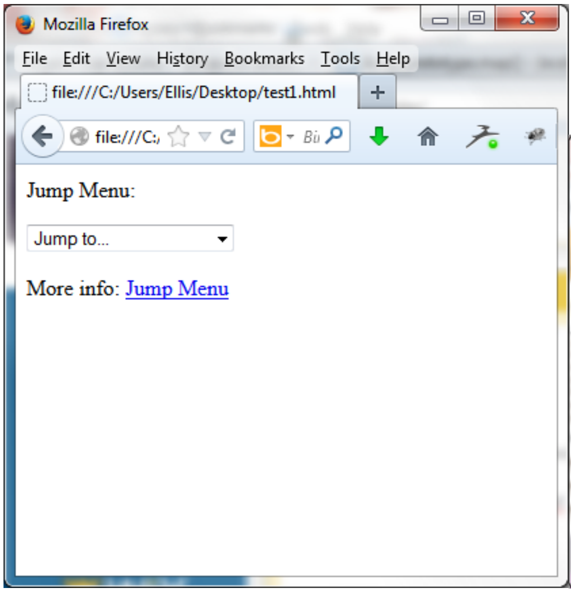
Computer Science 571 Exam #1
Prof. Papa
Tuesday, October 2, 2018, 5:30pm – 6:50pm

Sessions: SAL 101 [a-m] and THH 201 [n-z]

Name: **Id Number:** **USC Email:**

1. This is a closed book exam.
2. Please answer all questions.
3. Place all answers on the exam and return the entire exam

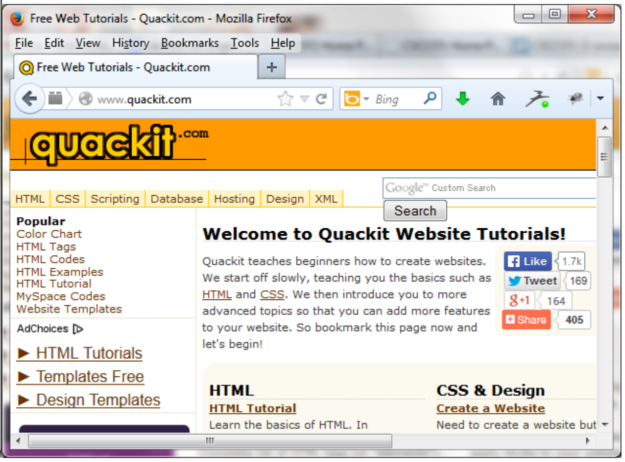
JAVASCRIPT QUESTIONS [30 POINTS]


Mozilla Firefox
File Edit View History Bookmarks Tools Help
file:///C:/Users/Ellis/Desktop/test1.html
file:///C:

Jump Menu:

Jump to...

More info: Jump Menu


Free Web Tutorials - Quackit.com - Mozilla Firefox
File Edit View History Bookmarks Tools Help
Free Web Tutorials - Quackit.com
www.quackit.com
Bing
quackit.com
HTML CSS Scripting Database Hosting Design XML
Google™ Custom Search
Search
Like 1.7k
Tweet 169
+1 164
Share 405
AdChoices

Popular
Color Chart
HTML Tags
HTML Codes
HTML Examples
HTML Tutorial
MySpace Codes
Website Templates

▶ HTML Tutorials
▶ Templates Free
▶ Design Templates

Welcome to Quackit Website Tutorials!

Quackit teaches beginners how to create websites. We start off slowly, teaching you the basics such as HTML and CSS. We then introduce you to more advanced topics so that you can add more features to your website. So bookmark this page now and let's begin!

HTML
HTML Tutorial
Learn the basics of HTML. In

CSS & Design
Create a Website
Need to create a website but

[10 pts] Above to the left is a screenshot of the web page produced by the source code below and to the right is a new page containing the web page that results when the first selection is chosen. Fill in the missing code below.

```
<script language="javascript" type="text/javascript" >
function jumpto(x){
       window.open(x, '_blank');
}

</script>
<p>Jump Menu:</p>
<form name="form1">
<select name="jumpmenu"
```

```
onChange=
"jumpto(document.form1.jumpmenu.options[document.form1.jumpmenu.options.selec
tedIndex].value)">

<option>Jump to...</option>

<option value=http://www.quackit.com>Quackit Homepage</option>
  <option value=http://www.quackit.com/javascript/>JavaScript</option>
  <option value=http://www.quackit.com/html/>HTML</option>
  <option value=http://www.quackit.com/css/>CSS</option>
  <option value=http://www.quackit.com/sql/tutorial/>SQL</option>
  <option value=http://www.quackit.com/database/tutorial/>Database
Tutorial</option>
  <option value=http://www.quackit.com/web_hosting/>Web Hosting
Tutorial</option>
</select>
</form>
<div class="more-info"> <p>More info: <a
href="/javascript/javascript_dropdown_menu.cfm">Jump Menu</a></p>
</div>
```

