Name : Soham Satpute **Class: D15A** **Roll No. 51**

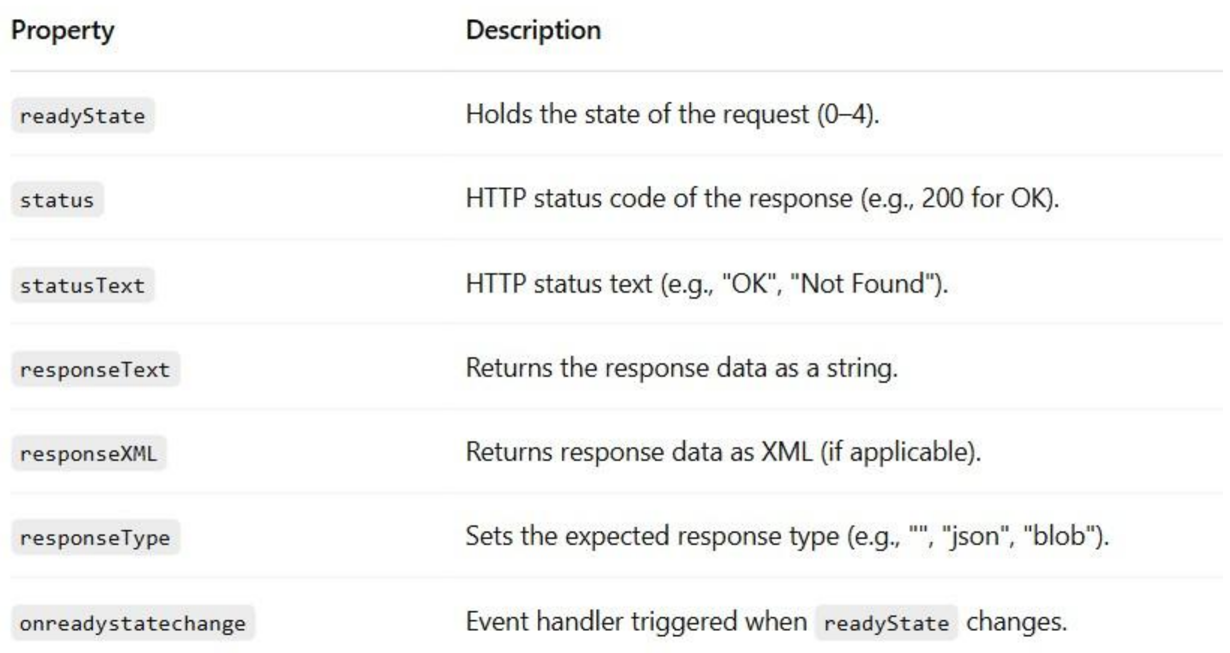
Experiment – 6: AJAX

Aim: To study AJAX

Theory:
How do Synchronous and Asynchronous Requests differ?
Ans:
Q1) In a Synchronous request, the request sent from the client receives the response in the same HTTP connection. Examples are responses from Internet gateway, phone calls, and video meetings.

Whereas for Asynchronous requests, multiple requests can be sent from clients and their responses can be received in subsequent connections. Examples are collaborative documents like assessments, online queries, emails, and online forums.

Q2) Describe various properties and methods used in XMLHttpRequest Object
Ans:



Property	Description
`readyState`	Holds the state of the request (0–4).
`status`	HTTP status code of the response (e.g., 200 for OK).
`statusText`	HTTP status text (e.g., "OK", "Not Found").
`responseText`	Returns the response data as a string.
`responseXML`	Returns response data as XML (if applicable).
`responseType`	Sets the expected response type (e.g., "", "json", "blob").
`onreadystatechange`	Event handler triggered when `readyState` changes.

Problem Statement:
Create a registration page having fields like Name, College, Username and Password (read password twice).
Validate the form by checking for
1. Username is not same as existing entries
2 Retyped password is matching with the earlier one. Prompt a message is 3
Auto suggest college names.
Show the message "Successfully Registered" on the same page below the submit button, on Successfully registration.

Name : Soham Satpute Class: D15A Roll No. 51

Let all the updations on the page be Asynchronously loaded. Implement the same using XMLHttpRequest Object.

