http://www.ptutorial.com/php-ajax-example-with-source-code

https://www.tutorialspoint.com/php/php_ajax_search.htm

AJAX- Asynchronous Java Script and XML

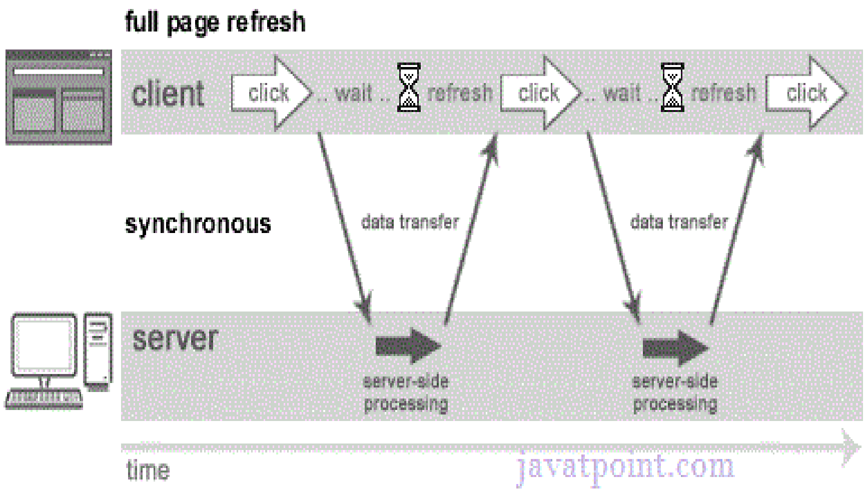
- AJAX is an acronym for **Asynchronous JavaScript and XML**. It is a group of inter-related technologies like JavaScript, DOM, XML, HTML, CSS etc.
- AJAX allows you to send and receive data asynchronously without reloading the web page. So it is fast.
- AJAX allows you to send only important information to the server not the entire page. So only valuable data from the client side is routed to the server side. It makes your application interactive and faster.

Where it is used?

- There are too many web applications running on the web that are using ajax technology like **gmail**, **facebook**,**twitter**, **google map**, **youtube** etc.

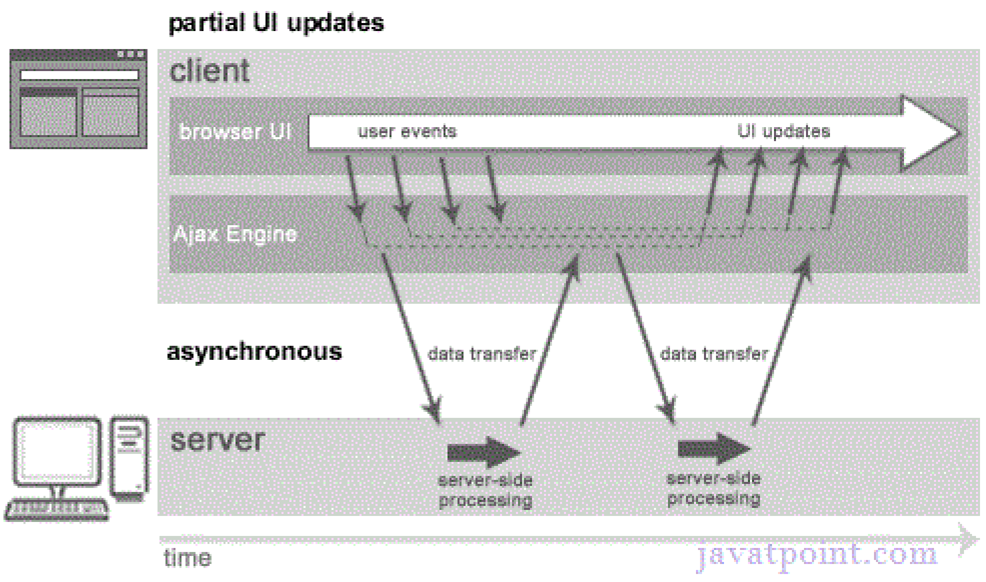
Synchronous and asynchronous

- Synchronous (Classic Web-Application Model)
- A synchronous request blocks the client until operation completes i.e. browser is not unresponsive. In such case, javascript engine of the browser is blocked.



