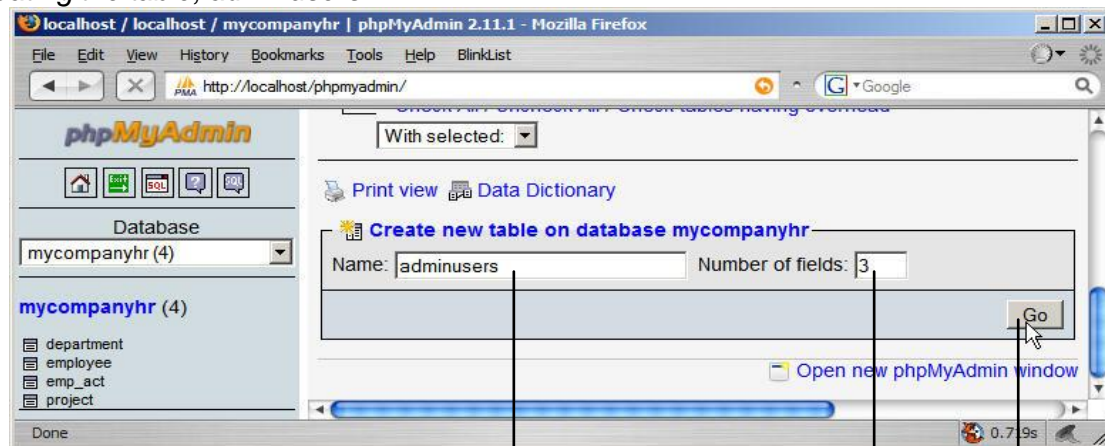


### Creating the *adminusers* table

For this exercise, we will be using the EMPNO as the username and a password given to the username. The information on the user's password and authorization level will be stored in another table named *adminusers* in the same *mycompanyhr* database.

So now lets create the *adminusers* table in the *mycompanyhr* table. Go to the phpmyadmin in your favorite browser and choose the database *mycompanyhr*.

#### Creating the table, *adminusers*



Type the table name here, *adminusers*

Number of fields, 3 is enough

Click 'Go' to create the table

### Setting the fields specifications, for table adminusers

The screenshot shows the phpMyAdmin interface for the 'mycompanyhr' database. The 'Table: adminusers' structure is being edited. The fields are defined as follows:

Field	Type	Length/Values	Collation	Attributes	Null	Default	Extra	Comments
EMPNO	VARCHAR	6			not null			
PASSWORD	VARCHAR	200			not null			
LEVEL	VARCHAR	1			not null			

The 'Storage Engine' is set to 'MyISAM' and the 'Collation' is empty. The 'Save' button is highlighted with a mouse cursor.

Use the **EMPNO** to maintain the employer's id. This way, only the employee in listed in the employee table has the right to become an administrator.

Save the fields' specifications.

### Insert an admin user with MD5 password

The screenshot shows the 'Insert' operation for the 'adminusers' table. The fields are defined as follows:

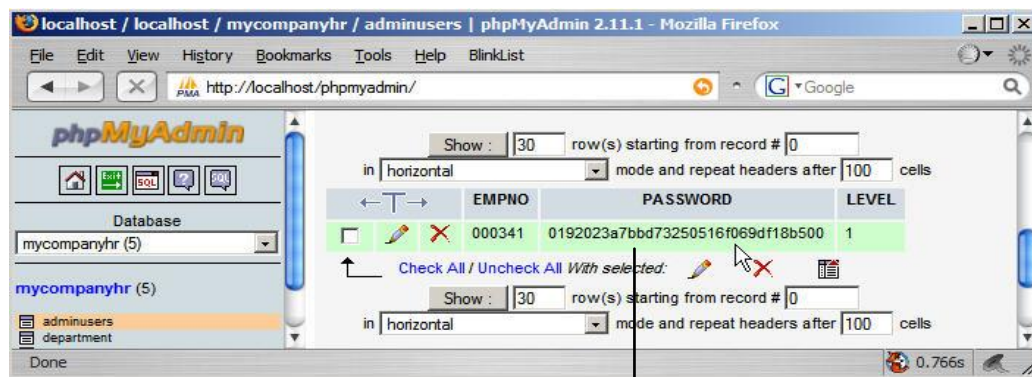
Field	Type	Function	Null	Value
EMPNO	varchar(6)			000341
PASSWORD	varchar(200)	MD5		admin123
LEVEL	varchar(1)			1

The 'Go' button is highlighted with a mouse cursor.

