Computer Science 571 MidTerm
Prof. Horowitz

Thursday, October 10, 2013, 9:30am – 10:45am

Name: **USC ID:**

1. This is a closed book exam.
2. Please answer all questions.
3. Place all answers on the exam and return the entire exam
4. The exam contains 14 pages (including the cover page). There are 33 questions which are mainly distributed into 4 sections as the following table:

Section	Student Grade	Max
DOM		**/30**
JavaScript		**/25**
HTTP		**/24**
CGI and PHP		**/21**
		/100

DOM QUESTIONS #1-#15 [30 Points]

The following set of examples makes use of this XML file (books.xml):

```
<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">XQuery Kick Start</title>
        <author>James McGovern</author>
        <author>Per Bothner</author>
        <author>Kurt Cagle</author>
        <author>James Linn</author>
        <author>Vaidyanathan Nagarajan</author>
        <year>2003</year>
        <price>49.99</price>
    </book>
    <book category="web" cover="paperback">
        <title lang="en">Learning XML</title>
        <author>Erik T. Ray</author>
        <year>2003</year>
        <price>39.95</price>
    </book>
</bookstore>
```

To start the examples we need a way of downloading the XML file. This is done using the function loadXMLDoc that follows. Fill in the missing code:

```
function loadXMLDoc(dname)
{

    if (window.XMLHttpRequest)
    { xhttp=new XMLHttpRequest(); }
    else
    { xhttp=new ActiveXObject("Microsoft.XMLHTTP"); }
    .........................................................................................................   1.
    .........................................................................................................   2.
    return xhttp.responseXML;

}
```

Example 1: Read the code, look at the output and then provide the missing code

```
<!DOCTYPE html><html>
<head>
<script src="loadxmldoc.js">
</script></head>
<body><script type="text/javascript">
xmlDoc=loadXMLDoc("books.xml");
document.write("Number of book nodes: ");
```
.... **3.**
```
document.write("<br>");
//Remove the first book
```
.... **4.**
.... **5.**
```
document.write("Number of book nodes after removing the first
book: ");
```
.... **6.**
```
</script></body></html>
```

Output

Number of book nodes: 4
Number of book nodes after removing the first book: 3

Example 2: Read the code, look at the output and then provide the missing code

```
<!DOCTYPE html><html><head>
<script src="loadxmldoc.js"> </script>
</head><body><script>
xmlDoc=loadXMLDoc("books.xml");
x=xmlDoc.getElementsByTagName('book');
```

.... **7.**
```
document.write("<br>");
document.write("Removing category attribute of the first book");
```
.... **8.**
.... **9.**
```
</script></body></html>
```

Output

Book 1 – category: cooking
Removing category attribute of the first book
Book 1 – category: null

Example 3: Read the code, look at the output and then provide the missing code

```
<!DOCTYPE html>
<html>
<head>
    <script src="loadxmldoc.js"></script>
</head>
<body><script>
xmlDoc=loadXMLDoc("books.xml");
newNode=xmlDoc.createElement("book");
....................................................................................................................    10.
....................................................................................................................    11.
document.write("<br>");
....................................................................................................................    12.
....................................................................................................................    13.
....................................................................................................................    14.
document.write("Book elements after: " + ...................................);  15.
</script></body></html>
```

Output

Book elements before: 4
Book elements after: 5

JavaScript Questions #16-#20 [25 Points]

