PHP Sample Application

Simple graphics and database

Sample PHP Application

• PHP is commonly used to:
  – Draw graphics on the fly
  – Access a database
• Let’s put these together to make an application that:
  – Lets a user make a query on a web form
  – Looks up some data from an Access database (similar if using SQL database)
  – Displays a graph of the data from the database
Simple Graphics

- First, let’s see how to draw some simple graphics
- PHP 5 includes the “GD” graphics library, created by Boutell.com
- PHP 4 requires an extension be added to access this library
- PNG format used
  - Similar but superior to GIF
  - GIF patented by Unisys, expired 2003

Base image functions

- `image imagecreate(width,height)`
  - Creates and returns an image of specified height
- `color imagecolorallocate(image, R,G,B)`
  - Creates a color for image with the given Red, Green, and Blue colors
- `boolean imagefill(image, x, y, color)`
  - Flood fills at the given point with the given color
- `header("Content-type: image/png");`
  - Displays the MIME header for a PNG image
- `imagepng(image)`
  - Displays the data for the PNG image
Example: Green image

```php
<?
$image = imagecreate(300,200);
$colorGreen = imagecolorallocate($image, 0, 255, 0);
imagefill($image, 0, 0, $colorGreen);
header("Content-type: image/png");
imagepng($image);
?>
```

More image functions

- `imagestring (image, int font, int x, int y, string s, int col)`
  - Draws the string s in the image at coordinates x, y in color col. If font is 1, 2, 3, 4 or 5, a built-in font is used.
- `imagefilledrectangle(image, x1,y1, x2,y2,color)`
  - Draws a filled rectangle at upper left corner x1,y1 bottom right corner x2,y2
- `imagefilledellipse(image, x1,y1, width,height,color)`
  - Draws a filled ellipse in the bounding rectangle specified by x1,y1,x2,y2
- `imagerectangle(image,x1,y1,x2,y2,color)`
  - Rectangle with no fill
- `imageellipse(image, x1,y1, width,height,color)`
  - Ellipse with no fill
Image example

```php
Cómo crear una imagen de prueba:

```php
<?
$image = imagecreate(300, 200);
$colorWhite = imagecolorallocate($image, 255, 255, 255);
imagefill($image, 0, 0, $colorWhite);
imagefilledrectangle($image, 50, 50, 100, 75, imagecolorallocate($image, 255, 0, 0));
imageellipse($image, 150, 50, 100, 50, imagecolorallocate($image, 0, 0, 255));
imagestring($image, 0, 10, 10, "Hello!", imagecolorallocate($image, 0, 0, 0));
header("Content-type: image/png");
imagepng($image);
?>
```

ImagePolygon and ImageLine

```php
<?
$image = imagecreate(300, 200);
$colorWhite = imagecolorallocate($image, 255, 255, 255);
imagefill($image, 0, 0, $colorWhite);

$colorBlack = imagecolorallocate($image, 0, 0, 0);
imageLine($image, 50, 0, 200, 150, $colorBlack);

$pointsTriangle = array(50, 10, 10, 90, 90, 90);
imagePolygon($image, $pointsTriangle, count($pointsTriangle)/2, $colorBlack);

header("Content-type: image/png");
imagepng($image);
?>
```

How could you display this image multiple times in a web page?
Many more image functions

- Read jpg, gif, etc.
- Rotate, blending
- Resize
- Draw arcs
- Gamma correct
- Tiling
- …
- PHP Reference or textbook is a great way to learn about these functions, with code samples

Drawing Graphs

- What if you wanted to draw charts and graphs?
  - Pie chart, bar chart, line chart, scatter plot, etc.
  - You could do it using the graphics primitives we have covered
- Fortunately, someone has already done this for you
  - JPGraph Library
  - http://www.aditus.nu/jpgraph/
Let’s look at a simple bar chart

```php
<?php
include("jpgraph2/jpgraph.php");
include("jpgraph2/jpgraph_bar.php");

$datay = array(12, 8, 19, 3, 10, 50);
$datax = array(10, 20, 30, 40, 50, 60);

// Create the graph. These two calls are always required
$graph = new Graph(300, 200, "auto");
$graph->SetScale("textlin"); // Natural nums

// Add a drop shadow
$graph->SetShadow();

// Adjust the margin a bit to make more room for titles
$graph->img->SetMargin(40, 30, 20, 40);

// Create a bar plot
$bpplot = new BarPlot($datay);
$graph->Add($bpplot);
```
Simple Bar chart, cont.

// Add data to X coordinate
// $graph->xaxis->SetTickLabels($datax);

// Create and add a new text
$txt=new Text("This is a text");
$txt->SetPos(0,0);
$txt->SetColor("red");
$graph->AddText($txt);

// Setup the titles
$graph->title->Set("A simple bar graph");
$graph->xaxis->title->Set("X-title");
$graph->yaxis->title->Set("Y-title");

$graph->title->SetFont(FF_FONT1,FS_BOLD);
$graph->xaxis->title->SetFont(FF_FONT1,FS_BOLD);
$graph->yaxis->title->SetFont(FF_FONT1,FS_BOLD);

// Display the graph
$graph->Stroke();
?>

Database Access

• Changing directions, reading from a database and displaying data in a graph
  – In our example, let’s read from a MySQL “Test” database with name popularity over the decades
    • name
    • yr1900
    • yr1910
    • …
    • yr2000
  – PHP app: let the user enter a name then graph the popularity
Database Schema

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Null</th>
<th>Key</th>
<th>Default</th>
<th>Extra</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>varchar(50)</td>
<td>NO</td>
<td>PRI</td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr1980</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr1990</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr1990</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr1990</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr1990</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr1990</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr1990</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr1990</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
<tr>
<td>yr2000</td>
<td>int(11)</td>
<td>YES</td>
<td></td>
<td>NULL</td>
<td></td>
</tr>
</tbody>
</table>

