# THE EXTENSIBLE MARKUP LANGUAGE (XML)

**Abdallah Karakra & Sobhi Ahmed**

STUDENTS-HUB.com

Uploaded By: Jibreel Bornat

# WHAT IS XML?

- XML stands for EXtensible Markup Language
- XML is a markup language much like HTML
- XML was designed to carry data, not to display data
- XML tags are not predefined. You must define your owntags
- XML is a W3C Recommendation


Uploaded By: Jibreel Bornat

# MAIN COMPONENTS OF AN XML DOCUMENT

- Elements: <hello>
- Attributes: <item id="33905">
- Entities: < (<)
- Advanced Components

STUDENTS-HUB.com

Uploaded By: Jibreel Bornat

# THE BASIC RULES

- XML is case sensitive
- All start tags must have end tags
- Elements must be properly nested
- Every document must contain a root element
- Attribute values must have quotation marks

STUDENTS-HUB.com

Uploaded By: Jibreel Bornat

# THE DIFFERENCE BETWEEN XML AND HTML

- XML is not a replacement for HTML.
- XML and HTML were designed with different goals:
  - XML was designed to transport and store data, with focus on what data is
  - HTML was designed to display data, with focus on how data looks
- HTML is about displaying information, while XML is about carrying information.

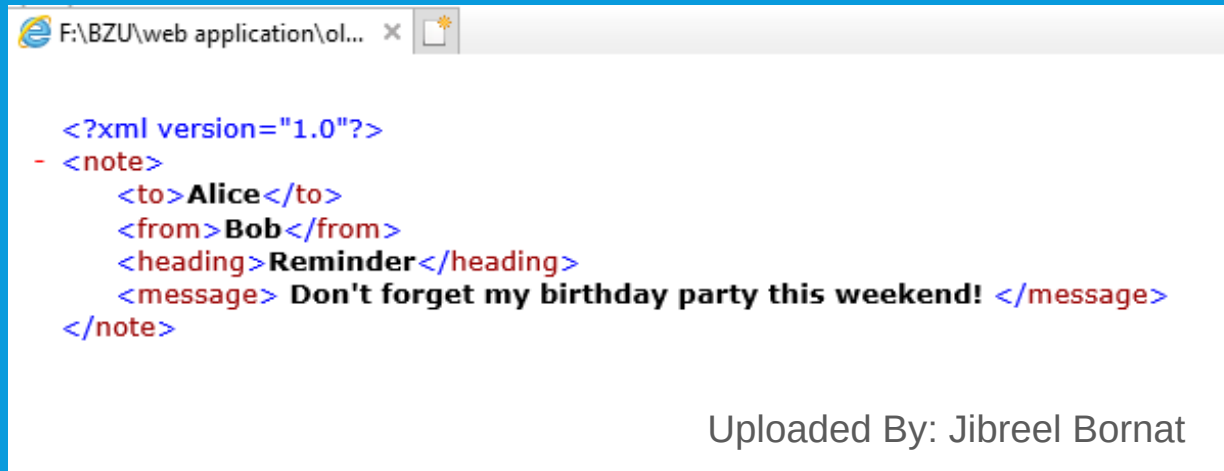
# XML DOES NOT DO ANYTHING

- Maybe it is a little hard to understand, but XML does not DO anything.

- XML was created to structure, store, and transport information.

# Look at the following example

```
<note>
 <to>Alice</to>
 <from>Bob</from>
 <heading>Reminder</heading>
 <message> Don't forget my birthday party this
weekend! </message>
</note>
```






