Exercises

1.

Write a function to return the HTML an <img /> tag. The function should accept a mandatory argument of the image URL and optional arguments for alt text, height, and width.

2.

Modify the function in the previous exercise so that the filename only is passed to the function in the URL argument. Inside the function, prepend a global variable to the filename to make the full URL. For example, if you pass *photo.png* to the function, and the global variable contains /images/, then the src attribute of the returned <img> tag would be /images/photo.png. A function like this is an easy way to keep your image tags correct, even if the images move to a new path or a new server. Just change the global variable – for example, from /images/ to http://images.example.com/.

3.

Put your function from the previous exercise in one file. Then make another file which loads the first file and uses it to print out some <img /> tags.

4.

What does the following code print out?

```
<?php

function restaurant_check($meal, $tax, $tip) {
        $tax_amount = $meal * ($tax / 100);
        $tip_amount = $meal * ($tip / 100);
        return $meal + $tax_amount + $tip_amount;
}
$cash_on_hand = 31;
$meal = 25;
$tax = 10;
$tip = 10;
while(($cost = restaurant_check($meal,$tax,$tip)) < $cash_on_hand) {
        $tip++;
        print "I can afford a tip of $tip% ($cost)\n";
}
?>
```

5.

Web colors such as #ffffff and #cc3399 are made by concatenating the hexadecimal color values for red, green, and blue. Write a function that accepts decimal red, green, and blue arguments and returns a string containing the

appropriate color for use in a web page. For example, if the arguments are 255, 0, and 255, then the returned string should be #ff00ff. You may find it helpful to use the built-in function dechex( ), which is documented at http://www.php.net/dechex.