Set the additional MD5 function to be performed to the password

Do remember the password

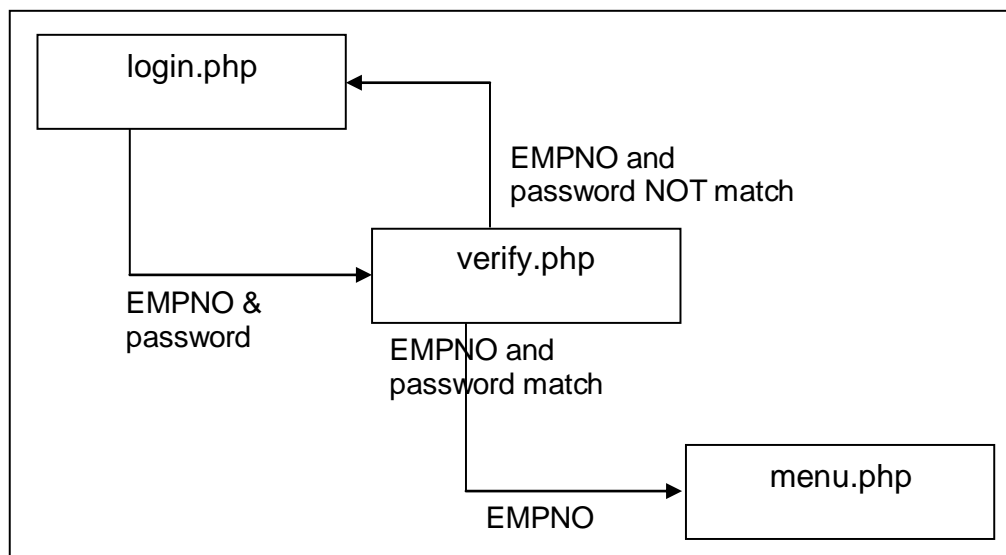
### The admin user with MD5 password



The MD5 password (the actual data is admin123)

### Simple Login page

This login facility contains a login form, a verification script and a simple system menu to access the other administration parts.



The first page is *login.php* which contains a form to receive user's username and password. The username and password are then sent to another page (*verify.php*) for verification. In this verification process the username and the password entered by the user will be compared to the username and the password stored in the table. The user will be directed to the administration menu (*menu.php*) if and only if the username and the password are match.

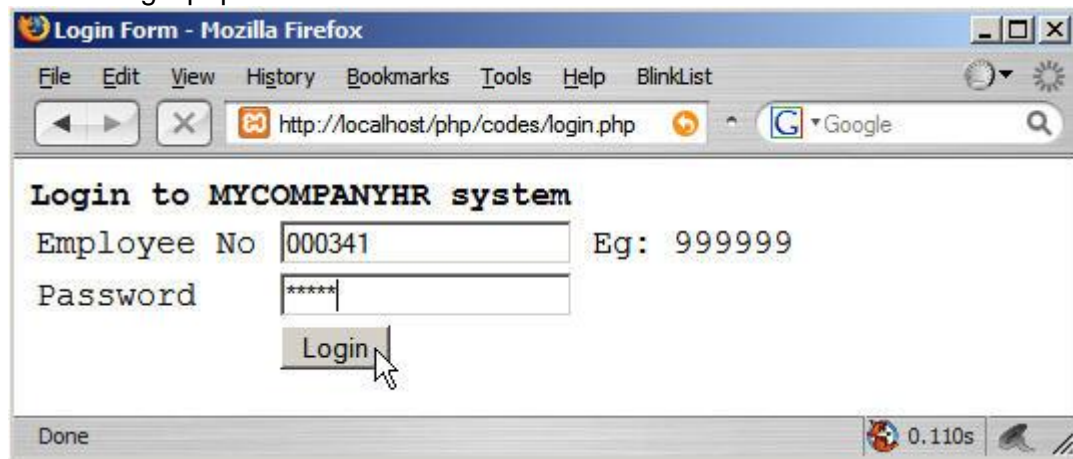
Filename: **login.php**

```
<html>
<head>
<title>Login Form</title>
</head>
<body>

<strong>Login to MYCOMPANYHR system</strong>
<form name="formlogin" method="post" action="verify.php">
<table width="400" border="0">
  <tr>
    <td>Employee No</td>
    <td><input name="EMPNO" type="text" maxlength="6">
      Eg: 999999 </td>
  </tr>
  <tr>
    <td>Password</td>
    <td><input name="PASSWORD" type="password"></td>
  </tr>
  <tr>
    <td>&nbsp;</td>
    <td><input name="submit" type="submit" value="Login"></td>
  </tr>
</table>
</form>

</body>
</html>
```

