Practical – 6

Aim: Perform following programs of PHP file handling.

Exercise:

1. **Write a PHP program to create a student registration form with validation for fields including Enrollment No., Full Name, Gender, Mobile, Email, Address, and Profile Photo. Implement validation for Full Name, Mobile, Email, and Profile Photo inputs. Store all submitted data in a CSV file.**

```html
<!DOCTYPE html>
<html>
<head>
  <title>Student Registration Form</title>
  <style>
    body {
        font-family: Arial, sans-serif;
        background-color: #f0f0f0;
        margin: 0;
        padding: 0;
        display: flex;
        justify-content: center;
        align-items: center;
        height: 100vh;
    }
    #container {
        background-color: #fff;
        padding: 20px;
        border-radius: 10px;
        box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
        width: 400px;
    }
    h2 {
        text-align: center;
    }
    input[type="text"], select, input[type="file"] {
        width: 100%;
        padding: 10px;
        margin: 8px 0;
        box-sizing: border-box;
        border: 1px solid #ccc;
        border-radius: 4px;
```

```
        }
      input[type="submit"] {
         background-color: #4CAF50;
         color: white;
         padding: 14px 20px;
         margin: 8px 0;
         border: none;
         border-radius: 4px;
         cursor: pointer;
         width: 100%;
      }
      input[type="submit"]:hover {
         background-color: #45a049;
      }
      .error {
         color: red;
         font-size: 12px;
      }
   </style>
</head>
<body>
   <div id="container">
      <h2>Student Registration Form</h2>
      <form method="post" enctype="multipart/form-data">
         <input type="text" name="enrollment_no" placeholder="Enrollment No."
required><br>
         <input type="text" name="full_name" placeholder="Full Name" required><br>
         <select name="gender" required>
            <option value="">Select Gender</option>
            <option value="male">Male</option>
            <option value="female">Female</option>
         </select><br>
         <input type="text" name="mobile" placeholder="Mobile" required><br>
         <input type="text" name="email" placeholder="Email" required><br>
         <input type="text" name="address" placeholder="Address"><br>
         <input type="file" name="profile_photo" accept="image/*" required><br>
         <input type="submit" name="submit" value="Register">
      </form>

      <?php
      if ($_SERVER["REQUEST_METHOD"] == "POST") {
         // Validate and store data
```

```php
        $errors = array();

        $enrollment_no = $_POST['enrollment_no'];
        $full_name = $_POST['full_name'];
        $gender = $_POST['gender'];
        $mobile = $_POST['mobile'];
        $email = $_POST['email'];
        $address = $_POST['address'];
        $profile_photo = $_FILES['profile_photo']['name'];
        $tmp_profile_photo = $_FILES['profile_photo']['tmp_name'];

        // Validate full name
        if (!preg_match("/^[a-zA-Z\s]+$/", $full_name)) {
          $errors['full_name'] = "Invalid full name";
        }

        // Validate mobile
        if (!preg_match("/^[0-9]{10}$/", $mobile)) {
          $errors['mobile'] = "Invalid mobile number";
        }

        // Validate email
        if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
          $errors['email'] = "Invalid email address";
        }

        // Validate profile photo
        $allowed_extensions = array('jpg', 'jpeg', 'png', 'gif');
        $file_extension = strtolower(pathinfo($profile_photo, PATHINFO_EXTENSION));
        if (!in_array($file_extension, $allowed_extensions)) {
          $errors['profile_photo'] = "Invalid file format. Allowed formats: jpg, jpeg, png,
gif";
        }

        // If no errors, store data in CSV file
        if (empty($errors)) {
          $csv_data = array($enrollment_no, $full_name, $gender, $mobile, $email,
$address, $profile_photo);
          $file = fopen("student_data.csv", "a");
          fputcsv($file, $csv_data);
          fclose($file);
          move_uploaded_file($tmp_profile_photo, "profile_photos/$profile_photo");
```

```
            echo "<p>Registration successful!</p>";
        } else {
            // Display validation errors
            echo "<div class='error'>";
            foreach ($errors as $error) {
                echo "<p>$error</p>";
            }
            echo "</div>";
        }
    }
    ?>
  </div>
</body>
</html>
```

Student Registration Form

21012021001

ADESHARA BRIJESH GAUTAMKUMAR

Male

9510667012

adesharabrijesh8@gmail.com

114 , astvinayak society

Choose File | Adeshara Brijesh Gautamkumar.png

Register

Registration successful!

