AJAX

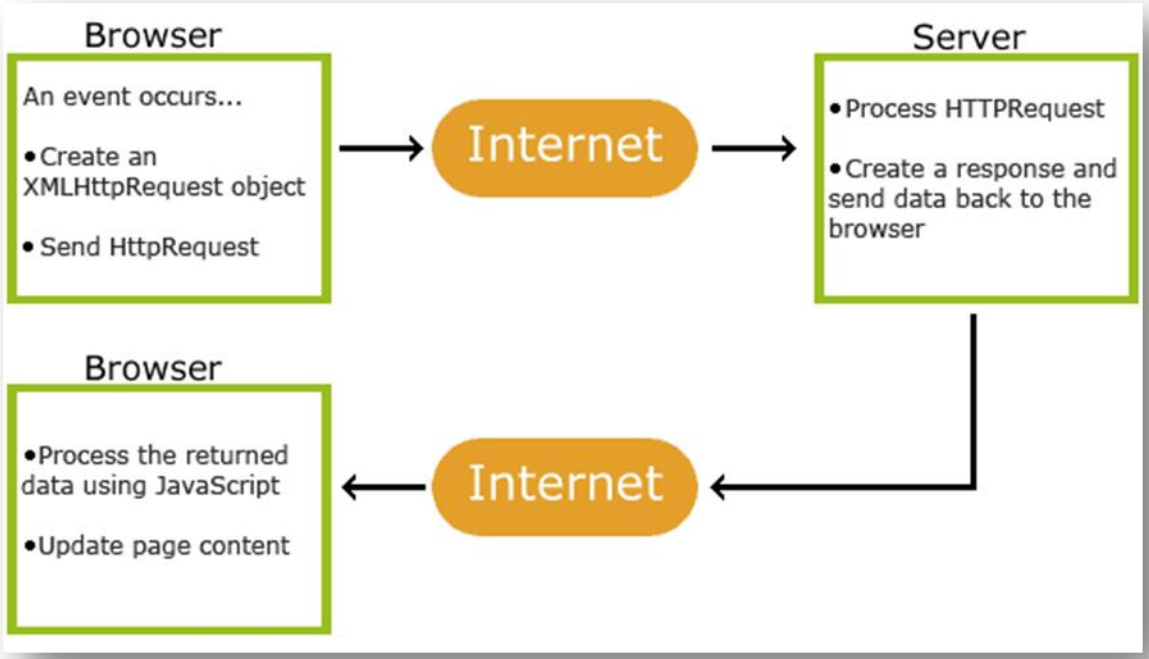
AJAX

- AJAX stands for Asynchronous JavaScript And XML
- AJAX is about updating parts of a web page, without reloading the whole page.
- Ajax or Asynchronous JavaScript and XML enables the programmer to execute a server-side script without refreshing the page.
- AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the whole page.
- Examples of applications using AJAX: Google Maps, Gmail, YouTube, and Facebook.

How AJAX Works

- The typical method for using Ajax is the following:
  1. A JavaScript creates an XMLHttpRequest object, initializes it with relevant information as necessary, and sends it to the server.
  2. The server responds by sending the contents of a file or the output of a server side program .
  3. When the response arrives from the server, a JavaScript function is triggered to act on the data supplied by the server.
  4. This JavaScript response function typically refreshes the display using the DOM, avoiding the requirement to reload or refresh the entire page.

How AJAX Works



The Back end

- The part of the Ajax application that resides on the web server is referred to as the "back end".
- This back end could be simply a file that the server passes back to the client.
- The back end could be a program, written in PHP, Perl, Ruby, Python, C, or some other language that performs an operation and sends results back to the client browser.

The XMLHttpRequest Object

- The XMLHttpRequest object is the backbone of every Ajax method. Each application requires the creation of one object.
- The XMLHttpRequest object is used to exchange data with a server behind the scenes.
- An XMLHttpRequest object can send information using the GET and POST methods to the server in the same way that an HTML form sends information.

Send a Request To a Server

- To send a request to a server, we use the open() and send() methods of the XMLHttpRequest object.
- The following table shows the methods of XMLHttpRequest object.