Uploaded By: Jibreel Bornat

# THE DIFFERENCE BETWEEN XML AND HTML

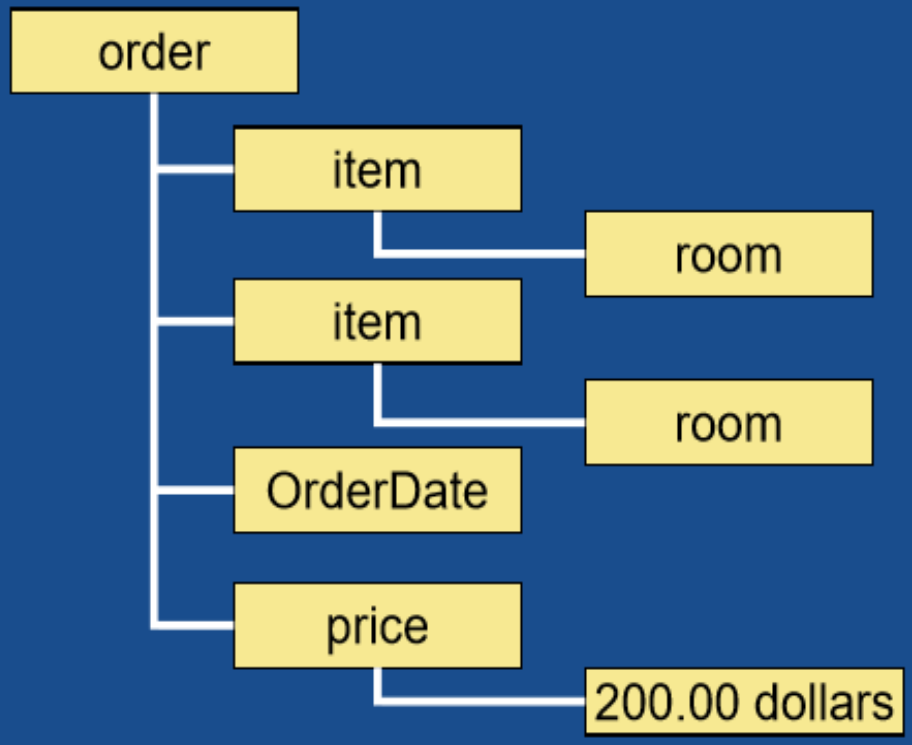
- How to write and store XML file?
- By using text file in .xml.

**Student Identification**
**ID Number:** 1
**Name:** Ali Ahmed
**BOD:** 9/9/1999
**Issuing Date:** 10/4/2011

XML flie:

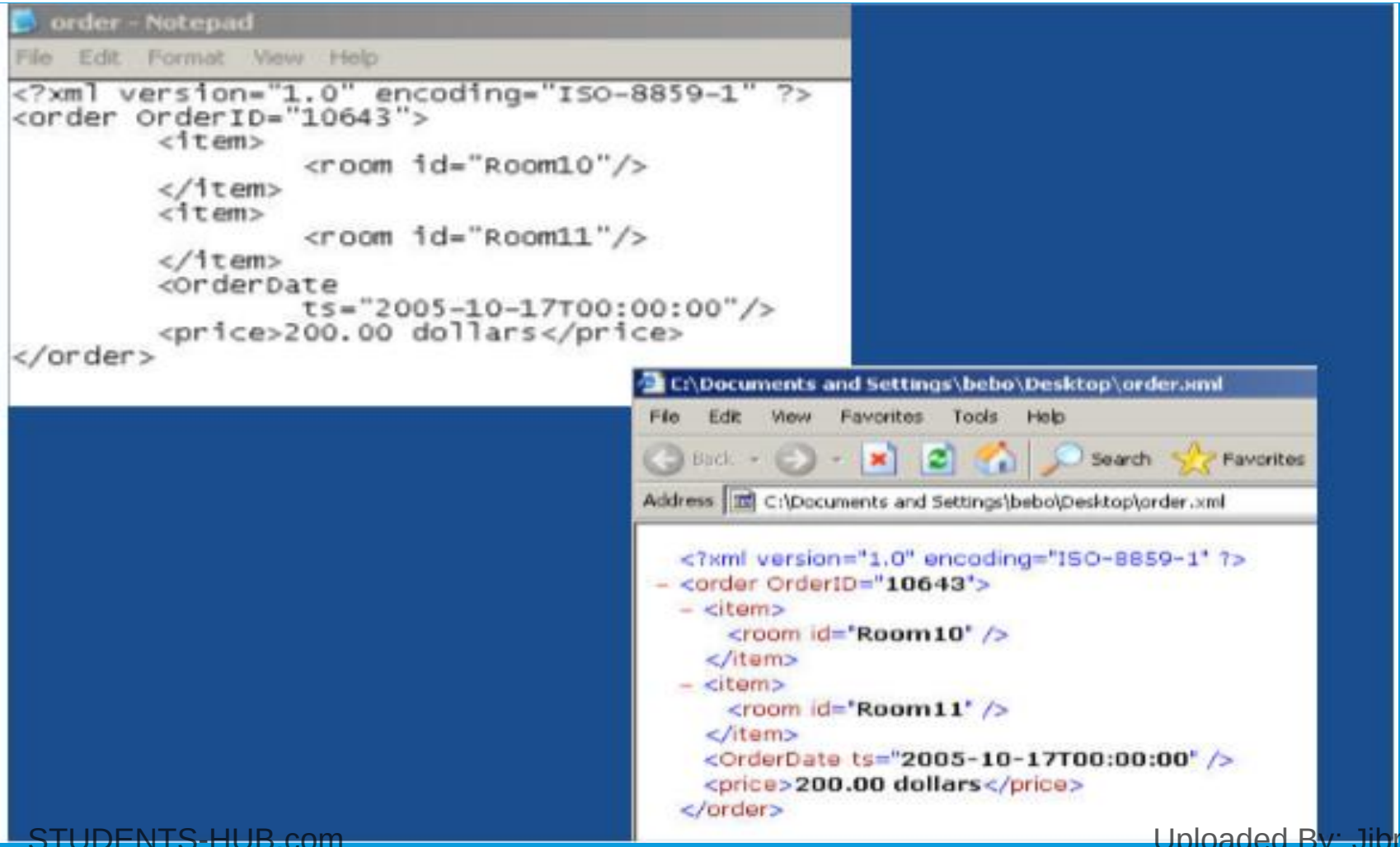
```
<studentID>
    <IdNumber>1</IdNumber>
    <Name> Ali Ahmed </Name>
    <BOD> 9/9/1999</BOD>
    <IssueDate> 10/4/2018</IssueDate>

</studentID>
```

# Look at the following example



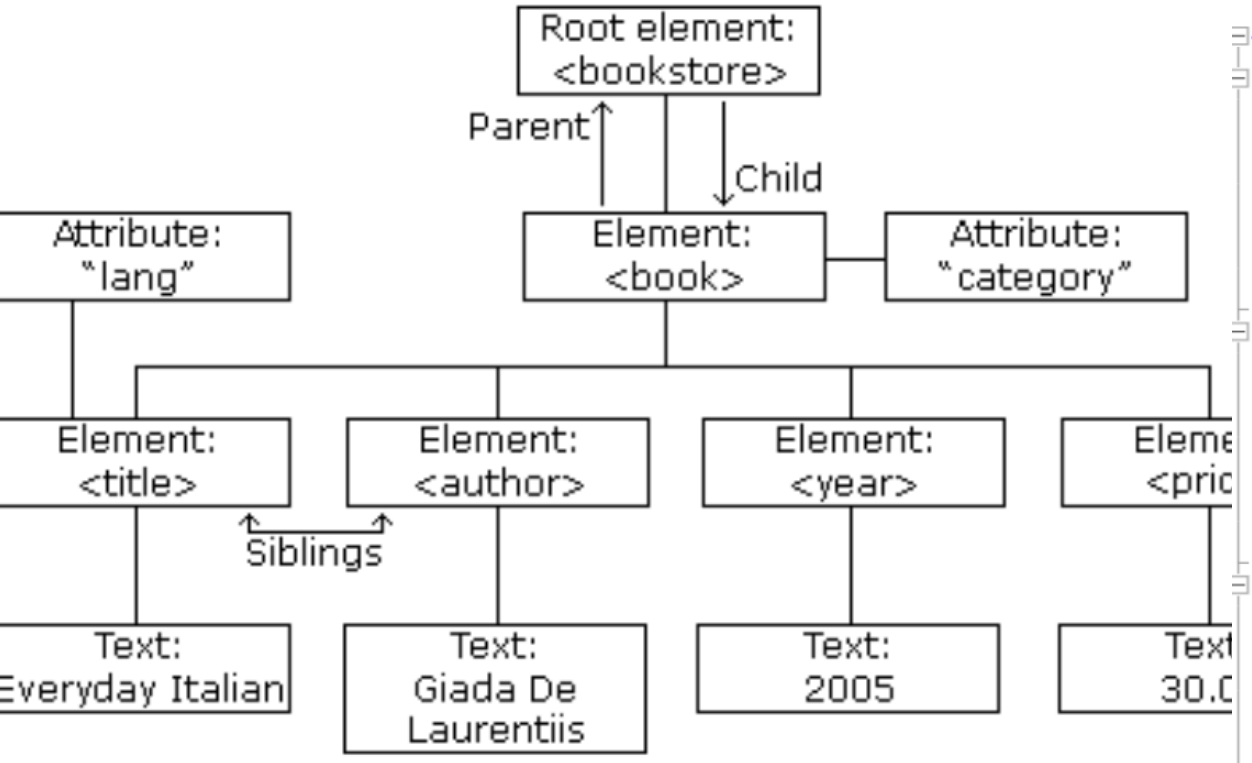

```
<?xml version="1.0“encoding="ISO-8859-1” ?>
<order OrderID="10643">
    <item>
        <room id=“Room10"l>
    </item>
    <item>
        <room id=“Room11"l>
    </item>
    <OrderDate
        ts="2005-10-17T00:00:00"l>
    <price>200.00 dollars</price>
</order>
```