Student Registration Form

21012021001

ADESHARA BRIJESH

Male

95106679120112

mrhashtag292@gmail.com

114 , astvinayak society

Choose File | No file chosen

Register

Invalid mobile number

	A	B	C	D	E	F	G
1	21012021001	ADESHARA BRIJESH	male	9510667912	mrhashtag292@gmail.com	114 , astvinayak society	Adeshara Brijesh Gautamkumar.png
2							

2. Create a form with a file input field to allow users to upload files. Implement server-side PHP code to handle the uploaded file:

a. Validate the file type and size to prevent malicious uploads and ensure server resource management.

b. Move the uploaded file to a secure location on the server.

c. Provide feedback to the user on the success or failure of the upload process.

```
<!DOCTYPE html>
<html>
<head>
  <title>File Upload Form</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f0f0f0;
      margin: 0;
      padding: 0;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }
    #container {
      background-color: #fff;
      padding: 20px;
      border-radius: 10px;
      box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
      width: 400px;
    }
    h2 {
      text-align: center;
    }
    input[type="file"] {
      width: 100%;
      padding: 10px;
      margin: 8px 0;
      box-sizing: border-box;
      border: 1px solid #ccc;
      border-radius: 4px;
    }
```

```css
      input[type="submit"] {
        background-color: #4CAF50;
        color: white;
        padding: 14px 20px;
        margin: 8px 0;
        border: none;
        border-radius: 4px;
        cursor: pointer;
        width: 100%;
      }
      input[type="submit"]:hover {
        background-color: #45a049;
      }
      .error {
        color: red;
        font-size: 12px;
      }
      .success {
        color: green;
        font-size: 14px;
        font-weight: bold;
        text-align: center;
      }
   </style>
</head>
<body>
   <div id="container">
      <h2>File Upload Form</h2>
      <form method="post" enctype="multipart/form-data">
        <input type="file" name="uploaded_file" required><br>
        <input type="submit" name="submit" value="Upload File">
      </form>

      <?php
      if ($_SERVER["REQUEST_METHOD"] == "POST") {
        // Check if file is uploaded successfully
        if (isset($_FILES["uploaded_file"]) && $_FILES["uploaded_file"]["error"] ==
UPLOAD_ERR_OK) {
            $uploaded_file_name = $_FILES["uploaded_file"]["name"];
            $uploaded_file_tmp = $_FILES["uploaded_file"]["tmp_name"];
            $uploaded_file_size = $_FILES["uploaded_file"]["size"];
```
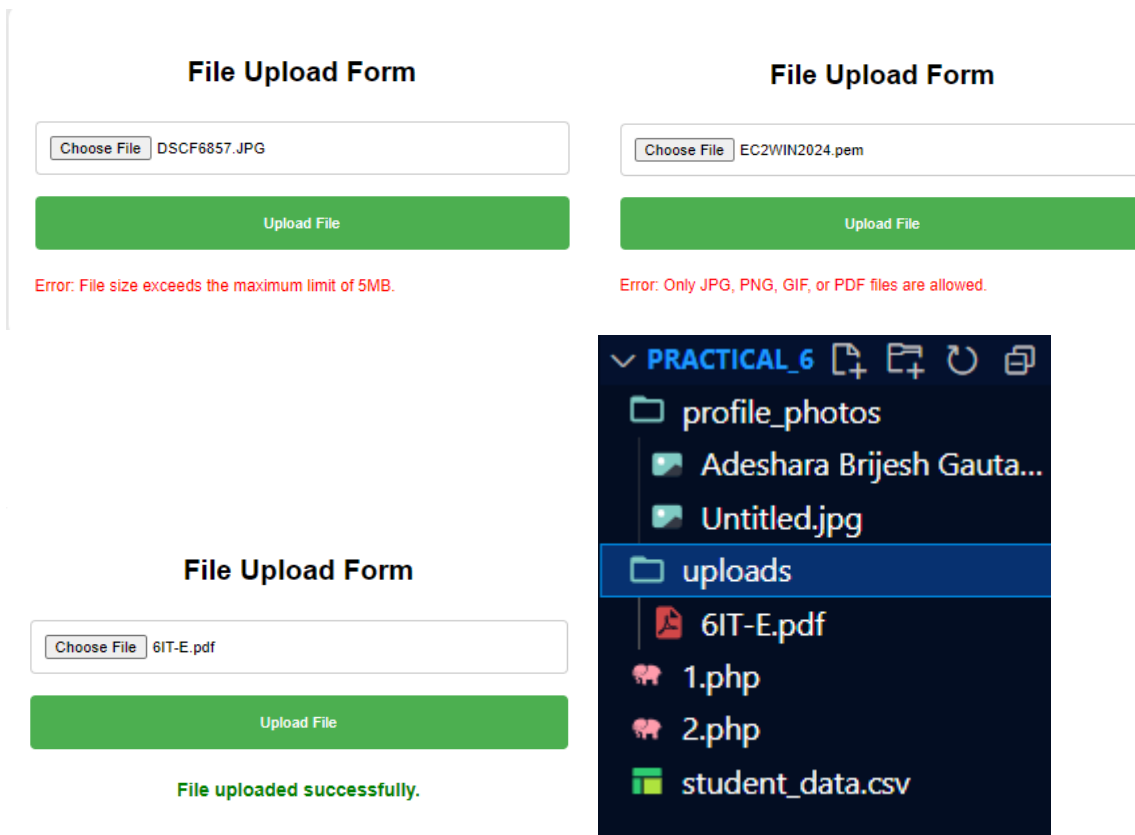
```php
        $uploaded_file_type = $_FILES["uploaded_file"]["type"];

        // Validate file type
        $allowed_types = array("image/jpeg", "image/png", "image/gif",
"application/pdf");
        if (!in_array($uploaded_file_type, $allowed_types)) {
            echo "<p class='error'>Error: Only JPG, PNG, GIF, or PDF files are
allowed.</p>";
            exit;
        }

        // Validate file size (limit to 5MB)
        $max_file_size = 5 * 1024 * 1024; // 5MB in bytes
        if ($uploaded_file_size > $max_file_size) {
            echo "<p class='error'>Error: File size exceeds the maximum limit of
5MB.</p>";
            exit;
        }

        // Move the uploaded file to a secure location
        $upload_dir = "uploads/";
        $destination_file = $upload_dir . basename($uploaded_file_name);
        if (move_uploaded_file($uploaded_file_tmp, $destination_file)) {
            echo "<p class='success'>File uploaded successfully.</p>";
        } else {
            echo "<p class='error'>Error: Failed to move the uploaded file.</p>";
        }
    } else {
        echo "<p class='error'>Error: No file uploaded or an error occurred during
upload.</p>";
    }
}
?>
  </div>
</body>
</html>
```

