

# CODES OF COMPUTER SCIENCE DEPARTMENT PORTAL

## BOOKS\_MANAGEMENT SYSTEM:

View\_books.php

```
<?php
include("../config/db_connect3.php");
session_start();

// Get semester from GET parameter or default
$semester = isset($_GET['semester']) ? intval($_GET['semester']) : 1;

// Fetch books for selected semester and that are not deleted
$query = "SELECT * FROM books WHERE semester = ? AND is_deleted = 0";
$stmt = $conn->prepare($query);
$stmt->bind_param("i", $semester);
$stmt->execute();
$result = $stmt->get_result();
//include header
include("../headnfoot/header.php");
?>

<!DOCTYPE html>
<html>
<head>
    <title>View Books</title>
    <link rel="stylesheet" href="view_books.css">
    <!-- <link rel="stylesheet" href="../homepage/css/styles.css"> -->
</head>
<body>

<h2>Books for Semester <?= $semester ?></h2>

<div class="semester-selector">
    <form method="GET" action="">
        <label for="semester">Select Semester:</label>
        <select name="semester" id="semester" onchange="this.form.submit()">
            <?php for ($i = 1; $i <= 6; $i++): ?>
                <option value="<?= $i ?>" <?= ($i == $semester) ? 'selected' :
'' ?>>Semester <?= $i ?></option>
            <?php endfor; ?>
        </select>
```

```

        </form>

</div>

<div class="books-container">
    <?php while ($book = $result->fetch_assoc()): ?>
        <div class="book-card">
            <?php
                $imagePath = "../" . $book['cover_image'];
                $filePath = isset($book['file']) ? "../" . $book['file'] :
null;

                // Display image
                if (!empty($book['cover_image']) && file_exists($imagePath)) {
                    echo "<img src='$imagePath' alt='Book Cover'>";
                } else {
                    echo "<img src='../uploads/placeholder.jpg' alt='No
Image'>";
                }

                $file_path = isset($book['file_path']) ? "../" .
$book['file_path'] : null;

                // Display book and author name
                echo "<h3>" . htmlspecialchars($book['title']) . "</h3>";
                echo "<p><strong>Author:</strong> " .
htmlspecialchars($book['author']) . "</p>";

                // Display download button
                if (!empty($book['file_path']) && file_exists($file_path)) {
                    echo "<a class='download-btn' href='$file_path'
download>Download Book</a>";
                } else {
                    echo "<p style='color:red'>Book file not available</p>";
                }

            ?>
        </div>
    <?php endwhile; ?>
</div>

</body>
</html>

```

## Upload\_books.php

```
<?php
    // NOTE: Copied this file from books folder to dashboard folder
include("../config/db_connect3.php");
session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $title = $_POST['title'];
    $author = $_POST['author'];
    $semester = $_POST['semester'];
    $subject = $_POST['subject'];

    // Handle Book PDF Upload
    $file_name = $_FILES['book_file']['name'];
    $file_tmp = $_FILES['book_file']['tmp_name'];
    $file_size = $_FILES['book_file']['size']; // Get file size
    $upload_dir = "../uploads/books/";
    $relative_file_path = "uploads/books/" . basename($file_name); // Fixed
path for browser download
    $file_path = $upload_dir . basename($file_name); // Full
path for moving file

    // Handle Book Cover Image Upload
    $cover_name = $_FILES['cover_image']['name'];
    $cover_tmp = $_FILES['cover_image']['tmp_name'];
    $cover_size = $_FILES['cover_image']['size']; // Get cover image size
    $cover_upload_dir = "../uploads/covers/";
    $relative_cover_path = "uploads/covers/" . basename($cover_name); // Match
for frontend
    $cover_path = $cover_upload_dir . basename($cover_name); //
Actual path

    // Validation for file types and sizes (Server-Side)
    $allowed_pdf_types = ['application/pdf'];
    $allowed_image_types = ['image/jpeg', 'image/png'];
    $max_file_size = 10 * 1024 * 1024; // Max size for PDF (10MB)
    $max_cover_size = 5 * 1024 * 1024; // Max size for Cover Image (5MB)

    // Check if the book file is PDF and within size limit
    if (!in_array($_FILES['book_file']['type'], $allowed_pdf_types)) {
        echo "<script>alert('Only PDF files are allowed for the
book.');
```

```

        // Check if the cover image is JPG/PNG and within size limit
        elseif (!in_array($_FILES['cover_image']['type'], $allowed_image_types)) {
            echo "<script>alert('Only JPG and PNG images are allowed for the
cover.');

