

**1)Design a personal portfolio webpage using HTML5 that includes headings, paragraphs, lists, tables, images, hyperlinks, and a simple form for contact submission**

```
<!DOCTYPE html>
<html>
<head>
    <title>My Portfolio</title>
</head>
<body>

    <h1>My Portfolio</h1>
    <p>Hello! My name is _____. Welcome to my personal portfolio webpage.</p>

    <h2>About Me</h2>
    <p>I am a student learning HTML and Web Designing.</p>

    <h2>My Skills</h2>
    <ul>
        <li>HTML</li>
        <li>CSS</li>
        <li>Java</li>
    </ul>

    <h2>My Project</h2>
    <table border="1">
        <tr>
            <th>Project</th>
            <th>Description</th>
        </tr>
        <tr>
            <td>Portfolio Website</td>
            <td>My own personal website</td>
        </tr>
    </table>

    <h2>My Image</h2>
    

    <h2>My Links</h2>
    <a href="https://www.google.com" target="_blank">Visit Google</a>

    <h2>Contact Me</h2>
    <form>
        Name: <input type="text"><br><br>
        Email: <input type="email"><br><br>
        Message: <textarea></textarea><br><br>
```

```
<input type="submit" value="Send">  
</form>  
  
</body>  
</html>
```

**2)Create a multipage website using HTML and CSS showcasing a college department, using internal and external style sheets, CSS selectors, and implementing basic layout design.**

**/\* External CSS \*/**

```
body {  
    font-family: Arial, sans-serif;  
    margin: 0;  
    background-color: #f2f2f2;  
}
```

```
header {  
    background-color: #003366;  
    color: white;  
    padding: 15px;  
    text-align: center;  
}
```

```
nav {  
    background-color: #0059b3;  
    padding: 10px;  
}
```

```
nav a {  
    color: white;  
    margin: 10px;  
    text-decoration: none;  
    font-weight: bold;  
}
```

```
nav a:hover {  
    color: yellow;  
}
```

```
.container {  
    width: 80%;  
    margin: auto;  
    background: white;  
    padding: 20px;  
}
```

---

```
<!DOCTYPE html>
<html>
<head>
    <title>Computer Science Department</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>

<header>
    <h1>Computer Science Department</h1>
</header>

<nav>
    <a href="index.html">Home</a>
    <a href="about.html">About</a>
    <a href="contact.html">Contact</a>
</nav>

<div class="container">
    <h2>Welcome to the CS Department</h2>
    <p>This department offers top-notch education in computing technologies.</p>

    <!-- Internal CSS -->
    <style>
        .highlight {
            background-color: #d9e6f2;
            padding: 15px;
            border-left: 4px solid #003366;
        }
    </style>

    <div class="highlight">
        <p><strong>Our Mission:</strong> To create future technology leaders.</p>
    </div>
</div>

</body>
</html>
```

---

```
<!DOCTYPE html>
<html>
<head>
    <title>About Department</title>
    <link rel="stylesheet" href="style.css">
```

```
<!-- Internal CSS -->
<style>
  h2 { color: #0059b3; }
  #vision {
    background-color: #e6f0ff;
    padding: 10px;
    border-radius: 5px;
  }
</style>
</head>
<body>

<header>
  <h1>Computer Science Department</h1>
</header>

nav>
  <a href="index.html">Home</a>
  <a href="about.html">About</a>
  <a href="contact.html">Contact</a>
</nav>

<div class="container">
  <h2>About Our Department</h2>
  <p>We provide undergraduate and postgraduate programs in the field of Computer Science.</p>

  <div id="vision">
    <h3>Vision</h3>
    <p>To deliver high-quality software professionals for the IT industry.</p>
  </div>
</div>

</body>
</html>



---


<!DOCTYPE html>
<html>
<head>
  <title>Contact Department</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>

<header>
```

```

<h1>Computer Science Department</h1>
</header>

<nav>
  <a href="index.html">Home</a>
  <a href="about.html">About</a>
  <a href="contact.html">Contact</a>
</nav>

<div class="container">
  <h2>Contact Us</h2>
  <p>Email: csdept@college.edu</p>
  <p>Phone: +91 9876543210</p>

  <h3>Faculty Members</h3>
  <ul>
    <li class="faculty">Dr. A. Sharma – HOD</li>
    <li class="faculty">Prof. R. Patil – Lecturer</li>
    <li class="faculty">Prof. K. Singh – Lecturer</li>
  </ul>

  <!-- Internal CSS -->
  <style>
    .faculty {
      background-color: #d1ecff;
      padding: 5px;
      margin: 5px 0;
    }
  </style>
</div>

</body>
</html>

