

Coding

-----Home Page-----

```
<?php
session_start();
if (isset($_SESSION['name'])) {
    header("Location: subjectchoose.php");
    exit();
}
if (isset($_SESSION['admin_id'])) {
    header("Location:admin_dashboard.php");
    exit();
}
?>
<?php
include 'db_connection.php';
$sql = "SELECT AVG(rating) AS average_rating FROM feedbacks";
$result = $conn->query($sql);
if ($result->num_rows > 0) {
    $row = $result->fetch_assoc();
    $avg_rating = round($row['average_rating'], 1);
} else {
    $avg_rating = 0;
}
?>
<?php
include 'db_connection.php';
$sql_users = "SELECT COUNT(*) AS total_users FROM users";
$user_result = $conn->query($sql_users);
if ($user_result->num_rows > 0) {
    $user_row = $user_result->fetch_assoc();
    $total_users = $user_row['total_users'];
} else {
    $total_users = 0;
}
?>
<?php
include 'db_connection.php';
$sql_user = "SELECT COUNT(*) AS total_users FROM quiz_results";
$quiz_result = $conn->query($sql_user);
if ($quiz_result->num_rows > 0) {
    $quiz_row = $quiz_result->fetch_assoc();
    $total_quiz = $quiz_row['total_users'];
} else {
    $total_users = 0;
}
?>
<?php
include 'db_connection.php';
$sql_above_80 = "SELECT COUNT(*) AS above_80 FROM quiz_results WHERE percentage > 80";
$above_80_result = $conn->query($sql_above_80);
if ($above_80_result->num_rows > 0) {
    $row_above_80 = $above_80_result->fetch_assoc();

```

```

    $above_80_users = $row_above_80['above_80'];
} else {
    $above_80_users = 0;
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Online Examination System</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">
    <!-- External Styles -->
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/4.1.1/animate.min.css" />
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css" />
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons/font/bootstrap-
icons.css" />
    <style>
        body {
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            padding-top: 70px;
            background-color: #f4f6f8;
        }
        .navbar-brand {
            font-weight: bold;
            font-size: 24px;
        }
        .content-section {
            display: none;
        }
        .content-section.active {
            display: block;
        }
        .container-image img {
            width: 100%;
            border-radius: 12px;
            box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
        }
        .form-section {
            background: #ffffff;
            padding: 30px;
            border-radius: 12px;
            box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
        }
        footer {
            background-color: #009688;
            color: white;
            padding: 20px 0;
            text-align: center;

```

```

        margin-top: 60px;
    }
    .forgot-password {
        font-size: 0.9rem;
    }
    .captcha {
        font-size: 18px;
        font-weight: bold;
        letter-spacing: 4px;
        color: #333;
        background: #e9ecef;
        padding: 10px;
        border-radius: 5px;
    }
    .hero {
        background: linear-gradient(rgba(44, 43, 43, 0.6), rgba(105, 104, 104, 0.7)),
        url('https://thinkexam.com/blog/wp-content/uploads/2020/01/scope-of-online-examination-
        system-in-india.png') center/cover no-repeat;
        height: 100vh;
        display: flex;
        flex-direction: column;
        justify-content: center;
        align-items: center;
        color: yellow;
        text-align: center;
    }
    .feature-card:hover {
        transform: translateY(-10px);
        box-shadow: 0 8px 20px rgba(0, 0, 0, 0.2);
        transition: all 0.4s ease-in-out;
    }
    .stats {
        background-color: #f8f9fa;
        padding: 50px 0;
    }
    .testimonial-card {
        background: #fff;
        border-radius: 10px;
        padding: 30px;
        box-shadow: 0 8px 16px rgba(0, 0, 0, 0.1);
        transition: transform 0.3s;
    }
    .testimonial-card:hover {
        transform: scale(1.05);
    }
    .rating-card {
        background: rgb(84, 82, 82);
        padding: 30px;
        border-radius: 12px;
        box-shadow: 0 5px 15px rgba(0, 0, 0, 0.1);
        display: inline-block;
    }

```

```

        text-align: center;
        transition: all 0.3s ease;
    }
    .rating-card:hover {
        transform: scale(1.05);
    }
    .stars {
        font-size: 36px;
        color: gold;
        letter-spacing: 5px;
    }
</style>
</head>
<body>
    <!-- Navbar -->
    <nav class="navbar navbar-expand-lg navbar-dark fixed-top" style="background-color: #009688;">
        <a class="navbar-brand animate__animated animate__fadeInDown" href="#">ONLINE
EXAMINATION SYSTEM</a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarNav">
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbarNav">
            <ul class="navbar-nav ml-auto">
                <li class="nav-item"><a class="nav-link" href="#"
onclick="toggleView('home')">Home</a></li>
                <li class="nav-item"><a class="nav-link" href="#"
onclick="toggleView('login')">Login</a></li>
                <li class="nav-item"><a class="nav-link" href="#"
onclick="toggleView('registration')">Registration</a>
                </li>
                <li class="nav-item"><a class="nav-link" href="#" onclick="toggleView('about')">About
Us</a></li>
                <li class="nav-item"><a class="nav-link" href="#" onclick="toggleView('help')">Help
Us</a></li>
            </ul>
        </div>
    </nav>
    <!-- Views -->
    <div class="container">
        <!-- Home -->
        <div id="homeView" class="content-section active">
            <section class="hero" id="home">
                <h1 class="display-6 fw-bold" data-aos="fade-down">Welcome to <br><span id="typing-
text"></span></h1>
                <p class="lead mt-3" data-aos="fade-up" data-aos-delay="300">Master your skills and test
your knowledge
                with ease.</p>
                <a href="#" class="btn btn-primary btn-lg mt-4" data-aos="zoom-in" data-aos-delay="500"
onclick="toggleView('login')">Get
                Started</a>

```

```

</section>
<!-- Features Section -->
<section class="py-5" id="features">
  <div class="container">
    <h2 class="text-center mb-5" data-aos="fade-up">Our Features</h2>
    <div class="row g-4">
      <div class="col-md-4" data-aos="fade-right">
        <div class="card feature-card p-4 text-center">
          <div class="mb-3"><i class="bi bi-pencil-square fs-1 text-primary"></i></div>
          <h5>Attempt Quizzes</h5>
          <p>Choose from multiple subjects and sharpen your skills with real-time
quizzes.</p>
        </div>
      </div>
      <div class="col-md-4" data-aos="fade-up">
        <div class="card feature-card p-4 text-center">
          <div class="mb-3"><i class="bi bi-bar-chart-line fs-1 text-success"></i></div>
          <h5>Analyze Progress</h5>
          <p>Track your scores, attempt history and improve your weak areas.</p>
        </div>
      </div>
      <div class="col-md-4" data-aos="fade-left">
        <div class="card feature-card p-4 text-center">
          <div class="mb-3"><i class="bi bi-award fs-1 text-warning"></i></div>
          <h5>Get Certified</h5>
          <p>Earn achievement certificates after completing quizzes successfully.</p>
        </div>
      </div>
    </div>
  </div>
</section>
<!-- Statistics Section -->
<section class="stats text-center py-5">
  <div class="container">
    <div class="row g-4">
      <div class="col-md-4" data-aos="fade-right">
        <h2 class="text-primary"><?= $total_users ?></h2>
        <p>Registered Students</p>
      </div>
      <div class="col-md-4" data-aos="fade-up">
        <h2 class="text-success"><?= $total_quiz ?></h2>
        <p>Quizzes Attempted</p>
      </div>
      <div class="col-md-4" data-aos="fade-left">
        <h2 class="text-warning"><?= $above_80_users ?></h2>
        <p>Certificates Awarded</p>
      </div>
    </div>
  </div>
</section>
<!-- About Section -->

```

```

<section class="py-5" id="about">
  <div class="container">
    <h2 class="text-center mb-4" data-aos="fade-up">About Us</h2>
    <p class="lead text-center" data-aos="fade-up" data-aos-delay="200">
      We are committed to providing the best platform for students to test their knowledge
across
      multiple
      subjects. Our system is designed to be easy, fast, and rewarding for learners
everywhere.
    </p>
  </div>
</section>
<!-- Testimonials Section -->
<section class="py-5 bg-light">
  <div class="container">
    <h2 class="text-center mb-5" data-aos="fade-up">What Students Say</h2>
    <div class="row g-4">
      <div class="col-md-4" data-aos="fade-right">
        <div class="testimonial-card p-4 text-center">
          <p>"Best platform to practice and earn certificates!"</p>
          <h6 class="mt-3">- Rahul M.</h6>
        </div>
      </div>
      <div class="col-md-4" data-aos="fade-up">
        <div class="testimonial-card p-4 text-center">
          <p>"User-friendly and very helpful for exam preparation."</p>
          <h6 class="mt-3">- Priya S.</h6>
        </div>
      </div>
      <div class="col-md-4" data-aos="fade-left">
        <div class="testimonial-card p-4 text-center">
          <p>"Loved the instant feedback and detailed results!"</p>
          <h6 class="mt-3">- Aman G.</h6>
        </div>
      </div>
    </div>
  </div>
</section>
<!-- Student Feedbacks Section -->
<section class="py-5 bg-white text-center">
  <div data-aos="zoom-in" data-aos-delay="200">
    <div class="rating-card">
      <!-- <h4 class="mb-3 text-dark">Average Rating</h4> -->
      <div class="stars mb-2">
        <?php
          for ($i = 1; $i <= 5; $i++) {
            echo $i <= round($avg_rating) ? "★" : "☆";
          }
        ?>
      </div>
      <p class="text-muted">Rated <?= $avg_rating ?> out of 5</p>
    </div>
  </div>
</section>

```

```

        </div>
    </div>
</section>
</div>
<!-- Login -->
<div id="loginView" class="content-section mt-4">
    <div class="row justify-content-center">
        <div class="col-md-6">
            <div class="form-section" id="user_id">
                <form id="uform_id">
                    <div class="btn-group w-100 mb-4">
                        <button type="button" class="btn btn-primary" disabled>USER</button>
                        <button type="button" class="btn btn-outline-secondary"
id="first_form">ADMIN</button>
                    </div>
                    <h3 class="text-center text-primary mb-4">User Login</h3>
                    <div class="form-group">
                        <label>Username OR Email</label>
                        <input type="text" name="user_name" class="form-control"
placeholder="Enter your username or email address" required />
                    </div>
                    <div class="form-group">
                        <label>Password</label>
                        <input type="password" name="password" class="form-control"
placeholder="Enter your password" required />
                    </div>
                    <div class="form-group form-check d-flex justify-content-between align-items-
center">
                        <div>
                            <input type="checkbox" class="form-check-input" id="rememberMe">
                            <label class="form-check-label" for="rememberMe">Remember Me</label>
                        </div>
                        <a href="otpsendgmail.html" target="_blank" class="text-danger forgot-
password">Forgot
                            Password?</a>
                    </div>
                    <button type="submit" class="btn btn-primary btn-block" id="login">Login</button>
                    <div class="text-center mt-3">
                        <p>Don't have an account? <a href="#"
onclick="toggleView('registration')">Register</a>
                    </p>
                    </div>
                </form>
            </div>
            <div class="form-section" id="admin_id" style="display: none;">
                <form id="aform_id">
                    <div class="btn-group w-100 mb-4">
                        <button type="button" class="btn btn-outline-secondary"
id="second_form">USER</button>
                        <button type="button" class="btn btn-primary" disabled>ADMIN</button>
                    </div>

```

```

<h3 class="text-center text-primary mb-4">Admin Login</h3>
<div class="form-group">
  <label>Username</label>
  <input type="text" name="username" class="form-control"
    placeholder="Enter your username" required />
</div>
<div class="form-group">
  <label>Password</label>
  <input type="password" name="password" class="form-control"
    placeholder="Enter your password" required />
</div>
<div class="form-group form-check d-flex justify-content-between align-items-
center">
  <div>
    <input type="checkbox" class="form-check-input" id="adminRememberMe">
    <label class="form-check-label" for="adminRememberMe">Remember
Me</label>
  </div>
  <a href="otpsendgmail.html" class="text-danger forgot-password"
target="_blank">Forgot
  Password?</a>
</div>
<button type="submit" class="btn btn-primary btn-block"
id="adminid">Login</button>
</form>
</div>
</div>
</div>
</div>
<div id="registrationView" class="content-section mt-4">
  <div class="row justify-content-center">
    <div class="col-md-8">
      <div class="form-section">
        <h3 class="text-center text-success mb-4">Registration Form</h3>
        <form id="registrationForm">
          <div class="form-group">
            <label for="user_name">Username</label>
            <input type="text" class="form-control" id="user_name" name="username"
required>
          </div>
          <div class="form-group">
            <label for="name">Full Name</label>
            <input type="text" class="form-control" id="name" name="name" required>
          </div>
          <div class="form-group">
            <label for="email">Email address</label>
            <input type="email" class="form-control" id="email" name="email" required>
          </div>
          <div class="form-group">
            <label for="password">Password (min 8 characters)</label>

```



```

        <input type="password" class="form-control" id="password" name="password"
minlength="8"
        required>
    </div>
    <div class="form-group">
        <label for="repassword">Re-type Password</label>
        <input type="password" class="form-control" id="repassword"
name="repassword"
        minlength="8" required>
    </div>
    <div class="form-group">
        <label for="phone">Contact Number</label>
        <input type="text" class="form-control" id="phone" name="phone_number"
maxlength="10"
        required>
    </div>
    <div class="form-group">
        <label for="city">City</label>
        <input type="text" class="form-control" id="city" name="city" required>
    </div>
    <div class="form-group">
        <label for="gender">Gender</label>
        <select class="form-control" id="gender" name="gender" required>
            <option value="">Select</option>
            <option value="male">Male</option>
            <option value="female">Female</option>
            <option value="other">Other</option>
        </select>
    </div>
    <div class="form-group text-center">
        <p class="captcha" id="captcha-text"></p>
        <button type="button" class="btn btn-sm btn-outline-secondary mb-2"
onclick="generateCaptcha()">Refresh</button>
        <input type="text" class="form-control" id="captcha-input" placeholder="Enter
CAPTCHA"
        required>
    </div>
    <button type="submit" class="btn btn-success btn-block" onclick="return
validateCaptcha()"
        id="send_otp">Submit</button>
    <p class="success-msg" id="otpMsg"></p>
    <p class="error-msg" id="otpErr"></p>
</form>
<form id="otpForm" style="display: none;">
    <input type="text" name="otp" placeholder="Enter OTP" id="otp_input" required>
    <button type="submit" onclick="verifyOTP(event)" class="btn-primary">Verify
OTP</button>
    <p class="success-msg" id="verifyMsg"></p>
    <p class="error-msg" id="verifyErr"></p>
</form>
</div>

```

```

        </div>
    </div>
</div>
<!-- About -->
<div id="aboutView" class="content-section">
    <div class="container mt-5">
        <h1 class="text-center">About Us</h1>
        <p class="mt-4">
            Welcome to the Online Examination System! We provide a seamless and secure platform
for conducting
            online exams. Our system ensures integrity, efficiency, and ease of access for students
and
            institutions.
        </p>
        <p>Key Features:</p>
        <ul>
            <li>Secure online examination process</li>
            <li>AI-based question generation</li>
            <li>Real-time monitoring</li>
            <li>Instant results and analytics</li>
            <li>Get Certificate</li>
        </ul>
    </div>
</div>
<div id="helpView" class="content-section">
    <div class="container my-5">
        <div class="card shadow border-0 rounded-4">
            <div class="card-header bg-primary text-white text-center rounded-top-4">
                <h3>Help & Frequently Asked Questions</h3>
            </div>
            <div class="card-body p-4">
                <div class="accordion" id="helpAccordion">
                    <!-- Question 1 -->
                    <div class="accordion-item mb-2">
                        <h2 class="accordion-header" id="headingOne">
                            <button class="accordion-button" type="button" data-bs-toggle="collapse"
                                data-bs-target="#collapseOne">
                                What is the Online Examination System?
                            </button>
                        </h2>
                        <div id="collapseOne" class="accordion-collapse collapse show"
                            data-bs-parent="#helpAccordion">
                            <div class="accordion-body">
                                It is a web-based platform where students can register, take quizzes, view
                                results,
                                write notes, and ask AI chatbot questions related to their subjects.
                            </div>
                        </div>
                    </div>
                    <!-- Question 2 -->
                    <div class="accordion-item mb-2">

```

```

        <h2 class="accordion-header" id="headingTwo">
            <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
                data-bs-target="#collapseTwo">
                    How do I register and log in?
            </button>
        </h2>
        <div id="collapseTwo" class="accordion-collapse collapse"
            data-bs-parent="#helpAccordion">
            <div class="accordion-body">
                Go to the Register page, enter your email or roll number, verify with OTP, and
                set a
                password. Once done, you can log in anytime using your credentials.
            </div>
        </div>
    <!-- Question 3 -->
    <div class="accordion-item mb-2">
        <h2 class="accordion-header" id="headingThree">
            <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
                data-bs-target="#collapseThree">
                    How does the quiz system work?
            </button>
        </h2>
        <div id="collapseThree" class="accordion-collapse collapse"
            data-bs-parent="#helpAccordion">
            <div class="accordion-body">
                Select a subject, start the quiz, and answer multiple-choice questions within
                the given
                time. After submitting, you will see your score and correct answers.
            </div>
        </div>
    </div>
    <!-- Question 4 -->
    <div class="accordion-item mb-2">
        <h2 class="accordion-header" id="headingFour">
            <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
                data-bs-target="#collapseFour">
                    How can I take notes?
            </button>
        </h2>
        <div id="collapseFour" class="accordion-collapse collapse"
            data-bs-parent="#helpAccordion">
            <div class="accordion-body">
                After login, go to "My Notes", select your subject, and write your notes. You
                can update
                them anytime. All notes are saved to your account.
            </div>
        </div>
    </div>

```

```

</div>
<!-- Question 5 -->
<div class="accordion-item mb-2">
  <h2 class="accordion-header" id="headingFive">
    <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
      data-bs-target="#collapseFive">
        What is the use of the chatbot?
    </button>
  </h2>
  <div id="collapseFive" class="accordion-collapse collapse"
    data-bs-parent="#helpAccordion">
    <div class="accordion-body">
      The AI chatbot helps you with your subject-related queries, explanations of
      answers, and
      can also guide you through the system usage.
    </div>
  </div>
</div>
<!-- Question 6 -->
<div class="accordion-item mb-2">
  <h2 class="accordion-header" id="headingSix">
    <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
      data-bs-target="#collapseSix">
        Can I download my quiz result?
    </button>
  </h2>
  <div id="collapseSix" class="accordion-collapse collapse"
    data-bs-parent="#helpAccordion">
    <div class="accordion-body">
      Yes! After completing the quiz, you will get an option to download your result
      as a PDF file including your score, percentage, and answer summary.
    </div>
  </div>
</div>
<!-- Question 6 -->
<div class="accordion-item mb-2">
  <h2 class="accordion-header" id="headingEleven">
    <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
      data-bs-target="#collapseEleven">
        Can I get a certificate after the quiz?
    </button>
  </h2>
  <div id="collapseEleven" class="accordion-collapse collapse"
    data-bs-parent="#helpAccordion">
    <div class="accordion-body">
      Yes! If you score more than 80% in any subject, you will unlock the certificate
      feature.
    </div>
  </div>
</div>

```

```

as
        You can then enter your name and download a beautifully designed certificate
        a PDF.
    </div>
</div>
</div>
<!-- Question 7 -->
<div class="accordion-item mb-2">
    <h2 class="accordion-header" id="headingSeven">
        <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
            data-bs-target="#collapseSeven">
                How do I update my profile or upload an image?
        </button>
    </h2>
    <div id="collapseSeven" class="accordion-collapse collapse"
        data-bs-parent="#helpAccordion">
        <div class="accordion-body">
            After logging in, go to your profile page. Click on the "Update Profile" button
            or image icon to upload a new photo and change your information.
        </div>
    </div>
</div>
<!-- Question 8 -->
<div class="accordion-item mb-2">
    <h2 class="accordion-header" id="headingEight">
        <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
            data-bs-target="#collapseEight">
                How can I give feedback or suggestions?
        </button>
    </h2>
    <div id="collapseEight" class="accordion-collapse collapse"
        data-bs-parent="#helpAccordion">
        <div class="accordion-body">
            Go to the Feedback section in your dashboard. You can rate the system and
submit
            your suggestions. The admin can view all feedbacks.
        </div>
    </div>
</div>
<!-- Question 9 -->
<div class="accordion-item mb-2">
    <h2 class="accordion-header" id="headingNine">
        <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
            data-bs-target="#collapseNine">
                What does the admin dashboard include?
        </button>
    </h2>
    <div id="collapseNine" class="accordion-collapse collapse"

```

```

        data-bs-parent="#helpAccordion">
</div>
    </div>
</div>
<!-- Question 10 -->
<div class="accordion-item mb-2">
    <h2 class="accordion-header" id="headingTen">
        <button class="accordion-button collapsed" type="button" data-bs-
toggle="collapse"
            data-bs-target="#collapseTen">
                How are the quiz questions managed?
        </button>
    </h2>
    <div id="collapseTen" class="accordion-collapse collapse"
        data-bs-parent="#helpAccordion">
        <div class="accordion-body">
            Each subject has its own JavaScript file where quiz questions are stored.

Admin
            can dynamically update these questions through the admin panel.
        </div>
    </div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
</div>

<!-- Footer -->
<footer>
    <p class="mb-1">© 2025 Online Examination System</p>
    <p>Email: onlineexaminationsystem26@gmail.com</p>
</footer>
<!-- Scripts -->
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/js/bootstrap.bundle.min.js"></script>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
<script src="https://unpkg.com/aos@2.3.4/dist/aos.js"></script>
<script src="https://cdn.jsdelivr.net/npm/typed.js@2.0.12"></script>
<script>
```

```

    AOS.init();
    var typed = new Typed("#typing-text", {
      strings: ["Online Examination System", "Your Learning Partner", "Achieve Excellence"],
      typeSpeed: 50,
      backSpeed: 30,
      loop: true
    });
  </script>
</body>
<script>
function toggleView(viewId) {
  document.querySelectorAll('.content-section').forEach(section => {
    section.classList.remove('active');
  });
  document.getElementById(viewId + 'View').classList.add('active');
}
let captchaCode = "";
function generateCaptcha() {
  captchaCode = Math.random().toString(36).substring(2, 8).toUpperCase();
  document.getElementById("captcha-text").innerText = captchaCode;
}
function validateCaptcha() {
  const userCaptcha = document.getElementById("captcha-input").value.toUpperCase();
  if (userCaptcha !== captchaCode) {
    alert("Invalid CAPTCHA. Please try again.");
    generateCaptcha();
    return false;
  }
  return true;
}
window.onload = generateCaptcha;
document.getElementById("registrationForm").addEventListener("submit", function (e) {
  e.preventDefault();
  const sendBtn = document.getElementById('send_otp');
  const otpMsg = document.getElementById("otpMsg");
  const otpErr = document.getElementById("otpErr");
  otpMsg.textContent = "";
  otpErr.textContent = "";
  sendBtn.disabled = true;
  sendBtn.style.cursor = "not-allowed";
  const formData = new FormData(this);
  fetch("send_email.php", {
    method: "POST",
    body: formData
  })
  .then(response => response.text())
  .then(data => {
    if (data.trim() === "success") {
      alert('✅ OTP sent Email in spam');
      document.getElementById("otpForm").style.display = "block";
    }
  })
});

```

```

else if (data.trim() === "already_registered") {
    alert('✅ already_registered');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
}
else if (data.trim() === "already_password") {
    alert('❌ Already password stored database');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
}
else if (data.trim() === "phone") {
    alert('❌ Invalid Phone Number');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
}
else if (data.trim() === "email") {
    alert('Already Email registered');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
}
else if (data.trim() === "password") {
    alert('❌ Password do not match');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
}
else {
    alert('❌ Failed to send OTP: ' + data);
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
}
});
.catch(error => {
    alert('❌ Error: ' + error);
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
});
});
function verifyOTP(event) {
    event.preventDefault();
    const otp = document.getElementById('otp_input').value;
    fetch("verify.php", {
        method: "POST",
        headers: { "Content-Type": "application/x-www-form-urlencoded" },
        body: "otp=" + encodeURIComponent(otp)
    })
    .then(response => response.text())
    .then(data => {
        if (data.trim().toLowerCase() === "success") {
            alert('✅ Registration successfully!');
            window.location.reload();
        }
    })

```



```

        else if (data.trim() === "Failed") {
            alert('❌ Something went wrong please try again');
        }
        else if (data.trim() === "invalid") {
            alert('❌ Invalid Otp');
        }
        else {
            alert('❌ Error: ' + data);
        }
    });
}
document.addEventListener("DOMContentLoaded", function () {
    const userForm = document.getElementById('user_id');
    const adminForm = document.getElementById('admin_id');
    document.getElementById('first_form').onclick = () => {
        userForm.style.display = 'none';
        adminForm.style.display = 'block';
    };
    document.getElementById('second_form').onclick = () => {
        userForm.style.display = 'block';
        adminForm.style.display = 'none';
    };
});
document.getElementById("uform_id").addEventListener("submit", function (e) {
    e.preventDefault();
    const sendBt = document.getElementById('login');
    sendBt.disabled=true;
    const formData = new FormData(this);
    fetch("user.php", {
        method: "POST",
        body: formData
    })
    .then(response => response.text())
    .then(data => {
        const result = data.trim();
        if (result === "success") {
            window.location.href = "subjectchoose.php";
        } else if (result === "wrong") {
            alert('❌ Incorrect password. ');
            sendBt.disabled=false;
        }
        else if (result === "username") {
            alert('❌ User not found. ');
            sendBt.disabled=false;
        }
        else {
            alert('❌ Something went wrong: ' + result);
            sendBt.disabled=false;
        }
    })
});

```

```

});
document.getElementById("aform_id").addEventListener("submit", function (e) {
    e.preventDefault();
    const sendB = document.getElementById('adminid');
    sendB.disabled=true;
    const formData = new FormData(this);
    fetch("admin.php", {
        method: "POST",
        body: formData
    })
    .then(response => response.text())
    .then(data => {
        const result = data.trim();
        if (result === "success") {
            window.location.href = "admin_dashboard.php";
        } else if (result === "wrong") {
            alert('✗ Incorrect password.');
```

sendB.disabled=false;

```

        } else if (result === "username") {
            alert('✗ User not found.');
```

sendB.disabled=false;

```

        } else {
            alert('✗ Something went wrong: ' + result);
            sendB.disabled=false;
        }
    })
});
<script>
</html>

```

-----User Dashboard-----

```

<?php include 'auth_check.php'; ?>
<?php include 'result.php'; ?>
<?php
include 'db_connection.php';
$name = $_SESSION['name'];
$sql = "SELECT AVG(rating) AS average_rating FROM feedbacks";
$result = $conn->query($sql);
if ($result->num_rows > 0) {
    $row = $result->fetch_assoc();
    $avg_rating = round($row['average_rating'], 1);
} else {
    $avg_rating = 0;
}
?>
<!DOCTYPE html>
<html lang="en">
<head>

```

```

<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Reading Page</title>
<link rel="icon" type="image/png" href="favicon-32x32.png">
<!-- Bootstrap -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet" />
<!-- Animate.css -->
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/animate.css/4.1.1/animate.min.css" />
<!-- AOS (Animate on Scroll) -->
<link rel="stylesheet" href="https://unpkg.com/aos@2.3.1/dist/aos.css" />
<!-- Bootstrap Icons -->
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-
icons.min.css" />
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
<link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.5/font/bootstrap-icons.css"
rel="stylesheet">
<style>
  body {
    background-color: #f8f9fa;
  }
  .navbar {
    background-color: #000;
    color: white;
  }
  .navbar a {
    color: white;
  }
  .container-section>div {
    background-color: #343a40;
    color: white;
    margin: 15px;
    padding: 30px;
    text-align: center;
    border-radius: 10px;
    transition: transform 0.4s ease, box-shadow 0.4s ease;
    cursor: pointer;
  }
  .container-section>div:hover {
    transform: scale(1.05);
    box-shadow: 0 8px 25px rgba(0, 0, 0, 0.3);
  }
  .q {
    margin: 10px;
    padding: 10px 20px;
    border: none;
    background: #007bff;
    color: white;
    font-size: 18px;
  }

```

```
    border-radius: 5px;
  }
.q a {
  text-decoration: none;
  color: white;
}
#exercise-questions,
#notetext,
#feedback,
#get-cer,
#ai {
  display: none;
  transition: all 0.3s ease;
}
.light-theme .container-section>div {
  background-color: #7a7b7c !important;
  color: black !important;
}
.chat-container {
  background: #ffffff;
  border-radius: 20px;
  box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
  width: 100%;
  height: 80vh;
  flex-direction: column;
  position: relative;
}
.chat-header {
  background: #0d6efd;
  color: #fff;
  padding: 15px;
  text-align: center;
  font-weight: 500;
  font-size: 1.2rem;
}
.chat-box {
  flex: 1;
  overflow-y: auto;
  padding: 20px;
  scroll-behavior: smooth;
}
.chat-box::-webkit-scrollbar {
  width: 6px;
}
.chat-box::-webkit-scrollbar-thumb {
  background-color: #fbf8f8;
  border-radius: 10px;
}
.input-area {
  display: flex;
  border-top: 1px solid #ddd;
```

```
padding: 10px;
background-color: #f8f9fa;
color: #000;
}
.input-area input {
  flex: 1;
  border: none;
  padding: 12px;
  border-radius: 10px;
  margin-right: 10px;
}
.input-area input:focus {
  outline: none;
}
.input-area button {
  background-color: #0d6efd;
  color: #f8f2f2;
  border: none;
  padding: 12px 20px;
  border-radius: 10px;
}
.user-msg,
.bot-msg {
  margin-bottom: 15px;
  padding: 10px 15px;
  border-radius: 10px;
  max-width: 80%;
  word-wrap: break-word;
}
.user-msg {
  background-color: #e9f5ff;
  align-self: flex-end;
  text-align: right;
  color: #000;
}
.bot-msg {
  background-color: #e0ffe6;
  align-self: flex-start;
  color: #000;
}
.icon-btn {
  position: absolute;
  top: 10px;
  right: 80px;
  color: #fff;
  font-size: 30px;
  cursor: pointer;
}
.icon-btns {
  position: absolute;
  top: 10px;
```

```

        right: 10px;
        color: #ea0a0a;
        font-size: 40px;
        cursor: pointer;
    }
    .stars input {
        display: none;
    }
    .stars label {
        font-size: 2rem;
        color: gray;
        cursor: pointer;
        transition: color 0.2s;
    }
    .stars input:checked~label,
    .stars label:hover,
    .stars label:hover~label {
        color: gold;
    }
    .feedback-section {
        max-height: 200px;
        overflow-y: auto;
    }
    .feedback-item {
        border-bottom: 1px solid #dee2e6;
        padding: 0.5rem 0;
    }
    #usern {
        text-transform: uppercase;
        color: #007bff;
    }
</style>
<script type="module"
src="https://unpkg.com/ionicons@7.1.0/dist/ionicons/ionicons.esm.js"></script>
<script nomodule src="https://unpkg.com/ionicons@7.1.0/dist/ionicons/ionicons.js"></script>
</head>
<body id="body">
    <!-- Navbar -->
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark px-4">
        <a class="navbar-brand" href="#">ONLINE EXAMINATION SYSTEM</a>
        <a class="navbar-brand" href="user_information.php">WELCOME, <span id="usern"><?=$name ?></span>!</a>
        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav">
            <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse justify-content-end" id="navbarNav">
            <ul class="navbar-nav align-items-center">
                <li class="nav-item">
                    <button class="btn btn-outline-light" id="searchBtn">
                        <i class="bi bi-search"></i>