```

```
</head>
<body>
<div class="main-container">
<!-- HTML Form for Uploading Books with Cover Image -->
<form method="POST" enctype="multipart/form-data">
  <h2>Upload Books</h2>

  <label>Title:</label>
  <input type="text" name="title" required>

  <label>Author:</label>
  <input type="text" name="author" required>

  <label>Semester:</label>
  <select name="semester">
    <option value="1">1st Semester</option>
    <option value="2">2nd Semester</option>
    <option value="3">3rd Semester</option>
    <option value="4">4th Semester</option>
    <option value="5">5th Semester</option>
    <option value="6">6th Semester</option>
  </select>

  <label>Subject:</label>
  <input type="text" name="subject" required>

  <label>Upload Book (PDF only, max 10MB):</label>
  <input type="file" name="book_file" accept=".pdf" required>

  <label>Upload Cover Image (JPG/PNG only, max 5MB):</label>
  <input type="file" name="cover_image" accept="image/*" required>

  <button type="submit">Upload Book</button>
</form>
</div>
</body>
</html>
```

## Delete\_books.php

```
<?php
// Inside delete_book.php
include(__DIR__ . "../config/db_connect3.php");

session_start();

if (isset($_GET['id'])) {
    $book_id = $_GET['id'];

    // Soft delete: Set is_deleted = 1
    $sql = "UPDATE books SET is_deleted = 1 WHERE id = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("i", $book_id);

    if ($stmt->execute()) {
        echo "<script>alert('Book moved to trash successfully!');
window.location.href='manage_books.php';</script>";
    } else {
        echo "<script>alert('Failed to delete book.');"
window.history.back();</script>";
    }

    $stmt->close();
    $conn->close();
} else {
    echo "<script>alert('Invalid request.');" window.history.back();</script>";
}
?>
```

## Restore\_books.php

```
<?php
include(__DIR__ . '/../config/db_connect3.php');

session_start();

if (isset($_GET['id'])) {
    $book_id = $_GET['id'];

    // Restore the book by setting is_deleted = 0
    $sql = "UPDATE books SET is_deleted = 0 WHERE id = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("i", $book_id);

    if ($stmt->execute()) {
        echo "<script>alert('Book restored successfully!');
window.location.href='manage_books.php';</script>";
    } else {
        echo "<script>alert('Failed to restore book.');"
window.history.back();</script>";
    }

    $stmt->close();
    $conn->close();
} else {
    echo "<script>alert('Invalid request.');" window.history.back();</script>";
}
?>
```

## Permanent\_delete\_book.php

```
<?php
include(__DIR__ . "../config/db_connect3.php");

session_start();

if (isset($_GET['id'])) {
    $book_id = $_GET['id'];

    // Delete the book permanently
    $sql = "DELETE FROM books WHERE id = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("i", $book_id);

    if ($stmt->execute()) {
        echo "<script>alert('Book permanently deleted!');
window.location.href='manage_books.php';</script>";
    } else {
        echo "<script>alert('Failed to delete book permanently. ');
window.history.back();</script>";
    }

    $stmt->close();
    $conn->close();
} else {
    echo "<script>alert('Invalid request. '); window.history.back();</script>";
}
?>
```



## Edit\_book.php

```
<?php
include("../config/db_connect3.php");
session_start();

if (!isset($_GET['id'])) {
    echo "Invalid request.";
    exit;
}

$book_id = $_GET['id'];

// Fetch book details
$stmt = $conn->prepare("SELECT * FROM books WHERE id = ?");
$stmt->bind_param("i", $book_id);
$stmt->execute();
$result = $stmt->get_result();
$book = $result->fetch_assoc();

if (!$book) {
    echo "Book not found.";
    exit;
}

// Handle form submission
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $title = $_POST['title'];
    $author = $_POST['author'];
    $semester = $_POST['semester'];
    $cover_image = $book['cover_image']; // Keep old image by default

    // Handle image upload
    if (!empty($_FILES['image']['name'])) {
        $upload_dir = "../uploads/";
        $new_image_name = basename($_FILES['image']['name']);
        $target_path = $upload_dir . $new_image_name;

        if (move_uploaded_file($_FILES['image']['tmp_name'], $target_path)) {
            // Save relative path for database
            $cover_image = "uploads/" . basename($new_image_name);
        } else {
            echo "<p style='color:red;'>Failed to upload image.</p>";
        }
    }
}
```

```

    // Update book
    $stmt = $conn->prepare("UPDATE books SET title = ?, author = ?, semester =
?, cover_image = ? WHERE id = ?");
    $stmt->bind_param("ssisi", $title, $author, $semester, $cover_image,
$book_id);
    $stmt->execute();

    header("Location: manage_books.php?success=updated");
    exit;
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>Edit Book</title>
    <style>
        * {
            margin: 0;
            padding: 0;
            box-sizing: border-box;
        }

        body {
            font-family: 'Segoe UI', sans-serif;
            background: #0e1117;
            color: #e0e0e0;
            min-height: 100vh;
            display: flex;
            flex-direction: column;
            align-items: center;
            padding: 40px 20px;
        }

        h2 {
            margin-bottom: 20px;
            font-size: 26px;
            color: #c2e9fb;
            text-shadow: 0 0 5px rgba(0,255,255,0.15);
        }

        form {
            background: rgba(255, 255, 255, 0.05);
            backdrop-filter: blur(12px);
            border-radius: 10px;
            padding: 30px;
            width: 100%;
            max-width: 500px;

