



IQRA NATIONAL UNIVERSITY

Department of Computer Science

Final Exam Spring 2020

ID: 11533

Name: Ashir Ali Khan

Subject : WEB TECH 3

Teacher: Sir Ayub

Q1. a. Why we use registration page explain?

Answer:

A registration form is a list of fields that a user will input data into and submit to a company or individual. There are many reasons why you would want a person to fill out a registration form. Companies use registration forms to sign up customers for subscriptions, services, or other programs or plan. Registration forms on landing pages are very important and should be optimized in order to encourage the users to sign up for your email list, product, subscription or service. We use registration form over the website because without filling up the registration form any website company can't get the details of any person who wants to login for future use. And from the users perspective it will be very helpful that they can login again into the website by entering credentials only.

b. Write a full program related to registration page in PHP as well as Html code?

The first thing we'll need to do is set up our database. create it on the MySQL prompt using the following SQL script:

```
CREATE TABLE `users` (  
  `id` int(11) NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  `username` varchar(100) NOT NULL,  
  `email` varchar(100) NOT NULL,  
  `password` varchar(100) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

CODING FOR REGISTRATION PAGE:

```
<?php include('server.php') ?>  
<!DOCTYPE html>  
<html>  
<head>
```

```

<title>Registration system PHP and MySQL</title>
<link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
  <div class="header">
    <h2>Register</h2>
  </div>

  <form method="post" action="register.php">
    <?php include('errors.php'); ?>
    <div class="input-group">
      <label>Username</label>
      <input type="text" name="username" value=""; ?>
    </div>
    <div class="input-group">
      <label>Email</label>
      <input type="email" name="email" value=""; ?>
    </div>
    <div class="input-group">
      <label>Password</label>
      <input type="password" name="password_1">
    </div>
    <div class="input-group">
      <label>Confirm password</label>
      <input type="password" name="password_2">
    </div>
    <div class="input-group">
      <button type="submit" class="btn" name="reg_user">Register</button>
    </div>
    <p>
      Already a member? <a href="login.php">Sign in</a>
    </p>
  </form>
</body>
</html>

```

Q2. a. Why Edit is used in pages explain?

Answer:

Edit means making changes or updating an existing record or page. Edit is one of the most useful creud operation. It allows the admin to edit all the data stored in the pages as well as in the database. we don't have to delete the whole line of code or data we can just edit it later on using edit. that edit is used to return the HTML form that is used to edit the resource values while update is the "action" that the edit form will be submitting to, and it responds to PUT or PATCH requests.

b. Write a full program related to edit page in PHP?

Answer:

While editing data, I have kept the entire thing in a single edit.php file. Edit form in HTML and database update code in PHP are present in the same file.

In the code below, at first a single row entry of data is fetched based on the id. The fetched data is displayed in the edit form. When user edits the data and submits the form, then some simple validation is done for empty data. When everything is correct, then that particular entry of data is updated in database. We will keep the file name as “edit.php”

CODE FOR EDIT PAGE:

```
<?php
// including the database connection file
include_once("config.php");

if(isset($_POST['update']))
{
    $id = $_POST['id'];

    $name=$_POST['name'];
    $age=$_POST['age'];
    $email=$_POST['email'];

    // checking empty fields
    if(empty($name) || empty($age) || empty($email)) {
        if(empty($name)) {
            echo "<font color='red'>Name field is empty.</font><br/>";
        }

        if(empty($age)) {
            echo "<font color='red'>Age field is empty.</font><br/>";
        }

        if(empty($email)) {
            echo "<font color='red'>Email field is empty.</font><br/>";
        }
    } else {
        //updating the table
```

```

    $result = mysqli_query($mysqli, "UPDATE users SET
name='$name',age='$age',email='$email' WHERE id=$id");
    //redirectig to the display page. In our case, it is index.php
    header("Location: index.php");
}
}
?>
<?php
//getting id from url
$id = $_GET['id'];
//selecting data associated with this particular id
$result = mysqli_query($mysqli, "SELECT * FROM users WHERE id=$id");

while($res = mysqli_fetch_array($result))
{
    $name = $res['name'];
    $age = $res['age'];
    $email = $res['email'];
}
?>
<html>
<head>
    <title>Edit Data</title>
</head>