```

```

        </button>
        <input type="text" class="form-control search-input" id="searchInput"
placeholder="Search..."
        style="width: 0; position: absolute; right: 0; top: 50%; transform: translate(-320px,-
20px); transition: width 0.5s ease;">
    </li>
    <li class="nav-item"><a class="nav-link" href="subjectchoose.php">HOME</a></li>
    <li class="nav-item"><a class="nav-link" href="user_information.php">PROFILE</a></li>
    <li class="nav-item"><a class="nav-link" href="logout.php">LOGOUT</a></li>
    <!-- Theme Toggle Button -->
    <li class="nav-item">
        <button class="btn btn-sm btn-light ms-2" id="themeToggle">
            <i class="bi bi-moon-fill"></i> Theme
        </button>
    </li>
    <!-- Search Icon with Animation -->
</ul>
</div>
</nav>
<!-- Sections -->
<div class="container mt-4 container-section">
    <div class="container0" data-aos="flip-up">
        <a href="c_programming.php" style="text-decoration: none; color: white;">
            <h1>C LANGUAGE</h1>
        </a>
    </div>
    <div class="container1" data-aos="flip-up">
        <a href="c++.php" style="text-decoration: none; color: white;">
            <h1>C++ LANGUAGE</h1>

        </a>
    </div>
    <div class="container1" data-aos="flip-up">
        <a href="java.php" style="text-decoration: none; color: white;">
            <h1>JAVA</h1>

        </a>
    </div>
    <div class="container2" data-aos="flip-up">
        <a href="python.php" style="text-decoration: none; color: white;">
            <h1>PYTHON</h1>

        </a>
    </div>
    <div class="container3" data-aos="flip-up">
        <a href="nodejs.php" style="text-decoration: none; color: white;">
            <h1>NODE JS</h1>
        </a>
    </div>
    <div class="container4" data-aos="flip-right">
        <a href="reactjs.php" style="text-decoration: none; color: white;">

```

```

        <h1>REACTJS</h1>
    </a>
</div>
<div class="container7" data-aos="flip-up" onclick="togglenote()">
    <h1>My Notes</h1>
</div>
<div class="container" id="notetext">
    <form id="noteForm">
        <ion-icon name="close-outline" class="icon-btns" onclick="togglenote()"></ion-icon>
        <div class="mb-3">
            <textarea class="form-control fs-5 p-3" id="textarea" name="texts" rows="15"
                placeholder="Enter your notes here..." required></textarea>
        </div>
        <div class="d-flex gap-3 mb-4">
            <button type="submit" class="btn btn-success px-4">Save</button>
            <button type="button" class="btn btn-primary px-4" id="start">
                <i class="bi bi-mic-fill me-1"></i>Start
            </button>
            <button type="button" class="btn btn-danger px-4" id="clear">Clear</button>
        </div>
    </form>
</div>
<div class="container8" data-aos="flip-up" onclick="togglefeed()">
    <h1>FEEDBACK</h1>
    <div class="text-center">
        <h4 class="text-success">Average Rating: <?= $avg_rating ?></h4>
        <div style="font-size: 24px; color: gold;">
            <?php
                for ($i = 1; $i <= 5; $i++) {
                    echo $i <= round($avg_rating) ? "★" : "☆";
                }
            ?>
        </div>
    </div>
</div>
<div class="feedbacks" id="feedback" style="display:none;">
    <div class="container d-flex justify-content-center align-items-center min-vh-100">
        <ion-icon name="close-outline" class="icon-btns" onclick="togglefeed()"></ion-icon>
        <div class="card shadow-lg w-100" style="max-width: 500px;">
            <div class="card-body">
                <h4 class="card-title text-center mb-4">Rate Us</h4>
                <form id="submitfeedback">
                    <div class="stars d-flex justify-content-center mb-3 flex-row-reverse">
                        <input type="radio" id="star5" name="rating" value="5">
                        <label for="star5">★</label>
                        <input type="radio" id="star4" name="rating" value="4">
                        <label for="star4">★</label>
                        <input type="radio" id="star3" name="rating" value="3">
                        <label for="star3">★</label>
                        <input type="radio" id="star2" name="rating" value="2">
                        <label for="star2">★</label>
                    </div>
                </form>
            </div>
        </div>
    </div>

```



```

        <input type="radio" id="star1" name="rating" value="1" required>
        <label for="star1">★</label>
    </div>
    <div class="mb-3">
        <textarea name="feedback" class="form-control" rows="3" placeholder="Your
Feedback"
            required></textarea>
    </div>
    <button type="submit" class="btn btn-success w-100" id="submit">Submit
Feedback</button>
    </form>
    <div class="feedback-section mt-4" id="feedback-list">
        <h5 class="border-bottom pb-2">Recent Feedbacks</h5>
    </div>
</div>
</div>
</div>
<div class="container11" data-aos="flip-up" onclick="toggleai()">
    <h1>AI CHATBOT</h1>
</div>
<div class="ai" id="ai">
    <div class="chat-container d-flex flex-column">
        <div class="chat-header">
            AI Chatbot
            <ion-icon name="scan-outline" class="icon-btn"></ion-icon>
            <ion-icon name="close-outline" class="icon-btns" onclick="toggleai()"></ion-icon>
        </div>
        <div class="chat-box d-flex flex-column" id="chatBox"></div>
        <div class="input-area">
            <input type="text" id="userInput" placeholder="Type your question..." class="form-
control">
            <button onclick="sendMessage()">Send</button>
        </div>
    </div>
</div>
<div class="container11" data-aos="flip-up" onclick="toggleExercise()">
    <h1>EXERCISE</h1>
</div>
<div id="exercise-questions" class="text-center mt-3">
    <h1>
        EXERCISE
    </h1>
    <ion-icon name="close-outline" class="icon-btns" onclick="toggleExercise()"></ion-icon>
    <div class="question text-center mt-5" data-aos="zoom-in">
        <button class="q"><a href="c_languagequestion.php" target="_blank">C
LANGUAGE</a></button>
        <button class="q"><a href="javaquestion.php" target="_blank">JAVA</a></button>
        <button class="q"><a href="pythonquestion.php" target="_blank">PYTHON</a></button>
        <button class="q"><a href="C++question.php" target="_blank">C++</a></button>
        <button class="q"><a href="reactjsquestion.php" target="_blank">REACTJS</a></button>

```

```

        <button class="q"><a href="nodejsquestion.php" target="_blank">NODEJS</a></button>
    </div>
</div>
<div class="container10" data-aos="flip-up">
    <a href="result.html" style="text-decoration: none; color: white;" target="_blank">
        <h1>RESULT</h1>
    </a>
</div>
<div class="container12" data-aos="flip-up" onclick="togglecer()">
    <h1>GET CERTIFICATE</h1>
</div>
<div id="get-cer" class="text-center mt-3">
    <ion-icon name="close-outline" class="icon-btns" onclick="togglecer()"></ion-icon>
    <div class="question text-center mt-5" data-aos="zoom-in">
        <button class="btn btn-secondary q" id="C"><a href="certificate.php?subject=C
LANGUAGE"
        class="text-white text-decoration-none">C
LANGUAGE</a></button>
        <button class="btn btn-secondary q" id="JAVA"><a href="certificate.php?subject= JAVA
PROGRAMMING"
        class="text-white text-decoration-none">JAVA</a></button>
        <button class="btn btn-secondary q" id="PYTHON"><a
href="certificate.php?subject=PYTHON PROGRAMMING"
        class="text-white text-decoration-none">PYTHON</a></button>
        <button class="btn btn-secondary q" id="CPP"><a href="certificate.php?subject=C++
PROGRAMMING"
        class="text-white text-decoration-none">C++</a></button>
        <button class="btn btn-secondary q" id="REACTJS"><a
href="certificate.php?subject=REACTJS PROGRAMMING"
        class="text-white text-decoration-none">REACTJS</a></button>
        <button class="btn btn-secondary q" id="NODEJS"><a href="certificate.php?subject=NODES
PROGRAMMING"
        class="text-white text-decoration-none">NODEJS</a></button>
    </div>
</div>
</div>
<div>
<div class="bg-black text-light" style="box-shadow: 0 4px 8px rgba(0, 0, 0, 0.3);">
    <div class="container">
        <div class="row">
            <div class="col-6 col-lg-4">
                <h4 class="pt-3 fw-bold"></h4>
                <p>Online Examination System</p>
                <p>6398428250</p>
            </div>
            <div class="col">
            </div>
            <div class="col">
                <h4 class="pt-3">More</h4>
                <ul class="list-unstyled">
                    <li><a href="result.html" class="text-decoration-none text-light">Result</a></li>
                    <li><a href="#" class="text-decoration-none text-light">Get Certificate</a></li>

```

```

    </ul>
  </div>
  <div class="col">
    <h4 class="pt-3">Categories</h4>
    <ul class="list-unstyled">
      <li><a href="#" class="text-decoration-none text-light">Update Profile</a></li>
    </ul>
  </div>
  <div class="col-6 col-lg-3 text-lg-end">
    <h4 class="pt-3">Social Media Icons</h4>
    <div>
      <a href="instagram.com" class="text-decoration-none text-light"><i
        class="bi bi-instagram fs-6 me-80" style="font-size:45px"></i></a>
      <a href="github.com" class="text-decoration-none text-light"><i class="bi bi-github fs-6
me-80"
        style="font-size:45px"></i></a>
      <a href="https://www.linkedin.com/in/mohd-tanzeem-
908161314?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=an
droid_app"
        class="text-decoration-none text-light"><i class="bi bi-linkedin fs-6 me-3"
        style="font-size:45px"></i></a>
      <a href="https://www.youtube.com/@CodingSeekhoAurSeekhao-q2i"
        class="text-decoration-none text-light"><i class="bi bi-youtube fs-6 me-3"
        style="font-size:45px"></i></a>
    </div>
  </div>
</div>
</div>
<hr>
<div class="d-flex justify-content-between">
  <p>2025 Coding Seekho Aur Seekhao. All Rights Reserved.</p>
  <div>
    <a href="#" class="text-decoration-none text-light me-4">Terms And Condition</a>
    <a href="#" class="text-decoration-none text-light">Privacy Policy</a>
  </div>
</div>
</div>
</footer>
<script src="https://unpkg.com/aos@2.3.1/dist/aos.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
<script>AOS.init();</script>
<script>
  // PHP से subject_names भेजना
  const passedSubjects = <?php echo json_encode($_SESSION['subject_names'] ?? []); ?>;
  const subjectMap = {
    'C': 'C Programming',
    'JAVA': 'JAVA Programming',
    'PYTHON': 'Python Programming',
    'CPP': 'C++ Programming',
    'REACTJS': 'ReactJS Programming',
    'NODEJS': 'NodeJS Programming'
  }

```

```

    });
    // Disable all buttons initially
    Object.keys(subjectMap).forEach(id => {
        const btn = document.getElementById(id);
        btn.disabled = true;
        btn.classList.remove('btn-success');
        btn.classList.add('btn-secondary');
        btn.title = "Score above 80%, certificate Unlocked";
    });
    // Enable if passedSubjects contains subject
    passedSubjects.forEach(subject => {
        for (let id in subjectMap) {
            if (subjectMap[id].toLowerCase() === subject.toLowerCase()) {
                const btn = document.getElementById(id);
                btn.disabled = false;
                btn.classList.remove('btn-secondary');
                btn.classList.add('btn-success');
            }
        }
    });
    function togglecer() {
        const div = document.getElementById("get-cer");
        div.style.display = div.style.display === "none" ? "block" : "none";
        document.getElementById("get-cer").classList.toggle("d-none");
    }
</script>
<script>
    const searchBtn = document.getElementById('searchBtn');
    const searchInput = document.getElementById('searchInput');
    const containerSection = document.querySelector('.container-section');
    const boxes = document.querySelectorAll('.container-section > div');
    // input initially hidden
    searchInput.style.width = "0";
    searchInput.style.display = "none";
    // Save original display style
    const originalDisplay = {};
    boxes.forEach(box => {
        originalDisplay[box.dataset.aos] = window.getComputedStyle(box).display;
    });
    // Handle search button click
    searchBtn.addEventListener('click', () => {
        if (searchInput.style.display === 'none' || searchInput.style.width === '0px') {
            searchInput.style.display = "block";
            searchBtn.style.display = "none";
            setTimeout(() => {
                searchInput.style.width = '200px';
                searchInput.focus();
            }, 10);
        } else {
            searchInput.style.width = '0';
            setTimeout(() => {

```

```

        searchBtn.style.display = "block";
        searchInput.style.display = "none";
        searchInput.value = "";
        showAllBoxes();
    }, 500);
}
});
// Handle blur
searchInput.addEventListener('blur', () => {
    searchInput.style.width = '0';
    setTimeout(() => {
        searchInput.style.display = "none";
        searchBtn.style.display = "block";
        searchInput.value = "";
        showAllBoxes();
    }, 500);
});
// Search input filter
searchInput.addEventListener('input', () => {
    const filter = searchInput.value.trim().toLowerCase();
    let anyMatch = false;
    boxes.forEach(box => {
        const heading = box.querySelector('h1');
        if (heading && heading.innerText.trim() !== "") {
            const text = heading.innerText.toLowerCase();
            if (text.includes(filter)) {
                box.style.display = originalDisplay[box.dataset.aos] || "block";
                anyMatch = true;
            } else {
                box.style.display = "none";
            }
        } else {
            box.style.display = originalDisplay[box.dataset.aos] || "block";
        }
    });
    const existing = document.getElementById('noResult');
    if (existing) existing.remove();
    if (!anyMatch) {
        const noResult = document.createElement('div');
        noResult.id = 'noResult';
        noResult.className = 'text-center text-danger mt-5';
        noResult.innerHTML = '<h3>No Result Found</h3>';
        containerSection.appendChild(noResult);
    }
});
function showAllBoxes() {
    boxes.forEach(box => {
        box.style.display = originalDisplay[box.dataset.aos] || "block";
    });
    const existing = document.getElementById('noResult');
    if (existing) existing.remove();
}

```

```

    }
  </script>
</body>
<script>
  const textarea = document.getElementById("textarea");
  const startBtn = document.getElementById("start");
  const clearBtn = document.getElementById("clear");
  // Toggle Sections
  function toggleExercise() {
    const div = document.getElementById("exercise-questions");
    div.style.display = div.style.display === "none" ? "block" : "none";
  }
  function togglenote() {
    fetchNote();
    const div = document.getElementById("notetext");
    div.style.display = div.style.display === "none" ? "block" : "none";
  }
  function togglefeed() {
    const div = document.getElementById("feedback");
    div.style.display = div.style.display === "none" ? "block" : "none";
  }
  function toggleai() {
    const ai = document.getElementById('ai');
    ai.style.display = ai.style.display === 'flex' ? 'none' : 'flex';
  }
  // Clear notes
  clearBtn.addEventListener("click", () => textarea.value = "");
  // Voice to Text
  const recognition = new (window.SpeechRecognition || window.webkitSpeechRecognition)();
  recognition.lang = 'en-US';
  recognition.onresult = (e) => textarea.value += e.results[0][0].transcript + " ";
  startBtn.addEventListener("click", () => recognition.start());
  // Save Notes
  document.getElementById("noteForm").addEventListener("submit", function (e) {
    e.preventDefault();
    const formData = new FormData(this);
    fetch("text.php", { method: "POST", body: formData })
      .then(res => res.text())
      .then(data => alert(data.trim() === "success" ? '✅ Saved successfully' : data))
      .catch(err => alert("Error:" + err));
  });
  // Fetch Note
  function fetchNote() {
    fetch("fetch_text.php")
      .then(res => res.json())
      .then(data => textarea.value = data.status === "success" ? data.text : "")
      .catch(err => console.error("Fetch Error:", err));
  }
  fetch("get_session.php")
    .then(res => res.json())
    .then(data => {

```

```

        if (data.status === "success") {
            sessionStorage.setItem("Roll_Number", data.roll_number);
            fetchNote();
        }
    });
    window.onload = () => {
        fetchNote();
    };
    async function sendMessage() {
        let userInput = document.getElementById("userInput").value;
        if (!userInput.trim()) return;
        let chatBox = document.getElementById("chatBox");
        // Show user's message
        let userDiv = document.createElement("div");
        userDiv.className = "user-msg align-self-end";
        userDiv.innerHTML = `<b>You:</b> ${userInput}`;
        chatBox.appendChild(userDiv);
        document.getElementById("userInput").value = "";
        // Scroll to the bottom
        chatBox.scrollTop = chatBox.scrollHeight;
        try {
            let response = await fetch("chatbot.php", {
                method: "POST",
                headers: { "Content-Type": "application/json" },
                body: JSON.stringify({ message: userInput })
            });
            let data = await response.json();
            let botDiv = document.createElement("div");
            botDiv.className = "bot-msg align-self-start";
            botDiv.innerHTML = `<b>Bot:</b> ${data.response || "Something went wrong"}`;
            chatBox.appendChild(botDiv);
            // Scroll to the bottom after bot message
            chatBox.scrollTop = chatBox.scrollHeight;
        } catch (err) {
            console.error("Error:", err);
            chatBox.innerHTML += `<div class='bot-msg align-self-start'><b>Bot:</b> Error connecting to
server</div>`;
            chatBox.scrollTop = chatBox.scrollHeight; // Scroll to bottom even in error case
        }
    }
    const toggleBtn = document.getElementById("themeToggle");
    const body = document.getElementById("body");
    toggleBtn.addEventListener("click", () => {
        document.body.classList.toggle("light-theme");
        const icon = toggleBtn.querySelector("i");
        if (document.body.classList.contains("light-theme")) {
            body.style.backgroundColor = "black";
            icon.classList.remove("bi-moon-fill");
            icon.classList.add("bi-sun-fill");
        } else {
            body.style.backgroundColor = "white";

```

```

        icon.classList.remove("bi-sun-fill");
        icon.classList.add("bi-moon-fill");
    }
});
// Prevent browser back button (as discussed earlier)
history.pushState(null, null, location.href);
window.onpopstate = function () {
    history.go(1);
};
// Automatically logout when the user closes the tab
window.onbeforeunload = function () {
    // Send an AJAX request to logout.php to handle session destroy
    var xhr = new XMLHttpRequest();
    xhr.open("GET", "logout.php", false); // synchronous request to block unloading
    xhr.send();
};
function generateStars(rating) {
    let stars = "";
    for (let i = 1; i <= 5; i++) {
        stars += i <= rating ? '★' : '☆';
    }
    return stars;
}
function loadFeedbacks() {
    fetch('fetch_feedbacks.php')
        .then(response => response.json())
        .then(data => {
            let feedbackHTML = '<h5 class="border-bottom pb-2">Feedbacks</h5>';
            data.forEach(item => {
                feedbackHTML += `
                <div class='feedback-item'>
                    <strong>${item.name}</strong> <span>${generateStars(item.rating)}</span>
                    <p class="mb-0">${item.feedback}</p>
                </div>`;
            });
            document.getElementById('feedback-list').innerHTML = feedbackHTML;
        })
        .catch(error => console.error('Error:', error));
}
window.onload = loadFeedbacks;
document.getElementById("submitfeedback").addEventListener("submit", function (e) {
    e.preventDefault();
    const sendBtn = document.getElementById('submit');
    const formData = new FormData(this);
    sendBtn.disabled = true;
    fetch("submit_feedback.php", {
        method: "POST",
        body: formData
    })
        .then(response => response.text())
        .then(data => {

```



```

const result = data.trim();
if (result === "success") {
    window.location.reload();
} else if (result === "failed") {
    alert(' ❌ Something went wrong please try again. ');
    sendBtn.disabled = false;
} else {
    alert(' ❌ Something went wrong: ' + result);
    sendBtn.disabled = false;
}
})
});
</script>
</html>

```

```

-----User Authentication-----
<?php
session_start();
if (!isset($_SESSION['Roll_Number']) || !isset($_SESSION['name']) || !isset($_SESSION['email'])) {
    header("Location: index.php");
    exit();
}
$roll = $_SESSION['Roll_Number'];
$name = $_SESSION['name'];
$email = $_SESSION['email'];
?>

```

-----All Content-----

-----C Language-----

```

<?php include 'auth_check.php'; ?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>C Tutorial</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

```

```

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" />
<style>
  body {
    background-color: #dae0e0;
  }

  .content-section {
    display: none;
  }

  .content-section.active {
    display: block;
  }

  pre {
    background-color: #f8f9fa;
    padding: 1rem;
    border-radius: 0.5rem;
    color: #cb0eaf;
  }

  @media (min-width: 768px) {
    #sidebarMenu {
      position: sticky;
      top: 0;
      height: 100vh;
    }
  }
</style>
</head>

<body id="body1">
  <nav class="navbar bg-success navbar-dark">
    <div class="container-fluid">
      <a class="navbar-brand" href="c_programming.php">C</a>
      <a class="navbar-brand" href="c++.php">C++</a>
      <a class="navbar-brand" href="java.php">JAVA</a>
      <a class="navbar-brand" href="python.php">PYTHON</a>
      <a class="navbar-brand" href="nodejs.php">NODEJS</a>
      <a class="navbar-brand" href="reactjs.php">REACTJS</a>
      <button class="btn btn-outline-light d-md-none" type="button" data-bs-toggle="collapse"
        data-bs-target="#sidebarMenu">
        ≡ Topics
      </button>
      <a class="navbar-brand" href="subjectchoose.php">HOME</a>
      <button class="btn btn-outline-light" ><a class="nav-link"
href="logout.php">LOGOUT</a></button>
      <button class="btn btn-outline-light" id="iconid">Theme</button>
    </div>
  </nav>

```

```

<div class="container-fluid">
  <div class="row">
    <!-- Sidebar -->
    <div class="col-md-3 col-lg-2 collapse d-md-block sidebar p-3 bg-light" id="sidebarMenu">
      <h5 class="text-center">Topics</h5>
      <div id="btnList" class="d-grid gap-1">
        <button class="btn btn-light btn-toggle" onclick="showSection('intro',
this)">Introduction</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('syntax',
this)">Syntax</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('variables',
this)">Variables</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('datatypes', this)">Data
Types</button>
        <button class="btn btn-light btn-toggle"
onclick="showSection('inputoutput', this)">Input/Output</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('ifelse', this)">If
Else</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('loops',
this)">Loops</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('functions',
this)">Functions</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('pointers',
this)">Pointers</button>
      </div>
    </div>

    <!-- Main content -->
    <div class="col-md-9 col-lg-10 p-4">
      <div id="intro" class="content-section active">
        <h1>C Tutorial</h1>
        <div class="bg-success text-white p-4 my-3 rounded">
          <h3>Learn C Language</h3>
          <p>
            C is a general-purpose programming language created by Dennis Ritchie at Bell Labs
in 1972.

            It is powerful, efficient, and widely used in system programming, operating systems,
and
            embedded systems.
          </p>
        </div>
        <h4>Example:</h4>
        <pre>
#include <stdio.h>

int main() {
    printf("Hello, C!\n");
    return 0;
}

        </pre>
      </div>
    </div>
  </div>
</div>

```

</div>

<div id="syntax" class="content-section">

<h2>C Syntax</h2>

<p>C code structure includes functions, curly braces, and semicolons.</p>

<pre>

```
#include <stdio.h>
```

```
int main() {
```

```
    printf("This is C syntax.\n");
```

```
    return 0;
```

```
}
```

</pre>

</div>

<div id="variables" class="content-section">

<h2>Variables</h2>

<p>Variables hold data types and values that can be manipulated during program execution.</p>

<pre>

```
int a = 10;
```

```
float b = 3.14;
```

```
char c = 'A';
```

</pre>

</div>

<div id="datatypes" class="content-section">

<h2>Data Types</h2>

<p>C includes several data types: int, float, double, char, etc.</p>

<pre>

```
int number = 5;
```

```
double pi = 3.14159;
```

```
bool flag = 1; // _Bool in older C standards
```

</pre>

</div>

<div id="inputoutput" class="content-section">

<h2>Input and Output</h2>

<p>Use scanf for input and printf for output.</p>

<pre>

```
int age;
```

```
printf("Enter your age: ");
```

```
scanf("%d", &age);
```

```
printf("You are %d years old", age);
```

</pre>

</div>

<div id="ifelse" class="content-section">

<h2>If Else</h2>

<p>Conditional statements control the program flow.</p>

<pre>

```

int number = 10;
if (number > 0) {
    printf("Positive\n");
} else {
    printf("Non-positive\n");
}

```

```

</pre>
</div>

```

```

<div id="loops" class="content-section">
    <h2>Loops</h2>
    <p>Loops repeat a block of code multiple times.</p>
    <pre>
for (int i = 0; i < 5; i++) {
    printf("%d ", i);
}
    </pre>
</div>

```

```

<div id="functions" class="content-section">
    <h2>Functions</h2>
    <p>Functions break a program into reusable blocks.</p>
    <pre>
int add(int a, int b) {
    return a + b;
}

int main() {
    printf("Sum: %d", add(5, 3));
    return 0;
}
    </pre>
</div>

```

```

<div id="pointers" class="content-section">
    <h2>Pointers</h2>
    <p>Pointers store the address of variables and are used for dynamic memory and array
handling.</p>
    <pre>
int a = 10;
int *ptr = &a;
printf("Value of a: %d", *ptr);
    </pre>
</div>
</div>
</div>

```

```

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
<script>

```

```

const btn = document.getElementById("iconid");
const body = document.getElementById("body1");
let lightMode = true;

btn.addEventListener("click", () => {
  body.style.backgroundColor = lightMode ? "#545353" : "#dae0e0";
  body.style.color = lightMode ? "white" : "black";
  lightMode = !lightMode;
});

function showSection(id, clickedBtn) {
  const sections = document.querySelectorAll(".content-section");
  sections.forEach((sec) => sec.classList.remove("active"));
  const active = document.getElementById(id);
  if (active) active.classList.add("active");

  const buttons = document.querySelectorAll("#btnList .btn-toggle");
  buttons.forEach((btn) => {
    btn.classList.remove("btn-success");
    btn.classList.add("btn-light");
  });

  clickedBtn.classList.remove("btn-light");
  clickedBtn.classList.add("btn-success");
}
</script>
</body>

</html>

```

-----C++ Language-----

```

<?php include 'auth_check.php'; ?>

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>C++ Tutorial</title>
  <link rel="icon" type="image/png" href="favicon-32x32.png">

  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" />
  <style>
    body {
      background-color: #dae0e0;

```

```

    }

    .content-section {
        display: none;
    }

    .content-section.active {
        display: block;
    }

    pre {
        background-color: #f8f9fa;
        padding: 1rem;
        border-radius: 0.5rem;
        color: #cb0eaf;
    }

    @media (min-width: 768px) {
        #sidebarMenu {
            position: sticky;
            top: 0;
            height: 100vh;
        }
    }
</style>
</head>

<body id="body1">
    <nav class="navbar bg-success navbar-dark">
        <div class="container-fluid">
            <a class="navbar-brand" href="c_programming.php">C</a>
            <a class="navbar-brand" href="c++.php">C++</a>
            <a class="navbar-brand" href="java.php">JAVA</a>
            <a class="navbar-brand" href="python.php">PYTHON</a>
            <a class="navbar-brand" href="nodejs.php">NODEJS</a>
            <a class="navbar-brand" href="reactjs.php">REACTJS</a>
            <button class="btn btn-outline-light d-md-none" type="button" data-bs-toggle="collapse"
                data-bs-target="#sidebarMenu">
                ≡ Topics
            </button>
            <a class="navbar-brand" href="subjectchoose.php">HOME</a>
            <button class="btn btn-outline-light"><a class="nav-link"
href="logout.php">LOGOUT</a></button>
            <button class="btn btn-outline-light" id="iconid">Theme</button>
        </div>
    </nav>

    <div class="container-fluid">
        <div class="row">
            <!-- Sidebar -->
            <div class="col-md-3 col-lg-2 collapse d-md-block sidebar p-3 bg-light" id="sidebarMenu">

```

```

        <h5 class="text-center">Topics</h5>
        <div id="btnList" class="d-grid gap-1">
            <button class="btn btn-light btn-toggle" onclick="showSection('intro',
this)">Introduction</button>
            <button class="btn btn-light btn-toggle" onclick="showSection('syntax',
this)">Syntax</button>
            <button class="btn btn-light btn-toggle" onclick="showSection('variables',
this)">Variables</button>
            <button class="btn btn-light btn-toggle" onclick="showSection('datatypes', this)">Data
Types</button>
            <button class="btn btn-light btn-toggle"
            onclick="showSection('inputoutput', this)">Input/Output</button>
            <button class="btn btn-light btn-toggle" onclick="showSection('ifelse', this)">If
Else</button>
            <button class="btn btn-light btn-toggle" onclick="showSection('loops',
this)">Loops</button>
            <button class="btn btn-light btn-toggle" onclick="showSection('functions',
this)">Functions</button>
            <button class="btn btn-light btn-toggle" onclick="showSection('oop', this)">OOP
Concepts</button>
        </div>
    </div>

    <!-- Main content -->
    <div class="col-md-9 col-lg-10 p-4">
        <div id="intro" class="content-section active">
            <h1>C++ Tutorial</h1>
            <div class="bg-success text-white p-4 my-3 rounded">
                <h3>Learn C++ Language</h3>
                <p>
                    C++ is a high-level, object-oriented programming language developed as an
extension of the C
                    language. It
                    supports features like classes, inheritance, polymorphism, and encapsulation, making
it
                    suitable for
                    system and application software.
                </p>
            </div>
            <h4>Example:</h4>
            <pre>
#include <iostream>
using namespace std;

int main() {
    cout <<< "Hello, C++!";
    return 0;
}
            </pre>
        </div>
    </div>

