Part 0: Setup

Create a new folder for Lab 2 if you have not done so already. You may choose to serve this lab from a new virtual host, though doing so is not required to receive credit.
Ensure that you have a text editor or other integrated development environment (IDE) ready. Which one you use doesn't matter, so long as you are comfortable and efficient in its use (or are willing to invest the time to learn it).
Finally, ensure that you have a browser add-on that allows you to view computed CSS values and visually browse through the box model. (ie Chrome and Chrome dev tools)

Part 1: Semantic HTML

For this part of the lab, you will be marking up a playlist of your six (6) favorite songs. Each entry must include the following information:
* Track Name
* Artist - with a link to the artists' website
* Album Name - with a link to its page (if any) on Allmusic.com
* Album Cover (Image)
* Release Date
* Genre(s) - list multiple if appropriate

Think about the semantic correctness of the elements you're using, keeping in mind that your document might not always be rendered by a browser.
To receive credit for Part 1: Name your HTML file *yourname*-lab2part1.html and save it in your lab2 folder. Create a *yourname*lab2READ.ME file, and in it explain the reasoning for the markup you chose and explain how your HTML is semantically correct. Additionally, add a screenshot of your browser with the html page and save it as *yourname*lab2part1.xxx (jpg, png, etc.) in your lab2 folder.
Part 1: 3 files - HTML, Read.me and Screenshot (jpg)

Part 2: XML

Create a new file, *yourname*-lab2part2.xml, and mark up the same content that you did in Part 1, this time in XML. Although you may choose whichever labels and attributes for your tags that make sense, take care to structure the document in a hierarchical, extensible way. Recall that XML may be valid, but must be well-formed.
To receive credit for Part 2: Place your XML file in your lab2 folder. In the comments of your Read.me file, explain the reasoning for the markup you chose and be prepared to explain how your XML is semantically correct.
Part 2: 1 file - XML

Part 3: HTML and CSS

Create a new file, *yourname*-lab2part3.css. Edit your HTML file to reference the CSS externally using the <link> tag in the <head> of your document and save this as *yourname*lab2part3.html. This is always the preferred way to add CSS to an HTML document, as it further separates the

presentation from semantic markup and promotes the reuse of CSS between documents.
Make sure that song title is in small-caps, the artist is in bold and in a 20 percent larger font, the album is italicized, the release date and genres are in a proportionally reduced font size, and that the genres appear as a list with dashes. The album cover should appear to the right of each listing, and the title of your favorite song should appear in green. There should be a border around your list as well as an entry centered at the bottom indicating the end of the list, in a smaller, italicised font. Make sure that you add a header for each column indicating the content for each.
You may **not** use tables for part 3.
To receive credit for Part 3: Place your HTML and CSS files in your lab2 folder. Using comments in your file(s), and in your Read.me file, explain the logic for the markup you have chosen and make sure your markup is well documented. Additionally, add a screenshot of your browser with the html page and save it as *yourname*lab2part3.xxx (jpg, png, etc.) in your lab2 folder.
Part 3: 3 files - HTML, CSS and Screenshot (jpg)

Part 4: XML and CSS

According to the base specification, XML documents may also refer to CSS. After the XML prologue:
<?xml-stylesheet type="text/css" href="x">
Create a new file called *yourname*lab2part4.css, and modify your XML document to refer to this new stylesheet. Edit the stylesheet such that the information should be displayed with each item on a separate line. The Song title should be first and should be slightly smaller than the Artist Name which should also be in blue. The Album name should remain in italics. The hyperlink should work, and the image should be last, not be part of the hyperlink, and must be forced to be the same size as the other images. Finally, there should be at least one line separating each entry in your list.

This time, in order to be able to display your image and hyperlink using the HTML tags, you will need to add the html namespace to your XML file by adding an attribute to the root_element tag - which is done by adding;
<root_element xmlns:html="http://www.w3.org/1999/xhtml">
to your XML file after the stylesheet directive.
Alternatively you can accomplish this without using the img element. However, you will need some other means of rendering the image on the page using CSS (hint: look into how backgrounds are displayed). Take note of what happens with the margins: one might assume that two elements vertically separated by opposing margins, the one with the greater value will apply, and the other will collapse to zero. This rendering behavior is known as “collapsing margins,” and is used to maintain vertical rhythm when elements are in the normal flow of the page. Elements outside of the normal page flow (those that are floated, positioned absolutely, etc.) will not have their margins collapsed.
To receive credit for Part 4: Show that the appropriate changes were made using your CSS. Add the two *yourname*lab2part4.css and xml files to your lab2 folder. Add a screenshot to the folder named *yourname*lab2part4.xxx (jpg, png, etc) as well. As above add any comments to your read.me file.
Part 4: 3 files - XML, CSS and Screenshot (jpg)

To submit your lab: zip (or tar) all of your files from your lab2 folder into a file named, yep you guessed it - *yourname*lab2.zip, and submit it to LMS under the assignment for Lab 2. It should include all of the files from each part including your screenshots and album cover photos and READ.ME file.

Remember to update your read.me files after each portion of the lab!