

Ресайз изображений "на лету" или простое создание превью изображений.

В этой статье расскажу о том, как сделать так, чтобы превью изображений генерировалось динамически из большой картинке. Такая задача возникает довольно часто и применяется для экономии места на сервере. Сразу оговорюсь, что будет использован специальный класс. Но сложного ничего не будет. Начнем с самого кода класса:

```
class imageResize {
public function __construct($src, $width=100) {
// $filename - ?????? ???? ? ??????.
$filename=$_SERVER['DOCUMENT_ROOT'].$src;
// ?????? getimagesize() ?????????? ?????????? ? ??????.
$size = getimagesize ($filename);
// ?????? ??????????????:
$w=$size['0'];
// ?????? ??????????????:
$h=$size['1'];
// mime ??? ??????:
$type = $size['mime'];
// ?????????? ?????? ? ??????, ?????? ?????? ?????????????? ??? ?????????????????????? ?????
?? ??????????????
$ratio = $w/$h;
// ?????? ??????:
$th_w = $width;
// ?????? ??????:
$th_h = $th_w/$ratio;
// ?????????? ?????????? ?????????? ??? ??????????:
header("Content-type: $type");
// ?????????????????? ?????, ?????????? ?????????? ??????????????: jpg, png ? gif:
switch ($type) {
case 'image/jpeg':
// ?????????? ?????????????? ?? ?????????? ?????????? ??????????????:
$src_img = imagecreatefromjpeg($filename);
// ?????????????????? ?????????? ??????????, ? ?????? ?????? ??? ?????? ??
// ??????????????, ?? ??? ?????? ?????, ??? ?????? ?????????? ?????????? ?????? ?? ??????????:
imagealphablending($src_img, true);
// ?????????? ?????? ?????????????? ?????? ?????? ? ??????,
// ? ?????? ?????? ?????????????? ?????????? ??????????????:
$thumbImage = imagecreatetruecolor($th_w, $th_h);
// ?????????? ?????????? ?????????????? ? ??????????:
imagecopyresampled($thumbImage, $src_img, 0, 0, 0, 0, $th_w, $th_h, $w, $h);
// ?????????? ? ?????????? ?????????????? ??????????????:
imagejpeg($thumbImage, '', 100);
break;
// ??? ?????????? ?????????? ??? ??????????????:
case 'image/png':
$src_img = imagecreatefrompng($filename);
imagealphablending($src_img, true);
$thumbImage = imagecreatetruecolor($th_w, $th_h);
imagecopyresampled($thumbImage, $src_img, 0, 0, 0, 0, $th_w, $th_h, $w, $h);
imagepng($thumbImage, '', 0);
break;
case 'image/gif':
```

```

$src_img = imagecreatefromgif($filename);
imagealphablending($src_img, true);
$thumbImage = imagecreatetruecolor($th_w, $th_h);
imagecopyresampled($thumbImage, $src_img, 0, 0, 0, 0, $th_w, $th_h, $w, $h);
imagegif($thumbImage, '', 100);
break;
default:
//echo "?????? ??????????? ?? ??????????????.";
break;
}
//??????? ?????????????? ?? ???????.
imagedestroy($thumbImage);
}
}

```

Ну вот и все, осталось записать вывод в браузер в нужном месте. Если вы плохо знакомы с ООП, то немного объясню логику. Чтобы создать экземпляр класса `imageResize`, нужно сделать следующую запись: `$imageObject = new imageResize();` Как только интерпретатор видит такую запись, он вызывает функцию `__construct()`. Как можно заметить, у этой функции в нашем классе есть два атрибута, `$src` – относительный путь к изображению, обязательный, `$width` – необязательный параметр, по умолчанию равен 100, но можно задать свою ширину для превью изображения. Ну и теперь можно делать вывод превью на странице. Для этого нам понадобится два файла: первый – сам класс `imageResize`, назовем его `imageResize.class.php` и файл, на который будет ссылаться атрибут `src` превью изображения, например `resize.php`. Положим оба файла в корневую папку сайта. Содержимое файла `imageResize.class.php` у нас уже есть, нужно просто скопировать код выше. Код файла `resize.php` будет содержать всего две строки: подключение класса `imageResize` и создание объекта класса:

```

include_once 'imageResize.class.php';
$im = new ImageResize($_GET['src']);

```

Вторая строка создает объект класса `ImageResize()` и передает ему путь к большому изображению из `$_GET` переменной `src`. Обратите внимание, что данный пример работает в случае, если оба файла лежат в корне сервера, для другой конфигурации придется прописать в инструкции `include_once` путь к классу `imageResize`. Ну и самое главное, как же нам вывести на странице собственно превью. Ничего сложного, в коде страницы нужно всего-лишь записать: `` Ну вот и все, теперь у нас есть готовый функционал для создания превью больших картинок, без сохранения превью на сервере.