```

```
        box-shadow: 0 0 20px rgba(0, 255, 255, 0.05);
    }

    label {
        display: block;
        margin: 15px 0 6px;
        font-weight: 500;
        color: #a0c4ff;
    }

    input[type="text"],
    input[type="number"],
    input[type="file"] {
        width: 100%;
        padding: 10px;
        background-color: #1e1e2f;
        border: none;
        border-radius: 6px;
        color: #e0e0e0;
        font-size: 14px;
        margin-bottom: 15px;
        outline: none;
    }

    img.preview {
        max-width: 100%;
        margin: 10px 0 20px;
        border: 1px solid #333;
        border-radius: 8px;
    }

    button {
        background: #1fb6ff;
        border: none;
        padding: 12px 20px;
        color: white;
        border-radius: 6px;
        cursor: pointer;
        font-weight: 600;
        transition: background 0.3s ease, transform 0.2s ease;
        display: inline-block;
    }

    button:hover {
        background: #009ee0;
        transform: translateY(-2px);
    }
}
</style>
```

```
</head>

<body>

    <h2>Edit Book Details</h2>

    <form method="POST" enctype="multipart/form-data">
        <label>Title</label>
        <input type="text" name="title" value="<?=  
htmlspecialchars($book['title']) ?>" required>

        <label>Author</label>
        <input type="text" name="author" value="<?=  
htmlspecialchars($book['author']) ?>" required>

        <label>Semester</label>
        <input type="number" name="semester" value="<?=  
$book['semester'] ?>" required min="1" max="6">

        <label>Current Image</label><br>
        

        <label>Upload New Image (optional)</label>
        <input type="file" name="image">

        <button type="submit">Update Book</button>
    </form>

</body>
</html>
```

## Manage\_books.php

```
<?php

include("../config/db_connect3.php");
session_start();
include("../headnfoot/header.php");

// Handle Delete (Soft Delete)
if (isset($_GET['delete'])) {
    $book_id = $_GET['delete'];
    $sql = "UPDATE books SET is_deleted = 1 WHERE id = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("i", $book_id);
    $stmt->execute();
}

// Handle Restore
if (isset($_GET['restore'])) {
    $book_id = $_GET['restore'];
    $sql = "UPDATE books SET is_deleted = 0 WHERE id = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("i", $book_id);
    $stmt->execute();
}

// Handle Permanent Delete
if (isset($_GET['permanent_delete'])) {
    $book_id = $_GET['permanent_delete'];
    $sql = "DELETE FROM books WHERE id = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("i", $book_id);
    $stmt->execute();
}

// Fetch Books
$sql = "SELECT * FROM books";
$result = $conn->query($sql);
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Manage Books</title>
    <link rel="stylesheet" href="manage_books.css">