- As you can see in the above image, full page is refreshed at request time and user is blocked until request completes.

Asynchronous

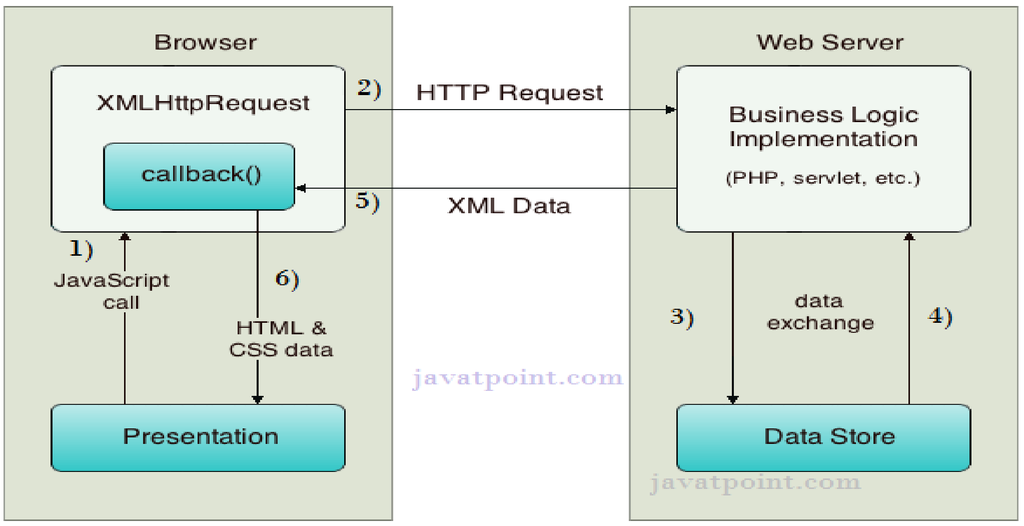
- An asynchronous request doesn’t block the client i.e. browser is responsive. At that time, user can perform another operations also. In such case, javascript engine of the browser is not blocked.
- 

- As you can see in the above image, full page is not refreshed at request time and user gets response from the ajax engine.

- ajax is not a technology but group of inter-related technologies. AJAX technologies includes:
- **HTML/XHTML and CSS** - These technologies are used for displaying content and style. It is mainly used for presentation.
- **DOM**-It is used for dynamic display and interaction with data.
- **XML or JSON**-For carrying data to and from server. JSON (Javascript Object Notation) is like XML but short and faster than XML.
- **XMLHttpRequest**-For asynchronous communication between client and server
- **JavaScript**-It is used to bring above technologies together.

How AJAX works?

- AJAX communicates with the server using XMLHttpRequest object.



- User sends a request from the UI and a javascript call goes to XMLHttpRequest object.
- HTTP Request is sent to the server by XMLHttpRequest object.
- Server interacts with the database using JSP, PHP, Servlet, ASP.net etc.
- Data is retrieved.
- Server sends XML data or JSON data to the XMLHttpRequest callback function.
- HTML and CSS data is displayed on the browser.

XMLHttpRequest object

An object of XMLHttpRequest is used for asynchronous communication between client and server.

It performs following operations:

- Sends data from the client in the background
- Receives the data from the server
- Updates the webpage without reloading it.

Methods of XMLHttpRequest Object

Method	**Description**
void open(method, URL)	opens the request specifying get or post method and url.
void open(method, URL, async)	same as above but specifies asynchronous or not.
void open(method, URL, async, username, password)	same as above but specifies username and password.
void send()	sends get request.
void send(string)	send post request.
setRequestHeader(header,value)	it adds request headers.

