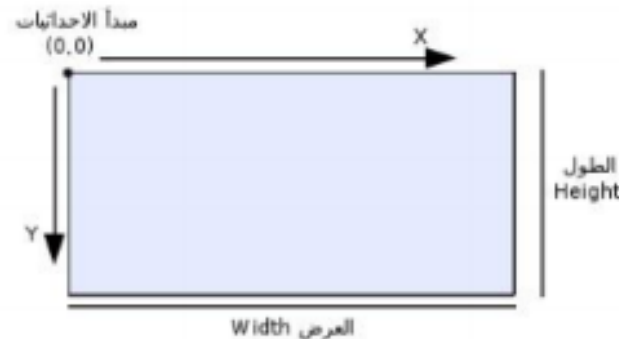


الوحدة 12

الصور

تستخدم php المكتبة GD المفتوحة المصدر للقيام بالعمليات على الصور وهذه المكتبة تأتي بشكل افتراضي مع php وتوفر عدد لا بأس به من الدوال للتعامل مع الصور. تتألف الصورة من مستطيل يحوي عدداً من النقاط المربعة الشكل , وتسمى هذه النقاط بالبكسل pixel . ولكل نقطة لون معين , ويكون هذا اللون بشكل عام مُحدد بثلاث مكونات : اللون الأحمر , اللون الأخضر و اللون الأزرق و تتراوح قيمهم بين 0 - 255 . وبعض صيغ الصور (كصيغة GIF) توفر عدداً محدوداً من الألوان في الصورة , بينما البعض الآخر يُوفر ما يُسمى الألوان الحقيقية (256^3 أو 16777216) .

يتم اعتبار مبدأ الأحداثيات عند التعامل مع الصور هو الزاوية العليا اليسرى و تتزايد قيمة العرض width عند الإنتقال الى اليمين , وقيمة الطول height عند الانتقال الى الأسفل كما في الصورة التالية:



انشاء الصور :

يتم انشاء مقبض للصورة اما بتحميل ملف الصورة المحفوظ في القرص الصلب الى الذاكرة او بانشاء صورة جديدة وذلك باستخدام الدوال التالية:

[imagecreatetruecolor](#), [imagecreatefromstring](#), [imagecreatefromjpeg](#), [imagecreatefrompng](#), [imagecreatefromgif](#) ..etc

الدوال [imagecreatefromjpeg](#) , [imagecreatefrompng](#) , [imagecreatefromgif](#) :

تعمل هذه الدوال الثلاث بنفس الآلية تقريبا حيث تقوم بانشاء مقبض للصورة عن طريق تحميل (load) الصورة من القرص , الشكل العام لاستعاء هذه الدوال هو :

```
$image = imagecreatefrompng('image.png');  
$image = imagecreatefromjpeg('image.jpg');  
$image = imagecreatefromgif('image.gif');
```

حيث تقبل هذه الدوال الثلاث وسيطا "وحيدا" هو مسار الصورة.

دالة [imagecreatetruecolor](#) :

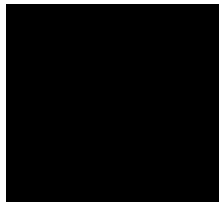
تقوم بانشاء مقبض لصورة جديدة بالابعاد المُمررة اليها كوسائط , الشكل العام :

```
$image = imagecreatetruecolor($width, $height);
```

حيث الوسيط الاول هو عرض الصورة مقدرا بالبكسل والثاني هو ارتفاعها .



