

## CS 328 - Final Exam Review Suggestions - Spring 2016

last modified: 2016-05-04

- You are responsible for material covered in class sessions and homeworks; but, here's a quick overview of especially important material.
- Final Exam is CUMULATIVE!
  - if it was fair game for Exam 1 or Exam 2, it is fair game for the Final Exam;
  - Thus, using the posted review suggestions for Exam 1 and Exam 2 in your studying for the Final Exam would be a good idea. (Note that they are still available from the public course web page, under "Homeworks and Handouts".)
  - studying your Exam 1 and Exam 2 would also be wise.
  - there may indeed be similar styles of questions on the Final Exam as on those exams.
- You are permitted to bring into the exam **UP TO THREE** pieces of paper (8.5" by 11") on which you have **handwritten** whatever you wish on one or both sides. EACH piece must include your name, it must be **handwritten** by you, and it will **NOT** be returned.
  - Other than these pieces of paper, the exam is closed-note, closed-book, and closed-computer.
- Final exams are **not returned**, although they will be kept on file for at least 2 years, and you are welcome to come by my office to look over your graded exam once it has been graded.
- This will be a pencil-and-paper exam, but you will be reading and writing code, statements, and expressions in this format. There will be questions about concepts as well.
- A packet of example code will be given out along with the exam, both for reference and for use directly in some exam questions. Because of the nature of this code (some being used directly in exam questions, for example), it cannot be made available in advance -- however, it will happen to include the following:
  - an uncommented version of `html5-template.html`
  - examples of a PL/SQL stored procedure, a PL/SQL stored function, and PL/SQL exception handling
  - example HTML5 that happens to include a hyperlink, a form, a textfield, a radio button, a checkbox, a drop-down box, and a submit button
  - examples of an external CSS3 file, an external PHP file, and an external JavaScript file
  - an example HTML5 document using external CSS3, external PHP, and external JavaScript
  - an example of connecting to Oracle from PHP
  - an example of executing a SQL query, PL/SQL stored procedure, and PL/SQL stored function from PHP
  - an example using JavaScript that happens to include setting an element's event handler when the window is loaded

- an example using the jQuery library for JavaScript that happens to include setting an element's event handler when the document is able to be manipulated
- Note that the ability to read and make use of existing code is an important skill.
  - It is possible that you may have to diagnose what is wrong with provided buggy code, and how it might be fixed, and/or perhaps you could be asked to modify code.
  - You might be asked to complete incomplete code (you could be given partial code, and asked to complete or modify or debug it in some way).
- There will be at least one question focused purely on SQL.

## Intro to (client-side) JavaScript

- What is the relationship between JavaScript and Java?
- JavaScript was initially designed to add interactivity to HTML pages; while it has now expanded to being able to do much more, we are focusing on so-called client-side JavaScript in CS 328.
  - When we mention "JavaScript" in this class, then, you should assume that client-side JavaScript is intended unless explicitly specified otherwise.
- Note that you are expected to use unobtrusive-style JavaScript in this course (including on exams).
- Consider an n-tier architecture. On which "tier" is JavaScript executed (given the assumption above)? Be comfortable with how a document containing JavaScript is handled/processed.
- What are some of JavaScript's capabilities?
- How would you name an HTML5 file containing JavaScript? Where would you normally place an HTML5 file containing JavaScript on nrs-projects? What permissions does this HTML5 file need to have there? What URL would you (or an HTML5 page) then use to access that HTML5 file containing JavaScript?
  - What if we are talking about an external JavaScript? How would you name that file? Assuming that you are using unobtrusive-style JavaScript, how could you use an external JavaScript's contents within an HTML5 file? What would be the CS 328-preferred element for including it, and where should this element be placed (according to CS 328 course coding standards)?
- Should be comfortable with the JavaScript syntax and features discussed in class and used in exercises and assignments (including, but not limited to:
  - What is the CS 328-preferred way to write and use variables in JavaScript?
  - how do you write a comment within JavaScript?
  - how can you concatenate strings? do basic arithmetic?
  - how do you write a function? call a function?
  - how do you do branching, repetition?
  - what is the difference between == and ===? What is NaN? (Note that you do indeed need to use the function `isNaN` to see if something has this special value.) How can you attempt to get the

numeric equivalent of something?

- What is the meaning of the `onload` attribute of an HTML5 `body` tag? ...of the `onsubmit` attribute of the HTML5 `form` tag? ...of the `onclick` attribute of the HTML5 `button` tag? ...of the `onchange` attribute of a number of HTML5 elements? How can these be used in conjunction with JavaScript?
  - note: you are expected to use unobtrusive JavaScript style
- Consider the DOM (Document Object Model) -- what is the `document` object?
  - How can you use its `getElementById` method to obtain a reference to a JavaScript object corresponding to a particular HTML5 element object within that page? What attribute should an HTML5 element have to allow it to work with this method?
  - How can you use its `getElementsByName` method to obtain a reference to a JavaScript object corresponding to the set of all of the HTML5 elements with that tag name within that page? What type object does this method return? So how can you obtain one of the elements within this set?
  - How can you use such a corresponding JavaScript object to set the value of an attribute for an HTML5 element within that page?
  - How can you use such a corresponding JavaScript object to obtain or set the content of an HTML5 element within that page?
  - Understand how, in the head of an HTML5 page, you can have a JavaScript that can set the `window` object's `onload` attribute to an anonymous function that sets the event handlers for event-related attributes of HTML5 elements within that page.
- In the context of JavaScript, what is meant by `truthy` and `falsey`?
- You should be able to use unobtrusive-style JavaScript to set an element's event handler when the window is loaded
- You should be able to use unobtrusive-style JavaScript to validate a form, and to prevent its submission if it is not filled out "appropriately"
- You should be able to use unobtrusive-style JavaScript to set an element's attribute, and to set an element's content (make sure you know the difference!)
- How can you use JavaScript to create a new element in the current page? ...to add such a new element to the current page? ...to remove an element from the current page?