```

```

</head>
<body>
    <h1>Manage Books</h1>
    <table border="1">
        <tr>
            <th>Title</th>
            <th>Author</th>
            <th>Semester</th>
            <th>Actions</th>
        </tr>
        <?php while ($row = $result->fetch_assoc()): ?>
            <tr>
                <td><?php echo htmlspecialchars($row['title']); ?></td>
                <td><?php echo htmlspecialchars($row['author']); ?></td>
                <td><?php echo htmlspecialchars($row['semester']); ?></td>
                <td>
                    <?php if ($row['is_deleted'] == 0): ?>
                        <a class="action-btn delete-btn"
href="delete_book.php?id=<?= $row['id']; ?>" onclick="return confirm('Are you
sure you want to delete this book?');">
                            Delete Book
                        </a>
                        <?php else: ?>
                            <a class="action-btn restore-btn"
href="restore_book.php?id=<?= $row['id']; ?>" onclick="return confirm('Restore
this book?');">
                                Restore Book
                            </a>
                        <?php endif; ?>
                    <br>
                        <a class="action-btn permdelete-btn"
href="permanent_delete_book.php?id=<?= $row['id']; ?>" onclick="return
confirm('This action is irreversible! Do you want to permanently delete this
book?');">
                            Permanent Delete
                        </a>
                        <a href="edit_book.php?id=<?= $row['id']; ?>" class="action-
btn edit-btn">
                            Edit Book
                        </a>
                    </td>
                </tr>
            <?php endwhile; ?>
        </table>
    </body>
</html>

```

## NOTICE MANAGEMENT SYSTEM

View\_notice.php

```
<?php
include("../config/db_connect.php");
session_start();

// ☉ Handle Date Filtering
$dateFilter = isset($_GET['filter_date']) ? $_GET['filter_date'] : '';
$query = "SELECT * FROM notices WHERE status = 'active'";

if ($dateFilter) {
    $query .= " AND DATE(uploaded_at) = '$dateFilter'";
}

$query .= " ORDER BY uploaded_at DESC";

// ✅ Run query and handle failure
$result = mysqli_query($conn, $query);
if (!$result) {
    die("❌ Query Failed: " . mysqli_error($conn));
}

//including the header
include("../headnfoot/header.php");
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>📄 View Notices</title>
    <link rel="stylesheet" href="notice.css">
    <script defer src="noticesss.js"></script>
</head>
<body>

<h1>📄 Notices</h1>

<!-- 📄 Filter Toggle Button -->
<button id="filter-toggle">📄 Show Filters</button>

<!-- 📄 Hidden Filter Section -->
<div class="filter-section" id="filter-section" style="display: none;">
```

```

    <form method="GET">
        <label for="filter_date">Filter by Date:</label>
        <input type="date" name="filter_date" value="<?php echo
htmlspecialchars($dateFilter); ?>">
        <button type="submit">Apply Filter</button>
    </form>
</div>

<!-- 📄 Notice Table -->
<table>
    <tr>
        <th>Title</th>
        <th>Date</th>
        <th>Size</th>
        <th>Download</th>
        <?php if (isset($_SESSION['role']) && $_SESSION['role'] === 'teacher')
{ ?>
            <th>Actions</th>
            <?php } ?>
    </tr>
    <?php while ($row = mysqli_fetch_assoc($result)) { ?>
    <tr>
        <td><?php echo htmlspecialchars($row['title']); ?></td>
        <td><?php echo date('d M Y', strtotime($row['uploaded_at'])); ?></td>
        <td><?php echo round($row['file_size'] / 1024, 2) . " KB"; ?></td>
        <td><a href=".."><?php echo htmlspecialchars($row['file_path']); ?>"
download>📄 Download</a></td>
        <?php if (isset($_SESSION['role']) && $_SESSION['role'] === 'teacher')
{ ?>
            <td class="actions">
                <a href="delete_notice.php?id=<?php echo (int)$row['id']; ?>"
class="delete">🗑 Delete</a>
            </td>
            <?php } ?>
        </tr>
    <?php } ?>
</table>

<!-- 🛠 Manage Buttons for Teachers -->

<!-- 📝 JS for Filter Toggle -->
<script>
document.getElementById('filter-toggle').addEventListener('click', function ()
{
    const section = document.getElementById('filter-section');

```



```
    section.style.display = section.style.display === 'none' ||
section.style.display === '' ? 'flex' : 'none';
});
</script>

</body>
</html>
```

## Edit\_notice.php

```
<?php
include("../config/db_connect.php");
session_start();

// Check if ID is provided
if (!isset($_GET['id'])) {
    die("Invalid request! No ID provided.");
}

$id = (int)$_GET['id'];

// Fetch existing notice
$stmt = $conn->prepare("SELECT * FROM notices WHERE id = ?");
$stmt->bind_param("i", $id);
$stmt->execute();
$result = $stmt->get_result();
$notice = $result->fetch_assoc();
$stmt->close();

if (!$notice) {
    die("Notice not found!");
}

// Handle form submission
if ($_SERVER["REQUEST_METHOD"] === "POST") {
    $title = trim($_POST['title']);
    $file_path = $notice['file_path']; // Keep current file by default

    // Handle new file upload
    if (isset($_FILES['notice_file']) && $_FILES['notice_file']['error'] ===
0) {
        $file_name = time() . '_' . basename($_FILES['notice_file']['name']);
        $target_dir = "../notices/notice_files/";

        if (!is_dir($target_dir)) {
            mkdir($target_dir, 0777, true);
        }

        $target_file = $target_dir . $file_name;
        $file_ext = strtolower(pathinfo($target_file, PATHINFO_EXTENSION));

        if ($file_ext === "pdf") {
            // Delete old file