صورة جاهزة
موجودة



صورة جديدة
فارغة

إخراج الصور :

ويتم ذلك بإخراجها (output) إلى المتصفح مباشرة أو بحفظها بملف مستقل وذلك باستخدام الدوال [imagepng](#), [imagejpeg](#), [imagegif](#) حيث تعمل بنفس الآلية مع اختلاف نوع الملف المُعاد

```
imagepng($image, $filename, $quality);
```

```
imagegif($image, $filename, $quality);
```

```
imagejpeg($image, $filename, $quality);
```

حيث الوسيط الأول هو مقبض الصورة والوسيط الثاني اسم الملف والثالث هو نسبة مئوية تُحدد جودة الصورة ، الوسيطين الثاني والثالث اختياريين وفي حال لم يتم تحديد اسم الملف سيتم طباعة الصورة مباشرة إلى المتصفح :

عندما تريد إظهار الصورة في المتصفح

[header](#) يجب عليك استخدام الدالة


مثال: انشاء واخراج

```
<?php
/*
$black = get_extension_funcs("gd"); // Grab function list
if (! $black) echo "GD not even installed."; exit; }
echo"<pre>".print_r($black, true). "</pre>";*/
```

```
$image = imagecreatetruecolor(200, 200);
//save image file to disk
imagepng($image, "image200.png");
// imagejpeg($image);
```

This PC > Local Disk (C:) > xampp > htdocs > unit12

Name	Date modified	Type	Size
nhproject	11/19/2021 4:22 PM	File folder	
1	12/6/2022 8:44 PM	PNG file	15
11	12/6/2022 8:48 PM	PNG file	15
23	12/1/2021 10:48 AM	PNG file	1
color	12/1/2021 10:36 AM	PHP file	1
colorat	11/20/2021 8:38 PM	PHP file	1
filter	11/20/2021 8:53 PM	PHP file	1
image	12/1/2021 9:51 AM	JPG file	1
image	12/1/2022 7:04 AM	PNG file	1
image200	12/11/2022 9:29 PM	PNG file	1
index	12/11/2022 9:29 PM	PHP file	1
lines	12/1/2021 10:43 AM	PHP file	1
rect	12/1/2021 10:41 AM	PHP file	1
sizexy	11/20/2021 8:40 PM	PHP file	1
text	12/1/2021 10:44 AM	PHP file	1
xx	12/1/2021 10:20 AM	PHP file	1



انشاء صورة من نص :

تستخدم الدالة imagecreatefromstring لانشاء مقبض لصورة جاهزة دون الحاجة الى وجود ملف لها حيث يمكن ان تكون بيانات الصورة مخزنة ضمن قاعدة بيانات او باستخدام دالة base64_encode مثال :

```
<?php
$base_64_data =
'/9j/4AAQSkZJRgABAQEASABIAAD/2wBDADIiJSwLHzIsKSw4NTI7S31RS0VFS5ltc
1p9tZ+
+u7Kfr6zI4f/zyNT/16yv+v/9////////wFD/////////2wBDATU40EtCS5NRU
ZP/zq/O//////////wAARCAAwADEDASIAAhEBAxEB/8QAGQAAAwEBAQAAAAAAAAAAAAAMEA
gUB/BQAJxAAAgiBAwMDBQEAAAAAAAAAAAAAECAxEEjEhQVETMmEFIkJxgZH/xAAyAQE
BAQEBAAAAAAAAAAAADAgABBP/EABwRAAMBAAMBAQAAAAAAAAAAAAABAhESITEDQ
f/aAAwDAQACEQMRAD8A6gALnLLwjjeHUtCVqTwurM0/H4mJrdNbfcjx124z9r+Cnp+
FpT+k71Nm7duf60lB7oJ+UQOKb6xw/kr09n+OHytjXehbXWoaAAIES6qbU1HtgXVJ5
b7FdLubPcsiNuyThxjgG0/RZpZg6mOI57tjCb11T7uAetqS6Zz4ElrCeLfhNw4goyX
L6C9HY3fjyhn97unl9EuEN+nwbtc+yWCcTelueMdnQAAEPOAuZvevDXDGAb0xzNTXY
p9U2vKEfx/4dLxT5PPTj4J4jt9WlhHTXBVrMU2+cL0njGMGorCyZnQ3JuHmsja4KEM
EzLT7IqtRoAAQg//Z';
$image = imagecreatefromstring(base64_decode($base_64_data));
header('Content-Type: image/png');
imagepng($image);
imagedestroy($image);
?>
```

تعريف الألوان في الصور :

ويتم ذلك باستخدام الدالة [imagecolorallocate](#) بالشكل التالي :

```
$color = imagecolorallocate($image, $red, $green, $blue);
```

حيث تقبل هذه الدالة اربعة وسائط اجبارية هي على التوالي و بالترتيب : مقبض الصورة المراد تعريف اللون لها الثاني قيمة اللون الاحمر , قيمة اللون الاخضر , ومن ثم قيمة اللون الازرق وتتراوح قيم آخر ثلاث وسائط بين 0 - 255 .