Output:

register.html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport"
content="width=device-width, initial-
scale=1.0">
   <title>Registration Form</title>
   <link rel="stylesheet" href="styles.css">
</head>
<body>
   <div class="container">
      <h2>Registration Form</h2>
      <form id="registrationForm">
         <div class="form-group">
            <label for="name">Name:</label>
            <input type="text" id="name"
name="name" required>
            <span class="error"
id="nameError"></span>
         </div>

         <div class="form-group">
            <label
for="college">College:</label>
            <input type="text" id="college"
name="college" autocomplete="off">
            <div id="collegeSuggestions"
class="suggestions"></div>
         </div>

         <div class="form-group">
            <label
for="username">Username:</label>
            <input type="text" id="username"
name="username" required>
            <span class="error"
id="usernameError"></span>
         </div>

         <div class="form-group">
            <label
for="password">Password:</label>
            <input type="password"
id="password" name="password" required>
            <span class="error"
id="passwordError"></span>
         </div>

         <div class="form-group">
            <label
for="confirmPassword">Confirm
Password:</label>
            <input type="password"
id="confirmPassword"
name="confirmPassword" required>
            <span class="error"
id="confirmPasswordError"></span>
         </div>

         <button type="submit"
id="submitBtn">Register</button>
      </form>

      <div id="registrationMessage"
class="message"></div>
   </div>

   <script src="users.js"></script>
   <script src="script.js"></script>
</body>
</html>
```

Name : Soham Satpute Class: D15A Roll No. 51

colleges.json
```
["VESIT", "DJ Sanghvi", "SPIT", "KJ Somaiya", "Thakur College"]
```

usernames.json
```
["Soham"]
```

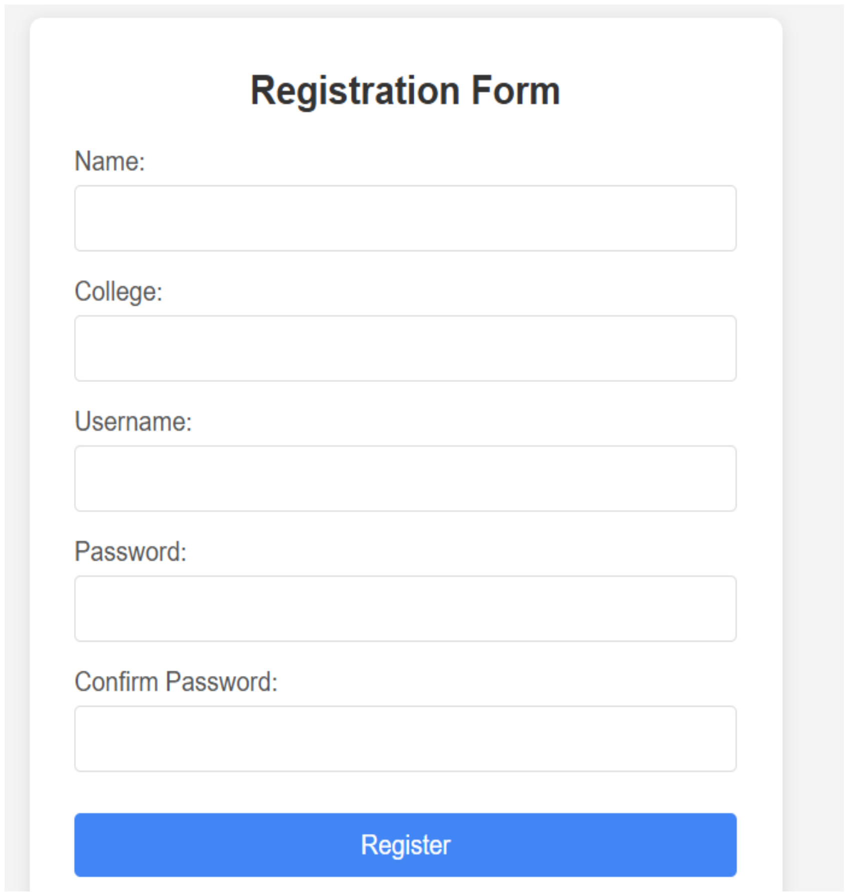
script.js
```
document.addEventListener('DOMContentLoaded'
, function() {
  const form =
document.getElementById('registrationForm');
  const nameInput =
document.getElementById('name');
  const collegeInput =
document.getElementById('college');
  const usernameInput =
document.getElementById('username');
  const passwordInput =
document.getElementById('password');
  const confirmPasswordInput =
document.getElementById('confirmPassword');
  const collegeSuggestions =
document.getElementById('collegeSuggestions');
  const registrationMessage =
document.getElementById('registrationMessage')
;