# Look at the following example




STUDENTS-HUB.com
Uploaded By: Jibreel Bornat

# XML TREE STRUCTURE




```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
 <book category="cooking">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
 </book>
 <book category="children">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
 </book>
 <book category="web">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
 </book>
</bookstore>
```

# COMMON ERRORS FOR ELEMENT NAMING

- Do not use white space when creating names for elements
- Element names cannot begin with a digit, although names can contain digits
- Only certain punctuation allowed – periods, colons, and hyphens

STUDENTS-HUB.com

Uploaded By: Jibreel Bornat

# W3C DOM WITH JAVASCRIPT

- Example 1: Loading the XML document: DOMDocument
  - The programmer can use a Microsoft Active X object to parse an XML file

```
//Instantiate DOMDocument object
var XMLfile = new ActiveXObject("Msxml2.DOMDocument");
XMLfile.load("newspaper.xml");
var rootElement = XMLfile.documentElement;
document.write("The root node of the XML file is: ");
document.writeln("<b>" + rootElement.nodeName +"</b>");
```

# W3C DOM WITH JAVASCRIPT

- Example 2: Accessing the Children Elements
  - The *childNodes* member of any element node gives the programmer access to all of the sibling nodes of that element

```
//traverse through each child of the root element
//and print out its name
for (i=0; i<rootElement.childNodes.length; i++) {
      var node = rootElement.childNodes.item(i);
      document.write("The name of the node is ");
      document.write("<b>" + node.nodeName + "</b>");
}
```