دالة [imagecolorallocatealpha](#) :

تعمل هذه الدالة كما تعمل الدالة السابقة باستثناء وجود وسيط جديد هو قيمة الشفافية alpha الذي تتراوح قيمته بين 0 - 127

```
$color = imagecolorallocatealpha($image, $red, $green, $blue, $alpha);
```

حيث القيمة 127 تمثل لون شفاف تماما والقيمة 0 تعني انعدام الشفافية.

دالة [imagecolorat](#) :

تستخدم هذه الدالة لإرجاع لون بكسل محدد باحداثياته من صورة مُحددة بمقبضها , ويكون شكلها العام كالتالي :

```
$color = imagecolorat($image, $x, $y);
```

لكن هذه الدالة تعيد قيمة RGB كرقم, ولاستخلاص قيم الالوان نستخدم الطريقة التالية :

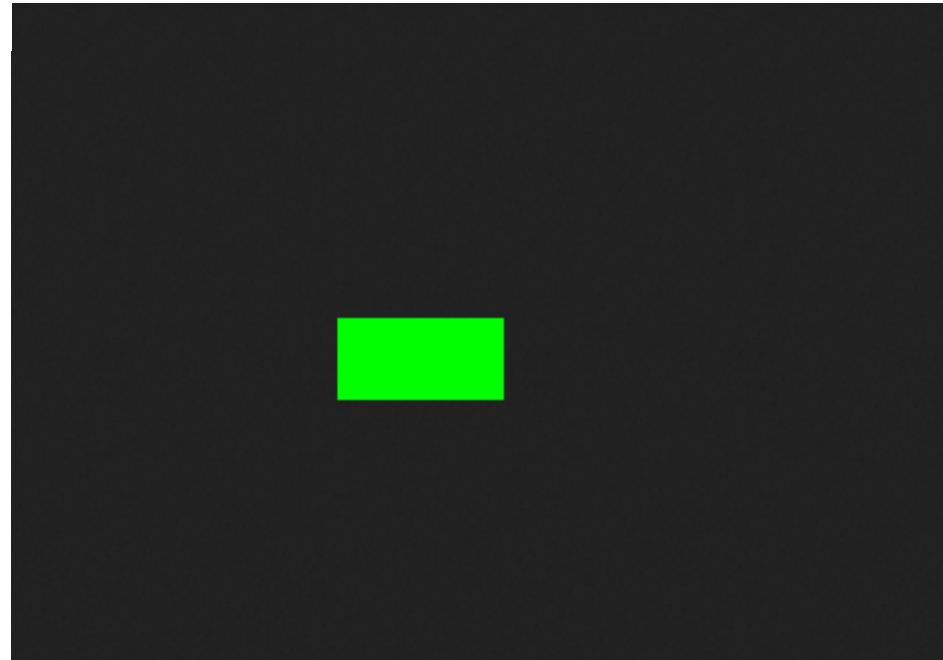
باستخدام دالة [imagecolorsforindex](#) التي تعيد مصفوفة تحوي قيم الالوان حيث يمثل كل لون عنصراً من عناصر تلك المصفوفة , يتم استدعائها بالشكل التالي :

```
<?php
$image = imagecreatefromjpeg('image.jpg');
$color1 = imagecolorat($image, 10, 10);
$colors1 = imagecolorsforindex($image, $color1);
```

مثال: انشاء لون وتعبئة الصورة

07.php

```
$image = imagecreatetruecolor(100, 50);  
$bg_color = imagecolorallocate($image, 0, 255, 0);  
imagefill($image, 0, 0, $bg_color);  
header('Content-Type: image/png');  
imagepng($image);  
imagedestroy($image);
```