```

```

        if ($file_path && file_exists("../" . $file_path)) {
            unlink("../" . $file_path);
        }

        // Move uploaded file
        if (move_uploaded_file($_FILES['notice_file']['tmp_name'],
$target_file)) {
            $file_path = "notices/notice_files/" . $file_name;
        } else {
            echo "<p class='error'>Error uploading the new file.</p>";
        }
    } else {
        echo "<p class='error'>Only PDF files are allowed.</p>";
    }
}

// Update notice in the database
$stmt = $conn->prepare("UPDATE notices SET title = ?, file_path = ? WHERE
id = ?");
$stmt->bind_param("ssi", $title, $file_path, $id);

if ($stmt->execute()) {
    echo "<script>alert('Notice updated successfully!');
window.location.href='manage_notices.php';</script>";
    exit;
} else {
    echo "<p class='error'>Error updating notice: " . $stmt->error .
"</p>";
}

$stmt->close();
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Edit Notice</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background: #f9f9f9;
            padding: 40px;
        }
        .container {
            background: #fff;
            padding: 25px;

```

```

        max-width: 600px;
        margin: auto;
        border-radius: 8px;
        box-shadow: 0 4px 8px rgba(0,0,0,0.1);
    }
    h1 {
        text-align: center;
    }
    form label {
        display: block;
        margin-top: 15px;
        margin-bottom: 5px;
    }
    form input[type="text"],
    form input[type="file"],
    form button {
        width: 100%;
        padding: 10px;
        margin-top: 5px;
    }
    form button {
        background: #007bff;
        color: white;
        border: none;
        cursor: pointer;
    }
    form button:hover {
        background: #0056b3;
    }
    a {
        display: block;
        margin-top: 20px;
        text-align: center;
        text-decoration: none;
        color: #007bff;
    }
    a:hover {
        text-decoration: underline;
    }
    .error { color: red; }
    .success { color: green; }
</style>
</head>
<body>

<div class="container">
    <h1>✎ Edit Notice</h1>
    <form method="POST" enctype="multipart/form-data">

```

```

        <label for="title">Notice Title:</label>
        <input type="text" name="title" value="<?php echo
htmlspecialchars($notice['title']); ?>" required>

        <label for="notice_file">Replace PDF (optional):</label>
        <input type="file" name="notice_file" accept=".pdf">

        <?php if ($notice['file_path']): ?>
            <p>📄 Current File: <a href=".."<?php echo $notice['file_path'];
?>" target="_blank">View PDF</a></p>
            <?php else: ?>
                <p>No file uploaded.</p>
            <?php endif; ?>

        <button type="submit">Update Notice</button>
    </form>

    <a href="manage_notices.php">⬅️ Back to Manage Notices</a>
</div>

</body>
</html>

```

Delete\_notice.php

```

<?php
include("../config/db_connect.php");
session_start();

// Ⓣ Only teachers can delete
if (!isset($_SESSION['role']) || $_SESSION['role'] !== 'teacher') {
    die("Access Denied! Ⓣ Only teachers can delete notices.");
}

if (isset($_GET['id'])) {
    $id = $_GET['id'];
    $query = "UPDATE notices SET status = 'deleted' WHERE id = ?";
    $stmt = mysqli_prepare($conn, $query);
    mysqli_stmt_bind_param($stmt, "i", $id);
    mysqli_stmt_execute($stmt);
}

header("Location: view_notices.php?msg=Notice moved to trash.");
exit();

