Name of Student	Aryan Dangat
Class Roll No	D15A_12
D.O.P.	
D.O.S.	
Sign and Grade	

Aim: To study AJAX

Theory:-

1. How do Synchronous and Asynchronous Requests differ?

Synchronous and Asynchronous requests are two different methods used in web development to communicate with a server, especially when using technologies like **AJAX (Asynchronous JavaScript and XML)**.

Synchronous Requests:

- In a synchronous request, the **browser waits** for the server to respond before continuing to execute the rest of the JavaScript code.

- The entire user interface (UI) **freezes or becomes unresponsive** until the request is complete.

- This approach can lead to a **poor user experience**, especially if the server takes a long time to respond.

   Example:
    If a synchronous request is made to load data from a server, the user cannot interact with the webpage (like clicking buttons or typing) until the response is received.

Asynchronous Requests:

- In an asynchronous request, the browser **does not wait** for the response.

- The remaining JavaScript code continues to execute, and the browser stays responsive.

- Once the server responds, a **callback function** (or event listener) is triggered to handle the response.

   Example:
   Fetching live search results without reloading the page. The user continues typing while results update in real time.

Key Differences:

Feature	Synchronous	Asynchronous
Browser Behavior	Waits for response	Does not wait
User Experience	Freezes UI	UI remains responsive
Implementation	Simpler, but less efficient	Requires callbacks or promises Common and preferred
Usage	Rare in modern web apps	

2. Describe various properties and methods used in XMLHttpRequest Object

The `XMLHttpRequest` object is a built-in JavaScript object used to interact with servers and fetch data without reloading the webpage. It is a core part of AJAX-based applications.
 Commonly Used Properties:

Property	Description
readyStat e	Returns the current state of the request (0 to 4).
status	Returns the HTTP status code (e.g., 200 for OK, 404 for Not Found).
statusTex t	Returns the textual status (e.g.,
responseT ext	"OK", "Not Found"). Returns the
responseX	

response data as a string.

_{ML}Returns the response data as an XML document (if the response is XML).

readyState Values:

Value State Description 0 UNSENT Request not initialized 1

OPENED `open()` has been called 2 HEADERS_RECEIVED

`send()` has been called 3 LOADING Receiving the response

4 DONE Request finished and response is ready

Commonly Used Methods:

Method	Description
`open(method, url, async)`	Initializes a request. `method` is "GET" or "POST", `async` is a boolean.
`send()`	Sends the request to the server.
`setRequestHeader(head er, value)`	Sets a request header (e.g., Content-Type). Used after `open()`, before `send()`.
`abort()`	Cancels an ongoing request.

Example:

Javascript

```javascript
let xhr = new XMLHttpRequest();
xhr.open("GET", "data.json", true); // Asynchronous GET request
xhr.onreadystatechange = function() {
  if (xhr.readyState === 4 && xhr.status === 200) {
  console.log(xhr.responseText); // Handle response }
};
```

```
xhr.send();
```

Problem Statement:

Create a registration page having fields like Name, College, Username and Password (read password twice).
Validate the form by checking for
 1. Usernameis not same as existing entries
 2. Name field is not empty
     3. Retyped password is matching with the earlier one. Prompt a
message is And also auto suggest college names.
Show the message "Successfully Registered" on the same page below the
submit button, on Successfully registration. Let all the updations on the page be
Asynchronously loaded. Implement the same using XMLHttpRequest Object.

