $filename and $mode. The $filename represents the file to be opended and $mode represents the file mode for example read-only, read-write, write-only etc.

Syntax

```
resource fopen ( string $filename , string $mode [, bool $use_include_path = false [, resource $context ]] )
```

PHP Open File Mode

Mode	Description
r	Opens file in **read-only** mode. It places the file pointer at the beginning of the file.

r+	Opens file in **read-write** mode. It places the file pointer at the beginning of the file.
w	Opens file in **write-only** mode. It places the file pointer to the beginning of the file and truncates the file to zero length. If file is not found, it creates a new file.
w+	Opens file in **read-write** mode. It places the file pointer to the beginning of the file and truncates the file to zero length. If file is not found, it creates a new file.
	Opens file in **write-only** mode. It places the file pointer to the end of the file. If file is not found, it creates a new file.
a+	Opens file in **read-write** mode. It places the file pointer to the end of the file. If file is not found, it creates a new file.
x	Creates and opens file in **write-only** mode. It places the file pointer at the beginning of the file. If file is found, fopen() function returns FALSE.
x+	It is same as x but it creates and opens file in **read-write** mode.
c	Opens file in **write-only** mode. If the file does not exist, it is created. If it exists, it is neither truncated (as opposed to 'w'), nor the call to this function fails (as is the case with 'x'). The file pointer is positioned on the beginning of the file
c+	It is same as c but it opens file in **read-write** mode.

PHP Open File Example

```
<?php
$handle = fopen("c:\\folder\\file.txt", "r");
?>
```

Next Topic PHP Read File

PHP Read File

PHP provides various functions to read data from file. There are different functions that allow you to read all file data, read data line by line and read data character by character.

The available PHP file read functions are given below.

- frea d()

- fgets()
- fgetc()

PHP Read File - fread()

The PHP fread() function is used to read data of the file. It requires two arguments: file resource and file size.

Syntax

```
string fread (resource $handle , int $length )
```

$handle represents file pointer that is created by fopen() function.

$length represents length of byte to be read.

Example

```
<?php
$filename = "c:\\file1.txt";
$fp = fopen($filename, "r");//open file in read mode

$contents = fread($fp, filesize($filename));//read file

echo "<pre>$contents</pre>";//printing data of file
fclose($fp);//close file
?>
```

Output

```
this is first line
this is another line
this is third line
```

PHP Read File - fgets()

The PHP fgets() function is used to read single line from the file.

Syntax

```
string fgets ( resource $handle [, int $length ] )
```

Example

```
<?php
$fp = fopen("c:\\file1.txt", "r");//open file in read mode
echo fgets($fp);
fclose($fp);
?>
```

Output

```
this is first line
```

PHP Read File - fgetc()

The PHP fgetc() function is used to read single character from the file. To get all data using fgetc() function, use !feof() function inside the while loop.

Syntax

```
string fgetc ( resource $handle )
```

Example

```
<?php
$fp = fopen("c:\\file1.txt", "r");//open file in read mode
while(!feof($fp)) {
  echo fgetc($fp);
}
fclose($fp);
?>
```

Output

```
this is first line this is another line this is third line
```

Next Topic PHP Write File

PHP Write File

PHP fwrite() and fputs() functions are used to write data into file. To write data into file, you need to use w, r+, w+, x, x+, c or c+ mode.

PHP Write File - fwrite()

The PHP fwrite() function is used to write content of the string into file.

Syntax

```
int fwrite ( resource $handle , string $string [, int $length ] )
```

Example

```php
<?php
$fp = fopen('data.txt', 'w');//opens file in write-only mode
fwrite($fp, 'welcome ');
fwrite($fp, 'to php file write');
fclose($fp);

echo "File written successfully";
?>
```

Output: data.txt

```
welcome to php file write
```

PHP Overwriting File

If you run the above code again, it will erase the previous data of the file and writes the new data. Let's see the code that writes only new data into data.txt file.

```php
<?php
$fp = fopen('data.txt', 'w');//opens file in write-only mode
fwrite($fp, 'hello');
fclose($fp);

echo "File written successfully";
?>
```

Output: data.txt

```
hello
```

PHP Append to File

If you use **a** mode, it will not erase the data of the file. It will write the data at the end of the file. Visit the next page to see the example of appending data into file.

PHP Append to File

You can append data into file by using a or a+ mode in fopen() function. Let's see a simple example that appends data into data.txt file.

Let's see the data of file first.

data.txt

```
welcome to php file write
```

PHP Append to File - fwrite()

The PHP fwrite() function is used to write and append data into file.

Example

```
<?php
$fp = fopen('data.txt', 'a');//opens file in append mode
fwrite($fp, ' this is additional text ');
fwrite($fp, 'appending data');
fclose($fp);

echo "File appended successfully";
?>
```

Output: data.txt

```
welcome to php file write this is additional text appending data
```

Next Topic PHP Delete File

PHP Delete File

In PHP, we can delete any file using unlink() function. The unlink() function accepts one argument only: file name. It is similar to UNIX C unlink() function.

PHP unlink() generates E_WARNING level error if file is not deleted. It returns TRUE if file is deleted successfully otherwise FALSE.

Syntax

```
bool unlink ( string $filename [, resource $context ] )
```

$filename represents the name of the file to be deleted.

PHP Delete File Example

```php
<?php
$status=unlink('data.txt');
if($status){
echo "File deleted successfully";
}else{
echo "Sorry!";
}
?>
```

Output

```
File deleted successfully
```

Next Topic PHP File Upload