

Extreme Website Makeover: Javascrpts, CSS, and Multimedia



Annette Lamb



**alamb@eduscapes.com
<http://eduscapes.com/sessions/makeover>**

Evaluate Your Website

Before you jump in, evaluate your current website. Then, explore other websites. It's like going to the parade of homes. Select elements to incorporate into your revised website.



Overall

- Is the website purpose and sponsorship immediately clear?
- Is the “look and feel” of the entry page appropriate for the audience and purpose?
- Can users quickly get the “big picture” of what is offered by the website?
- Is the overall site appealing and inviting?
- Are the mission, goals, and objectives of the website clear?
- Does the website address the need of the users?
- Is the website a model for other web developers?
- Does this website offer resources not available elsewhere?

Content Aspects

- Does the website contain accurate, high quality content (i.e., interesting, useful resources)?
- Is the authority of the website clear?
- Is the content objective, opinions identified, and perspectives balanced?
- Are authentic resources used (i.e., reviewed materials, established sources, primary materials)?
- Is the content timely and current?
- Is the content relevant, useful, and meaningful to the intended audience?
- Is the content presented in an efficient and effective manner?
- Is the breadth of the content effective (i.e., goes beyond basic facts and data)?
- Does the depth of content match the varied audience needs (i.e., levels of pages or information beyond the entry page)?
- Does the website provide resources that go beyond the ordinary (i.e., unique, local, special)?
- Does the website contain a mix of content formats (i.e., text, graphics, photographs, video, audio) to address specific needs?
- Is the content logically organized?
- Does the website contain current, timely information?
- Is the website free from spelling, grammatical, and other typographical errors?
- Is the writing clear and appropriate for the developmental and reading level of the audience?
- Is the content presented in a variety of ways to meet individual differences (i.e., text, graphics, photographs, diagrams, audio, video)?
- Is there quality content beyond links?
- Are links annotated, well-chosen, and organized in a logical manner?

Design Aspects

- Does the site have an effective entry page (i.e., attractive, logos, slogan, colors, navigation)?
- Does the site have a standard, consistent “look and feel” throughout (i.e., standard template or standard sections)?
- Does the site make appropriate use of fonts (i.e., font type, style, size, color, ease of reading)?
- Does the site make effective use of foreground colors, background colors, and/or images?
- Is the page layout effective and visually appealing?
- Does the graphic design reflect the purpose of the site (i.e., serious, whimsical, visually interesting)?
- Is the design attractive for the intended audience?
- Is there a consistent theme or layout throughout the website?
- Does the graphic design add to (not distract from) the site?
- Are the pages consistent, well-organized and free from clutter?

Navigation Aspects

- Does the entry page have effective navigation (i.e., menu, index, limited options)?
- Is the navigation between primary to secondary levels efficient (i.e., menus, returns)?
- Is the navigation within pages effective (i.e., overviews, lists, headings, anchors, return to top)?

Are helpful information retrieval tools (i.e., site map, index, search engine) provided?
Is the navigation appropriate for the audience?

Technical and Usability Aspects

Does the website function properly?

Does the website use correct HTML coding?

Do all graphics, animations, text, and other page elements appear in their proper places?

Do the web links function properly?

Do the mail links function properly?

Is the site easily accessed without special software/plug-ins, or are useful help resources or alternatives provided?

Does the entry page load quickly?

Do secondary pages load quickly?

Can important items on the entry page be viewed without extensive scrolling?

Is the site user-friendly and easy to navigate?

Are content elements well-labeled?

Is the website accessible to all users? Use the WebXACT website to check web accessibility.

Maintenance Aspects

Is the website regularly updated and revised?

Is new content added (i.e., special events, new options) periodically?

Is new content identified (i.e., new features, programs, facilities, resources, events)?

Does the entry page reflect updates and revisions (i.e., list of "what's new", highlighted sections)?

Are web page creation and revision dates indicated?

Are up-to-date news and/or press releases (i.e., about us, special events, photographs) provided?