Below is a web page that can be used to check the validity of regular expressions
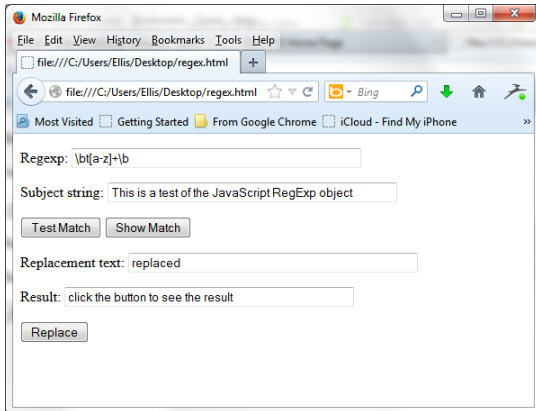
Figure 1: Original Screen
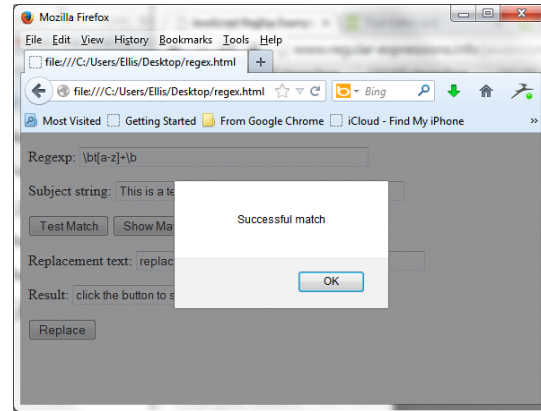
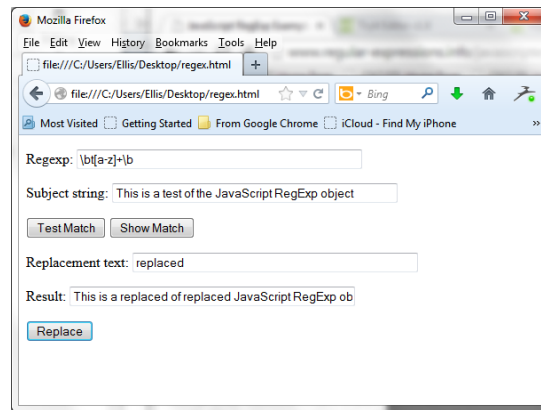Figure 2: Clicking on Test Match
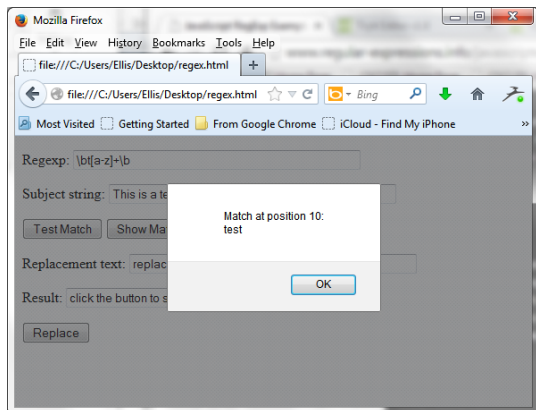
Figure 3: Click on Show Match

Figure 4: Click on Replace

Below is the code that implements the web pages above, but some of the code is missing. Supply the missing code:

```
<SCRIPT LANGUAGE="JavaScript"><!--
function demoMatchClick() {
```
                                                                        16.
```
  if (document.demoMatch.subject.value.match(re)) {
    alert("Successful match");
  } else {
    alert("No match");
  }
}
function demoShowMatchClick() {
```
                                                                        17.
                                                                        18.
```
  if (m == null) {
```

```
        alert("No match");
    } else {
        var s = "Match at position " + m.index + ":\n";
        for (i = 0; i < m.length; i++) {
            s = s + m[i] + "\n";
        }
        alert(s);
    }
}
function demoReplaceClick() {
    var re = new RegExp(document.demoMatch.regex.value, "g");
    document.demoMatch.result.value =
        document.demoMatch.subject.value.replace(re,
            ………………………………………………………………………………); 19.
}
// -->
</SCRIPT>

<FORM ID="demoMatch" NAME="demoMatch" METHOD=POST
ACTION="javascript:void(0)">
<P>Regexp:
<INPUT TYPE=TEXT NAME="regex"
VALUE="………………………………………………………………" SIZE=50></P>        20.
<P>Subject string:
<INPUT TYPE=TEXT NAME="subject" VALUE="This is a test of the
JavaScript RegExp object" SIZE=50>
</P>
<P>
<INPUT TYPE=SUBMIT VALUE="Test Match" NCLICK="demoMatchClick()">
<INPUT TYPE=SUBMIT VALUE="Show Match"
ONCLICK="demoShowMatchClick()"></P>

<P>Replacement text:
<INPUT TYPE=TEXT NAME="replacement" VALUE="replaced"
SIZE=50></P>
<P>Result:
<INPUT TYPE=TEXT NAME="result" VALUE="click the button to see
the result" SIZE=50></P>
<P><INPUT TYPE=SUBMIT VALUE="Replace"
ONCLICK="demoReplaceClick()"></P>
</FORM>
```