## A few words on Web Design, Usability, and Accessibility

- be familiar with the user interface design guidelines discussed in class; be familiar with the concepts from Chapter 7 in the course textbook, and from the posted "Web Design and Usability" slides included in the Week 10 Lecture 2 posted examples
- remember that the application screens should help the user tell the story of his/her task;
- what are some human factors that should be considered in interface design?

- ...and some general characteristics of users that should be kept in mind?
- what are some rules of thumb for visual design of web applications? ...for organization? ...for page layout? ...for navigation and links? ...for forms and controls?
- what are some guidelines for making accessible web applications?
- what are some common mistakes in interface design? some common web usability problems? some common content usability problems? some common link usability problems? some common feature usability problems?
  - (hint: there are some good examples of these in the posted Week 10 Lecture 2 "Web Design and Usability" slides)

## Intro to XML and JSON

- what does XML stand for? What was XML designed to do?
- what is an important aspect of XML elements?
- What can XML be used for? Why might this be beneficial?
- you are expected to be comfortable with XML syntax; you should be able to recognize a well-formed XML document, you should be able to write a well-formed XML document.
  - what is meant by root element? child element?
  - what is meant by an element having simple content? element content? empty content? mixed content? Given an XML page, you should be able to identify which elements have each of these kinds of content; you should be able to write example XML elements containing each type of content.
  - what is meant by an attribute?
  - What is necessary for an XML document to be said to be a well-formed XML document?
  - What is necessary for an XML document to be said to be a valid XML document? (two things are required, note)
- What is the purpose of DTDs and XML Schemas?
- According to the W3C XML specification, what should a program do if it is processing an XML document and it finds a syntax error?
- What are some advantages of XML? What are some disadvantages?
- What does JSON stand for? What is JSON?
- What is JSON syntax based on/very similar to?
- What does `JSON.parse(myJSONData)` do? What does `JSON.stringify(myJavaScriptObject)` do?
- What is the syntax for object literals in JavaScript? What are some differences between JSON syntax and JavaScript object literal syntax?

- What are some advantages of JSON? What are some disadvantages?

## Intro to jQuery library for JavaScript

- What is jQuery? What is the relationship between jQuery and JavaScript? Why is jQuery useful -- what does it provide?
- Be familiar with the difference between when a window is loaded, and when a document is "ready" (when it is able to be manipulated) -- which might be earlier? Why?
  - You should know how to use JavaScript to specify that event handlers be set for a page's HTML5 elements when a window is loaded -- you should know how to use jQuery to specify that event handlers be set for a page's HTML5 elements when the document is "ready" (able to be manipulated).
- We discussed how `$( )` is actually a jQuery "function in disguise", that returns what is within the parentheses as a jQuery object;
  - how can you use this to obtain a jQuery object for HTML5 element(s) of a particular type within a document?
  - how can you use this to obtain a jQuery object for the HTML5 element with a particular value of `id` attribute within a document?
  - how can you use this to obtain a jQuery object for the HTML5 element(s) with a particular value of `class` attribute within a document?
- NOTE: remember that it is a jQuery *convention* (<-- that is, style, not syntax) that, if you set a JavaScript variable to a jQuery object, you start that JavaScript variable's name with `$`:  

```
var $intro = $("#intro");
```

  - It can help the programmer/reader to distinguish between JavaScript variables that reference jQuery objects (and JavaScript variables that do not);
  - ...and this is important why? Because jQuery objects corresponding to DOM elements do not have the same methods and data fields as, say, the corresponding JavaScript objects for those same DOM elements.
  - This naming convention helps programmer and reader to perhaps keep these straight, both for understanding and for perhaps calling the appropriate methods and data fields for objects!
  - You are expected to follow this naming convention in CS 328 (including on exams...)

## Intro to Ajax

- what does Ajax stand for?
- which technologies are involved in Ajax? what is required of a browser for Ajax to work? is anything special required of a web server for Ajax to work?
- should be comfortable with the basic approach here, and how it differs from the standard web server-client browser behavior

- why might one choose to use Ajax? why might one choose to not use Ajax?
- what is (or can be) asynchronous about Ajax? What are the potential benefits to this?
- Using Ajax, what JavaScript object is used to request data from a web server? How do you make a request using this object? (If the request is asynchronous), how can you know when a response to a request has been received?

## Miscellany

- Should be able to discuss tradeoffs of choosing different approaches discussed this semester, and what kinds of considerations would arise with these different approaches.
  - what actions, validations, application logic, etc. can/should be done on the client tier? on one of the application tiers? on the data tier?  
...which should be done on MULTIPLE tiers? (to improve usability or the user experience BUT also help guard against cross-site scripting or SQL injection, for example)
  - using "plain" HTML5 pages vs. using dynamic pages generated by PHP
  - using a single PHP file for an application vs. using multiple PHP files for the same application
  - using PL/SQL stored procedures and functions in conjunction with the preceding approaches
  - using PL/SQL stored procedures/stored functions vs. using "plain" SQL statements
  - (you've only used PHP on the application tier in CS 328, but you know there are other options, also -- Java servlets, Java JSP pages, Python, \*server\*-side JavaScript, and more! So, hopefully you will be in a reasonable position to experiment with and learn some of these other application tier options at your leisure.)
  - etc.!