[5 pts] What are the values of x(6) and y(7)?

```
function makeAdder(a) {
    return function(b) {
    return a+b;
  };
}

x = makeAdder(5);
y = makeAdder(20);
```

`x(6) ?` **`[2 1⁄2 pts]`**

`y(7) ?` **`[2 1⁄2 pts]`**

Answer: X(6) returns 11 and y(7) returns 27; makeAdder(5) returns an anonymous function which adds 5 to the function argument; makeAdder(20) returns an anonymous function which adds 20 to the function argument;

The regular expression below includes three sets of parentheses (...). In the following questions, explain the purpose of the patterns contained in each of the parentheses.

var pattern = "^(\w+)@([a-zA-Z_]+)\.([a-zA-Z]{2,3})$";

Answer:

[5 pts] What is the purpose of the pattern within the first set of parentheses: (\w+)

Answer: One or more alphanumeric characters

[5 pts] What is the purpose of the pattern within the second set of parentheses:

([a-zA-Z_]+)

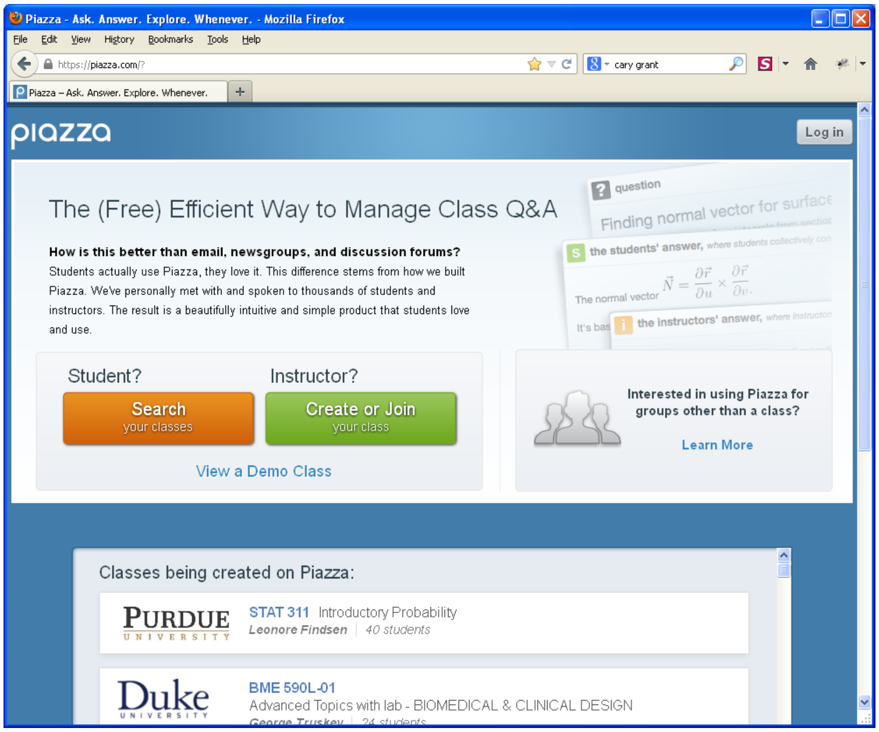
Answer: one or more alphabetic upper/lower case or underscore

[5 pts] What is the purpose of the pattern within the third set of parentheses:

([a-zA-Z]{2,3})

Answer: two or three alphabetic characters

HTTP QUESTIONS [20 POINTS]


Piazza - Ask. Answer. Explore. Whenever. - Mozilla Firefox
File Edit View History Bookmarks Tools Help
https://piazza.com/?
cary grant
Piazza – Ask. Answer. Explore. Whenever.
piazza
Log in
question
Finding normal vector for surface
the students' answer, where students collectively con
The normal vector
$\vec{N} = \frac{\partial \vec{r}}{\partial u} \times \frac{\partial \vec{r}}{\partial v}.$
It's bas
the instructors' answer, where instructor
Student?
Search
your classes
Instructor?
Create or Join
your class
View a Demo Class
Learn More
PURDUE UNIVERSITY
Duke UNIVERSITY

The (Free) Efficient Way to Manage Class Q&A

How is this better than email, newsgroups, and discussion forums?
Students actually use Piazza, they love it. This difference stems from how we built Piazza. We've personally met with and spoken to thousands of students and instructors. The result is a beautifully intuitive and simple product that students love and use.