HTTP Questions #21-#26 [24 Points]

Below, are the HTTP headers exchanged between the browser and www.usc.edu. The transactions have been edited, leaving only a few of the transactions.

```
http://www.usc.edu/

GET / HTTP/1.1
Host: www.usc.edu
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: __utma=185748653.285999088.1365278272.1380062183.1380911712.11;
__utmz=185748653.1380911712.11.5.utmccn=(referral)|utmcsr=www-
scf.usc.edu|utmcct=/~csci571/2013Fall/lynda.html|utmcmd=referral;
__unam=304c066-13dea736b4b-47b5ef49-13; soc_textsize=small;
desktopCookie=uschomepage; __utmc=185748653
Connection: keep-alive


HTTP/1.1 200 OK
Date: Fri, 04 Oct 2013 21:41:30 GMT
Server: Apache/2.2.11 (Unix) DAV/2 mod_ssl/2.2.11 OpenSSL/0.9.8j
Content-Encoding: gzip
Vary: Accept-Encoding
Content-Length: 4120
Keep-Alive: timeout=1, max=300
Connection: Keep-Alive
Content-Type: text/html
-----------------------------------------------------------
http://www.usc.edu/favicon.ico

GET /favicon.ico HTTP/1.1
Host: www.usc.edu
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: __utma=185748653.285999088.1365278272.1380062183.1380911712.11;
__utmz=185748653.1380911712.11.5.utmccn=(referral)|utmcsr=www-
scf.usc.edu|utmcct=/~csci571/2013Fall/lynda.html|utmcmd=referral;
__unam=304c066-13dea736b4b-47b5ef49-13; soc_textsize=small;
desktopCookie=uschomepage; __utmc=185748653
Connection: keep-alive
If-Modified-Since: Tue, 01 Nov 2011 00:14:16 GMT

HTTP/1.1 304 Not Modified
Date: Fri, 04 Oct 2013 21:41:30 GMT
Server: Apache/2.2.11 (Unix) DAV/2 mod_ssl/2.2.11 OpenSSL/0.9.8j
Connection: Keep-Alive
Keep-Alive: timeout=1, max=300
```

```
Expires: Thu, 15 Apr 2012 20:00:00 GMT
----------------------------------------------------------
http://www.usc.edu/x/usc_homepage/assets/css/usc.css?v=6

GET /x/usc_homepage/assets/css/usc.css?v=6 HTTP/1.1
Host: www.usc.edu
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
Accept: text/css,*/*;q=0.1
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.usc.edu/
Cookie: __utma=185748653.285999088.1365278272.1380062183.1380911712.11;
__utmz=185748653.1380911712.11.5.utmccn=(referral)|utmcsr=www-
scf.usc.edu|utmcct=/~csci571/2013Fall/lynda.html|utmcmd=referral;
__unam=304c066-13dea736b4b-47b5ef49-13; soc_textsize=small;
desktopCookie=uschomepage; __utmc=185748653
Connection: keep-alive

HTTP/1.1 200 OK
Date: Fri, 04 Oct 2013 21:41:30 GMT
Server: Apache/2.2.11 (Unix) DAV/2 mod_ssl/2.2.11 OpenSSL/0.9.8j
Last-Modified: Wed, 05 Jun 2013 22:56:22 GMT
Accept-Ranges: bytes
Content-Length: 27344
Expires: Thu, 15 Apr 2012 20:00:00 GMT
Keep-Alive: timeout=1, max=299
Connection: Keep-Alive
Content-Type: text/css
----------------------------------------------------------
http://www.usc.edu/x/usc_homepage/assets/images/weather_icons/34.png

GET /x/usc_homepage/assets/images/weather_icons/34.png HTTP/1.1
Host: www.usc.edu
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
Accept: image/png,image/*;q=0.8,*/*;q=0.5
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.usc.edu/
Cookie: __utma=185748653.285999088.1365278272.1380062183.1380911712.11;
__utmz=185748653.1380911712.11.5.utmccn=(referral)|utmcsr=www-
scf.usc.edu|utmcct=/~csci571/2013Fall/lynda.html|utmcmd=referral;
__unam=304c066-13dea736b4b-47b5ef49-13; soc_textsize=small;
desktopCookie=uschomepage; __utmc=185748653
Connection: keep-alive

HTTP/1.1 200 OK
Date: Fri, 04 Oct 2013 21:41:30 GMT
Server: Apache/2.2.11 (Unix) DAV/2 mod_ssl/2.2.11 OpenSSL/0.9.8j