Properties of XMLHttpRequest object

Property	**Description**
onReadyStateChange	It is called whenever readystate attribute changes. It must not be used with synchronous requests.
readyState	represents the state of the request. It ranges from 0 to 4. **0** UNOPENED open() is not called. **1** OPENED open is called but send() is not called. **2** HEADERS_RECEIVED send() is called, and headers and status are available. **3** LOADING Downloading data; responseText holds the data. **4** DONE The operation is completed fully.
reponseText	returns response as text.
responseXML	returns response as XML

Creating the XMLHttpRequest Object

- In JavaScript create an XMLHttpRequest object and store it in a variable named XMLHttpRequestObject. Here it is set to be false so that if we are unsuccessful in creating an XMLHttpRequest object, this variable will be left holding FALSE.

```
<script language = "javascript">
    var XMLHttpRequestObject = false;
      :
      :
</script>
```

Next depending on what browser we are dealing with, the object is created

```
<script language = "javascript">
    var XMLHttpRequestObject = false;


    if (window.XMLHttpRequest) {
      XMLHttpRequestObject = new XMLHttpRequest();
    }
  </script>
```

Other than internet explorer can be checked with this. if others then the object can be created using *new.*

For IE XMLHttpRequest object is created as an ActiveX object

```
<script language = "javascript">
    var XMLHttpRequestObject = false;
    if (window.XMLHttpRequest) {
      XMLHttpRequestObject = new XMLHttpRequest();
    }
  else if (window.ActiveXObject) {
      XMLHttpRequestObject = new  ActiveXObject("Microsoft.XMLHTTP");
    }
  </script>
```

When the user clicks the button with the caption “submit” from the html page, after entering input in the two text boxes named ‘a’ and ‘b’ respectively

```
<form name="f1">
<input type="text" id="a" name="a">
<input type="text" id="b" name="b">
<button type="button" onclick="loaddoc()"> Submit </button>
</form>
<h1> Addition of two numbers</h1>
<div id="c"></div>
</body>
```

This button is connected to JavaScript function, loaddoc(), and passes two arguments. These values are sent to server through url.

data.txt – the name of the file to be fetched from server

c – which is the name of <div> element to display the downloaded text in.

In the below example two files are used.

1) ajaxeg1.html : in this program a form “f1” is created. This has two input text box control and one button “submit”. Two inputs are received from the user and the data is passed as GET method to PHP program “ add_two-numbers.php” .
2) the object XMLHttpRequest() has to be created through which only the AJAX technology works.
3) This is different for various browser.
4) For example ; for Netscape Navigator, Apple Safari, Firefox will use the object with the syntax

   Window.XMLHttpRequest

   Whereas for Internet Explorer it uses the ActiveXObject as the following syntax;

   window.ActiveXObject

5) Both these checking has to be carried over as the developer may not know, which browser the user will use.
6) Once they are checked for the browser the XMLHttpRequest object is created accordingly.
7) Once the object is created , it has to be opened, which configures it for use with the server.
   The general syntax:
   Open(“method”,”url” [,asyncFlag [, “username” [, “password”]]])
   Where
   “method “ : This is the HTTP method used to open the connection , such as GET, POST, PUR, HEAD
   “url” : this is the requested url
   “asyncFlag : A Boolean value indicating whether the call is asynchronous. The default is TRUE

"username" : The user name
"password" : The password
This can be done as
XMLHttpRequestObject.open("GET",datasource)
In the given program it is done as

u.open("GET",url,true);

where u is the XMLHttpRequest object.

8) In the below code the function loaddoc() written in javascript is called from html on clicking the submit button after the user entered two input to be added in the textbox controls.

Example 1: ajaxeg1.html

```
<html>
<head>

<script>
function loaddoc()
{
var a=parseInt(document.f1.a.value);
var b=parseInt(document.f1.b.value);
var url="add_two_numbers.php?a="+a+"&b="+b+"";
var u;

if(window.XMLHttpRequest) //checking for firefox, safari etc
          {
             u=new XMLHttpRequest();  //creation of new object
          }
           else
           {
              u=new ActiveXObject("Microsoft.XMLHTTP") // creation of object for
internetexplorer

           }
u.onreadystatechange=function ()  // callback function
   {
       if(u.readyState==4 && u.status==200)  //checking the status of download and HTTP
status code
           {
             document.getElementById("c").innerHTML=u.responseText; //downloaded
information will
                                                                    // be stored in
responseText
```

