

Image Processing with PHP

Who the hell is this guy?

Name: Adrian Quek

Occupation: Programmer at [Ablewise](#)

Hobbies: Coding

Vital statistics: Secret

Software used

- Ubuntu 7.10 aka Gutsy Gibbon (Linux Rules!)
- Apache 2.2.4-3ubuntu0.1 ((Linux Rules!)*2)
- PHP Version 5.2.3-1ubuntu6.3 (PHP Rocks!)
- GD graphics library (ok, running out of enthusiasm)

GD2 Installation

- On Debian-based systems, just install php5-gd package
 - command: `apt-get install php5-gd`
- On generic *nux b0x3n
 - for manual install, use the configure option “`--with-gd`” (\geq PHP 4.3.0)
 - otherwise, get the package for your distro
- On MS Windows, GD is bundled with PHP, just enable it in php.ini (or so I'm told)
- Default supported types:
 - jpeg
 - png
 - gif

GD2 Demos

- Captcha
- Resize, copy, crop

Shut up and show us the code! (Captcha)

- *resource imagecreatetruecolor* (*int \$width* , *int \$height*)
 - returns an image resource
 - image created has the inputted dimensions
- *int imagecolorallocate* (*resource \$image* , *int \$red* , *int \$green* , *int \$blue*)
 - returns a color identifier for *\$image* with the inputted RGB components
 - RGB coefficients may be either decimal (0-255) or hex (0x00-0xFF)
- *bool imagefill* (*resource \$image* , *int \$x* , *int \$y* , *int \$color*)
 - fills the image with the color (from *colorallocate()*) from position (x,y)
 - returns *TRUE* on success and *FALSE* on failure
- *int imagefontheight* (*int \$font*), *int imagefontwidth* (*int \$font*)
 - returns the height and width of a font character in the given respectively
 - *\$font* takes values from 1-5 (default fonts)
 - can create own fonts and load them with *imageloadfont(string \$filename)*
- *bool imagechar* (*resource \$image* , *int \$font* , *int \$x* , *int \$y* , *string \$c* , *int \$color*)
 - draws the first character of string *\$c* at position (*\$x*,*\$y*) with *\$font* in *\$color*
 - returns *TRUE* on success and *FALSE* on failure
- *bool imagesetpixel* (*resource \$image* , *int \$x* , *int \$y* , *int \$color*)
 - draws a pixel at position (*\$x*,*\$y*) in *\$color*

Shut up and show us the code! (Captcha continued)

- *bool imagepng (resource \$image [, string \$filename [, int \$quality [, int \$filters]]])*
 - creates a png image from the given resource
 - if a filename is specified, it outputs the image to the file, else, it outputs the raw stream

Shut up and show us the code! (Resize, copy, crop)

- *resource imagecreatefromjpeg (string \$filename)*
 - creates an image from the given jpeg file
- *int imagesx (resource \$image)*
 - returns the width of the given image
- *int imagesy (resource \$image)*
 - erm, yeah
- *bool imagedestroy (resource \$image)*
 - frees the memory used by the given image resource
- *bool imagecopyresized (resource \$dst_image , resource \$src_image ,
int \$dst_x , int \$dst_y , int \$src_x , int \$src_y , int \$dst_w , int \$dst_h ,
int \$src_w , int \$src_h)*
 - copies a rectangular part of the source image to the destination image