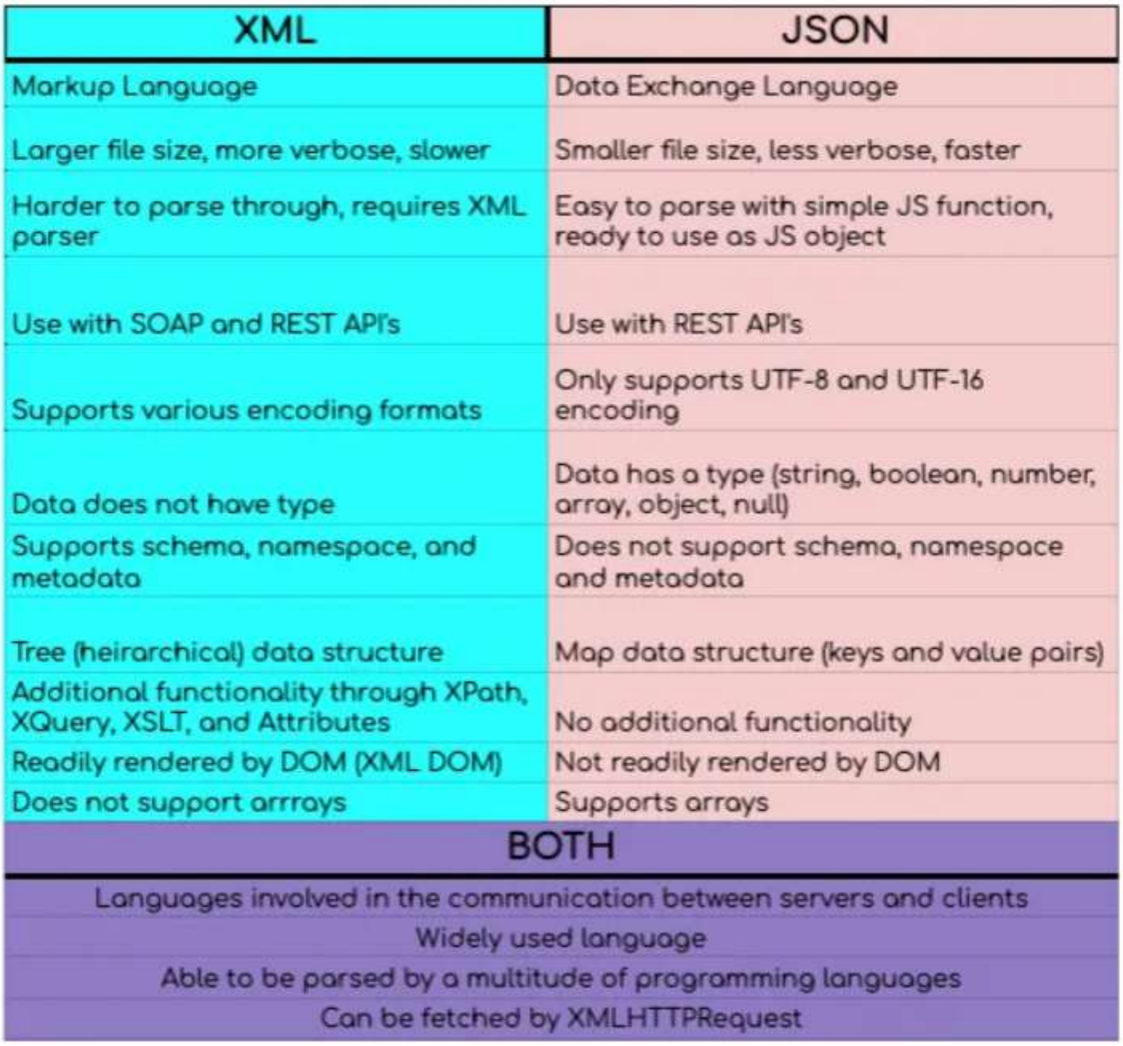
Today's task:

1)Difference between JSON and XML



XML	JSON
Markup Language	Data Exchange Language
Larger file size, more verbose, slower	Smaller file size, less verbose, faster
Harder to parse through, requires XML parser	Easy to parse with simple JS function, ready to use as JS object
Use with SOAP and REST API's	Use with REST API's
Supports various encoding formats	Only supports UTF-8 and UTF-16 encoding
Data does not have type	Data has a type (string, boolean, number, array, object, null)
Supports schema, namespace, and metadata	Does not support schema, namespace and metadata
Tree (heirarchical) data structure	Map data structure (keys and value pairs)
Additional functionality through XPath, XQuery, XSLT, and Attributes	No additional functionality
Readily rendered by DOM (XML DOM)	Not readily rendered by DOM
Does not support arrrays	Supports arrays
BOTH	
Languages involved in the communication between servers and clients	
Widely used language	
Able to be parsed by a multitude of programming languages	
Can be fetched by XMLHTTPRequest	

2)Create 3 XML and JSON files for department,year,student.

XML Files

1. Department XML

```
<?xml version="1.0" encoding="UTF-8"?>
<departments>
    <department>
        <id>1</id>
        <name>Computer Science</name>
        <head>Dr. Alice Smith</head>
```

```
  </department>
  <department>
    <id>2</id>
    <name>Mathematics</name>
    <head>Dr. John Doe</head>
  </department>
  <department>
    <id>3</id>
    <name>Physics</name>
    <head>Dr. Jane Johnson</head>
  </department>
</departments>
```

2. Year XML

```
<?xml version="1.0" encoding="UTF-8"?>
<years>
  <year>
    <id>1</id>
    <name>Freshman</name>
  </year>
  <year>
    <id>2</id>
    <name>Sophomore</name>
  </year>
  <year>
    <id>3</id>
    <name>Junior</name>
  </year>
```

```xml
    <year>
        <id>4</id>
        <name>Senior</name>
    </year>
</years>
```

3. Student XML

```xml
<?xml version="1.0" encoding="UTF-8"?>
<students>
    <student>
        <id>1001</id>
        <name>Emily Brown</name>
        <age>20</age>
        <department>Computer Science</department>
        <year>Junior</year>
    </student>
    <student>
        <id>1002</id>
        <name>Michael Green</name>
        <age>19</age>
        <department>Mathematics</department>
        <year>Sophomore</year>
    </student>
    <student>
        <id>1003</id>
        <name>Sarah White</name>
        <age>21</age>
        <department>Physics</department>
    <year>Senior</year>
  </student>
</students>
```

JSON Files

1. Department JSON

```
{
  "departments": [
    {
      "id": 1,
      "name": "Computer Science",
      "head": "Dr. Alice Smith"
    },
    {
      "id": 2,
      "name": "Mathematics",
      "head": "Dr. John Doe"
    },
    {
      "id": 3,
      "name": "Physics",
      "head": "Dr. Jane Johnson"
    }
  ]
}
```

2. Year JSON

```
{
  "years": [
        {
            "id": 1,
            "name": "Freshman"
        },
        {
            "id": 2,
            "name": "Sophomore"
        },
        {
            "id": 3,
            "name": "Junior"
        },
        {
            "id": 4,
            "name": "Senior"
        }
    ]
}
```

3. Student JSON

```
{
    "students": [
        {
            "id": 1001,
            "name": "Emily Brown",
            "age": 20,
            "department": "Computer Science",
            "year": "Junior"
        },
        {
            "id": 1002,
            "name": "Michael Green",
            "age": 19,
            "department": "Mathematics",
            "year": "Sophomore"
        },
        {
            "id": 1003,
            "name": "Sarah White",
            "age": 21,
            "department": "Physics",
            "year": "Senior"
        }
    ]
}
```