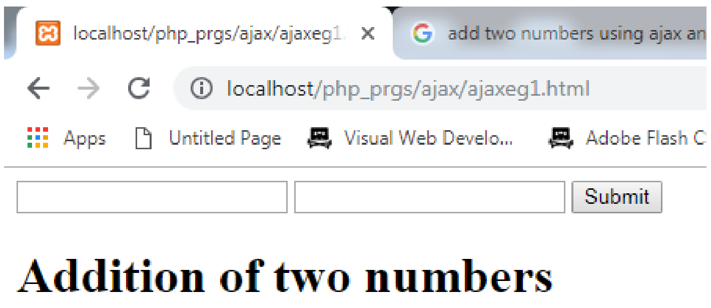
```
            }
     }
u.open("GET",url,true);  //this will open and connect to server
u.send();
} //end of function loaddoc()
</script>
</head>
<body>
<form name="f1">
<input type="text" id="a" name="a">
<input type="text" id="b" name="b">
<button type="button" onclick="loaddoc()"> Submit </button>
</form>
<h1> Addition of two numbers</h1>
<div id="c"></div>
</body>
</html>
```

9) Below is the php coding where it receives the value entered in the html controls through GET method which is send in the url.
10) The two values $num1 and $num2 are added and send to the browser.
11) This value is placed in the div element by using its getElementBYId.innerHTML and displayed on the browser.

Add_two_numbers.php

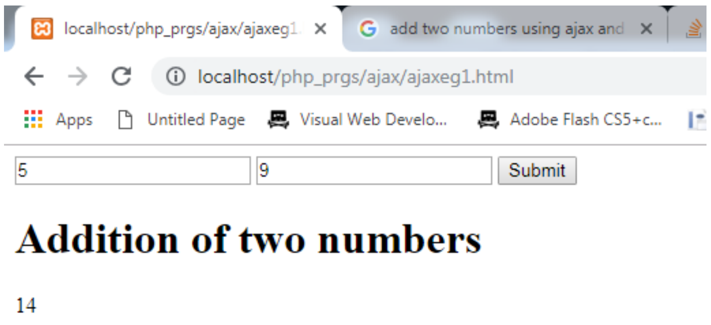
```
<?php
$num1=trim($_GET["a"]);
$num2=trim($_GET["b"]);
echo $num1+$num2."<br>";
?>
```

Output


localhost/php_prgs/ajax/ajaxeg1.html

Submit

Addition of two numbers


localhost/php_prgs/ajax/ajaxeg1.html

5 9 Submit

Addition of two numbers

14

In the below example two files are used.

1) Ajaxeg2.html : in this program a form “f1” is created. This has one button “submit” control. The values from input.php file are received and displayed in the div control .
2) the object XMLHttpRequest() has to be created through which only the AJAX technology works.
3) This is different for various browser.
4) For example ; for Netscape Navigator, Apple Safari, Firefox will use the object with the syntax

   Window.XMLHttpRequest

   Whereas for Internet Explorer it uses the ActiveXObject as the following syntax;

   window.ActiveXObject

5) Both these checking has to be carried over as the developer may not know, which browser the user will use.
6) Once they are checked for the browser the XMLHttpRequest object is created accordingly.
7) Once the object is created , it has to be opened, which configures it for use with the server.
   The general syntax:
   Open(“method”,”url” [,asyncFlag [, “username” [, “password”]]])
   Where
   “method “ : This is the HTTP method used to open the connection , such as GET, POST, PUR, HEAD
   “url” : this is the requested url
   “asyncFlag : A Boolean value indicating whether the call is asynchronous. The default is TRUE
   “username” : The user name
   “password” : The password
   This can be done as
   XMLHttpRequestObject.open(“GET”,datasource)
   In the given program it is done as

   u.open("GET",”input.php”,true);

   where u is the XMLHttpRequest object.

8) In the below code the function loaddoc() written in javascript is called from html on clicking the submit button.

Example 2:

```
<!DOCTYPE html>
<html>
   <head>
      <script>
      function loaddoc()
      {
           var u;
         if(window.XMLHttpRequest)
         {
            u=new XMLHttpRequest();
```