Method	Description
open(*method, url, async*)	Specifies the type of request. ***method***: the type of request: GET or POST ***url***: the server (file) location ***async***: true (asynchronous) or false (synchronous)
send()	Sends the request to the server (used for GET)
send(*string*)	Sends the request to the server (used for POST)

Server Response

- To get the response from a server, use the responseText or responseXML property of the XMLHttpRequest object.

Property	Description
responseText	get the response data as a string
responseXML	get the response data as XML data

The onreadystatechange Event

- To perform some actions based on the server response the onreadystatechange event is used.
- The onreadystatechange event is triggered every time the readyState changes.
- The readyState property holds the status of the XMLHttpRequest.
- Three important properties of the XMLHttpRequest object:

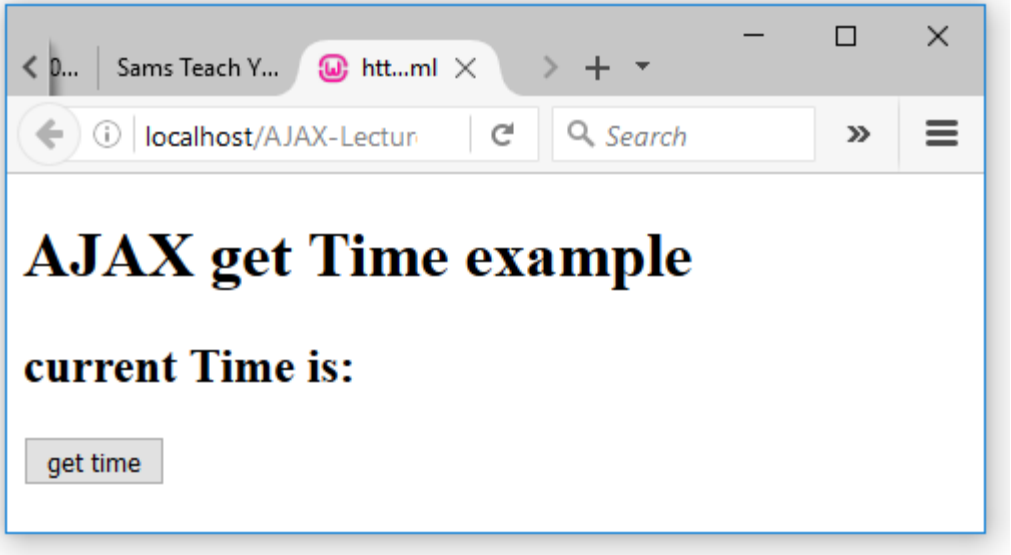
Property	Description
onreadystatechange	Stores a function (or the name of a function) to be called automatically each time the readyState property changes
readyState	Holds the status of the XMLHttpRequest. Changes from 0 to 4: 0: request not initialized 1: server connection established 2: request received 3: processing request 4: request finished and response is ready
status	200: "OK" 404: Page not found

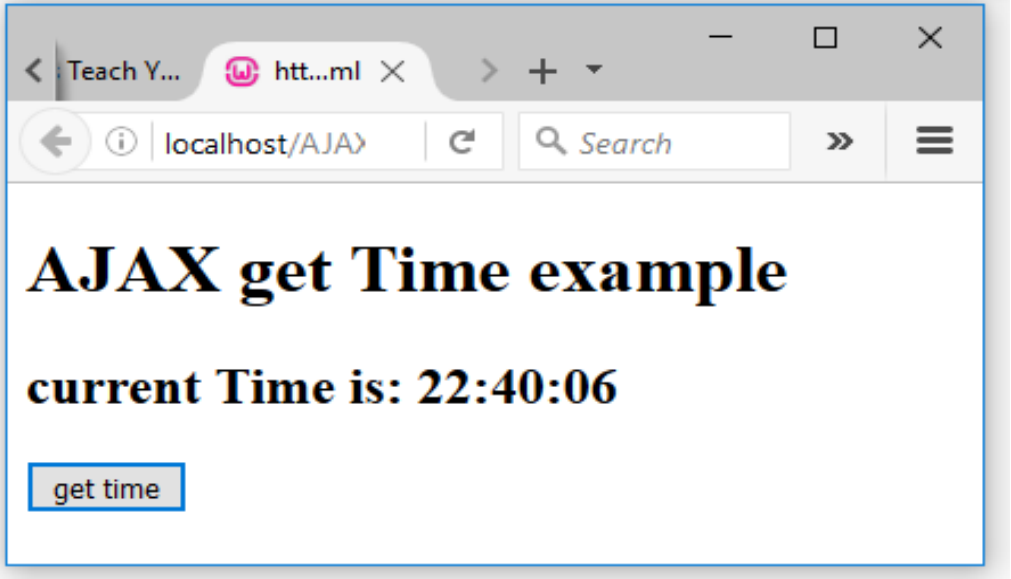
Example :

```
<html>
    <body>
        <h1>AJAX get Time example</h1>
        <div id="div1"><h2>current Time is:</h2></div>

        <button type="button" onclick="getDate()">get time</button>

        <script>
            function getDate() {
                var xhttp = new XMLHttpRequest();
                xhttp.onreadystatechange = function () {
                    if (xhttp.readyState == 4 && xhttp.status == 200) {
                        document.getElementById("div1").innerHTML =  xhttp.responseText;
                    }
                };
                xhttp.open("POST", "getTime.php", true);
                xhttp.send();
            }
        </script>

    </body>
</html>
```