Content Enhancements

There are many optional features that may be included depending on the particular focus of the website. Check those that apply:

About Site (i.e., name, mission, history, virtual tour, hello from administrator, overview)

Award Information (i.e., organization, web site)

Background Information (i.e., about, mission, philosophy, history, related links)

Contact Information (i.e., webmaster, email contact, organization, officials, staff)

Current Events (i.e., employment, volunteers, news, events)

Directories, Guides, Link Lists (i.e., grade level, homework helpers, pathfinders)

Educational Materials (i.e., tutorials, courses, activities, homework, quiz)

Feedback Opportunities (i.e., suggestion form, guestbook, surveys)

Frequently Asked Questions (i.e., questions & answers, hours, operational questions)

Interaction Opportunities (i.e., email, registrations, discussion groups, expert help, polls, learning tools)

Live Resources (i.e., streaming audio, video, chat, webcam, reference assistance)

Local Information (i.e., maps, weather, government, times, events)

Policies and Procedures (i.e., rules, guidelines, checkout procedures, renewals, disclaimers)

Programs/Activities/Opportunities (i.e., book clubs, PTA meetings, art exhibit, shows)

Promotional Materials (i.e., special events and awards, activities)

Publishing and Sharing (i.e., art galleries, student projects, local productions)

Research and Benchmarking Data (i.e., rankings, test scores, survey results, annual reports)

Schedule/Planner/Calendar (i.e., times, dates, schedules, events, activities)

Special Projects & Productions (i.e., virtual field trips and museums, databases)

Testimonials (i.e., students, patrons, awards, parents, praise)

Virtual Reference or Homework Help (i.e., ask an expert, email help, discussions, chats)

Library-Specific Resources

Blogs, Communications, Electronic Books, Electronic Databases, Government and Local Resources, Online Catalog, Online Clubs and Connections, Periodicals, Program Archives, Reference Resources, Research and Information Skills, Review Sites, Services, Special Collections, Subject Guides & Pathfinders

Beyond the Basics with JavaScript

Transparent JavaScript

Create Rollovers. Create a Dreamweaver document. Create simple roll-overs using Insert>Image Objects>Rollover Image option in Dreamweaver.

Landmarks Practice. To build your own interactive page using photos and correct/incorrect answer graphics:

Open the Landmarks (landmarkpractice.htm) document using Dreamweaver.

Click at the bottom of the text.

Use Insert>Image Objects>Rollover Image to insert each of the three rollovers.

Original image is the landmark such as *arch.jpg*

Rollover image is the answer such as *incorrect.gif*

Alternate text is a description like *St. Louis Arch*

Click OK.

Preview your document.

If you have time, create your own graphics

(150x150pixels) in FireWorks to add to the activity.

Try It! Build your own interactive page using photos and correct/incorrect answer graphics.

Adapt Rollovers. You may not get exactly the results you wish using the standard Dreamweaver options. You'll need to edit the JavaScript, however you don't need to know everything about JavaScript to make a small change.

Landmarks Practice. Click the Code option. Find each instance of the word **onmouseover** in your code. Replace it with **onclick**. Click the Design option. Preview your document.

Navigation Bars. Another JavaScript that comes with Dreamweaver is the navigation bar option. You'll need to create buttons in Fireworks or use buttons you find online. If you want them to look interesting, you'll have three different versions of your buttons: mouseup, mouseover, and mousedown.

Landmarks Practice. To add a navigation bar to your Landmarks project:

Click at the top of the page.

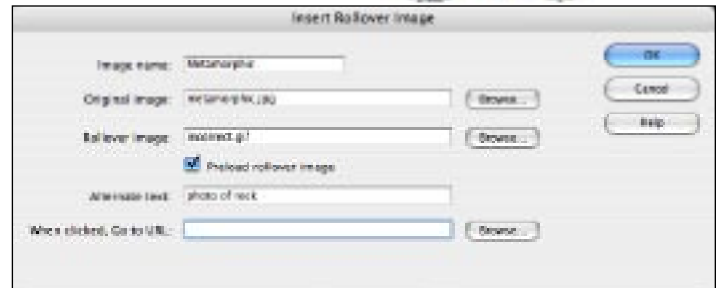
Use the Insert>Image Objects>Navigation Bar to insert a menu with three options.

Element name is *us* - Up image is *buttonus.gif* - Over image is *buttonusR.gif* - URL is *mapus.gif*

Click the + button at the top to add the *northamerica* and *world* buttons to your navigation bar.

Click OK. Use Modify>Navigation Bar to make changes.

Try It! Add a navigation bar to a web page.