مثال: اعادة مصفوفة بقيم الالوان

```
<?php
$image = imagecreatefrompng('1.PNG');
$color1 = imagecolorat($image, 10, 10);
$colors1 = imagecolorstotext($image, $color1);
//print_r($colors1) outputs :
//Array ( [red] => 247 [green] => 247 [blue] => 247 [alpha] => 0 )
echo "The first pixel color is : red = ".$colors1['red']." , green
= ".$colors1['green']." , blue = ".$colors1['blue']."<br>";
$color2 = imagecolorat($image, 50, 50);
$colors2 = imagecolorstotext($image, $color2);
echo "The second pixel color is : red = ".$colors2['red']." ,
green = ".$colors2['green']." , blue = ".$colors2['blue']."<br>";
imagedestroy($image);
?>
```

The first pixel color is : red = 247 , green = 247 , blue = 247

The second pixel color is : red = 219 , green = 225 , blue = 237

هدم مقبض الصورة :

يتم هدم الصورة لتحرير الذاكرة المستخدمة من قبلها وذلك عن طريق دالة [imagedestroy](#) حيث تقبل وسيطاً وحيداً هو مقبض الصورة :

```
imagedestroy($image);
```

تحديد أبعاد الصور :

وذلك باستخدام الدالتين [imagesx](#), [imagesy](#) حيث تعيد هاتين الدالتين قيم العرض والطول للصورة وتقبلان وسيطاً وحيداً هو مقبض الصورة :

```
<?php
$image = imagecreatefromjpeg('1.jpg');
$x = imagesx($image);
$y = imagesy($image);
echo "image width is $x and its height is $y";
imagedestroy($image);
?>
```

```
<?php
$image = imagecreatefromjpeg('image.jpg');
$x = imagesx($image);
$y = imagesy($image);
echo "image width is $x and its height is $y";
imagedestroy($image);
?>
```

 google  whatsapp  Freelancers platform

image width is 3840 and its height is 2160

الدالة [imagefill](#) :

تقوم هذه الدالة بتلوين منطقة محددة بلون واحد اي كما تقوم اداة التعبئة في برامج الرسم :

```
<?php
$image = imagecreatetruecolor(200, 200);
$bg_color = imagecolorallocate($image, 0, 255, 0);
imagefill($image, 0, 0, $bg_color);
header('Content-Type: image/png');
imagepng($image);
imagedestroy($image);
?>
```

حيث تقبل اربعة وسائط ,شكلها العام هو :

```
imagefill($image, $x, $y, $color);
```

الدالة [imagefilledrectangle](#) : تقوم هذه الدالة بملئ مستطيل بلون محدد :

```
imagefilledrectangle($image, $x1, $y1, $x2, $y2, $color);
```

حيث \$x1, \$y1 عي احداثيات الزاوية اليسرى العليا و \$x2, \$y2 هي احداثيات الزاوية اليمنى السفلى.

تدوير الصورة : يوجد في مكتبة GD دالة باسم [imagerotate](#) تقوم بتدوير الصورة حول مركزها , شكلها العام :

```
imagerotate($image, $angle, $bg_color);
```

حيث الزاوية بالدرجات و\$bg_color هو اللون الذي سيتم وضعه مكان الفراغ نتيجة التدوير .
مثال :

```
<?php
$image = imagecreatetruecolor(200, 200);
imagefill($image, 0, 0, 0xffffffff);
$color = imagecolorallocate($image, 0, 255, 0);
imagefilledrectangle($image, 50, 50, 149, 149, $color);
$image = imagerotate($image, 45, 0xffffffff);
header('Content-Type: image/png');
imagepng($image);
imagedestroy($image);
?>
```

الدالة [imagesetpixel](#) : لتلوين بكسل معين محدد بالاحداثيات في

صورة

مفعول هذه الدالة مثل اسمها حيث تقوم بتحديد لون بكسل معين بالاحداثيات x, y تبدو هذه الدالة بلا فائدة لكن في الحقيقة هي من اهم الدوال في مكتبة GD حيث تستخدم في كثير من التطبيقات المتقدمة على الصور كما سنرى لاحقاً في سياق هذا الفصل :

```
imagesetpixel($image, $x, $y, $color);
```