  // Sample college list (in a real application, this
would come from a database)
  const colleges = [
      "Massachusetts Institute of Technology",
      "Harvard University",
      "Stanford University",
      "California Institute of Technology",
      "University of Cambridge",
      "University of Oxford",
      "Princeton University",
      "Yale University",
      "Columbia University",
      "University of Chicago",
      "Imperial College London",
      "University of California, Berkeley",
      "University of Mumbai",
      "Indian Institute of Technology Bombay",
      "Delhi University",
      "Anna University",
      "Jadavpur University",
      "Banaras Hindu University",
      "Pune University"
  ];

  // College name auto-suggestion
  collegeInput.addEventListener('input', function()
{
      const query = this.value.toLowerCase();

      // Clear previous suggestions
      collegeSuggestions.innerHTML = '';

      if (query.length < 2) {
          collegeSuggestions.style.display = 'none';
          return;
      }

      // Filter colleges based on input
      const filteredColleges = colleges.filter(college
=>
          college.toLowerCase().includes(query)
      );
```

Name : Soham Satpute Class: D15A Roll No. 51

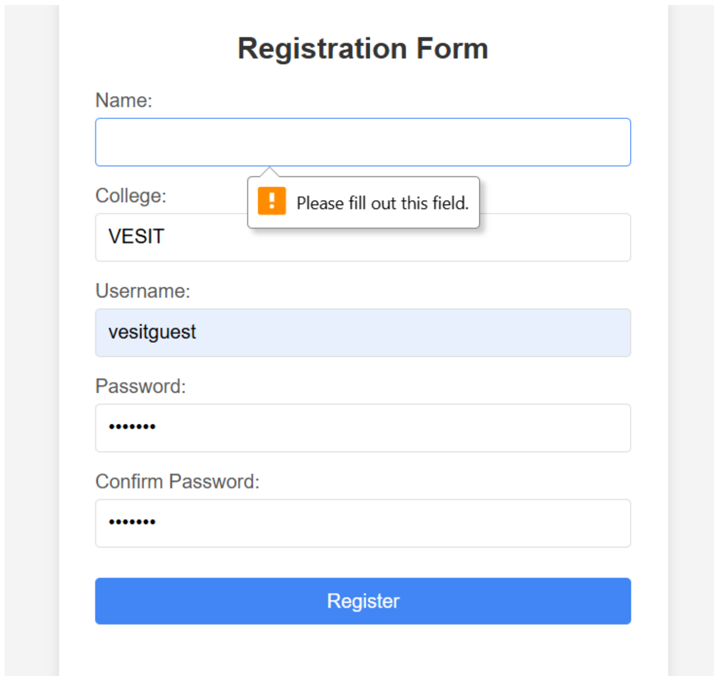
```
    if (filteredColleges.length > 0) {
        collegeSuggestions.style.display = 'block';

        filteredColleges.forEach(college => {
            const div =
document.createElement('div');
            div.className = 'suggestion-item';
            div.textContent = college;
            div.addEventListener('click', function() {
                collegeInput.value = college;
                collegeSuggestions.style.display =
'none';
            });
            collegeSuggestions.appendChild(div);
        });
    } else {
        collegeSuggestions.style.display = 'none';
    }
  });

  // Hide suggestions when clicking outside
  document.addEventListener('click', function(e) {
    if (e.target !== collegeInput) {
        collegeSuggestions.style.display = 'none';
    }
  });

  // Username validation (check if username
exists)
  usernameInput.addEventListener('blur',
function() {
    if (this.value.trim() === '') return;

    checkUsername(this.value);
  });

  function checkUsername(username) {
    // Simulate AJAX request with
XMLHttpRequest
    const xhr = new XMLHttpRequest();

