Computer Science 571 2nd Exam
Prof. Papa
Thursday, April 24, 2014, 5:30pm – 6:40pm

Name: **Student ID Number:**

1. **This is a closed book exam.**
2. **Please answer all questions on the test**

JSON Question [10 pts]

The REST Flickr Service includes a "getRecent" API, flickr.photos.getRecent, which returns a list of the latest public photos uploaded to flickr. It is defines as follows:

flickr.photos.getRecent

Returns a list of the latest public photos uploaded to flickr.

Authentication

This method does not require authentication.

Arguments

`api_key` (Required)
Your API application key.
`extras` (Optional)
A comma-delimited list of extra information to fetch for each returned record. Currently supported fields are: `license`, `date_upload`, `date_taken`, `owner_name`, `icon_server`, `original_format`, `last_update`, `geo`, `tags`, `machine_tags`, `o_dims`, `views`, `media`, `path_alias`, `url_sq`, `url_t`, `url_s`, `url_m`, `url_o`
`per_page` (Optional)
Number of photos to return per page. If this argument is omitted, it defaults to 100. The maximum allowed value is 500.
`page` (Optional)
The page of results to return. If this argument is omitted, it defaults to 1.

A sample XML REST call is shown below:

http://api.flickr.com/services/rest/?method=flickr.photos.getRecent&api_key=626cf9c993df85b49d193b9645fd2c0d

When the format is XML (the default), the following is an example of the data returned:

```
<?xml version="1.0" encoding="utf-8" ?>
<rsp stat="ok">
<photos page="1" pages="100" perpage="2" total="200">
     <photo id="4144809437" owner="9755447@N04"
secret="5d5e5dc80e" server="2582" farm="3" title="Big
Springs Monument Talladega" ispublic="1" isfriend="0"
isfamily="0" />
     <photo id="4144809495" owner="40432260@N04"
secret="2f3880845a" server="2689" farm="3"
title="IMG_1247" ispublic="1" isfriend="0"
isfamily="0" />

</photos>
</rsp>
```

When the "format=JSON", a JSONP response is returned. A sample JSON REST call is shown below:

A sample XML REST call is shown below:
http://api.flickr.com/services/rest/?method=flickr.photos.getRecent&api_key=626cf9c993df85b49d193b9645fd2c0d&format=json

Please fill in the missing JSON code that duplicates the XML result above:

```
jsonFlickrApi({"photos":{"page":1, "pages":100,
"perpage":2, "total":200,

"photo":[

{"id":"4144809437", "owner":"9755447@N04",
"secret":"5d5e5c80e", "server":"2582", "farm":3,
"title":"Big Spring Monument Talladega", "ispublic":1,
"isfriend":0, "isfamily":0},

{"id":"4144809495", "owner":"40432260@N04",
"secret":"2f3880845a", "server":"2689", "farm":3,
"title":"IMG_1247", "ispublic":1, "isfriend":0,
"isfamily":0}

]
```

```
}, "stat":"ok"})
```

XML Schema Questions [10 pts]

Rewrite the definition of the ISBN element, creating a new type, "ISBNType2007", using the new format of ISBN numbers, adopted on January 1, 2007, and consisting of a sequence of 13 digits, with dashes between fields separating EAN, Group, Publisher, Title and Checksum. Valid sequences are 3,1,3,5,1 digits, as in "978-0-470-10554-2" and 3,2,4,3,1 digits, as in "978-04-7010-554-2". Fill in the missing lines below.

```
<xsd:simpleType name="ISBNType2007">
   <xsd:restriction base="xsd:string">
      <xsd:pattern value="\d{3}-\d{1}-\d{3}-\d{5}-\d{1}"/>
      <xsd:pattern value="\d{3}-\d{2}-\d{4}-\d{3}-\d{1}"/>
  </xsd:restriction>
</xsd:simpleType>
```

Java Servlet Questions [10 pts]

