

Computer Science 571 2nd Exam
Prof. Horowitz

Thursday, December 2, 2010, 9:30am – 10:45am

Name:

Student ID Number:

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

JavaScript Questions [45 pts]

The following JavaScript regular expression is used to validate a URL.

```
var re = /^(file|http):\/\/\S+\/\S+\. (gif|jpg) $/i;
```

1. [5 pts] Provide a sample URL that successfully matches the regular expression
2. [5 pts] What does the S+ match?
3. [5 pts] what is the purpose of the ^?
4. [5 pts] What is the purpose of the i?

Below is JavaScript code that can be used to store how many times a particular user has accessed a page. Some of the code is missing (questions 5XXXXXXX, 6XXXXXXX, 7XXXXXXX, 8XXXXXXX, and 9XXXXXXX). Please provide the missing code:

```
window.onload = initPage;
function initPage( ) {
    var expireDate = new 5XXXXXXX;
    expireDate.setMonth(expireDate.getMonth( ) + 6);
    var hitCt = 6XXXXXXX(cookieVal("pageHit"));
```

```

7xxxxxxx
document.cookie = "pageHit=" + hitCt + ";expires=" + expireDate.toGMTString();
document.8xxxxxxxinnerHTML = "You have visited this page "
    + hitCt + " times.";
}
function cookieVal(cookieName) {
    var thisCookie = document.cookie.split("; ");
    for (var i = 0; i < thisCookie.length; i++) {
        if (cookieName == thisCookie[i].9xxxxxxx("=")[0]) {
            return thisCookie[i].split("=")[1];
        }
    }
    return 0;
}

```

5. [5 pts]

6. [5 pts]

7. [5 pts]

8. [5 pts]

9. [5 pts]

Web Performance Questions [15 pts]

The lecture on High Performance Web Sites listed 14 rules for improving the speed of a website. Answer the questions that follow:

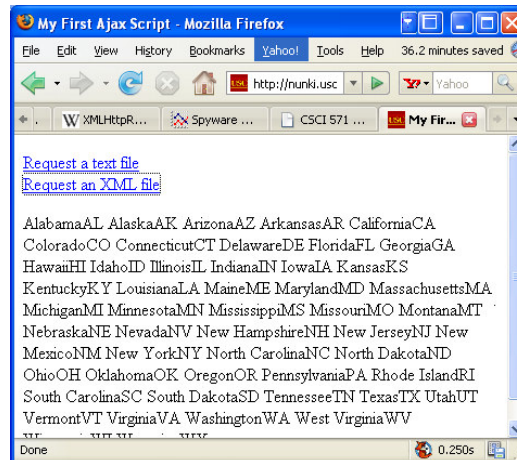
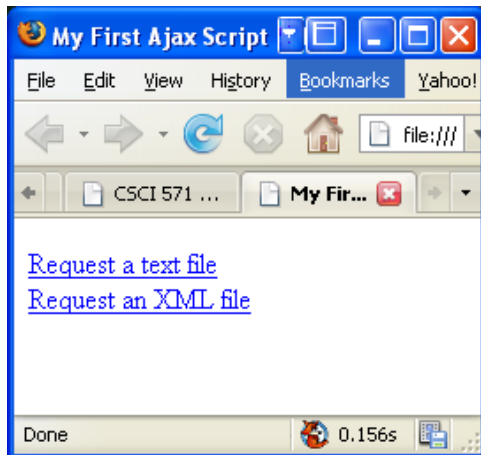
10. [5 pts] The 14 rules are incorporated in a Firefox add-on that was demonstrated in class. What is the name of the Add-on?

11. [5 pts] What was the point of moving JavaScript to the bottom of a web page?

12. [5 pts] One of the rules was to “minify”. In this context, what is meant by “minify”?