Interested in using Piazza for groups other than a class?

Classes being created on Piazza:

STAT 311 Introductory Probability
Leonore Findsen | 40 students

BME 590L-01
Advanced Topics with lab - BIOMEDICAL & CLINICAL DESIGN
George Truskey | 24 students

The picture above is a snapshot of the Piazza home page. Below, are the HTTP headers exchanged between the browser and Bing. The transactions have been edited, leaving only a few of the transactions.

```
-----------------------------------------------------------Transaction 1
```

```
http://ocsp.godaddy.com/
POST / HTTP/1.1
Host: ocsp.godaddy.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
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
Content-Length: 106
Content-Type: application/ocsp-request
Connection: keep-alive


HTTP/1.1 200 OK
Date: Fri, 22 Feb 2013 18:24:27 GMT
Server: Apache
content-transfer-encoding: Binary
Cache-Control: max-age=12905, public, no-transform, must-revalidate
Last-Modified: Fri, 22 Feb 2013 16:10:52 GMT
Expires: Fri, 22 Feb 2013 22:10:52 GMT
Etag: "2e38ac3a0ac40d876bb7ba482005e6c68ceb96a9"
p3p: CP="IDC DSP COR LAW CUR ADM DEV TAI PSA PSD IVA IVD HIS OUR SAM PUB LEG UNI COM
NAV STA"
Content-Length: 1923
Connection: close
Content-Type: application/ocsp-response
----------------------------------------------------------Transaction 2
https://www.piazza.com/

GET / HTTP/1.1
Host: www.piazza.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
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: piazza_session=EGyCDvHvKwuwCyLDEDGGtxGyFuHDEFGx.0;
__utma=231435806.1547084154.1357668061.1361132434.1361506957.15;
__utmz=231435806.1357668061.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none);
__utmv=231435806.|2=user_type=instructor=1
Connection: keep-alive

HTTP/1.1 302 Found
Server: nginx/1.2.1.1431
Date: Fri, 22 Feb 2013 18:24:27 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 86
Connection: close
Status: 302
X-Powered-By: Phusion Passenger (mod_rails/mod_rack) 3.0.7
Set-Cookie: piazza_session=EGyCDvHvKwuwCyLDEDGGtxGyFuHDEFGx.0; path=/; expires=Thu, 22
Aug 2013 18:24:27 GMT; HttpOnly
X-Runtime: 0
Location: https://piazza.com/?
Cache-Control: no-cache
----------------------------------------------------------Transaction 3
https://piazza.com/?

GET /? HTTP/1.1
Host: piazza.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
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=231435806.1547084154.1357668061.1361132434.1361506957.15;
__utmz=231435806.1357668061.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none);
last_piaz_user=gr81dafcQqZ; __utmv=231435806.|2=user_type=instructor=1
Connection: keep-alive
If-None-Match: "2549c6767dbab8e1f80868d7e90d7c02"

HTTP/1.1 200 OK
Server: nginx/1.2.1.1431
Date: Fri, 22 Feb 2013 18:24:27 GMT
Content-Type: text/html; charset=utf-8
Transfer-Encoding: chunked
Connection: close
Vary: Accept-Encoding
Status: 200
X-Powered-By: Phusion Passenger (mod_rails/mod_rack) 3.0.7
Set-Cookie:
piazza_session=HuyLKtGCDIIFLxvutxKxttyGJLvttLvy.5%26x%26%26%7C%22%21r%7Cw5M5Trab_rf%2C
V%2CV%5E5%3F5%275MD0; path=/; expires=Thu, 22 Aug 2013 18:24:27 GMT; HttpOnly
X-Runtime: 96
p3p: CP='IDC DSP CURa ADMa DEVa OUR ONL UNI COM NAV'
Etag: "5f45fe0637dfbc9241d25b51926dd804"
Cache-Control: private, max-age=0, must-revalidate
Content-Encoding: gzip
----------------------------------------------------------Transaction 4
https://piazza.com/css/splash/style.css?v=5202

GET /css/splash/style.css?v=5202 HTTP/1.1
Host: piazza.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
Accept: text/css,*/*;q=0.1
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: __utma=231435806.1547084154.1357668061.1361132434.1361506957.15;
__utmz=231435806.1357668061.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none);
last_piaz_user=gr81dafcQqZ; __utmv=231435806.|2=user_type=instructor=1;
piazza_session=HuyLKtGCDIIFLxvutxKxttyGJLvttLvy.5%26x%26%26%7C%22%21r%7Cw5M5Trab_rf%2C
V%2CV%5E5%3F5%275MD0
Connection: keep-alive

HTTP/1.1 200 OK
Server: nginx/1.2.1.1431
Date: Fri, 22 Feb 2013 18:24:28 GMT
Content-Type: text/css
Transfer-Encoding: chunked
Connection: close
Last-Modified: Tue, 19 Feb 2013 03:31:35 GMT
Vary: Accept-Encoding
Expires: Mon, 25 Mar 2013 18:24:28 GMT
Cache-Control: max-age=2678400
Content-Encoding: gzip
----------------------------------------------------------Transaction 5
https://connect.facebook.net/en_US/all.js

GET /en_US/all.js HTTP/1.1
Host: connect.facebook.net
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
If-None-Match: "094376e05ec9a60405bad2f7c3718bf0"

HTTP/1.1 304 Not Modified
```

```
Content-Type: application/x-javascript; charset=utf-8