    // Using a timeout to simulate network delay
    setTimeout(() => {
        // Check if username exists using our
UsersDB
        const exists =
UsersDB.usernameExists(username);

        if (exists) {
            document.getElementById('usernameErr
or').textContent = 'Username already exists.
Please choose another one.';
        } else {
            document.getElementById('usernameErr
or').textContent = '';
        }
    }, 300);
  }

  // Form submission
  form.addEventListener('submit', function(e) {
    e.preventDefault();

    // Reset error messages
    document.getElementById('nameError').textC
ontent = '';
    document.getElementById('usernameError').t
extContent = '';
    document.getElementById('passwordError').t
extContent = '';
    document.getElementById('confirmPassword
Error').textContent = '';

    // Client-side validation
    let isValid = true;

    // Name validation
    if (nameInput.value.trim() === '') {
      document.getElementById('nameError').textContent = 'Name cannot be empty';
      isValid = false;
    }

    // Password matching validation
    if (passwordInput.value !== confirmPasswordInput.value) {
      document.getElementById('confirmPasswordError').textContent = 'Passwords do not match';
      isValid = false;
    }

    // Username validation
    if (UsersDB.usernameExists(usernameInput.value)) {
      document.getElementById('usernameError').textContent = 'Username already exists. Please choose another one.';
      isValid = false;
    }

    if (isValid) {
      // Submit form using AJAX
      submitForm();
    }
  });

  function submitForm() {
    // Create a new XMLHttpRequest to simulate AJAX form submission
    const xhr = new XMLHttpRequest();

    // Simulate opening a connection
    xhr.open('POST', 'register', true);

    // Create a timeout to simulate network delay
    setTimeout(() => {
      try {
        // Get form data
        const userData = {
          name: nameInput.value.trim(),
          college: collegeInput.value.trim(),
          username: usernameInput.value.trim(),
          password: passwordInput.value
        };

        // Add user to our database
        const success = UsersDB.addUser(userData);

        if (success) {
          // Show success message
          registrationMessage.className = 'message success';
          registrationMessage.textContent = 'Successfully Registered';

          // Reset form
          form.reset();
        } else {
          // Show error message
          registrationMessage.className = 'message error-message';
          registrationMessage.textContent = 'Registration failed. Please try again.';
        }
      } catch (e) {
            registrationMessage.className =
'message error-message';
            registrationMessage.textContent = 'An
error occurred. Please try again.';
            console.error(e);
        }
    }, 500);
 }
```

