Web Programming

YJ – Aug 2015

Course Outline

There're 16 lectures (2.5 hrs each) in this course. We'll mainly focus on web programming using LAMP model.

During the 24 hours tutorials in each week, our tutors will help you go through all the contents in the lecture notes and help you with the assignments
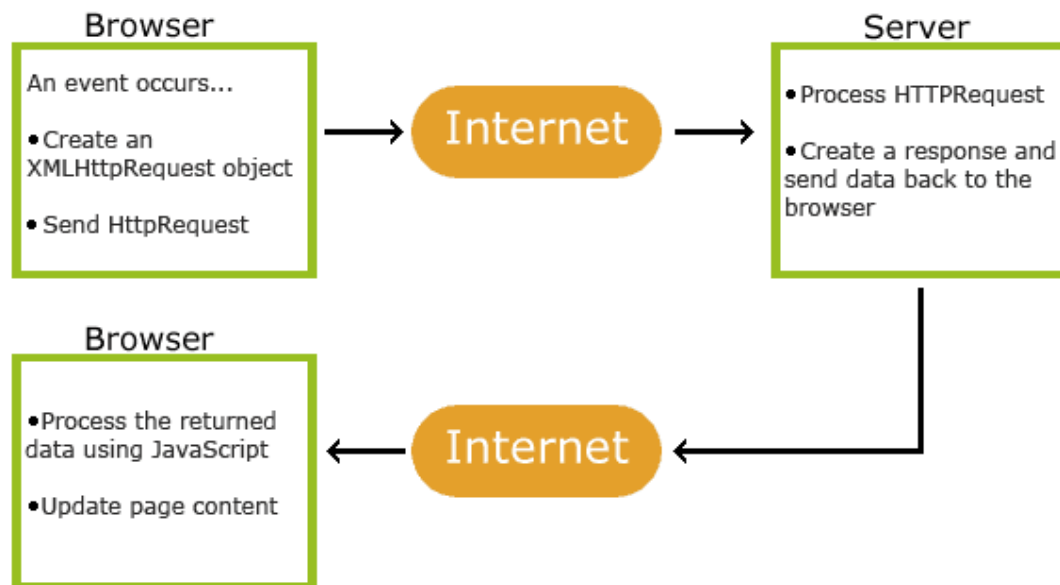
All scheduled lectures are learner-paced which means they could be altered depends on the situation. We help you gain rather than cram!

Course Outline

❖ L1 ~ Web programming intro

❖ L2 ~ HTML5

❖ L3 ~ CSS3

❖ L4 ~ JavaScripts

❖ L5 ~ PHP 1 Basic

❖ L6 ~ PHP 2 OO && Functions

❖ L7 ~ PHP 3 OO && Sessions

❖ L8 ~ PHP 4 Advanced features

❖ L9 ~ Mysql

❖ L10 ~ Mysql 2

❖ L11 ~ XML/JSON

❖ L12 ~ Ajax

❖ L13 ~ Framework: Bootstrap && Laravel

❖ L14 ~ CMS: Wordpress, Joomla, etc.

❖ L15 ~ Final Projects

❖ L16 ~ Final Projects

AJAX

❖ AJAX is about updating parts of a web page, without reloading the whole page.
❖ AJAX = Asynchronous JavaScript and XML.
❖ AJAX is a technique for creating fast and dynamic web pages.
❖ 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.
❖ Classic web pages, (which do not use AJAX) must reload the entire page if the content should change.

Browser

An event occurs...

• Create an XMLHttpRequest object

• Send HttpRequest

Internet

Server

• Process HTTPRequest

• Create a response and send data back to the browser

Browser

• Process the returned data using JavaScript

• Update page content

Internet

XMLHttpRequest

❖ XMLHttpRequest is a JavaScript object that provides an easy way to retrieve data from a URL without having to do a full page refresh.

❖ A Web page can update just a part of the page without disrupting what the user is doing. XMLHttpRequest is used heavily in AJAX programming.

❖ Despite its name, XMLHttpRequest can be used to retrieve any type of data, not just XML, and it supports protocols other than HTTP (including file and ftp).

var *myRequest* = **new** XMLHttpRequest();

Methods

❖ **open()**

Initializes a request. This method is to be used from JavaScript code.

void open(DOMString method, DOMString url, optional boolean async, optional DOMString? user, optional DOMString? password);

xmlhttp.open(**"GET"**, **"nameHint.php?q="** + str, **true**);

❖ **send()**

Sends the request. If the request is asynchronous (which is the default), this method returns as soon as the request is sent. If the request is synchronous, this method doesn't return until the response has arrived.

void send();
void send(*ArrayBufferView* data);
void send(**Blob** data);
void send(**Document** data);
void send(DOMString? data);
void send(**FormData** data);

Note: *Any event listeners you wish to set must be set before calling* send().

Properties

❖ XMLHttpRequest.onreadystatechange

Returns a EventHandler that is called whenever the readyState attribute changes. The callback is called from the user interface thread.

❖ XMLHttpRequest.readyState

Value	State	Description
0	UNSENT	open() has not been called yet.
1	OPENED	send() has been called.
2	HEADERS_RECEIVED	send() has been called, and headers and status are available.
3	LOADING	Downloading; responseText holds partial data.
4	DONE	The operation is complete.

Properties

❖ XMLHttpRequest.status

Returns an unsigned short with the status of the response of the request. This is the HTTP result code (for example, status is 200 for a successful request).

❖ XMLHttpRequest.statusText

Returns a DOMString containing the response string returned by the HTTP server. Unlike XMLHTTPRequest.status, this includes the entire text of the response message ("200 OK", for example).

Properties

❖ XMLHttpRequest.response

Returns an ArrayBuffer, Blob, Document, JavaScript object, or a DOMString, depending of the value of XMLHttpRequest.responseType. that contains the response entity body. This is null if the request is not complete or was not successful.

❖ XMLHttpRequest.responseText

Returns a DOMString that contains the response to the request as text, or null if the request was unsuccessful or has not yet been sent.

❖ XMLHttpRequest.responseType

Is an enumerated value that defines the response type. It can have the following values:""(DOMString (this is the default value)), "arraybuffer", "document", "json", "text"

Browser compatibility

Feature	Chrome	Firefox (Gecko)	Internet Explorer	Opera	Safari (WebKit)
Basic support (XHR1)	1	1.0 (1.7 or earlier)[1]	5[2] 7	(Yes)	1.2
send(ArrayBuffer)	9	9.0 (9.0)	10	11.60	?
send(ArrayBufferView)	22	20.0 (20.0)	?	?	?
send(Blob)	7	3.6 (1.9.2)	10	12	?
send(FormData)	6	4.0 (2.0)	10	12	?
sendAsBinary(DOMString) ⚠ 👎	Not supported[3]	2.0 (1.8.1)	Not supported	Not supported	Not supported
response	10	6.0 (6.0)	10	11.60	(Yes)
responseType = 'arraybuffer'	10	6.0 (6.0)	10	11.60	(Yes)
responseType = 'blob'	19	6.0 (6.0)	10	12	(Yes)
responseType = 'document'	18	11.0 (11.0)	10	Not supported	6.1
responseType = 'json'	31	10.0 (10.0)	Not supported	12[4] Not supported 16 17	(Yes)
Progress Events	7	3.5 (1.9.1)	10	12	(Yes)
withCredentials	3	3.5 (1.9.1)	10	12	4
timeout	29.0[5]	12.0 (12.0)	8	12[6] 16	(Yes)
responseType = 'moz-blob'	Not supported	12.0 (12.0)	Not supported	Not supported	Not supported