```


<div id="syntax" class="content-section">

<h2>C++ Syntax</h2>

<p>C++ uses similar syntax to C, but introduces object-oriented features.</p>

<pre>

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    cout <<< "Welcome to C++";
```

```
    return 0;
```

```
}</pre>
```

</div>

<div id="variables" class="content-section">

<h2>Variables</h2>

<p>Variables store data that can be used later in the program.</p>

<pre>

```
int age = 25;
```

```
float weight = 67.5;
```

```
char grade = 'A';</pre>
```

</div>

<div id="datatypes" class="content-section">

<h2>Data Types</h2>

<p>Basic data types in C++ include int, float, double, char, and bool.</p>

<pre>

```
int a = 10;
```

```
double pi = 3.1415;
```

```
bool isReady = true;</pre>
```

</div>

<div id="inputoutput" class="content-section">

<h2>Input and Output</h2>

<p>C++ uses cin for input and cout for output.</p>

<pre>

```
int age;
```

```
cout <<< "Enter age: ";
```

```
cin >>> age;
```

```
cout <<< "You are " <<< age <<< " years old."</pre>
```

</div>

<div id="ifelse" class="content-section">

<h2>If Else</h2>

<p>Conditional logic is implemented using if, else if, and else.</p>

<pre>

```
int num = 10;
```

```
if(num > 0) {
```

```
    cout <<< "Positive";
```

```
} else {
```

```
    cout <<< "Non-positive";
```

```
}</pre>
```

```

</div>

<div id="loops" class="content-section">
  <h2>Loops</h2>
  <p>C++ supports while, do-while, and for loops.</p>
  <pre>
for(int i = 0; i < 5; i++) {
  cout <<<< i <<<< " ";
}</pre>
</div>

<div id="functions" class="content-section">
  <h2>Functions</h2>
  <p>Functions allow code to be reused and organized.</p>
  <pre>
int add(int a, int b) {
  return a + b;
}</pre>
</div>

<div id="oop" class="content-section">
  <h2>OOP Concepts</h2>
  <p>C++ is known for supporting Object-Oriented Programming. Basic concepts
include:</p>
  <ul>
    <li><strong>Class</strong>: A blueprint for objects</li>
    <li><strong>Object</strong>: Instance of class</li>
    <li><strong>Inheritance</strong>: Deriving new classes from existing ones</li>
    <li><strong>Polymorphism</strong>: Same function behaves differently</li>
    <li><strong>Encapsulation</strong>: Wrapping data and code together</li>
  </ul>
  <pre>
class Person {
public:
  string name;
  void sayHello() {
    cout <<<< "Hello " <<<< name;
  }
};</pre>
</div>
</div>
</div>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
<script>
  const btn = document.getElementById("iconid");
  const body = document.getElementById("body1");
  let lightMode = true;

```

```

    btn.addEventListener("click", () => {
        body.style.backgroundColor = lightMode ? "#545353" : "#dae0e0";
        body.style.color = lightMode ? "white" : "black";
        lightMode = !lightMode;
    });

    function showSection(id, clickedBtn) {
        const sections = document.querySelectorAll(".content-section");
        sections.forEach((sec) => sec.classList.remove("active"));
        const active = document.getElementById(id);
        if (active) active.classList.add("active");

        // Reset all button backgrounds
        const buttons = document.querySelectorAll("#btnList .btn-toggle");
        buttons.forEach((btn) => {
            btn.classList.remove("btn-success");
            btn.classList.add("btn-light");
        });

        // Highlight clicked button
        clickedBtn.classList.remove("btn-light");
        clickedBtn.classList.add("btn-success");
    }
</script>
</body>

</html>

```

-----JAVA Language-----

```

<?php include 'auth_check.php'; ?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Java Tutorial</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" />
    <style>
        body {
            background-color: #dae0e0;
        }

        .content-section {

```

```

        display: none;
    }

    .content-section.active {
        display: block;
    }

    pre {
        background-color: #f8f9fa;
        padding: 1rem;
        border-radius: 0.5rem;
        color: #cb0eaf;
        overflow-x: auto;
    }

    @media (min-width: 768px) {
        #sidebarMenu {
            position: sticky;
            top: 0;
            height: 100vh;
        }
    }
</style>
</head>

<body id="body1">
<nav class="navbar bg-success navbar-dark">
    <div class="container-fluid">
        <a class="navbar-brand" href="c_programming.php">C</a>
        <a class="navbar-brand" href="c++.php">C++</a>
        <a class="navbar-brand" href="java.php">JAVA</a>
        <a class="navbar-brand" href="python.php">PYTHON</a>
        <a class="navbar-brand" href="nodejs.php">NODEJS</a>
        <a class="navbar-brand" href="reactjs.php">REACTJS</a>
        <button class="btn btn-outline-light d-md-none" type="button" data-bs-toggle="collapse" data-
bs-target="#sidebarMenu">
            ≡ Topics
        </button>
        <a class="navbar-brand" href="subjectchoose.php">HOME</a>
        <button class="btn btn-outline-light" ><a class="nav-link"
href="logout.php">LOGOUT</a></button>
        <button class="btn btn-outline-light" id="iconid">Theme</button>
    </div>
</nav>

<div class="container-fluid">
    <div class="row">
        <!-- Sidebar -->
        <div class="col-md-3 col-lg-2 collapse d-md-block sidebar p-3 bg-light" id="sidebarMenu">
            <h5 class="text-center">Topics</h5>
            <div id="btnList" class="d-grid gap-1">

```

```

        <button class="btn btn-light btn-toggle" onclick="showSection('intro',
this)">Introduction</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('syntax', this)">Syntax</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('variables',
this)">Variables</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('datatypes', this)">Data
Types</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('inputoutput',
this)">Input/Output</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('ifelse', this)">If Else</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('loops', this)">Loops</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('functions',
this)">Methods</button>
        <button class="btn btn-light btn-toggle" onclick="showSection('oop', this)">OOP
Concepts</button>
    </div>
</div>

```

```

<!-- Main content -->
<div class="col-md-9 col-lg-10 p-4">
    <div id="intro" class="content-section active">
        <h1>Java Tutorial</h1>
        <div class="bg-success text-white p-4 my-3 rounded">
            <h3>Learn Java Language</h3>
            <p>
                Java is a popular, high-level, object-oriented programming language developed by Sun
                Microsystems. It's platform-independent, secure, and widely used in web applications, mobile apps,
                and enterprise systems.
            </p>

```

```

        </div>
        <h4>Example:</h4>
        <pre>

```

```

public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, Java!");
    }
}

```

```

    </pre>
</div>

```

```

    <div id="syntax" class="content-section">
        <h2>Java Syntax</h2>
        <p>Java syntax includes class definitions, method declarations, and statements.</p>
        <pre>

```

```

public class MyClass {
    public static void main(String[] args) {
        System.out.println("This is Java syntax.");
    }
}

```

```

    </pre>
</div>

```

```

    <div id="variables" class="content-section">

```

```
<h2>Variables</h2>
<p>Variables store data to be used later.</p>
<pre>
int age = 30;
double salary = 55000.75;
String name = "Rahul";
boolean isJavaFun = true;</pre>
</div>

<div id="datatypes" class="content-section">
  <h2>Data Types</h2>
  <p>Java has primitive and non-primitive data types.</p>
  <pre>
byte a = 10;
short b = 5000;
int c = 100000;
long d = 150000000000L;
float e = 5.75f;
double f = 19.99;
char g = 'A';
boolean h = true;</pre>
</div>

<div id="inputoutput" class="content-section">
  <h2>Input and Output</h2>
  <p>Use Scanner for input and System.out for output.</p>
  <pre>
import java.util.Scanner;

public class InputExample {
  public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter your name: ");
    String name = scanner.nextLine();
    System.out.println("Hello, " + name);
  }
}</pre>
</div>

<div id="ifelse" class="content-section">
  <h2>If Else</h2>
  <p>Conditional logic using if-else statements.</p>
  <pre>
int number = 10;
if(number > 0) {
  System.out.println("Positive number");
} else {
  System.out.println("Negative or zero");
}</pre>
</div>
```

```
<div id="loops" class="content-section">
  <h2>Loops</h2>
  <p>Java supports for, while, and do-while loops.</p>
  <pre>
for(int i = 0; i < 5; i++) {
  System.out.println("i = " + i);
}</pre>
</div>
```

```
<div id="functions" class="content-section">
  <h2>Methods</h2>
  <p>Methods are blocks of code to perform a task.</p>
  <pre>
public class Main {
  static int add(int x, int y) {
    return x + y;
  }

  public static void main(String[] args) {
    int result = add(10, 20);
    System.out.println("Sum = " + result);
  }
}</pre>
</div>
```

```
<div id="oop" class="content-section">
  <h2>OOP Concepts</h2>
  <p>Java is an Object-Oriented Programming language.</p>
  <ul>
    <li><strong>Class</strong>: Template for objects</li>
    <li><strong>Object</strong>: Instance of a class</li>
    <li><strong>Inheritance</strong>: Acquire properties of another class</li>
    <li><strong>Polymorphism</strong>: Many forms of a method</li>
    <li><strong>Encapsulation</strong>: Protect data using access modifiers</li>
  </ul>
  <pre>
class Animal {
  void sound() {
    System.out.println("Animal makes a sound");
  }
}

class Dog extends Animal {
  void sound() {
    System.out.println("Dog barks");
  }
}

public class Main {
  public static void main(String[] args) {
    Animal a = new Dog();
  }
}</pre>
```

```

        a.sound();
    }
} </pre>
</div>

</div>
</div>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
<script>
    const btn = document.getElementById("iconid");
    const body = document.getElementById("body1");
    let lightMode = true;

    btn.addEventListener("click", () => {
        body.style.backgroundColor = lightMode ? "#212529" : "#dae0e0";
        body.style.color = lightMode ? "white" : "black";
        lightMode = !lightMode;
    });

    function showSection(id, clickedBtn) {
        const sections = document.querySelectorAll(".content-section");
        sections.forEach((sec) => sec.classList.remove("active"));
        const active = document.getElementById(id);
        if (active) active.classList.add("active");

        const buttons = document.querySelectorAll("#btnList .btn-toggle");
        buttons.forEach((btn) => {
            btn.classList.remove("btn-success");
            btn.classList.add("btn-light");
        });

        clickedBtn.classList.remove("btn-light");
        clickedBtn.classList.add("btn-success");
    }
</script>
</body>

</html>

```

-----Python Language-----

```
<?php include 'auth_check.php'; ?>
```

```
<!DOCTYPE html>
<html lang="en">
```



```

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Python Tutorial</title>
  <link rel="icon" type="image/png" href="favicon-32x32.png">

  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" />
  <style>
    body {
      background-color: #dae0e0;
    }

    .content-section {
      display: none;
    }

    .content-section.active {
      display: block;
    }

    pre {
      background-color: #f8f9fa;
      padding: 1rem;
      border-radius: 0.5rem;
      color: #cb0eaf;
    }

    @media (min-width: 768px) {
      #sidebarMenu {
        position: sticky;
        top: 0;
        height: 100vh;
      }
    }
  </style>
</head>

<body id="body1">
  <nav class="navbar bg-success navbar-dark">
    <div class="container-fluid">
      <a class="navbar-brand" href="c_programming.php">C</a>
      <a class="navbar-brand" href="c++.php">C++</a>
      <a class="navbar-brand" href="java.php">JAVA</a>
      <a class="navbar-brand" href="python.php">PYTHON</a>
      <a class="navbar-brand" href="nodejs.php">NODEJS</a>
      <a class="navbar-brand" href="reactjs.php">REACTJS</a>
      <button class="btn btn-outline-light d-md-none" type="button" data-bs-toggle="collapse"
        data-bs-target="#sidebarMenu">
        ≡ Topics
      </button>
    </div>
  </nav>

```

```

        <a class="navbar-brand" href="subjectchoose.php">HOME</a>
        <button class="btn btn-outline-light" ><a class="nav-link"
href="logout.php">LOGOUT</a></button>
        <button class="btn btn-outline-light" id="iconid">Theme</button>
    </div>
</nav>

<div class="container-fluid">
    <div class="row">
        <!-- Sidebar -->
        <div class="col-md-3 col-lg-2 collapse d-md-block sidebar p-3 bg-light" id="sidebarMenu">
            <h5 class="text-center">Topics</h5>
            <div id="btnList" class="d-grid gap-1">
                <button class="btn btn-light btn-toggle" onclick="showSection('intro',
this)">Introduction</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('syntax',
this)">Syntax</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('variables',
this)">Variables</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('datatypes', this)">Data
Types</button>
                <button class="btn btn-light btn-toggle"
onclick="showSection('inputoutput', this)">Input/Output</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('ifelse', this)">If
Else</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('loops',
this)">Loops</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('functions',
this)">Functions</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('oop', this)">OOP
Concepts</button>
            </div>
        </div>

        <!-- Main content -->
        <div class="col-md-9 col-lg-10 p-4">
            <div id="intro" class="content-section active">
                <h1>Python Tutorial</h1>
                <div class="bg-success text-white p-4 my-3 rounded">
                    <h3>Learn Python Language</h3>
                    <p>
                        Python is an easy-to-learn, high-level, interpreted programming language. It is widely
used
                        for
                        web development, data science, artificial intelligence, automation, and more.

Python's
                        simple
                        syntax makes it ideal for beginners.
                    </p>
                </div>
                <h4>Example:</h4>
            </div>
        </div>
    </div>

```

```
<pre>
print("Hello, Python!")
</pre>
</div>
```

```
<div id="syntax" class="content-section">
  <h2>Python Syntax</h2>
  <p>Python uses indentation to define code blocks, making it easy to read.</p>
  <pre>
# This is a comment
x = 5
print(x)
  </pre>
</div>
```

```
<div id="variables" class="content-section">
  <h2>Variables</h2>
  <p>Variables store values in Python and don't require explicit data type declarations.</p>
  <pre>
x = 10
name = "John"
is_active = True
  </pre>
</div>
```

```
<div id="datatypes" class="content-section">
  <h2>Data Types</h2>
  <p>Python supports various data types like int, float, string, and boolean.</p>
  <pre>
age = 25 # int
height = 5.9 # float
name = "John" # string
is_active = True # boolean
  </pre>
</div>
```

```
<div id="inputoutput" class="content-section">
  <h2>Input and Output</h2>
  <p>Python uses input() to take input and print() to display output.</p>
  <pre>
name = input("Enter your name: ")
print("Hello, " + name)
  </pre>
</div>
```

```
<div id="ifelse" class="content-section">
  <h2>If Else</h2>
  <p>Conditional statements in Python are implemented using if, elif, and else.</p>
  <pre>
x = 10
if x > 0:
```

```

    print("Positive")
else:
    print("Non-positive")
    </pre>
</div>

<div id="loops" class="content-section">
    <h2>Loops</h2>
    <p>Python supports for and while loops for iteration.</p>
    <pre>

# For loop
for i in range(5):
    print(i)

# While loop
i = 0
while i < 5:
    print(i)
    i += 1
    </pre>
</div>

<div id="functions" class="content-section">
    <h2>Functions</h2>
    <p>Functions in Python allow you to reuse code efficiently.</p>
    <pre>

def add(a, b):
    return a + b

result = add(3, 5)
print(result)
    </pre>
</div>

<div id="oop" class="content-section">
    <h2>OOP Concepts</h2>
    <p>Python supports object-oriented programming with concepts like classes, objects,
inheritance, and
    polymorphism.</p>
    <pre>

class Person:
    def __init__(self, name, age):
        self.name = name
        self.age = age

    def greet(self):
        print(f"Hello, my name is {self.name} and I am {self.age} years old.")

# Creating an object of the Person class
person = Person("John", 30)
person.greet()
    </pre>
</div>

```

```

        </pre>
    </div>
</div>
</div>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
<script>
    const btn = document.getElementById("iconid");
    const body = document.getElementById("body1");
    let lightMode = true;

    btn.addEventListener("click", () => {
        body.style.backgroundColor = lightMode ? "#545353" : "#dae0e0";
        body.style.color = lightMode ? "white" : "black";
        lightMode = !lightMode;
    });

    function showSection(id, clickedBtn) {
        const sections = document.querySelectorAll(".content-section");
        sections.forEach((sec) => sec.classList.remove("active"));
        const active = document.getElementById(id);
        if (active) active.classList.add("active");

        // Reset all button backgrounds
        const buttons = document.querySelectorAll("#btnList .btn-toggle");
        buttons.forEach((btn) => {
            btn.classList.remove("btn-success");
            btn.classList.add("btn-light");
        });

        // Highlight clicked button
        clickedBtn.classList.remove("btn-light");
        clickedBtn.classList.add("btn-success");
    }
</script>
</body>

</html>

```

-----NodeJs Language-----

```
<?php include 'auth_check.php'; ?>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```

<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<title>Node.js Tutorial</title>
<link rel="icon" type="image/png" href="favicon-32x32.png">

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" />
<style>
  body {
    background-color: #dae0e0;
  }

  .content-section {
    display: none;
  }

  .content-section.active {
    display: block;
  }

  pre {
    background-color: #f8f9fa;
    padding: 1rem;
    border-radius: 0.5rem;
    color: #cb0eaf;
  }

  @media (min-width: 768px) {
    #sidebarMenu {
      position: sticky;
      top: 0;
      height: 100vh;
    }
  }
</style>
</head>

<body id="body1">
  <nav class="navbar bg-success navbar-dark">
    <div class="container-fluid">
      <a class="navbar-brand" href="c_programming.php">C</a>
      <a class="navbar-brand" href="c++.php">C++</a>
      <a class="navbar-brand" href="java.php">JAVA</a>
      <a class="navbar-brand" href="python.php">PYTHON</a>
      <a class="navbar-brand" href="nodejs.php">NODEJS</a>
      <a class="navbar-brand" href="reactjs.php">REACTJS</a>
      <button class="btn btn-outline-light d-md-none" type="button" data-bs-toggle="collapse"
        data-bs-target="#sidebarMenu">
        ≡ Topics
      </button>
      <a class="navbar-brand" href="subjectchoose.php">HOME</a>
    </div>
  </nav>

```

```

        <button class="btn btn-outline-light" ><a class="nav-link"
href="logout.php">LOGOUT</a></button>
        <button class="btn btn-outline-light" id="iconid">Theme</button>
    </div>
</nav>

<div class="container-fluid">
    <div class="row">
        <!-- Sidebar -->
        <div class="col-md-3 col-lg-2 collapse d-md-block sidebar p-3 bg-light" id="sidebarMenu">
            <h5 class="text-center">Topics</h5>
            <div id="btnList" class="d-grid gap-1">
                <button class="btn btn-light btn-toggle" onclick="showSection('intro',
this)">Introduction</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('syntax',
this)">Syntax</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('modules',
this)">Modules</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('express',
this)">Express.js</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('httpserver', this)">HTTP
                Server</button>
                <button class="btn btn-light btn-toggle" onclick="showSection('filehandling', this)">File
                Handling</button>
            </div>
        </div>

        <!-- Main content -->
        <div class="col-md-9 col-lg-10 p-4">
            <div id="intro" class="content-section active">
                <h1>Node.js Tutorial</h1>
                <div class="bg-success text-white p-4 my-3 rounded">
                    <h3>Learn Node.js</h3>
                    <p>
                        Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine. It is used for
                        building fast,
                        scalable network applications. Node.js is particularly suited for building real-time
                        applications,
                        web servers, and APIs.
                    </p>
                </div>
                <h4>Example:</h4>
                <pre>
console.log("Hello, Node.js!");
                </pre>
            </div>

            <div id="syntax" class="content-section">
                <h2>Node.js Syntax</h2>
                <p>Node.js follows JavaScript syntax and uses asynchronous, event-driven programming.
                Below is a

```

```
simple Node.js example.</p>
<pre>
console.log("This is Node.js!");
</pre>
</div>
```

```
<div id="modules" class="content-section">
  <h2>Node.js Modules</h2>
  <p>Node.js comes with a set of built-in modules to handle various tasks such as file
handling,
    networking, and more.</p>
  <h4>Example of Importing a Module:</h4>
  <pre>
const fs = require('fs');

fs.readFile('example.txt', 'utf8', (err, data) => {
  if (err) {
    console.log(err);
    return;
  }
  console.log(data);
});
  </pre>
</div>
```

```
<div id="express" class="content-section">
  <h2>Express.js Framework</h2>
  <p>Express is a minimal and flexible Node.js web application framework that provides a
robust set of
    features for building web and mobile applications.</p>
  <h4>Setting up an Express Server:</h4>
  <pre>
const express = require('express');
const app = express();

app.get('/', (req, res) => {
  res.send('Hello, Express!');
});

app.listen(3000, () => {
  console.log('Server is running on port 3000');
});
  </pre>
</div>
```

```
<div id="httpserver" class="content-section">
  <h2>Node.js HTTP Server</h2>
  <p>Node.js can also be used to create an HTTP server to handle client requests and
responses.</p>
  <h4>Basic HTTP Server Example:</h4>
  <pre>
```



```

const http = require('http');

const server = http.createServer((req, res) => {
  res.writeHead(200, { 'Content-Type': 'text/plain' });
  res.end('Hello, HTTP Server!');
});

server.listen(3000, () => {
  console.log('Server running at http://localhost:3000/');
});

```

</pre>
 </div>

<div id="filehandling" class="content-section">
 <h2>File Handling in Node.js</h2>
 <p>Node.js provides a File System (fs) module to interact with the file system, allowing us

to read

from and write to files.</p>
 <h4>Reading a File Example:</h4>
 <pre>
const fs = require('fs');

fs.readFile('example.txt', 'utf8', (err, data) => {
 if (err) {
 console.log(err);
 return;
 }
 console.log(data);
});

 </div>
 <h4>Writing to a File Example:</h4>
 <pre>
fs.writeFile('output.txt', 'Hello, Node.js!', (err) => {
 if (err) throw err;
 console.log('File has been saved!');
});

 </div>
 </div>
 </div>
 </div>
 <div>
 <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
 <script>
const btn = document.getElementById("iconid");
const body = document.getElementById("body1");
let lightMode = true;

btn.addEventListener("click", () => {

```

        body.style.backgroundColor = lightMode ? "#545353" : "#dae0e0";
        body.style.color = lightMode ? "white" : "black";
        lightMode = !lightMode;
    });

    function showSection(id, clickedBtn) {
        const sections = document.querySelectorAll(".content-section");
        sections.forEach((sec) => sec.classList.remove("active"));
        const active = document.getElementById(id);
        if (active) active.classList.add("active");

        // Reset all button backgrounds
        const buttons = document.querySelectorAll("#btnList .btn-toggle");
        buttons.forEach((btn) => {
            btn.classList.remove("btn-success");
            btn.classList.add("btn-light");
        });

        // Highlight clicked button
        clickedBtn.classList.remove("btn-light");
        clickedBtn.classList.add("btn-success");
    }
</script>
</body>

</html>

```

-----ReactJs Language-----

```

<?php include 'auth_check.php'; ?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>React.js Tutorial</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" />
    <style>

```

```

body {
    background-color: #dae0e0;
}

.content-section {
    display: none;
}

.content-section.active {
    display: block;
}

pre {
    background-color: #f8f9fa;
    padding: 1rem;
    border-radius: 0.5rem;
    color: #cb0eaf;
}

@media (min-width: 768px) {
    #sidebarMenu {
        position: sticky;
        top: 0;
        height: 100vh;
    }
}
</style>
</head>

<body id="body1">
    <nav class="navbar bg-success navbar-dark">
        <div class="container-fluid">
            <a class="navbar-brand" href="c_programming.php">C</a>
            <a class="navbar-brand" href="c++.php">C++</a>
            <a class="navbar-brand" href="java.php">JAVA</a>
            <a class="navbar-brand" href="python.php">PYTHON</a>
            <a class="navbar-brand" href="nodejs.php">NODEJS</a>
            <a class="navbar-brand" href="reactjs.php">REACTJS</a>
            <button class="btn btn-outline-light d-md-none" type="button" data-bs-toggle="collapse"
                data-bs-target="#sidebarMenu">
                ≡ Topics
            </button>
            <a class="navbar-brand" href="subjectchoose.php">HOME</a>
            <button class="btn btn-outline-light"><a class="nav-link"
href="logout.php">LOGOUT</a></button>
            <button class="btn btn-outline-light" id="iconid">Theme</button>
        </div>
    </nav>

    <div class="container-fluid">
        <div class="row">

```

```

<!-- Sidebar -->
<div class="col-md-3 col-lg-2 collapse d-md-block sidebar p-3 bg-light" id="sidebarMenu">
  <h5 class="text-center" style="color: black !important;">Topics</h5>
  <div id="btnList" class="d-grid gap-1">
    <button class="btn btn-light btn-toggle" onclick="showSection('intro',
this)">Introduction</button>
    <button class="btn btn-light btn-toggle" onclick="showSection('syntax',
this)">Syntax</button>
    <button class="btn btn-light btn-toggle"
      onclick="showSection('components', this)">Components</button>
    <button class="btn btn-light btn-toggle" onclick="showSection('jsx', this)">JSX</button>
    <button class="btn btn-light btn-toggle" onclick="showSection('props',
this)">Props</button>
    <button class="btn btn-light btn-toggle" onclick="showSection('state',
this)">State</button>
    <button class="btn btn-light btn-toggle" onclick="showSection('hooks',
this)">Hooks</button>
  </div>
</div>

<!-- Main content -->
<div class="col-md-9 col-lg-10 p-4">
  <div id="intro" class="content-section active">
    <h1>React.js Tutorial</h1>
    <div class="bg-success text-white p-4 my-3 rounded">
      <h3>Learn React.js</h3>
      <p>
        React is a JavaScript library for building user interfaces. It allows developers to create
        large web applications that can change data, without reloading the page. It uses a
        component-based architecture for creating interactive UIs.
      </p>
    </div>
    <h4>Example:</h4>
    <pre>
console.log("Hello, React!");
  </pre>
    </div>

    <div id="syntax" class="content-section">
      <h2>React.js Syntax</h2>
      <p>React uses JavaScript syntax along with JSX for writing components. Below is a basic
example:</p>
      <pre>
function Hello() {
  return <h1>Hello, React!</h1>;
}
  </pre>
    </div>

    <div id="components" class="content-section">
      <h2>React Components</h2>

```

In React, a component is a reusable, self-contained piece of code that can manage its state and

render UI elements. Components can be functional or class-based.</p>

Example of a Functional Component:</h4>

<pre>

```
function MyComponent() {  
  return <h1>Hello from MyComponent</h1>;  
}
```

</pre>

Example of a Class Component:</h4>

<pre>

```
class MyComponent extends React.Component {  
  render() {  
    return <h1>Hello from MyComponent</h1>;  
  }  
}
```

</pre>

</div>

<div id="jsx" class="content-section">

<h2>JSX (JavaScript XML)</h2>

<p>JSX is a syntax extension that allows you to write HTML-like elements in JavaScript. It

makes

React code easier to write and understand.</p>

JSX Example:</h4>

<pre>

```
function App() {  
  return (  
    <div>  
      <h1>Hello, JSX!</h1>  
      <p>This is JSX syntax inside a React component.</p>  
    </div>  
  );  
}
```

</pre>

</div>

<div id="props" class="content-section">

<h2>Props in React</h2>

<p>Props (short for properties) are read-only attributes passed from a parent component

to a child

component. They allow data to flow from one component to another.</p>

Example of Passing Props:</h4>

<pre>

```
function Welcome(props) {  
  return <h1>Hello, {props.name}!</h1>;  
}
```

```
function App() {  
  return <Welcome name="React" />;  
}
```

```

    </pre>
  </div>

  <div id="state" class="content-section">
    <h2>State in React</h2>
    <p>State is an object that stores a component's dynamic data. When the state changes,
React
    re-renders the component.</p>
    <h4>Example of Using State:</h4>
    <pre>
import React, { useState } from 'react';

function Counter() {
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>Click me</button>
    </div>
  );
}

    </pre>
  </div>

  <div id="hooks" class="content-section">
    <h2>React Hooks</h2>
    <p>React Hooks are functions that let you use state and other React features in
functional
    components. The most common hooks are useState and useEffect.</p>
    <h4>useState Hook Example:</h4>
    <pre>
import React, { useState } from 'react';

function Counter() {
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>Click me</button>
    </div>
  );
}

    </pre>
    <h4>useEffect Hook Example:</h4>
    <pre>
import React, { useState, useEffect } from 'react';

function Timer() {
  const [seconds, setSeconds] = useState(0);

```

```

useEffect(() => {
  const interval = setInterval(() => {
    setSeconds(seconds + 1);
  }, 1000);

  return () => clearInterval(interval); // Clean up the interval when component unmounts
}, [seconds]);

return <h1>Timer: {seconds} seconds</h1>;
}

</pre>
</div>
</div>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
<script>
  const btn = document.getElementById("iconid");
  const body = document.getElementById("body1");
  let lightMode = true;

  btn.addEventListener("click", () => {
    body.style.backgroundColor = lightMode ? "#545353" : "#dae0e0";
    body.style.color = lightMode ? "white" : "black";
    lightMode = !lightMode;
  });

  function showSection(id, clickedBtn) {
    const sections = document.querySelectorAll(".content-section");
    sections.forEach((sec) => sec.classList.remove("active"));
    const active = document.getElementById(id);
    if (active) active.classList.add("active");

    // Reset all button backgrounds
    const buttons = document.querySelectorAll("#btnList .btn-toggle");
    buttons.forEach((btn) => {
      btn.classList.remove("btn-success");
      btn.classList.add("btn-light");
    });

    // Highlight clicked button
    clickedBtn.classList.remove("btn-light");
    clickedBtn.classList.add("btn-success");
  }
</script>
</body>

</html>

```

-----C Language Question-----

```
const allQuestions = [
```

```
    { question: "What is the size of int in C?", options: ["2 bytes", "4 bytes", "8 bytes", "Depends on Compiler"], answer: "Depends on Compiler" },
```

```
    { question: "Which symbol is used for single-line comments in C?", options: ["//", "/* */", "#", "--"], answer: "//" },
```

```
    { question: "Which function is used to print in C?", options: ["print()", "printf()", "cout", "display()"], answer: "printf()" },
```

```
    { question: "What is used to declare a constant in C?", options: ["const", "final", "constant", "define"], answer: "const" },
```

```
    { question: "Which data type is used for decimal values in C?", options: ["int", "float", "char", "bool"], answer: "float" },
```

```
    { question: "Which loop runs at least once in C?", options: ["for", "while", "do-while", "None"], answer: "do-while" },
```

```
    { question: "Which header file is required for printf() function?", options: ["stdio.h", "stdlib.h", "conio.h", "math.h"], answer: "stdio.h" },
```

