

CS 318 - Exam 1 Review Suggestions - Spring 2013

last modified: 2-26-13

- You are responsible for material covered in class sessions and homeworks; but, here's a quick overview of especially important material.
- You are permitted to bring into the exam a single piece of paper (8.5" by 11") on which you have **handwritten** whatever you wish on one or both sides. This paper must include your name, it must be handwritten by you, and it will **not** be returned.
 - Other than this piece of paper, the exam is closed-note, closed-book, and closed-computer.
- 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.
- Note that the ability to read and make use of existing code is an important skill.
 - Some code may be included along with the exam, both for reference and for use directly in some exam questions.
 - 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).
- Your studying should include careful study of posted examples and notes as well as the homeworks (and posted homework example solutions) thus far.
- Note: you are NOT responsible for Java on Exam 1 (although you are responsible for knowing it is not the same as JavaScript...!). You are responsible for the material covered through and including JavaScript, up to and including Homework 4.

N-Tier Architecture

- What are the traditional components of an interactive database application?
- What do we mean by an n-tier architecture?
- What is a 1-tier architecture? a 2-tier architecture? an n-tier/3+-tier architecture?
 - what are some of the potential advantages of a 1-tier architecture over a 2-tier architecture? ...of an n-tier architecture over a 2-tier architecture?
 - on which tier are those interactive database application components typically placed in a 2-tier architecture? ...in an n-tier architecture?
 - what are the tiers in an n-tier architecture?

SQL

- Consider an n-tier architecture. On which "tier" is SQL executed?

- Note that you will be asked to write SQL queries on this exam, as they are very important in database applications. Your skills (and comfort) in writing them should be increasing during the course of this semester.
- You should be comfortable with the `nvl` function -- what it does, where it can be used, and why/when you might use it.
- You should be able to read a SQL query and, given example tables, determine what it would do; you should be able to modify and/or debug a SQL query.

PL/SQL

- You are expected to be able to distinguish between SQL statements, PL/SQL statements that are not also SQL statements, and SQL*Plus statements.
- Consider an n-tier architecture. On which "tier" is PL/SQL executed?
- What does PL/SQL need to "add" to SQL, so that procedural programming will be possible?
- You are expected to be comfortable reading and writing PL/SQL stored procedures and stored functions.
 - What are the differences between a PL/SQL stored procedure and a PL/SQL stored function?
 - What are the primary goals/purpose of these?
 - What is the syntax for each?
- How must you conclude a PL/SQL stored procedure or function so that it will be compiled?
 - What SQL*Plus command should you enter to see the compilation errors for a PL/SQL stored procedure or function?
 - What SQL*Plus command should you enter to be able to see `dbms_output.put_line` output?
- Executing PL/SQL code
 - How do you call a PL/SQL stored procedure from within SQL*Plus? How do you call a PL/SQL stored procedure from within another PL/SQL stored procedure or function?
 - How do you call a PL/SQL stored function from within SQL*Plus? How do you call a PL/SQL stored function from within another PL/SQL stored procedure or function?
- You are responsible for those PL/SQL features that have been discussed in class, as well as for those PL/SQL features that have been used in posted course examples and in homeworks.
- Given PL/SQL, example tables, and example calls, could you tell what would happen? Given error messages or errant behavior, could you debug PL/SQL and SQL?
- What kind of parameter is a PL/SQL parameter, by default? What other two kinds of PL/SQL parameters have we discussed? How do you indicate that a parameter is one of those other kinds of parameters?
 - What are the restrictions (if any) on each kind of PL/SQL parameter?

- How can you call a PL/SQL stored procedure or stored function that expects parameters?

PL/SQL Exception Handling

- How can you handle exceptions within your PL/SQL code?
- You should be comfortable with the syntax and semantics of PL/SQL exception handling; you should be able to read and write PL/SQL code that includes exception-handling.
- What is a pre-defined exception? What are some of the common Oracle pre-defined exceptions, and when are they raised?
- How can you raise an exception? How can you define your own exceptions (a user-defined exception)?
- You should be able to write PL/SQL code that performs desired actions when a particular exception occurs.