<body>
    <a href="index.php">Home</a>
    <br/><br/>

    <form name="form1" method="post" action="edit.php">
        <table border="0">
            <tr>
                <td>Name</td>
                <td><input type="text" name="name" value="<?php echo $name;?>"></td>
            </tr>
            <tr>
                <td>Age</td>
                <td><input type="text" name="age" value="<?php echo $age;?>"></td>
            </tr>
            <tr>
                <td>Email</td>
                <td><input type="text" name="email" value="<?php echo $email;?>"></td>
            </tr>
            <tr>
                <td><input type="hidden" name="id" value="<?php echo $_GET['id'];?>"></td>
                <td><input type="submit" name="update" value="Update"></td>
            </tr>
        </table>
    </form>

```

```
</table>
</form>
</body>
</html>
```

Each row of data can be edited separately. Row ID is passed in the URL of edit.php. ID uniquely identifies the data entry.

Q3. a. Why user authentication is used in pages explain?

Answer:

A user authentication is a process in which you verify that someone who is attempting to access services and applications is who they claim to be. Authentication is used to verify that you are who you say you are. After a user's identity is confirmed, for instance with a username and password, that identity may be used in an authorization policy to determine the appropriate access privileges. In php code, it executes a SELECT query to check if a user is found in the database with the entered login credentials if a match is found, then the authentication process will be cleared by the user who attempts login. After successful login, the authenticated user will be allowed to enter into the application.

b. Write a full program related to user authentication page in PHP?

CODE FOR USER AUTHENTICATION:

The registered user can log in to the application via this login form. The login authentication is done by matching the login email and password with the registered user database. If a match found then the application will allow the user to proceed further. Otherwise, the login panel will acknowledge the user about the invalid attempt.

```
<form action="" method="POST" onsubmit="return loginvalidation();">
  <div class="row">
    <label>Email</label> <span id="email_error"></span>
  <div>
    <input type="text" name="email" id="email"
      class="form-control" placeholder="Enter your Email ID">
  </div>
</div>
<div class="row">
  <label>Password</label><span id="password_error"></span>
  <div>
    <input type="Password" name="password" id="password"
      class="form-control" placeholder="Enter your password">
  </div>
</div>
<div class="row">
  <div>
    <button type="submit" name="submit" class="btn login">Login</button>
  </div>
```

```

</div>
<div class="row">
  <div>
    <a href="index.php"><button type="button" name="submit"
      class="btn signup">Signup</button></a>
  </div>
</div>
</form>

```

The PHP code for validating the user login with the database is shown below. After a successful login, I redirect the user to the dashboard.php. The dashboard will show a welcome message by addressing the logged-in member. Also, there will be an option to log out of the current session.

```

<?php
session_start();
if (isset($_POST["submit"])) {
  include_once 'dbConnect.php';

  $email = $_POST['email'];
  $password = md5($_POST['password']);

  $database = new dbConnect();

  $db = $database->openConnection();

  $sql = "select * from tbl_registered_users where email = '$email' and password=
'$password'";
  $user = $db->query($sql);
  $result = $user->fetchAll(PDO::FETCH_ASSOC);

  $id = $result[0]['id'];
  $name = $result[0]['name'];
  $email = $result[0]['email'];
  $_SESSION['name'] = $name;
  $_SESSION['id'] = $id;

  $database->closeConnection();
  header('location: dashboard.php');
}
?>

```

Q4. a. Why logout is used in pages for users explain?

Answer:

The true purpose of the logout page is to let the member know that they are officially logged out. It's important to include some type of message letting them know that they were just logged out of the site. It helps the user to logout of the site safely. Log out is used for closing a running session. Logging out helps prevent other users from accessing the system without verifying their credentials. It also helps protect the current user's access or prevent unauthorized actions on the current login session and is thus an important part of security. Logging out ensures that user access and user credentials are safe after the login session.

b. Write a full program related to logout page in PHP as well as Html code?

CODE FOR LOGOUT PAGE AND SCRIPT:

Here's the code of our "welcome.php" file, where user is redirected after successful login:

```
<?php
// Initialize the session
session_start();
// Check if the user is logged in, if not then redirect him to login page
if(!isset($_SESSION["loggedin"]) || $_SESSION["loggedin"] !== true){
    header("location: login.php");
    exit;
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Welcome</title>
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.css">
    <style type="text/css">
        body{ font: 14px sans-serif; text-align: center; }
    </style>
</head>
<body>
    <div class="page-header">
        <h1>Hi, <b><?php echo htmlspecialchars($_SESSION["username"]); ?>
</b>. Welcome to our site.</h1>
    </div>
    <p>
        <a href="reset-password.php" class="btn btn-warning">Reset Your Password</a>
        <a href="logout.php" class="btn btn-danger">Sign Out of Your Account</a>
    </p>
</body>
```

Now, let's create a "logout.php" file. When the user clicks on the log out or sign out link, the script inside this file destroys the session and redirect the user back to the login page.

```
<?php
// Initialize the session
session_start();

// Unset all of the session variables
$_SESSION = array();

// Destroy the session.
session_destroy();

// Redirect to login page
header("location: login.php");
exit;
?>
```

Q5. a. Why use delete operation in pages explain?

Answer:

The delete command is used to remove data that is no longer required from a table. The "Where clause" is used to limit the number of rows affected by the DELETE query. Once data has been deleted, it cannot be recovered, it is therefore strongly recommend make backups before deleting data The DELETE query is often used along with "Select" statement to delete only those records that satisfy a specific condition.

b. Write a full program related to delete records from pages in PHP as well as html code?

CODE OF DELETE RECORD :

```
<?php
$servername= "localhost";
$username= "username";
$password= "password";
$dbname= "myDB";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
```



```

// sql to delete a record
$sql = "DELETE FROM MyGuests WHERE id=3";

if ($conn->query($sql) === TRUE) {
    echo "Record deleted successfully";
} else {
    echo "Error deleting record: " . $conn->error;
}

$conn->close();
?>

```

ANOTHER CODE TO DELETE RECORD:

```

<?php
// Process delete operation after confirmation
if(isset($_POST["id"]) && !empty($_POST["id"])){
    // Include config file
    require_once "config.php";
    // Prepare a delete statement
    $sql = "DELETE FROM employees WHERE id = ?";
    if($stmt = mysqli_prepare($link, $sql)){
        // Bind variables to the prepared statement as parameters
        mysqli_stmt_bind_param($stmt, "i", $param_id);
        // Set parameters
        $param_id = trim($_POST["id"]);
    // Attempt to execute the prepared statement
    if(mysqli_stmt_execute($stmt)){
        // Records deleted successfully. Redirect to landing page
        header("location: index.php");
        exit();

    } else{
        echo "Oops! Something went wrong. Please try again later.";
    }
}

// Close statement
mysqli_stmt_close($stmt);
// Close connection
mysqli_close($link);
} else{
    // Check existence of id parameter
    if(empty(trim($_GET["id"]))) {
        // URL doesn't contain id parameter. Redirect to error page
        header("location: error.php");
        exit();
    }
}

```

```

    }
  }
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>View Record</title>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.css">
  <style type="text/css">
    .wrapper{
      width: 500px;
      margin: 0 auto;
    }
  </style>
</head>
<body>
  <div class="wrapper">
    <div class="container-fluid">
      <div class="row">
        <div class="col-md-12">
          <div class="page-header">
            <h1>Delete Record</h1>
          </div>
          <form action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]); ?>" method="post">
            <div class="alert alert-danger fade in">
              <input type="hidden" name="id" value="<?php
echo trim($_GET["id"]); ?>" />
              <p>Are you sure you want to delete this record?</p><br>
              <p>
                <input type="submit" value="Yes"
class="btn btn-danger">
                <a href="index.php" class="btn btn-default">No</a>
              </p>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
</body>
</html>

```

+++++