Last-Modified: Wed, 07 Jul 2010 20:33:18 GMT
Accept-Ranges: bytes
Content-Length: 1509
Expires: Thu, 15 Apr 2012 20:00:00 GMT
Keep-Alive: timeout=1, max=298
Connection: Keep-Alive
```

```
Content-Type: image/png
------------------------------------------------------------
http://www.usc.edu/x/usc_homepage/assets/js/usc_v4.js?v=4

GET /x/usc_homepage/assets/js/usc_v4.js?v=4 HTTP/1.1
Host: www.usc.edu
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.usc.edu/
Cookie: __utma=185748653.285999088.1365278272.1380062183.1380911712.11;
__utmz=185748653.1380911712.11.5.utmccn=(referral)|utmcsr=www-
scf.usc.edu|utmcct=/~csci571/2013Fall/lynda.html|utmcmd=referral;
__unam=304c066-13dea736b4b-47b5ef49-13; soc_textsize=small;
desktopCookie=uschomepage; __utmc=185748653
Connection: keep-alive

HTTP/1.1 200 OK
Date: Fri, 04 Oct 2013 21:41:30 GMT
Server: Apache/2.2.11 (Unix) DAV/2 mod_ssl/2.2.11 OpenSSL/0.9.8j
Last-Modified: Fri, 06 Jan 2012 18:04:12 GMT
Accept-Ranges: bytes
Content-Length: 9447
Expires: Thu, 15 Apr 2012 20:00:00 GMT
Keep-Alive: timeout=1, max=299
Connection: Keep-Alive
Content-Type: application/x-javascript
------------------------------------------------------------
http://www.usc.edu/x/usc_homepage/assets/css/fonts/ProximaNova-Reg-
webfont.woff

GET /x/usc_homepage/assets/css/fonts/ProximaNova-Reg-webfont.woff HTTP/1.1
Host: www.usc.edu
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.usc.edu/x/usc_homepage/assets/css/usc.css?v=6
Cookie: __utma=185748653.285999088.1365278272.1380062183.1380911712.11;
__utmz=185748653.1380911712.11.5.utmccn=(referral)|utmcsr=www-
scf.usc.edu|utmcct=/~csci571/2013Fall/lynda.html|utmcmd=referral;
__unam=304c066-13dea736b4b-47b5ef49-13; soc_textsize=small;
desktopCookie=uschomepage; __utmc=185748653
Connection: keep-alive

HTTP/1.1 200 OK
Date: Fri, 04 Oct 2013 21:41:30 GMT
Server: Apache/2.2.11 (Unix) DAV/2 mod_ssl/2.2.11 OpenSSL/0.9.8j
Last-Modified: Wed, 21 Dec 2011 23:58:18 GMT
Etag: "49312-5cf0-4b4a2f533cf87"
Accept-Ranges: bytes
Content-Length: 23792
Keep-Alive: timeout=1, max=297
```

```
Connection: Keep-Alive
Content-Type: text/plain
-------------------------------------------------------------
http://www.google-analytics.com/ga.js

GET /ga.js HTTP/1.1
Host: www.google-analytics.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://www.usc.edu/
Connection: keep-alive
If-Modified-Since: Thu, 26 Sep 2013 19:47:03 GMT

HTTP/1.1 304 Not Modified
Date: Fri, 04 Oct 2013 19:02:25 GMT
Expires: Sat, 05 Oct 2013 07:02:25 GMT
Age: 9545
Server: GFE/2.0
Alternate-Protocol: 80:quic
-------------------------------------------------------------
http://www.usc.edu/x/usc_homepage/slideshow.php

GET /x/usc_homepage/slideshow.php HTTP/1.1
Host: www.usc.edu
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
X-Requested-With: XMLHttpRequest
Referer: http://www.usc.edu/
Cookie: __utma=185748653.285999088.1365278272.1380062183.1380911712.11;
__utmz=185748653.1380911712.11.5.utmccn=(referral)|utmcsr=www-
scf.usc.edu|utmcct=/~csci571/2013Fall/lynda.html|utmcmd=referral;
__unam=304c066-13dea736b4b-47b5ef49-13; soc_textsize=small;
desktopCookie=uschomepage; __utmc=185748653
Connection: keep-alive

HTTP/1.1 200 OK
Date: Fri, 04 Oct 2013 21:41:30 GMT
Server: Apache/2.2.11 (Unix) DAV/2 mod_ssl/2.2.11 OpenSSL/0.9.8j
Content-Length: 3711
Keep-Alive: timeout=1, max=299
Connection: Keep-Alive
Content-Type: text/html
```

