Intro to Dreamweaver CS4

and HTML

Office of Information Technology

West Virginia University

OIT Help Desk – (304) 293-4444, oithelp@mail.wvu.edu

http://oit.wvu.edu/training/classmat/

Table of Contents

Course Description

Dreamweaver CS4 is the de facto choice for gui-based website design. Adobe had its own web design application (GoLive), but after acquiring Macromedia (the makers of Dreamweaver and Fireworks), they integrated the best features of one in to the other. GoLive was discontinued, and Dreamweaver now has a robust and easy-to-use combination of features that continues to improve.

Dreamweaver allows a user with no, or limited, knowledge of XHTML to create and maintain websites. It is not a magic bullet, and will not do everything for you and wrap it up with a bow. It will, however, allow the average user the chance to get more done, in a shorter length of time, than they could learning and writing a website in straight XHTML.

We'll cover best practices in web design, how to get a picture of site structure, an intro to HTML, and more.

As with all software applications that are featured in OIT workshops, additional support can be obtained through the OIT Help Center by calling (304) 293.4444 or by sending an email to OITHelp@mail.wvu.edu .

The Office of Information Technology Trainers thank you for attending today's workshop. If there is anything we can do to make your learning experience more productive, please speak to the instructor.

Workshops currently offered in our Web Page Development Series

- Introduction to Dreamweaver and HTML
- Intermediate HTML using Dreamweaver
- Web Site Tools in Dreamweaver
- Cascading Style Sheets
- Creating Accessible Web Pages

Other Workshops that might be of interest to web developers

- Introduction to Adobe Acrobat
- Creating Accessible PDF Files

Additional and advanced training online via lynda.com – short-term free loaner accounts available from the Office of Information Technology. Go to oit.wvu.edu/training/lynda/, send email to workshop@mail.wvu.edu, or ask your instructor for more information.

Link and Learn

Useful Links

- **w3c.org**
  The World Wide Web Consortium (W3C) is an international community that develops standards to ensure the long-term growth of the Web. Join groups, and participate in W3C blogs and other discussion. We welcome your help to fulfill the W3C mission: to lead the Web to its full potential. (From w3c.org)

- **w3schools.com**
  The largest Web Developer site on the Net. Full web building tutorials- ALL FREE! (From w3c.org)

- **help.adobe.com**
  Tutorials and information from the producer of Dreamweaver.

- **www.section508.gov**
  Section 508 requires that Federal agencies' electronic and information technology is accessible to people with disabilities. IT Accessibility & Workforce Division, in the U.S. General Services Administration's Office of Governmentwide Policy, has been charged with the task of educating Federal employees and building the infrastructure necessary to support Section 508 implementation. Using this web site, Federal employees and the public can access resources for understanding and implementing the requirements of Section 508. (From section508.gov)

- **www.section508.info/**
  wave.webaim.org
  Check the accessibility of your pages/site with one of these handy tools.

- **www.maxdesign.com.au/presentation/checklist.htm**
  A partial, hyperlinked list of web standards to consider when building any site, and especially one that must be Section 508 compliant.

The Web. What is it, exactly?

WWW

World **W**ide **W**eb

The World Wide Web is a system of interlinked hypertext documents accessed via the Internet. With a web browser, one can view Web pages that may contain text, images, videos, and other multimedia and navigate between them using hyperlinks. Using concepts from earlier hypertext systems, English physicist Tim Berners-Lee, now the Director of the World Wide Web Consortium, wrote a proposal in March 1989 for what would eventually become the World Wide Web. He was later joined by Belgian computer scientist Robert Cailliau while both were working at CERN in Geneva, Switzerland. In 1990, they proposed using "HyperText [...] to link and access information of various kinds as a web of nodes in which the user can browse at will", and released that web in December.

Connected by the existing Internet, other websites were created, around the world, adding international standards for domain names and the HTML. Since then, Berners-Lee has played an active role in guiding the development of Web standards (such as the markup languages in which Web pages are composed), and in recent years has advocated his vision of a Semantic Web. The World Wide Web enabled the spread of information over the Internet through an easy-to-use and flexible format. It thus played an important role in popularizing use of the Internet. Although the two terms are sometimes conflated in popular use, *World Wide Web* is not synonymous with *Internet*. The Web is an application built on top of the Internet. (From Wikipedia)

URL

Uniform **R**esource **L**ocator

The Uniform Resource Locator was created in 1994 by Tim Berners-Lee as part of the URI.[3] The Uniform Resource Locator evolved out of the *Universal* Resource Locator. Berners-Lee has also "apologised" for the two slashes that precede the server name. (From Wikipedia)

HTML, XHTML, XML, and CSS. What are they, exactly?

HTML

Hyper**t**ext **M**arkup **L**anguage

Hyper Text Markup Language (HTML) is the encoding scheme used to create and format a web document. (From Wikipedia)

XHTML

E**x**tensible **H**yper**t**ext **M**arkup **L**anguage

Extensible Hypertext Markup Language (XHTML) is a family of markup languages that mirror or extend versions of the widely used HTML.

The only essential difference between XHTML and HTML is that XHTML must be well-formed XML, while HTML does not. Some examples of differences this imposes in practice are:

- In HTML, some tags (e.g.,
) are always empty and may not have closing tags. All elements must be explicitly closed in XHTML. XML permits two types of closing tag for empty elements:
 and
</br>. In XML these are interchangeable, and either can be used freely for any tag. However, if XHTML content is to be served under a text/html MIME type to legacy browsers, only the self-closing form should be used for always-empty elements (like
), and only the explicit closing tag should be used for elements that are not always empty (like). Otherwise, browsers will usually parse the tag incorrectly.