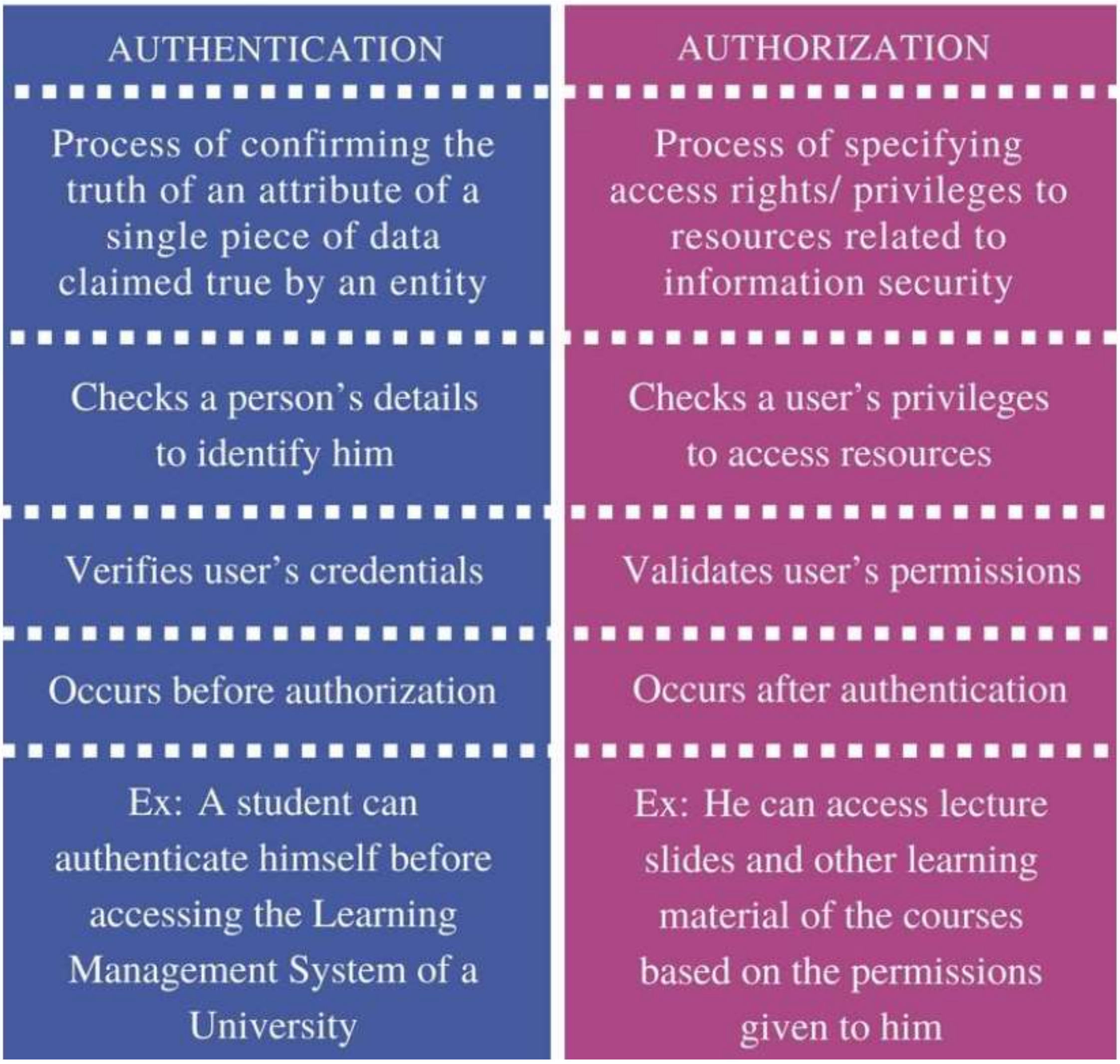
3)Create a file with depertment as root,year as subroot and student as an element

```
<?xml version="1.0" encoding="UTF-8"?>
<departments>
    <department>
        <id>1</id>
        <name>Computer Science</name>
        <head>Dr. Alice Smith</head>
        <years>
            <year>
```

```
      <id>3</id>
      <name>Junior</name>
      <students>
        <student>
          <id>1001</id>
          <name>Emily Brown</name>
          <age>20</age>
        </student>
      </students>
    </year>
  </years>
</department>
<department>
  <id>2</id>
  <name>Mathematics</name>
  <head>Dr. John Doe</head>
  <years>
    <year>
      <id>2</id>
      <name>Sophomore</name>
      <students>
        <student>
          <id>1002</id>
          <name>Michael Green</name>
          <age>19</age>
        </student>
      </students>
```

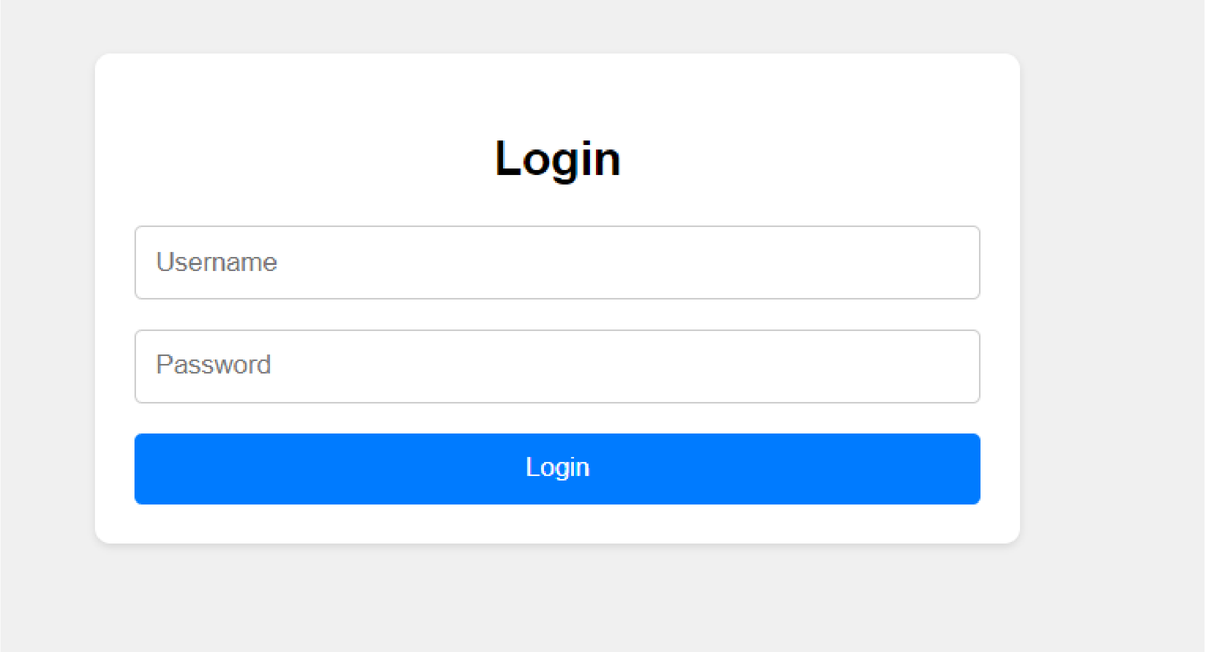
```xml
            </year>
        </years>
    </department>
    <department>
        <id>3</id>
        <name>Physics</name>
        <head>Dr. Jane Johnson</head>
        <years>
            <year>
                <id>4</id>
                <name>Senior</name>
                <students>
                    <student>
                        <id>1003</id>
                        <name>Sarah White</name>
                        <age>21</age>
                    </student>
                </students>
            </year>
        </years>
    </department>
</departments>
```

