

## Build a Dynamic Web Page Using JavaScript and AJAX to Fetch Data

### Step 1: Set Up Your Development Environment

- **Install a Code Editor:** Download and install a code editor like Visual Studio Code, Sublime Text, or Atom.
- **Create a Project Folder:** Create a new folder on your computer to store your project files.

### Step 2: Create the Basic HTML Structure

- **Create an HTML File:** In your project folder, create a file named `index.html`.
- **Add Basic HTML Structure:** Open `index.html` in your code editor and add the basic HTML structure:

**CODE:-**

```
<!DOCTYPE html>

<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Dynamic Web Page</title>
    <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">
    <link href="styles.css" rel="stylesheet">
  </head>
  <body>
    <div class="container">
      <h1 class="my-4">Dynamic Data Fetching with AJAX</h1>
      <button id="fetchDataBtn" class="btn btn-primary mb-4">Fetch Data</button>
      <div id="dataContainer"></div>
    </div>
    <script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
    <script src="scripts.js"></script>
  </body>
</html>
```

### Step 3: Add Custom CSS

- **Create a CSS File:** In your project folder, create a file named `styles.css`.
- **Add Custom Styles:** Open `styles.css` in your code editor and add custom styles:

#### CODE:-

```
body {  
    font-family: Arial, sans-serif;  
    background-color: #f8f9fa;  
    padding-top: 20px;  
}
```

```
#dataContainer {  
    margin-top: 20px;  
}
```

### Step 4: Create a JavaScript File

- **Create a JavaScript File:** In your project folder, create a file named `scripts.js`.
- **Add AJAX Functionality:** Open `scripts.js` in your code editor and add the following code to fetch data dynamically using AJAX:

#### CODE:-

```
document.getElementById('fetchDataBtn').addEventListener('click', fetchData);
```

```
function fetchData() {  
  
    const xhr = new XMLHttpRequest();  
  
    xhr.open('GET', 'https://jsonplaceholder.typicode.com/posts', true);  
  
  
    xhr.onload = function() {  
  
        if (this.status === 200) {  
  
            const posts = JSON.parse(this.responseText);  
        }  
    };  
}
```

```
let output = "";

posts.forEach(function(post) {

    output += `

<div class="card mb-3">

<div class="card-body">

<h5 class="card-title">${post.title}</h5>

<p class="card-text">${post.body}</p>

</div>

</div>

`;

});

document.getElementById('dataContainer').innerHTML = output;

}

};

xhr.onerror = function() {

    console.error('Request error...');

};

xhr.send();

}
```

## Step 5: Test and Refine

- **Open the HTML File in a Browser:** Open `index.html` in a web browser to view your dynamic web page.
- **Fetch Data:** Click the "Fetch Data" button to fetch data from the external API and display it dynamically on the web page.
- **Refine the Design:** Make any necessary adjustments to the HTML, CSS, and JavaScript to improve the functionality and appearance of your dynamic web page.

## Step 6: Add Error Handling (Optional)

- **Enhance Error Handling:** Update the JavaScript code to handle potential errors gracefully

CODE:-

```
function fetchData() {  
  
    const xhr = new XMLHttpRequest();  
  
    xhr.open('GET', 'https://jsonplaceholder.typicode.com/posts', true);  
  
  
    xhr.onload = function() {  
  
        if (this.status === 200) {  
  
            const posts = JSON.parse(this.responseText);  
  
            let output = '';  
  
  
            posts.forEach(function(post) {  
  
                output += `  


##### ${post.title}

  


${post.body}

`  
            });  
  
            document.getElementById('posts').innerHTML = output;  
        }  
    };  
};  
  
document.getElementById('fetchData').addEventListener('click', fetchData);
```

```
`;

});

document.getElementById('dataContainer').innerHTML = output;

} else {

    document.getElementById('dataContainer').innerHTML = '<p class="text-danger">Error
fetching data</p>';

}

};

xhr.onerror = function() {

    document.getElementById('dataContainer').innerHTML = '<p class="text-danger">Request
error...</p>';

};

xhr.send();

}
```

#### Step 7: Publish Your Web Page

- **Choose a Hosting Service:** Select a web hosting service like GitHub Pages, Netlify, or Vercel to publish your web page.
- **Deploy Your Site:** Follow the hosting service's instructions to deploy your dynamic web page, making it accessible online.

By following these steps, you'll create a dynamic web page using JavaScript and AJAX to fetch and display data from an external source.