Etag: "094376e05ec9a60405bad2f7c3718bf0"
Cache-Control: public, max-age=124
Expires: Fri, 22 Feb 2013 18:26:32 GMT
Date: Fri, 22 Feb 2013 18:24:28 GMT
Connection: keep-alive
----------------------------------------------------------Transaction 6
https://piazza.com/javascripts/sliding-panel.js?v=5202

GET /javascripts/sliding-panel.js?v=5202 HTTP/1.1
Host: piazza.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: __utma=231435806.1547084154.1357668061.1361132434.1361506957.15;
__utmz=231435806.1357668061.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none);
last_piaz_user=gr81dafcQqZ; __utmv=231435806.|2=user_type=instructor=1;
piazza_session=HuyLKtGCDIIFLxvutxKxttyGJLvttLvy.5%26x%26%26%7C%22%21r%7Cw5M5Trab_rf%2C
V%2CV%5E5%3F5%275MD0
Connection: keep-alive

HTTP/1.1 200 OK
Server: nginx/1.2.1.1431
Date: Fri, 22 Feb 2013 18:24:28 GMT
Content-Type: application/x-javascript
Transfer-Encoding: chunked
Connection: close
Last-Modified: Thu, 29 Nov 2012 09:02:40 GMT
Vary: Accept-Encoding
Expires: Mon, 25 Mar 2013 18:24:28 GMT
Cache-Control: max-age=2678400
Content-Encoding: gzip
----------------------------------------------------------Transaction 7
https://piazza.com/js/splash/splash.js?v=5202

GET /js/splash/splash.js?v=5202 HTTP/1.1
Host: piazza.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: __utma=231435806.1547084154.1357668061.1361132434.1361506957.15;
__utmz=231435806.1357668061.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none);
last_piaz_user=gr81dafcQqZ; __utmv=231435806.|2=user_type=instructor=1;
piazza_session=HuyLKtGCDIIFLxvutxKxttyGJLvttLvy.5%26x%26%26%7C%22%21r%7Cw5M5Trab_rf%2C
V%2CV%5E5%3F5%275MD0
Connection: keep-alive

HTTP/1.1 200 OK
Server: nginx/1.2.1.1431
Date: Fri, 22 Feb 2013 18:24:28 GMT
Content-Type: application/x-javascript
Transfer-Encoding: chunked
Connection: close
Last-Modified: Thu, 17 Jan 2013 20:43:36 GMT
Vary: Accept-Encoding
Expires: Mon, 25 Mar 2013 18:24:28 GMT
Cache-Control: max-age=2678400
Content-Encoding: gzip
----------------------------------------------------------Transaction 8
https://piazza.com/piazzza_js/ajax.js?v=5202
```

```
GET /piazzza_js/ajax.js?v=5202 HTTP/1.1
Host: piazza.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: __utma=231435806.1547084154.1357668061.1361132434.1361506957.15;
__utmz=231435806.1357668061.1.1.utmcsr=(direct)|utmccn=(direct)|utmcmd=(none);
last_piaz_user=gr81dafcQqZ; __utmv=231435806.|2=user_type=instructor=1;
piazza_session=HuyLKtGCDIIFLxvutxKxttyGJLvttLvy.5%26x%26%26%7C%22%21r%7Cw5M5Trab_rf%2C
V%2CV%5E5%3F5%275MD0
Connection: keep-alive

HTTP/1.1 200 OK
Server: nginx/1.2.1.1431
Date: Fri, 22 Feb 2013 18:24:28 GMT
Content-Type: application/x-javascript
Transfer-Encoding: chunked
Connection: close
Last-Modified: Thu, 07 Feb 2013 08:37:42 GMT
Vary: Accept-Encoding
Expires: Mon, 25 Mar 2013 18:24:28 GMT
Cache-Control: max-age=2678400
Content-Encoding: gzip
----------------------------------------------------------Transaction 9
https://ssl.google-
analytics.com/__utm.gif?utmwv=5.3.9&utms=1&utmn=1800083830&utmhn=piazza.com&utme=8(2!u
ser_type)9(2!instructor)11(2!1)&utmcs=UTF-8&utmsr=1920x1080&utmvp=1006x754&utmsc=24-
bit&utmul=en-
us&utmje=1&utmfl=11.5%20r502&utmdt=Piazza%20%E2%80%93%20Ask.%20Answer.%20Explore.%20Wh
enever.&utmhid=381092738&utmr=-&utmp=%2F&utmac=UA-21514163-
1&utmcc=__utma%3D231435806.1547084154.1357668061.1361506957.1361557560.16%3B%2B__utmz%
3D231435806.1357668061.1.1.utmcsr%3D(direct)%7Cutmccn%3D(direct)%7Cutmcmd%3D(none)%3B&
utmu=q~

GET
/__utm.gif?utmwv=5.3.9&utms=1&utmn=1800083830&utmhn=piazza.com&utme=8(2!user_type)9(2!
instructor)11(2!1)&utmcs=UTF-8&utmsr=1920x1080&utmvp=1006x754&utmsc=24-bit&utmul=en-
us&utmje=1&utmfl=11.5%20r502&utmdt=Piazza%20%E2%80%93%20Ask.%20Answer.%20Explore.%20Wh
enever.&utmhid=381092738&utmr=-&utmp=%2F&utmac=UA-21514163-
1&utmcc=__utma%3D231435806.1547084154.1357668061.1361506957.1361557560.16%3B%2B__utmz%
3D231435806.1357668061.1.1.utmcsr%3D(direct)%7Cutmccn%3D(direct)%7Cutmcmd%3D(none)%3B&
utmu=q~ HTTP/1.1
Host: ssl.google-analytics.com
User-Agent: Mozilla/5.0 (Windows NT 5.1; rv:19.0) Gecko/20100101 Firefox/19.0
Accept: image/png,image/*;q=0.8,*/*;q=0.5
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive

HTTP/1.1 200 OK
Age: 87787
Cache-Control: private, no-cache, no-cache=Set-Cookie, proxy-revalidate
Content-Length: 35
Content-Type: image/gif
Date: Thu, 21 Feb 2013 18:01:53 GMT
Expires: Wed, 19 Apr 2000 11:43:00 GMT
Last-Modified: Wed, 21 Jan 2004 19:51:30 GMT
Pragma: no-cache
Server: GFE/2.0
X-Content-Type-Options: nosniff
X-Firefox-Spdy: 3
----------------------------------------------------------
```

