CS 328 - Week 15 Lab Exercise - 2025-05-08/09

Deadline

Due by the end of lab.

Purpose

To practice lightly with XML and JSON, and to try out two example PHP packages using data in these formats.

How to submit

Submit your files using **`~st10/328submit`** on nrs-projects, each time entering a lab number of **95**.

Requirements

- You are required to work in **pairs** for this lab exercise.
  - This means **two** people working at **ONE** computer, one typing ("driving"), one saying what to type ("navigating"),

    while **BOTH** are looking at the **shared** computer screen and **discussing** concepts/issues along the way.
- Make sure **BOTH** of your names appear in each file submitted.
- When you are done, before you leave lab, **BOTH** of you should submit appropriate versions of these files using **`~st10/328submit`** on nrs-projects, with a lab number of **95**.

Problem 1

Consider the posted XML examples.

Create an **XML document** in a file **`328lab15-1.xml`**. The contents must be **different** from **any** of the in-class examples -- you decide the theme of your document, and also meet the following requirements:

- it should be **well-formed** XML
- it should include your names in it somewhere
  - These can be within a comment, an element's attribute, or an element's content.
- include at least three elements with **simple content**
- include at least one element with **element content**
- include at least one element with **mixed content**
- include at least one element with **empty content**
- include at least one element with an **attribute**

If you open your **`328lab15-1.xml`** file in a browser, and it displays like the "runnable" versions of in-class XML examples do, it is likely to be well-formed XML.

Submit your file **`328lab15-1.xml`**.

Problem 2

Consider the posted JSON examples.

Create syntactically-correct JSON in a file **`328lab15-2.json`**. The contents must be **different** from **any** of the in-class examples -- you decide the theme of your document, and also meet the following requirements:

- it should be **syntactically-correct** JSON
- DO NOT INCLUDE COMMENTS IN THIS FILE! JSON does not look kindly on comments(!!)
  - To include your names in this file, then, include a data field -- perhaps `"authors"` or `"createdBy"`, for example -- whose value is a string containing your names.
- include at least five data fields with appropriate values of your choice.

Submit your file **`328lab15-2.json`**.

Problem 3

PHP's SimpleXML package

- Fun PHP fact: PHP has a number of XML-related packages for dealing with XML, including one named **SimpleXML**. (There's way more about this than you need to know at http://php.net/simplexml .)
- Amongst the contents of this package: if a file **`play.xml`** contains XML, you can obtain a PHP object version of that file's XML contents with:

  ```
  $myXml = simplexml_load_file("play.xml");
  ```

  - You can now obtain the contents of an element **`elem`** within **`play.xml`** with the expression:

    ```
    $myXml->elem
    ```

  - And if you'd like leading and trailing blanks removed from that element content, you can use PHP's **`trim`** function:

    ```
    trim($myXml->elem)
    ```

Your Problem 3 task

- Use the course HTML template to start up a PHP document **`328lab15.php`**:

  ```
  cp ~st10/html-template.html 328lab15.php
  ```

- Fill in the opening comment block, putting in your **names**, the last modified **date**, and the **URL** that can be used to run your document.
  - (You *will* lose some credit if this URL does not work when I or the grader paste it into a browser!)
- Use **SimpleXML**'s **`simplexml_load_file`** to create a PHP object from your XML file **`328lab15-1.xml`**.
- From this PHP object, get the content from **at least three** of your XML elements that happened to have simple content
  - OPTIONALLY, you may do more than this, but you may need to consult the **SimpleXML** documentation if you decide to do so.
- Display this content in the resulting document in a way that is:

  - readable
  - includes **both** the **name** of the element, and then its **content** from that PHP object

(You will be adding to this **`328lab15.php`** in Problem 4.)

Problem 4

PHP's JSON extension

- Fun PHP fact: PHP has a JSON extension. (There's more about this than you need to know at http://php.net/manual/en/book.json.php)
- If you have a JSON string -- either expressed literally or read from a file:

  ```
  $myJSON = '{"sound": "moo", "volume": 13}';
  ```

  or

  ```
  $myJSON = file_get_contents("desired_file.json");
  ```

  ...then you can get the value encoded in the JSON decoded into a PHP associative array using the **2-argument** version of **`json_decode`** with a second argument of **`true`**:

  ```
  $phpVersion = json_decode($myJSON, true);
  ```

  and now:

  ```
  $phpVersion['sound'] === "moo"
  $phpVersion['volume'] === 13
  ```

Your Problem 4 task

ADD to your document **`328lab15.php`** from Problem 3:

- Use PHP's **`file_get_contents`** to read your JSON string from your file **`328lab15-2.json`**.
- Use PHP's JSON extension to decode this JSON string into a corresponding PHP associative array.
- From this result, get the value of **at least three** of the JSON data fields.
  - OPTIONALLY, you may do more than this, but you may need to consult the PHP JSON extension documentation if you decide to do so.
- Display these values in the resulting document in a way that is:
  - readable
  - includes **both** the **name** of the data field, and then its **value** as obtained from the JSON

Submit your file **`328lab15.php`**.

BEFORE you leave lab:

Make sure that you **both** have copies of the files:

- **`328lab15-1.xml`**
- **`328lab15-2.json`**
- **`328lab15.php`**

...and you BOTH submit these using **`~st10/328submit`** on nrs-projects, with a lab number of **95**.