```

**3)Create an XML document for a book store with details like book title, author, price, and genre. Validate it using an XML Schema (XSD).**

```

<?xml version="1.0"?>
<bookstore>
  <book>
    <title>The Odyssey</title>
    <author>Homer</author>
    <price>12.95</price>
    <genre>Classics</genre>
  </book>

```

```

<book>
  <title>Learning XML</title>
  <author>Erik T. Ray</author>
  <price>29.99</price>
  <genre>Education</genre>
</book>
</bookstore>



---


<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="bookstore">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="book" maxOccurs="unbounded">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="title" type="xs:string"/>
            <xs:element name="author" type="xs:string"/>
            <xs:element name="price" type="xs:decimal"/>
            <xs:element name="genre" type="xs:string"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

</xs:schema>

```

**4)Develop an XML document for student records and transform it into a readable HTML format using XSLT.**

```

<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="students.xsl"?>

<students>
  <student>
    <rollno>101</rollno>
    <name>Payal Pagire</name>
    <department>Computer Science</department>
    <marks>85</marks>
  </student>

  <student>

```

```

<rollno>102</rollno>
<name>Amit Sharma</name>
<department>Information Technology</department>
<marks>78</marks>
</student>

<student>
  <rollno>103</rollno>
  <name>Priya Desai</name>
  <department>Electronics</department>
  <marks>92</marks>
</student>
</students>

```

```

<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template match="/">
  <html>
    <head>
      <title>Student Records</title>
    </head>
    <body>
      <h2>Student Records Table</h2>
      <table border="1" cellpadding="6" style="border-collapse: collapse;">
        <tr style="background-color: #cce5ff;">
          <th>Roll No</th>
          <th>Name</th>
          <th>Department</th>
          <th>Marks</th>
        </tr>

        <!-- Loop through each student -->
        <xsl:for-each select="students/student">
          <tr>
            <td><xsl:value-of select="rollno"/></td>
            <td><xsl:value-of select="name"/></td>
            <td><xsl:value-of select="department"/></td>
            <td><xsl:value-of select="marks"/></td>
          </tr>
        </xsl:for-each>

      </table>
    </body>

```

```
</html>
</xsl:template>
</xsl:stylesheet>
```

**5)Create a JavaScript program to accept user input (name and age), validate the input, and display a greeting message using conditional and looping statements.**

```
<!DOCTYPE html>
<html>
<head>
    <title>Greeting Program</title>
</head>
<body>

<h2>User Greeting Program</h2>

<!-- Button to start input -->
<button onclick="getUserDetails()">Enter Your Details</button>

<!-- Display output -->
<p id="greeting"></p>

<script>
function getUserDetails() {
    // Accept name
    let name = "";
    while (name.trim() === "") { // Loop until a valid name is entered
        name = prompt("Enter your name:");
        if (name.trim() === "") {
            alert("Name cannot be empty. Please enter your name.");
        }
    }

    // Accept age
    let age;
    while (true) { // Loop until a valid age is entered
        age = prompt("Enter your age:");
        if (age !== "" && !isNaN(age) && age > 0 && age <= 120) {
            age = Number(age); // Convert age to number
            break;
        } else {
            alert("Invalid age! Enter a number between 1 and 120.");
        }
    }

    // Conditional greeting based on age
    let message = "Hello " + name + "! ";