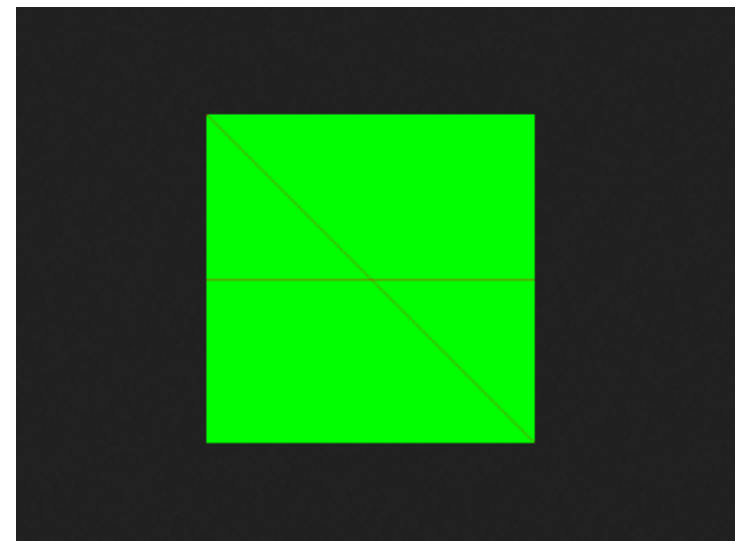
رسم مستقيمات :

يتم رسم المستقيمات بواسطة الدالة [imageline](#) التي تقوم برسم مستقيم بين نقطتين محددين :

```
imageline($image, $x1, $y1, $x2, $y2, $color);
```

حيث \$x1 , \$y1 هي إحداثيات نقطة البداية و \$x2 , \$y2 هي إحداثيات نقطة النهاية

```
<?php  
$image = imagecreatetruecolor(200, 200);  
$bg_color = imagecolorallocate($image, 0, 255, 0);  
imagefill($image, 0, 0, $bg_color);  
$color = imagecolorallocatealpha($image, 255, 0, 0, 75);  
//imageextthickness($image, 5);  
imageline($image, 0, 100, 199, 100, $color);  
imageline($image, 0, 0, 199, 199, $color);  
header('Content-Type: image/png');  
imagepng($image);  
imagedestroy($image);  
?>
```



تحديد سمك خط الرسم :

وذلك بواسطة الدالة [imagesetthickness](#) التي تقبل وسيطين الاول هو مقبض الصورة و الثاني هو سمك الخط مقدراً بالبكسل :

```
imagesetthickness($image, $thickness);
```

نحدد سمك الخط من خلال هذه
الدالة

قبل رسم الخط المستقيم

```
imagesetthickness($image, 5);
```

كتابة نص على صورة :

ويوجد طريقتين : الاولى بواسطة الدالة imagestring والتي تستخدم الخطوط المدمجة مع مكتبة GD والثانية عن طريق دالة imagefttext التي تستخدم خطوط ttf او true-type fonts المتوفرة بكثرة (يوجد عدد من المواقع توفر خطوط مجانية بصيغة ttf).

دالة **imagestring** : الشكل العام لهذه الدالة :

1

```
imagestring($image, $font, $x, $y, $string, $color);
```

حيث الوسيط font يمثل حجم الخط و يأخذ قيمة عددية تتراوح بين 1-5

```
<?php
$image = imagecreatetruecolor(250, 100);
$bg_color = imagecolorallocate($image, 255, 255, 255);
imagefill($image, 0, 0, $bg_color);
$color = imagecolorallocatealpha($image, 0, 0, 0, 75);
for($i = 1; $i <= 5; $i++)
{
    imagestring($image, $i, 20, 15 * $i, 'zain',
    $color);
}
header('Content-Type: image/png');
imagepng($image);
imagedestroy($image);
?>
```