Below is the Java Servlet used in the reference implementation for Homework #8, with line numbers preceding the actual code. Please answer the questions following the code.

```
 1 import java.io.BufferedReader;
 2 import java.io.IOException;
 3 import java.io.InputStreamReader;
 4 import java.io.PrintWriter;
 5 import java.net.URL;
 6 import java.net.URLEncoder;
 7
 8 import javax.servlet.ServletException;
 9 import javax.servlet.http.HttpServlet;
10 import javax.servlet.http.HttpServletRequest;
11 import javax.servlet.http.HttpServletResponse;
12
13 import org.json.JSONObject;
14 import org.json.XML;
15
16 /**
17  * Servlet implementation class SearchServlet
18  */
19 public class SearchServlet extends HttpServlet {
20     private static final long serialVersionUID = 1L;
21
22     protected void doGet(HttpServletRequest request,
23             HttpServletResponse response) throws
ServletException, IOException {
24
25          response.setContentType("XXXXXXXXXX");
```

```
26
27          String companyName = request.getParameter("compName");
28
29          if (companyName == null) {
30              return;
31          }
32          companyName = companyName.trim();
33
34          if (companyName.length() == 0) {
35              return;
36          }
37          PrintWriter out = response.getWriter();
38           //out.print("{result : {}}");
39
40          try {
41              URL url = new URL(
42                      "http://default-environment-
qiibrmf8pn.elasticbeanstalk.com/?compName="
43                              + URLEncoder.encode(companyName,
"UTF-8"));
44

45              BufferedReader bufferedReader = new
BufferedReader(
46                      new InputStreamReader(url.openStream(),
"UTF-8"));
47              String line;
48              String xmlContent = "";
49              while ((line = bufferedReader.readLine()) != null)
{
50                  xmlContent += line;
51              }
52              JSONObject jsonObject =
XML.toJSONObject(xmlContent);
53              out.print(jsonObject);
54          } catch (Exception e) {
55              out.print(e.getMessage());
56          }
57      }
58
59      protected void doPost(HttpServletRequest request,
60              HttpServletResponse response) throws
ServletException, IOException {
61          doGet(request, response);
62      }
63 }
```

Q1: If the servlet is hosted on a Tomcat server at csc-server.usc.edu:37309, what would be a sample REST call?

A1: http://cs-server.usc.edu:37309/examples/servlet/SearchServlet?compName=GOOG

(Notice the capital M in compName, and the capital "s" in SearchServlet, no partial credit)

Q2: Where is the method toJSONObject() at line 52 implemented?

A2: In class XML of org.json.JSONObject JAR at line 14

Q3: What should be the Content Type set at line 25"

A3: application/json

Q4: What exceptions are caught by line 54?

A4: all Exceptions

Q5: What is the purpose of line 53?

A5: return the JSON data with stock information to the AJAX call in the JavaScript

Tomcat Questions [10 pts]

[2 pts] Define the acronym JAR and in one sentence give its definition:

Answer: Java Archive. A jar file is a standard way of packaging a collection of Java files into a single file for faster download.

[2 pts] What is a major difference between Tomcat and Apache?

Answer: Tomcat supports the execution of Java Servlets, whereas Apache requires additional modules to execute Java servlets.

[2 pts] In what folder of Tomcat does one place Java servlets that are part of a user-developed application?

Answer: WEB-INF/classes

[3 pts] What is the name of the deployment descriptor file that must be modified to deploy a Java servlet?

Answer: web.xml

[3 pts] Below are 4 possible addresses of Tomcat servers for this class, but only one is correct. Choose the correct one.

a. http://csci571.usc.edu:9980/index.jsp
b. http://csci571.usc.edu:80/index.jsp
c. http://www-scf.usc.edu/~csci571/index.jsp
d. http://www-scf.usc.edu/tomcat/index.jsp

Answer: a

Web Performance Questions [10 pts]

List the 5 rules out of the 14 rules for faster Web pages from Steve Souders that help speed up delivery of HTML (3 rules) and improve caching using HTTP headers (2 rules).