```
    { question: "How do you declare a pointer in C?", options: ["*ptr", "ptr*", "&ptr", "ptr&"], answer: "*ptr" },
```

```
    { question: "What is used to terminate a statement in C?", options: [";", ":", "}", "None"], answer: ";" },
```

```
    { question: "Which operator is used to access a structure member?", options: [". (dot)", "-> (arrow)", "*" (asterisk)", "& (ampersand)"], answer: "-> (arrow)" },
```

```
    { question: "Which keyword is used to define a function in C?", options: ["function", "def", "void", "define"], answer: "void" },
```


{ question: "Which function is used to take input in C?", options: ["input()", "scanf()", "cin", "read()"], answer: "scanf()" },

{ question: "Which data type is used to store a single character in C?", options: ["char", "string", "int", "float"], answer: "char" },

{ question: "Which of the following is not a loop in C?", options: ["for", "while", "do-while", "repeat"], answer: "repeat" },

{ question: "Which function is used to allocate memory dynamically in C?", options: ["malloc()", "alloc()", "new", "memalloc()"], answer: "malloc()" },

{ question: "Which function is used to free allocated memory in C?", options: ["free()", "delete", "clear()", "dealloc()"], answer: "free()" },

{ question: "Which operator is used for modulus operation in C?", options: ["%", "/", "//", "mod"], answer: "%" },

{ question: "Which loop is used when the number of iterations is fixed?", options: ["for", "while", "do-while", "switch"], answer: "for" },

{ question: "Which function is used to copy a string in C?", options: ["strcpy()", "copy()", "strcopy()", "memcpy()"], answer: "strcpy()" },

{ question: "Which of the following is not a storage class in C?", options: ["auto", "static", "register", "dynamic"], answer: "dynamic" },

{ question: "Which function is used to find the length of a string in C?", options: ["strlen()", "length()", "size()", "strlength()"], answer: "strlen()" },

{ question: "Which function is used to compare two strings in C?", options: ["strcmp()", "strcomp()", "compare()", "strncmp()"], answer: "strcmp()" },

{ question: "Which operator is used for logical AND in C?", options: ["&&", "||", "&", "|"], answer: "&&" },

{ question: "Which operator is used for logical OR in C?", options: ["&&", "||", "&", "|"], answer: "||" },

{ question: "Which operator is used for logical NOT in C?", options: ["!", "~", "not", "^"], answer: "!" },

{ question: "Which header file is required for mathematical functions in C?", options: ["math.h", "stdlib.h", "stdio.h", "conio.h"], answer: "math.h" },

{ question: "Which function is used to convert string to integer in C?", options: ["atoi()", "strtoi()", "toInt()", "convert()"], answer: "atoi()" },

{ question: "Which function is used to generate random numbers in C?", options: ["rand()", "random()", "getrand()", "srand()"], answer: "rand()" },

{ question: "Which keyword is used to return a value from a function in C?", options: ["return", "break", "exit", "end"], answer: "return" },

{ question: "Which data type is used to store large integers in C?", options: ["int", "long", "float", "double"], answer: "long" },

{ question: "Which function is used to concatenate two strings in C?", options: ["strcat()", "concat()", "append()", "join()"], answer: "strcat()" },

{ question: "Which function is used to find a character in a string in C?", options: ["strchr()", "find()", "search()", "locate()"], answer: "strchr()" },

{ question: "Which operator is used to get the address of a variable in C?", options: ["&", "*", "@", "^"], answer: "&" },

{ question: "Which keyword is used to define a structure in C?", options: ["struct", "structure", "class", "object"], answer: "struct" },

{ question: "Which function is used to open a file in C?", options: ["fopen()", "open()", "fileopen()", "openfile()"], answer: "fopen()" },

{ question: "Which function is used to close a file in C?", options: ["fclose()", "close()", "fileclose()", "endfile()"], answer: "fclose()" },

```
{ question: "Which function is used to write to a file in C?", options: ["fprintf()", "fwrite()",  
"write()", "fprint()"], answer: "fprintf()" },
```

```
{ question: "Which function is used to read from a file in C?", options: ["fscanf()", "fread()",  
"read()", "fget()"], answer: "fscanf()" },
```

```
{ question: "Which function is used to move the file pointer in C?", options: ["fseek()", "seek()",  
"move()", "pointer()"], answer: "fseek()" },
```

```
{ question: "Which keyword is used to define an enumeration in C?", options: ["enum",  
"enumerate", "list", "define"], answer: "enum" },
```

```
{ question: "Which function is used to clear the screen in C?", options: ["clrscr()", "clear()", "cls()",  
"system('cls')"], answer: "system('cls')" },
```

```
{ question: "Which function is used to get the size of a variable in C?", options: ["sizeof()", "size()",  
"length()", "memsize()"], answer: "sizeof()" },
```

```
{ question: "Which keyword is used to exit a loop in C?", options: ["break", "exit", "return",  
"stop"], answer: "break" },
```

```
{ question: "Which keyword is used to skip an iteration in a loop in C?", options: ["continue",  
"skip", "jump", "next"], answer: "continue" },
```

```
{ question: "Which function is used to pause the execution in C?", options: ["sleep()", "pause()",  
"wait()", "halt()"], answer: "sleep()" },
```

```
{ question: "Which function is used to convert lowercase to uppercase in C?", options:  
["toupper()", "touppercase()", "upper()", "toupperstr()"], answer: "toupper()" }
```

```
];
```

```
export default allQuestions;
```

-----C++ Language Question-----

```
const allQuestions =
    [{ "question": "Who invented Java programming?", "options": ["Guido van Rossum", "James Gosling", "Dennis Ritchie", "Bjarne Stroustrup"], "answer": "James Gosling" },
      { "question": "Who is known as the father of C++?", "options": ["Dennis Ritchie", "Bjarne Stroustrup", "James Gosling", "Guido van Rossum"], "answer": "Bjarne Stroustrup" },
      { "question": "What is the default access specifier for class members in C++?", "options": ["private", "public", "protected", "none"], "answer": "private" },
      { "question": "Which keyword is used to create a new object in C++?", "options": ["new", "create", "object", "malloc"], "answer": "new" },
      { "question": "Which of the following is used for exception handling in C++?", "options": ["throw", "catch", "try", "all of the above"], "answer": "all of the above" },
      { "question": "Which operator is used to allocate memory dynamically in C++?", "options": ["malloc", "alloc", "new", "calloc"], "answer": "new" },
      { "question": "What is a constructor in C++?", "options": ["A function that returns a value", "A function to initialize objects", "A function to handle errors", "A type of loop"], "answer": "A function to initialize objects" },
      { "question": "Which of the following is not a feature of object-oriented programming?", "options": ["Encapsulation", "Inheritance", "Polymorphism", "Compilation"], "answer": "Compilation" },
      { "question": "What does the 'virtual' keyword in C++ indicate?", "options": ["Indicates a function is virtual", "Indicates inheritance", "Indicates abstraction", "Indicates polymorphism"], "answer": "Indicates a function is virtual" },
      { "question": "What is the result of the expression 5 / 2 in C++?", "options": ["2", "2.5", "5", "3"], "answer": "2" },
      { "question": "What is the purpose of the 'friend' function in C++?", "options": ["To access private and protected members of a class", "To create an instance of a class", "To make a class inherit from another", "None of the above"], "answer": "To access private and protected members of a class" },
      { "question": "What does 'std' stand for in C++?", "options": ["Standard", "Stream Data", "System Data", "Standard Device"], "answer": "Standard" },
      { "question": "Which C++ keyword is used to prevent a class from being inherited?", "options": ["final", "private", "sealed", "static"], "answer": "final" },
      { "question": "Which of the following is a correct way to define a constant in C++?", "options": ["const int a = 10;", "#define a 10;", "int const a = 10;", "all of the above"], "answer": "all of the above" },
      { "question": "What is the output of the following code? int a = 10; cout << ++a;", "options": ["9", "10", "11", "error"], "answer": "11" },
      { "question": "What is the primary purpose of a destructor in C++?", "options": ["To initialize objects", "To free memory and clean up", "To create objects", "None of the above"], "answer": "To free memory and clean up" },
```

```

{ "question": "What is a pure virtual function in C++?", "options": ["A function that has no implementation", "A function that cannot be overridden", "A function that is public", "A function that throws exceptions"], "answer": "A function that has no implementation" },
{ "question": "What does the 'static' keyword do in C++?", "options": ["It prevents memory allocation", "It makes a function or variable available throughout the program", "It ensures type safety", "It makes the class private"], "answer": "It makes a function or variable available throughout the program" },
{ "question": "What is the main difference between 'struct' and 'class' in C++?", "options": ["'struct' defaults to private access", "'class' defaults to public access", "'struct' defaults to public access", "'class' defaults to private access"], "answer": "'struct' defaults to public access" },
{ "question": "What is the purpose of the 'this' pointer in C++?", "options": ["Points to the current instance of the object", "Points to the parent class", "Points to the memory address of the function", "Points to the static member"], "answer": "Points to the current instance of the object" },
{ "question": "Which of the following is the correct way to initialize a vector in C++?", "options": ["vector<int> v = {1, 2, 3};", "vector v = {1, 2, 3};", "vector<int> v(3);", "None of the above"], "answer": "vector<int> v = {1, 2, 3};" },
{ "question": "Which of the following is the correct syntax for creating a template function in C++?", "options": ["template <typename T> T func(T a) {}", "template <T> T func(T a) {}", "func<T>(T a) {}", "None of the above"], "answer": "template <typename T> T func(T a) {}" },
{ "question": "In C++, which operator is used to access a member of a class object?", "options": ["&", "->", "::", "."], "answer": "." },
{ "question": "What is the purpose of the 'continue' statement in C++?", "options": ["Exit from the loop", "Skip the current iteration and move to the next", "Pause the loop", "None of the above"], "answer": "Skip the current iteration and move to the next" },
{ "question": "What is the size of 'int' in C++?", "options": ["2 bytes", "4 bytes", "8 bytes", "Varies depending on the system"], "answer": "Varies depending on the system" },
{ "question": "Which C++ standard library header is needed to use the 'vector' container?", "options": ["<iostream>", "<vector>", "<string>", "<array>"], "answer": "<vector>" },
{ "question": "Which of the following operators can be overloaded in C++?", "options": ["+ (Addition)", "- (Subtraction)", "[] (Array indexing)", "All of the above"], "answer": "All of the above" },
{ "question": "What is a function pointer in C++?", "options": ["A pointer that stores the address of a function", "A pointer that points to an array", "A pointer that stores the return type of a function", "None of the above"], "answer": "A pointer that stores the address of a function" },
{ "question": "What is the difference between 'new' and 'malloc' in C++?", "options": ["'new' initializes the memory while 'malloc' does not", "'new' is for arrays, 'malloc' is for objects", "'new' is slower than 'malloc'", "'new' is used for dynamic memory allocation in arrays only"], "answer": "'new' initializes the memory while 'malloc' does not" },
{ "question": "What is the main purpose of a template class in C++?", "options": ["To define a class for a specific data type", "To create reusable code for multiple data types", "To store dynamic data", "To manage memory efficiently"], "answer": "To create reusable code for multiple data types" },
{ "question": "Which of the following is correct for declaring an array in C++?", "options": ["int arr[5];", "int[5] arr;", "array<int> arr(5);", "All of the above"], "answer": "int arr[5];" },
{ "question": "What does the 'sizeof' operator do in C++?", "options": ["Returns the size of a variable", "Returns the size of a data type", "Both of the above", "None of the above"], "answer": "Both of the above" },
{ "question": "Which type of loop is guaranteed to execute at least once in C++?", "options": ["while", "for", "do-while", "None of the above"], "answer": "do-while" },
{ "question": "What does the 'inline' keyword do in C++?", "options": ["It tells the compiler to insert the code of the function directly", "It tells the compiler to create a copy of the function", "It prevents function overloading", "None of the above"], "answer": "It tells the compiler to insert the code of the function directly" },

```

```

    { "question": "Which of the following is true about 'virtual' functions in C++?", "options": ["They are used for function overloading", "They can be overridden in derived classes", "They can only be called from base classes", "None of the above"], "answer": "They can be overridden in derived classes" },
    { "question": "What is a destructor in C++?", "options": ["A function that initializes the object", "A function that destroys the object", "A function that is called after main()", "A function that calls the constructor"], "answer": "A function that destroys the object" },
    { "question": "Which of the following is an example of a polymorphic behavior in C++?", "options": ["Function overloading", "Operator overloading", "Method overriding", "All of the above"], "answer": "All of the above" },
    { "question": "Which header file is required for input/output operations in C++?", "options": ["<inputoutput>", "<iostream>", "<iostream.h>", "<stdio.h>"], "answer": "<iostream>" },
    { "question": "What is the main advantage of using 'const' with pointers in C++?", "options": ["To make sure the pointer cannot change the address", "To prevent the pointer from pointing to null", "To make the pointer type safe", "To prevent the pointer value from changing"], "answer": "To make sure the pointer cannot change the address" },
    { "question": "Which C++ concept allows a derived class to inherit features from a base class?", "options": ["Abstraction", "Encapsulation", "Polymorphism", "Inheritance"], "answer": "Inheritance" },
    { "question": "What is a static function in C++?", "options": ["A function that is restricted to the class scope", "A function that can be called without an object", "A function that is inherited", "None of the above"], "answer": "A function that can be called without an object" },
    { "question": "Which of the following is used to allocate memory for an array dynamically?", "options": ["malloc", "new", "calloc", "None of the above"], "answer": "new" },
    { "question": "What is the role of 'namespace' in C++?", "options": ["To define the scope of variables and functions", "To create an alias for a class", "To define a class", "None of the above"], "answer": "To define the scope of variables and functions" },
    { "question": "Which of the following is true about multi-threading in C++?", "options": ["It improves the performance of a program", "It allows for concurrent execution of code", "It is supported by the C++ standard library", "All of the above"], "answer": "All of the above" },
    { "question": "What is the difference between 'public' and 'private' members of a class in C++?", "options": ["'public' members are accessible from outside the class, 'private' are not", "'private' members are accessible from outside the class, 'public' are not", "Both are accessible from outside the class", "None of the above"], "answer": "'public' members are accessible from outside the class, 'private' are not" }

];

```

```
export default allQuestions;
```

-----Java Language Question-----

```
const allQuestions =
[
```

```

    { "question": "Who invented Java programming?", "options": ["Guido van Rossum", "James Gosling", "Dennis Ritchie", "Bjarne Stroustrup"], "answer": "James Gosling" },
    { "question": "Which of the following is the entry point of a Java application?", "options": ["public static void start(String[] args)", "public void main(String[] args)", "public static void main(String[] args)", "void main(String args[])"], "answer": "public static void main(String[] args)" },
    { "question": "Which of these keywords is used to declare a constant in Java?", "options": ["const", "final", "static", "immutable"], "answer": "final" },
    { "question": "Which of the following is the correct way to create an object in Java?", "options": ["ClassName obj = new ClassName();", "ClassName obj = ClassName();", "new ClassName obj = new ClassName();", "None of the above"], "answer": "ClassName obj = new ClassName();" },
    { "question": "Which of the following is a Java primitive data type?", "options": ["String", "Integer", "Boolean", "Object"], "answer": "Boolean" },
    { "question": "Which method is used to start a thread in Java?", "options": ["start()", "init()", "run()", "execute()"], "answer": "start()" },
    { "question": "Which of the following statements is true about constructors in Java?", "options": ["A constructor can have a return type", "A constructor can be overloaded", "A constructor can have a return type of void", "None of the above"], "answer": "A constructor can be overloaded" },
    { "question": "What does the 'this' keyword refer to in Java?", "options": ["The current class", "The current object", "The current method", "None of the above"], "answer": "The current object" },
    { "question": "Which method is used to compare two strings in Java?", "options": ["==", "compare()", "equals()", "compareTo()"], "answer": "equals()" },
    { "question": "What is the size of an int in Java?", "options": ["2 bytes", "4 bytes", "8 bytes", "Depends on the system"], "answer": "4 bytes" },
    { "question": "Which of the following is used to handle exceptions in Java?", "options": ["throw", "throws", "try-catch", "None of the above"], "answer": "try-catch" },
    { "question": "Which of the following class is the superclass of all classes in Java?", "options": ["Object", "String", "Class", "Exception"], "answer": "Object" },
    { "question": "What is the default value of an instance variable in Java?", "options": ["null", "0", "false", "undefined"], "answer": "null" },
    { "question": "Which of the following is a valid loop construct in Java?", "options": ["while", "do-while", "for", "All of the above"], "answer": "All of the above" },
    { "question": "Which these keywords is used to prevent inheritance in Java?", "options": ["final", "static", "private", "abstract"], "answer": "final" },
    { "question": "What does the 'super' keyword refer to in Java?", "options": ["The current class", "The superclass of the current class", "The current object", "None of the above"], "answer": "The superclass of the current class" },
    { "question": "Which of the following is a valid declaration of a method in Java?", "options": ["public void method()", "void public method()", "public method void()", "void method public()"], "answer": "public void method()" },
    { "question": "What is the main purpose of the final keyword in Java?", "options": ["To declare a constant", "To prevent inheritance", "To restrict method overriding", "All of the above"], "answer": "All of the above" },
    { "question": "What will be the result of dividing an integer by 0 in Java?", "options": ["0", "Infinity", "RuntimeException", "ArithmeticException"], "answer": "ArithmeticException" },
    { "question": "Which of the following is an example of an interface in Java?", "options": ["abstract class", "interface", "implements", "extends"], "answer": "interface" },
    { "question": "Which class is used to read user input in Java?", "options": ["Scanner", "Reader", "BufferedReader", "FileReader"], "answer": "Scanner" },
    { "question": "Which method is used to get the length of a string in Java?", "options": ["length()", "size()", "getLength()", "count()"], "answer": "length()" },

```

```

    { "question": "Which of the following is used to create an array in Java?", "options": ["int[] arr = new int[10];", "int arr[] = new int(10);", "int[10] arr = new int[];", "int[] arr;"], "answer": "int[] arr = new int[10];" },
    { "question": "Which method is used to convert a string to lowercase in Java?", "options": ["toLowerCase()", "lowerCase()", "toLowerCase()", "convertToLower()"], "answer": "toLowerCase()" },
    { "question": "Which of these access modifiers allows the most accessibility in Java?", "options": ["private", "protected", "default", "public"], "answer": "public" },
    { "question": "What is the result of the following expression in Java: 3 + 2 * 2?", "options": ["7", "10", "12", "9"], "answer": "7" },
    { "question": "What does the term 'Java Virtual Machine' (JVM) refer to?", "options": ["A hardware device for running Java programs", "A virtual environment to execute Java programs", "A Java interpreter", "None of the above"], "answer": "A virtual environment to execute Java programs" },
    { "question": "Which method is used to remove an element from an ArrayList in Java?", "options": ["removeElement()", "delete()", "remove()", "pop()"], "answer": "remove()" },
    { "question": "Which of the following is the correct syntax for implementing an interface in Java?", "options": ["class MyClass implements MyInterface {}", "class MyClass extends MyInterface {}", "class MyClass interface MyInterface {}", "class MyClass : MyInterface {}"], "answer": "class MyClass implements MyInterface {}" },
    { "question": "Which type of collection is unordered and allows duplicate elements in Java?", "options": ["Set", "List", "Map", "Queue"], "answer": "List" },
    { "question": "Which class is part of the java.util package?", "options": ["ArrayList", "String", "Thread", "None of the above"], "answer": "ArrayList" },
    { "question": "What is the default size of an ArrayList in Java?", "options": ["10", "0", "5", "Depends on the number of elements"], "answer": "10" },
    { "question": "Which class is used for basic file operations in Java?", "options": ["File", "FileReader", "BufferedReader", "FileInputStream"], "answer": "File" },
    { "question": "Which of the following is not a valid method for a Thread in Java?", "options": ["run()", "sleep()", "start()", "get()"], "answer": "get()" },
    { "question": "Which of the following interfaces does the List collection implement?", "options": ["Queue", "Iterable", "Set", "None"], "answer": "Iterable" },
    { "question": "What will happen if you try to divide by 0 in Java?", "options": ["It will throw ArithmeticException", "It will return Infinity", "It will return NaN", "None of the above"], "answer": "It will throw ArithmeticException" },
    { "question": "Which method in Java is used to remove an element from a HashMap?", "options": ["removeElement()", "delete()", "remove()", "clear()"], "answer": "remove()" },
    { "question": "Which of these is not an access modifier in Java?", "options": ["public", "protected", "private", "abstract"], "answer": "abstract" },
    { "question": "Which of these is used to compare two strings in Java?", "options": ["==", "equals()", "compare()", "compareTo()"], "answer": "equals()" },
    { "question": "Which of the following is true about 'String' objects in Java?", "options": ["Strings are mutable", "Strings are immutable", "Strings can be modified", "Strings are not objects"], "answer": "Strings are immutable" },
    { "question": "Which of the following classes does not implement the Cloneable interface in Java?", "options": ["ArrayList", "String", "Object", "None of the above"], "answer": "String" },
    { "question": "Which of the following exceptions is thrown when an array index is out of bounds?", "options": ["ArrayIndexOutOfBoundsException", "NullPointerException", "IllegalArgumentException", "IOException"], "answer": "ArrayIndexOutOfBoundsException" },
    { "question": "Which method is used to start a thread in Java?", "options": ["start()", "run()", "execute()", "begin()"], "answer": "start()" },

```



```

    { "question": "Which of these operators is used to compare two objects for equality in Java?",
      "options": ["=", "==", "equals()", "is()"], "answer": "equals()" },
    { "question": "What is the default value of a reference variable in Java?", "options": ["null", "0",
      "false", "undefined"], "answer": "null" },
    { "question": "Which of these data types is not a primitive data type in Java?", "options": ["int",
      "boolean", "char", "String"], "answer": "String" },
    { "question": "Which of these statements is used to return a value from a method?", "options":
      ["return", "exit", "yield", "break"], "answer": "return" },

];

export default allQuestions;

```

-----Python Language Question-----

```

const allQuestions =
[
  {
    "question": "Who is the creator of Python programming language?",
    "options": ["Guido van Rossum", "James Gosling", "Dennis Ritchie", "Bjarne Stroustrup"],
    "answer": "Guido van Rossum"
  },
  {
    "question": "Which of the following is a Python data type?",
    "options": ["int", "char", "float", "string"],
    "answer": "int"
  },
  {
    "question": "What is the correct extension of a Python file?",
    "options": [".py", ".pt", ".python", ".p"],
    "answer": ".py"
  },
  {
    "question": "Which of the following is used to define a block of code in Python?",
    "options": ["Indentation", "Brackets", "Parentheses", "Curly braces"],
    "answer": "Indentation"
  },
  {
    "question": "What is the output of print(2**3)?",
    "options": ["8", "6", "9", "16"],
    "answer": "8"
  },
  {
    "question": "Which Python keyword is used to create a function?",
    "options": ["def", "function", "create", "fun"],

```

```

    "answer": "def"
},
{
    "question": "Which function is used to find the length of an object in Python?",
    "options": ["len()", "length()", "size()", "count()"],
    "answer": "len()"
},
{
    "question": "Which of the following is NOT a valid Python data structure?",
    "options": ["list", "tuple", "array", "set"],
    "answer": "array"
},
{
    "question": "How do you create a variable in Python?",
    "options": ["var = 10", "let var = 10", "int var = 10", "variable = 10"],
    "answer": "var = 10"
},
{
    "question": "What is the method to add an element to the end of a list?",
    "options": ["add()", "append()", "insert()", "extend()"],
    "answer": "append()"
},
{
    "question": "Which of the following is used to handle exceptions in Python?",
    "options": ["try-except", "throw-catch", "exception-handling", "catch-except"],
    "answer": "try-except"
},
{
    "question": "Which operator is used to check equality in Python?",
    "options": ["=", "==", "!=", "="],
    "answer": "=="
},
{
    "question": "What is the purpose of the 'self' keyword in Python?",
    "options": ["It refers to the instance of the class", "It is used for private variables", "It is used to define static variables", "It is used to create instances of a class"],
    "answer": "It refers to the instance of the class"
},
{
    "question": "What does the 'break' statement do in Python?",
    "options": ["Exits the current loop", "Skips the current iteration", "Exits the current function", "Returns from the current loop"],
    "answer": "Exits the current loop"
},
{
    "question": "What is a Python lambda function?",
    "options": ["An anonymous function", "A type of list", "A keyword in the 'while' loop", "A function with multiple parameters"],
    "answer": "An anonymous function"
},
{

```

```

    "question": "Which of the following is NOT a valid Python string operation?",
    "options": ["concatenation", "multiplication", "subtraction", "repetition"],
    "answer": "subtraction"
},
{
    "question": "Which of the following Python libraries is used for data analysis?",
    "options": ["NumPy", "Matplotlib", "Pandas", "SciPy"],
    "answer": "Pandas"
},
{
    "question": "How do you start a comment in Python?",
    "options": ["//", "#", "/*", "--"],
    "answer": "#"
},
{
    "question": "What is the method used to remove an item from a dictionary in Python?",
    "options": ["remove()", "pop()", "delete()", "clear()"],
    "answer": "pop()"
},
{
    "question": "What does the 'continue' statement do in Python?",
    "options": ["Skips the current iteration of the loop", "Exits the loop", "Exits the current function", "Returns from the function"],
    "answer": "Skips the current iteration of the loop"
},
{
    "question": "What does 'is' keyword in Python check?",
    "options": ["Equality of values", "Identity of objects", "Membership in a list", "Existence of a variable"],
    "answer": "Identity of objects"
},
{
    "question": "Which of the following is the correct syntax to import a module in Python?",
    "options": ["import module", "include module", "using module", "import(module)"],
    "answer": "import module"
},
{
    "question": "How do you define a class in Python?",
    "options": ["class ClassName:", "def ClassName():", "class ClassName{}", "function ClassName()"],
    "answer": "class ClassName:"
},
{
    "question": "Which of the following is NOT a valid Python loop?",
    "options": ["for", "while", "until", "break"],
    "answer": "until"
},
{
    "question": "What is the output of the following Python code: 'print(3 + 2 * 4)'?",
    "options": ["14", "11", "7", "20"],
    "answer": "11"
}

```

```

    },
    {
        "question": "Which of the following methods is used to create a new empty list in Python?",
        "options": ["list()", "[]", "new()", "create()"],
        "answer": "[]"
    },
    {
        "question": "What is the result of the following operation: '4 // 2'",
        "options": ["2", "4", "1", "None"],
        "answer": "2"
    },
    {
        "question": "What does the 'pass' statement do in Python?",
        "options": ["It terminates a function", "It performs a no-operation action", "It throws an exception", "It is used in the for loop"],
        "answer": "It performs a no-operation action"
    },
    {
        "question": "What is the purpose of the 'zip' function in Python?",
        "options": ["Combines two or more iterables", "Zips files into one", "Creates a new dictionary", "Splits iterables into tuples"],
        "answer": "Combines two or more iterables"
    },
    {
        "question": "Which of the following is the correct way to write a Python decorator?",
        "options": ["@decorator", "def decorator()", "decorate@", "function @decorator"],
        "answer": "@decorator"
    },
    {
        "question": "What is the purpose of the 'yield' keyword in Python?",
        "options": ["To stop a function", "To create a generator function", "To return a value", "To throw an exception"],
        "answer": "To create a generator function"
    },
    {
        "question": "Which of the following is NOT a valid Python exception?",
        "options": ["TypeError", "IndexError", "MemoryError", "DivisionZeroError"],
        "answer": "DivisionZeroError"
    },
    {
        "question": "Which of the following is used to create a virtual environment in Python?",
        "options": ["virtualenv", "venv", "env", "python-virtual"],
        "answer": "venv"
    },
    {
        "question": "What is a module in Python?",
        "options": ["A file containing Python code", "A built-in function", "A class", "An instance of a function"],
        "answer": "A file containing Python code"
    },
    {

```

```

    "question": "What is the output of 'print('hello' + 3)' in Python?",
    "options": ["hello3", "error", "hello", "None"],
    "answer": "error"
  },
  {
    "question": "Which of the following is used to create a new dictionary in Python?",
    "options": [{"{}", "["], "dict()", "new()"},
    "answer": "{}"
  },
  {
    "question": "How do you declare a global variable in Python?",
    "options": ["global variable", "global x", "x = global", "global"],
    "answer": "global x"
  },
  {
    "question": "Which of the following methods is used to sort a list in Python?",
    "options": ["sort()", "sorted()", "order()", "arrange()"],
    "answer": "sort()"
  },
  {
    "question": "What is the purpose of the 'in' keyword in Python?",
    "options": ["To check membership", "To assign values", "To iterate over a sequence", "To define
a function"],
    "answer": "To check membership"
  },
  {
    "question": "What is the purpose of the 'del' keyword in Python?",
    "options": ["To delete a variable", "To delete an object", "To delete a class", "To delete a
dictionary key"],
    "answer": "To delete a variable"
  },
  {
    "question": "How do you access the first element of a list 'lst' in Python?",
    "options": ["lst[1]", "lst[0]", "lst(-1)", "lst.first()"],
    "answer": "lst[0]"
  },
  {
    "question": "What is the output of 'print('Python'.lower())'?",
    "options": ["python", "Python", "PYTHON", "Error"],
    "answer": "python"
  },
  {
    "question": "Which of the following is used to define an anonymous function in Python?",
    "options": ["def", "lambda", "function", "func"],
    "answer": "lambda"
  },
  {
    "question": "Which of the following is a mutable data type in Python?",
    "options": ["tuple", "list", "set", "string"],
    "answer": "list"
  },

```

```

{
  "question": "Which of the following Python functions returns the highest value?",
  "options": ["max()", "highest()", "top()", "biggest()"],
  "answer": "max()"
},
{
  "question": "Which Python method is used to remove an element from a list?",
  "options": ["remove()", "pop()", "delete()", "clear()"],
  "answer": "pop()"
},
{
  "question": "What is the result of '3 == 3' in Python?",
  "options": ["True", "False", "None", "Error"],
  "answer": "True"
}

];

export default allQuestions;

```

-----NodeJs Language Question-----

```

const allQuestions =
[
  {
    "question": "What is Node.js?",
    "options": ["A JavaScript library", "A JavaScript runtime", "A web browser", "A programming language"],
    "answer": "A JavaScript runtime"
  },
  {
    "question": "Who created Node.js?",
    "options": ["Ryan Dahl", "Brendan Eich", "Guido van Rossum", "James Gosling"],
    "answer": "Ryan Dahl"
  },
  {
    "question": "Which engine does Node.js use?",
    "options": ["SpiderMonkey", "Chakra", "V8", "JavaScriptCore"],
    "answer": "V8"
  },
  {
    "question": "Which module is used to create a web server in Node.js?",
    "options": ["fs", "http", "url", "querystring"],
    "answer": "http"
  }
]

```

