

使用 PHP QR Code 生成二维码

HP QR Code 是一个 PHP 二维码生成类库，利用它可以轻松生成二维码，官网提供了下载和多个演示 demo，查看地址：<http://phpqrcode.sourceforge.net/>

下载官网提供的类库后，只需要使用 `phpqrcode.php` 就可以生成二维码了，当然您的 PHP 环境必须开启支持 GD2。

`phpqrcode.php` 提供了一个关键的方法，其中参数 `$text` 表示生成二位的信息文本；参数 `$outfile` 表示是否输出二维码图片文件，默认否；参数 `$level` 表示容错率，也就是有被覆盖的区域还能识别，分别是 `L (QR_ECLEVEL_L , 7%)` , `M (QR_ECLEVEL_M , 15%)` , `Q (QR_ECLEVEL_Q , 25%)` , `H (QR_ECLEVEL_H , 30%)`；参数 `$size` 表示生成图片大小，默认是 3；参数 `$margin` 表示二维码周围边框空白区域间距值；参数 `$saveandprint` 表示是否保存二维码并显示。代码如下：

```
public static function png($text, $outfile=false, $level=QR_ECLEVEL_L, $size=3,
$margin=4, $saveandprint=false)
{
    $enc= QRencode::factory($level, $size, $margin);
    return $enc->encodePNG($text, $outfile, $saveandprint=false);
}
```

调用 PHP QR Code 非常简单，如下代码即可生成一张内容为
"http://www.cnblogs.com/txw1958/" 的二维码。

```
include 'phpqrcode.php';
QRcode::png('http://www.cnblogs.com/txw1958/');
```

那么实际应用中，我们会在二维码的中间加上自己的 LOGO，已增强宣传效果。那如何生成含有 logo 的二维码呢？其实原理很简单，先使用 PHP QR Code 生成一张二维码图片，然后再利用 php 的 image 相关函数，将事先准备好的 logo 图片加入到刚生成的原始二维码图片中间，然后重新生成一张新的二维码图片。
代码如下：



```
include 'phpqrcode.php';
$value = 'http://www.cnblogs.com/txw1958/'; //二维码内容
$errorCorrectionLevel = 'L'; //容错级别
$matrixPointSize = 6; //生成图片大小
//生成二维码图片
QRcode::png($value, 'qrcode.png', $errorCorrectionLevel, $matrixPointSize, 2);
$logo = 'logo.png'; //准备好的 logo 图片
$QR = 'qrcode.png'; //已经生成的原始二维码图

if ($logo !== FALSE) {
    $QR = imagecreatefromstring(file_get_contents($QR));
```

```

    $logo = imagecreatefromstring(file_get_contents($logo));
    $QR_width = imagesx($QR); // 二维码图片宽度
    $QR_height = imagesy($QR); // 二维码图片高度
    $logo_width = imagesx($logo); // logo 图片宽度
    $logo_height = imagesy($logo); // logo 图片高度
    $logo_qr_width = $QR_width / 5;
    $scale = $logo_width / $logo_qr_width;
    $logo_qr_height = $logo_height / $scale;
    $from_width = ($QR_width - $logo_qr_width) / 2;
    // 重新组合图片并调整大小
    imagecopyresampled($QR, $logo, $from_width, $from_width, 0, 0,
$logo_qr_width,
    $logo_qr_height, $logo_width, $logo_height);
}
// 输出图片
imagepng($QR, 'helloweixin.png');
echo '';

```



```

include 'phpqrcode.php';
$value = 'http://www.cnblogs.com/txw1958/'; // 二维码内容
$errorCorrectionLevel = 'L'; // 容错级别
$matrixPointSize = 6; // 生成图片大小
// 生成二维码图片
QRcode::png($value, 'qrcode.png', $errorCorrectionLevel, $matrixPointSize, 2);
$logo = 'logo.png'; // 准备好的 logo 图片
$QR = 'qrcode.png'; // 已经生成的原始二维码图

if ($logo !== FALSE) {
    $QR = imagecreatefromstring(file_get_contents($QR));
    $logo = imagecreatefromstring(file_get_contents($logo));
    $QR_width = imagesx($QR); // 二维码图片宽度
    $QR_height = imagesy($QR); // 二维码图片高度
    $logo_width = imagesx($logo); // logo 图片宽度
    $logo_height = imagesy($logo); // logo 图片高度
    $logo_qr_width = $QR_width / 5;
    $scale = $logo_width / $logo_qr_width;
    $logo_qr_height = $logo_height / $scale;
    $from_width = ($QR_width - $logo_qr_width) / 2;
    // 重新组合图片并调整大小
    imagecopyresampled($QR, $logo, $from_width, $from_width, 0, 0,
$logo_qr_width,
    $logo_qr_height, $logo_width, $logo_height);
}

```

```
//输出图片
imagepng($QR, 'helloweixin.png');
echo '';
```



由于二维码允许有一定的容错性，一般的二维码即使在遮住部分但仍然能够解码，经常我们扫描二维码的时候扫描到甚至不到一半时就能解码扫描结果，这是因为生成器会将部分信息重复表示来提高其容错度，这就是为什么我们在二维码中间加个 LOGO 图片并不影响解码结果的原因。