```

add\_notice.php

```
<?php
include("../config/db_connect.php");
session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $title = $_POST['title'];
    $file_path = NULL;

    // Ensure upload directory exists
    $target_dir = "../notices/notice_files/";
    if (!is_dir($target_dir)) {
        mkdir($target_dir, 0777, true);
    }

    // Handle file upload
    if (isset($_FILES['notice_file']) && $_FILES['notice_file']['error'] ===
0) {
        $file_name = time() . "_" .
basename($_FILES['notice_file']['name']);
        $target_file = $target_dir . $file_name;
        $file_size = $_FILES['notice_file']['size'];

        if (move_uploaded_file($_FILES['notice_file']['tmp_name'],
$target_file)) {
            $file_path = "notices/notice_files/" . $file_name;
        } else {
            die("Error: File upload failed.");
        }
    }

    // Insert into database
    $query = "INSERT INTO notices (title, file_path, file_size) VALUES (?, ?,
?)";
    $stmt = mysqli_prepare($conn, $query);
    mysqli_stmt_bind_param($stmt, "ssi", $title, $file_path, $file_size);

    if (mysqli_stmt_execute($stmt)) {
        echo "<script>alert('Notice added successfully!');
window.location.href='view_notices.php';</script>";
    } else {
        echo "Error: " . mysqli_error($conn);
    }
}
```

```
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Upload Notice</title>
    <link rel="stylesheet" href="add_notice.css">
</head>
<body>

<h1>Upload Notice</h1>

<form action="add_notice.php" method="POST" enctype="multipart/form-data"
onsubmit="return confirm('Are you sure you want to add this notice?');">
    <label for="title">Notice Title:</label>
    <input type="text" name="title" required>

    <label for="notice_file">Upload PDF:</label>
    <input type="file" name="notice_file" accept=".pdf" required>

    <button type="submit">Upload Notice</button>
</form>

<a href="view_notices.php"><img alt="back arrow icon" data-bbox="385 495 405 510"/> View Notices</a>

</body>
</html>
```

## Manage\_notices.php

```
<?php
session_start();
require '../config/db_connect.php';

// Handle form submissions
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if (isset($_POST['upload_notice'])) {
        $title = trim($_POST['title']);
        $file_path = NULL;
        $file_size = 0;

        $target_dir = "../notices/notice_files/";
        if (!is_dir($target_dir)) {
            mkdir($target_dir, 0777, true);
        }

        if (isset($_FILES['file']) && $_FILES['file']['error'] === 0) {
            $file_name = time() . "_" . basename($_FILES['file']['name']);
            $target_file = $target_dir . $file_name;
            $file_size = $_FILES['file']['size'];

            if (move_uploaded_file($_FILES['file']['tmp_name'], $target_file))
            {
                $file_path = "notices/notice_files/" . $file_name;
            } else {
                die("Error uploading file.");
            }
        }

        $query = "INSERT INTO notices (title, file_path, file_size) VALUES (?, ?, ?)";
        $stmt = $conn->prepare($query);
        $stmt->bind_param("ssi", $title, $file_path, $file_size);
        $stmt->execute();
        $stmt->close();

        header("Location: manage_notices.php");
        exit;
    }

    if (isset($_POST['delete_notice'])) {
```



```

        $id = $_POST['id'];
        $res = $conn->query("SELECT file_path FROM notices WHERE id = $id");
        if ($res && $res->num_rows > 0) {
            $file = $res->fetch_assoc()['file_path'];
            if (file_exists("../" . $file)) {
                unlink("../" . $file);
            }
        }
        $conn->query("DELETE FROM notices WHERE id = $id");

        header("Location: manage_notices.php");
        exit;
    }
}

$result = $conn->query("SELECT * FROM notices ORDER BY id DESC");
include("../headnfoot/header.php");
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Manage Notices</title>
    <style>
        body {
            font-family: 'Segoe UI', sans-serif;
            background: linear-gradient(to right, #0f2027, #203a43, #2c5364);
            color: #fff;
            padding: 30px;
        }

        h2 {
            text-align: center;
            margin-bottom: 30px;
            font-size: 28px;
        }

        form {
            background-color: rgba(255, 255, 255, 0.05);
            padding: 20px;
            border-radius: 12px;
            backdrop-filter: blur(10px);
            max-width: 700px;
            margin: 0 auto 30px;
            box-shadow: 0 8px 20px rgba(0,0,0,0.2);
        }
    </style>