getTime.php

```php
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
    </head>
    <body>
        <?php
        echo "<h2> current Time is: ".date('H:i:s')."</h2>";
        ?>
    </body>
</html>
```


Sams Teach Y...
htt...ml
localhost/AJAX-Lectur
Search
AJAX get Time example
current Time is:
get time

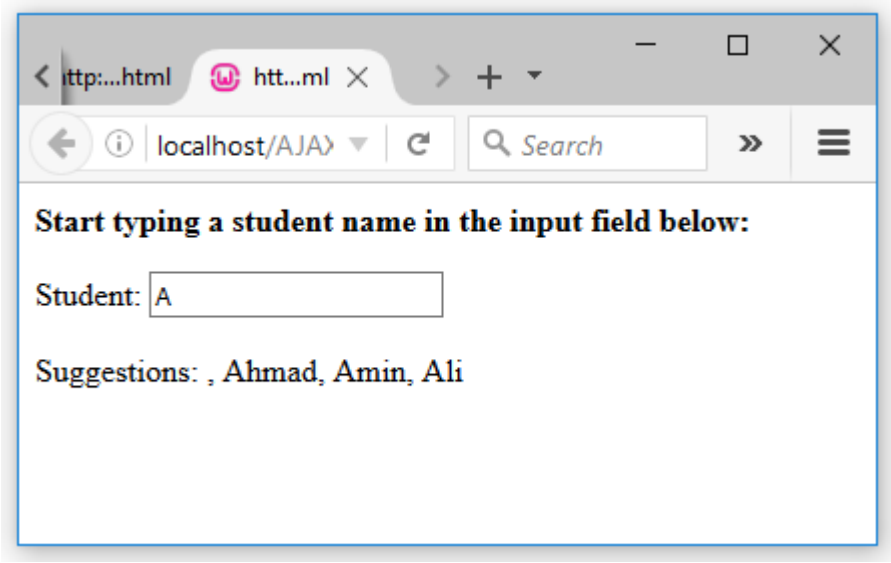

Teach Y...
htt...ml
localhost/AJA
Search
AJAX get Time example
current Time is: 22:40:06
get time

Example:

```
<html>
    <head>
        <script>
            function showHint(str) {
                if (str.length == 0) {
                    document.getElementById("txtHint").innerHTML = "";
                    return;
                } else {
                    var xmlhttp = new XMLHttpRequest();
                    xmlhttp.onreadystatechange = function () {
                        if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
                            document.getElementById("txtHint").innerHTML = xmlhttp.responseText;
                        }
                    };
                    xmlhttp.open("GET", "getStudents.php?q=" + str, true);
                    xmlhttp.send();
                }
            }
        </script>
    </head>
    <body>

        <p><b>Start typing a student name in the input field below:</b></p>
        <form>
            Student: <input type="text" onkeyup="showHint(this.value)">
        </form>
        <p>Suggestions: <span id="txtHint"></span></p>
    </body>
</html>
```

getStudents.php

```
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
    </head>
    <body>
        <?php
        $students = array("Ahmad","Khalid","Samira","Jomaa","salem",
                          "Amin","Tamer","Zyad","Salma","Khaeri","Ali");
        $q = $_GET["q"];
        $hint = "";
        if ($q !== "") {
            $len = strlen($q);
            foreach ($students as $name) {
                if (stristr($q, substr($name, 0, $len))) {
                    $hint .= ", $name";
                }
            }
        }
        echo $hint === "" ? "no suggestion" : $hint;
        ?>
    </body>
</html>
```

