Computer Science 571 2nd Exam
Prof. Horowitz
Tuesday, December 3, 2013, 9:30am – 10:45am

Name: **Student ID Number:**

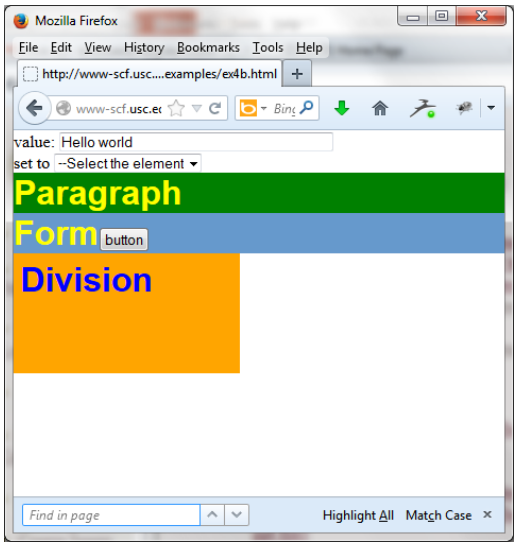
1. This is a closed book exam.
2. Please answer all questions.
3. Place all answers on the exam and return the entire exam

Section	Student Grade	Max
jQuery		**/20**
Javascript + Ajax		**/20**
PHP + Regular Expressions		**/15**
Java Servlets		**/10**
HTML5		**/15**
REST		**/20**
		/100

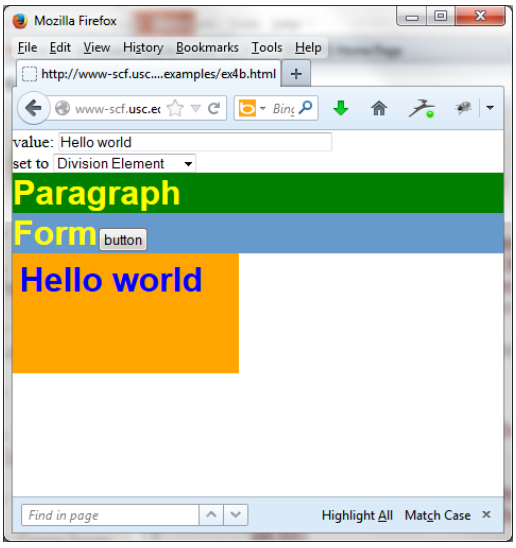
jQuery [20 pts]

1. [10 pts]

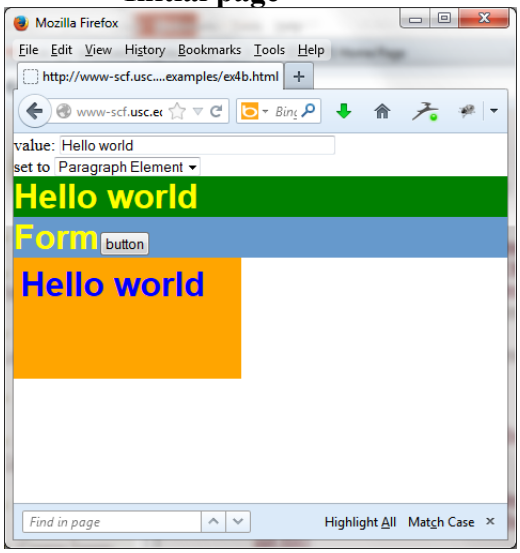
Below are 4 screenshots and following that the source code with some removed. Using jQuery supply the missing code



Initial page



Select Division Element



Select Paragraph Element

Select Form Element

```
<html><head><meta charset="UTF-8">
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"
></script>
<script>
$(function() {
```

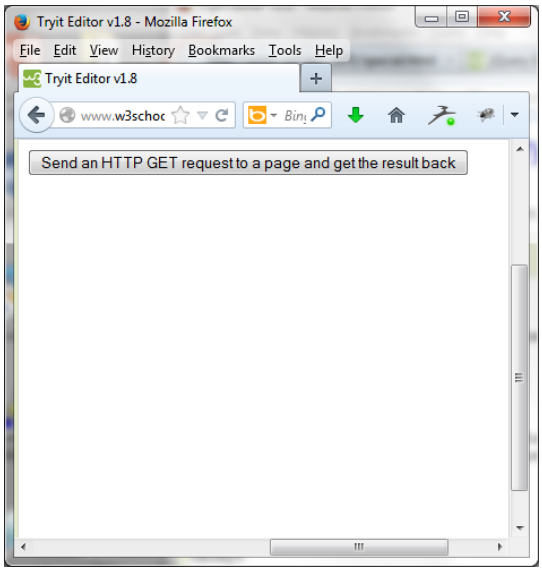
.....