Each of the following questions is worth 2 points. There is no partial credit allowed.

Q1: How many web servers are involved in these HTTP transactions and what are their complete names?

A1: Apache, nginx/1.2.1.1431, GFE/2.0

Q2: What is the name and version of the browser used in these transactions?

A2: Firefox version 19.0.

Q3: What kind of MIME content types are accepted by the browser?

A3: text/html, application/xhtml+xml, application/xml, text/css, */* (any MIME type)

Q4: Are any transactions returning compressed data? If so, which ones?

A4: All transactions returning the header "Content-Encoding: gzip"): transactions 3, 4, 6, 7, and 8.

Q5: Transaction 2 request includes the "Location" header. What is that used for?

A5: To "redirect" the browser to a new page at https://piazza.com/?

Q6: What data is returned by the browser in transaction 5 and why?

A6: No data is returned (304 Not Modified) because the cached file is "fresh"

Q7: In transaction 5 request the header "If-None-Match" is present. What is the header value asking to match?

A7: The Etag of the file "all.js"

Q8: Name 4 headers used in these transactions for cache control.

A8: Any 4 of "Cache-Control", "Last-Modified", "Expires", "Etag", "If-None-Match", "Date", "Vary", "Pragma".

Q9: Are any transactions using HTTP/2 or SPDY?

A9: Yes, transaction 9, given the response header “X-Firefox-Spdy: 3”

Q10: What type(s) of files are transmitted in “compressed” format?

A10: HTML (transaction 3, text/html, gzip), CSS (transaction 4, text/css, gzip), JavaScript (transaction 6, 7 and 8, application-x-javascript, gzip),

DOM QUESTIONS [20 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>
```

You will be asked to fill some missing lines. Each missing line is worth 2 points. There is no partial credit.