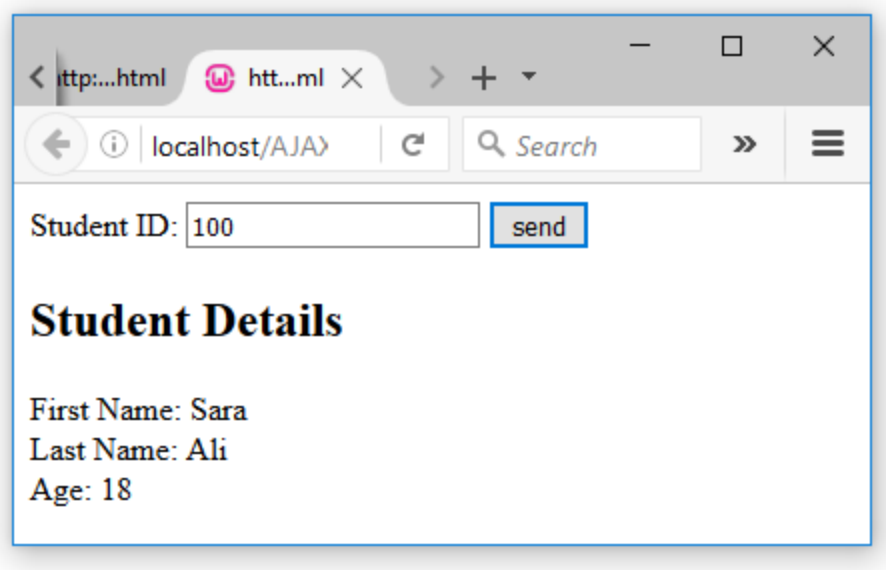

ttp:...html
htt...ml
localhost/AJAX
Search
Start typing a student name in the input field below:
Student: A
Suggestions: , Ahmad, Amin, Ali

AJAX DataBase Example:

```
<html>
    <head>
        <script>
            function showDetails(str) {

                    var xmlhttp = new XMLHttpRequest();
                    xmlhttp.onreadystatechange = function () {
                        if (xmlhttp.readyState == 4 && xmlhttp.status == 200) {
                            document.getElementById("student").innerHTML = xmlhttp.responseText;
                        }
                    };
                    xmlhttp.open("GET", "student-details.php?q=" + str, true);
                    xmlhttp.send();
                }

        </script>
    </head>
    <body>

        <form>
            Student ID: <input id="bt1" type="text" >
            <input type="button" value="send" onclick="showDetails(bt1.value)"/>
        </form>
        <p id="student"></p>
    </body>
</html>
```

Student-details.php

```php
<?php

$database_connection = mysqli_connect("localhost", "root", "", "students");
if ($database_connection === false) {
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
$q = $_GET["q"];
$sql = "SELECT * FROM registration where id = $q ";
if ($result = mysqli_query($database_connection, $sql)) {
    if (mysqli_num_rows($result) == 1) {
        $row = mysqli_fetch_assoc($result);

        echo " <h2> Student Details</h2>";
        echo "<Student ID: " . $row['id'] . "<br/>";
        echo "First Name: " . $row['first'] . "<br/>";
        echo "Last Name: " . $row['last'] . "<br/>";
        echo "Age: " . $row['age'] . "<br/>";
    } else {
        echo "<h3> No data found </h3>";
    }
} else {
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($database_connection);
}
mysqli_close($database_connection);

?>
```


ttp:...html
htt...ml
localhost/AJAX
Search
Student ID: 100
send
Student Details
First Name: Sara
Last Name: Ali
Age: 18

Thanks!