Connecting to MySQL from PHP

```
<?
   $link = mysql_connect($hostname, $username, $password)
         or die("Could not open connection to database");
?>
```

Now the variable $link contains the information
for your connection to MySQL. You can disconnect using:

```
<?
   mysql_close($link)
        or die("Could not close connection to database");
?>
```

Select a database, issue queries

Once you have connected successfully, you should select the database you will use. Then you can issue queries to the database.

```
<?
  mysql_select_db("mysqlf01??_db", $link)
     or die("Could not select database");

  $result = mysql_query("select * from some_table")
     or die("Could not issue MySQL query");
?>
```

Now the variable $result will be used to reference the query we just made.

Array handling in PHP

In PHP, arrays are associative. Any number or string can be used to index an array.

```
<?
    $array["hello"] = 3;
    $array[5] = "how are you?";
    while (list($column, $value) = each($array)) {
        print("$column = $value\n");
    }
    reset($array);
?>
```

This example would print:

```
hello = 3
5 = how are you?
```

The command reset($array) puts the array iterator back to the beginning of the array.

Getting the results from a query

After issuing a query, you retreive the results using the
variable $result.

```
<?
    if (mysql_num_rows($result) == 0) {
        print("No results matching your query<BR>\n");
    } else {
        print("here are the results:<BR>\n");
        while ($row = mysql_fetch_row($result)) {
            while (list($colname, $value) = each($row)) {
                print("$value  ");
            }
            print("<BR>\n");
        }
    }
?>
```

HTML forms and PHP variables

From an HTML page that had this form:

```
<FORM ACTION="target.php" METHOD="GET">
 <INPUT TYPE="TEXT" NAME="myvar">
 <INPUT TYPE="TEXT" NAME="yourvar">
</FORM>
```

The PHP script (target.php) will receive the variables from the form by the same name:

```
<?
    print("you entered myvar = $myvar, yourvar = $yourvar\n");
?>
```

Useful string stuff in PHP

Difference between single- and double-quoted strings:

```
<?   $name = "Joe";
     print("hello, your name is $name\n");
     print('hello, your name is $name\n');
?>
Output:  hello, your name is Joe
         hello, your name is $name
```

```
$trimmed = trim("  this string has whitespace   ");
```
removes leading and trailing whitespace

```
$encoded = urlencode("this is a non-encoded url string");
```
changes the argument so that it will be part of a valid URL