```

    },
    {
      "question": "Which command is used to install a Node.js package globally?",
      "options": ["npm install <package>", "npm install -g <package>", "npm add <package>", "node
install <package>"],
      "answer": "npm install -g <package>"
    },
    {
      "question": "What is the purpose of the package.json file in a Node.js project?",
      "options": ["To define project metadata", "To list dependencies", "To specify scripts", "All of
the above"],
      "answer": "All of the above"
    },
    {
      "question": "How can you handle asynchronous operations in Node.js?",
      "options": ["Callbacks", "Promises", "Async/Await", "All of the above"],
      "answer": "All of the above"
    },
    {
      "question": "Which module helps in performing file operations in Node.js?",
      "options": ["fs", "http", "os", "path"],
      "answer": "fs"
    },
    {
      "question": "What is the purpose of the 'os' module in Node.js?",
      "options": ["To interact with the operating system", "To manage HTTP requests", "To handle
file uploads", "To create UI components"],
      "answer": "To interact with the operating system"
    },
    {
      "question": "How can you create an HTTP server in Node.js?",
      "options": ["Using the 'http' module", "Using the 'fs' module", "Using the 'os' module", "Using
the 'stream' module"],
      "answer": "Using the 'http' module"
    },
    {
      "question": "What does NPM stand for?",
      "options": ["Node Package Manager", "Node Program Module", "Node Project Management",
"New Package Manager"],
      "answer": "Node Package Manager"
    },
    {
      "question": "What is the purpose of the 'path' module?",
      "options": ["To handle file paths", "To manage HTTP requests", "To interact with the OS", "To
create web servers"],
      "answer": "To handle file paths"
    },
    {
      "question": "What is a callback function in Node.js?",
      "options": ["A function passed as an argument to another function", "A function that returns a
value", "A function that handles errors", "A function that creates a server"],

```

```

    "answer": "A function passed as an argument to another function"
  },
  {
    "question": "What is the purpose of 'require()' in Node.js?",
    "options": ["To include modules", "To define variables", "To create functions", "To handle errors"],
    "answer": "To include modules"
  },
  {
    "question": "What is the event loop in Node.js?",
    "options": ["A mechanism for handling asynchronous operations", "A loop for creating web servers", "A loop for file operations", "A loop for managing variables"],
    "answer": "A mechanism for handling asynchronous operations"
  },
  {
    "question": "What is the purpose of 'module.exports'?",
    "options": ["To export functions and variables from a module", "To import modules", "To create variables", "To handle errors"],
    "answer": "To export functions and variables from a module"
  },
  {
    "question": "What is the purpose of 'process' object in Node.js?",
    "options": ["Provides information about the current Node.js process", "Manages file operations", "Creates web servers", "Handles HTTP requests"],
    "answer": "Provides information about the current Node.js process"
  },
  {
    "question": "What is the purpose of 'Buffer' class in Node.js?",
    "options": ["To handle binary data", "To manage text data", "To handle HTTP requests", "To create web servers"],
    "answer": "To handle binary data"
  },
  {
    "question": "What is the purpose of 'stream' module in Node.js?",
    "options": ["To handle streaming data", "To manage file operations", "To create web servers", "To handle HTTP requests"],
    "answer": "To handle streaming data"
  },
  {
    "question": "What is middleware in Express.js?",
    "options": ["Functions that have access to the request object, the response object, and the next middleware function in the application's request-response cycle", "Functions that create web servers", "Functions that handle file operations", "Functions that manage variables"],
    "answer": "Functions that have access to the request object, the response object, and the next middleware function in the application's request-response cycle"
  },
  {
    "question": "What is the purpose of Express.js?",
    "options": ["To create web applications and APIs", "To handle file operations", "To manage variables", "To create operating systems"],
    "answer": "To create web applications and APIs"
  }

```



```

    },
    {
      "question": "What is a RESTful API?",
      "options": ["An architectural style for building web services", "A method for creating web servers", "A method for handling file operations", "A method for managing variables"],
      "answer": "An architectural style for building web services"
    },
    {
      "question": "What is the purpose of 'EventEmitter' in Node.js?",
      "options": ["To handle events and emit events", "To create web servers", "To manage file operations", "To handle HTTP requests"],
      "answer": "To handle events and emit events"
    },
    {
      "question": "What is the purpose of 'cluster' module in Node.js?",
      "options": ["To create child processes for load balancing", "To manage file operations", "To create web servers", "To handle HTTP requests"],
      "answer": "To create child processes for load balancing"
    },
    {
      "question": "What is the purpose of '.gitignore' file?",
      "options": ["To specify files and directories that should be ignored by Git", "To manage file operations", "To create web servers", "To handle HTTP requests"],
      "answer": "To specify files and directories that should be ignored by Git"
    },
    {
      "question": "What is the purpose of 'async/await' in Node.js?",
      "options": ["To handle asynchronous operations in a synchronous style", "To manage file operations", "To create web servers", "To handle HTTP requests"],
      "answer": "To handle asynchronous operations in a synchronous style"
    },
    {
      "question": "What is the purpose of 'promises' in Node.js?",
      "options": ["To handle asynchronous operations", "To manage file operations", "To create web servers", "To handle HTTP requests"],
      "answer": "To handle asynchronous operations"
    },
    {
      "question": "What is the purpose of the 'URL' module?",
      "options": ["To parse and manipulate URLs", "To manage file operations", "To create web servers", "To handle HTTP requests"],
      "answer": "To parse and manipulate URLs"
    },
    {
      "question": "What is the purpose of the 'querystring' module?",
      "options": ["To parse and stringify query strings", "To manage file operations", "To create web servers", "To handle HTTP requests"],
      "answer": "To parse and stringify query strings"
    },
    {
      "question": "What is the purpose of 'child_process' module?",

```

```

    "options": ["To create child processes", "To manage file operations", "To create web servers",
    "To handle HTTP requests"],
    "answer": "To create child processes"
  },
  {
    "question": "What is REPL in Node.js?",
    "options": ["Read-Eval-Print Loop", "Read-Execute-Print Line", "Recursive Eval Print Loop",
    "Remote Execution Programming Language"],
    "answer": "Read-Eval-Print Loop"
  }
];

export default allQuestions;

```

-----ReactJs Language Question-----

```

const allQuestions =
[
  {
    "question": "What is React.js?",
    "options": ["A JavaScript library for building user interfaces", "A server-side scripting
language", "A database management system", "A CSS framework"],
    "answer": "A JavaScript library for building user interfaces"
  },
  {
    "question": "Who developed React.js?",
    "options": ["Google", "Microsoft", "Facebook (Meta)", "Amazon"],
    "answer": "Facebook (Meta)"
  },
  {
    "question": "What is JSX in React?",
    "options": ["A syntax extension for JavaScript", "A styling language", "A database query
language", "A server-side framework"],
    "answer": "A syntax extension for JavaScript"
  },
  {
    "question": "What are components in React?",
    "options": ["Reusable UI elements", "Data storage units", "Server-side scripts", "Database
tables"],
    "answer": "Reusable UI elements"
  },
  {

```

```

    "question": "What is the purpose of the 'state' in React?",
    "options": ["To store mutable data that affects the component's rendering", "To store static data", "To manage CSS styles", "To handle server requests"],
    "answer": "To store mutable data that affects the component's rendering"
  },
  {
    "question": "What is the purpose of 'props' in React?",
    "options": ["To pass data from parent to child components", "To store component state", "To manage CSS styles", "To handle server requests"],
    "answer": "To pass data from parent to child components"
  },
  {
    "question": "What is the virtual DOM in React?",
    "options": ["An in-memory representation of the actual DOM", "A physical DOM implementation", "A database for storing DOM elements", "A server-side DOM rendering engine"],
    "answer": "An in-memory representation of the actual DOM"
  },
  {
    "question": "What is the purpose of 'useEffect' hook in React?",
    "options": ["To perform side effects in functional components", "To manage component state", "To handle server requests", "To style components"],
    "answer": "To perform side effects in functional components"
  },
  {
    "question": "What is the purpose of 'useState' hook in React?",
    "options": ["To manage state in functional components", "To perform side effects", "To handle server requests", "To style components"],
    "answer": "To manage state in functional components"
  },
  {
    "question": "What is the purpose of 'React Router'?",
    "options": ["To handle navigation in React applications", "To manage component state", "To handle server requests", "To style components"],
    "answer": "To handle navigation in React applications"
  },
  {
    "question": "What is the purpose of 'keys' in React lists?",
    "options": ["To uniquely identify list items", "To style list items", "To manage list state", "To handle list events"],
    "answer": "To uniquely identify list items"
  },
  {
    "question": "What is the purpose of 'context' in React?",
    "options": ["To share state between components without passing props manually at every level", "To manage component state", "To handle server requests", "To style components"],
    "answer": "To share state between components without passing props manually at every level"
  },
  {
    "question": "What is the purpose of 'refs' in React?",

```

```

      "options": ["To access DOM nodes or React elements created in the render method", "To
manage component state", "To handle server requests", "To style components"],
      "answer": "To access DOM nodes or React elements created in the render method"
    },
    {
      "question": "What is the purpose of 'componentDidMount' lifecycle method?",
      "options": ["To perform actions after the component is inserted into the DOM", "To manage
component state", "To handle server requests", "To style components"],
      "answer": "To perform actions after the component is inserted into the DOM"
    },
    {
      "question": "What is the purpose of 'componentDidUpdate' lifecycle method?",
      "options": ["To perform actions after the component is updated", "To manage component
state", "To handle server requests", "To style components"],
      "answer": "To perform actions after the component is updated"
    },
    {
      "question": "What is the purpose of 'componentWillUnmount' lifecycle method?",
      "options": ["To perform cleanup actions before the component is removed from the DOM",
"To manage component state", "To handle server requests", "To style components"],
      "answer": "To perform cleanup actions before the component is removed from the DOM"
    },
    {
      "question": "What is the purpose of 'Redux' in React?",
      "options": ["To manage application state in a predictable way", "To manage component
state", "To handle server requests", "To style components"],
      "answer": "To manage application state in a predictable way"
    },
    {
      "question": "What is the purpose of 'Hooks' in React?",
      "options": ["To use state and other React features without writing a class", "To manage
component state", "To handle server requests", "To style components"],
      "answer": "To use state and other React features without writing a class"
    },
    {
      "question": "What is the purpose of 'React Fragments'?",
      "options": ["To group a list of children without adding extra nodes to the DOM", "To manage
component state", "To handle server requests", "To style components"],
      "answer": "To group a list of children without adding extra nodes to the DOM"
    },
    {
      "question": "What is the purpose of 'Higher-Order Components (HOCs)' in React?",
      "options": ["To reuse component logic", "To manage component state", "To handle server
requests", "To style components"],
      "answer": "To reuse component logic"
    },
    {
      "question": "What is the purpose of 'React Memo'?",
      "options": ["To memoize functional components", "To manage component state", "To handle
server requests", "To style components"],
      "answer": "To memoize functional components"
    }
  ]

```

```

    },
    {
      "question": "What is the purpose of 'useMemo' hook?",
      "options": ["To memoize expensive calculations", "To manage component state", "To handle
server requests", "To style components"],
      "answer": "To memoize expensive calculations"
    },
    {
      "question": "What is the purpose of 'useCallback' hook?",
      "options": ["To memoize callback functions", "To manage component state", "To handle
server requests", "To style components"],
      "answer": "To memoize callback functions"
    },
    {
      "question": "What is the purpose of 'forwardRef' in React?",
      "options": ["To pass a ref from a parent component to a child component", "To manage
component state", "To handle server requests", "To style components"],
      "answer": "To pass a ref from a parent component to a child component"
    },
    {
      "question": "What is the purpose of 'React.lazy'?",
      "options": ["To lazy load components", "To manage component state", "To handle server
requests", "To style components"],
      "answer": "To lazy load components"
    },
    {
      "question": "What is the purpose of 'Suspense' in React?",
      "options": ["To display a fallback UI while lazy-loaded components are loading", "To manage
component state", "To handle server requests", "To style components"],
      "answer": "To display a fallback UI while lazy-loaded components are loading"
    },
    {
      "question": "What is the purpose of 'Custom Hooks'?",
      "options": ["To extract component logic into reusable functions", "To manage component
state", "To handle server requests", "To style components"],
      "answer": "To extract component logic into reusable functions"
    },
    {
      "question": "What is the purpose of 'React Testing Library'?",
      "options": ["To test React components", "To manage component state", "To handle server
requests", "To style components"],
      "answer": "To test React components"
    },
    {
      "question": "What is the purpose of 'React Developer Tools'?",
      "options": ["To debug React applications", "To manage component state", "To handle server
requests", "To style components"],
      "answer": "To debug React applications"
    },
  ];

```

```
export default allQuestions;
```

----- C Queries -----

```
<?php include 'auth_check.php'; ?>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>C Language Quiz</title>
```

```
  <link rel="icon" type="image/png" href="favicon-32x32.png">
```

```
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">
```

```
  <style>
```

```
    /* your CSS styles here (same as before) */
```

```
    body {
```

```
      margin: 0;
```

```
      padding: 0;
```

```
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
```

```
      background: linear-gradient(to right, #e0c3fc, #8ec5fc);
```

```
      min-height: 100vh;
```

```
    }
```

```
    .quiz-container {
```

```
      max-width: 800px;
```

```
      margin: 40px auto;
```

```
      background: white;
```

```
      border-radius: 20px;
```

```
      box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
```

```
      padding: 30px;
```

```
      transition: transform 0.4s ease;
```

```
    }
```

```
    h1 {
```

```
      text-align: center;
```

```
      color: #6a1b9a;
```

```
      font-size: 36px;
```

```
      margin-bottom: 20px;
```

```
    }
```

```
    .option {
```

```
      background: #f2f2f2;
```

```
      padding: 15px;
```

```
border-radius: 10px;
margin: 10px 0;
cursor: pointer;
transition: all 0.3s ease;
font-size: 18px;
}

.option:hover {
  background: #d1c4e9;
}

.correct {
  background-color: #4caf50 !important;
  color: white;
}

.wrong {
  background-color: #f44336 !important;
  color: white;
}

#next-button {
  display: none;
  background: #6a1b9a;
  color: white;
  padding: 12px 25px;
  border: none;
  border-radius: 10px;
  margin-top: 20px;
  cursor: pointer;
  font-size: 16px;
}

#next-button:hover {
  background: #7b1fa2;
}

.show {
  display: none;
  background: white;
  padding: 25px;
  border-radius: 15px;
  max-width: 600px;
  margin: 20px auto;
  text-align: left;
  box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
}

.show input {
  width: 100%;
  padding: 10px;
```

```

        margin: 8px 0;
        border: none;
        background: #f5f5f5;
        border-radius: 8px;
    }

    #progress {
        height: 8px;
        background: #ddd;
        border-radius: 10px;
        overflow: hidden;
        margin-bottom: 20px;
    }

    #progress-bar {
        height: 100%;
        background: #6a1b9a;
        width: 0%;
        transition: width 0.4s ease;
    }

    .red-countdown {
        color: red;
        font-weight: bold;
    }
}
</style>
</head>

<body>
<script>
    // Storing PHP session values into localStorage
    localStorage.setItem("username", "<?php echo $name; ?>");
    localStorage.setItem("roll_number", "<?php echo $roll; ?>");
</script>

<div class="quiz-container">
    <h1>C Language Quiz</h1>
    <div style="text-align:center; font-size: 20px; margin-bottom: 10px;">
        Time Left: <span id="countdown">20</span> seconds
    </div>
    <div id="progress-container">
        <div id="question-label">Question 1 of 25</div>
    </div>
    <div id="progress">
        <div id="progress-bar"></div>
    </div>
    <div id="question"></div>
    <div id="options"></div>
    <button id="next-button" onclick="nextQuestion()">Next</button>
</div>

```



```

<div class="show" id="result-box">
  <h2>Quiz Result</h2>
  <input type="text" id="studentName" readonly>
  <input type="text" id="subjectName" readonly>
  <input type="text" id="totalQuestions" readonly>
  <input type="text" id="attempted" readonly>
  <input type="text" id="correct" readonly>
  <input type="text" id="wrong" readonly>
  <input type="text" id="percentage" readonly>
  <input type="text" id="score" readonly>
</div>

<script type="module">
  import allQuestions from './c.js';

  let questions = getRandomQuestions(allQuestions, 25);
  let currentQuestionIndex = 0;
  let score = 0;
  let attempted = 0;
  let correct = 0;
  let wrong = 0;
  let timeLeft = 20;
  let timer;
  let subjectName = "C Programming";
  const studentName = localStorage.getItem("username") || "Unknown Student";
  const Roll_Number = localStorage.getItem("roll_number") || "NA";

  function loadQuestion() {
    clearInterval(timer);
    const current = questions[currentQuestionIndex];
    document.getElementById('question').innerHTML = `<strong>Q${currentQuestionIndex +
1}</strong> ${current.question}`;
    const optionsContainer = document.getElementById('options');
    optionsContainer.innerHTML = "";

    updateProgress();

    [...current.options].sort(() => Math.random() - 0.5).forEach(opt => {
      const div = document.createElement('div');
      div.className = 'option';
      div.textContent = opt;
      div.onclick = () => selectOption(div, opt);
      optionsContainer.appendChild(div);
    });

    document.getElementById('next-button').style.display = 'none';
    startTimer();
  }

  function startTimer() {
    clearInterval(timer);
  }

```

```

timeLeft = 20;

const countdownElement = document.getElementById("countdown");
countdownElement.textContent = timeLeft;
countdownElement.classList.remove('red-countdown');

timer = setInterval(() => {
  timeLeft--;

  if (timeLeft <= 5 && timeLeft > 0) {
    countdownElement.classList.add('red-countdown');
  }

  if (timeLeft <= 0) {
    timeLeft = 0;
    clearInterval(timer);
    countdownElement.textContent = "0";

    document.querySelectorAll('.option').forEach(opt => {
      opt.style.pointerEvents = 'none';
      if (opt.textContent === questions[currentQuestionIndex].answer) {
        opt.classList.add('correct');
      }
    });
    wrong++;
    document.getElementById('next-button').style.display = 'block';
  } else {
    countdownElement.textContent = timeLeft;
  }
}, 1000);
}

function getRandomQuestions(all, count = 25) {
  const used = new Set();
  const out = [];
  while (out.length < count && used.size < all.length) {
    const i = Math.floor(Math.random() * all.length);
    if (!used.has(i)) {
      out.push(all[i]);
      used.add(i);
    }
  }
  return out;
}

function selectOption(el, value) {
  clearInterval(timer);
  const current = questions[currentQuestionIndex];
  attempted++;
  document.querySelectorAll('.option').forEach(opt => opt.style.pointerEvents = 'none');

```

```

    if (value === current.answer) {
        correct++;
        score += 10;
        el.classList.add('correct');
    } else {
        wrong++;
        el.classList.add('wrong');
        document.querySelectorAll('.option').forEach(opt => {
            if (opt.textContent === current.answer) opt.classList.add('correct');
        });
    }
    document.getElementById('next-button').style.display = 'block';
}

function nextQuestion() {
    currentQuestionIndex++;
    if (currentQuestionIndex < questions.length) loadQuestion();
    else {
        showResult();
        saveResult();
    }
}

function updateProgress() {
    const percent = (currentQuestionIndex / questions.length) * 100;
    document.getElementById('progress-bar').style.width = `${percent}%`;
    document.getElementById('question-label').textContent = `Question ${currentQuestionIndex
+ 1} of ${questions.length}`;
}

function showResult() {
    const percent = (correct / attempted) * 100;
    document.getElementById('studentName').value = `Student Name: ${studentName}`;
    document.getElementById('subjectName').value = `Subject: ${subjectName}`;
    document.getElementById('totalQuestions').value = `Total Questions: ${questions.length}`;
    document.getElementById('attempted').value = `Attempted: ${attempted}`;
    document.getElementById('correct').value = `Correct: ${correct}`;
    document.getElementById('wrong').value = `Wrong: ${wrong}`;
    document.getElementById('percentage').value = `Percentage: ${percent.toFixed(2)}%`;
    document.getElementById('score').value = `Score: ${score}`;
    document.querySelector('.quiz-container').style.display = 'none';
    document.getElementById('result-box').style.display = 'block';
}

function saveResult() {
    const percentage = (correct / questions.length) * 100;
    fetch('save_result.php', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify({
            student_name: studentName,

```

```

        subject_name: subjectName,
        total_questions: questions.length,
        attempted,
        correct,
        wrong,
        percentage: percentage.toFixed(2),
        total_score: score,
        Roll_Number
    })
    });
}

window.nextQuestion = nextQuestion;
loadQuestion();
</script>
</body>

</html>

```

----- C++ Queries -----

```

<?php include 'auth_check.php'; ?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>C++ Language Quiz</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.4.0/css/all.min.css">
    <style>
        /* your CSS styles here (same as before) */
        body {
            margin: 0;
            padding: 0;
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            background: linear-gradient(to right, #e0c3fc, #8ec5fc);
            min-height: 100vh;

```

```
}

.quiz-container {
  max-width: 800px;
  margin: 40px auto;
  background: white;
  border-radius: 20px;
  box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
  padding: 30px;
  transition: transform 0.4s ease;
}

h1 {
  text-align: center;
  color: #6a1b9a;
  font-size: 36px;
  margin-bottom: 20px;
}

.option {
  background: #f2f2f2;
  padding: 15px;
  border-radius: 10px;
  margin: 10px 0;
  cursor: pointer;
  transition: all 0.3s ease;
  font-size: 18px;
}

.option:hover {
  background: #d1c4e9;
}

.correct {
  background-color: #4caf50 !important;
  color: white;
}

.wrong {
  background-color: #f44336 !important;
  color: white;
}

#next-button {
  display: none;
  background: #6a1b9a;
  color: white;
  padding: 12px 25px;
  border: none;
  border-radius: 10px;
  margin-top: 20px;
}
```

```
        cursor: pointer;
        font-size: 16px;
    }

    #next-button:hover {
        background: #7b1fa2;
    }

    .show {
        display: none;
        background: white;
        padding: 25px;
        border-radius: 15px;
        max-width: 600px;
        margin: 20px auto;
        text-align: left;
        box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
    }

    .show input {
        width: 100%;
        padding: 10px;
        margin: 8px 0;
        border: none;
        background: #f5f5f5;
        border-radius: 8px;
    }

    #progress {
        height: 8px;
        background: #ddd;
        border-radius: 10px;
        overflow: hidden;
        margin-bottom: 20px;
    }

    #progress-bar {
        height: 100%;
        background: #6a1b9a;
        width: 0%;
        transition: width 0.4s ease;
    }

    .red-countdown {
        color: red;
        font-weight: bold;
    }
</style>
</head>

<body>
```

```

<script>
  // Storing PHP session values into localStorage
  localStorage.setItem("username", "<?php echo $name; ?>");
  localStorage.setItem("roll_number", "<?php echo $roll; ?>");
</script>

<div class="quiz-container">
  <h1>C++ Language Quiz</h1>
  <div style="text-align:center; font-size: 20px; margin-bottom: 10px;">
    Time Left: <span id="countdown">20</span> seconds
  </div>
  <div id="progress-container">
    <div id="question-label">Question 1 of 25</div>
  </div>
  <div id="progress">
    <div id="progress-bar"></div>
  </div>
  <div id="question"></div>
  <div id="options"></div>
  <button id="next-button" onclick="nextQuestion()">Next</button>
</div>

<div class="show" id="result-box">
  <div style="text-align: right;">
    <a href="subjectchoose.php">
      <i class="fas fa-times" style="cursor: pointer; color: red; font-size: 24px;"></i>
    </a>
  </div>
  <h2>Quiz Result</h2>
  <input type="text" id="studentName" readonly>
  <input type="text" id="subjectName" readonly>
  <input type="text" id="totalQuestions" readonly>
  <input type="text" id="attempted" readonly>
  <input type="text" id="correct" readonly>
  <input type="text" id="wrong" readonly>
  <input type="text" id="percentage" readonly>
  <input type="text" id="score" readonly>
</div>

<script type="module">
  import allQuestions from './cpp.js';

  let questions = getRandomQuestions(allQuestions, 25);
  let currentQuestionIndex = 0;
  let score = 0;
  let attempted = 0;
  let correct = 0;
  let wrong = 0;
  let timeLeft = 20;
  let timer;
  let subjectName = "C++ Programming";

```

```

const studentName = localStorage.getItem("username") || "Unknown Student";
const Roll_Number = localStorage.getItem("roll_number") || "NA";

function loadQuestion() {
  clearInterval(timer);
  const current = questions[currentQuestionIndex];
  document.getElementById('question').innerHTML = `<strong>Q${currentQuestionIndex +
1}</strong> ${current.question}`;
  const optionsContainer = document.getElementById('options');
  optionsContainer.innerHTML = "";

  updateProgress();

  [...current.options].sort(() => Math.random() - 0.5).forEach(opt => {
    const div = document.createElement('div');
    div.className = 'option';
    div.textContent = opt;
    div.onclick = () => selectOption(div, opt);
    optionsContainer.appendChild(div);
  });

  document.getElementById('next-button').style.display = 'none';
  startTimer();
}

function startTimer() {
  clearInterval(timer);
  timeLeft = 20;

  const countdownElement = document.getElementById("countdown");
  countdownElement.textContent = timeLeft;
  countdownElement.classList.remove('red-countdown');

  timer = setInterval(() => {
    timeLeft--;

    if (timeLeft <= 5 && timeLeft > 0) {
      countdownElement.classList.add('red-countdown');
    }

    if (timeLeft <= 0) {
      timeLeft = 0;
      clearInterval(timer);
      countdownElement.textContent = "0";

      document.querySelectorAll('.option').forEach(opt => {
        opt.style.pointerEvents = 'none';
        if (opt.textContent === questions[currentQuestionIndex].answer) {
          opt.classList.add('correct');
        }
      });
    }
  });
}

```



```

        wrong++;
        document.getElementById('next-button').style.display = 'block';
    } else {
        countdownElement.textContent = timeLeft;
    }
}, 1000);
}

function getRandomQuestions(all, count = 25) {
    const used = new Set();
    const out = [];
    while (out.length < count && used.size < all.length) {
        const i = Math.floor(Math.random() * all.length);
        if (!used.has(i)) {
            out.push(all[i]);
            used.add(i);
        }
    }
    return out;
}

function selectOption(el, value) {
    clearInterval(timer);
    const current = questions[currentQuestionIndex];
    attempted++;
    document.querySelectorAll('.option').forEach(opt => opt.style.pointerEvents = 'none');

    if (value === current.answer) {
        correct++;
        score += 10;
        el.classList.add('correct');
    } else {
        wrong++;
        el.classList.add('wrong');
        document.querySelectorAll('.option').forEach(opt => {
            if (opt.textContent === current.answer) opt.classList.add('correct');
        });
    }
    document.getElementById('next-button').style.display = 'block';
}

function nextQuestion() {
    currentQuestionIndex++;
    if (currentQuestionIndex < questions.length) loadQuestion();
    else {
        saveResult();
        showResult();
    }
}

function updateProgress() {

```

```

        const percent = (currentQuestionIndex / questions.length) * 100;
        document.getElementById('progress-bar').style.width = `${percent}%`;
        document.getElementById('question-label').textContent = `Question ${currentQuestionIndex
+ 1} of ${questions.length}`;
    }

    function showResult() {
        const percent = (correct / attempted) * 100;
        document.getElementById('studentName').value = `Student Name: ${studentName}`;
        document.getElementById('subjectName').value = `Subject: ${subjectName}`;
        document.getElementById('totalQuestions').value = `Total Questions: ${questions.length}`;
        document.getElementById('attempted').value = `Attempted: ${attempted}`;
        document.getElementById('correct').value = `Correct: ${correct}`;
        document.getElementById('wrong').value = `Wrong: ${wrong}`;
        document.getElementById('percentage').value = `Percentage: ${percent.toFixed(2)}%`;
        document.getElementById('score').value = `Score: ${score}`;
        document.querySelector('.quiz-container').style.display = 'none';
        document.getElementById('result-box').style.display = 'block';
    }

    function saveResult() {
        const percentage = (correct / questions.length) * 100;
        fetch('save_result.php', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify({
                student_name: studentName,
                subject_name: subjectName,
                total_questions: questions.length,
                attempted,
                correct,
                wrong,
                percentage: percentage.toFixed(2),
                total_score: score,
                Roll_Number
            })
        });
    }

    window.nextQuestion = nextQuestion;
    loadQuestion();
    function closes() {
    }
</script>
</body>

</html>

```

----- JAVA Queries -----

```
<?php include 'auth_check.php'; ?>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>JAVA Language Quiz</title>
```

```
  <link rel="icon" type="image/png" href="favicon-32x32.png">
```

```
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">
```

```
  <style>
```

```
    /* your CSS styles here (same as before) */
```

```
    body {
```

```
      margin: 0;
```

```
      padding: 0;
```

```
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
```

```
      background: linear-gradient(to right, #e0c3fc, #8ec5fc);
```

```
      min-height: 100vh;
```

```
    }
```

```
    .quiz-container {
```

```
      max-width: 800px;
```

```
      margin: 40px auto;
```

```
      background: white;
```

```
      border-radius: 20px;
```

```
      box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
```

```
      padding: 30px;
```

```
      transition: transform 0.4s ease;
```

```
    }
```

```
    h1 {
```

```
      text-align: center;
```

```
      color: #6a1b9a;
```

```
      font-size: 36px;
```

```
      margin-bottom: 20px;
```

```
    }
```

```
    .option {
```

```
background: #f2f2f2;
padding: 15px;
border-radius: 10px;
margin: 10px 0;
cursor: pointer;
transition: all 0.3s ease;
font-size: 18px;
}

.option:hover {
background: #d1c4e9;
}

.correct {
background-color: #4caf50 !important;
color: white;
}

.wrong {
background-color: #f44336 !important;
color: white;
}

#next-button {
display: none;
background: #6a1b9a;
color: white;
padding: 12px 25px;
border: none;
border-radius: 10px;
margin-top: 20px;
cursor: pointer;
font-size: 16px;
}

#next-button:hover {
background: #7b1fa2;
}

.show {
display: none;
background: white;
padding: 25px;
border-radius: 15px;
max-width: 600px;
margin: 20px auto;
text-align: left;
box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
}

.show input {
```

```

        width: 100%;
        padding: 10px;
        margin: 8px 0;
        border: none;
        background: #f5f5f5;
        border-radius: 8px;
    }

    #progress {
        height: 8px;
        background: #ddd;
        border-radius: 10px;
        overflow: hidden;
        margin-bottom: 20px;
    }