```

```

if (age < 18) {
    message += "You are a minor.";
} else if (age <= 60) {
    message += "You are an adult.";
} else {
    message += "You are a senior citizen.";
}

// Display greeting
document.getElementById("greeting").innerText = message;
}
</script>

```

</body>  
</html>

**6)Build a webpage with form validation using JavaScript functions, event handlers, string manipulation, and display of current date and time**

```

<!DOCTYPE html>
<html>
<head>
    <title>Simple Form Validation</title>
</head>
<body>

<h2>User Form</h2>

<form id="userForm" onsubmit="return validateForm()">
    Name: <input type="text" id="name"><br><br>
    Age: <input type="text" id="age"><br><br>
    <button type="submit">Submit</button>
</form>

<p id="greeting"></p>
<p id="dateTime"></p>

<script>
// Show current date and time
function showDateTime() {
    const now = new Date();
    document.getElementById("dateTime").innerText = "Current Date & Time: " +
now.toLocaleString();
}

// Call once on page load
showDateTime();

```

```

// Update every second
setInterval(showDateTime, 1000);

// Simple form validation
function validateForm() {
    const name = document.getElementById("name").value.trim();
    const age = document.getElementById("age").value.trim();
    const greeting = document.getElementById("greeting");

    // Validate name
    if(name === "") {
        alert("Please enter your name");
        return false;
    }

    // Validate age
    const ageNum = Number(age);
    if(isNaN(ageNum) || ageNum <= 0 || ageNum > 120) {
        alert("Please enter a valid age between 1 and 120");
        return false;
    }

    // Display greeting
    greeting.innerText = "Hello " + name + "! You are " + ageNum + " years old.";
    return false; // Prevent form submission
}
</script>

</body>
</html>

```

**7)Develop a simple AngularJS application to display a list of students with their marks using AngularJS expressions, directives, and filters.**

```

<!DOCTYPE html>
<html ng-app="studentApp">
<head>
    <title>Student List</title>
    <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
</head>
<body ng-controller="StudentController">

<h2>Students and Marks</h2>

<table border="1" cellpadding="5">
    <tr>

```

```

<th>Roll No</th>
<th>Name</th>
<th>Marks</th>
<th>Status</th>
</tr>
<tr ng-repeat="student in students">
  <td>{{ student.rollno }}</td>
  <td>{{ student.name }}</td>
  <td>{{ student.marks }}</td>
  <td>{{ student.marks }} >= 50 ? 'Pass' : 'Fail'</td>
</tr>
</table>

<script>
  var app = angular.module('studentApp', []);

  app.controller('StudentController', function($scope) {
    $scope.students = [
      { rollno: 101, name: 'Payal Pagire', marks: 85 },
      { rollno: 102, name: 'Amit Sharma', marks: 42 },
      { rollno: 103, name: 'Priya Desai', marks: 78 }
    ];
  });
</script>

```

```

</body>
</html>

```

**8)Create a form using AngularJS with data binding and validation that captures user details like name, email, and course, and displays the entered information in a formatted table.**

```

<!DOCTYPE html>
<html ng-app="userApp">
<head>
  <title>Simple User Form</title>
  <script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
</head>
<body ng-controller="UserController">

```

```

<h2>User Form</h2>

```

```

<form ng-submit="addUser()">

```

Name: <input type="text" ng-model="user.name" required><br><br>  
 Email: <input type="email" ng-model="user.email" required><br><br>  
 Course:

```

<select ng-model="user.course" required>
  <option value="">--Select--</option>
  <option>Computer Science</option>
  <option>IT</option>
  <option>Electronics</option>
</select><br><br>

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

<h3>Entered Users</h3>
<table border="1" cellpadding="5">
  <tr>
    <th>Name</th>
    <th>Email</th>
    <th>Course</th>
  </tr>
  <tr ng-repeat="u in users">
    <td>{{ u.name }}</td>
    <td>{{ u.email }}</td>
    <td>{{ u.course }}</td>
  </tr>
</table>

<script>
  var app = angular.module('userApp', []);
  app.controller('UserController', function($scope) {
    $scope.users = [];
    $scope.user = {};

    $scope.addUser = function() {
      if ($scope.user.name && $scope.user.email && $scope.user.course) {
        $scope.users.push(angular.copy($scope.user));
        $scope.user = {};// reset form
      } else {
        alert("Please fill all fields!");
      }
    };
  });
</script>

</body>
</html>