- Similarly, HTML permits omitting end tags for some elements, such as <p>. XHTML forbids this.

- In HTML, almost everything is case-insensitive, while in XML, all element and attribute names are case-sensitive. XHTML requires all element and attribute names to be lowercase, while in HTML documents it's common to find uppercase or even mixed-case names.

- Various versions of HTML often permit quotes to be omitted from attribute values, e.g., <body lang=en>. In XHTML, all attribute values must be enclosed by quotes, either single or double: <body lang="en"> or <body lang='en'>.

- HTML permits "attribute minimization", where boolean attributes can have their value omitted entirely, e.g., <option selected>. All XML attributes must have explicit quoted values, so in XML this would be written as <option selected="selected">.
- Some required elements may be omitted in HTML, in which case they are implicitly added by the parser. For instance, various versions of HTML don't require <html>, <head>, or <body> tags to be present unless they're intended to have attributes. (From Wikipedia)

XML

E**x**tensible **M**arkup **L**anguage

Extensible Markup Language (XML) is a set of rules for encoding documents electronically. (From Wikipedia)

- XML stands for EXtensible Markup Language
- XML is a markup language much like HTML
- XML was designed to carry data, not to display data
- XML tags are not predefined. You must define your own tags
- XML is designed to be self-descriptive
- XML is a W3C Recommendation (From W3C.org)

The Difference Between XML and HTML

XML is not a replacement for HTML.

XML and HTML were designed with different goals:

- XML was designed to transport and store data, with focus on what data is.
- HTML was designed to display data, with focus on how data looks.

HTML is about displaying information, while XML is about carrying information. Maybe it is a little hard to understand, but XML does not DO anything. XML was created to structure, store, and transport information. (From W3C.org)

CSS

Cascading **S**tyle **S**heets

Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation semantics (that is, the look and formatting) of a document written in a markup language. Its most common application is to style web pages written in HTML and XHTML, but the language can be applied to any kind of XML document. (From Wikipedia)

- **CSS** stands for **C**ascading **S**tyle **S**heets
- Styles define **how to display** HTML elements
- Styles were added to HTML 4.0 **to solve a problem**
- **External Style Sheets** can save a lot of work
- External Style Sheets are stored in **CSS files**

Styles Solved a Big Problem

HTML was never intended to contain tags for formatting a document.

HTML was intended to define the content of a document, like:

```
<h1>This is a heading</h1>

<p>This is a paragraph.</p>
```

When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large web sites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS.

In HTML 4.0, all formatting could be removed from the HTML document, and stored in a separate CSS file.

All browsers support CSS today. (From W3C.org)

Accessibility and Standards

Accessibility

Accessibility is an important idea behind many web standards, especially HTML.

Not only does this mean allowing the web to be used by people with disabilities, but also allowing web pages to be understood by people using browsers other than the usual ones – including voice browsers that read web pages aloud to people with sight impairments, Braille browsers that translate text into Braille, hand-held browsers with very little monitor space, teletext displays, and other unusual output devices.

As the variety of web access methods increases, adjusting or duplicating websites to satisfy all needs will become increasingly difficult (indeed, some say it's impossible even today). Following standards is a major step towards solving this problem. Making your sites standards-compliant will help ensure not only that traditional browsers, old and new, will all be able to present sites properly, but also that they will work with unusual browsers and media.

Some consequences of ignoring standards are obvious: the most basic consequence is that you will restrict access to your site. How much business sense does it make to limit your audience to only a fraction of those who wish to be a part of it? For a business site, denying access to even small portions of a target audience can make a big difference to your profit margin. For an educational site, it makes sense to allow access not only to affluent, able-bodied school-children with graphical browsers, but also to children in regions with poorly-developed infrastructure who are best served by text-only browsing, or disabled students using specialized browsers.

The same principle applies to all types of websites — while straying from the standards and taking advantage of browser-specific features may be tempting, the increased accessibility which comes from standards-compliance will lead to far greater rewards in the long run. (From webstandards.org)

Standards, the argument for

Tim Berners-Lee's dream for the World Wide Web was as a common space where users could share information, work together, play, and to socialize. As web developers -creating business, social, and educational sites- we turn this dream into reality.

But in this period of tremendous growth, the Web needs guidance to realize its full potential. Web standards are this guidance. These standards help ensure that everyone has access to the information we are providing. Standardization also makes web development faster and more enjoyable.

Standards compliance makes it easier for people with special needs to use the Web. Blind people may have their computer read web pages to them. People with poor eyesight may have pages rearranged and magnified for easier reading. And people using hand-held devices can browse the Web just as easily as those using high-end workstations.

As we will explain, there are also many practical reasons for developers to be concerned with web standards. Search engines can do a better job of indexing sites, for example. Using browser-specific code often doubles or triples the work to create web pages, and leaves a lot to be desired when new media are introduced. This situation will only get worse without the sound direction of web standards.