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

```
function loadXMLDoc(dname)
{
if (window.XMLHttpRequest)
{ xhttp= new XMLHttpRequest(); } 1.
else
{ xhttp=new ActiveXObject("Microsoft.XMLHTTP"); }
xhttp.open("GET",dname,false);
xhttp.send();
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" > 2.
</script></head>
<body><script>
xmlDoc=loadXMLDoc("books.xml");
document.write("Number of book nodes: ");
document.write(xmlDoc.getElementsByTagName('book').length);
     3.
document.write("<br>");
//Remove the first book
y=xmlDoc.getElementsByTagName("book")[0];
xmlDoc.documentElement.removeChild(y);                        4.
document.write("Number of book nodes after removing the first
book: ");
document.write(xmlDoc.getElementsByTagName('book').length); 5.
</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');

document.write("Book 1 - category:  " +
x[0].getAttribute('category'));                          6.
document.write("<br>");
document.write(“Removing category attribute of the first book”);
x[0].removeAttribute('category');                        7.
document.write("Book 1 - category:  " +
x[0].getAttribute('category'));
</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

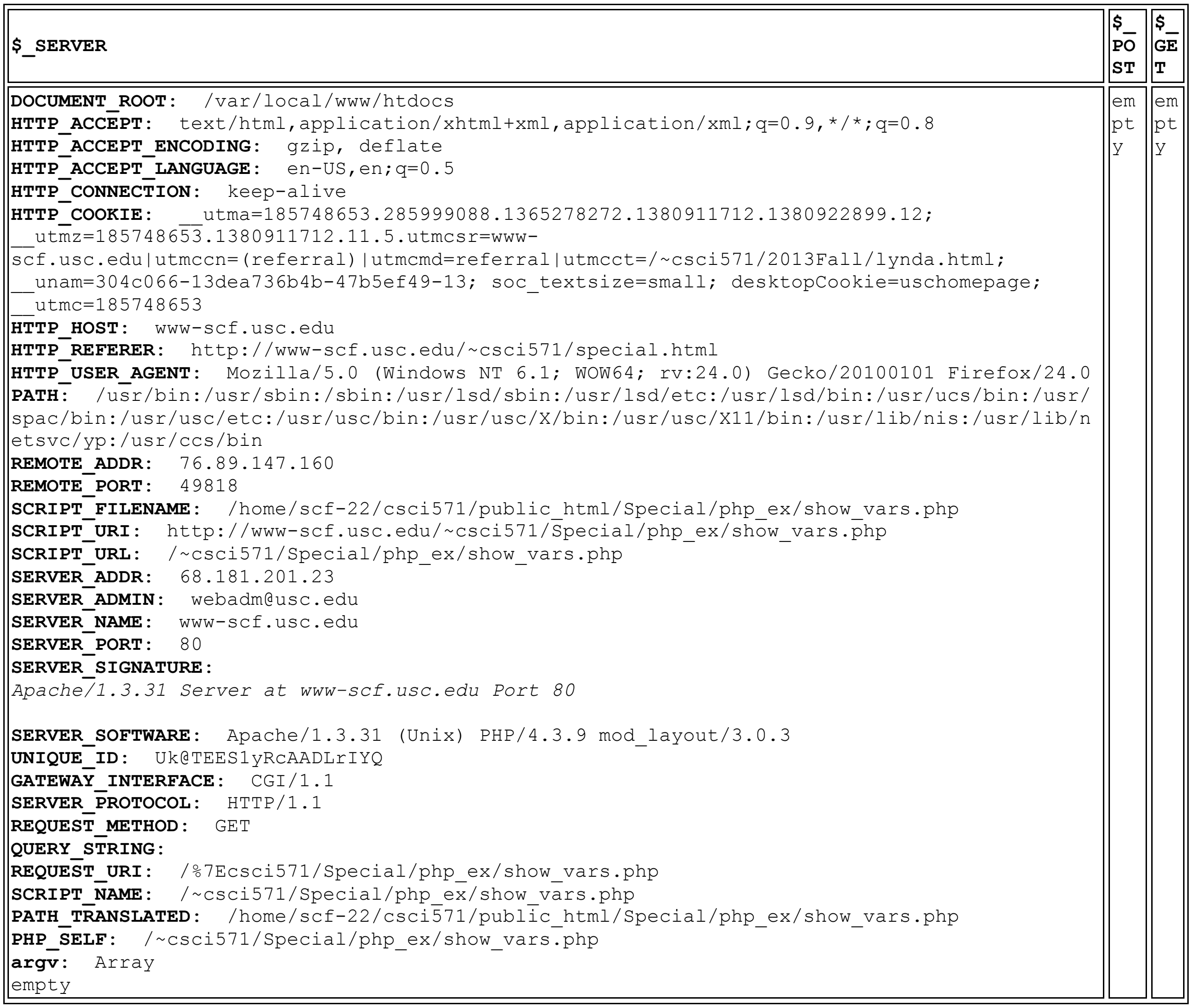
```
<!DOCTYPE html>
<html>
<body>

<p id="demo"></p>

<script>
var x, i, xmlDoc;
var txt = "";
var text = "<book>" +
"<title>Everyday Italian</title>" +
"<author>Giada De Laurentiis</author>" +
"<year>2005</year>" +
"</book>";

parser = new DOMParser();
xmlDoc = parser.parseFromString(text,"text/xml");

// documentElement always represents the root node
x = xmlDoc.documentElement.childNodes; 8.
for (i = 0; i < x.length ;i++) {
  txt += x[i].nodeName + ": " + x[i].childNodes[0].nodeValue +
"<br>"; 9.
}
document.getElementById("demo").innerHTML = txt; 10.
</script>
```

```
</body>
</html>
```