## Screen shot: login.php



## Verification process

The next page is verification page which consists of script to compare the entered user's password against the password stored in the database. If the username (in this case the EMPNO) and password match, this script will direct the user to the administration menu page. If not, the user will be asked to re-enter the username and the password.

Filename: **verify.php**

```
<?php
$EMPNO=$_POST['EMPNO'];
$PASSWORD=$_POST['PASSWORD'];
include "connection.php";
$sql="select * from adminusers where EMPNO='$EMPNO'";
$rs=mysqli_query($db, $sql);
?>
<html>
<head>
<title>MyCOMPANYHR-verify</title>
</head>

<body>
<strong>Verify employee number and password</strong><br>
<?php
if(mysqli_num_rows($rs)==1){ //found one user
    $record=mysqli_fetch_array($rs);
    $DBPASSWORD=$record['PASSWORD'];//password from database
    $USERPASSWORD=md5($PASSWORD);//MD5 password key-in by user
    if($DBPASSWORD==$USERPASSWORD){
        //compare password from database against password entered by user
        echo "Username and password match,<br>";
        echo "WELCOME $EMPNO !!!<br>";
        echo "<a href='menu.php?EMPNO=$EMPNO'>
            Click to ADMIN Menu </a><br>";
    }

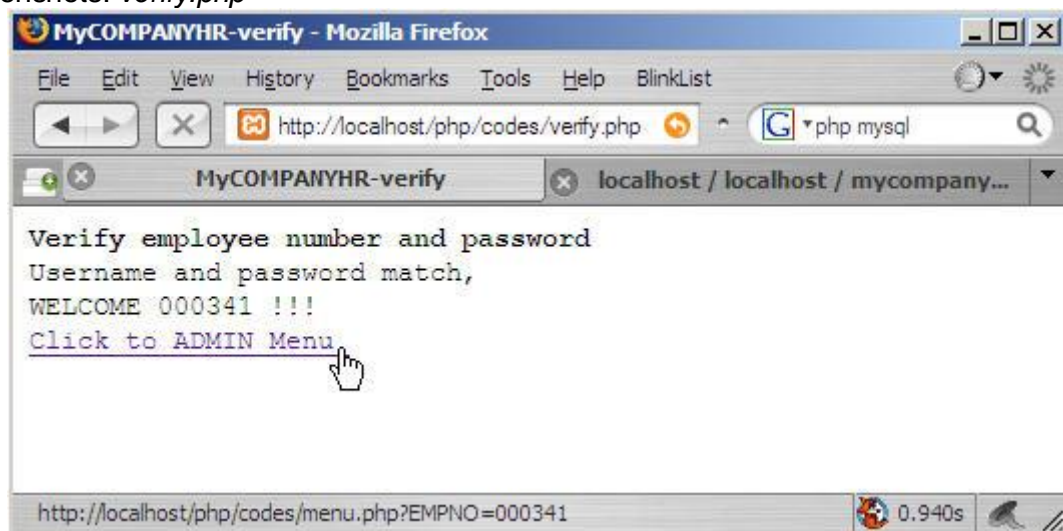
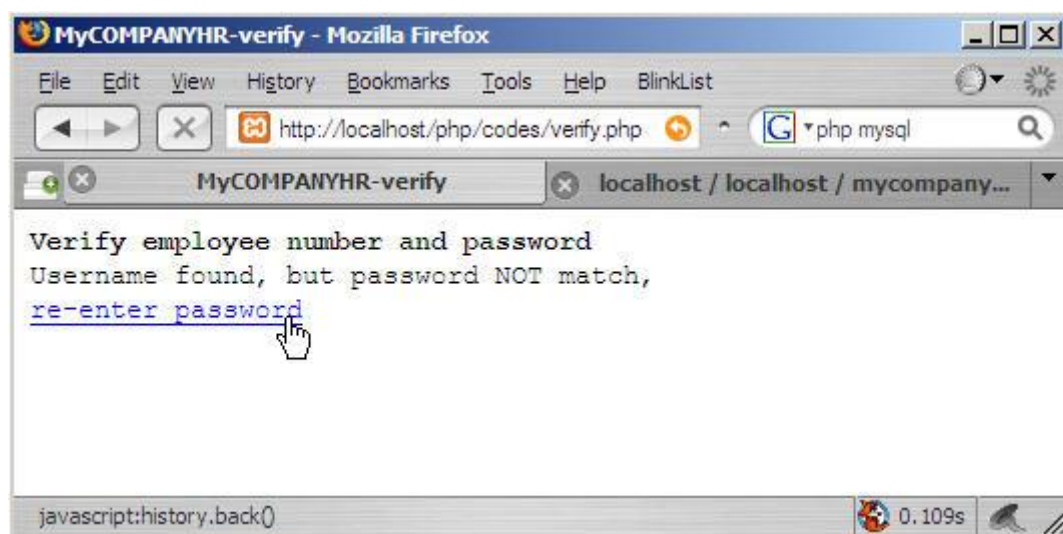
    else{
        echo "Username found, but password NOT match,<br>";
        echo "<a href='javascript:history.back()'>
            re-enter password</a><br>";
    }
}

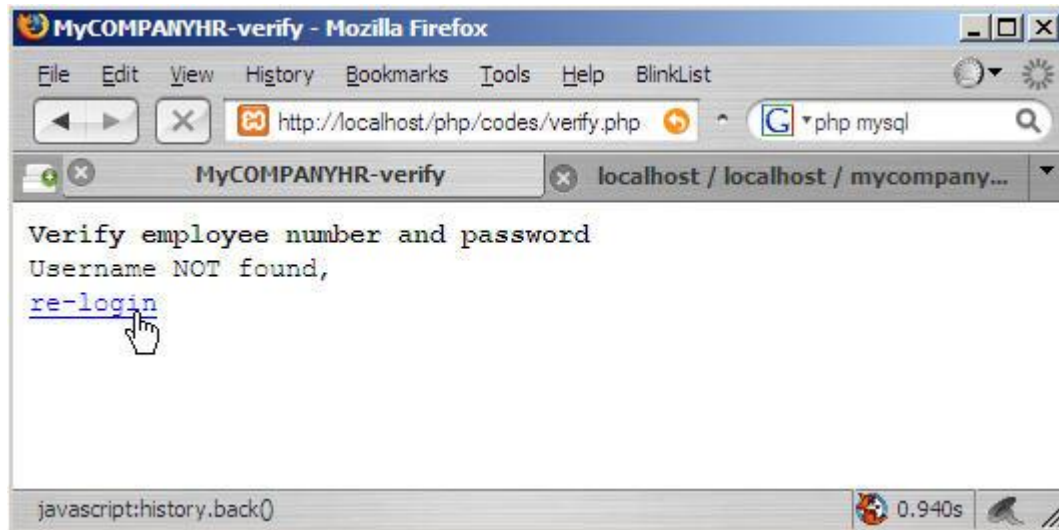
else{
    echo "Username NOT found,<br>";
    echo "<a href='javascript:history.back()'>re-login</a><br>";
}
?>
</body>
</html>
```



Code descriptions: *verify.php*

1. `$DBPASSWORD=$record['PASSWORD'];` - This script extracts the password (for the user by the EMPNO) from the table *adminusers*.  
`$USERPASSWORD=md5($PASSWORD);` - Meanwhile the password entered by the user from the previous form is extracted and decoded using MD5 function that also available in PHP.  
`if ($DBPASSWORD==$USERPASSWORD) {` - If the password entered by the user (after MD5 encoding) is equal to the MD5 encoded password in the *adminusers* table, then the username and password are matched.
2. This is the actions if the user is found in the *adminusers* table, but the password provided by the user is not match to the password stored in the table.
3. While this is the action if the username (provided by the user's EMPNO) is not found in the *adminusers* table.