```

**9)Create a PHP application to take user input from a form, store it in an array, and display the data using different array functions.**

```

<?php
// Initialize an array to store user data
$users = [];

// Check if form is submitted
if (isset($_POST['submit'])) {
    // Get form data
    $name = $_POST['name'];
    $email = $_POST['email'];
    $age = $_POST['age'];

    // Store data in associative array
    $user = [
        "name" => $name,
        "email" => $email,
        "age" => $age
    ];

    // Add user to users array
    $users[] = $user;

    // For demonstration, save the data in session (optional)
    session_start();
    if (isset($_SESSION['users'])) {
        $users = $_SESSION['users'];
    }
    $users[] = $user;
    $_SESSION['users'] = $users;
}

?>

<!DOCTYPE html>
<html>
<head>
    <title>PHP Array Example</title>
</head>
<body>

<h2>User Input Form</h2>

<form method="post" action="">
    Name: <input type="text" name="name" required><br><br>
    Email: <input type="email" name="email" required><br><br>
    Age: <input type="number" name="age" required><br><br>
    <input type="submit" name="submit" value="Submit">

```

```

</form>

<?php
// Display stored data using different array functions
if (!empty($users)) {
    echo "<h2>Stored User Data</h2>";

    // Example 1: Display all users using print_r
    echo "<h3>Using print_r:</h3>";
    echo "<pre>";
    print_r($users);
    echo "</pre>";

    // Example 2: Display all users using foreach
    echo "<h3>Using foreach:</h3>";
    foreach ($users as $index => $user) {
        echo "User " . ($index + 1) . ": ";
        echo $user['name'] . ", " . $user['email'] . ", " . $user['age'] . "<br>";
    }

    // Example 3: Display all names using array_column
    echo "<h3>Using array_column (Names only):</h3>";
    $names = array_column($users, 'name');
    echo implode(", ", $names);

    // Example 4: Count number of users
    echo "<h3>Total Users: " . count($users) . "</h3>";
}
?>

```

```

</body>
</html>

```

**10)Develop a simple PHP-based login system using sessions and cookies that verifies a username and password and displays a welcome page upon successful login**

```

<?php
session_start();

// Hardcoded credentials for demo
$valid_username = "admin";
$valid_password = "1234";

// Check form submission
if (isset($_POST['login'])) {
    $username = $_POST['username'];
    $password = $_POST['password'];

```

```

if ($username === $valid_username && $password === $valid_password) {
    $_SESSION['username'] = $username;
    header("Location: welcome.php");
    exit();
} else {
    $error = "Invalid username or password!";
}
?>

<!DOCTYPE html>
<html>
<head>
    <title>Login</title>
</head>
<body>

<h2>Login</h2>

<?php
if (isset($error)) {
    echo "<p style='color:red;'>$error</p>";
}
?>

<form method="post" action="">
    Username: <input type="text" name="username" required><br><br>
    Password: <input type="password" name="password" required><br><br>
    <input type="submit" name="login" value="Login">
</form>

</body>
</html>

_____
<?php
session_start();

// Redirect if not logged in
if (!isset($_SESSION['username'])) {
    header("Location: login.php");
    exit();
}
?>

```

```
<!DOCTYPE html>
<html>
<head>
    <title>Welcome</title>
</head>
<body>

<h2>Welcome Page</h2>
<p>Hello, <?php echo $_SESSION['username']; ?>!</p>

<p><a href="logout.php">Logout</a></p>

</body>
</html>
```

---

```
<?php
session_start();
session_destroy(); // End session
header("Location: login.php");
exit();
?>
```