4)Difference between Authorization and Authentication

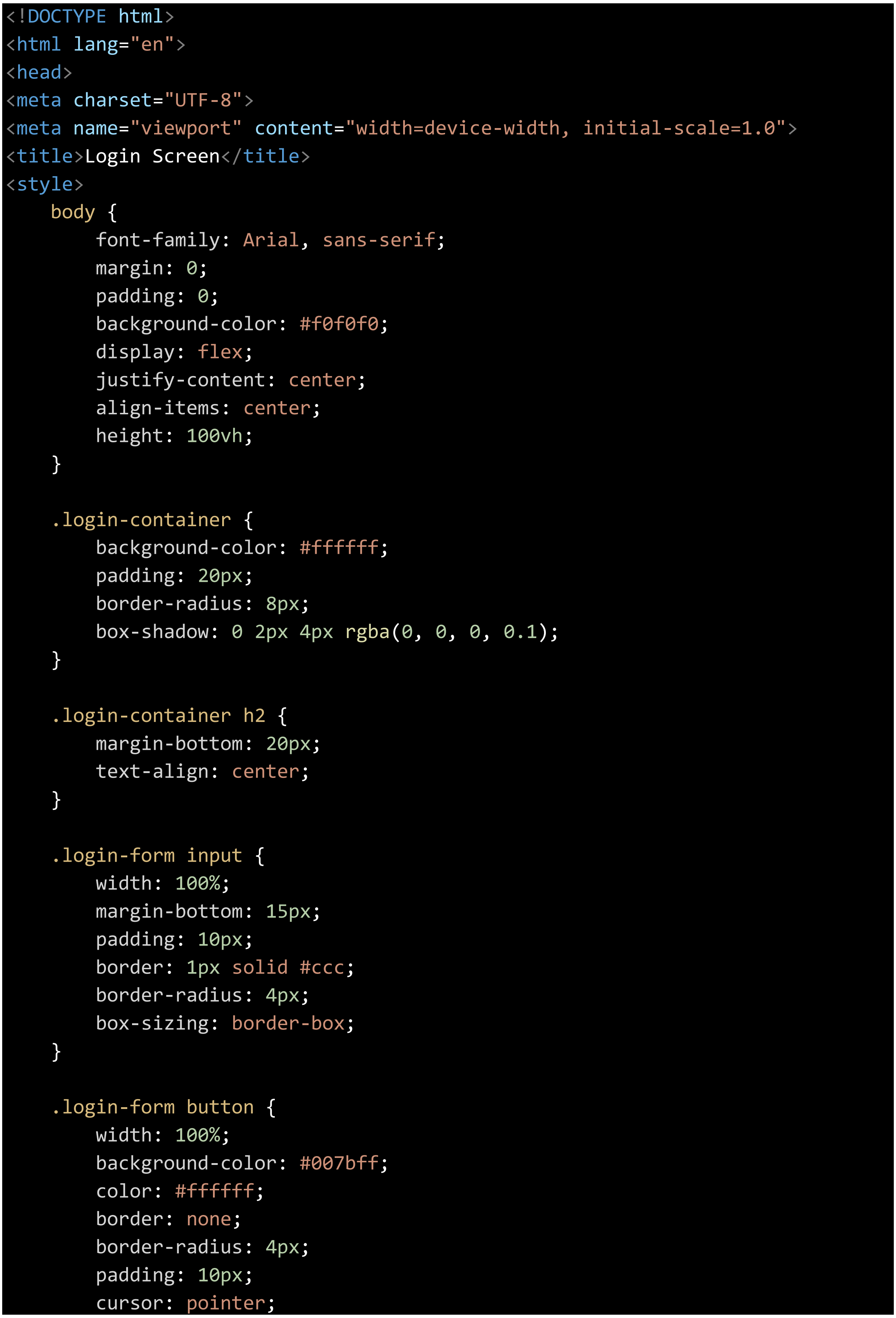


AUTHENTICATION	AUTHORIZATION
Process of confirming the truth of an attribute of a single piece of data claimed true by an entity	Process of specifying access rights/ privileges to resources related to information security
Checks a person’s details to identify him	Checks a user’s privileges to access resources
Verifies user’s credentials	Validates user’s permissions
Occurs before authorization	Occurs after authentication
Ex: A student can authenticate himself before accessing the Learning Management System of a University	Ex: He can access lecture slides and other learning material of the courses based on the permissions given to him

5)Create a Login Screen



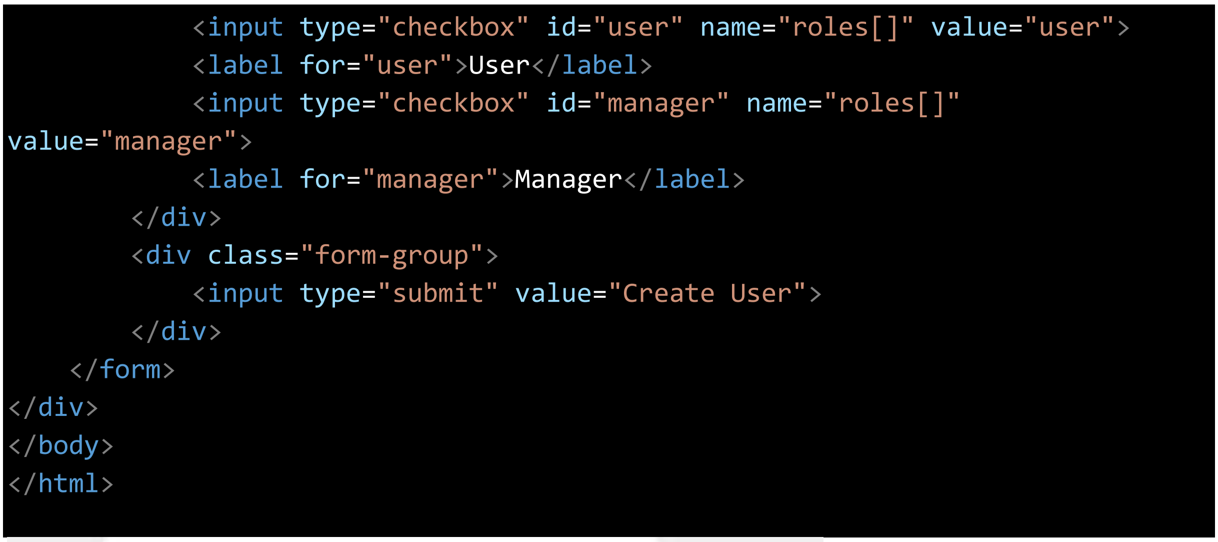
Code:



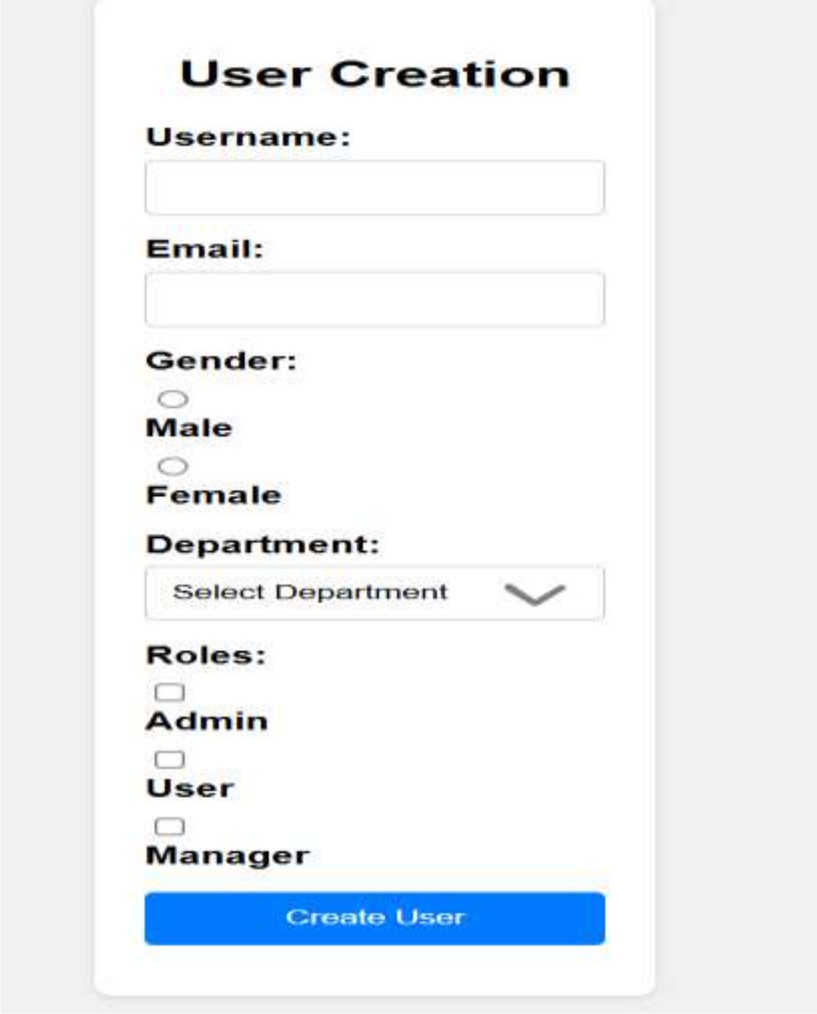
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Login Screen</title>
<style>
    body {
        font-family: Arial, sans-serif;
        margin: 0;
        padding: 0;
        background-color: #f0f0f0;
        display: flex;
        justify-content: center;
        align-items: center;
        height: 100vh;
    }

    .login-container {
        background-color: #ffffff;
        padding: 20px;
        border-radius: 8px;
        box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
    }

    .login-container h2 {
        margin-bottom: 20px;
        text-align: center;
    }

    .login-form input {
        width: 100%;
        margin-bottom: 15px;
        padding: 10px;
        border: 1px solid #ccc;
        border-radius: 4px;
        box-sizing: border-box;
    }

    .login-form button {
        width: 100%;
        background-color: #007bff;
        color: #ffffff;
        border: none;
        border-radius: 4px;
        padding: 10px;
        cursor: pointer;
    }

    .login-form button:hover {
        background-color: #0056b3;
    }
</style>
</head>
<body>
<div class="login-container">
    <h2>Login</h2>
    <form class="login-form" action="#">
        <input type="text" name="username" placeholder="Username" required>
        <input type="password" name="password" placeholder="Password"
required>
        <button type="submit">Login</button>
    </form>
</div>
</body>
</html>
```

6)Create a User Creation Screen by using all elements in it(like List, Radio button, Drop down, CheckBox)

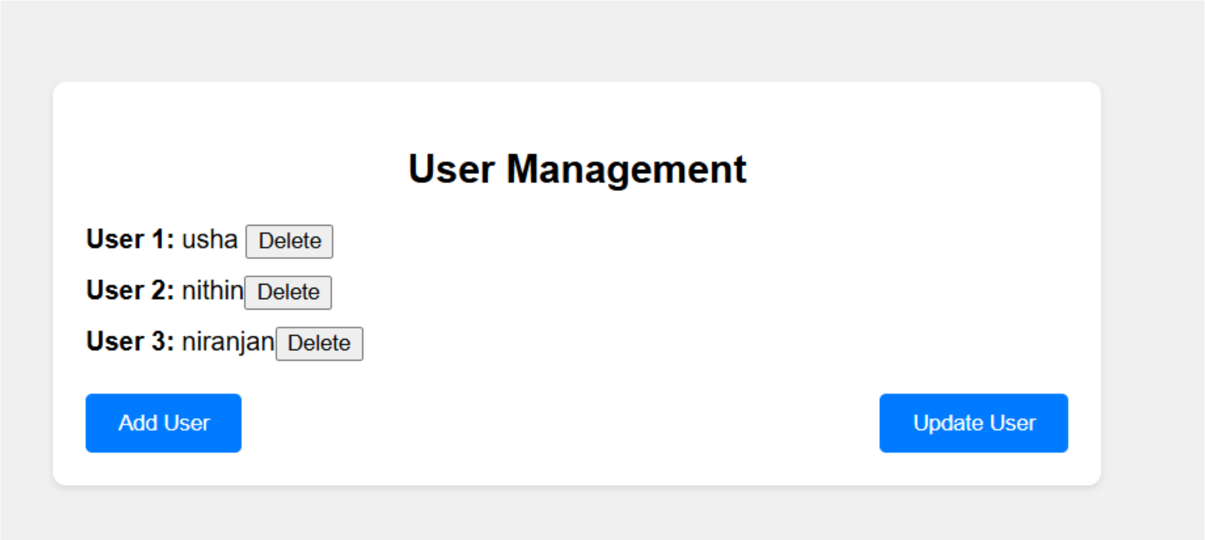
```
<!DOCTYPE html>

<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>User Creation Screen</title>
<style>
    body {
        font-family: Arial, sans-serif;
        margin: 0;
        padding: 0;
        background-color: #f0f0f0;
        display: flex;
        justify-content: center;
        align-items: center;
        height: 100vh;
    }

    .user-form {
        background-color: #ffffff;
        padding: 20px;
        border-radius: 8px;
        box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
    }
    .user-form h2 {
        margin-bottom: 20px;
        text-align: center;
    }

    .form-group {
        margin-bottom: 15px;
    }

    .form-group label {
        display: block;
        font-weight: bold;
        margin-bottom: 5px;
    }

    .form-group input[type="text"],
    .form-group input[type="email"],
    .form-group select {
        width: 100%;
        padding: 10px;
        border: 1px solid #ccc;
        border-radius: 4px;
        box-sizing: border-box;
    }

    .form-group select {
        appearance: none;
        -webkit-appearance: none;
        background-image: url('data:image/svg+xml;utf8,<svg fill="%237f7f7f"
viewBox="0 0 20 20" xmlns="http://www.w3.org/2000/svg"><path d="M3.293 6.707a1
1 0 011.414 0L10 12.586l5.293-5.293a1 1 0 111.414 1.414l-6 6a1 1 0 01-1.414
0l-6-6a1 1 0 010-1.414z"/></svg>');
        background-repeat: no-repeat;
        background-position: right 10px top 50%;
        padding-right: 30px;
    }

    .form-group select:focus {
        outline: none;
        border-color: #007bff;
    }

    .form-group input[type="radio"],
    .form-group input[type="checkbox"] {
        margin-right: 5px;
        margin-top: 5px;
    }
    .form-group input[type="submit"] {
        width: 100%;
        background-color: #007bff;
        color: #ffffff;
        border: none;
        border-radius: 4px;
        padding: 10px;
        cursor: pointer;
    }