File Upload Form

Choose File | DSCF6857.JPG

Upload File

Error: File size exceeds the maximum limit of 5MB.

File Upload Form

Choose File | EC2WIN2024.pem

Upload File

Error: Only JPG, PNG, GIF, or PDF files are allowed.

File Upload Form

Choose File | 6IT-E.pdf

Upload File

File uploaded successfully.

∨ PRACTICAL_6
  ▢ profile_photos
    Adeshara Brijesh Gauta...
    Untitled.jpg
  ▢ uploads
    6IT-E.pdf
  1.php
  2.php
  student_data.csv

3. **Develop functionality to manage uploaded files:**
   a. **List uploaded files with options for viewing, downloading, or deleting them.**
   Manage_files.php

```
<!DOCTYPE html>
<html>
<head>
  <title>Manage Uploaded Files</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f0f0f0;
      margin: 0;
      padding: 20px;
    }
    h2 {
      text-align: center;
    }
    table {
```

```
          width: 100%;
          border-collapse: collapse;
          margin-bottom: 20px;
       }
       th, td {
          border: 1px solid #ddd;
          padding: 8px;
          text-align: left;
       }
       th {
          background-color: #f2f2f2;
       }
       .actions {
          display: flex;
          justify-content: center;
       }
       .actions a {
          margin-right: 10px;
       }
    </style>
</head>
<body>
    <h2>Uploaded Files</h2>
    <?php
    // Directory where uploaded files are stored
    $upload_dir = "uploads/";

    // Check if the directory exists
    if (is_dir($upload_dir)) {
       // Open directory
       if ($dh = opendir($upload_dir)) {
          // Display table header
          echo "<table>";
          echo "<tr><th>Filename</th><th>Size</th><th>Actions</th></tr>";

          // Read files from the directory
          while (($file = readdir($dh)) !== false) {
             // Skip directories and hidden files
             if ($file != "." && $file != "..") {
                // Get file size
                $file_size = filesize($upload_dir . $file);
```

```php
              // Format file size
              $size_units = array('B', 'KB', 'MB', 'GB', 'TB');
              $size_index = 0;
              while ($file_size >= 1024 && $size_index < count($size_units) - 1) {
                 $file_size /= 1024;
                 $size_index++;
              }
              $formatted_size = round($file_size, 2) . " " . $size_units[$size_index];

              // Display file details in table rows
              echo "<tr>";
              echo "<td>$file</td>";
              echo "<td>$formatted_size</td>";
              echo "<td class='actions'><a href='uploads/$file' target='_blank'>View</a> <a
href='download.php?file=$file'>Download</a> <a
href='delete.php?file=$file'>Delete</a></td>";
              echo "</tr>";
            }
          }

          // Close directory and table
          closedir($dh);
          echo "</table>";
       } else {
          echo "<p>Error: Unable to open directory.</p>";
       }
    } else {
       echo "<p>Error: Upload directory does not exist.</p>";
    }
    ?>
</body>
</html>
```

Download.php

```php
<?php
// File to be downloaded
$file = $_GET['file'];

// Path to the file
$file_path = "uploads/" . $file;
```

```php
// Check if the file exists
if (file_exists($file_path)) {
   // Set headers to force download
   header("Content-type: application/octet-stream");
   header("Content-Disposition: attachment; filename=\"$file\"");

   // Read the file and output its contents
   readfile($file_path);
   exit;
} else {
   echo "Error: File not found.";
}
?>
```

Delete.php

```php
<?php
$file = $_GET['file'];
$file_path = "uploads/" . $file;
if (file_exists($file_path)) {
   if (unlink($file_path)) {
      echo "File deleted successfully.";
   } else {
      echo "Error: Failed to delete file.";
   }
} else {
   echo "Error: File not found."; }
?>
```