JavaScript FAQs

JavaScript is an easy-to-use scripting language based on an object-oriented programming philosophy. In other words, it's designed in easy-to-copy "chunks" of code. The code is written directly into the HTML page. It is a client-side scripting language, so you don't need any special software installed on your web server for the scripts to work. Java and JavaScript are totally different languages. Java is a compiled language developed by Sun. Java is written and then packaged into a self-executing file. To create an application with a compiled language, you typically need a special programming tool or development environment.

JavaScript is great for adding interactivity to your page. Although most browsers support JavaScript, some programmers avoid them because they can cause problems with some browsers. Rather than using JavaScripts for fun, think about useful applications that really contribute to the quality of your page rather than silly uses that may interfere with page operation. For example, a rollover button is nice because users can see that their mouse is in the correct location. A question and answer button with a pop-up window is nice for practice tests.

Lingo

A **script** is a set of directions that accomplishes a task.

Objects are building blocks of the script

Objects have **properties** (like adjective) that name, indicate size, and values.

Objects have **methods** (like verbs) that do things like display text, record information.

Events are actions the user performs to make things happen.

Actions are things such as clicking a mouse that cause events.

Event handlers are commands such as onClick - user clicked an object

onLoad - objected finished loading

onMouseover - moved over an object

onMouseout - moved off an object

onSelect - user select object

onSubmit - user submitted a form

onUnload - user left the window

A **variable** is a storage container for a value and can't contain spaces or punctuation.

An **array** is a set of variables saved under the same name.

A **value** is a piece of information such as a number.

A **string** is a character or set of characters such as a word expressed with quotes like "Welcome".

Operators go with variables to make things happen.

Assignments are when you put a value on a variable such as store="Costco".

A **function** is a set of statements under one name that perform a task. They are usually in the

<head> area and are called as needed such as function testQuestion()

JavaScript Basics

Open Dreamweaver. Select File>New. Create a new document.

Click the Code button.

Enter the following language declaration meta tag in your head area above </head>.

```
<meta name="author" content="Annette Lamb" />
```

```
<meta name="keywords" content="pet, dog, puppy, cat, kitten, fish" />
```

```
<meta name="description" content="Choosing a pet." />
```

```
<meta name="copyright" content="&copy; 2006 Annette Lamb" />
```

```
<meta http-equiv="Content-Script-Type" content="text/javascript" />
```

Save your document as practice1.html in a folder called workshop.

JavaScript contains several elements. Identify scripts with the <script language="Javascript" type="text/javascript"> start tag and </script> end tag. Place lines of Javascript between the start and

end tag. Enter the javascript in the <head> of your HTML document below your language declaration.

```
<script language="Javascript" type="text/javascript">
document.write("Choosing a pet is fun!");
</script>
```

Preview your document in a browser by clicking on the Preview button



Let's add an alert box too!

```
<script language="Javascript" type="text/javascript">
document.write("Choosing a pet is fun!");
alert("Welcome to the World of Pets!");
</script>
```

Let's replace the simple alert to one that involves an if/else condition. Then, save it as practice2.html

```
if (confirm("Click OK if you love animals. ")) {
alert("Great! Let's find the perfect pet for you!")
}
else {
alert("A pet rock would be a good choice for you.")
}
```

Scripts can be placed in the <head> or <body> of your HTML document depending on the task. In the next example, the function is in the head and the form is in the body. You're probably thinking this is a lot of work. The key for beginners is re-purposing scripts rather than building your own from scratch.

Try It! To repurpose a script:

Open the Eating Right page in a web browser. View the Source code. Copy the Source code. Create a new Dreamweaver document. Select the Code view. Paste the code into Dreamweaver.

Or, download and open the (eat.html) document and select the Code view.

Change the Question, the buttons, and the replies for the pet choice example.

Change function name to petQuestion everywhere.

Change the text for each case and each alert.

External JavaScript

Sometimes you want to run the same script on many pages. In this case, you'll write a JavaScript as an external file. Then, simply call the .js file when you need it. You don't include the <script> tag in an external JavaScript. Check out examples at Landmarks JavaScript.

```
<script src="footer.js"></script>
```

Footers are a good use of an external javascript. You can create these files in Dreamweaver or a text editor. Then simply save them with the .js file extension. You can add text you wish. For an example, go to footer1.js.

```
document.write("All Rights Reserved. Copyright 2006.");
```

You can add some text and today's date. For an example, go to footer2.js.

```
var today = new Date();
document.write('Based on the Internet clock ' + today.toLocaleString() );
```

You can also add a graphic in addition to text. For an example, go to footer3.js.

```
document.write("<img src='logo.jpg'> Enjoy this student-produced page.");
```