    #progress-bar {
        height: 100%;
        background: #6a1b9a;
        width: 0%;
        transition: width 0.4s ease;
    }

    .red-countdown {
        color: red;
        font-weight: bold;
    }
}
</style>
</head>

<body>
    <script>
        // Storing PHP session values into localStorage
        localStorage.setItem("username", "<?php echo $name; ?>");
        localStorage.setItem("roll_number", "<?php echo $roll; ?>");
    </script>

    <div class="quiz-container">
        <h1>JAVA Language Quiz</h1>
        <div style="text-align:center; font-size: 20px; margin-bottom: 10px;">
            Time Left: <span id="countdown">20</span> seconds
        </div>
        <div id="progress-container">
            <div id="question-label">Question 1 of 25</div>
        </div>
        <div id="progress">
            <div id="progress-bar"></div>
        </div>
        <div id="question"></div>
        <div id="options"></div>
        <button id="next-button" onclick="nextQuestion()">Next</button>
    </div>

```

```
</div>
```

```
<div class="show" id="result-box">
  <h2>Quiz Result</h2>
  <input type="text" id="studentName" readonly>
  <input type="text" id="subjectName" readonly>
  <input type="text" id="totalQuestions" readonly>
  <input type="text" id="attempted" readonly>
  <input type="text" id="correct" readonly>
  <input type="text" id="wrong" readonly>
  <input type="text" id="percentage" readonly>
  <input type="text" id="score" readonly>
</div>
```

```
<script type="module">
  import allQuestions from './JAVA.js';

  let questions = getRandomQuestions(allQuestions, 25);
  let currentQuestionIndex = 0;
  let score = 0;
  let attempted = 0;
  let correct = 0;
  let wrong = 0;
  let timeLeft = 20;
  let timer;
  let subjectName = "JAVA Programming";
  const studentName = localStorage.getItem("username") || "Unknown Student";
  const Roll_Number = localStorage.getItem("roll_number") || "NA";

  function loadQuestion() {
    clearInterval(timer);
    const current = questions[currentQuestionIndex];
    document.getElementById('question').innerHTML = `<strong>Q${currentQuestionIndex +
1}</strong> ${current.question}`;
    const optionsContainer = document.getElementById('options');
    optionsContainer.innerHTML = "";

    updateProgress();

    [...current.options].sort(() => Math.random() - 0.5).forEach(opt => {
      const div = document.createElement('div');
      div.className = 'option';
      div.textContent = opt;
      div.onclick = () => selectOption(div, opt);
      optionsContainer.appendChild(div);
    });

    document.getElementById('next-button').style.display = 'none';
    startTimer();
  }
}
```

```

function startTimer() {
  clearInterval(timer);
  timeLeft = 20;

  const countdownElement = document.getElementById("countdown");
  countdownElement.textContent = timeLeft;
  countdownElement.classList.remove('red-countdown');

  timer = setInterval(() => {
    timeLeft--;

    if (timeLeft <= 5 && timeLeft > 0) {
      countdownElement.classList.add('red-countdown');
    }

    if (timeLeft <= 0) {
      timeLeft = 0;
      clearInterval(timer);
      countdownElement.textContent = "0";

      document.querySelectorAll('.option').forEach(opt => {
        opt.style.pointerEvents = 'none';
        if (opt.textContent === questions[currentQuestionIndex].answer) {
          opt.classList.add('correct');
        }
      });
      wrong++;
      document.getElementById('next-button').style.display = 'block';
    } else {
      countdownElement.textContent = timeLeft;
    }
  }, 1000);
}

function getRandomQuestions(all, count = 25) {
  const used = new Set();
  const out = [];
  while (out.length < count && used.size < all.length) {
    const i = Math.floor(Math.random() * all.length);
    if (!used.has(i)) {
      out.push(all[i]);
      used.add(i);
    }
  }
  return out;
}

function selectOption(el, value) {
  clearInterval(timer);
  const current = questions[currentQuestionIndex];
  attempted++;

```

```

document.querySelectorAll('.option').forEach(opt => opt.style.pointerEvents = 'none');

if (value === current.answer) {
    correct++;
    score += 10;
    el.classList.add('correct');
} else {
    wrong++;
    el.classList.add('wrong');
    document.querySelectorAll('.option').forEach(opt => {
        if (opt.textContent === current.answer) opt.classList.add('correct');
    });
}
document.getElementById('next-button').style.display = 'block';
}

function nextQuestion() {
    currentQuestionIndex++;
    if (currentQuestionIndex < questions.length) loadQuestion();
    else {
        showResult();
        saveResult();
    }
}

function updateProgress() {
    const percent = (currentQuestionIndex / questions.length) * 100;
    document.getElementById('progress-bar').style.width = `${percent}%`;
    document.getElementById('question-label').textContent = `Question ${currentQuestionIndex
+ 1} of ${questions.length}`;
}

function showResult() {
    const percent = (correct / attempted) * 100;
    document.getElementById('studentName').value = `Student Name: ${studentName}`;
    document.getElementById('subjectName').value = `Subject: ${subjectName}`;
    document.getElementById('totalQuestions').value = `Total Questions: ${questions.length}`;
    document.getElementById('attempted').value = `Attempted: ${attempted}`;
    document.getElementById('correct').value = `Correct: ${correct}`;
    document.getElementById('wrong').value = `Wrong: ${wrong}`;
    document.getElementById('percentage').value = `Percentage: ${percent.toFixed(2)}%`;
    document.getElementById('score').value = `Score: ${score}`;
    document.querySelector('.quiz-container').style.display = 'none';
    document.getElementById('result-box').style.display = 'block';
}

function saveResult() {
    const percentage = (correct / questions.length) * 100;
    fetch('save_result.php', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },

```



```

        body: JSON.stringify({
            student_name: studentName,
            subject_name: subjectName,
            total_questions: questions.length,
            attempted,
            correct,
            wrong,
            percentage: percentage.toFixed(2),
            total_score: score,
            Roll_Number
        })
    });
}

window.nextQuestion = nextQuestion;
loadQuestion();
</script>
</body>

</html>

```

----- Python Queries -----

```

<?php include 'auth_check.php'; ?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PYTHON Language Quiz</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.4.0/css/all.min.css">
    <style>
        body {
            margin: 0;

```

```
padding: 0;
font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
background: linear-gradient(to right, #e0c3fc, #8ec5fc);
min-height: 100vh;
}
```

```
.quiz-container {
  max-width: 800px;
  margin: 40px auto;
  background: white;
  border-radius: 20px;
  box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
  padding: 30px;
  transition: transform 0.4s ease;
}
```

```
h1 {
  text-align: center;
  color: #6a1b9a;
  font-size: 36px;
  margin-bottom: 20px;
}
```

```
.option {
  background: #f2f2f2;
  padding: 15px;
  border-radius: 10px;
  margin: 10px 0;
  cursor: pointer;
  transition: all 0.3s ease;
  font-size: 18px;
}
```

```
.option:hover {
  background: #d1c4e9;
}
```

```
.correct {
  background-color: #4caf50 !important;
  color: white;
}
```

```
.wrong {
  background-color: #f44336 !important;
  color: white;
}
```

```
#next-button {
  display: none;
  background: #6a1b9a;
  color: white;
}
```

```
padding: 12px 25px;
border: none;
border-radius: 10px;
margin-top: 20px;
cursor: pointer;
font-size: 16px;
}

#next-button:hover {
  background: #7b1fa2;
}

.show {
  display: none;
  background: white;
  padding: 25px;
  border-radius: 15px;
  max-width: 600px;
  margin: 20px auto;
  text-align: left;
  box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
}

.show input {
  width: 100%;
  padding: 10px;
  margin: 8px 0;
  border: none;
  background: #f5f5f5;
  border-radius: 8px;
}

#progress {
  height: 8px;
  background: #ddd;
  border-radius: 10px;
  overflow: hidden;
  margin-bottom: 20px;
}

#progress-bar {
  height: 100%;
  background: #6a1b9a;
  width: 0%;
  transition: width 0.4s ease;
}

.red-countdown {
  color: red;
  font-weight: bold;
}
```

```

</style>
</head>

<body>
  <script>
    // Storing PHP session values into localStorage
    localStorage.setItem("username", "<?php echo $name; ?>");
    localStorage.setItem("roll_number", "<?php echo $roll; ?>");
  </script>

  <div class="quiz-container">
    <h1>PYTHON Language Quiz</h1>
    <div style="text-align:center; font-size: 20px; margin-bottom: 10px;">
      Time Left: <span id="countdown">20</span> seconds
    </div>
    <div id="progress-container">
      <div id="question-label">Question 1 of 25</div>
    </div>
    <div id="progress">
      <div id="progress-bar"></div>
    </div>
    <div id="question"></div>
    <div id="options"></div>
    <button id="next-button" onclick="nextQuestion()">Next</button>
  </div>

  <div class="show" id="result-box">
    <h2>Quiz Result</h2>
    <input type="text" id="studentName" readonly>
    <input type="text" id="subjectName" readonly>
    <input type="text" id="totalQuestions" readonly>
    <input type="text" id="attempted" readonly>
    <input type="text" id="correct" readonly>
    <input type="text" id="wrong" readonly>
    <input type="text" id="percentage" readonly>
    <input type="text" id="score" readonly>
  </div>

  <script type="module">
    import allQuestions from './Python.js';

    let questions = getRandomQuestions(allQuestions, 25);
    let currentQuestionIndex = 0;
    let score = 0;
    let attempted = 0;
    let correct = 0;
    let wrong = 0;
    let timeLeft = 20;
    let timer;
    let subjectName = "Python Programming";
    const studentName = localStorage.getItem("username") || "Unknown Student";
  </script>

```

```

const Roll_Number = localStorage.getItem("roll_number") || "NA";

function loadQuestion() {
  clearInterval(timer);
  const current = questions[currentQuestionIndex];
  document.getElementById('question').innerHTML = `<strong>Q${currentQuestionIndex +
1}</strong> ${current.question}`;
  const optionsContainer = document.getElementById('options');
  optionsContainer.innerHTML = "";

  updateProgress();

  [...current.options].sort(() => Math.random() - 0.5).forEach(opt => {
    const div = document.createElement('div');
    div.className = 'option';
    div.textContent = opt;
    div.onclick = () => selectOption(div, opt);
    optionsContainer.appendChild(div);
  });

  document.getElementById('next-button').style.display = 'none';
  startTimer();
}

function startTimer() {
  clearInterval(timer);
  timeLeft = 20;

  const countdownElement = document.getElementById("countdown");
  countdownElement.textContent = timeLeft;
  countdownElement.classList.remove('red-countdown');

  timer = setInterval(() => {
    timeLeft--;

    if (timeLeft <= 5 && timeLeft > 0) {
      countdownElement.classList.add('red-countdown');
    }

    if (timeLeft <= 0) {
      timeLeft = 0;
      clearInterval(timer);
      countdownElement.textContent = "0";

      document.querySelectorAll('.option').forEach(opt => {
        opt.style.pointerEvents = 'none';
        if (opt.textContent === questions[currentQuestionIndex].answer) {
          opt.classList.add('correct');
        }
      });
      wrong++;
    }
  }, 1000);
}

```

```

        document.getElementById('next-button').style.display = 'block';
    } else {
        countdownElement.textContent = timeLeft;
    }
}, 1000);
}

function getRandomQuestions(all, count = 25) {
    const used = new Set();
    const out = [];
    while (out.length < count && used.size < all.length) {
        const i = Math.floor(Math.random() * all.length);
        if (!used.has(i)) {
            out.push(all[i]);
            used.add(i);
        }
    }
    return out;
}

function selectOption(el, value) {
    clearInterval(timer);
    const current = questions[currentQuestionIndex];
    attempted++;
    document.querySelectorAll('.option').forEach(opt => opt.style.pointerEvents = 'none');

    if (value === current.answer) {
        correct++;
        score += 10;
        el.classList.add('correct');
    } else {
        wrong++;
        el.classList.add('wrong');
        document.querySelectorAll('.option').forEach(opt => {
            if (opt.textContent === current.answer) opt.classList.add('correct');
        });
    }
    document.getElementById('next-button').style.display = 'block';
}

function nextQuestion() {
    currentQuestionIndex++;
    if (currentQuestionIndex < questions.length) loadQuestion();
    else {
        showResult();
        saveResult();
    }
}

function updateProgress() {
    const percent = (currentQuestionIndex / questions.length) * 100;

```

```

        document.getElementById('progress-bar').style.width = `${percent}%`;
        document.getElementById('question-label').textContent = `Question ${currentQuestionIndex
+ 1} of ${questions.length}`;
    }

    function showResult() {
        const percent = (correct / attempted) * 100;
        document.getElementById('studentName').value = `Student Name: ${studentName}`;
        document.getElementById('subjectName').value = `Subject: ${subjectName}`;
        document.getElementById('totalQuestions').value = `Total Questions: ${questions.length}`;
        document.getElementById('attempted').value = `Attempted: ${attempted}`;
        document.getElementById('correct').value = `Correct: ${correct}`;
        document.getElementById('wrong').value = `Wrong: ${wrong}`;
        document.getElementById('percentage').value = `Percentage: ${percent.toFixed(2)}%`;
        document.getElementById('score').value = `Score: ${score}`;
        document.querySelector('.quiz-container').style.display = 'none';
        document.getElementById('result-box').style.display = 'block';
    }

    function saveResult() {
        const percentage = (correct / questions.length) * 100;
        fetch('save_result.php', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify({
                student_name: studentName,
                subject_name: subjectName,
                total_questions: questions.length,
                attempted,
                correct,
                wrong,
                percentage: percentage.toFixed(2),
                total_score: score,
                Roll_Number
            })
        });
    }

    window.nextQuestion = nextQuestion;
    loadQuestion();
</script>
</body>

</html>

```

----- NodeJs Queries -----

```
<?php include 'auth_check.php'; ?>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>NodeJS Language Quiz</title>
```

```
  <link rel="icon" type="image/png" href="favicon-32x32.png">
```

```
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">
```

```
  <style>
```

```
    /* your CSS styles here (same as before) */
```

```
    body {
```

```
      margin: 0;
```

```
      padding: 0;
```

```
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
```

```
      background: linear-gradient(to right, #e0c3fc, #8ec5fc);
```

```
      min-height: 100vh;
```

```
    }
```

```
    .quiz-container {
```

```
      max-width: 800px;
```

```
      margin: 40px auto;
```

```
      background: white;
```

```
      border-radius: 20px;
```

```
      box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
```

```
      padding: 30px;
```

```
      transition: transform 0.4s ease;
```

```
    }
```

```
    h1 {
```

```
      text-align: center;
```

```
      color: #6a1b9a;
```

```
      font-size: 36px;
```

```
      margin-bottom: 20px;
```

```
    }
```

```
    .option {
```

```
      background: #f2f2f2;
```

```
      padding: 15px;
```



```
border-radius: 10px;
margin: 10px 0;
cursor: pointer;
transition: all 0.3s ease;
font-size: 18px;
}

.option:hover {
  background: #d1c4e9;
}

.correct {
  background-color: #4caf50 !important;
  color: white;
}

.wrong {
  background-color: #f44336 !important;
  color: white;
}

#next-button {
  display: none;
  background: #6a1b9a;
  color: white;
  padding: 12px 25px;
  border: none;
  border-radius: 10px;
  margin-top: 20px;
  cursor: pointer;
  font-size: 16px;
}

#next-button:hover {
  background: #7b1fa2;
}

.show {
  display: none;
  background: white;
  padding: 25px;
  border-radius: 15px;
  max-width: 600px;
  margin: 20px auto;
  text-align: left;
  box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
}

.show input {
  width: 100%;
  padding: 10px;
```

```

        margin: 8px 0;
        border: none;
        background: #f5f5f5;
        border-radius: 8px;
    }

    #progress {
        height: 8px;
        background: #ddd;
        border-radius: 10px;
        overflow: hidden;
        margin-bottom: 20px;
    }

    #progress-bar {
        height: 100%;
        background: #6a1b9a;
        width: 0%;
        transition: width 0.4s ease;
    }

    .red-countdown {
        color: red;
        font-weight: bold;
    }
}
</style>
</head>

<body>
<script>
    // Storing PHP session values into localStorage
    localStorage.setItem("username", "<?php echo $name; ?>");
    localStorage.setItem("roll_number", "<?php echo $roll; ?>");
</script>

<div class="quiz-container">
    <h1>NodeJS Language Quiz</h1>
    <div style="text-align:center; font-size: 20px; margin-bottom: 10px;">
        Time Left: <span id="countdown">20</span> seconds
    </div>
    <div id="progress-container">
        <div id="question-label">Question 1 of 25</div>
    </div>
    <div id="progress">
        <div id="progress-bar"></div>
    </div>
    <div id="question"></div>
    <div id="options"></div>
    <button id="next-button" onclick="nextQuestion()">Next</button>
</div>

```

```

<div class="show" id="result-box">
  <h2>Quiz Result</h2>
  <input type="text" id="studentName" readonly>
  <input type="text" id="subjectName" readonly>
  <input type="text" id="totalQuestions" readonly>
  <input type="text" id="attempted" readonly>
  <input type="text" id="correct" readonly>
  <input type="text" id="wrong" readonly>
  <input type="text" id="percentage" readonly>
  <input type="text" id="score" readonly>
</div>

<script type="module">
  import allQuestions from './NodeJS.js';

  let questions = getRandomQuestions(allQuestions, 25);
  let currentQuestionIndex = 0;
  let score = 0;
  let attempted = 0;
  let correct = 0;
  let wrong = 0;
  let timeLeft = 20;
  let timer;
  let subjectName = "NodeJS Programming";
  const studentName = localStorage.getItem("username") || "Unknown Student";
  const Roll_Number = localStorage.getItem("roll_number") || "NA";

  function loadQuestion() {
    clearInterval(timer);
    const current = questions[currentQuestionIndex];
    document.getElementById('question').innerHTML = `<strong>Q${currentQuestionIndex +
1}</strong> ${current.question}`;
    const optionsContainer = document.getElementById('options');
    optionsContainer.innerHTML = "";

    updateProgress();

    [...current.options].sort(() => Math.random() - 0.5).forEach(opt => {
      const div = document.createElement('div');
      div.className = 'option';
      div.textContent = opt;
      div.onclick = () => selectOption(div, opt);
      optionsContainer.appendChild(div);
    });

    document.getElementById('next-button').style.display = 'none';
    startTimer();
  }

  function startTimer() {
    clearInterval(timer);
  }

```

```

timeLeft = 20;

const countdownElement = document.getElementById("countdown");
countdownElement.textContent = timeLeft;
countdownElement.classList.remove('red-countdown');

timer = setInterval(() => {
  timeLeft--;

  if (timeLeft <= 5 && timeLeft > 0) {
    countdownElement.classList.add('red-countdown');
  }

  if (timeLeft <= 0) {
    timeLeft = 0;
    clearInterval(timer);
    countdownElement.textContent = "0";

    document.querySelectorAll('.option').forEach(opt => {
      opt.style.pointerEvents = 'none';
      if (opt.textContent === questions[currentQuestionIndex].answer) {
        opt.classList.add('correct');
      }
    });
    wrong++;
    document.getElementById('next-button').style.display = 'block';
  } else {
    countdownElement.textContent = timeLeft;
  }
}, 1000);
}

function getRandomQuestions(all, count = 25) {
  const used = new Set();
  const out = [];
  while (out.length < count && used.size < all.length) {
    const i = Math.floor(Math.random() * all.length);
    if (!used.has(i)) {
      out.push(all[i]);
      used.add(i);
    }
  }
  return out;
}

function selectOption(el, value) {
  clearInterval(timer);
  const current = questions[currentQuestionIndex];
  attempted++;
  document.querySelectorAll('.option').forEach(opt => opt.style.pointerEvents = 'none');

```

```

    if (value === current.answer) {
        correct++;
        score += 10;
        el.classList.add('correct');
    } else {
        wrong++;
        el.classList.add('wrong');
        document.querySelectorAll('.option').forEach(opt => {
            if (opt.textContent === current.answer) opt.classList.add('correct');
        });
    }
    document.getElementById('next-button').style.display = 'block';
}

function nextQuestion() {
    currentQuestionIndex++;
    if (currentQuestionIndex < questions.length) loadQuestion();
    else {
        showResult();
        saveResult();
    }
}

function updateProgress() {
    const percent = (currentQuestionIndex / questions.length) * 100;
    document.getElementById('progress-bar').style.width = `${percent}%`;
    document.getElementById('question-label').textContent = `Question ${currentQuestionIndex
+ 1} of ${questions.length}`;
}

function showResult() {
    const percent = (correct / attempted) * 100;
    document.getElementById('studentName').value = `Student Name: ${studentName}`;
    document.getElementById('subjectName').value = `Subject: ${subjectName}`;
    document.getElementById('totalQuestions').value = `Total Questions: ${questions.length}`;
    document.getElementById('attempted').value = `Attempted: ${attempted}`;
    document.getElementById('correct').value = `Correct: ${correct}`;
    document.getElementById('wrong').value = `Wrong: ${wrong}`;
    document.getElementById('percentage').value = `Percentage: ${percent.toFixed(2)}%`;
    document.getElementById('score').value = `Score: ${score}`;
    document.querySelector('.quiz-container').style.display = 'none';
    document.getElementById('result-box').style.display = 'block';
}

function saveResult() {
    const percentage = (correct / questions.length) * 100;
    fetch('save_result.php', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify({
            student_name: studentName,

```

```

        subject_name: subjectName,
        total_questions: questions.length,
        attempted,
        correct,
        wrong,
        percentage: percentage.toFixed(2),
        total_score: score,
        Roll_Number
    })
    });
}

window.nextQuestion = nextQuestion;
loadQuestion();
</script>
</body>

</html>

```

----- ReactJs Queries -----

```

<?php include 'auth_check.php'; ?>

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>ReactJS Language Quiz</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.4.0/css/all.min.css">
    <style>
        /* your CSS styles here (same as before) */
        body {
            margin: 0;
            padding: 0;
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            background: linear-gradient(to right, #e0c3fc, #8ec5fc);
            min-height: 100vh;

```

```
}

.quiz-container {
  max-width: 800px;
  margin: 40px auto;
  background: white;
  border-radius: 20px;
  box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
  padding: 30px;
  transition: transform 0.4s ease;
}

h1 {
  text-align: center;
  color: #6a1b9a;
  font-size: 36px;
  margin-bottom: 20px;
}

.option {
  background: #f2f2f2;
  padding: 15px;
  border-radius: 10px;
  margin: 10px 0;
  cursor: pointer;
  transition: all 0.3s ease;
  font-size: 18px;
}

.option:hover {
  background: #d1c4e9;
}

.correct {
  background-color: #4caf50 !important;
  color: white;
}

.wrong {
  background-color: #f44336 !important;
  color: white;
}

#next-button {
  display: none;
  background: #6a1b9a;
  color: white;
  padding: 12px 25px;
  border: none;
  border-radius: 10px;
  margin-top: 20px;
}
```

```
        cursor: pointer;
        font-size: 16px;
    }

    #next-button:hover {
        background: #7b1fa2;
    }

    .show {
        display: none;
        background: white;
        padding: 25px;
        border-radius: 15px;
        max-width: 600px;
        margin: 20px auto;
        text-align: left;
        box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
    }

    .show input {
        width: 100%;
        padding: 10px;
        margin: 8px 0;
        border: none;
        background: #f5f5f5;
        border-radius: 8px;
    }

    #progress {
        height: 8px;
        background: #ddd;
        border-radius: 10px;
        overflow: hidden;
        margin-bottom: 20px;
    }

    #progress-bar {
        height: 100%;
        background: #6a1b9a;
        width: 0%;
        transition: width 0.4s ease;
    }

    .red-countdown {
        color: red;
        font-weight: bold;
    }
</style>
</head>

<body>
```



```

<script>
  // Storing PHP session values into localStorage
  localStorage.setItem("username", "<?php echo $name; ?>");
  localStorage.setItem("roll_number", "<?php echo $roll; ?>");
</script>

<div class="quiz-container">
  <h1>ReactJS Language Quiz</h1>
  <div style="text-align:center; font-size: 20px; margin-bottom: 10px;">
    Time Left: <span id="countdown">20</span> seconds
  </div>
  <div id="progress-container">
    <div id="question-label">Question 1 of 25</div>
  </div>
  <div id="progress">
    <div id="progress-bar"></div>
  </div>
  <div id="question"></div>
  <div id="options"></div>
  <button id="next-button" onclick="nextQuestion()">Next</button>
</div>

<div class="show" id="result-box">
  <h2>Quiz Result</h2>
  <input type="text" id="studentName" readonly>
  <input type="text" id="subjectName" readonly>
  <input type="text" id="totalQuestions" readonly>
  <input type="text" id="attempted" readonly>
  <input type="text" id="correct" readonly>
  <input type="text" id="wrong" readonly>
  <input type="text" id="percentage" readonly>
  <input type="text" id="score" readonly>
</div>

<script type="module">
  import allQuestions from './Reactjs.js';

  let questions = getRandomQuestions(allQuestions, 25);
  let currentQuestionIndex = 0;
  let score = 0;
  let attempted = 0;
  let correct = 0;
  let wrong = 0;
  let timeLeft = 20;
  let timer;
  let subjectName = "ReactJS Programming";
  const studentName = localStorage.getItem("username") || "Unknown Student";
  const Roll_Number = localStorage.getItem("roll_number") || "NA";

  function loadQuestion() {
    clearInterval(timer);
  }

```

```

    const current = questions[currentQuestionIndex];
    document.getElementById('question').innerHTML = `<strong>Q${currentQuestionIndex +
1}</strong> ${current.question}`;
    const optionsContainer = document.getElementById('options');
    optionsContainer.innerHTML = "";

    updateProgress();

    [...current.options].sort(() => Math.random() - 0.5).forEach(opt => {
        const div = document.createElement('div');
        div.className = 'option';
        div.textContent = opt;
        div.onclick = () => selectOption(div, opt);
        optionsContainer.appendChild(div);
    });

    document.getElementById('next-button').style.display = 'none';
    startTimer();
}

function startTimer() {
    clearInterval(timer);
    timeLeft = 20;

    const countdownElement = document.getElementById("countdown");
    countdownElement.textContent = timeLeft;
    countdownElement.classList.remove('red-countdown');

    timer = setInterval(() => {
        timeLeft--;

        if (timeLeft <= 5 && timeLeft > 0) {
            countdownElement.classList.add('red-countdown');
        }

        if (timeLeft <= 0) {
            timeLeft = 0;
            clearInterval(timer);
            countdownElement.textContent = "0";

            document.querySelectorAll('.option').forEach(opt => {
                opt.style.pointerEvents = 'none';
                if (opt.textContent === questions[currentQuestionIndex].answer) {
                    opt.classList.add('correct');
                }
            });
            wrong++;
            document.getElementById('next-button').style.display = 'block';
        } else {
            countdownElement.textContent = timeLeft;
        }
    });
}

```

```

    }, 1000);
}

function getRandomQuestions(all, count = 25) {
  const used = new Set();
  const out = [];
  while (out.length < count && used.size < all.length) {
    const i = Math.floor(Math.random() * all.length);
    if (!used.has(i)) {
      out.push(all[i]);
      used.add(i);
    }
  }
  return out;
}

function selectOption(el, value) {
  clearInterval(timer);
  const current = questions[currentQuestionIndex];
  attempted++;
  document.querySelectorAll('.option').forEach(opt => opt.style.pointerEvents = 'none');

  if (value === current.answer) {
    correct++;
    score += 10;
    el.classList.add('correct');
  } else {
    wrong++;
    el.classList.add('wrong');
    document.querySelectorAll('.option').forEach(opt => {
      if (opt.textContent === current.answer) opt.classList.add('correct');
    });
  }
  document.getElementById('next-button').style.display = 'block';
}

function nextQuestion() {
  currentQuestionIndex++;
  if (currentQuestionIndex < questions.length) loadQuestion();
  else {
    showResult();
    saveResult();
  }
}

function updateProgress() {
  const percent = (currentQuestionIndex / questions.length) * 100;
  document.getElementById('progress-bar').style.width = `${percent}%`;
  document.getElementById('question-label').textContent = `Question ${currentQuestionIndex
+ 1} of ${questions.length}`;
}

```

```

function showResult() {
    const percent = (correct / attempted) * 100;
    document.getElementById('studentName').value = `Student Name: ${studentName}`;
    document.getElementById('subjectName').value = `Subject: ${subjectName}`;
    document.getElementById('totalQuestions').value = `Total Questions: ${questions.length}`;
    document.getElementById('attempted').value = `Attempted: ${attempted}`;
    document.getElementById('correct').value = `Correct: ${correct}`;
    document.getElementById('wrong').value = `Wrong: ${wrong}`;
    document.getElementById('percentage').value = `Percentage: ${percent.toFixed(2)}%`;
    document.getElementById('score').value = `Score: ${score}`;
    document.querySelector('.quiz-container').style.display = 'none';
    document.getElementById('result-box').style.display = 'block';
}

function saveResult() {
    const percentage = (correct / questions.length) * 100;
    fetch('save_result.php', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify({
            student_name: studentName,
            subject_name: subjectName,
            total_questions: questions.length,
            attempted,
            correct,
            wrong,
            percentage: percentage.toFixed(2),
            total_score: score,
            Roll_Number
        })
    });
}

window.nextQuestion = nextQuestion;
loadQuestion();
</script>
</body>

</html>

```

-----Certificate-----

```

<?php
$name = "John Doe";
$subject = isset($_GET['subject']) ? $_GET['subject'] : "Unknown Subject";
$date = date("F d, Y");
?>

```

```

<!DOCTYPE html>

```

```
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>Certificate of Achievement</title>
  <link rel="icon" type="image/png" href="favicon-32x32.png" />
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet" />
  <script
src="https://cdnjs.cloudflare.com/ajax/libs/html2pdf.js/0.10.1/html2pdf.bundle.min.js"></script>
  <style>
    body {
      background-color: #e6f0fa;
      font-family: 'Georgia', serif;
      padding: 20px;
    }

    .certificate {
      background: #fff;
      padding: 60px 50px;
      border: 10px double #333;
      width: 100%;
      max-width: 900px;
      margin: 30px auto;
      box-shadow: 0 0 25px rgba(0, 0, 0, 0.15);
      position: relative;
    }

    .certificate h1 {
      font-size: 42px;
      font-weight: bold;
      text-transform: uppercase;
      margin-bottom: 10px;
      color: #2c3e50;
    }

    .certificate h3 {
      font-size: 22px;
      margin-bottom: 30px;
      color: #555;
    }