تقوم هذه الدالة بطباعة نص باستخدام خطوط ttf بأي مقاس خط وبأي زاوية ,الشكل العام هو :

```
imagettftext($image, $size, $angle, $x, $y, $color, $fontfile,
$text);
```

حيث الزاوية يمكن ان تكون موجبة او سالبة (القيمة الموجبة تؤدي الى الدوران عكس عقارب الساعة و القيمة السالبة تؤدي الى الدوران مع عقارب الساعة) و \$fontfile هو مسار الخط المستخدم , جرب المثال التالي لكن مع وضع اي خط ttf في نفس المجلد (أو كتابة مساره) وليكن اسم الخط font.ttf :

```
<?php
$image = imagecreatetruecolor(200, 200);
$bg_color = imagecolorallocate($image, 255, 255, 255);
imagefill($image, 0, 0, $bg_color);
$color = imagecolorallocatealpha($image, 0, 0, 0, 75);
imagettftext($image, 25, 0, 25, 110, $color, 'font.ttf',
'PHP:hypertext processor');
header('Content-Type: image/png');
imagepng($image);
imagedestroy($image);
?>
```

2

هنا جب ان يكون لديك بالمجلد ملف لانواع خطوط

نسخ صورة الى صورة :

دالة imagecopy :

1

```
imagecopy($dst_im, $src_im, $dst_x, $dst_y, $src_x, $src_y,  
$src_w, $src_h);
```

تقوم هذه الدالة بنسخ جزء من صورة الى صورة اخرى حيث تقبل الوسائط التالية :

الوسيط \$dst_im هو الصورة التي سيتم النسخ اليها (الصورة الهدف)

الوسيط \$src_im الصورة التي سيتم النسخ منها

الوسائط \$dst_x, \$dst_y, \$src_x, \$src_y هي احداثيات بداية النسخ و اللصق

الوسيطين \$src_w, \$src_h هم عرض و طول الجزء المنسوخ

دالة imagecopyresized : تقوم هذه الدالة بنسخ جزء من صورة ولصقه في صورة أخرى مع تغيير أبعاده :

```
imagecopyresized($dst_image, $src_image, $dst_x, $dst_y, $src_x,  
$src_y, $dst_w, $dst_h, $src_w, $src_h);
```

حيث دلالات الوسائط كما في الدالة السابقة .

تطبيق التأثيرات على الصور :

تأتي الدالة [imagefilter](#) حاملةً العديد من التأثيرات أو ما يُسمى "الفلاتر" التي يُمكن تطبيقها على الصور برمجياً، وتأخذ هذه الدالة عدد متغير من الوسائط بحسب التأثير المُمرر إليها ، و لكن كما هو مُعتاد يكون الوسيط الأول هو مقبض الصورة والثاني هو الثابت الخاص بالفلتر المُستخدم و باقي الوسائط هي وسائط تختلف حسب التأثير المُستخدم . شكل الدالة العام :

```
imagefilter($image, $filtertype, $arg1, $arg2, $arg3);
```

تغيير الإضاءة في الصور :

عند تمرير الثابت `IMG_FILTER_BRIGHTNESS` الى الدالة `imagefilter` يمكن تغيير الإضاءة في الصور وعند استخدام هذا التأثير يجب تمرير وسيط آخر هو قيمة الإضاءة التي تتراوح قيمتها بين -255 الى 255 حيث القيمة 255 تمثل إضاءة كاملة (اللون الأبيض) أما القيمة -255 فتتمثل اللون الاسود و القيمة 0 تبقى الإضاءة على حالها .

```
imagefilter($image, IMG_FILTER_BRIGHTNESS, 100);
```



2

تطبيق تأثير الضبابية **blur** :

وذلك عند استخدام الثابت `IMG_FILTER_GAUSSIAN_BLUR` أو `IMG_FILTER_SELECTIVE_BLUR` ولا داعي لاستخدام أي وسيط إضافي .

```
imagefilter($image, IMG_FILTER_SELECTIVE_BLUR) ;
```