    .form-group input[type="submit"]:hover {
        background-color: #0056b3;
    }
</style>
</head>
<body>
<div class="user-form">
    <h2>User Creation</h2>
    <form action="#" method="post">
        <div class="form-group">
            <label for="username">Username:</label>
            <input type="text" id="username" name="username" required>
        </div>
        <div class="form-group">
            <label for="email">Email:</label>
            <input type="email" id="email" name="email" required>
        </div>
        <div class="form-group">
            <label for="gender">Gender:</label>
            <input type="radio" id="male" name="gender" value="male" required>
            <label for="male">Male</label>
            <input type="radio" id="female" name="gender" value="female"
required>
            <label for="female">Female</label>
        </div>
        <div class="form-group">
            <label for="department">Department:</label>
            <select id="department" name="department" required>
                <option value="">Select Department</option>
                <option value="IT">IT</option>
                <option value="HR">HR</option>
                <option value="Finance">Finance</option>
            </select>
        </div>
        <div class="form-group">
            <label for="roles">Roles:</label>
            <input type="checkbox" id="admin" name="roles[]" value="admin">
            <label for="admin">Admin</label>
```



```
            <input type="checkbox" id="user" name="roles[]" value="user">
            <label for="user">User</label>
            <input type="checkbox" id="manager" name="roles[]"
value="manager">
            <label for="manager">Manager</label>
        </div>
        <div class="form-group">
            <input type="submit" value="Create User">
        </div>
    </form>
</div>
</body>
</html>
```



7)List all Users,Update user and Delete user(Popup for confirmation eg:Are you sure do you want to delete)

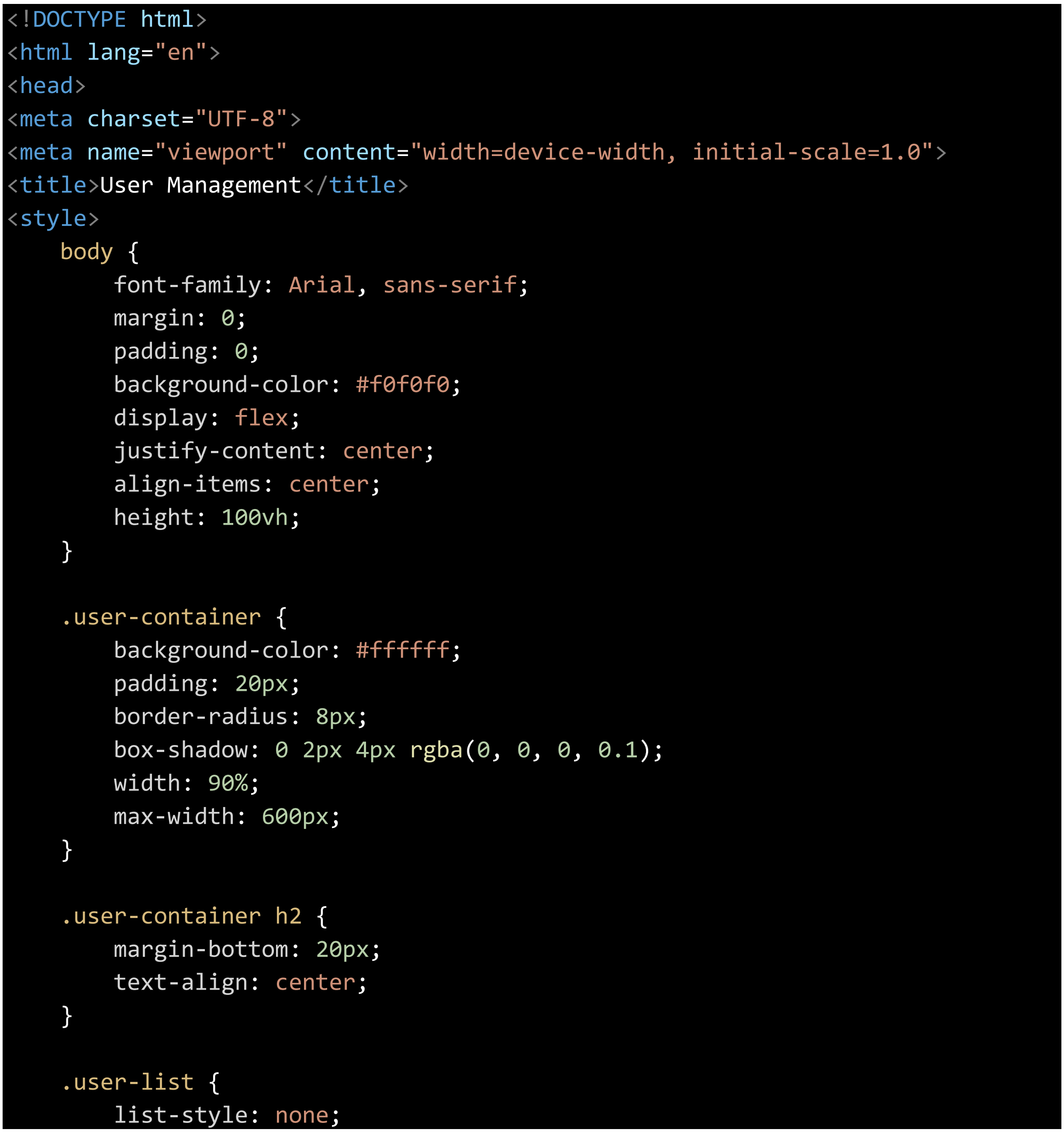


User Management

User 1: usha

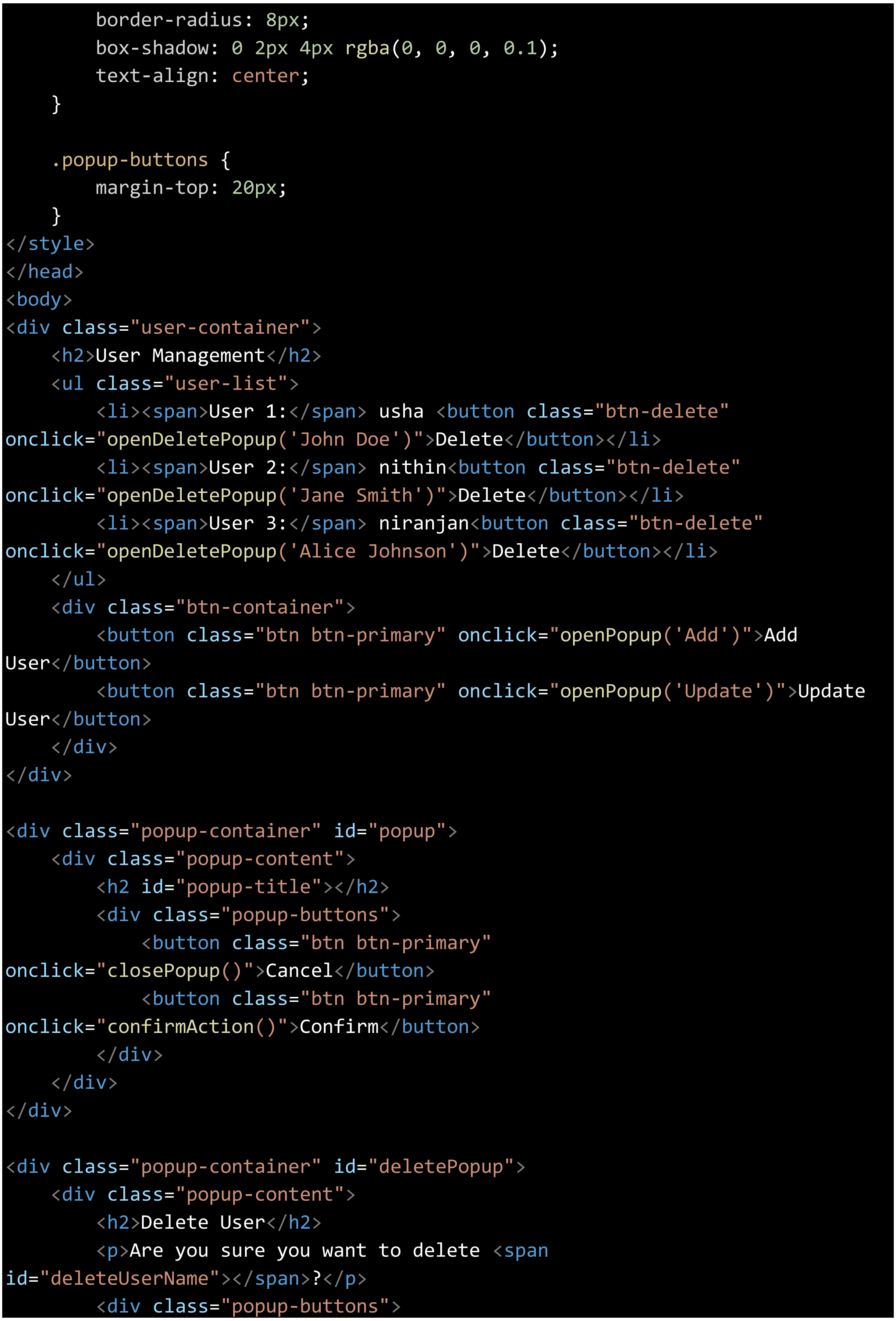
User 2: nithin

User 3: niranjan

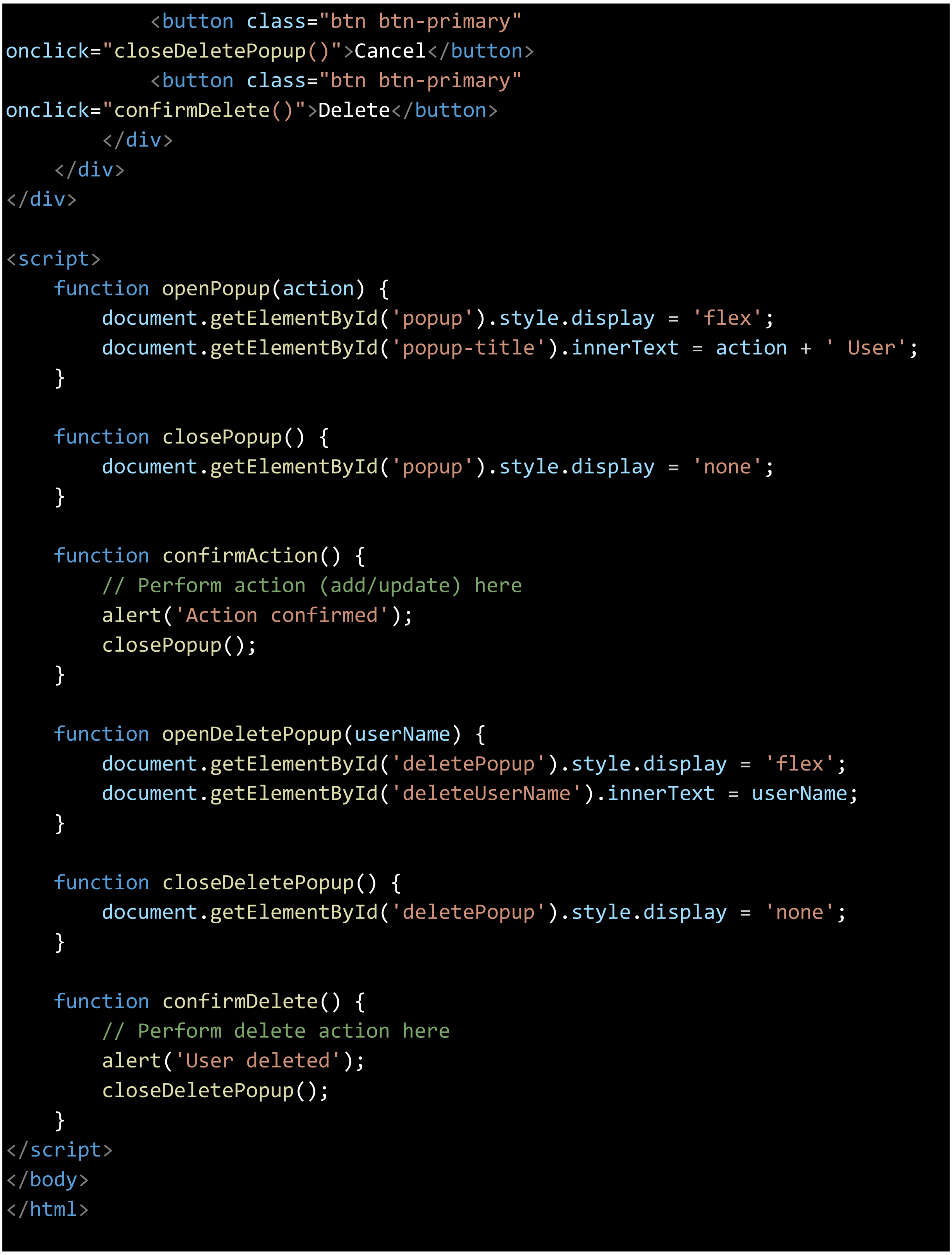