    .logo {
      text-align: center;
      margin-bottom: 30px;
    }

    .logo img {
      width: 120px;
    }
  </style>
</head>

<body>
  <div class="certificate">
    <h1>Certificate of Achievement</h1>
    <h3>Awarded to</h3>
    <div class="logo">
      <img alt="Logo" data-bbox="100 800 200 900"/>
    </div>
  </div>
</body>
</html>
```

```
.recipient-name {
  text-transform: uppercase;
  font-size: 32px;
  font-weight: bold;
  color: #1a73e8;
  margin: 20px 0;
}

.certificate-body {
  font-size: 18px;
  color: #333;
  line-height: 1.6;
  margin-top: 20px;
}

.footer {
  display: flex;
  justify-content: space-between;
  margin-top: 60px;
  font-size: 16px;
  font-weight: bold;
}

#nameInput {
  text-transform: uppercase;
  border: none;
  border-bottom: 2px solid #1a73e8;
  font-size: 24px;
  text-align: center;
  font-weight: bold;
  color: #1a73e8;
  width: 60%;
  margin: 10px auto;
  display: block;
}

#pdf {
  display: block;
  margin: 20px auto;
}

@media print {

  #nameInput,
  #pdf {
    display: none !important;
  }
}

.logo-circle {
```

```

width: 110px;
height: 110px;
background: linear-gradient(to right, #28a745, #1e7e34);
border-radius: 50%;
display: flex;
align-items: center;
justify-content: center;
color: #fff;
font-size: 36px;
font-weight: bold;
margin: 0 auto 20px;
}
</style>
</head>

<body>

  <button class="btn btn-success" id="pdf" onclick="downloadPDF()" disabled>Download
  PDF</button>

  <div class="certificate text-center" id="certificate">
    <div class="logo-circle">OES</div>

    <h1 style="color:green;">Certificate of Achievement</h1>
    <h3 style="color:green;">Presented by Online Examination System</h3>

    <p>This is to proudly certify that</p>

    <div class="recipient-name" id="nameDisplay">_____</div>
    <input type="text" id="nameInput" placeholder="Enter your full name" oninput="updateName()"
    />

    <div class="certificate-body">
      has successfully completed the course on <strong><?= htmlspecialchars($subject)
      ?></strong><br />
      demonstrating exceptional knowledge and commitment.
    </div>

    <div class="footer">
      <div>Authorized by:<br>OES Management</div>
      <div>Issued on:<br><?= $date ?></div>
    </div>
  </div>

  <script>
  function updateName() {
    const input = document.getElementById('nameInput');
    const display = document.getElementById('nameDisplay');
    const btn = document.getElementById('pdf');
    display.textContent = input.value.trim() || "_____";
    btn.disabled = input.value.trim() === "";
  }
  </script>

```

```

    }

    function downloadPDF() {
      const input = document.getElementById('nameInput');
      if (input.value.trim() === "") {
        alert("Please enter your name.");
        return;
      }
      input.style.display = "none";

      html2pdf()
        .set({
          margin: 0.5,
          filename: 'certificate.pdf',
          image: { type: 'jpeg', quality: 0.98 },
          html2canvas: { scale: 4 },
          jsPDF: { unit: 'in', format: 'a4', orientation: 'portrait' }
        })
        .from(document.getElementById("certificate"))
        .save()
        .then(() => {
          input.style.display = "block";
        });
    }
  </script>

</body>

</html>

```

----- Result-----

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Quiz Results</title>
  <link rel="icon" type="image/png" href="favicon-32x32.png">

  <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.8.2/css/all.css" />
  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.0.0/dist/css/bootstrap.min.css"
  crossorigin="anonymous" />

  <script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf-
autotable/3.5.21/jspdf.plugin.autotable.min.js"></script>

```



```
<style>
body {
  background: #f5f7fa;
}

.form-box {
  max-width: 500px;
  margin: 100px auto;
  background: white;
  padding: 30px;
  border-radius: 15px;
  box-shadow: 0 8px 20px rgba(0, 0, 0, 0.1);
  animation: slideUp 1s ease;
  position: relative;
}

@keyframes slideUp {
  from {
    transform: translateY(50px);
    opacity: 0;
  }

  to {
    transform: translateY(0);
    opacity: 1;
  }
}

#resultsContainer {
  margin-top: 30px;
  animation: fadeIn 1s ease;
}

@keyframes fadeIn {
  from {
    opacity: 0;
    transform: scale(0.95);
  }

  to {
    opacity: 1;
    transform: scale(1);
  }
}

.close-icon {
  position: absolute;
  top: 15px;
  right: 20px;
  cursor: pointer;
}
```

```

.table-responsive {
  animation: fadeIn 1s ease;
}

#pdf {
  position: fixed;
  top: 50px;
  right: 30px;
  z-index: 1000;

}
</style>
</head>

<body>

<div class="form-box" id="tex">
  <i class="fas fa-times fa-lg text-danger close-icon" onclick="closeOpenedWindow()"></i>
  <h4 class="mb-4">Enter Roll Number
    <a href="user_information.php" class="btn btn-sm btn-outline-secondary ml-2">
      <i class="fas fa-question-circle"></i>
    </a>
  </h4>
  <form>
    <div class="form-group">
      <input type="text" name="Roll_Number" minlength="12" maxlength="12" class="form-control"
        placeholder="12-digit Roll Number" onkeypress="return isNumberKey(event)"
oninput="validateInput()"/>
    </div>
    <button type="button" id="get" class="btn btn-primary btn-block" onclick="showResults()"
disabled>
      Get Result
    </button>
  </form>
</div>

<div id="qu" class="text-center hidden mt-4" style="display: none;">
  <h2>Quiz Results</h2>
</div>
<i id="closeAfterResult" class="fas fa-times fa-lg text-danger close-icon" style="display:
none;" onclick="closeOpenedWindow()"></i>

<button onclick="generatePDF()" id="pdf" class="hidden btn btn-success" style="display: none;">
  <i class="fas fa-file-download"></i> Download PDF
</button>

<div id="resultsContainer" class="container mt-4"></div>

<script>

```

```

function isNumberKey(evt) {
    var charCode = evt.which ? evt.which : evt.keyCode;
    return !(charCode < 48 || charCode > 57);
}

function validateInput() {
    let rollNumber = document.querySelector("input[name='Roll_Number']").value;
    document.getElementById("get").disabled = rollNumber.length !== 12;
}

let resultsData = [];

function showResults() {
    document.getElementById('tex').style.display = "none";
    document.getElementById('pdf').style.display = "block";
    document.getElementById('qu').style.display = "block";
    document.getElementById('closeAfterResult').style.display = "block";

    let rollNumber = document.querySelector("input[name='Roll_Number']").value;

    fetch("view_result.php", {
        method: "POST",
        headers: {
            "Content-Type": "application/x-www-form-urlencoded"
        },
        body: "Roll_Number=" + encodeURIComponent(rollNumber)
    })
    .then(response => response.json())
    .then(data => {
        if (!data.success) {
            alert(data.message);
            window.location.href = "subjectchoose.php";
            return;
        }

        let resultsContainer = document.getElementById("resultsContainer");
        resultsContainer.innerHTML = `<h4>Results for
<strong>${data.results[0].student_name}</strong></h4>`;

        let resultHTML = `
<div class="table-responsive mt-3">
<table class="table table-bordered table-striped">
<thead class="thead-dark">
<tr>
<th>#</th>
<th>Name</th>
<th>Subject</th>
<th>Total Qs</th>
<th>Attempted</th>
<th>Correct</th>
<th>Wrong</th>

```

```

        <th>%</th>
        <th>Score</th>
        <th>Date</th>
        <th>Time</th>
    </tr>
</thead>
<tbody>`;

resultsData = [];
data.results.forEach((result, index) => {
    resultHTML += `
        <tr>
            <td>${index + 1}</td>
            <td>${result.student_name}</td>
            <td>${result.subject_name}</td>
            <td>${result.total_questions}</td>
            <td>${result.attempted}</td>
            <td>${result.correct}</td>
            <td>${result.wrong}</td>
            <td>${result.percentage}%</td>
            <td>${result.total_score}</td>
            <td>${result.quiz_date}</td>
            <td>${result.quiz_time}</td>
        </tr>`;
    resultsData.push([
        index + 1,
        result.student_name,
        result.subject_name,
        result.total_questions,
        result.attempted,
        result.correct,
        result.wrong,
        result.percentage + "%",
        result.total_score,
        result.quiz_date,
        result.quiz_time
    ]);
});

resultHTML += `
    </tbody>
</table>
</div>`;

resultsContainer.innerHTML += resultHTML;
})
.catch(error => console.error("Error:", error));
}

function generatePDF() {
    const { jsPDF } = window.jspdf;

```

```

const doc = new jsPDF();

doc.setFontSize(10);
doc.text("Quiz Results", 14, 15);
doc.autoTable({
  startY: 25,
  head: [["Test", "Name", "Subject", "Total Qs", "Attempted", "Correct", "Wrong", "Percentage",
"Score", "Date", "Time"]],
  body: resultsData
});

const studentName = resultsData[0] ? resultsData[0][1].replace(/\s+/g, '_') : "quiz_results";
doc.save(`${studentName}.pdf`);
}

function closeOpenedWindow() {
  window.close();
}
</script>
</body>

</html>

```

----- AI Chatbot -----

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>AI Chatbot</title>
  <link rel="icon" type="image/png" href="favicon-32x32.png">

  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet">
  <style>
    body {
      background: linear-gradient(135deg, #f5f7fa, #c3cfe2);
      min-height: 100vh;
      display: flex;
      align-items: center;
      justify-content: center;
    }

    .chat-container {

```

```
background: #ffffff;
border-radius: 20px;
box-shadow: 0 10px 25px rgba(0, 0, 0, 0.15);
width: 100%;
max-width: 600px;
height: 80vh;
display: flex;
flex-direction: column;
overflow: hidden;
position: relative;
}
```

```
.chat-header {
  background: #0d6efd;
  color: #fff;
  padding: 15px;
  text-align: center;
  font-weight: 500;
  font-size: 1.2rem;
}
```

```
.chat-box {
  flex: 1;
  overflow-y: auto;
  scroll-behavior: smooth;
  /* Smooth scroll effect */
}
```

```
.chat-box::-webkit-scrollbar {
  width: 6px;
}
```

```
.chat-box::-webkit-scrollbar-thumb {
  background-color: #ccc;
  border-radius: 20px;
}
```

```
.input-area {
  display: flex;
  border-top: 1px solid #ddd;
  padding: 10px;
  background-color: #f8f9fa;
}
```

```
.input-area input {
  flex: 1;
  border: none;
  padding: 12px;
  border-radius: 10px;
  margin-right: 10px;
}
```

```

.input-area input:focus {
  outline: none;
}

.input-area button {
  background-color: #0d6efd;
  color: #fff;
  border: none;
  padding: 12px 20px;
  border-radius: 10px;
}

.user-msg,
.bot-msg {
  margin-bottom: 6px;
  padding: 10px 15px;
  border-radius: 10px;
  max-width: 80%;
  word-wrap: break-word;
}

.user-msg {
  background-color: #e9f5ff;
  align-self: flex-end;
  text-align: right;
}

.bot-msg {
  background-color: #e0ffe6;
  align-self: flex-start;
}

.icon-btn {
  position: absolute;
  top: 10px;
  right: 10px;
  color: #fff;
  font-size: 24px;
  cursor: pointer;
}
</style>
<script type="module"
src="https://unpkg.com/ionicons@7.1.0/dist/ionicons/ionicons.esm.js"></script>
<script nomodule src="https://unpkg.com/ionicons@7.1.0/dist/ionicons/ionicons.js"></script>
</head>

<body>
<div class="chat-container d-flex flex-column">
  <div class="chat-header">
    AI Chatbot

```

```

        <ion-icon name="scan-outline" class="icon-btn"></ion-icon>
    </div>
    <div class="chat-box d-flex flex-column" id="chatBox"></div>
    <div class="input-area">
        <input type="text" id="userInput" placeholder="Type your question..." class="form-control">
        <button onclick="sendMessage()">Send</button>
    </div>
</div>

<script>
    async function sendMessage() {
        let userInput = document.getElementById("userInput").value;
        if (!userInput.trim()) return;
        let chatBox = document.getElementById("chatBox");

        // Show user's message
        let userDiv = document.createElement("div");
        userDiv.className = "user-msg align-self-end";
        userDiv.innerHTML = `<b>You:</b> ${userInput}`;
        chatBox.appendChild(userDiv);

        document.getElementById("userInput").value = "";

        // Scroll to the bottom
        chatBox.scrollTop = chatBox.scrollHeight;

        try {
            let response = await fetch("chatbot.php", {
                method: "POST",
                headers: { "Content-Type": "application/json" },
                body: JSON.stringify({ message: userInput })
            });
            let data = await response.json();

            let botDiv = document.createElement("div");
            botDiv.className = "bot-msg align-self-start";
            botDiv.innerHTML = `<b>Bot:</b> ${data.response || "Something went wrong"}`;
            chatBox.appendChild(botDiv);

            // Scroll to the bottom after bot message
            chatBox.scrollTop = chatBox.scrollHeight;
        } catch (err) {
            console.error("Error:", err);
            chatBox.innerHTML += `<div class='bot-msg align-self-start'><b>Bot:</b> Error connecting
to server</div>`;
            chatBox.scrollTop = chatBox.scrollHeight; // Scroll to bottom even in error case
        }
    }

</script>
</body>

```



```
</html>
```

----- User profile -----

```
<?php include 'auth_check.php'; ?>
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="UTF-8" />
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
```

```
<title>Update Info with Image</title>
```

```
<link rel="icon" type="image/png" href="favicon-32x32.png">
```

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css"
rel="stylesheet" />
```

```
<link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.11.3/font/bootstrap-icons.min.css"
rel="stylesheet">
```

```
<style>
```

```
body {
```

```
background: linear-gradient(135deg, #00c6ff, #0072ff);
```

```
font-family: 'Segoe UI', sans-serif;
```

```
}
```

```
.profile-container {
```

```
max-width: 800px;
```

```
margin: 80px auto;
```

```
background-color: #fff;
```

```
padding: 30px;
```

```
border-radius: 15px;
```

```
box-shadow: 0 10px 30px rgba(0, 0, 0, 0.1);
```

```
}
```

```
.profile-pic {
```

```
width: 120px;
```

```
height: 120px;
```

```
border-radius: 50%;
```

```
object-fit: cover;
```

```
border: 3px solid #fff;
```

```
box-shadow: 0 0 10px rgba(0, 0, 0, 0.3);
```

```
}
```

```
.profile-pic:hover {
  opacity: 0.8;
  cursor: pointer;
}

.file-input {
  display: none;
}

.profile-header {
  text-align: center;
  margin-bottom: 20px;
}

.profile-info p {
  font-size: 16px;
  color: #555;
}

.modal-header,
.modal-footer {
  background-color: #0072ff;
  color: #fff;
}

.modal-body input {
  border-radius: 10px;
}

footer {
  background-color: #232323;
  color: white;
  text-align: center;
  padding: 20px 0;
  margin-top: 50px;
}

.btn-custom {
  background-color: #0072ff;
  color: white;
  border: none;
  border-radius: 5px;
  padding: 8px 16px;
  transition: 0.3s;
}

.btn-custom:hover {
  background-color: #0056b3;
}
</style>
```

```

</head>

<body id="body">
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark px-4">
    <a class="navbar-brand" href="#">ONLINE EXAMINATION SYSTEM</a>
    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarNav">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse justify-content-end" id="navbarNav">
      <ul class="navbar-nav align-items-center">
        <li class="nav-item"><a class="nav-link" href="subjectchoose.php">HOME</a></li>
        <li class="nav-item"><a class="nav-link" href="logout.php">LOGOUT</a></li>
        <!-- Theme Toggle Button -->
        <li class="nav-item">
          <button class="btn btn-sm btn-light ms-2" id="themeToggle">
            <i class="bi bi-moon-fill"></i> Theme
          </button>
        </li>

      </ul>
    </div>
  </nav>

  <div class="container mt-5">
    <div class="profile-container">
      <div class="profile-header">
        <form id="imageForm" enctype="multipart/form-data">
          <label for="fileUpload">
            
          </label>
          <input type="file" name="image" id="fileUpload" class="file-input" accept="image/*">
          <input type="hidden" name="Roll_Number" id="roll_hidden">
        </form>
        <h3 id="user_name">Username</h3>
        <p id="name">Full Name</p>
      </div>
      <div class="profile-info">
        <p>Email: <span id="email">---</span></p>
        <p>Phone: <span id="phone_number">---</span></p>
        <p>Roll: <span id="Roll_Number">---</span></p>
        <button class="btn-custom mt-3" data-bs-toggle="modal" data-bs-target="#editModal">Edit
Info</button>
      </div>
    </div>
  </div>

  <!-- EDIT MODAL -->

  <div class="modal fade" id="editModal" tabindex="-1">
    <div class="modal-dialog">

```

```

<div class="modal-content">
  <!-- UPDATE FORM -->
  <form id="updateForm">
    <div class="modal-header">
      <h5 class="modal-title">Edit Information</h5>
      <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
    </div>
    <div class="modal-body">
      <input type="hidden" name="Roll_Number" id="editRoll">
      <div class="mb-3">
        <label>Name</label>
        <input type="text" name="name" id="editName" class="form-control" required>
      </div>
      <div class="mb-3">
        <label>Username</label>
        <input type="text" name="username" id="user" class="form-control" required>
      </div>
      <div class="mb-3">
        <label>Email</label>
        <input type="email" name="email" id="editEmail" class="form-control" required>
      </div>
      <div class="mb-3">
        <label>Password</label>
        <input type="password" name="password" id="editpassword" class="form-control"
required>
      </div>
      <div class="mb-3">
        <label>Phone</label>
        <input type="text" name="phone_number" id="editPhone" class="form-control" required>
      </div>
    </div>
    <div class="modal-footer">
      <button class="btn btn-success" type="submit">Update</button>
    </div>
  </form>

  <!-- OTP FORM -->
  <form id="otp" style="display: none;">
    <div class="modal-body">
      <div class="mb-3">
        <label>Enter OTP</label>
        <input type="text" name="otp" id="otp_input" class="form-control" required>
      </div>
    </div>
    <div class="modal-footer">
      <button class="btn btn-success" type="submit" onclick="verifyOTP(event)">Verify</button>
    </div>
  </form>
</div>
</div>
</div>

```

```

<footer class="bg-black text-light" style="box-shadow: 0 4px 8px rgba(0, 0, 0, 0.3);">
  <div class="container">
    <div class="row">
      <div class="col-6 col-lg-4">
        <h4 class="pt-3 fw-bold"></h4>
        <p>Online Examination System</p>
        <p>6398428250</p>
      </div>
      <div class="col">
      </div>
      <div class="col">
        <h4 class="pt-3">More</h4>
        <ul class="list-unstyled">
          <li><a href="#" class="text-decoration-none text-light">Result</a></li>
          <li><a href="#" class="text-decoration-none text-light">Get Certificate</a></li>
        </ul>
      </div>
      <div class="col">
        <h4 class="pt-3">Categories</h4>
        <ul class="list-unstyled">
          <li><a href="#" class="text-decoration-none text-light">Update Profile</a></li>
        </ul>
      </div>
      <div class="col-6 col-lg-3 text-lg-end">
        <h4 class="pt-3">Social Media Icons</h4>
        <div>
          <a href="instagram.com" class="text-decoration-none text-light"><i class="bi bi-
instagram fs-6 me-80"
            style="font-size:45px"></i></a>
          <a href="github.com" class="text-decoration-none text-light"><i class="bi bi-github fs-6
me-80"
            style="font-size:45px"></i></a>
          <a href="https://www.linkedin.com/in/mohd-tanzeem-
908161314?utm_source=share&utm_campaign=share_via&utm_content=profile&utm_medium=an
droid_app"
            class="text-decoration-none text-light"><i class="bi bi-linkedin fs-6 me-3"
            style="font-size:45px"></i></a>
          <a href="https://www.youtube.com/@CodingSeekhoAurSeekhao-q2i"
            class="text-decoration-none text-light"><i class="bi bi-youtube fs-6 me-3"
            style="font-size:45px"></i></a>
        </div>
      </div>
    </div>
  </div>
  <hr>
  <div class="d-flex justify-content-between">
    <p>2025 Coding Seekho Aur Seekhao. All Rights Reserved.</p>
    <div>
      <a href="#" class="text-decoration-none text-light me-4">Terms And Condition</a>
      <a href="#" class="text-decoration-none text-light">Privacy Policy</a>
    </div>
  </div>

```

```

        </div>
    </div>
</footer>
<script>
const roll = localStorage.getItem("roll_number");
document.getElementById("roll_hidden").value = roll;

function fetchData() {
    fetch("fetch_name.php")
        .then(res => res.json())
        .then(data => {
            if (data.success) {
                document.getElementById("user_name").innerText = data.user_name;
                document.getElementById("user").value = data.user_name;
                document.getElementById("name").innerText = data.name;
                document.getElementById("email").innerText = data.email;
                document.getElementById("phone_number").innerText = data.phone_number;
                document.getElementById("Roll_Number").innerText = data.Roll_Number;
                document.getElementById("editRoll").value = data.Roll_Number;
                document.getElementById("editName").value = data.name;
                document.getElementById("editEmail").value = data.email;
                document.getElementById("editPhone").value = data.phone_number;
                document.getElementById("profileImage").src = "uploads/" + data.image;
            }
        });
}

// Upload profile image when file is selected
document.getElementById("fileUpload").addEventListener("change", function () {
    const imgForm = document.getElementById("imageForm");
    const formData = new FormData(imgForm);
    fetch("upload_image.php", {
        method: "POST",
        body: formData
    })
        .then(res => res.json())
        .then(data => {
            if (data.success) {
                alert("Image uploaded successfully!");
                document.getElementById("profileImage").src = "uploads/" + data.image;
            } else {
                alert("Image upload failed!");
            }
        });
});

fetchData();
</script>
<script>
document.getElementById("updateForm").addEventListener("submit", function (e) {
    e.preventDefault();

```

```
const sendBtn = document.getElementById('otp');

sendBtn.disabled = true;
sendBtn.style.cursor = "not-allowed";

const formData = new FormData(this);

fetch("update.php", {
  method: "POST",
  body: formData
})
.then(response => response.text())
.then(data => {
  if (data.trim() === "success") {
    alert('✅ OTP sent Email in spam');
    document.getElementById("otp").style.display = "block";
  }

  else if (data.trim() === "already_registered") {
    alert('✅ already_data Present');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
  }

  else if (data.trim() === "already_password") {
    alert('❌ Already password stored database');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
  }

  else if (data.trim() === "phone") {
    alert('❌ already stored Phone Number');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
  }

  else if (data.trim() === "email") {
    alert('Already Email registered');
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
  }

  else {
    alert('❌ Failed to send OTP: ' + data);
    sendBtn.disabled = false;
    sendBtn.style.cursor = "pointer";
  }
})
.catch(error => {
  alert('❌ Error: ' + error);
  sendBtn.disabled = false;
  sendBtn.style.cursor = "pointer";
});
```

```

    });
  });
  function verifyOTP(event) {
    event.preventDefault();
    const otp = document.getElementById('otp_input').value;

    fetch("update_verify.php", {
      method: "POST",
      headers: { "Content-Type": "application/x-www-form-urlencoded" },
      body: "otp=" + encodeURIComponent(otp)
    })
    .then(response => response.text())
    .then(data => {
      if (data.trim().toLowerCase() === "success") {
        alert('✅ Update successfully!');
        fetchData();
      }

      else if (data.trim() === "Failed") {
        alert('❌ Something went wrong please try again');
      }

      else if (data.trim() === "invalid") {
        alert('❌ Invalid Otp');
      }

      else {
        alert('❌ Error: ' + data);
      }
    });
  }
</script>
<script>
const toggleBtn = document.getElementById("themeToggle");
const body = document.getElementById("body");

toggleBtn.addEventListener("click", () => {
  document.body.classList.toggle("light-theme");
  const icon = toggleBtn.querySelector("i");
  if (document.body.classList.contains("light-theme")) {
    body.style.backgroundColor = "black";
    icon.classList.remove("bi-moon-fill");
    icon.classList.add("bi-sun-fill");
  } else {
    body.style.backgroundColor = "white";
    icon.classList.remove("bi-sun-fill");
    icon.classList.add("bi-moon-fill");
  }
});
</script>

```



```
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
</body>

</html>
```

----- Backend Fetch Code -----

```
<?php
session_start();
include 'db_connection.php';

if (!isset($_SESSION["name"])) {
    die(json_encode(["success" => false, "message" => "Please login first!"]));
}

$Roll_Number = $_POST['Roll_Number'];
$sql = "SELECT * FROM quiz_results WHERE Roll_Number = ? ORDER BY id ASC";
$stmt = $conn->prepare($sql);
$stmt->bind_param("i", $Roll_Number);
$stmt->execute();
$result = $stmt->get_result();

$results = [];
if ($result->num_rows > 0) {
    while ($row = $result->fetch_assoc()) {
        $results[] = [
            "student_name" => $row["student_name"],
            "subject_name" => $row["subject_name"],
            "total_questions" => $row["total_questions"],
            "attempted" => $row["attempted"],
            "correct" => $row["correct"],
            "wrong" => $row["wrong"],
            "percentage" => $row["percentage"],
            "total_score" => $row["total_score"],
            "quiz_date" => $row["quiz_date"],
            "quiz_time" => $row["quiz_time"],
            "Roll_Number" => $row["Roll_Number"]
        ];
    }
    echo json_encode(["success" => true, "results" => $results]);
}
else {
    echo json_encode(["success" => false, "message" => "No result found"]);
}
$stmt->close();
```

```
$conn->close();  
?>
```

```
<?php  
session_start();  
include 'db_connection.php';  
  
$roll = $_SESSION["Roll_Number"];  
if (isset($_FILES["image"])) {  
    $img_name = $_FILES["image"]["name"];  
    $tmp_name = $_FILES["image"]["tmp_name"];  
    $target = "uploads/" . basename($img_name);  
  
    if (move_uploaded_file($tmp_name, $target)) {  
        $stmt = $conn->prepare("UPDATE users SET image=? WHERE Roll_Number=?");  
        $stmt->bind_param("ss", $img_name, $roll);  
        if ($stmt->execute()) {  
            echo json_encode(["success" => true, "image" => $img_name]);  
        } else {  
            echo json_encode(["success" => false, "message" => "DB update error"]);  
        }  
    } else {  
        echo json_encode(["success" => false, "message" => "Upload failed"]);  
    }  
}  
  
?>
```

```
<?php  
use PHPMailer\PHPMailer\PHPMailer;  
use PHPMailer\PHPMailer\Exception;  
  
session_start();  
require 'PHPMailer/src/PHPMailer.php';  
require 'PHPMailer/src/SMTP.php';  
require 'PHPMailer/src/Exception.php';  
require 'db_connection.php';  
  
function otp($length = 4)  
{  
    $digit = "";  
    for ($i = 0; $i < $length; $i++) {  
        $digit .= mt_rand(2, 9);  
    }  
    return $digit;  
}  
  
if ($_SERVER["REQUEST_METHOD"] == "POST") {  
    $roll = $_POST['Roll_Number']; // current user Roll_Number  
    $name = $_POST['name'];
```

```

$toEmail = $_POST['email'];
$username = $_POST['username'];
$password = $_POST['password'];
$phone = $_POST['phone_number'];
$city = $_POST['city'] ?? '';
$gender = $_POST['gender'] ?? '';

// Check email, username, phone duplicate (excluding current Roll_Number)
$stmt = $conn->prepare("SELECT * FROM users WHERE email = ? AND user_name = ? AND
phone_number = ? AND Roll_Number != ?");
$stmt->bind_param("ssss", $toEmail, $username, $phone, $roll);
$stmt->execute();
$result = $stmt->get_result();
if ($result->num_rows > 0) {
    echo "already_registered";
    exit;
}


// Check phone exists (excluding current Roll_Number)
$stmt = $conn->prepare("SELECT * FROM users WHERE phone_number = ? AND Roll_Number !=
?");
$stmt->bind_param("ss", $phone, $roll);
$stmt->execute();
$result = $stmt->get_result();
if ($result->num_rows > 0) {
    echo "phone";
    exit;
}

// Check password exists (excluding current Roll_Number)
$stmt = $conn->prepare("SELECT password FROM users WHERE Roll_Number != ?");
$stmt->bind_param("s", $roll);
$stmt->execute();
$result = $stmt->get_result();
while ($row = $result->fetch_assoc()) {
    if (password_verify($password, $row['password'])) {
        echo "already_password";
        exit;
    }
}


// Check email exists (excluding current Roll_Number)
$stmt = $conn->prepare("SELECT * FROM users WHERE email = ? AND Roll_Number != ?");
$stmt->bind_param("ss", $toEmail, $roll);
$stmt->execute();
$result = $stmt->get_result();
if ($result->num_rows > 0) {
    echo "email";
    exit;
}

```

```

//  Send OTP
$otp = otp();
$_SESSION['otp'] = $otp;
$_SESSION['email'] = $toEmail;
$_SESSION['username'] = $username;
$_SESSION['name'] = $name;
$_SESSION['password'] = $password;
$_SESSION['phone_number'] = $phone;
$_SESSION['Roll_Number'] = $roll;

$mail = new PHPMailer(true);
try {
    $mail->isSMTP();
    $mail->Host = 'smtp.gmail.com';
    $mail->SMTPAuth = true;
    $mail->Username = 'onlineexaminationsystem26@gmail.com';
    $mail->Password = 'eccf sgef haxf swpi'; // App password
    $mail->SMTPSecure = 'tls';
    $mail->Port = 587;

    $mail->setFrom('onlineexaminationsystem26@gmail.com', 'Online Examination System');
    $mail->addAddress($toEmail);
    $mail->isHTML(true);
    $mail->Subject = 'Your OTP Code - Online Examination System';
    $mail->Body = "
        <div style='font-family: Arial, sans-serif; color: #000;'>
            <h3 style='color: #2E86C1;'>OTP Verification</h3>
            <p>Your One-Time Password (OTP) is:</p>
            <h2 style='background: #f2f2f2; padding: 10px; display: inline-block;'>$otp</h2>
            <p>Please use this OTP to confirm your profile update.</p>
            <br>
            <small style='color: gray;'> Do not share this code. It is valid for a limited time
only.</small>
        </div>
    ";
    $mail->AltBody = "Your OTP is: $otp\nUse it to confirm your profile update.";

    $mail->send();
    echo "success";
} catch (Exception $e) {
    echo "error: " . $mail->ErrorInfo;
}
}
?>

<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require 'PHPMailer/src/PHPMailer.php';