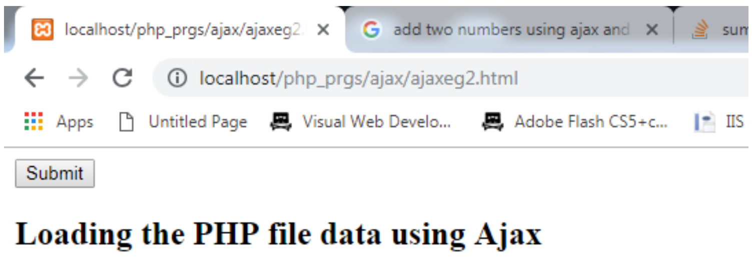
```
          }
          else
          {
             u=new ActiveXObject("Microsoft.XMLHTTP")

          }
            u.onreadystatechange=function ()
            {
               if(u.readyState==4 && u.status==200)
               {
                  document.getElementById("c").innerHTML=u.responseText;
               }

            }
            u.open("GET","input.php",true);
            u.send();
      }
      </script>
   </head>


<body>

<form name="f1">

<button type="button" onclick="loaddoc()">Submit</button>
      </form>
<H2>Loading the PHP file data using Ajax</H2>
      <div id="c"> </div>
   </body>
</html>
```

1) Below is the php coding where it has echo statements .
2) This value is placed in the div element by using its getElementBYId.innerHTML and displayed on the browser.

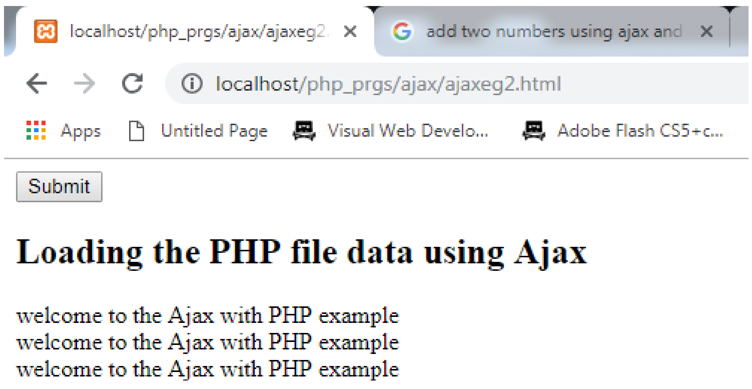
Input.php

```
<?php
echo "welcome to the Ajax with PHP example<br>";
echo "welcome to the Ajax with PHP example<br>";
echo "welcome to the Ajax with PHP example<br>";
?>
```

Output


localhost/php_prgs/ajax/ajaxeg2
add two numbers using ajax and
localhost/php_prgs/ajax/ajaxeg2.html
Apps
Untitled Page
Visual Web Develo...
Adobe Flash CS5+c...
IIS
Submit

Loading the PHP file data using Ajax


localhost/php_prgs/ajax/ajaxeg2
add two numbers using ajax and
localhost/php_prgs/ajax/ajaxeg2.html
Apps
Untitled Page
Visual Web Develo...
Adobe Flash CS5+c...
Submit

Loading the PHP file data using Ajax

```
welcome to the Ajax with PHP example
welcome to the Ajax with PHP example
welcome to the Ajax with PHP example
```

In the below example two files are used.

1) ajaxeg1.html : in this program a form “f1” is created. This has oneinput text box control and one button “submit”. The input is received from the user and the data is passed as GET method to PHP program “ mult_table.php” .
2) the object XMLHttpRequest() has to be created through which only the AJAX technology works.
3) This is different for various browser.

4) For example ; for Netscape Navigator, Apple Safari, Firefox will use the object with the syntax

   Window.XMLHttpRequest

   Whereas for Internet Explorer it uses the ActiveXObject as the following syntax;

   window.ActiveXObject

5) Both these checking has to be carried over as the developer may not know, which browser the user will use.
6) Once they are checked for the browser the XMLHttpRequest object is created accordingly.
7) Once the object is created , it has to be opened, which configures it for use with the server.
   The general syntax:
   Open("method","url" [,asyncFlag [, "username" [, "password"]]])
   Where
   "method " : This is the HTTP method used to open the connection , such as GET, POST, PUR, HEAD
   "url" : this is the requested url
   "asyncFlag : A Boolean value indicating whether the call is asynchronous. The default is TRUE
   "username" : The user name
   "password" : The password
   This can be done as
   XMLHttpRequestObject.open("GET",datasource)
   In the given program it is done as

   ```
   u.open("GET",url,true);
   ```

   where u is the XMLHttpRequest object.

8) In the below code the function loaddoc() written in javascript is called from html on clicking the submit button after the user entered an input in the text box for which the multiplication table.