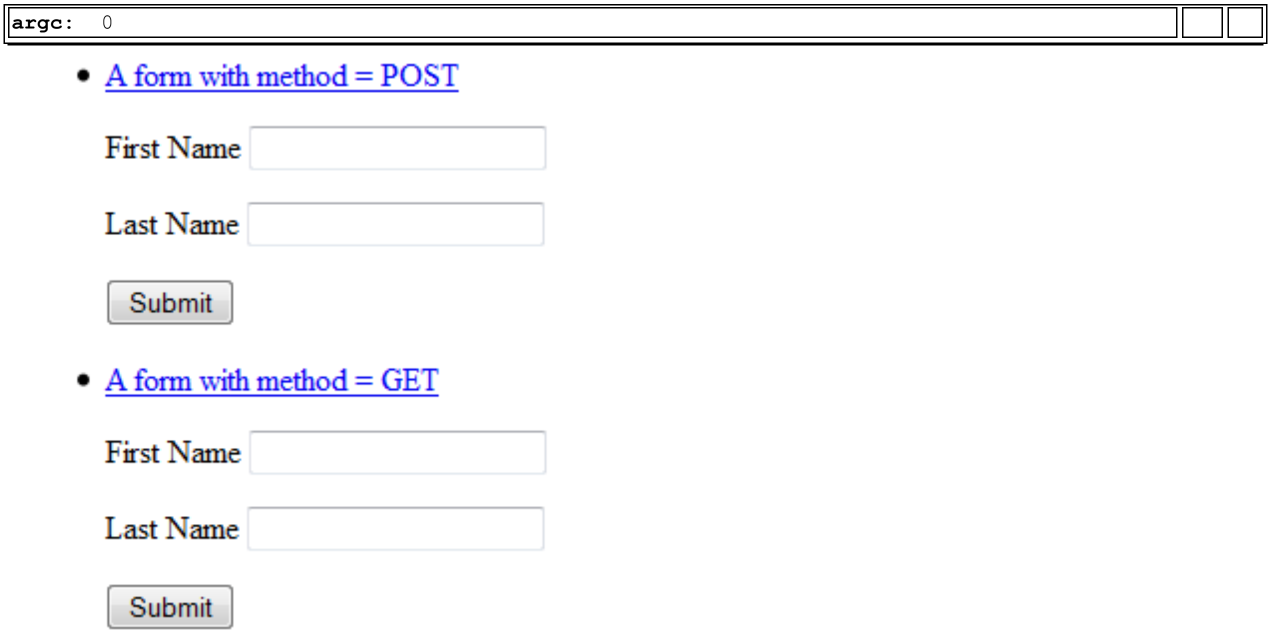
Output
title: Everyday Italian
author: Giada De Laurentiis
year: 2005