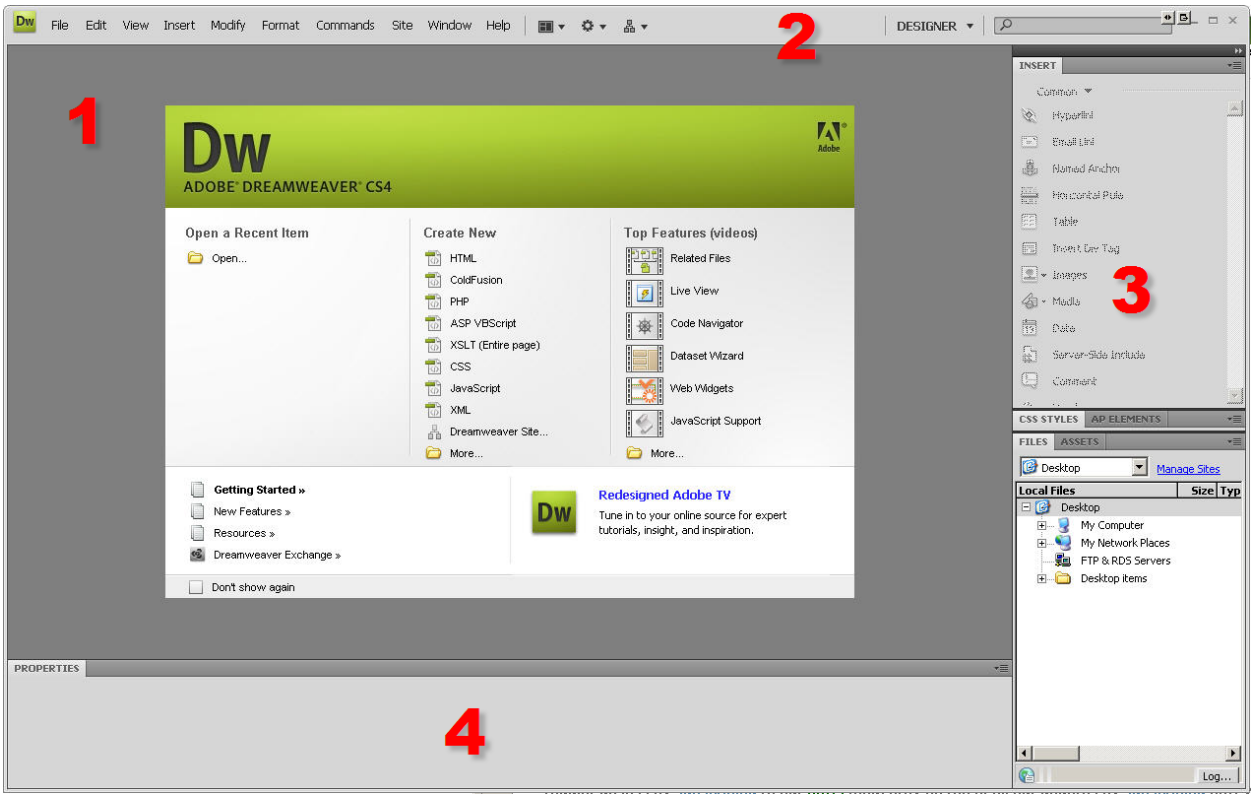
Some people fear that standards are limiting. In reality, they remove much of the tedious labour involved in web development, and give developers more time and more flexibility to be truly creative. They are both open to future improvement and mindful of past technology.

Many uses of the Web, including some that are only dreamed of today, will not be possible, or will be more difficult, without widespread standards compliance. At the moment, there are systems and software that are very common, seemingly close to universal, but who knows what tomorrow will bring? Tying ourselves to the control of any single company means limiting our future to the fortunes and misfortunes which that one company can or will provide. Maintaining universal standards will allow the Web to survive while encouraging innovation to continue at its current pace.

(Edited from webstandards.org)

Dreamweaver CS4 Interface

Dreamweaver CS4 is an attempt to match the user experience between Adobe's various applications. A new, grey-themed and unobtrusive setting greets each user when they start the program:

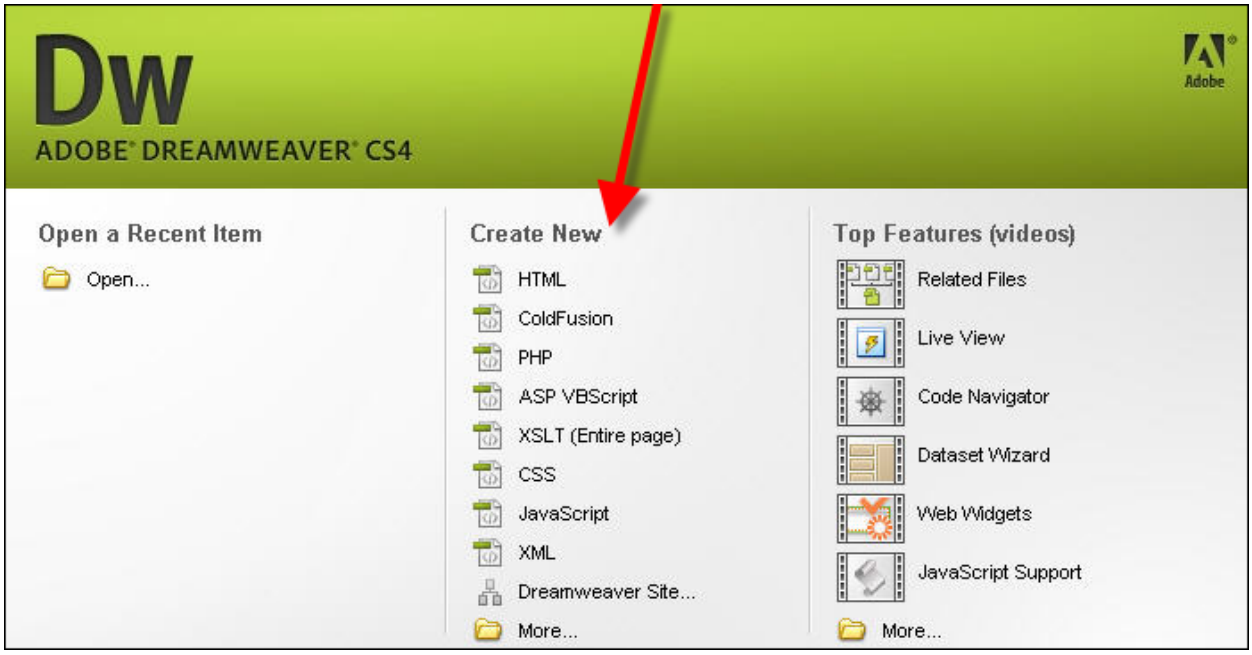


The screen is split in to four distinct parts:

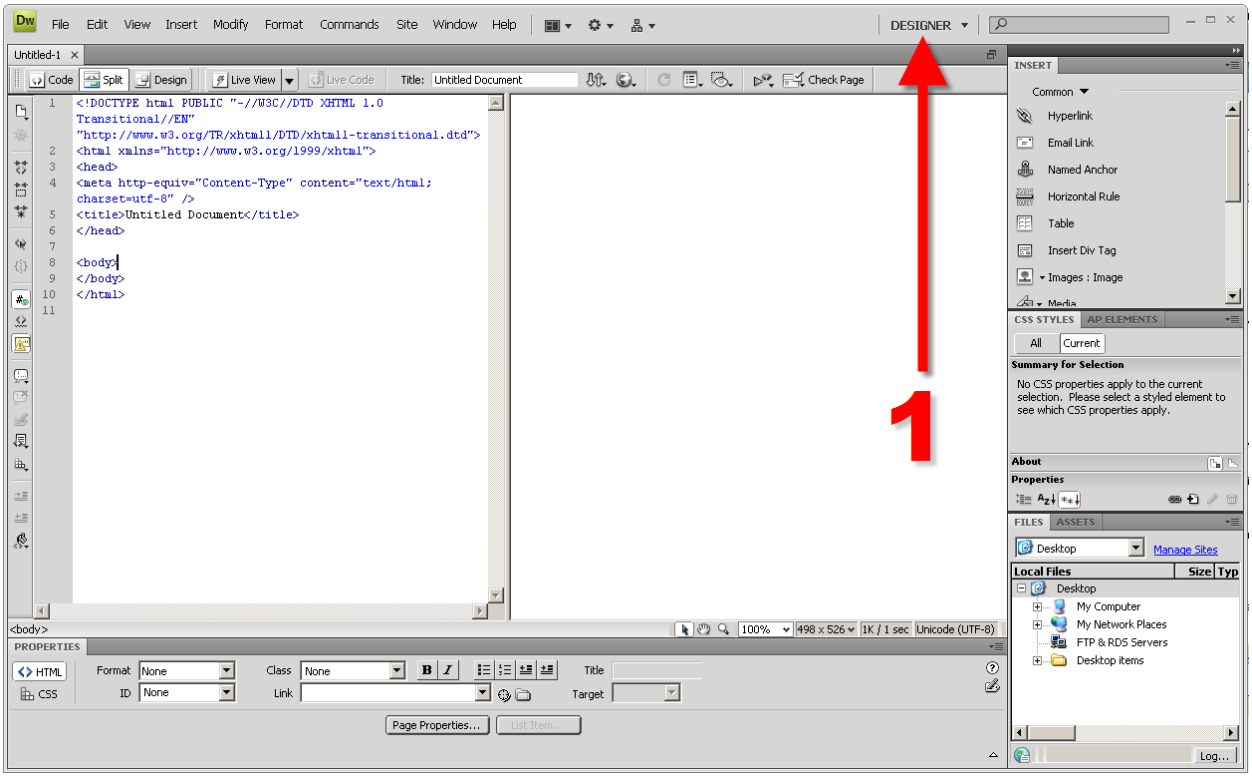
1) Document Window
2) Application Bar
3) Panels
4) Property Inspector

Dreamweaver CS4, New Document and Workspaces

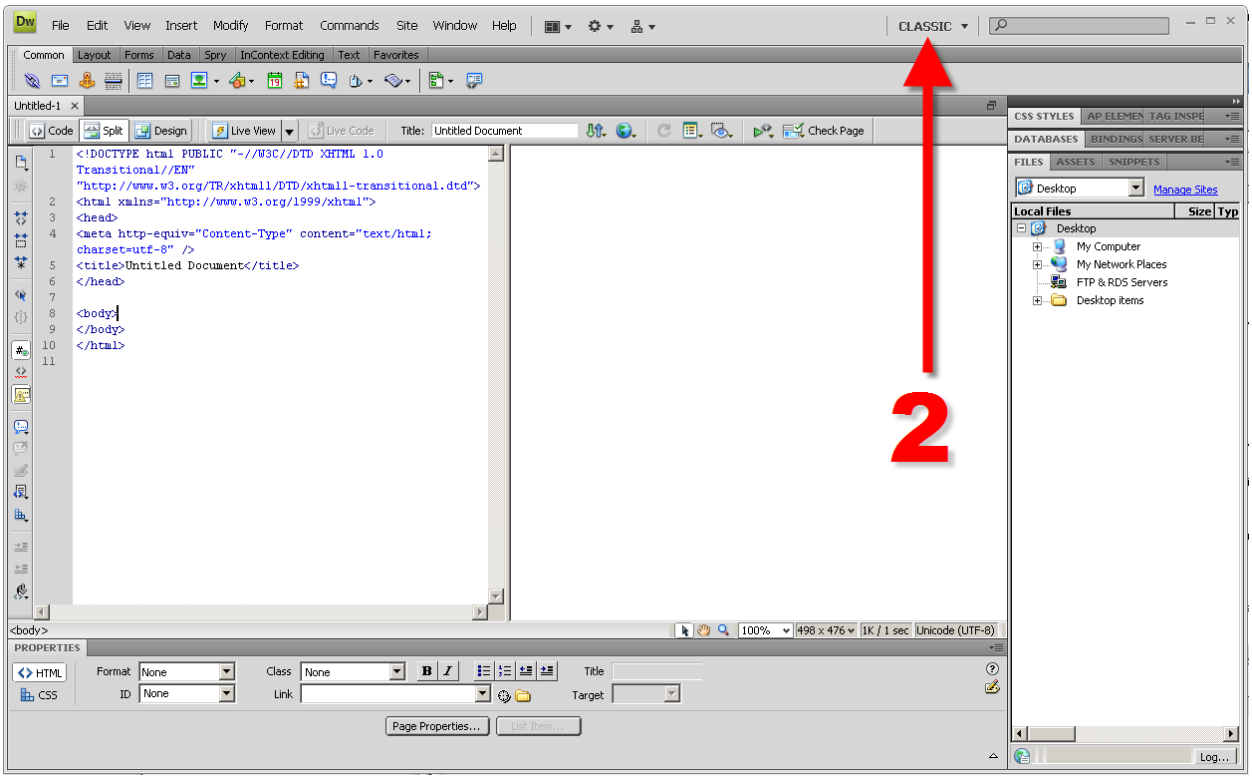
Create a new HTML document by clicking HTML under the Create New column:



Step 1: Change the Workspace to **Designer**. This is the default layout we'll use.



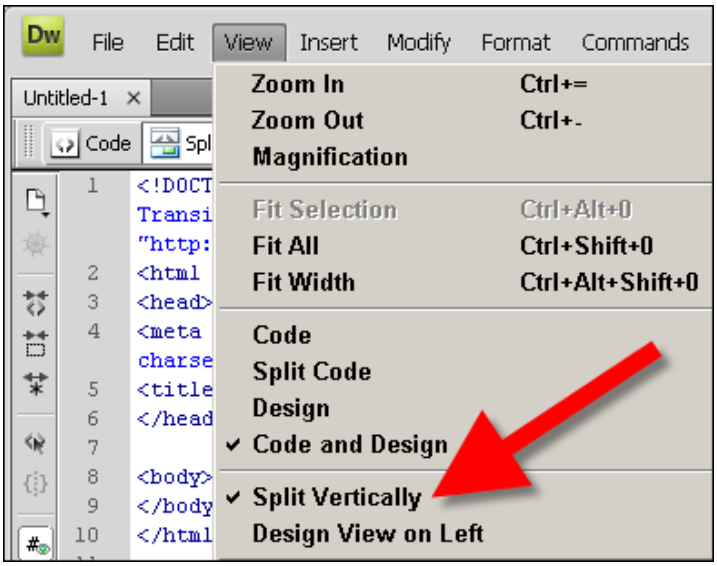
Step 2: Change the Workspace to **Classic**. This will help those transitioning from CS3 and earlier to this new version. You may create your own layouts and save them here.



Step 3: Change the Workspace back to **Designer** for the remainder of this workshop.

If your screen does not look like the examples, with two side-by-side windows in the middle of your Dreamweaver layout, do the following-

Step 4 (if needed): Go to Menu Bar > **View** > **Split Vertically**



You should have a check next to Split Vertically. If not, click on it to activate.

Jumping in. What have we here?

```
1  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
   Transitional//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2  <html xmlns="http://www.w3.org/1999/xhtml">
3  <head>
4  <meta http-equiv="Content-Type" content="text/html;
   charset=utf-8" />
5  <title>Untitled Document</title>
6  </head>
7
8  <body>
9  </body>
10 </html>
11
```

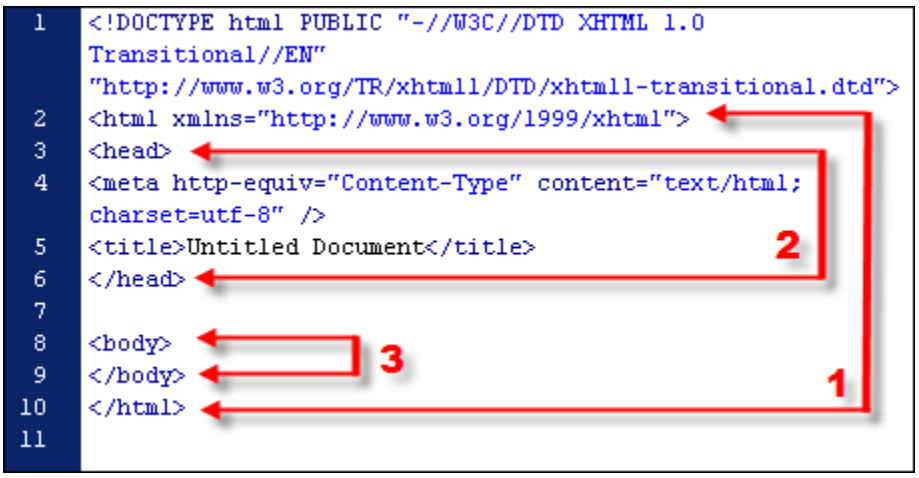
The image above should be what you see in the left side of our Dreamweaver window.

In Split Mode, the left window is called the Code view, and the right window is called the Design view.

Code view gives us a behind the scenes look at the HTML that determines how everything on your page looks.

Design view is a preview of what should display in your web browser.

So, what does this HTML mean?