```

```

require 'PHPMailer/src/SMTP.php';
require 'PHPMailer/src/Exception.php';
require 'db_connection.php';

session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $userOtp = $_POST['otp'];
    $name = $_SESSION['name'];
    $email = $_SESSION['email'];
    $username = $_SESSION['username'];
    $rawPassword = $_SESSION['password'];
    $phone = $_SESSION['phone_number'];
    $roll = $_SESSION['Roll_Number'];

    if (isset($_SESSION['otp']) && $userOtp == $_SESSION['otp']) {
        // 1. Insert into DB
        $hashedPassword = password_hash($rawPassword, PASSWORD_BCRYPT);

        $stmt = $conn->prepare("UPDATE users SET name = ?, email = ?, user_name = ?,
phone_number = ?, password = ? WHERE Roll_Number = ?");
        $stmt->bind_param("sssssd", $name, $email, $username, $phone, $hashedPassword, $roll);

        if ($stmt->execute()) {
            // 2. Send Email
            $mail = new PHPMailer(true);
            try {
                $mail->isSMTP();
                $mail->Host = 'smtp.gmail.com';
                $mail->SMTPAuth = true;
                $mail->Username = 'onlineexaminationsystem26@gmail.com';
                $mail->Password = 'eccf sgef haxf swpi';
                $mail->SMTPSecure = 'tls';
                $mail->Port = 587;

                $mail->setFrom('onlineexaminationsystem@gmail.com', 'Online Examination System');
                $mail->addAddress($email);
                $mail->isHTML(true);
                $mail->Subject = 'Your Login Info';
                $mail->Body = "
                    <h3>📧 Welcome to Online Examination System</h3>
                    <p><strong>Name:</strong> $name</p>
                    <p><strong>Username:</strong> $username</p>
                    <p><strong>Password:</strong> $rawPassword</p>
                    <p><strong>Phone:</strong> $phone</p>
                    <br>
                    <small>🚫 Keep this info safe and do not share it with anyone.</small>
                ";
                $mail->send();
            } catch (Exception $e) {
                echo "❌ Mail error: " . $mail->ErrorInfo;
            }
        }
    }
}

```

```

    }

    // Success
    echo "success";
} else {
    echo "Failed";
}

$stmt->close();
$conn->close();

} else {
    echo "invalid";
}
}
?>

<?php
include 'db_connection.php';
session_start();

$Roll_Number = $_SESSION['Roll_Number'] ?? null;
$text = $_POST['texts'] ?? null;

if (empty($Roll_Number)) {
    echo "Session expired. Please log in again.";
    exit;
}

if (empty($text)) {
    echo "Note text is missing.";
    exit;
}

// Check if the record already exists
$check_sql = "SELECT * FROM note_text WHERE Roll_Number = ?";
$check_stmt = $conn->prepare($check_sql);
$check_stmt->bind_param("s", $Roll_Number);
$check_stmt->execute();
$result = $check_stmt->get_result();

if ($result->num_rows > 0) {
    // Update existing record
    $update_sql = "UPDATE note_text SET text = ? WHERE Roll_Number = ?";
    $update_stmt = $conn->prepare($update_sql);
    $update_stmt->bind_param("ss", $text, $Roll_Number);

    echo $update_stmt->execute() ? "success" : "Error updating record: " . $conn->error;

    $update_stmt->close();
} else {

```

```

// Insert new record
$insert_sql = "INSERT INTO note_text (Roll_Number, text) VALUES (?, ?)";
$insert_stmt = $conn->prepare($insert_sql);
$insert_stmt->bind_param("ss", $Roll_Number, $text);

echo $insert_stmt->execute() ? "success" : "Error inserting record: " . $conn->error;

$insert_stmt->close();
}

$check_stmt->close();
$conn->close();

?>

<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require 'PHPMailer/src/PHPMailer.php';
require 'PHPMailer/src/SMTP.php';
require 'PHPMailer/src/Exception.php';
require 'db_connection.php';

session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $rating = $_POST['rating'];
    $name = $_SESSION['name'];
    $email = $_SESSION['email'];
    $feedback = htmlspecialchars($_POST['feedback']); // Sanitize input

    $stmt = $conn->prepare("INSERT INTO feedbacks (name, email, rating, feedback) VALUES (?, ?, ?, ?)");
    $stmt->bind_param("ssis", $name, $email, $rating, $feedback);

    if ($stmt->execute()) {
        $mail = new PHPMailer(true);
        try {
            // SMTP Configuration
            $mail->isSMTP();
            $mail->Host = 'smtp.gmail.com';
            $mail->SMTPAuth = true;
            $mail->Username = 'onlineexaminationsystem26@gmail.com';
            $mail->Password = 'eccf sgef haxf swpi'; // Should be app-specific password
            $mail->SMTPSecure = 'tls';
            $mail->Port = 587;

            // Email Content
            $mail->setFrom('onlineexaminationsystem26@gmail.com', 'Online Examination System');
            $mail->addAddress($email);

```

```

$mail->isHTML(true);
$mail->Subject = 'FEEDBACK';
$mail->Body = "
    <h3>📧 Welcome to Online Examination System</h3>
    <h1>📧 Thanks For Your Feedback!</h1>
    <p>Dear <strong>$feedback</strong>,</p>
    <p>We really appreciate your valuable input.</p>
    <br>
    <small>🚫 Please keep this email for your reference and do not share it.</small>
";
$mail->send();
} catch (Exception $e) {
    echo "❌ Mail error: " . $mail->ErrorInfo;
}
echo "success";
} else {
    echo "Failed";
}

$stmt->close();
$conn->close();
}
?>

<?php
include 'db_connection.php';

$data = json_decode(file_get_contents("php://input"), true);

if (!$data) {
    die(json_encode(["success" => false, "message" => "Invalid JSON data"]));
}
$student_name = $data['student_name'];
$subject_name = $data['subject_name'];
$total_questions = $data['total_questions'];
$attempted = $data['attempted'];
$correct = $data['correct'];
$wrong = $data['wrong'];
$percentage = $data['percentage'];
$total_score = $data['total_score'];
$Roll_Number=$data['Roll_Number'];
$sql = "INSERT INTO quiz_results (student_name, subject_name, total_questions, attempted,
correct, wrong, percentage, total_score,Roll_Number)
VALUES ('$student_name', '$subject_name', '$total_questions', '$attempted', '$correct',
'$wrong', '$percentage', '$total_score', '$Roll_Number')";

if ($conn->query($sql) === TRUE) {
    echo json_encode(["success" => true, "message" => "Result saved successfully"]);
} else {
    echo json_encode(["success" => false, "message" => "Error: " . $sql . " - " . $conn->error]);
}

```



```

$conn->close();
?>

<?php
session_start();
include 'db_connection.php'; // Database connection

// Check if the Roll Number is set in the session
if (!isset($_SESSION['Roll_Number'])) {
    echo json_encode(['status' => 'error', 'message' => 'Roll Number not found in session']);
    exit;
}

// Get Roll Number from session
$Roll_Number = $_SESSION['Roll_Number'];

// Check if Roll Number exists in the note_text table
$check_sql = "SELECT text FROM note_text WHERE Roll_Number = ?";
$check_stmt = $conn->prepare($check_sql);
$check_stmt->bind_param("s", $Roll_Number);
$check_stmt->execute();
$result = $check_stmt->get_result();

if ($result->num_rows > 0) {
    // Fetch the note text from the database
    $row = $result->fetch_assoc();
    $noteText = $row['text'];
    echo json_encode(['status' => 'success', 'text' => $noteText]); // Return the note text as JSON
} else {
    echo json_encode(['status' => 'error', 'message' => 'No notes found for this Roll Number']); // If no
    note found
}

$check_stmt->close();
$conn->close();
?>

<?php
session_start();
header('Content-Type: application/json');

include 'db_connection.php';

if (!isset($_SESSION['Roll_Number'])) {
    echo json_encode(['success' => false, 'message' => "Roll Number not found in session"]);
    exit();
}

$roll_number = $_SESSION['Roll_Number'];

```

```

$sql = "SELECT user_name, name, email, phone_number,password, Roll_Number,image FROM users
WHERE Roll_Number = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("d", $roll_number);
$stmt->execute();
$result = $stmt->get_result();

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

    echo json_encode([
        "success" => true,
        "user_name" => $row["user_name"],
        "name" => $row["name"],
        "email" => $row["email"],
        "password" => $row["password"],
        "phone_number" => $row["phone_number"],
        "Roll_Number" => $row["Roll_Number"],
        "image" => $row["image"]
    ]);
} else {
    echo json_encode(["success" => false, "message" => "No data found"]);
}

$stmt->close();
$conn->close();
?>

```

```

<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require 'PHPMailer/src/PHPMailer.php';
require 'PHPMailer/src/SMTP.php';
require 'PHPMailer/src/Exception.php';
require 'db_connection.php';

session_start();
$message = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {

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

    $user_input = $_POST['user_name']; // yeh input ho sakta hai user_name ya email
    $password = $_POST['password'];

    // user_name ya email dono match karne ka SQL

```

```

    $sql = "SELECT name, Roll_Number, email, password FROM users WHERE user_name = ? OR email
= ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("ss", $user_input, $user_input);
    $stmt->execute();
    $result = $stmt->get_result();

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

        if (password_verify($password, $user['password'])) {
            $_SESSION['name'] = $user['name'];
            $_SESSION['Roll_Number'] = $user['Roll_Number'];
            $email = $user['email'];

            if (!isset($_SESSION['email'])) {
                $mail = new PHPMailer(true);
                try {
                    $mail->isSMTP();
                    $mail->Host = 'smtp.gmail.com';
                    $mail->SMTPAuth = true;
                    $mail->Username = 'onlineexaminationsystem26@gmail.com';
                    $mail->Password = 'eccf sgef haxf swpi';
                    $mail->SMTPSecure = 'tls';
                    $mail->Port = 587;

                    $mail->setFrom('onlineexaminationsystem26@gmail.com', 'Online Examination System');
                    $mail->addAddress($email);
                    $mail->isHTML(true);
                    $mail->Subject = 'Your Login Info';
                    $mail->Body = "
                        <h3>📧 Welcome to Online Examination System</h3>
                        <br>
                        <h1>✅ Login Successful</h1>
                        <br>
                        <small>🚫 Keep this info safe and do not share it with anyone.</small>
                    ";
                    $mail->send();

                    $_SESSION['email'] = $email;

                } catch (Exception $e) {
                    echo "❌ Mail error: " . $mail->ErrorInfo;
                }
            }

            echo "success";

        } else {
            echo "wrong"; // wrong password
        }
    }

```

```

    } else {
        echo "username"; // username or email not found
    }

    $stmt->close();
    $conn->close();
}
?>

<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require 'PHPMailer/src/PHPMailer.php';
require 'PHPMailer/src/SMTP.php';
require 'PHPMailer/src/Exception.php';
require 'db_connection.php';

session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $userOtp = $_POST['otp'];
    $email = $_SESSION['email'];
    $password = $_SESSION['password'];
    $sessionOtp = $_SESSION['otp'];

    if ($userOtp === $sessionOtp) {
        $hashedPassword = password_hash($password, PASSWORD_BCRYPT);

        $stmt = $conn->prepare("UPDATE users SET password = ? WHERE email = ?");
        $stmt->bind_param("ss", $hashedPassword, $email);

        if ($stmt->execute()) {
            // Send confirmation email
            $mail = new PHPMailer(true);
            try {
                $mail->isSMTP();
                $mail->Host = 'smtp.gmail.com';
                $mail->SMTPAuth = true;
                $mail->Username = 'onlineexaminationsystem26@gmail.com';
                $mail->Password = 'eccf sgef haxf swpi';
                $mail->SMTPSecure = 'tls';
                $mail->Port = 587;

                $mail->setFrom('onlineexaminationsystem26@gmail.com', 'Online Examination System');
                $mail->addAddress($email);
                $mail->isHTML(true);
                $mail->Subject = 'Password Updated Successfully';
                $mail->Body = "
                    <h3>Password Updated</h3>
                    <p>Your password has been successfully updated.</p>

```

```

        <p><strong>Email:</strong> $email</p>
        <p><strong>New Password:</strong> $password</p>
        <small>If you did not request this change, contact support immediately.</small>
    ";

    $mail->send();
} catch (Exception $e) {
    echo "mail_error: " . $mail->ErrorInfo;
    exit;
}

echo "success";
session_unset();
session_destroy();
} else {
    echo "db_error: " . $stmt->error;
}

$stmt->close();
$conn->close();
} else {
    echo "invalid";
}
}
?>

```

```

<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require 'PHPMailer/src/PHPMailer.php';
require 'PHPMailer/src/SMTP.php';
require 'PHPMailer/src/Exception.php';
require 'db_connection.php';

session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $userOtp = $_POST['otp'];
    $name = $_SESSION['name'];
    $email = $_SESSION['email'];
    $username = $_SESSION['username'];
    $rawPassword = $_SESSION['password'];
    $phone = $_SESSION['phone_number'];
    $city = $_SESSION['city'];
    $gender = $_SESSION['gender'];

    if (isset($_SESSION['otp']) && $userOtp == $_SESSION['otp']) {
        // 1. Insert into DB
        $Roll_Number = rand(1000000000000, 999999999999);
        $hashedPassword = password_hash($rawPassword, PASSWORD_BCRYPT);
    }
}

```

```

$stmt = $conn->prepare("INSERT INTO users (name, email, user_name, phone_number,
Roll_Number, city, gender, password) VALUES (?, ?, ?, ?, ?, ?, ?, ?)");
$stmt->bind_param("ssddsss", $name, $email, $username, $phone, $Roll_Number, $city,
$gender, $hashedPassword);

if ($stmt->execute()) {
    // 2. Send Email
    $mail = new PHPMailer(true);
    try {
        $mail->isSMTP();
        $mail->Host = 'smtp.gmail.com';
        $mail->SMTPAuth = true;
        $mail->Username = 'onlineexaminationsystem26@gmail.com';
        $mail->Password = 'eccf sgef haxf swpi';
        $mail->SMTPSecure = 'tls';
        $mail->Port = 587;

        $mail->setFrom('onlineexaminationsystem@gmail.com', 'Online Examination System');
        $mail->addAddress($email);
        $mail->isHTML(true);
        $mail->Subject = 'Your Login Info';
        $mail->Body = "
            <h3>📧 Welcome to Online Examination System</h3>
            <p><strong>Name:</strong> $name</p>
            <p><strong>Username:</strong> $username</p>
            <p><strong>Password:</strong> $rawPassword</p>
            <p><strong>Phone:</strong> $phone</p>
            <p><strong>City:</strong> $city</p>
            <p><strong>Gender:</strong> $gender</p>
            <br>
            <small>🔒 Keep this info safe and do not share it with anyone.</small>
        ";
        $mail->send();
    } catch (Exception $e) {
        echo "✖ Mail error: " . $mail->ErrorInfo;
    }

    session_unset();
    session_destroy();
    echo "success";
} else {
    echo "Failed";
}

$stmt->close();
$conn->close();

} else {
    echo "invalid";
}
}

```

```

}
?>
<?php
include 'db_connection.php';

$Roll_Number = $_SESSION['Roll_Number'] ?? null;
$subjects = [];

if ($Roll_Number) {
    $sql = "SELECT subject_name FROM quiz_results WHERE Roll_Number = ? AND percentage >= 80";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("s", $Roll_Number);
    $stmt->execute();
    $result = $stmt->get_result();

    while ($row = $result->fetch_assoc()) {
        $subjects[] = $row['subject_name'];
    }
    $_SESSION['subject_names'] = $subjects;
}
?>

```

----- Logout -----

```

<?php
session_start();
session_unset();
session_destroy();
header("Location: index.php");
exit();

```

----- Admin Dashboard -----

```

<?php include 'admin_check.php'; ?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Admin Dashboard | Online Examination System</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.5/font/bootstrap-
icons.css">
    <!-- In your <head> -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.5/font/bootstrap-
icons.css">

```

```

<style>
body {
  background-color: #f8f9fa;
}

.sidebar {
  height: 100vh;
  background-color: #343a40;
}

.sidebar .nav-link {
  color: #fff;
  cursor: pointer;
}

.sidebar .nav-link:hover,
.sidebar .nav-link.active {
  background-color: #495057;
  color: #fff;
}

.content {
  padding: 20px;
}

iframe {
  width: 100%;
  height: 85vh;
  border: none;
}
</style>
</head>

<body>
<div class="container-fluid">
<div class="row">

  <!-- Sidebar -->
  <nav class="col-md-2 d-none d-md-block bg-dark sidebar">
    <div class="position-sticky pt-3">
      <h4 class="text-center text-white mb-4">Admin Panel</h4>
      <ul class="nav flex-column">
        <li class="nav-item">
          <a class="nav-link active" onclick="loadContent('home')"><i class="bi bi-speedometer2"></i>
Dashboard</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" onclick="loadContent('manage_users.php')"><i class="bi bi-people"></i>
Manage
Users</a>

```



```

        </li>
        <li class="nav-item">
            <a class="nav-link" onclick="loadContent('manage_exams.php')"><i class="bi bi-journal-
text"></i> Manage
            Exams</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" onclick="loadContent('view_results.php')"><i class="bi bi-bar-chart-
line"></i> View
            Results</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" onclick="loadContent('view_feedback.php')"><i class="bi bi-chat-
dots"></i>
            Feedback</a>
        </li>
        <li class="nav-item">
            <a class="nav-link text-danger" href="admin_logout.php"><i class="bi bi-box-arrow-
right"></i> Logout</a>
        </li>
    </ul>
</div>
</nav>

```

```

<!-- Main Content -->

```

```

<main class="col-md-10 ms-sm-auto col-lg-10 px-md-4 content">
    <iframe id="main-frame" src="admin_home.php"></iframe>
</main>

```

```

</div>
</div>

```

```

<script>
function loadContent(page) {
    const frame = document.getElementById('main-frame');
    if (page === 'home') {
        frame.src = 'admin_home.php';
    } else {
        frame.src = page;
    }
}

```

```

// Set active link
const links = document.querySelectorAll('.nav-link');
links.forEach(link => link.classList.remove('active'));
event.target.classList.add('active');
}
</script>

```

```

</body>

```

```

</html>

```

```

<!-- In your <head> -->
<?php include 'admin_check.php'; ?>

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.10.5/font/bootstrap-
icons.css">
<link rel="icon" type="image/png" href="favicon-32x32.png">

<?php
include 'db_connection.php'; // ← apni DB connection file ka naam

// Fetch total users
$userQuery = mysqli_query($conn, "SELECT COUNT(*) AS total_users FROM users");
$userData = mysqli_fetch_assoc($userQuery);
$totalUsers = $userData['total_users'];

// Fetch total exams
$examQuery = mysqli_query($conn, "SELECT COUNT(*) AS total_exams FROM quiz_results");
$examData = mysqli_fetch_assoc($examQuery);
$totalExams = $examData['total_exams'];

// Fetch total feedbacks
$feedbackQuery = mysqli_query($conn, "SELECT COUNT(*) AS total_feedbacks FROM feedbacks");
$feedbackData = mysqli_fetch_assoc($feedbackQuery);
$totalFeedbacks = $feedbackData['total_feedbacks'];
?>

<?php
// include 'db_connection.php';

// 1. Recent Registrations
$regQuery = mysqli_query($conn, "SELECT name, created_at FROM users ORDER BY id DESC LIMIT
5");

// 2. Recent Exam Submissions
$examQuery = mysqli_query($conn, "
SELECT student_name,subject_name,percentage,quiz_date
FROM quiz_results ORDER BY quiz_results.id DESC LIMIT 5");

// // 3. Recent Feedbacks
$fbQuery = mysqli_query($conn, "
SELECT name, feedback,created_at
FROM feedbacks ORDER BY feedbacks.id DESC
LIMIT 5");
?>

```

```

<div class="row g-4 mb-5">
  <div class="col-md-4">
    <div class="card text-white bg-primary shadow-sm">
      <div class="card-body">
        <h5 class="card-title">Total Users</h5>
        <p class="card-text fs-3"><i class="bi bi-person-lines-fill"></i> <?= $totalUsers ?></p>
      </div>
    </div>
  </div>
  <div class="col-md-4">
    <div class="card text-white bg-success shadow-sm">
      <div class="card-body">
        <h5 class="card-title">Total Exams</h5>
        <p class="card-text fs-3"><i class="bi bi-journals"></i> <?= $totalExams ?></p>
      </div>
    </div>
  </div>
  <div class="col-md-4">
    <div class="card text-white bg-warning shadow-sm">
      <div class="card-body">
        <h5 class="card-title">Feedbacks</h5>
        <p class="card-text fs-3"><i class="bi bi-chat-square-text"></i> <?= $totalFeedbacks ?></p>
      </div>
    </div>
  </div>
</div>

<div class="mt-5">
  <h5 class="mb-3">Recent Activity</h5>
  <ul class="list-group">

    <!-- Recent Registrations -->
    <?php while ($row = mysqli_fetch_assoc($regQuery)): ?>
      <li class="list-group-item">
        <i class="bi bi-person-plus-fill text-primary"></i>
        New user registered: <strong><?= $row['name'] ?></strong>
        (<?= date("d M Y", strtotime($row['created_at'])) ?>)
      </li>
    <?php endwhile; ?>

    <!-- Recent Exams -->
    <?php while ($row = mysqli_fetch_assoc($examQuery)): ?>
      <li class="list-group-item">
        <i class="bi bi-person-check-fill text-success"></i>
        <strong><?= $row['student_name'] ?></strong> submitted <strong><?=
$row['subject_name'] ?></strong> on
        <?= date("d M Y", strtotime($row['quiz_date'])) ?> —
        <span class="text-primary fw-semibold">Scored <?= $row['percentage'] ?>%</span>
      </li>
    </li>
  </ul>

```

```

        <?php endwhile; ?>

        <!-- Recent Feedbacks -->
        <?php while ($row = mysqli_fetch_assoc($fbQuery)): ?>
            <li class="list-group-item">
                <i class="bi bi-chat-dots text-warning"></i>
                <strong><?= $row['name'] ?></strong> gave feedback: "<?= $row['feedback'] ?>"
            </li>
        <?php endwhile; ?>

    </ul>
</div>

```

----- Manage Exam -----

```

<?php include 'admin_check.php'; ?>

<?php
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    // Get the subject and questions from the form
    $subject = $_POST['subject'];
    $questions = $_POST['questions'];

    // Define the file path for each subject
    $file_path = "$subject.js";

    // Check if the file exists and is writable
    if (file_exists($file_path) && is_writable($file_path)) {
        // Save the questions into the respective file
        file_put_contents($file_path, $questions);
        $message = "Questions saved successfully!";
    } else {
        $message = "Failed to save questions. File not found or not writable.";
    }
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Manage Exams</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
<div class="container mt-4">
    <h3 class="mb-4">Manage Exam Questions</h3>

```

```

<?php if (isset($message)): ?>
  <div class="alert alert-info"><?= $message; ?></div>
<?php endif; ?>

<form id="questionForm" method="post">
  <div class="mb-3">
    <label for="subject" class="form-label">Select Subject</label>
    <select class="form-select" id="subject" name="subject" required>
      <option value="">-- Select Subject --</option>
      <option value="c">C</option>
      <option value="cpp">C++</option>
      <option value="java">Java</option>
      <option value="python">Python</option>
      <option value="nodejs">Nodejs</option>
      <option value="reactjs">Reactjs</option>
    </select>
  </div>
  <div class="mb-3">
    <label for="questions" class="form-label">Edit Questions</label>
    <textarea class="form-control" id="questions" name="questions" rows="15"
placeholder="Questions will load here..." required></textarea>
  </div>
  <button type="submit" class="btn btn-primary">Save Changes</button>
</form>
</div>

<script>
  // Load questions on subject change
  document.getElementById('subject').addEventListener('change', function () {
    const subject = this.value;
    if (subject) {
      fetch(`load_question.php?subject=${subject}`)
        .then(response => response.text())
        .then(data => {
          // Add spacing after each question in the textarea
          document.getElementById('questions').value = data.replace(/\n/g, '\n\n'); // Adds a double
newline
        });
    } else {
      document.getElementById('questions').value = '';
    }
  });
</script>
</body>
</html>

```

----- Manage User-----

```

<?php include 'admin_check.php'; ?>

```

```

<?php

include 'db_connection.php';

if (isset($_GET['delete'])) {
    $id = $_GET['delete'];
    $conn->query("DELETE FROM users WHERE id = $id");
    header("Location: manage_users.php");
    exit();
}

// Fetch all users
$result = $conn->query("SELECT * FROM users ");
?>

<!DOCTYPE html>
<html>
<head>
    <title>Manage Users</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
    <link rel="icon" type="image/png" href="favicon-32x32.png">

</head>
<body class="bg-light">
<div class="container mt-5">
    <table class="table table-bordered table-hover bg-white">
        <thead class="table-dark">
            <tr>
                <th>ID</th>
                <th>Full Name</th>
                <th>Email</th>
                <th>Profile Image</th>
                <th>Joined On</th>
                <th>Action</th>
            </tr>
        </thead>
        <tbody>
            <?php while ($user = $result->fetch_assoc()): ?>
                <tr>
                    <td><?= $user['id'] ?></td>
                    <td><?= htmlspecialchars($user['name']) ?></td>
                    <td><?= htmlspecialchars($user['email']) ?></td>
                    <td>
                        <?php if ($user['image']): ?>
                            
                        <?php else: ?>
                            <span class="text-muted">No image</span>
                        <?php endif; ?>
                    </td>
                </tr>
            </tbody>
        </table>
    </div>

```

```

        </td>
        <td><?= date('d M Y', strtotime($user['created_at'])) ?></td>
        <td>
            <a href="?delete=<?= $user['id'] ?>" onclick="return confirm('Delete this user?')"
class="btn btn-sm btn-danger">Delete</a>
        </td>
    </tr>
<?php endwhile; ?>
</tbody>
</table>
</div>
</body>
</html>

```

-----View Results-----

```
<?php include 'admin_check.php'; ?>
```

```

<?php
include 'db_connection.php';
?>

```

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>View Results | Admin Panel</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

```

```

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
    <link rel="stylesheet" href="https://cdn.datatables.net/1.13.4/css/dataTables.bootstrap5.min.css">
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
    <script src="https://cdn.datatables.net/1.13.4/js/jquery.dataTables.min.js"></script>
    <script src="https://cdn.datatables.net/1.13.4/js/dataTables.bootstrap5.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/jspdf-
autotable/3.5.23/jspdf.plugin.autotable.min.js"></script>
</head>
<body class="bg-light">

```

```

<div class="container mt-5">
    <h3 class="mb-4 text-center">Quiz Results</h3>

```

```

    <div class="mb-3 text-end">
        <button class="btn btn-success" onclick="exportTableToPDF()">Download PDF</button>
    </div>

```

```

<div class="table-responsive">
<table id="resultsTable" class="table table-bordered table-striped table-hover bg-white">
  <thead class="table-dark">
    <tr>
      <th>#</th>
      <th>Student Name</th>
      <th>Roll No</th>
      <th>Subject</th>
      <th>Total Questions</th>
      <th>Attempted</th>
      <th>Correct</th>
      <th>Wrong</th>
      <th>Score</th>
      <th>Percentage</th>
      <th>Date</th>
      <th>Time</th>
    </tr>
  </thead>
  <tbody>
    <?php
      $query = "SELECT * FROM quiz_results ORDER BY quiz_date DESC, quiz_time DESC";
      $result = $conn->query($query);
      if ($result->num_rows > 0):
        $i = 1;
        while ($row = $result->fetch_assoc()):
          ?>
          <tr>
            <td><?= $i++ ?></td>
            <td><?= htmlspecialchars($row['student_name']) ?></td>
            <td><?= htmlspecialchars($row['Roll_Number']) ?></td>
            <td><?= htmlspecialchars($row['subject_name']) ?></td>
            <td><?= $row['total_questions'] ?></td>
            <td><?= $row['attempted'] ?></td>
            <td><?= $row['correct'] ?></td>
            <td><?= $row['wrong'] ?></td>
            <td><?= $row['total_score'] ?></td>
            <td><?= number_format($row['percentage'], 2) ?>%</td>
            <td><?= date("d M Y", strtotime($row['quiz_date'])) ?></td>
            <td><?= date("h:i A", strtotime($row['quiz_time'])) ?></td>
          </tr>
          <?php endwhile; else: ?>
          <tr>
            <td colspan="12" class="text-center text-muted">No results found.</td>
          </tr>
          <?php endif; ?>
        </tbody>
      </table>
    </div>
  </div>

<script>

```



```

$(document).ready(function () {
    $('#resultsTable').DataTable({
        responsive: true,
        lengthChange: true,
        pageLength: 10,
        ordering: true,
        buttons: []
    });
});

function exportTableToPDF() {
    const { jsPDF } = window.jspdf;
    const doc = new jsPDF();
    doc.autoTable({
        html: '#resultsTable',
        startY: 20,
        headStyles: { fillColor: [41, 128, 185] },
        margin: { top: 10 },
        styles: { fontSize: 8 }
    });
    doc.save('quiz_results.pdf');
}
</script>

</body>
</html>

```

----- View Feedbacks -----

```

<?php include 'admin_check.php'; ?>

<?php
include 'db_connection.php';

// Fetch feedbacks
$result = $conn->query("SELECT * FROM feedbacks ORDER BY id DESC");
?>

<!DOCTYPE html>
<html>
<head>
    <title>View Feedback</title>
    <link rel="icon" type="image/png" href="favicon-32x32.png">

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body class="bg-light">
    <div class="container mt-5">
        <h3 class="mb-4 text-center">User Feedback</h3>

```

```

<div class="table-responsive">
  <table class="table table-bordered table-hover bg-white">
    <thead class="table-dark">
      <tr>
        <th>#</th>
        <th>Name</th>
        <th>Email</th>
        <th>Rating</th>
        <th>Feedback</th>
        <th>Submitted At</th>
      </tr>
    </thead>
    <tbody>
      <?php if ($result->num_rows > 0): ?>
        <?php while ($row = $result->fetch_assoc()): ?>
          <tr>
            <td><?= $row['id'] ?></td>
            <td><?= htmlspecialchars($row['name']) ?></td>
            <td><?= htmlspecialchars($row['email']) ?></td>
            <td>
              <?php for ($i = 0; $i < $row['rating']; $i++): ?>
                ★
              <?php endfor; ?>
            </td>
            <td><?= htmlspecialchars($row['feedback']) ?></td>
            <td><?= date('d M Y, h:i A', strtotime($row['created_at'])) ?></td>
          </tr>
        <?php endwhile; ?>
      <?php else: ?>
        <tr>
          <td colspan="6" class="text-center text-muted">No feedbacks found</td>
        </tr>
      <?php endif; ?>
    </tbody>
  </table>
</div>
</div>
</body>
</html>

```