21. [4 pts] How many web servers are involved in these HTTP transactions and what are their complete names?

22. [4 pts] How many cookies are being set in these transactions and what are their names?

23. [4 pts] What is the name and version of the browser used in these transactions?

24. [4 pts] What webkit is the browser using?

25. [4 pts] what is the full URL of either one of the JavaScript files that are downloaded?

26. [4 pts] Transaction 2 returns "304 Not Modified". What does that mean in terms of dates?

CGI and PHP Questions #27-#33 [21 Points]

Below is one example of the output of the program show_vars.php, a program that prints the CGI environment variables for a particular invocation. Following the output is the actual PHP source code, with some of the code removed. Please provide the missing code.

$_SERVER	$_POST	$_GET

DOCUMENT_ROOT: /var/local/www/htdocs
HTTP_ACCEPT: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
HTTP_ACCEPT_ENCODING: gzip, deflate
HTTP_ACCEPT_LANGUAGE: en-US,en;q=0.5
HTTP_CONNECTION: keep-alive
HTTP_COOKIE: __utma=185748653.285999088.1365278272.1380911712.1380922899.12;
__utmz=185748653.1380911712.11.5.utmcsr=www-
scf.usc.edu|utmccn=(referral)|utmcmd=referral|utmcct=/~csci571/2013Fall/lynda.html;
__unam=304c066-13dea736b4b-47b5ef49-13; soc_textsize=small;
desktopCookie=uschomepage; __utmc=185748653
HTTP_HOST: www-scf.usc.edu
HTTP_REFERER: http://www-scf.usc.edu/~csci571/special.html
HTTP_USER_AGENT: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:24.0) Gecko/20100101
Firefox/24.0
PATH:
 /usr/bin:/usr/sbin:/sbin:/usr/lsd/sbin:/usr/lsd/etc:/usr/lsd/bin:/usr/ucs/bin:/usr/
spac/bin:/usr/usc/etc:/usr/usc/bin:/usr/usc/X/bin:/usr/usc/X11/bin:/usr/lib/nis:/usr
/lib/netsvc/yp:/usr/ccs/bin
REMOTE_ADDR: 76.89.147.160
REMOTE_PORT: 49818
SCRIPT_FILENAME: /home/scf-22/csci571/public_html/Special/php_ex/show_vars.php
SCRIPT_URI: http://www-scf.usc.edu/~csci571/Special/php_ex/show_vars.php
SCRIPT_URL: /~csci571/Special/php_ex/show_vars.php
SERVER_ADDR: 68.181.201.23
	empty	empty

```
SERVER_ADMIN:   webadm@usc.edu
SERVER_NAME:   www-scf.usc.edu
SERVER_PORT:   80
SERVER_SIGNATURE:
Apache/1.3.31 Server at www-scf.usc.edu Port 80

SERVER_SOFTWARE:   Apache/1.3.31 (Unix) PHP/4.3.9 mod_layout/3.0.3
UNIQUE_ID:   Uk@TEES1yRcAADLrIYQ
GATEWAY_INTERFACE:   CGI/1.1
SERVER_PROTOCOL:   HTTP/1.1
REQUEST_METHOD:   GET
QUERY_STRING:
REQUEST_URI:   /%7Ecsci571/Special/php_ex/show_vars.php
SCRIPT_NAME:   /~csci571/Special/php_ex/show_vars.php