```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>User Management</title>
<style>
    body {
        font-family: Arial, sans-serif;
        margin: 0;
        padding: 0;
        background-color: #f0f0f0;
        display: flex;
        justify-content: center;
        align-items: center;
        height: 100vh;
    }

    .user-container {
        background-color: #ffffff;
        padding: 20px;
        border-radius: 8px;
        box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
        width: 90%;
        max-width: 600px;
    }

    .user-container h2 {
        margin-bottom: 20px;
        text-align: center;
    }

    .user-list {
        list-style: none;
```

```css
        padding: 0;
    }

    .user-list li {
        margin-bottom: 10px;
    }

    .user-list li span {
        font-weight: bold;
    }

    .btn-container {
        display: flex;
        justify-content: space-between;
        margin-top: 20px;
    }

    .btn {
        padding: 10px 20px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
    }

    .btn-primary {
        background-color: #007bff;
        color: #ffffff;
    }

    .btn-primary:hover {
        background-color: #0056b3;
    }

    .popup-container {
        display: none;
        position: fixed;
        top: 0;
        left: 0;
        width: 100%;
        height: 100%;
        background-color: rgba(0, 0, 0, 0.5);
        justify-content: center;
        align-items: center;
    }

    .popup-content {
        background-color: #ffffff;
        padding: 20px;
```

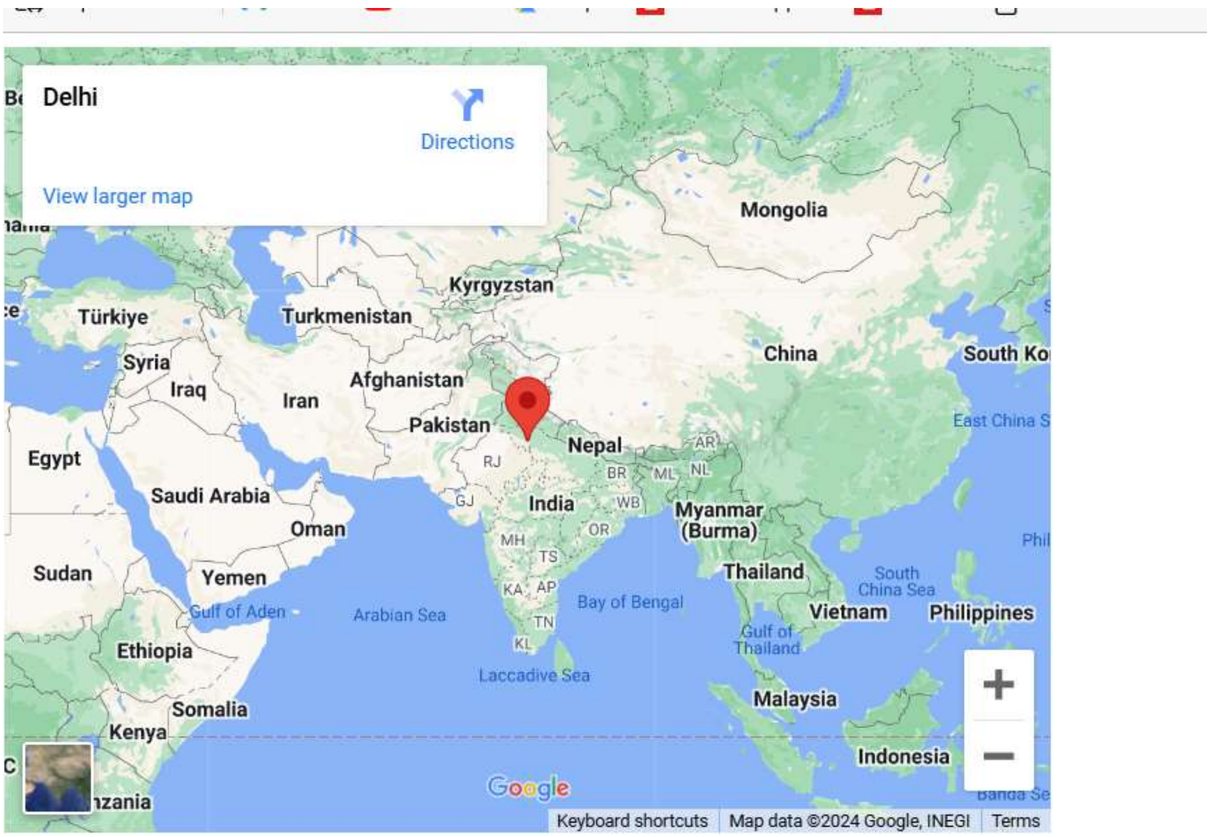


```
        border-radius: 8px;
        box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
        text-align: center;
    }

    .popup-buttons {
        margin-top: 20px;
    }
</style>
</head>
<body>
<div class="user-container">
    <h2>User Management</h2>
    <ul class="user-list">
        <li><span>User 1:</span> usha <button class="btn-delete"
onclick="openDeletePopup('John Doe')">Delete</button></li>
        <li><span>User 2:</span> nithin<button class="btn-delete"
onclick="openDeletePopup('Jane Smith')">Delete</button></li>
        <li><span>User 3:</span> niranjan<button class="btn-delete"
onclick="openDeletePopup('Alice Johnson')">Delete</button></li>
    </ul>
    <div class="btn-container">
        <button class="btn btn-primary" onclick="openPopup('Add')">Add
User</button>
        <button class="btn btn-primary" onclick="openPopup('Update')">Update
User</button>
    </div>
</div>

<div class="popup-container" id="popup">
    <div class="popup-content">
        <h2 id="popup-title"></h2>
        <div class="popup-buttons">
            <button class="btn btn-primary"
onclick="closePopup()">Cancel</button>
            <button class="btn btn-primary"
onclick="confirmAction()">Confirm</button>
        </div>
    </div>
</div>

<div class="popup-container" id="deletePopup">
    <div class="popup-content">
        <h2>Delete User</h2>
        <p>Are you sure you want to delete <span
id="deleteUserName"></span>?</p>
        <div class="popup-buttons">
```



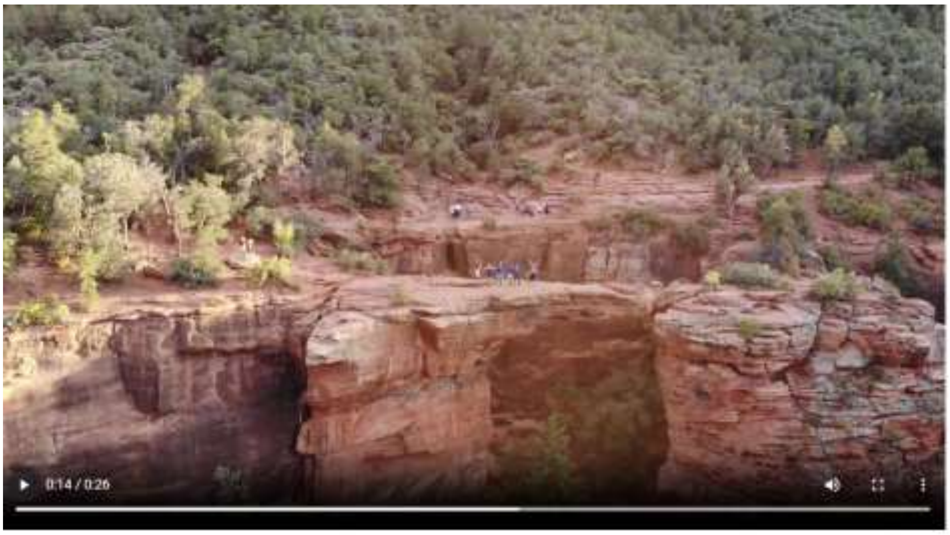
```
            <button class="btn btn-primary"
onclick="closeDeletePopup()">Cancel</button>
            <button class="btn btn-primary"
onclick="confirmDelete()">Delete</button>
        </div>
    </div>
</div>

<script>
    function openPopup(action) {
        document.getElementById('popup').style.display = 'flex';
        document.getElementById('popup-title').innerText = action + ' User';
    }

    function closePopup() {
        document.getElementById('popup').style.display = 'none';
    }

    function confirmAction() {
        // Perform action (add/update) here
        alert('Action confirmed');
        closePopup();
    }

    function openDeletePopup(userName) {
        document.getElementById('deletePopup').style.display = 'flex';
        document.getElementById('deleteUserName').innerText = userName;
    }

    function closeDeletePopup() {
        document.getElementById('deletePopup').style.display = 'none';
    }

    function confirmDelete() {
        // Perform delete action here
        alert('User deleted');
        closeDeletePopup();
    }
</script>
</body>
</html>
```

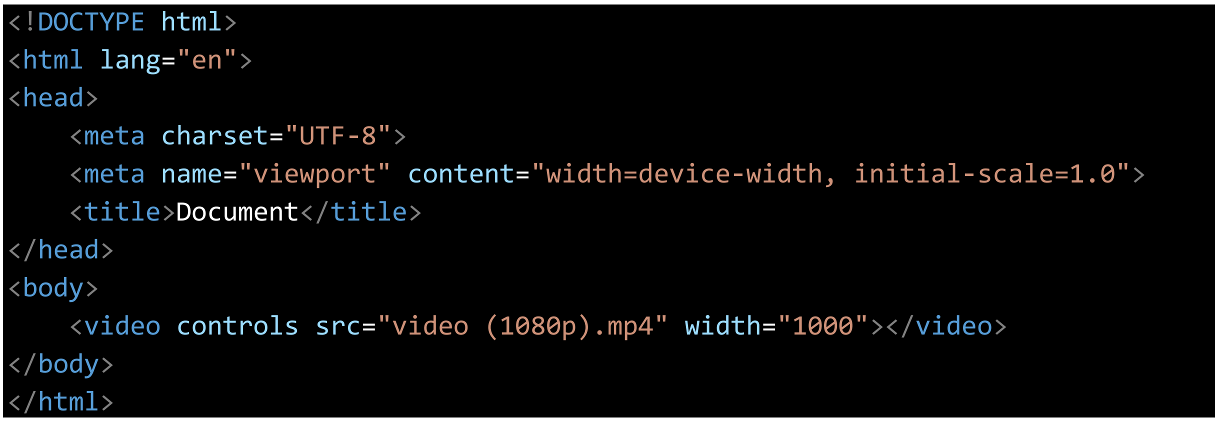
8)Create a HTML page with Google Map



