```
<?php

/**
 * interface/patient_file/barcode_label.php Displaying a PDF file of Labels for printing.
 *
 * Program for displaying Barcode Label
 * via the popups on the left nav screen
 *
 * this is from the barcode-coder and FPDF website I used the examples and code snippets
 * listed on the sites to create this program
 *
 *
 * @package   OpenEMR
 * @link      http://www.open-emr.org
 * @author    Terry Hill <terry@lillysystems.com>
 * @copyright Copyright (c) 2014 Terry Hill <terry@lillysystems.com>
 * @license   https://github.com/openemr/openemr/blob/master/LICENSE GNU General Public License 3
 */

require_once("../globals.php");
require_once("{$srcdir}/formatting.inc.php");



ini_set('display_errors', 'on');
error_reporting(E_ALL);
//Get the data to place on labels

$sql = "SELECT
```

```
    p.fname
    , p.mname
    , p.lname
    , p.pubpid
    , p.DOB
    , p.street
    , p.city
    , p.state
    , p.postal_code
    , p.pid
    FROM patient_data AS p
    WHERE p.pid = ? LIMIT 1";
$patdata = sqlQuery($sql, [$pid]);
$today = date('m/d/Y');
$dob = oeFormatShortDate($patdata['DOB']);



// ---------------------------------------------------- //
//          BARCODE DATA AND TYPE
// ---------------------------------------------------- //

$code    = $patdata['pubpid']; // what is wanted as the barcode
$bartype = $GLOBALS['barcode_label_type'] ; // Get barcode type

switch ($bartype) {
    case '1':
        $type    = 'std25';
        break;
```

```
case '2':
  $type    = 'int25';
  break;
case '3':
  $type    = 'ean8';
  break;
case '4':
  $type    = 'ean13';
  break;
case '5':
  $type    = 'upc';
  break;
case '6':
  $type    = 'code11';
  break;
case '7':
  $type    = 'code39';
  break;
case '8':
  $type    = 'code93';
  break;
case '9':
  $type    = 'code128';
  break;
case '10':
  $type    = 'codabar';
  break;
case '11':
  $type    = 'msi';
```

```
      break;
   case '12':
      $type     = 'datamatrix';
      break;
}

// ---------------------------------------------------- //
//               PROPERTIES
// ---------------------------------------------------- //
$fontSize = 8;
$angle    = 90;   // rotation in degrees
$black    = '000000'; // color in hexa
$primaryLogoSrc = imagerotate($image, 90, 0);
$rotate = imagerotate ($primaryLogSrc, degrees, 0);
$degrees = 90;



if ($GLOBALS['barcode_label_type'] == '12') {   // datamatrix
   $marge    = 0;   // between barcode and hri in pixel
   $x       = 35;  // barcode center
   $y       = 120;  // barcode center
   $height   = 20;   // barcode height in 1D ; module size in 2D
   $width    = 4;    // barcode height in 1D ; not use in 2D
} else {
   $marge    = 5;   // between barcode and hri in pixel
   $x       = 30;  // barcode center
   $y       = 120;  // barcode center
   $height   = 20;   // barcode height in 1D ; module size in 2D
```

```
   $width    = 1;    // barcode height in 1D ; not use in 2D
}

// ----------------------------------------------------- //
//          ALLOCATE FPDF RESSOURCE
// ----------------------------------------------------- //

$pdf = new eFPDF('P', 'mm', array(102,252)); // set the orentation, unit of measure and size of the page
$pdf->AddPage();

// ----------------------------------------------------- //
//                  BARCODE
// ----------------------------------------------------- //

$data = Barcode::fpdf($pdf, $black, $x, $y, $angle, $type, array('code' => $code), $width, $height);
$pdf->SetFont('Arial', 'B', $fontSize);
$pdf->SetTextColor(0, 0, 0);

$hri_string = "MRN: {$data['hri']}";
$patName = $patdata['fname'];
$patName .= (strlen($patdata['mname']) > 0) ? " {$patdata['mname']}" : "";
$patName .= " {$patdata['lname']}";
$pt_name_string = $patName;
$dob_string = "DOB: {$dob}";
$patAdd = $patdata['street']";
$patAdd = $patdata['city']";
$patAdd = " {$patdata['state']}";
$patAdd = " {$patdata['postal_code']}";
```

```
$pt_add_string = $patAdd;



$primaryLogoSrc = $GLOBALS["images_static_absolute"] . "/eiu-logo.png";

$hri_len = $pdf->GetStringWidth($hri_string);
$dob_len = $pdf->GetStringWidth($dob_string);
$len = ($hri_len > $dob_len) ? $hri_len : $dob_len;
Barcode::rotate(-$len / 2, ($data['height'] / 2) + $fontSize + $marge, $angle, $xt, $yt);

// ------------------------------------------------- //
//                 OUTPUT
// ------------------------------------------------- //



$pdf->Image($primaryLogoSrc, $x + $xt - 20, $y + $yt, $angle);
//$pdf->TextWithRotation($x + $xt - 20, $y + $yt, $$pdf->Image($primaryLogoSrc, $angle);//
$pdf->TextWithRotation($x + $xt - 11, $y + $yt, $pt_name_string, $angle);
$pdf->TextWithRotation($x + $xt - 7, $y + $yt, $pt_add_string, $angle);
$pdf->TextWithRotation($x + $xt - 4, $y + $yt, $hri_string, $angle);
$pdf->TextWithRotation($x + $xt + 1, $y + $yt, $dob_string, $angle);
$pdf->Output();
```