.....

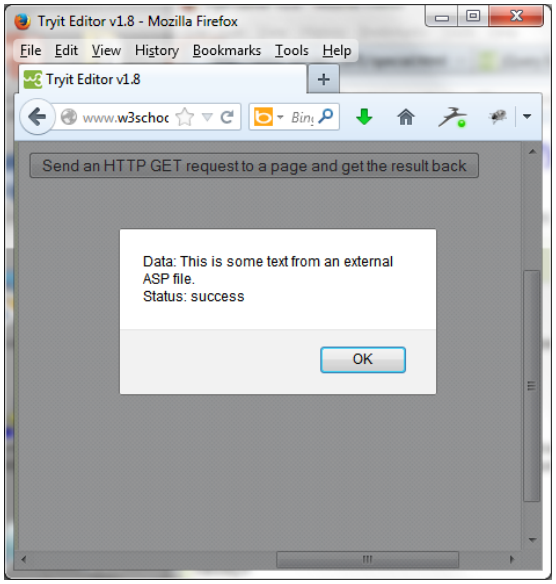
```
.....
.....
.....
.....
});
</script>
<style type="text/css">
* { padding:0px; margin:0px; }
.sample {  height:40px; min-height:40px; display:block;  width:100%; }
.smp { PADDING-RIGHT: 0.2em; PADDING-LEFT: 0.2em; PADDING-BOTTOM:
0.2em; WIDTH: 200px;PADDING-TOP: 0.2em; POSITION: absolute; height:
100px}
#t1 {FONT-WEIGHT:700; FONT-SIZE: 2em; color: blue;
     FONT-FAMILY: sans-serif; BACKGROUND-COLOR: orange}
#t2{FONT-WEIGHT:700; FONT-SIZE: 2em; LEFT: 120px;
     color: yellow; FONT-FAMILY: sans-serif; top: 200px; BACKGROUND-
COLOR: green}
#t3{FONT-WEIGHT:700; FONT-SIZE: 2em;
     color: yellow; FONT-FAMILY: sans-serif; BACKGROUND-COLOR: #6699cc}

</style></head><body>
<form>
value: <input size=40 value="Hello world" name=t><br>
set to
<select name="sel" id="sel">
<option value="" selected>--Select the element
<OPTION value="t1">Division Element<OPTION value="t2">Paragraph Element
<OPTION value="t3">Form Element</OPTION></select></form>
<p id=t2>Paragraph</p>
<form id=t3 name=t3>
Form<input type=button value=button>
</form>
<div class=smp id=t1>Division</DIV><table height=100 width=250>
 <tr><td></td></tr></tbody></table>
</body></html></html>
```

2. [10 pts]

Below are two screenshots. The first has a button and the second shows what happens once the button is clicked. The button issues a GET request and returns the result in an alert box. Below that is the code that implements the button, with some of the jQuery code removed. Fill in the missing jQuery code.



Initial screen



After the button is clicked

```
<!DOCTYPE html><html><head>
<script
src="http://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js">
</script>
<script>
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
</script>
</head><body>
<button>Send an HTTP GET request to a page and get the result
back</button>
</body></html>
```

JavaScript + Ajax [20 pts]

3. [10 pts]

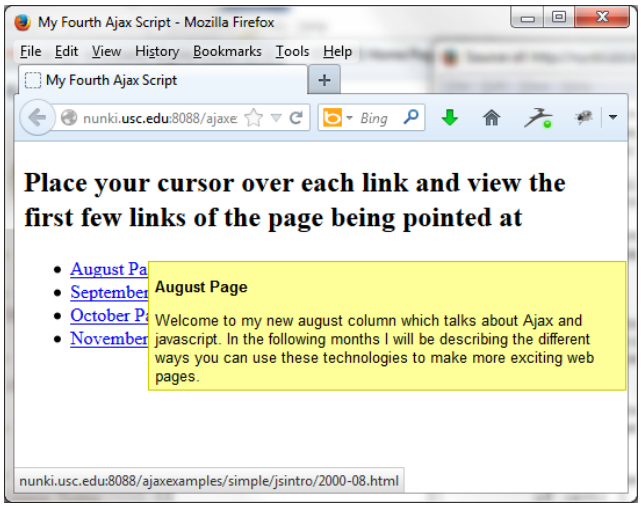
JSONscriptRequest is a simple class for making HTTP requests using dynamically generated script tags and JSON.