```
<!DOCTYPE html>
<html>
<head>
    <title>Google Maps Example</title>
    <style>
        /* Set the size of the map container */
        #map {
            height: 400px;  /* The height can be changed as needed */
            width: 100%;    /* The width can be changed as needed */
        }
    </style>
</head>
<body>
    <iframe
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d448193.9510303648!2
d76.76356891374634!3d28.644287353477026!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.
1!3m3!1m2!1s0x390cfd5b347eb62d%3A0x37205b715389640!2sDelhi!5e0!3m2!1sen!2sin!4
v1716360464880!5m2!1sen!2sin" width="600" height="450" style="border:0;"
allowfullscreen="" loading="lazy" referrerpolicy="no-referrer-when-
downgrade"></iframe>
</body>
</html>
```

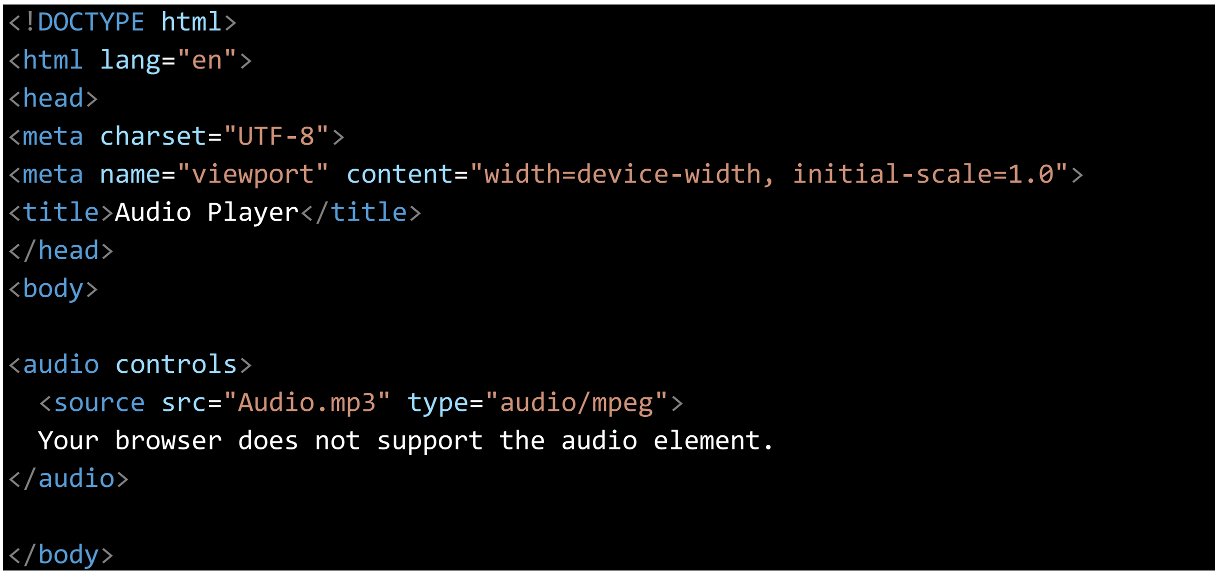
9)Create a HTML page with Video file



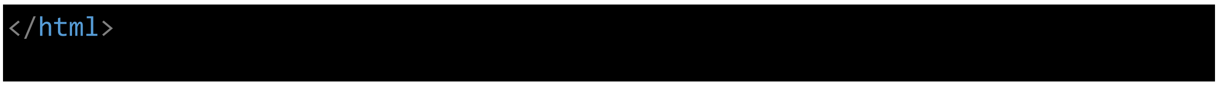


```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <video controls src="video (1080p).mp4" width="1000"></video>
</body>
</html>
```

10)Create a HTML page with Audio file

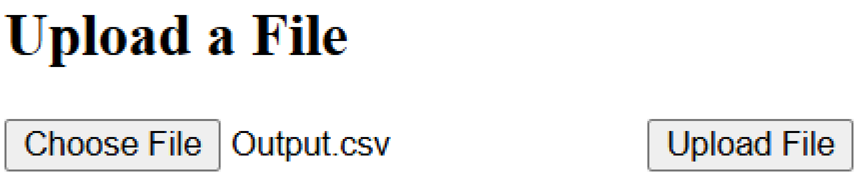


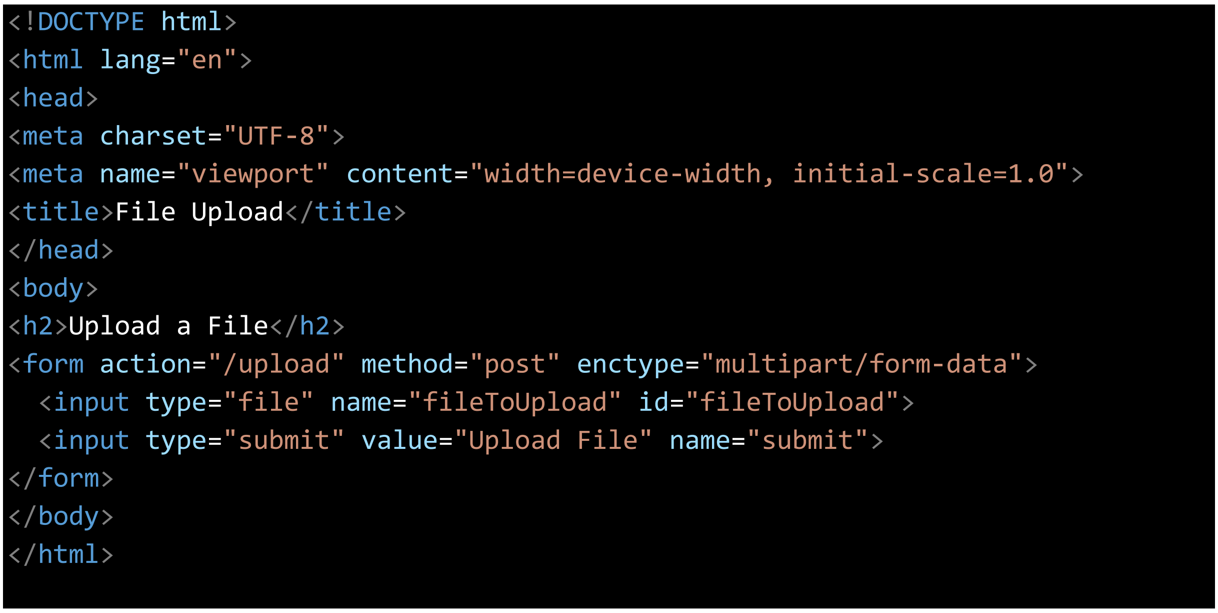
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Audio Player</title>
</head>
<body>

<audio controls>
  <source src="Audio.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>

</body>
```



```
</html>
```

11)Create a HTML page to upload a file





```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>File Upload</title>
</head>
<body>
<h2>Upload a File</h2>
<form action="/upload" method="post" enctype="multipart/form-data">
  <input type="file" name="fileToUpload" id="fileToUpload">
  <input type="submit" value="Upload File" name="submit">
</form>
</body>
</html>
```