الجدول التالي يوضح الفلاتر المدعومة من قبل الدالة `imagefilter` :

الثابت	الشرح
<code>IMG_FILTER_NEGATE</code>	عكس جميع ألوان الصورة
<code>IMG_FILTER_GRAYSCALE</code>	تحويل الصورة الى صورة رمادية (أبيض و أسود)
<code>IMG_FILTER_BRIGHTNESS</code>	كما في الفقرة السابقة تُستخدم للتحكم بالإضاءة . تقبل وسيطاً إضافياً هو مقدار الإضاءة
<code>IMG_FILTER_CONTRAST</code>	تغيير تباين الصورة ، تقبل وسيطاً إضافياً هو مقدار التباين
<code>IMG_FILTER_EDGEDETECT</code>	تطبيق خوارزمية لإظهار حواف مكونات الصورة
<code>IMG_FILTER_GAUSSIAN_BLUR</code>	إضافة تأثير الضبابية للصورة باستخدام خوارزمية <code>gaussian blur</code>
<code>IMG_FILTER_SELECTIVE_BLUR</code>	إضافة تأثير الضبابية للصورة
<code>IMG_FILTER_PIXELATE</code>	إضافة تأثير <code>pixelate</code> للصورة

```
<?php
$image = imagecreatefromjpeg('1.jpg');
//imagefilter($image, IMG_FILTER_BRIGHTNESS, 100);
imagefilter($image, IMG_FILTER_NEGATE, 100);
header('Content-Type: image/png');
imagejpeg($image);
?>
```



تطبيقات أخرى
إضافية

التطبيق الاول تغيير صيغة صورة :

هذا التطبيق من ابسط التطبيقات المفيدة التي يمكن الاستفادة من مكتبة GD فيها ويكون بالطريقة التالية : في البداية ننشئ مقبض للصورة ولنفرض ان صيغتها jpg ثم نستخدم الدالة imagepng لكتابتها الى ملف :

```
<?php
$image = imagecreatefromjpeg('image.jpg');
imagepng($image, 'image.png');
imagedestroy($image);
?>
```

قلب الصورة flipping image :

لا توفر مكتبة GD دالة لقلب الصورة رأسياً او أفقياً لكن يمكن بسهولة عمل دالة للقيام بتلك المهمة , مثلاً لقلب الصورة رأسياً نجعل كل y هي y - مع بقاء x ثابتة كما في المثال التالي :

قمت في الدالة السابقة بانشاء صورة مؤقتة و تمكنا باستخدام حلقات for من الوصول الى جميع بكسلات الصورة الاصلية ووضع محل كل y قيمة تساوي ارتفاع الصورة - y

فقط نستبدل بهذه الجملة للقلب
افقياً

```
imagesetpixel($tmp_image, imagesx($image)-$x, $y, $color);
```

```
<?php
$image = imagecreatefromjpeg('image.jpg');
header('Content-Type: image/png');
$fliped_image = imagefliphorizontal($image);
imagepng($fliped_image);
imagedestroy($image);
imagedestroy($fliped_image);

function imagefliphorizontal($image)
{
    $tmp_image = imagecreatetruecolor(imagesx($image),
imagesy($image));
    for($x = 0, $w = imagesx($image); $x < $w; $x++)
    {
        for($y = 0 , $h = imagesy($image); $y < $h; $y++)
        {
            //to get each pixel
            $color = imagecolorat($image, $x, $y);
            imagesetpixel($tmp_image, $x, imagesy($image)-$y, $color);
        }
    }

    return $tmp_image;
}
?>
```

انشاء صور المصغرات :

عند عرض الصور في صفحة ويب لا يجب ان يكون حجمها كبيرا , لانها تتطلب وقتا طويلا لكي تتحمل من الانترنت , وخصوصا عند عرض عدد كبير من الصور في الصفحة الواحدة. يمكن انشاء صور المصغرات عن طريق اعادة تحجيم الصورة باستخدام احدى الدالتين :

[imagecopyresized](#) او [imagecopyresampled](#)