JavaScript Ideas

Navigation. Go back one page. It's like clicking the back button on your browser. This is useful when people are coming from different places and need to go back to different places.

```
<a href="javascript:history.go(-1)">Go back </a>
```

Popups. Popups can be useful for news or instructional materials. For example, you might use a popup for definitions, help, or additional information. The problem with popups is that some people have turned off their pop-up options in their browser.

```
<head> <script type="text/javascript">
window.open('annoy.html','popup','width=250,height=250');
</script> </head>
```

Rollovers. Rollovers occur when a user rolls the cursor over an object such as a graphic and triggers an action such as a change in a graphic.

Using Forms

There are many times when you may wish users to complete forms. In most cases, you'll want a record of the information entered into the form. This information might be sent to the webmaster through email, posted in a guestbook, or recorded in a database. Teachers use forms for online homework. Librarians use forms for patron inquiries, surveys, and book reviews. Most forms involve a combination of JavaScripts and CGI scripts. Many people like to add interactive elements to their website such as surveys, guestbooks, and registration forms. Forms are used for people to enter their information.

It's easy to incorporate the form tags into your page. The tough part is getting the form to work in practice. This is because the data handling part of a form is often handled by your web server. You may or may not have control over the scripts on your web server. Before jumping into the development of forms, check with your service provider and ask about access to CGI scripts and other server-side processing tools.

You can categorize forms into four basic types:

- **checkbox:** one or more choices can be made by selecting a checkbox
- **radio button:** only one choice can be made of any radio-button group
- **drop-down selection:** one or more choices can be made from a list
- **text entry:** free data entry is allowed

Every type of form is contained within a <form> tag. You commonly use the method and the action attributes. The script looks like this:

```
<form method="post" action="mailto:alamb@eduscapes.com">The Form Goes Here</form>
```

The method of submitting or resetting the form data is often via a pre-established set of buttons indicated in the <input> tag using the type attribute:

```
<input type="submit" value="text that appears on the submit button" />
<input type="reset" value="text that appears on the reset button" />
```

Your form needs a way to make selections or enter text.

Try It! Use the Response-O-Matic Form Processor template.

Redesign with CSS



CSS stands for Cascading Style Sheets and is the easiest way to maintain consistency throughout your pages. You can use CSS two ways: internal and external. First, you can enter the style tags at the top of a single document or within a single document. These are called an internal style rules and will help you maintain consistency within a document. Second, you can create a separate document called an external style sheet that can be attached to a single document or a series of pages such as a website.

Your style sheet will go between the start head tag `<head>` and the end head tag `</head>` at the top of your document. Most people place it under the meta name tags. Start with the style tag. Between the style start and end tags, you place the rules you want to be followed. For example, you might want to use a black text color and white background color on your body font. If you want to use a color, you use a number sign and a color number. Or, you can use standard color names such as black, red, purple, blue, green.

You can control many other font functions such as style, size, and font family. There are many other things that can be added. For example, you can set the page margins. This will provide a margin on the right and left side of the browser window. An example of an internal style sheet:

```
<style type="text/css">
body {font-family: "Arial", sans-serif; margin-left: 15%; margin-right: 15%; }
p {color: black; font-size: 100%; font-family: "Arial", sans-serif; }
h1 {color: red; font-size: 200%; font-family: "Verdana", sans-serif; }
h2 {color: green; font-size: 150%; font-family: "Verdana", sans-serif; }
h3 {color: blue; font-size: 125%; font-family: "Verdana", sans-serif; }
</style>
```

Add styles to your own page. You may wish to specify styles for web links.

```
p {font-family: Verdana, Arial, Helvetica, sans-serif; text-decoration: none;}
a:link.p {font-family: Verdana, Arial, Helvetica, sans-serif; color: #000066; text-decoration: none;}
a:visited.p
{font-family: Verdana, Arial, Helvetica, sans-serif; color: #FF0000; text-decoration: none;}
a:hover.p
{font-family: Verdana, Arial, Helvetica, sans-serif; color: #FF0000; text-decoration: none;}
```

Also, consider styles for images and other objects.

```
img {float: right; padding: 5px;}
```

Use "class" to indicate a specific style such as `<p class="activity">`

```
.activity {font-family: Verdana, Arial, Helvetica, sans-serif; font-size: 75%; background-color:
#BBE3FF; display: block; padding: 15px;}
```

External Cascading Style Sheet

If you plan to use the same styles across pages in a website, it's a good idea to create an external style sheet that can be shared. This style sheet is a text file that is named with the .css file extension and called in the head of the document.

```
<link href="pets.css" rel="stylesheet" type="text/css" />
```

<http://eduscapes.com/sessions/makeover/css.htm>

Breadth and Depth through Multimedia

From slide shows to videos, multimedia can bring your website alive. The key is careful testing to ensure that all end users have access to the resources.

