

INSERT DATA

let's break down the process of inserting data using AJAX into a server. I'll provide a step-by-step tutorial in text format first and then follow it up with code examples.

1. Create HTML Form:

Start by creating an HTML form that includes the input fields for the data you want to insert. For example:

```
``html

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>AJAX Insert Data</title>

</head>

<body>

    <form id="insertForm">

        <label for="name">Name:</label>

        <input type="text" id="name" name="name">
```

```
<label for="email">Email:</label>

<input type="email" id="email" name="email">


<button type="button" onclick="insertData()">Insert
Data</button>

</form>


<div id="result"></div>


<script src="your_ajax_library.js"></script>

<script src="your_script.js"></script>

</body>

</html>

...

```

2. Write AJAX Function:

Create a JavaScript file (e.g., `your_script.js`) and write a function to handle the AJAX request:

```
```javascript

function insertData() {

```

```
var name = document.getElementById('name').value;

var email = document.getElementById('email').value;

var xhr = new XMLHttpRequest();

xhr.open('POST', 'insert_data.php', true);

xhr.setRequestHeader('Content-Type',
'application/x-www-form-urlencoded');

xhr.onreadystatechange = function () {

 if (xhr.readyState == 4 && xhr.status == 200) {

 document.getElementById('result').innerHTML =
xhr.responseText;

 }

};

var data = 'name=' + encodeURIComponent(name) + '&email='
+ encodeURIComponent(email);

xhr.send(data);

}

...

```

Make sure to replace ``your\_ajax\_library.js`` with the path to your AJAX library (e.g., jQuery) if you are using one.

### 3. Create Server-Side Script:

On the server side, create a script (e.g., ``insert\_data.php``) to handle the data insertion:

```
```.php
<?php
$name = $_POST['name'];
$email = $_POST['email'];

// Perform data insertion logic here (e.g., insert into a database)

// Provide a response (e.g., success or error message)
echo 'Data inserted successfully!';

?>
````
```

Adjust the server-side script based on your server environment and data storage mechanism.

## 4. Handle Response:

In the AJAX function, the `onreadystatechange` event is triggered when the server responds. Update the HTML element with the result:

```
````javascript
xhr.onreadystatechange = function () {
    if (xhr.readyState == 4 && xhr.status == 200) {
        document.getElementById('result').innerHTML =
xhr.responseText;
    }
};
````
```

The server's response will be displayed in the element with the ID 'result'.

## 5. Test:

Open your HTML file in a browser and test the data insertion by entering values in the form and clicking the "Insert Data" button.

That's a basic tutorial on how to insert data using AJAX. The server-side script should be adapted based on your server environment and how you want to handle the data insertion.