Screenshots: *verify.php*Screen shots: *verify.php* – verification succeed.Screen shots: *verify.php* – password does not match.

Screen shots: *verify.php* – username not found.

### Menu page for full administration features and limited access

The authorised user is directed to this page. Before the menu is displayed to the user, full information regarding the user's access level, his first name, last name and working department is retrieved.

The list of pages authorised to the user are depend on the user access level. For this exercise there are user level 1 and 2. The users with access level 1 has the full administration access. The menu contains all the administration features such as searching, insert, update and delete employees' record. While the users with access level 2 is limited access users, they can only search the employees' records and update their own personel particulars.

This is the SQL to select user's level, firstname, lastname, workdept, deptname. Use JOIN statement since most of the informations are separated in different table.

```
SELECT
    adminusers.EMPNO,
    adminusers.LEVEL,
    employee.FIRSTNAME,
    employee.LASTNAME,
    employee.WORKDEPT,
    department.DEPTNAME
FROM
    adminusers
    INNER JOIN employee
        ON adminusers.EMPNO = employee.EMPNO
    INNER JOIN department
        ON employee.WORKDEPT = department.DEPTNO
WHERE adminusers.EMPNO='000341'
```

Filename: *menu.php*

```

<?php
include "connection.php";
?>
<html>
<head>
<title>MyCOMPANYHR-menu</title>
</head>

<body>
<strong>Menu for MyCOMPANYHR administration</strong><br>
<?php
//script to display employees information
$EMPNO=$_GET['EMPNO'];
/*this SQL commad will fetch the employee's administration level,
firstname, lastname, workdept, deptname in their respective tables*/
$sql="SELECT
        adminusers.EMPNO,
        adminusers.LEVEL,
        employee.FIRSTNAME,
        employee.LASTNAME,
        employee.WORKDEPT,
        department.DEPTNAME
    FROM adminusers
        INNER JOIN employee
            ON adminusers.EMPNO = employee.EMPNO
        INNER JOIN department
            ON employee.WORKDEPT = department.DEPTNO
    WHERE adminusers.EMPNO='$EMPNO'";

$rs=mysqli_query($db, $sql);
$record=mysqli_fetch_array($rs);
$level=$record['LEVEL'];
$firstname=$record['FIRSTNAME'];
$lastname=$record['LASTNAME'];
$workdept=$record['WORKDEPT'];
$deptname=$record['DEPTNAME'];

?>
Welcome, <? echo "$EMPNO $firstname $lastname"?> <br>
From department: <? echo "$workdept $deptname"?> <br>

<?php
//this menu displays depending on the users level
//if level is 1, full access
//if level is 2, limited access
if($level==1){
?>
    Menu : full access administration<br>
    1. <a href="searchform.php">Search employee</a><br>
    2. <a href="forminsert.php">Insert a new employee</a><br>
    3. <a href="listing4update.php">Update information of
        existing employee</a><br>
    4. <a href="listing4delete.php">Delete existing
        employee</a><br>

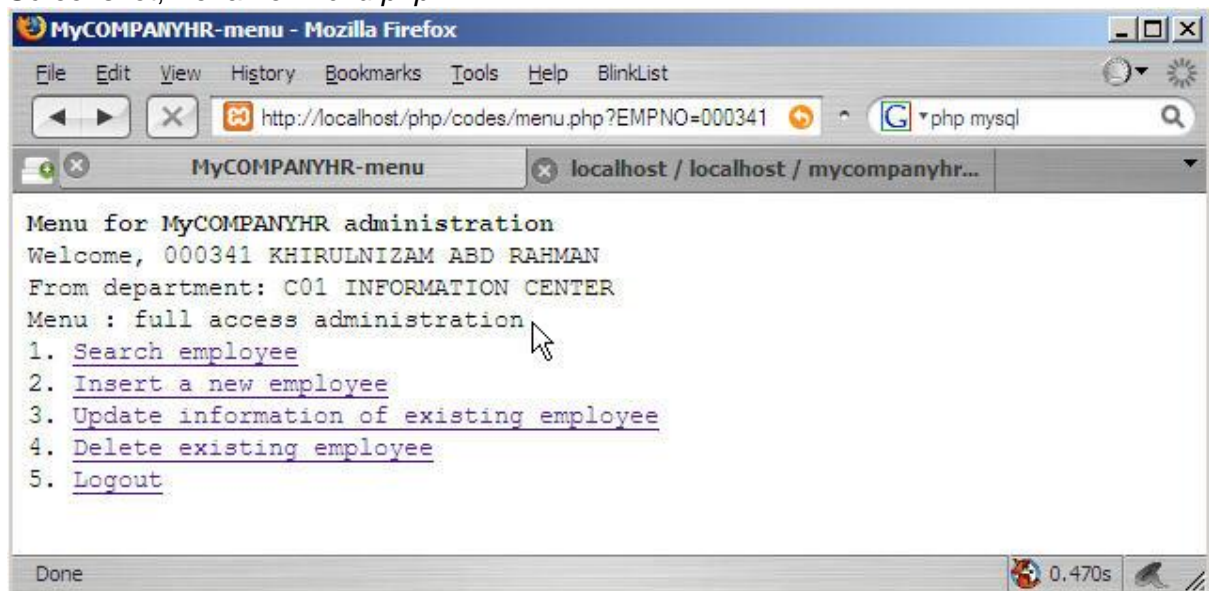
```