Keep It Simple

When possible, add features that don't require special software or plugins. Slide shows are a simple example.

Try It! Open a new document in Dreamweaver. Choose Commands>Create Web Photo Album. Create a name and provide a source folder "landmarks" and create a new folder for the slide show called "landmarksshow".



Using Plug-Ins

Plug-ins are external programs that you can add to the browser to enhance its capabilities. Add power to the browser and provide support to many other file types. Popular Plug-ins = Adobe Acrobat Reader, RealAudio and RealVideo, QuickTime, Shockwave and Flash and VRML. Be sure to provide help.

Audio and Podcasting

There are many ways to integrate audio clips into web pages. You can use various file formats including .wav, .au, .midi, and .mp3 files. You can use a text or graphic link. You can also put audio in a page background. Be careful when using audio because it can be VERY annoying when played in the background! Use `<bgsound src="lumberjack.wav">` to add a sound to the background of the page. You can add the loop attribute such as `loop="3"` to indicate how many times you wish to loop the sound. Audio can be very useful for emergent readers, people who speak or are learning foreign languages, or those learning new vocabulary. You can record sounds and save them as separate files to upload.

*<p>Click the sound icon *

. Can you name the television show? </p>

Video

Use your digital still camera to produce very short videos: Interviews with students and teachers about class activities. Interview people at special events such as book, science, and media fairs. Interview players at sporting events. Short short clips from musical and drama events. It's easy to embed a movie in Dreamweaver using the Insert>Media>Plugin option.

Separate Page. You can simply link to multimedia resources.

Prairie Dogs

Embed in Page. You can also embed a movie on a web page. For example, click Bears to view a page containing an embedded movie that automatically plays when the page opens.

<embed src="http://tipt3.utoledo.edu/starters/bears/grizzbear.mov" autoplay="true"> </embed>

Finding the Right Tools

Management and Discussion Forums

From simple discussion tools to complex course management systems, many tools are available for sharing information and resources.

Moodle - open source classroom management that includes forums, wikis, and chats.

Blogs

Blogs allow people to comment on postings.

Wikis

Wiki (pronounced “weekee”) use collaborative software to create dynamic web documents. Once established many wiki have no designated editor. Instead anyone can contribute.

Search Engines

When people visit your website, it’s essential that they can find what they need quickly. Sometimes the entry page navigation provides the answer. At other times, users are seeking something specific and really need access to a quality search tool.

Share Your Site

Once your website is online you’ll want to increase your traffic by submitting your website for inclusion in the search engines. This process takes some patience, but it will be worth it down the road. Although most of the search engine submission groups charge a fee, most have a small free section. You may have to look for the free link on their page, but it’s there somewhere.

Nifty Tools and Tips

PDFs. Consider using PDF files rather than web pages or graphics for documents such as newsletters or guidebooks.



Revitalize

Conduct Periodic Site Evaluation

- Ask people what they like and don't like
- Who comes to the site and why?
- Why do people not come to the site?
- What are they seeking?

Expand User Interactivity

- Use Macromedia Contribute
- Post PDF files
- Encourage blogging

Add Simple Elements

- Online forms
 - contact information
 - request for information
- Site search engine
- Virtual tour
- Slide shows
- Language options (Spanish, etc.)

Produce Quality Content

- Design original content rather than reproducing materials already available at other sites:
 - Local news
 - Historical information
 - Interviews
 - Discussions
 - Student-produced materials
- Be sure the website is frequently updated through the use of special features: blogs, news

Redesign for Intuitive Navigation

- Maintain a consistent layout for easy navigation.
- Limit the depth of the site so users don't need to go more than 3 layers down for information.

Save Time

- Eliminate things that are hard to update calendars - menus, sports, news, how can you handle differently?
- If you can't maintain website links, connect to a few unchanging resources rather than many individual websites.
- When possible, use external CSS and external JavaScripts so it's easy to make website wide changes.

