

verot.net - class.upload.php

[php_class_upload.htm](#)



This PHP script uploads files and manipulates images very easily. The perfect script to generate thumbnails or create a photo gallery! It can convert, resize and work on uploaded images in many ways, add labels, watermarks and reflections and other image editing features. It uses the GD library. This script is released under the [GPL Version 2](#). If your project is not GPL, [commercial licenses](#) are available.

[download v0.25](#) [license](#) [docs](#) [source](#)

version 0.25 released! Version 0.25 has been released, with a great number of [new features](#). It is a major rewrite of the graphic manipulation code, with improvements for transparency support and true color images. [Check out some demo samples](#).

What is it?

An example will be more useful than any blurb I can write :

```
$foo = new Upload($_FILES['form_field']);
if ($foo->uploaded) {
    // save uploaded image with no changes
    $foo->Process('/home/user/files/');
    if ($foo->processed) {
        echo 'original image copied';
    } else {
        echo 'error : ' . $foo->error;
    }
}
// save uploaded image with a new name
$foo->file_new_name_body = 'foo';
$foo->Process('/home/user/files/');
if ($foo->processed) {
    echo 'image renamed "foo" copied';
} else {
    echo 'error : ' . $foo->error;
}
```

```

}
// save uploaded image with a new name,
// resized to 100px wide
$foo->file_new_name_body = 'image_resized';
$foo->image_resize = true;
$foo->image_convert = gif;
$foo->image_x = 100;
$foo->image_ratio_y = true;
$foo->Process('/home/user/files/');
if ($foo->processed) {
    echo 'image renamed, resized x=100
        and converted to GIF';
    $foo->Clean();
} else {
    echo 'error : ' . $foo->error;
}
}

```

What does it actually do in this example?

This example uses a file uploaded via a form. With this single file, it will :

- copy the file without any modification
- copy the file after having changed its name to 'foo'
- copy the file, resizing it to x=100, y=auto, converting it to GIF

How does it work?

- the class constructor *upload* handles a uploaded file (it can also handle a local file)
- some optional parameters can be set up to act on the file during the *process*
- the process is called with as an argument the destination directory on the server. If some parameters have been set up, the class will rename, resize, convert files and images
- when the uploaded file is not needed anymore, we can delete it using *clean*.

Why use this class?

- with one uploaded file, you can do as many copies, conversions, resizing you want.
- you can resize each image dimension, or both, keeping the image ratio or not.
- you can choose to resize an image only if it is bigger -or smaller- than the wanted sizes
- you can manipulate the image in many ways, play with colors, add borders, labels and watermarks...
- its use is simplistic but powerful
- more than 60 documented configuration settings
- a lot of variables are set up during the process. You can retrieve all these values after a process.
- error messages are understandable, and a variable log allows you to see what the class does.
- it is already widely used on Internet
- it is free :)

Is it out there on Internet?

class.upload.php is featured on [HotScripts](#). If you like the script, please rate it or write a review for it. You can find it here : <http://www.hotscripts.com/Detailed/45364.html>

class.upload.php is featured on [PHP Classes](#). If you like the script, please rate it there. You can find it here : <http://www.phpclasses.org/browse/package/2181.html>

class.upload.php is featured on [Freshmeat](#). If you like the script, please rate it there. You can find it here : http://freshmeat.net/projects/class_upload_php/

Commercial licenses

[Commercial licenses](#) are available [here](#).