```

```
input[type="text"], input[type="file"], button {
  width: 100%;
  padding: 12px;
  margin-bottom: 15px;
  border: none;
  border-radius: 8px;
  font-size: 16px;
}

input[type="text"], input[type="file"] {
  background: rgba(255,255,255,0.1);
  color: #fff;
}

input::file-selector-button {
  padding: 8px;
  border: none;
  background-color: #3498db;
  color: white;
  border-radius: 6px;
  cursor: pointer;
}

button {
  background-color: #28a745;
  color: white;
  transition: background-color 0.3s ease;
}

button:hover {
  background-color: #218838;
}

table {
  width: 90%;
  margin: 0 auto;
  border-collapse: collapse;
  background-color: rgba(255, 255, 255, 0.05);
  backdrop-filter: blur(10px);
  box-shadow: 0 8px 20px rgba(0,0,0,0.2);
  border-radius: 12px;
  overflow: hidden;
}

th, td {
  padding: 15px;
  text-align: left;
  border-bottom: 1px solid rgba(255,255,255,0.1);
}
```

```

    }

    th {
        background-color: rgba(255,255,255,0.1);
    }

    td form {
        display: inline;
    }

    a {
        color: #00c3ff;
        text-decoration: none;
    }

    a:hover {
        text-decoration: underline;
    }

    .action-buttons input[type="text"] {
        width: 60%;
        display: inline-block;
    }

    .action-buttons button {
        width: auto;
        display: inline-block;
        margin-left: 5px;
    }

    .centered {
        text-align: center;
    }
</style>
</head>
<body>

<h2>📁 Manage Notices</h2>

<!-- Upload Notice Form -->
<form method="POST" enctype="multipart/form-data">
    <input type="text" name="title" placeholder="Enter notice title" required>
    <input type="file" name="file" accept=".pdf" required>
    <button type="submit" name="upload_notice">Upload Notice</button>
</form>

<!-- Notices Table -->
<table>

```

```

<thead>
  <tr>
    <th>ID</th>
    <th>Title</th>
    <th>File</th>
    <th class="centered">Actions</th>
  </tr>
</thead>
<tbody>
  <?php if ($result->num_rows > 0): ?>
    <?php while ($row = $result->fetch_assoc()): ?>
      <tr>
        <td><?=$row['id'] ?></td>
        <td><?=$htmlspecialchars($row['title']) ?></td>
        <td><a href=".."<?=$row['file_path'] ?>"
target="_blank">View</a></td>
        <td class="action-buttons">
          <form method="POST" style="display:inline;">
            <input type="hidden" name="id" value="<?=$row['id'] ?>">
            <button type="submit" name="delete_notice"
onclick="return confirm('Delete this notice?')">🗑️</button>
          </form>
        </td>
      </tr>
    <?php endwhile; ?>
    <?php else: ?>
      <tr><td colspan="4" class="centered">No notices
available.</td></tr>
    <?php endif; ?>
  </tbody>
</table>

</body>
</html>

```

## PYQ MANAGEMENT SYSTEM

### Upload\_pyqs.php

```
<?php
include("../config/db_connect2.php");
session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $subject = $_POST['subject'];
    $semester = $_POST['semester'];
    $year = $_POST['year'];
    $file_path = NULL;

    // File upload path (relative to current file location)
    $target_dir = "../pyqs/pyq_files/";

    // Create folder if it doesn't exist
    if (!is_dir($target_dir)) {
        mkdir($target_dir, 0777, true);
    }

    // Handle file upload
    if (isset($_FILES['pyq_file']) && $_FILES['pyq_file']['error'] === 0) {
        $file_name = time() . "_" . basename($_FILES['pyq_file']['name']);
        $target_file = $target_dir . $file_name;

        if (move_uploaded_file($_FILES['pyq_file']['tmp_name'], $target_file))
        {
            // File path to be saved in DB (relative to root)
            $file_path = "pyqs/pyq_files/" . $file_name;
        } else {
            echo "<p class='error'>Error: File upload failed.</p>";
            exit();
        }
    }

    // Insert into database with status field
    $query = "INSERT INTO question_papers (subject, semester, year, file_url,
status) VALUES (?, ?, ?, ?, 'active')";
    $stmt = mysqli_prepare($conn, $query);

    if ($stmt) {
        mysqli_stmt_bind_param($stmt, "siis", $subject, $semester, $year,
$file_path);
        if (mysqli_stmt_execute($stmt)) {
            header("Location: ../pyqs/view_pyqs.php?msg=PYQ uploaded
successfully!");
            exit();
        }
    }
}
```

```

        } else {
            echo "<p class='error'>Database Execution Error: " .
mysqli_stmt_error($stmt) . "</p>";
        }
    } else {
        echo "<p class='error'>Query Prepare Error: " . mysqli_error($conn) .
"</p>";
    }
}
//including header
include("../headnfoot/header.php");
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Upload PYQ</title>
    <link rel="stylesheet" href="upload.css">
</head>
<body>

<h1>Upload Past Year Question Paper</h1>

<div class="container">
    <form action="upload_pyq.php" method="POST" enctype="multipart/form-data">
        <label for="semester">Semester:</label>
        <select name="semester">
            <?php for ($i = 1; $i <= 6; $i++) {
                echo "<option value='$i'>Semester $i</option>";
            } ?>
        </select>

        <label for="subject">Subject:</label>
        <select name="subject">
            <?php
                $subjects = [
                    "Programming in C", "Computer Architecture", "Programming in
JAVA", "Discrete Mathematics",
                    "Data Structures", "Operating System", "Computer Networks",
"Python Programming",
                    "Design and Analysis of Algorithms", "Software Engineering",
"DBMS", "Matlab",
                    "Internet Technology", "TOC", "Microprocessor", "Numerical
Methods", "AI",
                    "Computer Graphics", "Data Science"
                ];