```
mysql> select name, yr1960, yr1970, yr1980, yr1990, yr2000 from names where name = 'Kyler'

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Kyler</td>
<td>1001</td>
<td>1001</td>
<td>978</td>
<td>978</td>
<td>281</td>
</tr>
</tbody>
</table>
```

Building our App

• First, make a form that asks for a username and then retrieves and prints all the yearly data that matches the name in the table
<?php
header("Content-Type: text/html");
print("<HTML><HEAD><TITLE>Name Popularity Surfer</TITLE>");
print("</HEAD>" negótye $\text{\textbackslash n}!

if($_SERVER['REQUEST_METHOD'] != "POST")
{
    print("<FORM method=post action='names.php'>");
    print("Enter name.<p>");
    print("<INPUT type=text name='name'>");
    print("<INPUT type=submit>");
    print("</FORM>");
}

else
{
    $name = $_REQUEST['name'];
    // Database parameters
    $db_location = "localhost";
    $db_user_name = "test";
    $db_password = "test";
    $database_name = "test";
    // Connect to the DB
    $dbcnx = mysql_connect($db_location,$db_user_name,$db_password);
    mysql_select_db($database_name);
    $sql="SELECT * FROM names where name=" . $name . "\n;";
    $rs=mysql_query($sql);
}
if ($row = mysql_fetch_assoc($rs))
{
    $arr = array();
    $arr[] = $row['yr1900'];
    for ($i = 10; $i < 100; $i+=10)
        $arr[] = $row['yr19'. $i];
    $arr[] = $row['yr2000'];
    print_r($arr);
    print("<br>");
}
if (!isset($arr))
    print("$name not found in the database.");
print("<\/BODY>");
?>

Drawing Graph

• Now let’s hook this up with jpgraph to draw a graph of the popularity over time
• You could run this yourself if you copy/install the jpgraph folder to your own HTML folder
  – Can access database from any account using the supplied username/password
<?php
include("jpgraph.php");
include("jpgraph_line.php");

if($_SERVER['REQUEST_METHOD'] != "POST")
{
    print("<FORM method=post action="names.php">\n    Enter name.<p>\n    <INPUT type=text name='name'>\n    <INPUT type=submit>\n    </FORM>\n    ");
}
else
{
    $name = $_REQUEST['name'];
// Database parameters
$db_location = "localhost";
$db_user_name = "test";
$db_password = "test";
$database_name = "test";
// Connect to the DB
$dbcnx = mysql_connect($db_location,$db_user_name,$db_password);
mysql_select_db($database_name);
$sql="SELECT * FROM names where name='$name';";
$rs=mysql_query($sql);
if ($row = mysql_fetch_assoc($rs))
{
    $ydata = array();
    $xdata = array();
    $ydata[] = $row['yr1900'];
    $xdata[] = 1900;
    for ($i = 10; $i < 100; $i+=10)
    {
        $ydata[] = $row['yr19'.$i];
        $xdata[] = 1900 + $i;
    }
    $ydata[] = $row['yr2000'];
    $xdata[] = 2000;
    // Create the graph. These two calls are always required
    $graph = new Graph(600,400,"auto");
    $graph->SetScale("linlin");
    // Add a drop shadow
    $graph->SetShadow();
    // Adjust the margin a bit to make room for titles
    $graph->img->SetMargin(40,30,20,40);
    // Create a line plot
    $plot = new LinePlot($ydata,$xdata);
    $plot->SetColor("blue");
    $plot->SetWeight(2);
    $graph->Add($plot);
    // Add data to X coordinate
    $graph->xaxis->SetTickLabels($xdata);
    // Setup the titles
    $graph->title->Set("Popularity for "$name");
    $graph->xaxis->title->Set("Year");
    $graph->yaxis->title->Set("Ranking");
    $graph->title->SetFont(FF_FONT1,FS_BOLD);
    $graph->yaxis->title->SetFont(FF_FONT1,FS_BOLD);
    $graph->xaxis->title->SetFont(FF_FONT1,FS_BOLD);
    // Display the graph
    $graph->Stroke();
}
else
{
    header("Content-Type: text/html");
    print("<HTML><HEAD><TITLE>Name Surfer</TITLE><TITLE>\n    </HEAD>\n    <BODY>
    $name not found in the database.
    </BODY></HTML>\n    ");
}
?>
Output

Summary

- We discussed how to integrate business graphics with a simple database web application
- Should have error checking for no user found, SQL injection, etc.
- Similar process for updating and modifying the database
- Very easy to make simple graphics
- Libraries for more complex graphs
- Overall it is fairly easy to create sophisticated web applications
  - Other environments like .NET encapsulate much of the primitive HTML information, e.g. datagrid