1) Every web page has a beginning and end. All the html on a page should exist between the beginning tag <html> and the end tag </html>

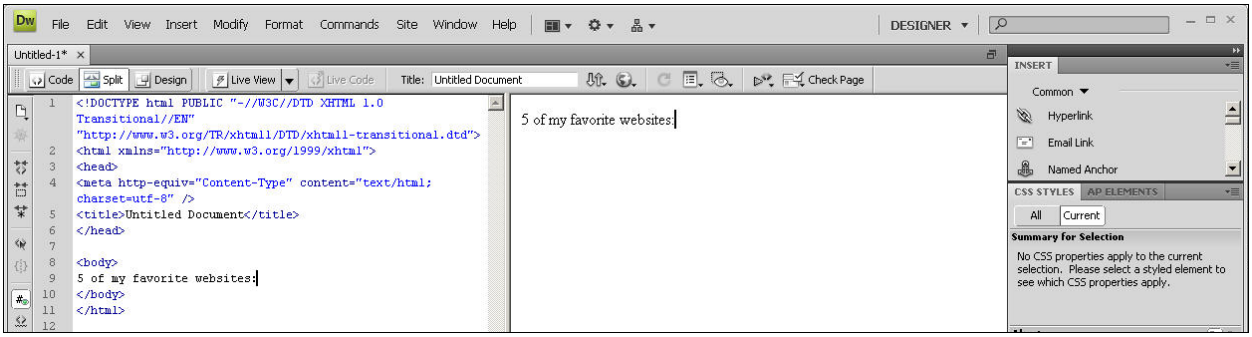
2) Every web page has a header section, where the title of the page lives. This is the title that shows up at the top of your web browser window(or on tabs within the web browser window).

Almost everything you'll put on your page lives in the body. We'll see this in action throughout our exercises.

Exercises

Intro to HTML formatting.

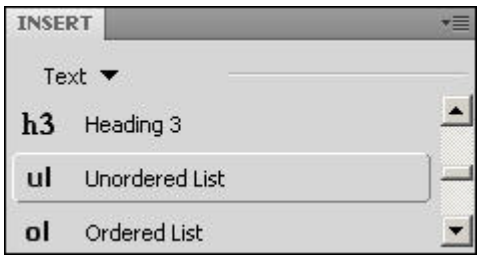
In the Design area of our window, type *5 of my favorite websites*



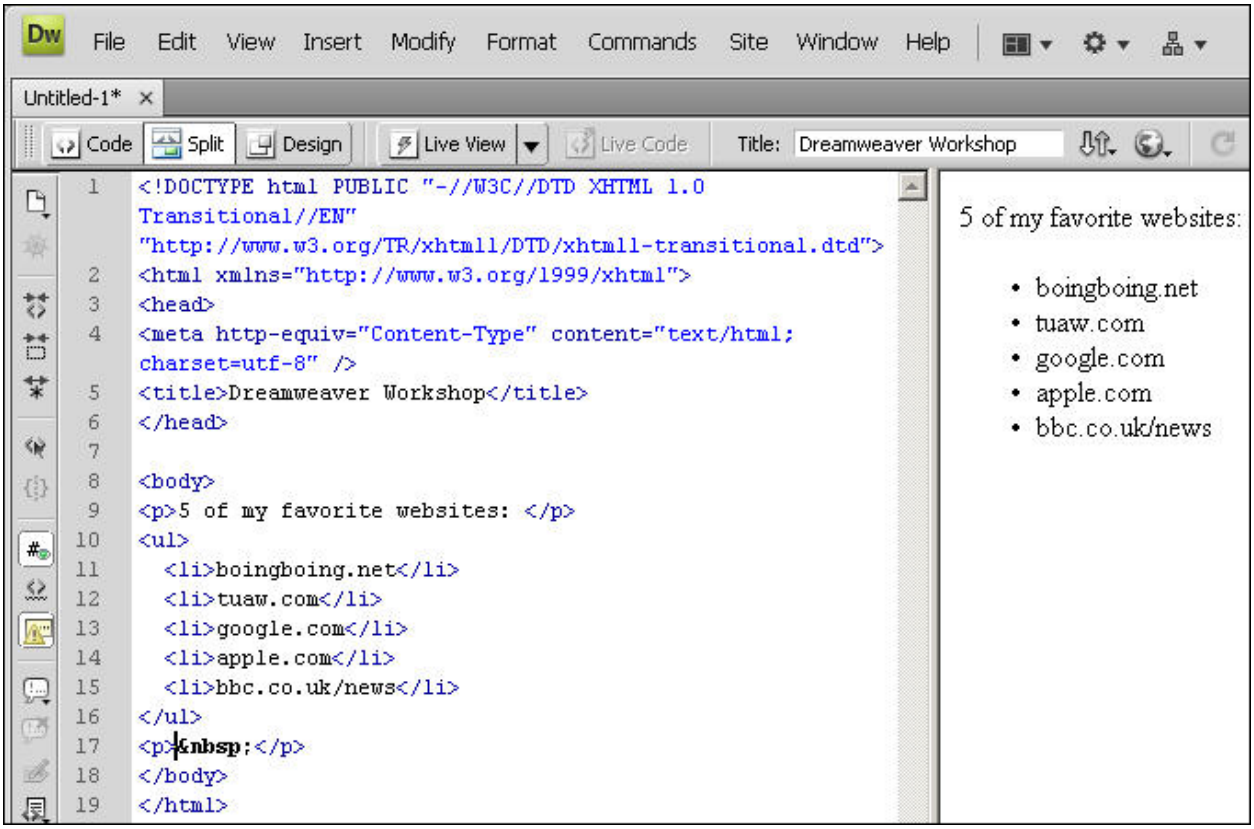
In the Coding area of our window, replace *Untitled Document* between the <title> tags with *Dreamweaver Workshop*

Go back to the Design side and put a hard return after *…websites:*

Look to the right, and in the Panels area click on the pull-down menu under the *Insert* tab. Choose *Text*, and click once on *ul Unordered List* to insert and Unordered List.