JavaScript and AJAX Questions [18 pts]

Below are two screenshots of web pages. The first screen shows an initial page, and the second screen shows the result after the second link is clicked. After the screenshots you will find a template for the code that produced the result. Fill in the missing phrases that are shown as 13XXXXXXX, 14XXXXXXX, 15XXXXXXX, 16XXXXXXX, 17XXXXXXX, 18XXXXXXX, 19XXXXXXX, 20XXXXXXX, 21XXXXXXX.



```
<!13XXXXXXX html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN">
<html><head>
  <title>My First Ajax Script</title>
  <script src="script01.js" type="text/javascript" language="Javascript">
  </script>
</head><body>
<p><a id="makeTextRequest" href="gAddress.txt">Request a text file</a><br />
<a id="makeXMLRequest" href="us-states.xml">Request an XML file</a></p>
  <div id="updateArea">&nbsp;</div>
</body>
</html>
```

Where the contents of script01.js is as follows:

```
window.onload = 14XXXXXXX;
var xhr = false;
function initAll() {
  document.getElementById("makeTextRequest").15XXXXXXX = getNewFile;
  document.getElementById("makeXMLRequest").16XXXXXXX = getNewFile;
}
function getNewFile() {
  makeRequest(17XXXXXXX); return false;
}
function makeRequest(url) {
  if (window.XMLHttpRequest) { xhr = new 18XXXXXXX(); }
  else { if (window.ActiveXObject) {
    try { xhr = new ActiveXObject("Microsoft.XMLHTTP"); }
    catch (e) { }
  }
}
```

```

    }
    }
    if (xhr) { xhr.19xxxxxxx = showContents;
        xhr.open("GET", url, true); xhr.send(null); }
    else { document.getElementById("updateArea").innerHTML = "Sorry, but I
couldn't create an XMLHttpRequest"; } }
function showContents() {
    if (xhr.readyState == 4) {
        if (xhr.status == 20xxxxxxx) {
            var outMsg = (xhr.responseXML &&
xhr.responseXML.contentType=="text/xml") ?
xhr.responseXML.getElementsByTagName("choices")[0].textContent :
xhr.responseText;
        } else { var outMsg = "There was a problem with the request " +
xhr.status; }
        document.getElementById("updateArea").innerHTML = 21xxxxxxx;
    } }

```

13. [2 pts]

14. [2 pts]

15. [2 pts]

16. [2 pts]

17. [2 pts]

18. [2 pts]

19. [2 pts]

20. [2 pts]

21. [2 pts]

XML Schema Questions [10 pts]

```

<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"

```

```

        targetNamespace="http://www.books.org"
        xmlns="http://www.books.org"
        elementFormDefault="qualified">
<xsd:element name="BookStore">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Book" maxOccurs="unbounded">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="Title" type="xsd:string"/>
            <xsd:element name="Author" type="xsd:string"/>
            <xsd:element name="Date" type="xsd:string"/>
            <xsd:element name="ISBN" type="xsd:string"/>
            <xsd:element name="Publisher" type="xsd:string"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:schema>

```

Above is an XML Schema file. Answer the following questions:

22. [3 pts] What is the default namespace?

23. [7 pts] Produce an instance file which conforms to the definitions above. Your answer does not have to refer to an actual book.

Java Servlet Questions [12 pts]

Below is a Java Servlet demonstrating the creation of a session. The servlet begins by setting the content type. It then creates the session, extracts the current time, the creation time, and saves it. It then gets and prints the session id. Fill in the missing details for questions 24XXXXXXX, 25XXXXXXX, 26XXXXXXX, 27XXXXXXX, 28XXXXXXX, and 29XXXXXXX below.

```
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class SessionExample extends HttpServlet {
    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
        throws IOException, ServletException
    {response.24XXXXXXX;
        PrintWriter out = response.getWriter();
        HttpSession session = request.25XXXXXXX;
        // print session info
        Date created = new Date(session.getCreationTime());
        Date accessed = new Date(session.getLastAccessedTime());
        out.println("ID " + session.26XXXXXXX());
        out.println("Created: " + created);
        out.println("Last Accessed: " + accessed);
        // set session info if needed
        String dataName = request.getParameter("27XXXXXXX");
        if (dataName != null && dataName.length() > 0)
            {String dataValue = request.getParameter("dataValue");
             session.setAttribute(dataName, dataValue);}
        // print session contents
        28XXXXXXX e = session.getAttributeNames();
        while (e.hasMoreElements()) {
            String name = (String)e.nextElement();
            String value = session.getAttribute(29XXXXXXX).toString();
            out.println(name + " = " + value);}}}
```

24. [2 pts]

25. [2 pts]

26. [2 pts]

27. [2 pts]

28. [2 pts]

29. [2 pts]