```

```

        foreach ($subjects as $sub) {
            echo "<option value='$sub'>$sub</option>";
        }
    ?>
</select>

<label for="year">Year:</label>
<input type="number" name="year" min="2019" max="<?php echo date('Y');
?>" required>

<label for="pyq_file">Upload PDF:</label>
<input type="file" name="pyq_file" accept=".pdf" required>

<button type="submit">Upload</button>
</form>

<p><a href="../../pyqs/view_pyqs.php"><img alt="back icon" data-bbox="500 358 520 375"/> View PYQs</a></p>
<div class="upload-manage-container">
    <a href="../../dashboard/upload_pyq.php" class="upload-btn">+ Upload
New PYQ (Teachers Only)</a>
    <a href="../../dashboard/manage_deleted_pyqs.php" class="manage-btn"><img alt="trash icon" data-bbox="830 425 850 440"/>
Manage Deleted PYQs</a>
</div>
</div>

</body>
</html>

```

## DATABASE CONNECT FILES PROGRAMS

### Db\_connect.php

```
<?php
error_reporting(E_ALL);
ini_set('display_errors', 1);

$host = "localhost";
$username = "root";
$password = "";
$database = "college_portal"; //database name

$conn = mysqli_connect($host, $username, $password, $database);

if (!$conn) {
    die("Database Connection Failed: " . mysqli_connect_error());
}

?>
```

### Db\_connect2.php

```
<?php
$servername = "localhost"; // XAMPP server
$username = "root";        // default username
$password = "";            // default password
$database = "pyq_db";      // your database name

// Create connection
$conn = mysqli_connect($servername, $username, $password, $database);

// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

?>
```

### Db\_connect3.php

```
<?php
// Database credentials
$servername = "localhost"; // usually "localhost" for local development
$username = "root";        // default MySQL username (replace if different)
$password = "";            // default MySQL password (replace if different)
$dbname = "book_db";       // New database name

// Create a connection
```



```
$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

?>
```

Connect.php

```
<?php

$host="localhost";
$user="root";
$pass="";
$db="admin_db";
$conn=new mysqli($host,$user,$pass,$db);

if($conn->connect_error){
    die("Failed to connect DB".$conn->connect_error);
}

?>
```

## LOGIN AND AUTHENTICATION SYSTEM

Login.php

```
<?php
session_start();
include "connect.php"; // Database connection include

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $username = trim($_POST["username"]);
    $password = trim($_POST["password"]);

    // Debug: Display the username you're searching for
    echo "Searching for username: " . $username;

    $stmt = $conn->prepare("SELECT * FROM users WHERE username = ?");
    $stmt->bind_param("s", $username);
    $stmt->execute();
    $result = $stmt->get_result();

    // Debug: Show how many rows were returned
    echo "Rows returned: " . $result->num_rows;

    if ($result->num_rows > 0) {
        $row = $result->fetch_assoc();

        // Verify password against the hashed password stored in the database
        if (password_verify($password, $row['password'])) {
            $_SESSION["admin_username"] = $username;
            $_SESSION["user_type"] = $row['user_type'];
            // Set user type (admin or other)

            // Redirect to dashboard
            header("Location: ../dashboard/dashboard.php");
            exit();
        } else {
            echo "✗ Invalid password!";
        }
    } else {
        echo "✗ Invalid username!";
    }
}

$stmt->close(); // Corrected this line
$conn->close();
?>
```

## Logout.php

```
<?php
session_start();
session_unset(); // Clear session variables
session_destroy(); // Destroy the session
header("Location: ../index.php"); // Redirect to home or Login page
exit();
```