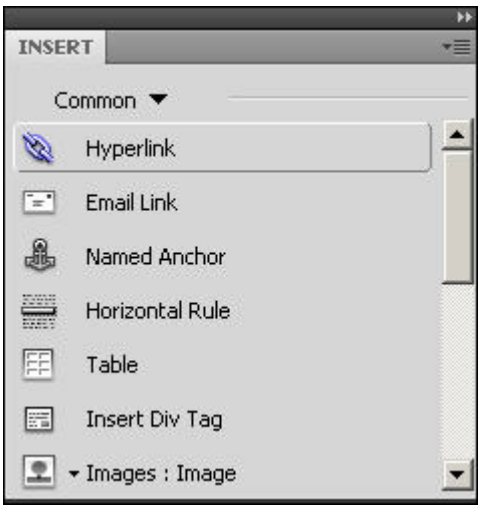
Type out the 5 websites you wish to display.
Dreamweaver inserts the appropriate HTML formatting for us *automagically*.

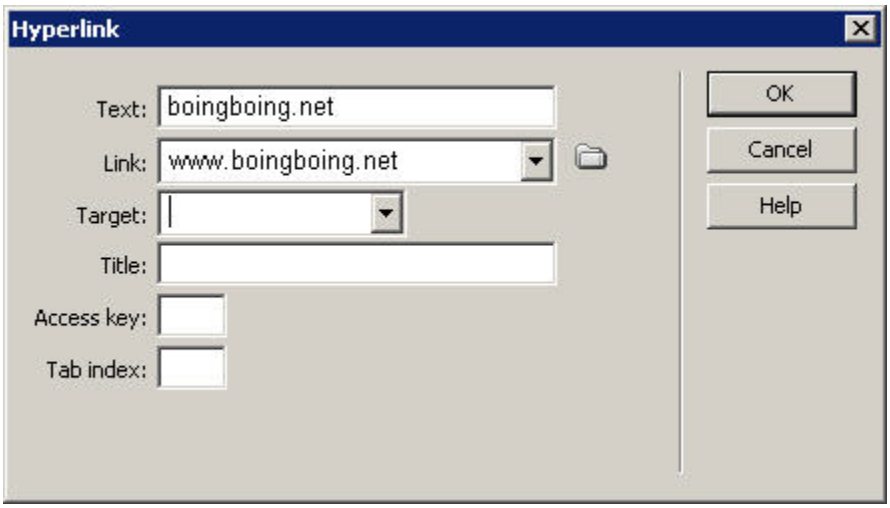


Inserting Hyperlinks

Highlight the text you want to link.

Switch from *Text* to *Common* under the Insert tab, then click once on *Hyperlink*.

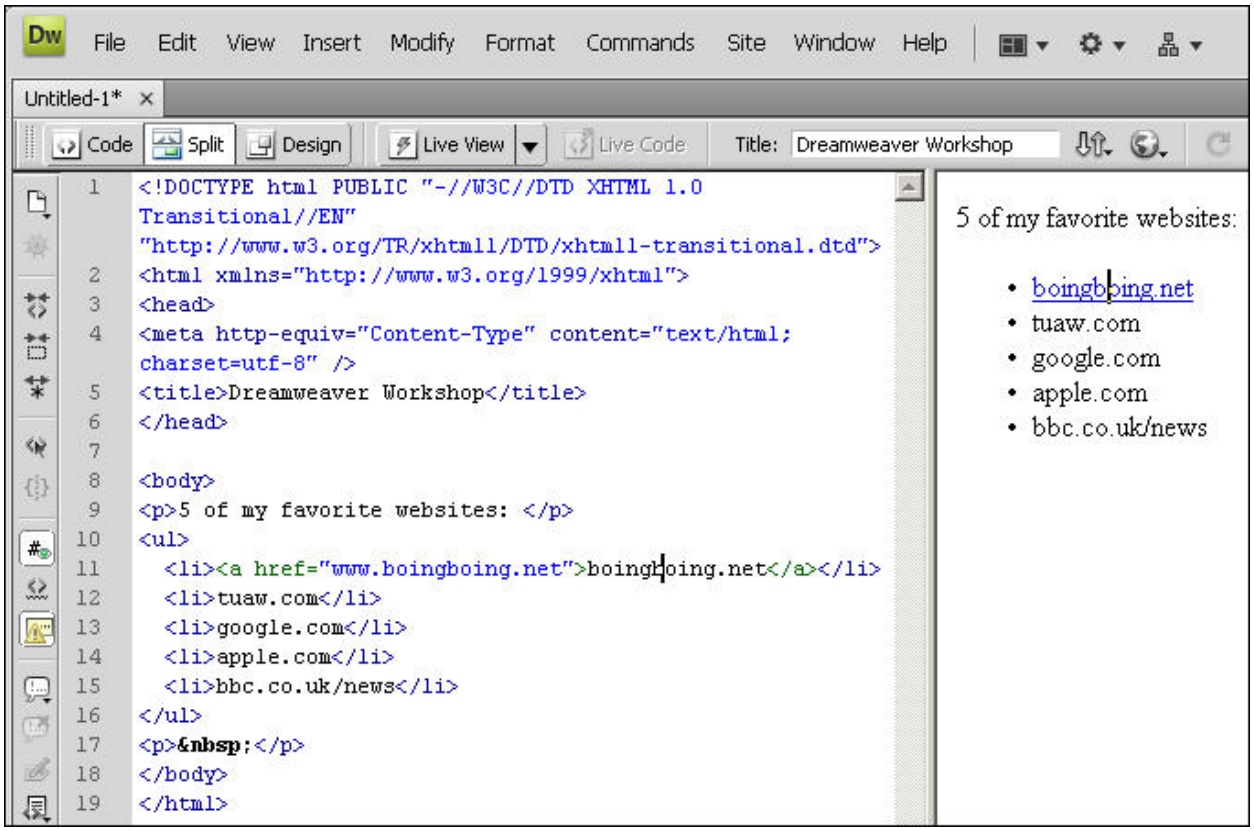




In the Hyperlink window, type the url you want in the *Link* field.