Example 3 :ajaxeg3.html

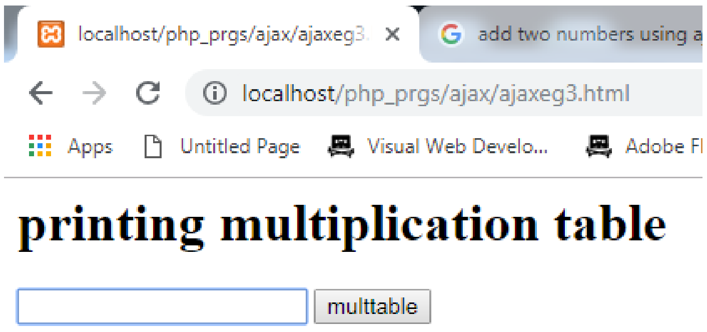
```
<html>
<head>
<h1> printing multiplication table</h1>
<script>

function multtable()
{
var u=document.f1.num.value;
var url="mult_table.php?num="+u+"";
var obj;

if(window.XMLHttpRequest)
   {
    obj=new XMLHttpRequest();
   }
else if(window.ActiveXObject)
   {
   obj=new ActiveXObject("Microsoft.XMLHTTP");
   }
obj.onreadystatechange=function()
   {
     if(obj.readyState==4 && obj.status==200)
      {
       document.getElementById("mult").innerHTML=obj.responseText;
      }
   }
obj.open("GET",url,true);
obj.send();
}
</script>
</head>
<body>
<form name="f1">
<input type="text" id="num" name="num">
<button type="button" onclick="multtable()">multtable</button>

</form>
<div id="mult"></div>
</body>
</html>
```

1) Below is the php coding where it receives the value entered in the html controls through GET method which is send in the url.
2) The multiplication table for the value received through GET method in the url is calculated and displayed by php which is downloaded in the responseText of ajax.
3) This value is placed in the div element by using its getElementBYId.innerHTML and displayed on the browser.

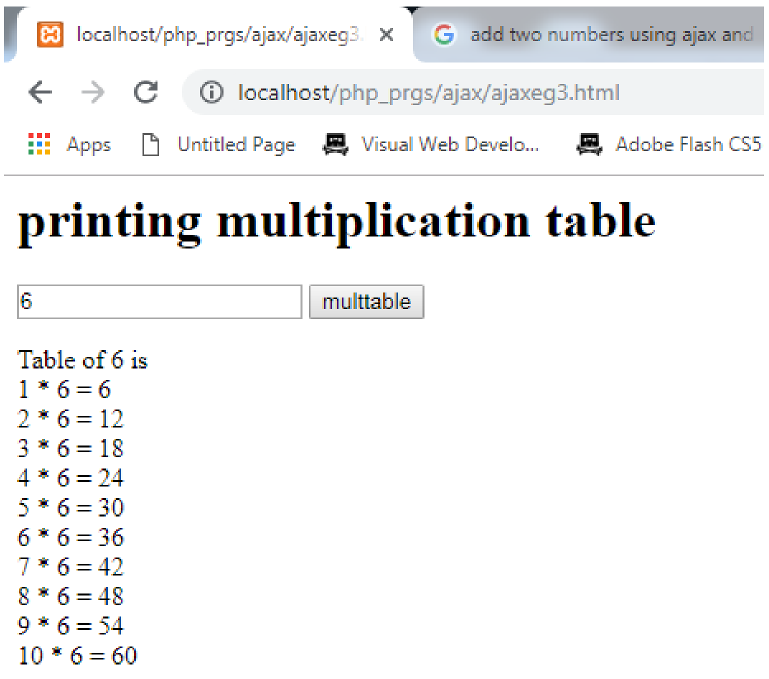
mult_table.php

```
<?php
// printing multiplication table
$num=trim($_GET["num"]);
echo "Table of ".$num." is <br>";
for($i=1;$i<=10;$i++)
{
$x=$num*$i;
echo $i ." * ".$num." = ".$x."<br>";

}
?>
```

Output



printing multiplication table



localhost/php_prgs/ajax/ajaxeg3.
add two numbers using ajax and
localhost/php_prgs/ajax/ajaxeg3.html
Apps
Untitled Page
Visual Web Develo...
Adobe Flash CS5
multtable

printing multiplication table

```
6
Table of 6 is
1 * 6 = 6
2 * 6 = 12
3 * 6 = 18
4 * 6 = 24
5 * 6 = 30
6 * 6 = 36
7 * 6 = 42
8 * 6 = 48
9 * 6 = 54
10 * 6 = 60
```

In the below example two files are used.