PATH_TRANSLATED:   /home/scf-22/csci571/public_html/Special/php_ex/show_vars.php
PHP_SELF:   /~csci571/Special/php_ex/show_vars.php
argv:   Array
empty
argc:   0
```

- **A form with method = POST**

  First Name [   ]

  Last Name [   ]

  [Submit]

- **A form with method = GET**

  First Name [   ]

  Last Name [   ]

  [Submit]

Source code for show_vars.php

```
<!doctype html>
<html>
<head><title>Show $_SERVER, $_POST, $_GET
Variables</title></head>
<style type="text/css">
th {   text-align:left;}
</style>
<body>
…………………………………………………………………………………………………………………………    27.
      function print_tabs($tabs) {
            for($i = 0; $i < …………………………………………; $i++)    28.
            {
                  echo "     ";
            }
      }
```

```php
        function print_array($arr, $tabs = 0) {
                if(!empty($arr))
                {
                foreach($arr as $k=> $v)
                {
                print_tabs($tabs);
                echo ……………………  . ":" . ……………………  . "<br/>";    .29.30
                if(is_array($v))
                        print_array($v, $tabs+1);
                }//end for loop
                }//end outer if
                else
                        echo "empty";
        }
?>
<table border='1'>
<tr>
        <th width="33%">$_SERVER</th>
        <th width="34%">$_POST</th>
        <th width="33%">$_GET</th>
</tr>
<tr>
        <td valign="top">
        ……………………………………………………………………………………………………    .31
        </td>
        <td valign="top">
        ……………………………………………………………………………………………………    .32
        </td>
        <td valign="top">
        ……………………………………………………………………………………………………    .33
        </td>
        </tr>
</table>
<ul>
<li>
<a href="">A form with method = POST</a>
<div>
<form method="POST" action="show_vars.php">
<p>
        <label for="fname">First Name</label>
        <input type="text" value="" name="fname">
</p>
<p>
        <label for="lname">Last Name</label>
        <input type="text" value="" name="lname">
</p>
```

```
<p>
      <input type="submit" value="Submit" name="submit">
</p>
</form>
</div>
</li>
<li>
<a href="">A form with method = GET</a>
<div>
<form method="GET" action="show_vars.php">
<p>
      <label for="fname">First Name</label>
      <input type="text" value="" name="fname">
</p>
<p>
      <label for="lname">Last Name</label>
      <input type="text" value="" name="lname">
</p>
<p>
      <input type="submit" value="Submit" name="submit">
</p>
</form>
</div>
</li>
</ul>
</body>
</html>
```

The End
Good Luck
☺