HTML:

Gzip components

Move scripts to the bottom

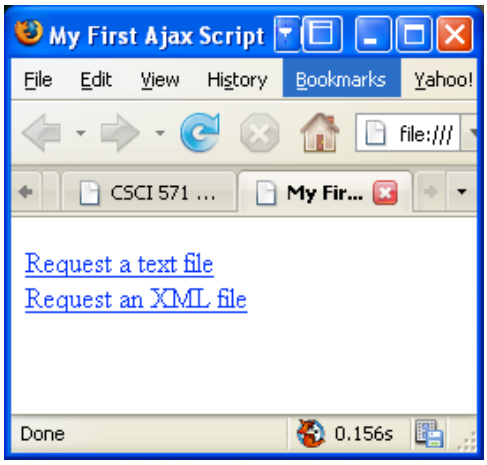
Make CSS external

Caching:

Add an Expiry header

Configure Etags

JavaScript + AJAX Questions [10 pts]



Below is the source code that generated the web page above. There are two links on the page. The first one causes a text file to be displayed in the page beneath the links. The second link causes an XML file to be displayed in the same place.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN">
<html><head>
      <title>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"> </div>
</body>
</html>
```

Below is the JavaScript source that was imported into the HTML above, but some of the lines are missing, replaced by XXXXXXXs. Fill in the missing lines on the answer sheet.

```
window.onload = initAll;
var xhr = false;
function initAll() {
document.getElementById("makeTextRequest").onclick = getNewFile;
document.getElementById("makeXMLRequest").onclick = getNewFile;}
function getNewFile() {
      makeRequest(this.href);
      return false;}

function makeRequest(url) {
      if (window.XMLHttpRequest) {
            xhr = new XMLHttpRequest();}
      else { if (window.ActiveXObject) {
            try { xhr = new ActiveXObject("Microsoft.XMLHTTP"); }
                  catch (e) { }
            }   }
      if (xhr) {
            xhr.onreadystatechange = 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 == 200) {
                  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 = outMsg;
} }
```

HTML5 Questions [10 pts]

Each question is worth 2 points.

Q1: What are Quicktime (.MOV) and Flash (.FLV) files?

A1: video "containers"

Q2: Which of the following are new in HTML5?

[ ] video and audio support
[ ] graphics support
[ ] local storage
[ ] SQL support
[ ] Geocoding support
[ ] Semantic elements
[ ] CSS3 support
[X] ALL OF THE ABOVE

Q3: What happened to the HTML 4.01 elements and <font> in HTML5?

A3: They have been moved to CSS

Q4: Name 4 Audio Codecs

A4: Any 4 from MP3, AAC, AAC+, VORBIS, FLAC

Q5: If you needed to make sure that your video files could be viewed on the large majority of browsers, what two video containers would you pick?

A5: MPEG4 and OGG

Web Security Questions [10 pts]

Each question is worth 2 points.

Q1: What does the TOR network provide?

A1: A protective layer between the user and the Internet, which 1) encrypts all information and 2) makes the user anonymous.

Q2: What do PGP and S/MIME provide?

A2: Data encryption technology for encrypting and signing e-mails

Q3: What software library is vulnerable to the Heartbleed Bug?

A3: OpenSSL

Q4: What type of attack is Stuxnet?

A4: a worm

Q5: What is recently the most recommended way to generate strong passwords?

A5: Using a very large number of characters

JQuery Questions [10 pts]

Q1: If you were to use JQuery make your code independent of Browser differences, which set of functions would you pick?

A1: JQuery AJAX Functions

Q2: Name 2 categories of JQuery selectors

A2: Any 2 of Attribute, Basic, Basic Filter, Child Filter, Content Filter, Form

Q3: [This question is worth 6 points] Consider the following example without JQuery:

```
<input id='countTags' type='button' onlick='handleAllTags()'>

function handleAllTags()
  { var arrayOfDocFonts;
    if (document.all || document.getElementById)
     { arrayOfDocFonts = document.getElementsByTagName("font"); }
    else { document.write("Unrecognized Browser Detected"); }
   alert("Number of font tags in this document are " +
arrayOfDocFonts.length + ".");
  }
```

A3: Rewrite it using JQuery.

```
$(function() { // when document is ready
      // when countTags is clicked,
      $("#countTags").click(function() {
            // alert the number of font tags in the HTML
            alert("Number of font tags in this document are " +
$("font").length);

      });
});
```

Cookies and Privacy Questions [10 pts]

Q1: What is displayed by the alert in the following program?

```
document.cookie = "test1=Hello";
document.cookie = "test2=World";

var myCookie =
document.cookie.replace(/(?:(?:^|.*;\s*)test2\s*\=\s*([^;]*).*$)|^.*$/,
"$1");

alert(myCookie);
```

A1: World

Q2: Define 3 types of cookies.

A2: Any 3 of session, persistent, third party, secure, conversion-tracking, server-side, client-side,