

# PHP Create Image Thumbnail

In this tutorial you will learn how to generate thumbnails from form uploaded images using GD php image library. The image ratio is maintained, so the thumbnail images will not be deformed. An image verification is performed before uploading and the image size is verified.

## Requirements :

You will need PHP and GD Graphics on your server, create

Library library installed. To see if you have gd installed a file named info.php, place the following content in it:

```
<?php
phpinfo();
?>
```

It should appear a page with infos and you

can find if you have gd installed or not.

Note1: You must set the enctype form attribute to "multipart/form-data", otherwise the form will not show you any errors, but it will just not upload your images!

Note2: The only supported files are the one with png, jpg and jpeg extension. The gif format is copyrighted, so is not supported by GD library.

Create a folder named images located in the path you are planning to place the php script you are about to create, and another folder thumbs inside the folder images. Make sure the created folders have write rights for everybody or the scripts won't work ( it won't be able to upload the files into the directory).

Next you will find the code with comments for each action. Please read them carefully.

```
<?php
//define a maxim size for the uploaded images
define ("MAX_SIZE","100");
// define the width and height for the thumbnail
// note that these dimensions are considered the maximum dimension and are not fixed,
// because we have to keep the image ratio intact or it will be deformed
define ("WIDTH","150");
define ("HEIGHT","100");

// this is the function that will create the thumbnail image from the uploaded image
// the resize will be done considering the width and height defined, but without deforming the image
function make_thumb($img_name,$filename,$new_w,$new_h)
{
//get image extension.
$ext=getExtension($img_name);
//creates the new image using the appropriate function from gd library
if(!strcmp("jpg",$ext) || !strcmp("jpeg",$ext))
$src_img=imagecreatefromjpeg($img_name);

if(!strcmp("png",$ext))
$src_img=imagecreatefrompng($img_name);

//gets the dimensions of the image
$old_x=imageSX($src_img);
$old_y=imageSY($src_img);

// next we will calculate the new dimensions for the thumbnail image
// the next steps will be taken:
// 1. calculate the ratio by dividing the old dimensions with the new ones
// 2. if the ratio for the width is higher, the width will remain the one define in WIDTH variable
// and the height will be calculated so the image ratio will not change
// 3. otherwise we will use the height ratio for the image
```

```

// as a result, only one of the dimmensions will be from the fixed ones
$ratio1=$old_x/$new_w;
$ratio2=$old_y/$new_h;
if($ratio1>$ratio2) {
    $thumb_w=$new_w;
    $thumb_h=$old_y/$ratio1;
}
else {
    $thumb_h=$new_h;
    $thumb_w=$old_x/$ratio2;
}

// we create a new image with the new dimmensions
$dst_img=ImageCreateTrueColor($thumb_w,$thumb_h);

// resize the big image to the new created one
imagecopyresampled($dst_img,$src_img,0,0,0,0,$thumb_w,$thumb_h,$old_x,$old_y);

// output the created image to the file. Now we will have the thumbnail into the file named by $filename
if(!strcmp("png",$ext))
    imagepng($dst_img,$filename);
else
    imagejpeg($dst_img,$filename);

//destroys source and destination images.
imagedestroy($dst_img);
imagedestroy($src_img);
}

// This function reads the extension of the file.
// It is used to determine if the file is an image by checking the extension.
function getExtension($str) {
    $i = strrpos($str,".");
    if (!$i) { return ""; }
    $l = strlen($str) - $i;
    $ext = substr($str,$i+1,$l);
    return $ext;
}

```

Here follows the php code to process post data.

The code contains image extension and size verification as well as uploading and resizing image.

```

// This variable is used as a flag. The value is initialized with 0 (meaning no error found)
and it will be changed to 1 if an erro occurs. If the error occurs the file will not be uploaded.
$errors=0;
// checks if the form has been submitted
if(isset($_POST['Submit']))
{
//reads the name of the file the user submitted for uploading
$image=$_FILES['image']['name'];
// if it is not empty
if ($image)
{
// get the original name of the file from the clients machine
$filename = stripslashes($_FILES['image']['name']);

// get the extension of the file in a lower case format
$extension = getExtension($filename);
$extension = strtolower($extension);
// if it is not a known extension, we will suppose it is an error, print an error message
and will not upload the file, otherwise we continue
if (($extension != "jpg") && ($extension != "jpeg") && ($extension != "png"))

```

```

{
echo '<h1>Unknown extension!</h1>';
$errors=1;
}
else
{
// get the size of the image in bytes
// $_FILES['image']['tmp_name'] is the temporary filename of the file in which
//the uploaded file was stored on the server
$size=getimagesize($_FILES['image']['tmp_name']);
$sizekb=filesize($_FILES['image']['tmp_name']);

//compare the size with the maxim size we defined and print error if bigger
if ($sizekb > MAX_SIZE*1024)
{
echo '<h1>You have exceeded the size limit!</h1>';
$errors=1;
}

//we will give an unique name, for example the time in unix time format
$image_name=time().'.'.$extension;
//the new name will be containing the full path where will be stored (images folder)
$newname="images/".$image_name;
$copied = copy($_FILES['image']['tmp_name'], $newname);
//we verify if the image has been uploaded, and print error instead
if (!$copied)
{
echo '<h1>Copy unsuccessful!</h1>';
$errors=1;
}
else
{
// the new thumbnail image will be placed in images/thumbs/ folder
$thumb_name='images/thumbs/thumb_'.$image_name;
// call the function that will create the thumbnail. The function will get as parameters
the image name, the thumbnail name and the width and height desired for the thumbnail
$thumb=make_thumb($newname,$thumb_name,WIDTH,HEIGHT);
}}}

//If no errors registred, print the success message and show the thumbnail image created
if(isset($_POST['Submit']) && !$errors)
{
echo "<h1>Thumbnail created Successfully!</h1>";
echo '';
}

?>

```

Now all you have to do is add the form that is needed for uploading the image.

The form will have the enctype attribute to "multipart/form-data" and the input should have "file" type.

```

<form name="newad" method="post" enctype="multipart/form-data" action="">
<table>
<tr><td><input type="file" name="image" ></td></tr>
<tr><td><input name="Submit" type="submit" value="Upload image"></td></tr>
</table>
</form>

```

This code was tested and works fine, I hope you will find it useful.

If you found this tutorial useful you might be interested to read the following:

Installing Apache, PHP, MySQL on a Windows 98/2000/XP Computer

PHP Form image verification

Count Words in a string using PHP

PHP Form Image Upload

PHP print random image from a folder