Below is a sample that uses the Yahoo service for returning Latitude and Longitude given a zip code. Fill in the missing code.

```
<script type="text/javascript" src="jsr_class.js"></script>

function callbackfunc(jsonData) {
    alert('Latitude = ' + jsonData.ResultSet.Result[0].Latitude +
         '  Longitude = ' + jsonData.ResultSet.Result[0].Longitude);
     aObj.removeScriptTag();
 }

request =
'http://api.local.yahoo.com/MapsService/V1/geocode?appid=YahooDemo&
            output=json&callback=callbackfunc&location=78704';
aObj = new JSONscriptRequest(request);
aObj.buildScriptTag();
aObj.addScriptTag();

function JSONscriptRequest(fullUrl) {
     this.fullUrl = fullUrl;
    this.noCacheIE = '&noCacheIE=' + (new Date()).getTime();
    this.headLoc = document.getElementsByTagName("head").item(0);
    this.scriptId = 'JscriptId' + JSONscriptRequest.scriptCounter++;
}
JSONscriptRequest.prototype.buildScriptTag = function () {
    this.scriptObj =.....;
    this.scriptObj.setAttribute("type",.....);
    this.scriptObj.setAttribute("charset", .....);
    this.scriptObj.setAttribute(....., ..... + this.noCacheIE);
    this.scriptObj.setAttribute("id", this.scriptId);
}
```

4. [10 pts]

Below is a screenshot showing 4 bullet symbols and the months August, September, October and November. When a user moves the cursor over one of the bullets, a portion of the page that is linked to appears in the popup window. Below the screenshot is the source code. Fill in the missing details.



Here is the code that creates the web page

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN">
<html>
<head>
        <title>My Fourth Ajax Script</title>
        <link rel="stylesheet" rev="stylesheet" href="script04.css" />
        <script src="script04.js" type="text/javascript"
                language="Javascript"></script>
</head>
<body>
<h2>Place your cursor over each link and view the first few links of
the page being pointed at</h2>
<ul>
        <li><a href="jsintro/2000-08.html">August Page</a></li>
        <li><a href="jsintro/2000-09.html">September Page</a></li>
        <li><a href="jsintro/2000-10.html">October Page</a></li>
        <li><a href="jsintro/2000-11.html">November Page</a></li>
</ul>
<div id="previewWin"> </div>
</body>
</html>
```

And here are the contents of script04.js. Fill in the missing code.

```
window.onload = .....;
var xhr = false;
var xPos, yPos;

function initAll() {
        var allLinks = document. .....("a");
        for (var i=0; i< allLinks.length; i++)
        {
                allLinks[i].onmouseover = .....;
        }
}

function showPreview(evt){
        getPreview(evt);
```

```
        return false;
}

function hidePreview() {
        document.getElementById("previewWin")
        .style.visibility = "……";
}

function getPreview(evt) {
        if (evt) {
                var url = evt.target;
        }
        else {
                evt = window.event;
                var url = evt.srcElement;
        }

        xPos = evt.clientX;   yPos = evt.clientY;
        if (window.XMLHttpRequest) {
                xhr = new XMLHttpRequest();
        }
        else {
                if (window.ActiveXObject)
                {
                        try {
                        xhr = new ActiveXObject("Microsoft.XMLHTTP");
                        }
                        catch (e) { }
                }
        }
        if (xhr) {
                xhr.onreadystatechange = ……;
                xhr.open("……", url, true);
                xhr.send(null);
        }
        else {
                alert("Sorry, but I couldn't create an XMLHttpRequest");
        }
}

function showContents()
{
        var prevWin = ……("previewWin");
        if (xhr.readyState == ……)
        {
        prevWin.innerHTML =
                (xhr.status == ……) ?
                xhr.responseText :
                "There was a problem with the request " + xhr.status;
        prevWin.style.top = parseInt(yPos)+2 + "px";
        prevWin.style.left = parseInt(xPos)+2 + "px";
        prevWin.style.visibility = "……";
        prevWin.onmouseout = …… ;
        }
}
```

PHP and Regular Expressions [15 pts]

5. [5 pts]
Using one or two sentences describe the purpose of the following PHP code:

```
<?php
$string = $_GET["input_string”];
if (preg_match("/^.*(?=.{8,})(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).*$/",
$string)) {
    echo "OK.";
} else {
    echo "Not OK.";
}
?>
```

Answer:

.....
.....
.....

6. [5 pts]
Determine what the following PHP code outputs:

$input_line = “last_name, first_name”;

```
preg_match("/(.*), (.*)/", $input_line, $output_array);
print_r($output_array);
```

Answer:

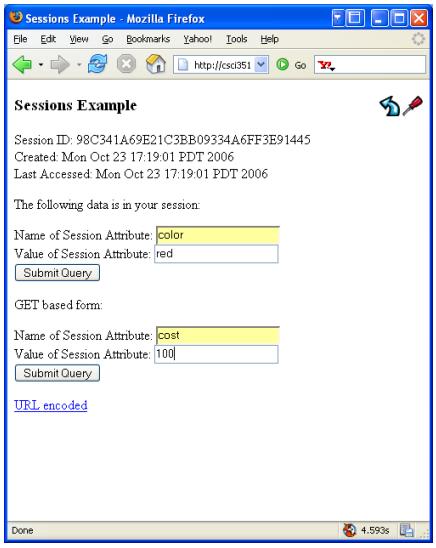
.....
.....
.....
.....
.....
.....

7. [5 pts]

The following example is designed to change the date format from "yyyy-mm-dd" to "mm/dd/yyy". Fill in the missing code

```
<?php
        echo preg_replace("/(\d+)-(\d+)-(\d+)/",
                          "...................",
                          "2007-01-25");
?>
```

Java Servlet Questions [10 pts]



8. [5 pts]

Above is a web page that is implemented by the Java Servlet below. The following Java servlet returns the web page above that indicates when a session was started and when it was last accessed and various session variables. Fill in the missing code.

```
import java.io.*;
import java.util.*;
import javax. ..........*;
import javax. ..........  ..........*;
public class SessionExample extends .......... {
    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
                      throws IOException, ServletException
    {
      response.setContentType("..........");
      PrintWriter out = response.getWriter();
      HttpSession session = request. .......... (true);

      // print session info
      Date created = new Date(session. .......... ());
      Date accessed = new Date(session. .......... ());
      out.println("ID " + session.getId());
      out.println("Created: " + created);
      out.println("Last Accessed: " + accessed);

      // set session info if needed
      String dataName = request.getParameter("dataName");
```

```
        if (dataName != null && dataName.length() > 0) {
              String dataValue = request.getParameter("dataValue");
              session.setAttribute(dataName, dataValue);
        }

        // print session contents
        Enumeration e = session.getAttributeNames();
        while (e. ..... ()) {
              String name = (String)e. ..... ();
              String value = session.getAttribute(name).toString();
              out.println(name + " = " + value);
        }//end while
      }}
```

9. [5 pts]

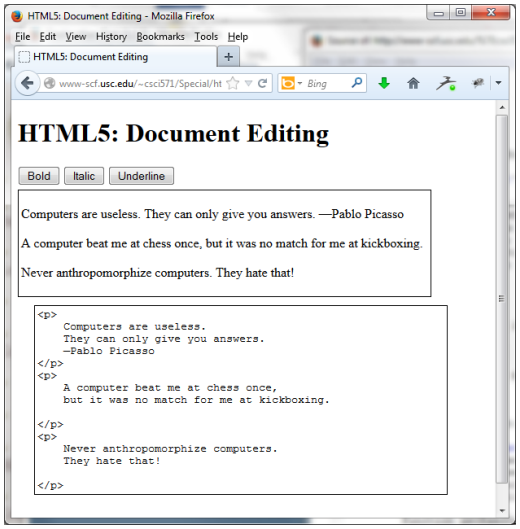
Name one possible folder of your Tomcat web server where the *.class file can be placed?

Answer:

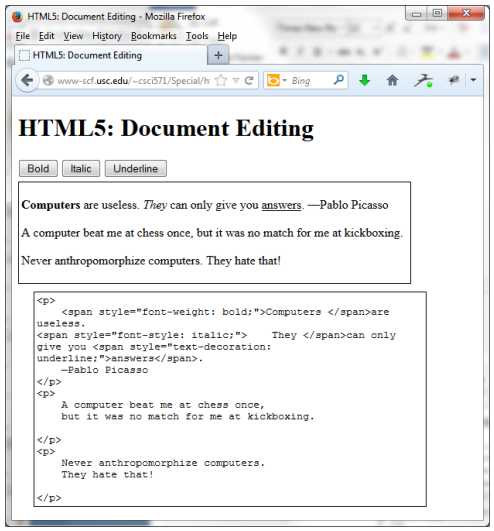
.....
.....
.....

HTML5 [15 pts]

10. [15 pts]



Initial Page



After Setting Bold, Italics and Underline

The HTML5 execCommand method on the HTMLDocument interface allows scripts to perform actions on the current selection. Generally this command is used to implement an editor in the User Interface. The HTML5 `contenteditable` attribute is an enumerated attribute whose keywords are the empty string, true, and false. The contenteditable attribute specifies whether the content of an element is editable or not. Below is the code that implements the web page above. Fill in the missing code.

