

PHP and MySQL CMS Login

Many interactive websites nowadays require a user to log in into the website's system to provide a customized experience for the user. Once the user has logged in, the website will be able to provide a presentation that is personalized to the user's preferences. A basic login system typically contains 3 components which can be created using PHP and MySQL :

Component 1: Allows registration of preferred login Id and password. This is created in simple HTML form that contains 3 fields and 2 buttons:

1. A preferred login id field
2. A preferred password field
3. A valid email address field
4. A Submit button
5. A Reset button

Lets say the form is coded into a file named register.html. The following HTML code extract is a typical example. When the user has filled in all the fields and clicks on the submit button, the register.php page is called for.

```
[form name="register" method="post" action="register.php"] [input name="login id" type="text" value="loginid" size="20"/>  
[input name="password" type="text" value="password" size="20"/>  
[input name="email" type="text" value="email" size="50"/>  
[input type="submit" name="submit" value="submit"/>  
[input type="reset" name="reset" value="reset"/>  
[/form]
```

The following code extract can also be used as part of register.php to process the registration. The code connects to the MySQL database and inserts a line of data into the table used to store the registration information.

```
@mysql_connect("localhost", "mysql_login", "mysql_pwd") or die("Cannot connect to DB!");  
@mysql_select_db("tbl_login") or die("Cannot select DB!");  
$sql="INSERT INTO login_tbl (loginid, password and email) VALUES
```

```
(".$loginid.", ".$password.", ".$email.");  
$r = mysql_query($sql);  
if(!$r) { $err=mysql_error(); print $err; exit(); }
```

The code extract assumes that the MySQL table that is used to store the registration data is named tbl_login and contains 3 fields - the loginid, password and email fields. The values of the \$loginid, \$password and \$email variables are passed in from the form in register.html using the post method.

Component 2: Verification and authentication of the user. In this the HTML form typically contains 2 fields and 2 buttons:

1. A login id field
2. A password field
3. A Submit button
4. A Reset button

Assume that such a form is coded into a file named authenticate.html. The following HTML code extract is a typical example. When the user has filled in all the fields, the authenticate.php page is called when the user clicks on the Submit button.

```
[form name="authenticate" method="post" action="authenticate.php"]  
[input name="login id" type="text" value="loginid" size="20"/>  
[input name="password" type="text" value="password" size="20"/>  
[input type="submit" name="submit" value="submit"/>  
[input type="reset" name="reset" value="reset"/>  
[/form]
```

The following code extract can be used as part of authenticate.php to process the login request. It connects to the MySQL database and queries the table used to store the registration information.

```
@mysql_connect("localhost", "mysql_login", "mysql_pwd") or die("Cannot connect to DB!");  
@mysql_select_db("tbl_login") or die("Cannot select DB!");  
$sql="SELECT loginid FROM login_tbl WHERE loginid='.$loginid.' and password='.$password.'";  
$r = mysql_query($sql);  
if(!$r) { $err=mysql_error(); print $err; exit(); }
```

```
if(mysql_affected_rows()==0){ print "no such login in the system. please try again."; exit(); }  
else{ print "successfully logged into system."; //proceed to perform website's functionality - e.g. present information to the user }
```

As in component 1, the code excerpt assumes that the MySQL table that is used to store the registration data is named tbl_login and contains 3 fields - the loginid, password and email fields. The values of the \$loginid and \$password variables are passed in from the form in authenticate.html using the post method.

Component 3: When the user forgets his logion password this 3rd component sends his password to the users registered email address. The HTML form typically contains 1 field and 2 buttons:

- A login id field
- A Submit button
- A Reset button

Assume that such a form is coded into a file named forgot.html. The following HTML code excerpt is a typical example. When the user has filled in all the fields, the forgot.php page is called when the user clicks on the Submit button.

```
[form name="forgot" method="post" action="forgot.php"] [input name="login id" type="text" value="loginid" size="20"/>  
[input type="submit" name="submit" value="submit"/>  
[input type="reset" name="reset" value="reset"/>  
[/form]
```

The following code excerpt can be used as part of forgot.php to process the login request. It connects to the MySQL database and queries the table used to store the registration information.

```
@mysql_connect("localhost", "mysql_login", "mysql_pwd") or die("Cannot connect to DB!");  
@mysql_select_db("tbl_login") or die("Cannot select DB!");  
$sql="SELECT password, email FROM login_tbl WHERE loginid='.$loginid.'";  
$r = mysql_query($sql);  
if(!$r) { $err=mysql_error(); print $err; exit(); }
```

```
if(mysql_affected_rows()==0){ print "no such login in the system. please try again."; exit(); }  
else { $row=mysql_fetch_array($r);  
$password=$row["password"];  
$email=$row["email"];  
$subject="your password";
```

```
$header="from:you@yourdomain.com";  
$content="your password is ".$password; mail($email, $subject, $row, $header);  
print "An email containing the password has been sent to you"; }
```

As in component 1, the code excerpt assumes that the MySQL table that is used to store the registration data is named tbl_login and contains 3 fields - the loginid, password and email fields. The value of the \$loginid variable is passed from the form in forgot.html using the post method.

This is how a basic login system can be created. The software developer can include additional tools like password encryption, access to the user profile in case they wish to edit their profile etc.

This article has been compiled by the content development team at Pegasus InfoCorp which pulls subject matter specialists from different work domains. They can be contacted through the Pegasus InfoCorp website at info@pegasusinfocorp.com.

Pegasus InfoCorp is an India based web design, web development and online/offline software development company. Please visit <http://www.pegasusinfocorp.com> to read more articles and know more about us!

Other

companies and organizations are welcome to reprint this article on their websites provided the following conditions are met. ? The article is not changed in any manner ? The article is copied as is in its entirety (including back links to the Pegasus InfoCorp website). ? The company/ organization reprinting the article agrees to defend, indemnify and hold harmless Pegasus InfoCorp, its employees, directors, officers, agents, partners and their successors and assigns from and against any and all liabilities, damages, losses, costs and expenses, including attorney's fees, caused by or arising out of claims based upon the use of the article, including any claim of libel, defamation, violation of rights of privacy or publicity, loss of service by subscribers and infringement of intellectual property or other rights

About the Author

Source: <http://>