1) Ajaxeg4.html : in this program a form “f1” is created. This has one input text box control. The input is received from the user and the data is passed as GET method to PHP program “ php_connect.php” .
2) The object XMLHttpRequest() has to be created through which only the AJAX technology works.
3) This is different for various browser.
4) For example , for Netscape Navigator, Apple Safari, Firefox will use the object with the syntax

   Window.XMLHttpRequest

   Whereas for Internet Explorer it uses the ActiveXObject as the following syntax;

   window.ActiveXObject

5) Both these checking has to be carried over as the developer may not know, which browser the user will use.
6) Once they are checked for the browser the XMLHttpRequest object is created accordingly.
7) Once the object is created , it has to be opened, which configures it for use with the server.
   The general syntax:
   Open(“method”,”url” [,asyncFlag [, “username” [, “password”]]])
   Where

"method " : This is the HTTP method used to open the connection , such as GET, POST, PUR, HEAD
"url" : this is the requested url
"asyncFlag : A Boolean value indicating whether the call is asynchronous. The default is TRUE
"username" : The user name
"password" : The password
This can be done as
XMLHttpRequestObject.open("GET",datasource,true)
In the given program it is done as

```
xmlhttpobj.open("GET",url,true);
```

where xmlhttpobj is the XMLHttpRequest object.

8) In the below code the function loaddoc() written in javascript is called from html when the user enters the data in the text box, the value is passed as argument in the url with GET method.
9) This is extracted in PHP program php_connect.php which connects to the database and table respectively and extract the data from database server.
10) This information is sent back to client side browser which is caught in the responseText method of XMLHttpObject and displayed in the div element of HTML.

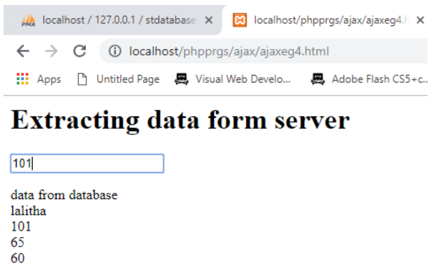
Example 4: ajaxeg4.html

```
<html>
<head>
<h1> Extracting data form server</h1>
<script>

function load_data()
{
var v1=document.f1.ct.value;
var url="php_connect.php?val="+v1;

if(window.XMLHttpRequest)
  {
   xmlhttpobj=new XMLHttpRequest();
  }
else if(window.ActiveXObject)
  {
  xmlhttpobj=new ActiveXObject("Microsoft.XMLHTTP");
  }
xmlhttpobj.onreadystatechange=function()
  {
     if(xmlhttpobj.readyState==4 && xmlhttpobj.status==200)
      {
       document.getElementById("c").innerHTML=xmlhttpobj.responseText;
      }
   }
xmlhttpobj.open("GET",url,true);
xmlhttpobj.send();
}
</script>
</head>
<body>
<form name="f1">
<input type="text" name="ct" onkeyup="load_data()">
</form>
<div id="c"></div>
</body>
</html>
```

php_connect.php

```
<?php
$dbhandle=mysql_connect("localhost","root","") or die("connection  failed");
$f=mysql_select_db("stdatabase",$dbhandle) or die("con fail 2");
$a=$_GET['val'];
$query="select * from student where rollno=$a";
$data1=mysql_query($query) or die(mysql_error);
echo "data from database <br>";
while($row=mysql_fetch_array($data1))
{
echo $row['stname']."<br>";
echo $row['rollno']."<br>";
echo $row['mark1']."<br>";
echo $row['mark2']."<br>";
}
?>
```

output


localhost / 127.0.0.1 / stdatabase
localhost/phpprgs/ajax/ajaxeg4.h
localhost/phpprgs/ajax/ajaxeg4.html
Apps
Untitled Page
Visual Web Develo...
Adobe Flash CS5+c...

Extracting data form server

101

```
data from database
lalitha
101
65
60
```