Intro to HTML5

- What does HTML stand for? What is HTML5?
- Consider an n-tier architecture. On which "tier" is HTML executed?
- Note that you are expected to write "strict"-style HTML5 (on exam questions as well as in homework/lab submissions).
- You are responsible for those HTML5 features that have been discussed in class, as well as those HTML5 features that have been used in posted course examples and in homeworks.
 - That includes (but is not limited to) basic HTML5 document structure, title, paragraphs, headings, numbered and unnumbered lists, images, hypertext links, tables, forms, text boxes, submit buttons, reset buttons, radio buttons, checkboxes, textareas, and drop-down boxes.
 - What is the `id` attribute? What is the particular rule with regard to its value?
 - You should also be familiar with the basic rules of "strict"-style HTML5 syntax (for example: how all tags with content must be closed, the way that contentless tags must be written, how attribute values must be written, the case rules, how tags must be properly nested, etc.)
 - You should be familiar with HTML5 terminology (for example, attribute, tag, content, etc.) You should be comfortable with the differences between HTML5 tags, attributes within a tag, and values of attributes within a tag.
 - an uncommented version of the posted example `html5-template.html` will be provided on a references page along with the exam.
- What happens when an HTML5 form is submitted? If an HTML5 page has multiple forms, you should be able to tell what happens if specified actions are taken on any of the individual forms.
 - What is the difference between a form submitted using the (default) `get` method and one submitted using the `post` method?
 - Given a form whose method is `get` (or is not specified, so that the default `get` method is used),

you should be able to give the URL that would result given what has been done to the form at the time that it is submitted.

- Where would you normally place an HTML5 page on nrs-projects? What permissions does the HTML5 file need to have there? What permissions do all of the directories in that file's path need to have? What URL would you then use to access that page?
 - How could you write a link to another HTML5 page in the same directory?
 - How could you insert an image stored in the same directory?

Intro to CSS3

- what does CSS stand for? What is its purpose?
- Consider an n-tier architecture. On which "tier" is CSS executed?
- what does "cascading" mean in CSS? what does "style" mean?
- what are some of the potential benefits of CSS?
- You should know **what** an external style sheet, an internal style sheet, and an in-line style sheet are, but you will **not** be asked to read or write any internal style sheets or in-line style sheets.
 - You **should** know **why** external style sheets are preferable to internal style sheets or in-line style sheets,
 - and you should expect to have to read and write external style sheets.
- How can one set up an external style sheet? What do you do to have an HTML5 page use an external style sheet? (Be sure to know the CS 318-approved HTML5 element for this.)
 - What do you do to HTML5 elements within an HTML5 page to have external CSS3 rules apply to them? (This varies based on the rule selector, of course.)
- what is the basic CSS3 rule syntax? what are its three main parts? what is the selector? a property? a value?
 - given a set of CSS3 rules, you should be able to predict how a given HTML5 page or given HTML5 elements would be displayed;
 - should be able to read, write such rules;
 - should be able to (if necessary) modify an HTML5 element to use a given rule;
 - what does a selector do/indicate? what does a property do/indicate? what does the value (in a style) do/indicate?
- You are not responsible for knowing the relative priorities of in-line, internal, and external style sheets -- but if rules within an external style sheet conflict, or if you include multiple external style sheets and any of their rules conflict, you should know which rule and/or property's value will be used.
- You should know the different kinds of selectors discussed in course sessions and used in course examples and/or assignments.

- You should know what a class selector is -- how to write a class selector, and how to give an HTML5 element that class -- and why you might want to do so.
- You should know how to write and use an attribute selector, and why you might want to do so.
- You should know how to write and use an id attribute selector, and why you might want to do so.
- What is the CSS Box Model? What does it describe? You should be familiar with this.
- You are responsible for those CSS3 features that have been discussed in lecture and in lab, as well as those CSS3 features that have been used in posted course examples and in course assignments.

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 318.
 - When we mention "JavaScript" in this class, then, you should assume that client-side JavaScript is intended unless explicitly specified otherwise.
- 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 318-preferred element for including it, and where should this element be placed?
- 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 318-preferred way to write and use variables in JavaScript? What is the scope of such JavaScript variables?
 - how do you write a comment within JavaScript?
 - how can you concatenate strings? do basic arithmetic?
 - what can you put in a page so that a browser that does not support JavaScript will at least include text warning the user of this when that page is displayed?
 - 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? 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 an HTML5 element object within that page? What attribute should an HTML5 element have to allow it to work with this method?
 - How can you obtain or set the value of an HTML5 element object so obtained?
 - How can you set the value of an attribute for such an HTML5 element object?
 - 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`?
- How can you use unobtrusive JavaScript to do simple form validation (to prevent a form from being submitted unless it is filled out "appropriately")?