PHP QUESTIONS [10 POINTS]

[5 pts] Below is one example of the output of the program `show_vars.php`, a program that prints the PHP environment variables and the POST and GET name=value pairs. 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 **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	empty	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, $_GET
Variables</title></head>
<style type="text/css">
th {   text-align:left;}
</style>
<body>
<?php
        function print_tabs($tabs) {
                for($i = 0; $i < $tabs; $i++) {                1.
                        echo "&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;";
                }
        }
        function print_array($arr, $tabs = 0) {
                if(!empty($arr)) {
                        foreach($arr as $k=> $v) {
                                print_tabs($tabs);
                                echo $k . ":" . $v . "<br/>";   2.
                                if(is_array($v)) {
                                        print_array($v, $tabs+1);
                                }
                        }
                } else { echo "empty";          }
        }
?>
<table border='1'>
```

```
<tr><th width="33%">$_SERVER</th><th width="34%">$_POST</th><th
width="33%">$_GET</th></tr>
      <tr>
<td valign="top">  <?php print_array($_SERVER); ?>  </td> 3.           3.
<td valign="top">  <?php print_array($_POST); ?>    </td> 4.
<td valign="top">  <?php print_array($_GET); ?>     </td> 5.           5.
      </tr>  </table>
<ul>
      <li><a href="?fname=Hello&lname=World">A form with method
= POST</a>
      <div>
      <form method="POST" action="">
      <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="?fname=Hello&lname=World">A form with method
= GET</a>
      <div>
      <form method="GET" action="">
      <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>
```

[5 points] Consider the following program and complete the output that is produced.

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

<?php
```

```
$a=array("a"=>"red","b"=>"green");
array_push($a,"blue","yellow");
print_r($a);
?>

</body>
</html>
```

Output:

Array (**[a] => red [b] => green [0] => blue [1] => yellow**)

JAVASCRIPT REGULAR EXPRESSIONS QUESTIONS [10 POINTS]

Q1. [5 points] Describe in one sentence what does the pattern checks for a match:

```
var pattern = /(^-*\d+$)|(^-*\d+\.\d+$)/
```

A1. Either zero or more minus signs, followed by one or more digits, or zero or more minus signs followed by one or more digits, followed by a dot (.), and followed by one or more additional digits.

Q2. [5 points] Describe in one sentence what does the pattern checks for a match:

```
var pattern = /^[0-9]{3}[\-]?[0-9]{2}[\-]?[0-9]{4}$/;
```

A2. Anchored at the beginning and at the end the pattern matches: 3 digits, an optional hyphen, two digits, an optional hyphen, followed by 4 digits; in effect a Social Security Number.

CSS QUESTIONS [10 POINTS]

Below are lines extracted from some CSS style sheet. For each line answer the question. Each question is worth 2 points.

Q1. Give an HTML example usage of the declaration below:

```
#sh_rdiv a {position:relative;
float:left; display:block;
text-decoration:none; cursor:default; outline:none}
```

A1.

```
<div ID="sh_rdiv">
          <a href="http://www.hello.com">My link</a>
</div>
```

href value can be any URL or "local" html page name
"My link" value can be any string.

Q2. What does this declaration do?

```
.hpVpv_Tile_home_theme a,.hpVpv_TilePanel {
background-color:#0072bc; border-color:#fff; color:#fff;
font-family:Segoe UI Light,SegoeUI,Tahoma,Arial,Verdana,sans-serif}
```

A2. Changes the background-color, border color, foreground color, and font-family of a class named hpVpv_TilePanel and the A (Anchor) tag INSIDE any tag with class name hpVpv_Tile_home_theme

Q3. Give two (2) HTML examples of usage of the declaration from Q2:

A3.

```
<div class="hpVpv_TilePanel"> </div>

<div class="hpVpv_Tile_home_theme">
     <a href="http://www.hell.com">My link</a>some text
</div>
```

Q4. Write in the CSS to create a "class" named hidden **that changes the CSS properties** display **to none,** height **to 0,** width **to 0 and** overflow **to hidden.**

A4.

```
.hidden {display:none; height:0; width:0; overflow:hidden}
```

Q5. Give an HTML example usage of the declaration below:

```
span { display: block }
```

A5.

```
<span>A display property with a value of "block" results
in</span> a line break between the two elements
```