```
});
```

Output:



Name : Soham Satpute **Class: D15A** **Roll No. 51**


Registration Form

Name:

Please fill out this field.

College:

VESIT

Username:

vesitguest

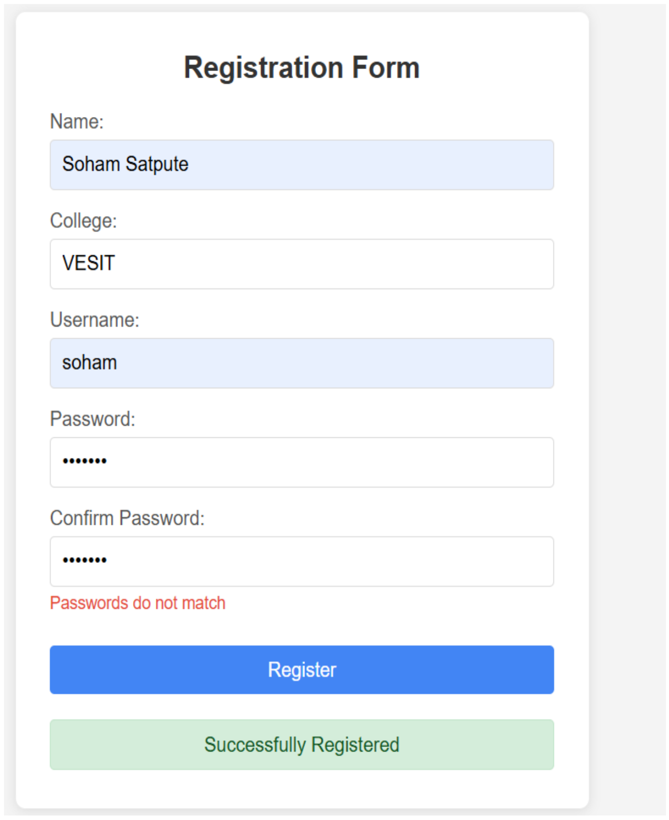
Password:

•••••••

Confirm Password:

•••••••

2) Passwords do not match


Registration Form

Name:

Soham Satpute

College:

VESIT

Username:

soham

Password:

•••••••

Confirm Password:

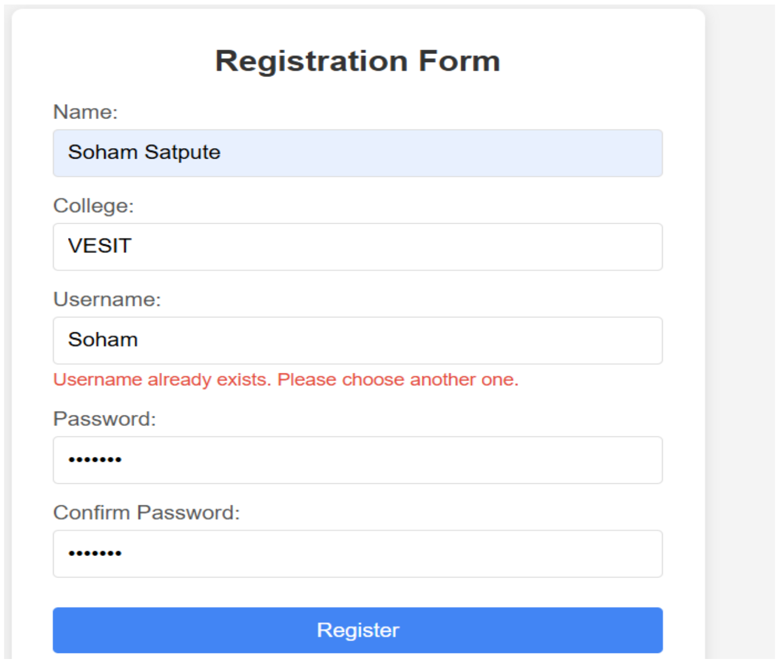
•••••••

Passwords do not match

Successfully Registered

Name : Soham Satpute **Class: D15A** **Roll No. 51**

3)Username already Exists:



Registration Form

Name: Soham Satpute

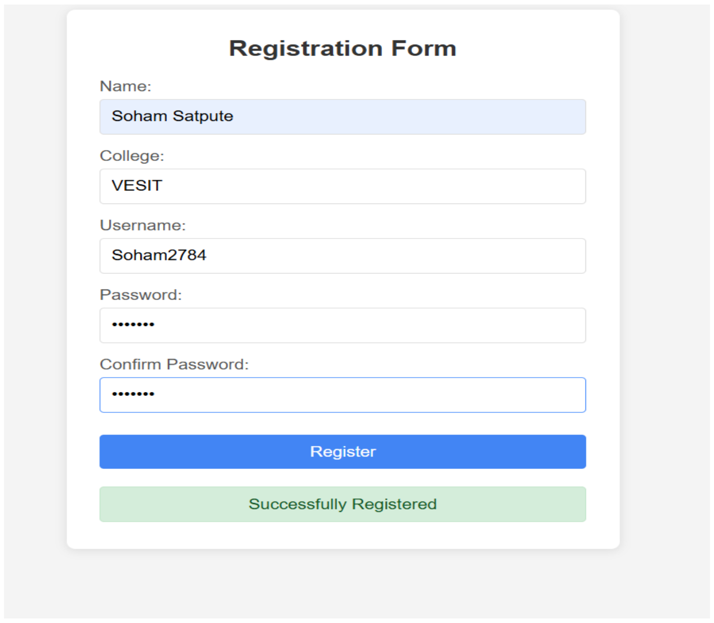
College: VESIT

Username: Soham

Username already exists. Please choose another one.

Password: •••••••

Confirm Password: •••••••



Registration Form

Name: Soham Satpute

College: VESIT

Username: Soham2784

Password: •••••••

Confirm Password: •••••••

Successfully Registered

Name : Soham Satpute **Class: D15A** **Roll No. 51**

Conclusion:

This experiment helped in understanding how to create a dynamic registration form using XMLHttpRequest for asynchronous operations. It demonstrated validation techniques such as password matching and checking for existing usernames. College name autosuggestion improved user experience through real-time data loading.