# AJAX

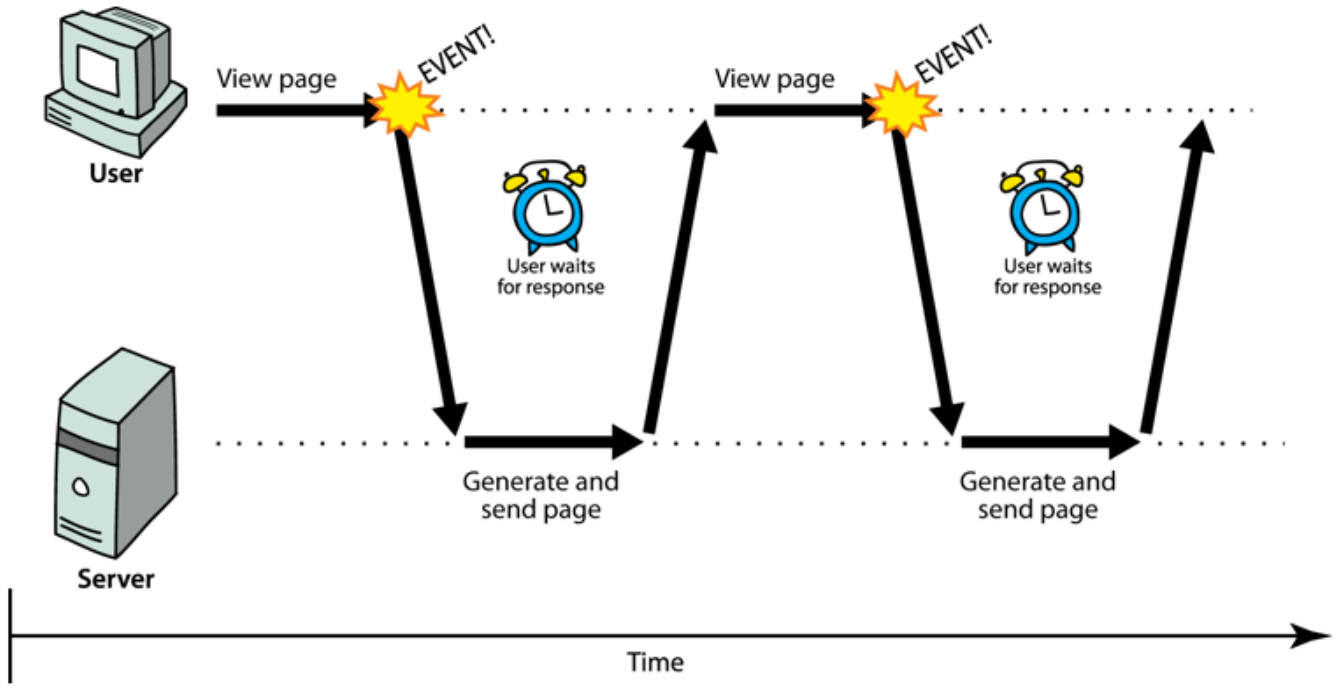
## Asynchronous JavaScript and XML

**Abdallah Karakra & Sobhi Ahmed**

STUDENTS-HUB.com

Uploaded By: Jibreel Bornat

# SYNCHRONOUS WEB COMMUNICATION




- **synchronous**: user must wait while new pages load

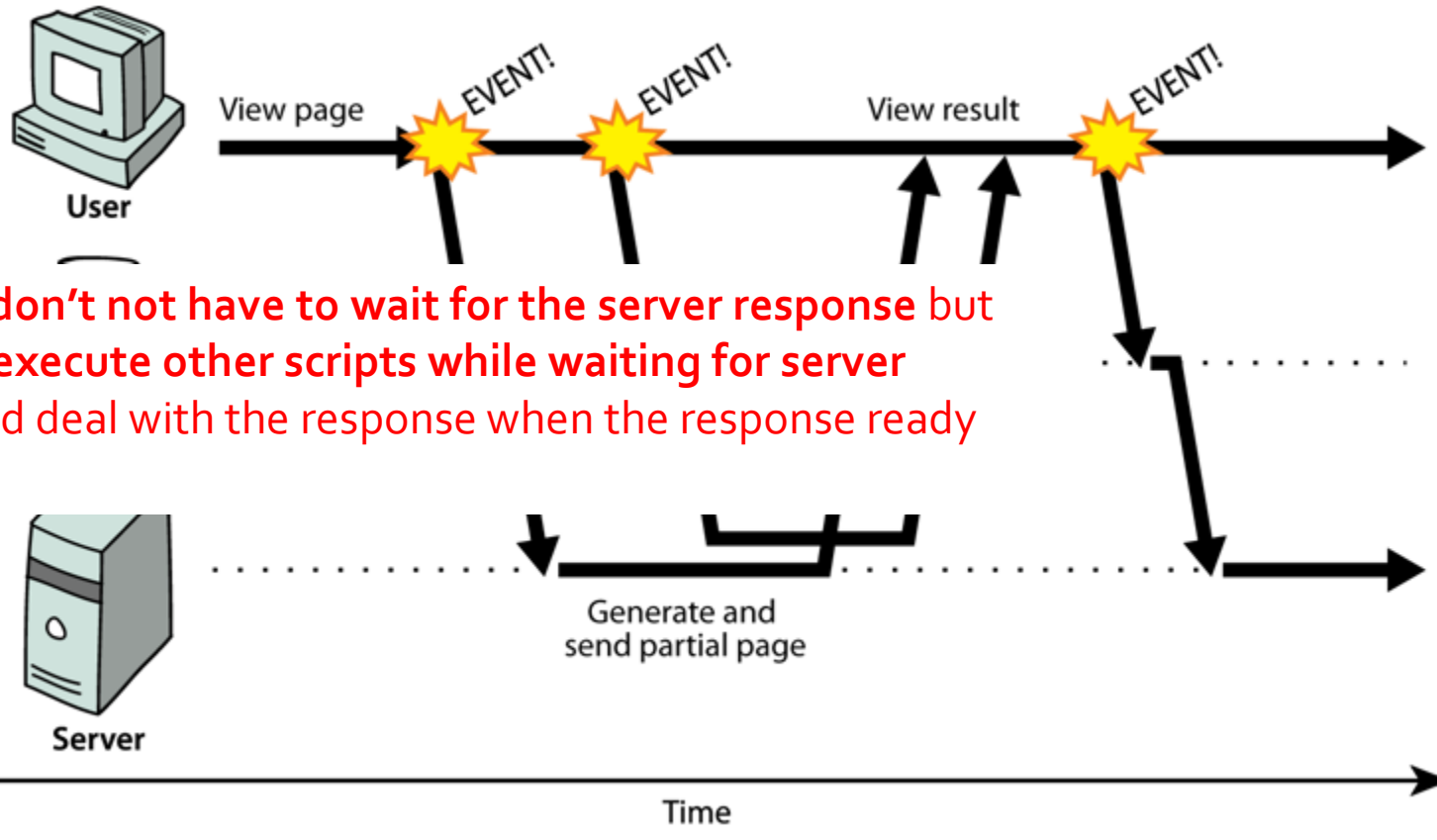
# **AJAX**: ASYNCHRONOUS JAVASCRIPT AND XM

- **Ajax**: Asynchronous JavaScript and XML
- not a programming language; a particular way of using JavaScript
- allows dynamically updating a page without making the user wait
- avoids the "click-wait-refresh" pattern

STUDENTS-HUB.com

Uploaded By: Jibreel Bornat

# ASYNCHRONOUS WEB COMMUNICATION




With AJAX, **don't not have to wait for the server response** but can instead **execute other scripts while waiting for server response** and deal with the response when the response ready

# SIMPLE EXAMPLE

```
<!DOCTYPE html>
<html>
<body>

<div id="demo">
<h1>The XMLHttpRequest Object</h1>
<button type="button" onclick="loadDoc()">Change Content</button>
</div>

<script>
function loadDoc() {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      document.getElementById("demo").innerHTML =
      this.responseText;
    }
  };
  xhttp.open("GET", "ajax_info.txt", true);
  xhttp.send();
}
</script>

</body>
</html>
```


STUDENTS-HUB.com
Uploaded By: Jibreel Bornat

# SIMPLE EXAMPLE (EXPLAINED)

1. Need a JavaScript function to get an 'XMLHttpRequest' object.
2. Once we have an XMLHttpRequest, we need to write a function that waits for it to get a response.
3. There are five possible states that the request can be in:
   - 0 = uninitialized
   - 1 = loading
   - 2 = loaded
   - 3 = interactive
   - 4 = complete

   - A method name is associated with these state changes
   - xhrequest.onreadystatechange = processResponse;
4. We need to know that the request was successful (http code 200 – 'OK')
5. Once everything is set up, we can send requests to URLs and be ready to get a response, This one sends a 'GET' request (asynchronous).

STUDENTS-HUB.com Uploaded By: Jibreel Bornat