```
<!DOCTYPE html><html lang="en"><head><title>HTML5: Document Editing
</title>
    <link rel="stylesheet" type="text/css" href="../CSS/main.css">
    <style>
        #left { float: left; }
        #editable, #viewHTML {
            border: solid 1px black;
            padding: 3px;
            margin: 5px 0;
            width: 500px;
            float: left;
        }
        #viewHTML {
            margin-left: 20px;
        }
        #viewPre {
            white-space: pre-wrap;
            margin: 0;
        }
        #content { height: 500px }
    </style>

<script type="application/javascript">
var haveExecCommand = false;
var loaded = false;
var editable = null;
var viewPre = null;

function getExecCommand() {
      try {
      if( !! document. ..... )
        haveExecCommand = true;
      else
        alert('This browser does not support the document editing API.');
      }
      catch(e) {
        alert('This browser does not support the document editing API.');
      }
}

function format(cmd, promptString) {
      var value = null;

      if(!loaded || ! .....)
            return true;

      if(promptString) value = prompt(..... );
            document.execCommand(cmd, false, value);
```

```
      loadView();
}

function loadView() {
      viewPre.textContent = editable. ……;
}

function init() {
      loaded = true;
      getExecCommand();
      editable = document. …… ('editable')
      viewPre  = document. …… ('viewPre');

      document.execCommand('styleWithCSS', false, true);
      editable.onblur = loadView;
      loadView();
}

window.onload = init;
</script>
</head>
<body>
<div id="content">
<h1> HTML5: Document Editing</h1>
<div id="left">
<form id="editButtons">
<input type="button" value="Bold" onClick="format(' …… ');">
<input type="button" value="Italic" onClick="format('italic');">
<input type="button" value="Underline" onClick="format('underline');">
</form>

<div id="editable" contentEditable="……"><p>
    Computers are useless.
    They can only give you answers.
    &mdash;Pablo Picasso
</p>
<p>
    A computer beat me at chess once,
    but it was no match for me at kickboxing.
</p>
<p>Never anthropomorphize computers. They hate that!</p>
</div></div>
<!-- left -->
<div id="viewHTML">
<pre id="viewPre"></pre>
</div>
</div>
<p class="clear"></p>
</body>
</html>
```

REST [20 pts]

11. [5 pts]
What does REST stand for?

Answer:

……
……
……

12. [10 pts]

The following lines of code issue a GET request and returns the entire response as one long string. Supply the missing code.

```
public static String httpGet(String urlStr) throws IOException
{
     URL url = new URL(urlStr);
    HttpURLConnection conn = (HttpURLConnection) url. …… ();

     if (conn.getResponseCode() != 200) {
       throw new IOException(conn.getResponseMessage()); }

     // Buffer the result into a string
     BufferedReader rd = new BufferedReader(
             new InputStreamReader(conn. …… ()));

     StringBuilder sb = new StringBuilder();
     String line;

     while ((line = rd. …… ()) != null) {
       sb.append(line);
     }

     rd. …… ();
     conn.disconnect();
     return sb.toString();
}
```

13. [5 pts]
Which of the two URLs below best represent the use of REST calls

a. http://www.boeing.com/airplanes/747

b. http://www.boeing.com/airplanes/747.html

Answer:

.....
.....
.....
.....

The End
Good Luck