CODE:-

script .js

```
const colleges = [

   "Vivekanand Education Society's Institute of Technology (VESIT)",

   "Vivekanand Education Society's Institute of Management Studies and Research

   (VESIM)", "Vivekanand Education Society's College of Pharmacy (VESCOP)",

   "Vivekanand Education Society's College of Arts, Science and Commerce

   (VESCOASC)", "Vivekanand Education Society's College of Law (VESCOL)",]


const existingUsernames = ['user1', 'user2', 'admin'];

document.addEventListener('DOMContentLoaded', function() {

   const form =

   document.getElementById('registrationForm');

   const collegeInput =

   document.getElementById('college');

   const suggestionsDiv =

   document.getElementById('collegeSuggestions'); const

   messageDiv =

   document.getElementById('registrationMessage');

      collegeInput.addEventListener('input'
```

```javascript
, function() { const input =
this.value.toLowerCase();

if (input.length < 2) {

  suggestionsDiv.style.dis

  play = 'none'; return;

}


const filteredColleges = colleges.filter(college

    =>

    college.toLowerCase().includes(input)

) if (filteredColleges.length > 0) {

  suggestionsDiv.innerHTML =

  filteredColleges


    .map(college => `<div>${college}</div>`)

    .join('');

suggestionsDiv.style.displa

y = 'block';

} else {


  suggestionsDiv.style.display =

'none';

suggestionsDiv.addEventListener('click',

function(e) {


  if (e.target.tagName === 'DIV') {
```

```javascript
        collegeInput.value =
        e.target.textContent;

        suggestionsDiv.style.display =

        'none';


    }



    }); document.addEventListener('click',

        function(e) { if (e.target !==

        collegeInput) {

        suggestionsDiv.style.display = 'none';

    }
});


    form.addEventListener('submit

        ', function(e) {

        e.preventDefault();

        messageDiv.className =

        'message';

        messageDiv.textContent =

        '';

    const name = document.getElementById('name').value.trim();

    const username =

    document.getElementById('username').value.trim();

    const college =

    document.getElementById('college').value.trim(); const

    password =
```

```
    document.getElementById('password').value;
    const confirmPassword = document.getElementById('confirmPassword').value;

    Let

isValid =

true; if

(name ===

") {

        document.getElementById('nameError').textContent = 'Name is

        required'; isValid = false;

    } else {

        document.getElementById('nameError').textContent = '';

    }

    if (college === ") {

        document.getElementById('collegeError').textContent = 'College

        is required'; isValid = false;

    } else {

        document.getElementById('collegeError').textContent = '';

    }

        if (password !== confirmPassword) {

            document.getElementById('confirmPasswordError').textContent =

            'Passwords do not match'; isValid = false;

    } else {

        document.getElementById('confirmPasswordError').textContent = '';

    }

        if

            (isV
```

```javascript
alid)
{
const data = {
name
usename
e,
college,
passwor
d
};

// Create XMLHttpRequest

const xhr = new XMLHttpRequest();
xhr.open('POST',
'http://localhost:3000/register', true);

xhr.setRequestHeader('Content-Type', 'application/json');
xhr.onload =
function() {
try {
```

```javascript
        const response =

        JSON.parse(xhr.responseText); if

        (xhr.status === 200 &&

        response.success) {

            messageDiv.className =

            'message success';

            messageDiv.textContent =

            response.message; form.reset();

        } else {


            messageDiv.className = 'message error';


            messageDiv.textContent = response.message || 'Registration failed. Please try again.';


        }


    } catch (error) {


        messageDiv.className = 'message error';


        messageDiv.textContent = 'Error processing response. Please try

    again.'; }

};
```

```javascript
        xhr.onerror = function() {

          messageDiv.className =

          'message error';

          messageDiv.textContent = 'An error occurred. Please try again.';

        };

        xhr.send(JSON.stringify(data));

      }

    }
  )
  ;

});
```
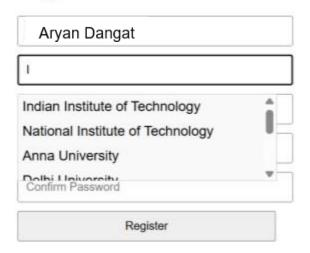
OutPut:

OUTPUT:

← → C ⓘ http://**127.0.0.1**:5500/index.html

Registration Form

Name

College

Username

Password

Confirm Password

Register

Registration Form

Aryan Dangat

I

Indian Institute of Technology
National Institute of Technology
Anna University
Delhi University
Confirm Password

Register

Aryan Dangat

Registration Form

Aryan Dangat

Indian Institute of Technology

user123

••••

••••

Register

Username already exists.

Registration Form

Aryan Dangat

Indian Institute of Technology

pepper

••••

••••

Register

Passwords do not match.

Registration Form

Aryan Dangat

Indian Institute of Technology

pepper

••••

••••

Register

Successfully Registered