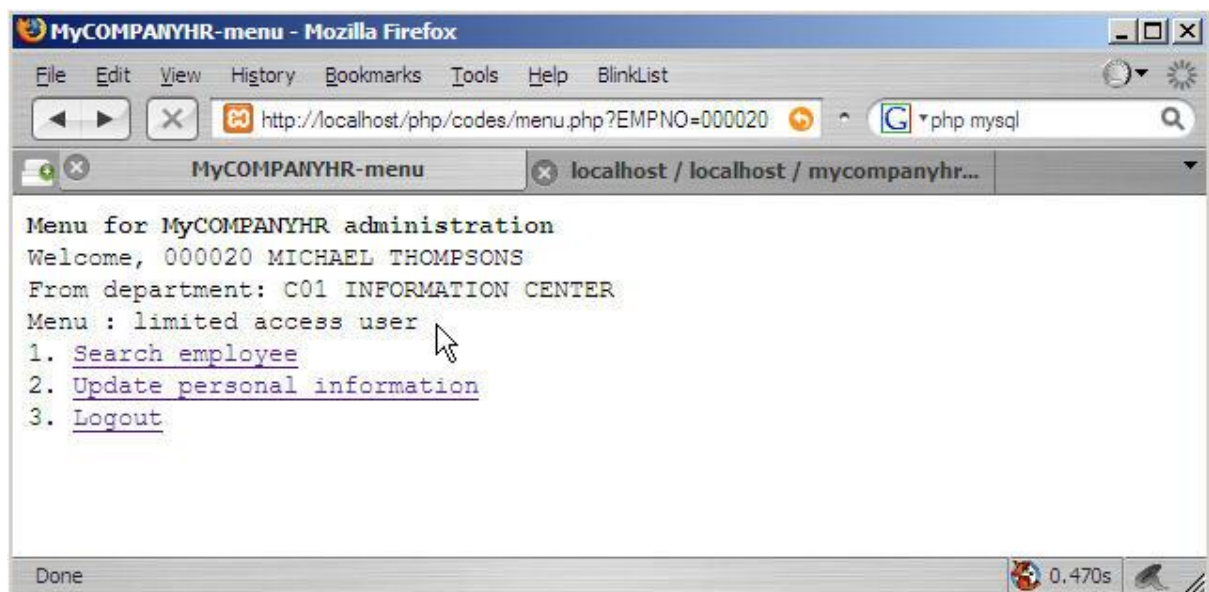
```

        5. <a href="logout.php">Logout</a><br>
<?php
} //end level 1
else if($level==2){
?>
    Menu : limited access user <br>
    1. <a href="searchform.php">Search employee</a><br>
    2. <a href="formupdate-personal.php">Update personal
        information</a><br>
    3. <a href="logout.php">Logout</a><br>
<?php
} //end level 2
?>
</body>
</html>

```

Screenshot, filename: *menu.php*

Menu for users with access level 1.



Menu for users with access level 2.