

1. Create an index.php file

First of all, create an index file, in this file create a html form of captcha. Also, add JavaScript that will refresh captcha without refreshing page. After captcha form submission, entered captcha code will be validated with the generated captcha code. If both are same user will see a message of success otherwise failure.

Now create **index.php** page and copy paste the below html captcha form in it.

HTML

```
<form name="form" method="post" action="">
<label><strong>Enter Captcha:</strong></label><br />
<input type="text" name="captcha" />
<p><br />

</p>
<p>Can't read the image?
<a href='javascript: refreshCaptcha();'>click here</a>
to refresh</p>
<input type="submit" name="submit" value="Submit">
</form>
```

JavaScript

```
<script>
//Refresh Captcha
function refreshCaptcha(){
```

```

var img = document.images['captcha_image'];

img.src = img.src.substring(
    0, img.src.lastIndexOf("?")
) + "?rand=" + Math.random() * 1000;
}
</script>

```

Add the above JavaScript in the footer of **index.php**. This script will refresh the captcha if it is very difficult to read, so that user can insert new captcha code.

PHP Script

Enter the below php validation script in the header of **index.php** file, before starting the **<html>** tag.

```

<?php

session_start();

$status = '';

if ( isset($_POST['captcha']) && ($_POST['captcha']!="") ){

// Validation: Checking entered captcha code with the generated
captcha code

if(strcasecmp($_SESSION['captcha'], $_POST['captcha']) != 0){

// Note: the captcha code is compared case insensitively.

// if you want case sensitive match, check above with strcmp()

$status = "<p style='color:#FFFFFF; font-size:20px'>

```

```

<span style='background-color:#FF0000;'>Entered captcha code does not
match!

Kindly try again.</span></p>";

}else{

$status = "<p style='color:#FFFFFF; font-size:20px'>

<span style='background-color:#46ab4a;'>Your captcha code is
match.</span>

</p>";

        }

}

echo $status;

?>

```

The above script is matching the captcha code with the users input and showing message of success or failure.

2. Create a captcha.php file

Now most important step, create a **captcha.php** file and copy paste the below script in it.

PHP Script

```

<?php

session_start();

//You can customize your captcha settings here

```

```
$captcha_code = '';

$captcha_image_height = 50;

$captcha_image_width = 130;

$total_characters_on_image = 6;

//The characters that can be used in the CAPTCHA code.

//avoid all confusing characters and numbers (For example: l, 1 and i)

$possible_captcha_letters = 'bcdfghjkmnpqrstvwxyz23456789';

$captcha_font = './monofont.ttf';

$random_captcha_dots = 50;

$random_captcha_lines = 25;

$captcha_text_color = "0x142864";

$captcha_noise_color = "0x142864";

$count = 0;

while ($count < $total_characters_on_image) {

$captcha_code .= substr(

    $possible_captcha_letters,
```

```
        mt_rand(0, strlen($possible_captcha_letters)-1),
        1);
$count++;
}

$captcha_font_size = $captcha_image_height * 0.65;
$captcha_image = @imagecreate(
    $captcha_image_width,
    $captcha_image_height
);

/* setting the background, text and noise colours here */
$background_color = imagecolorallocate(
    $captcha_image,
    255,
    255,
    255
);

$array_text_color = hextorgb($captcha_text_color);
```

```
$captcha_text_color = imagecolorallocate(
    $captcha_image,
    $array_text_color['red'],
    $array_text_color['green'],
    $array_text_color['blue']
);

$array_noise_color = hextorgb($captcha_noise_color);
$image_noise_color = imagecolorallocate(
    $captcha_image,
    $array_noise_color['red'],
    $array_noise_color['green'],
    $array_noise_color['blue']
);

/* Generate random dots in background of the captcha image */
for( $count=0; $count<$random_captcha_dots; $count++ ) {
    imagefilledellipse(
        $captcha_image,
        mt_rand(0,$captcha_image_width),
```

```
        mt_rand(0,$captcha_image_height),

        2,

        3,

        $image_noise_color

    );

}

/* Generate random lines in background of the captcha image */
for( $count=0; $count<$random_captcha_lines; $count++ ) {
    imageline(

        $captcha_image,

        mt_rand(0,$captcha_image_width),

        mt_rand(0,$captcha_image_height),

        mt_rand(0,$captcha_image_width),

        mt_rand(0,$captcha_image_height),

        $image_noise_color

    );

}

/* Create a text box and add 6 captcha letters code in it */
```

```
$text_box = imagettfbbox(
    $captcha_font_size,
    0,
    $captcha_font,
    $captcha_code
);

$x = ($captcha_image_width - $text_box[4])/2;
$y = ($captcha_image_height - $text_box[5])/2;

imagettftext(
    $captcha_image,
    $captcha_font_size,
    0,
    $x,
    $y,
    $captcha_text_color,
    $captcha_font,
    $captcha_code
);

/* Show captcha image in the html page */
```



```

// defining the image type to be shown in browser widow

header('Content-Type: image/jpeg');

imagejpeg($captcha_image); //showing the image

imagedestroy($captcha_image); //destroying the image instance

$_SESSION['captcha'] = $captcha_code;

function hextorgb ($hexstring){

    $integar = hexdec($hexstring);

    return array("red" => 0xFF & ($integar >> 0x10),

                "green" => 0xFF & ($integar >> 0x8),

                "blue" => 0xFF & $integar);

    }

?>

```

In the above code, I include a font name monofont **\$captcha_font = './monofont.ttf'**; you can use any font that you want otherwise you can download this font from [here](#).

Make sure that you keep this font in the same folder where you are keeping **index.php** and **captcha.php** files.

Now you can browse **index.php** file on your local host or online web host if you have. Our captcha form is now ready to use.