Click OK and Dreamweaver does all the heavy-lifting for you. Voila! A link!

You can hyperlink images, graphics, text- just about anything you can select with the mouse.



Alt tags

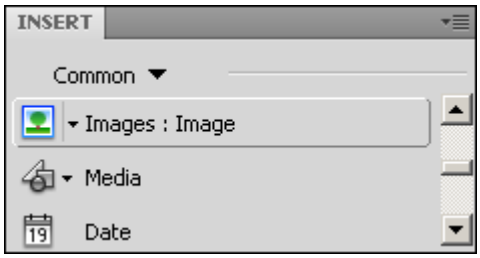
Have you ever seen an image on a website and thought 'I wonder who that is?' or 'I wonder what that is?' or even 'I wonder where that is?'.

An alt tag allows supplemental text to show briefly when the mouse hovers over an object. In an effort to comply with Section 508 guidelines and web standards, all your images and navigation graphics should have descriptive alt tags. Not only does this specifically help the sight impaired, it's good design that provides a richer experience for all users.

Alt tags are easy to create, too. Dreamweaver does all the hard work for you.

Let's insert an image on our page.

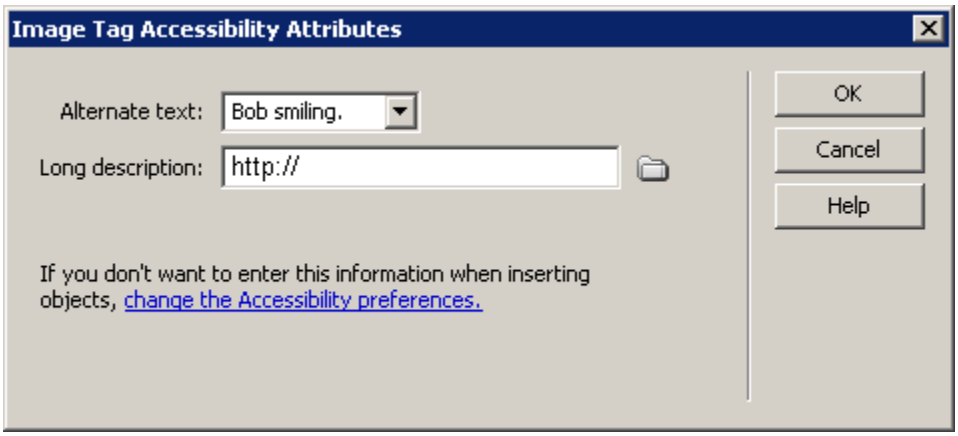
Under Common, click on the *Insert Image* option:



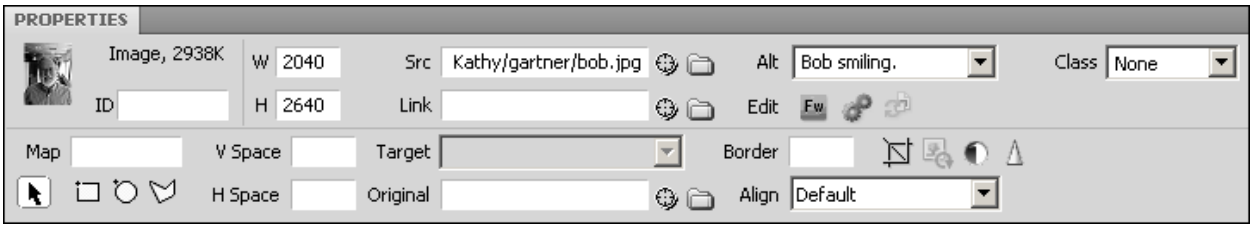
Once you choose an image to insert, Dreamweaver prompts you to add *Alternative text.*

Your description should be to the point and accurate. The more information you give, the better the experience for the user.

Type in the description for the image, and hit *OK*.

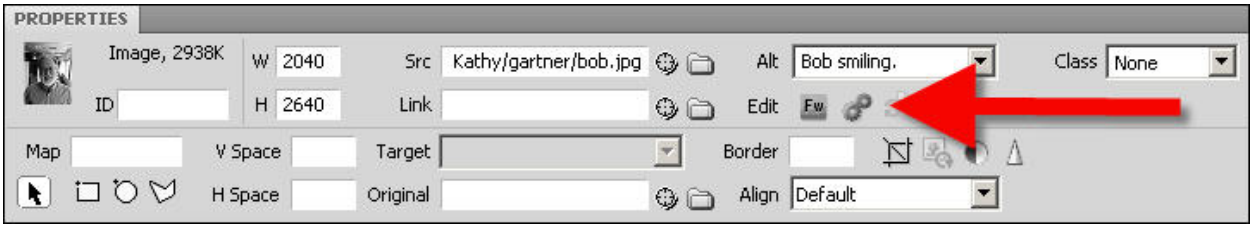


Odds are, the image you inserted for this exercise is too big. If you look at the contextual Properties panel at the bottom of the screen, we can change this and more.



Take note of the image size in Kilobytes in the upper left of the panel.

To the right of Edit, you'll have the option to work on the image in an outside editor. Additionally, if you click on the gears, you'll get to adjust the image so you can contrain proportions. Click on the two cogs (*Edit Image Settings*):

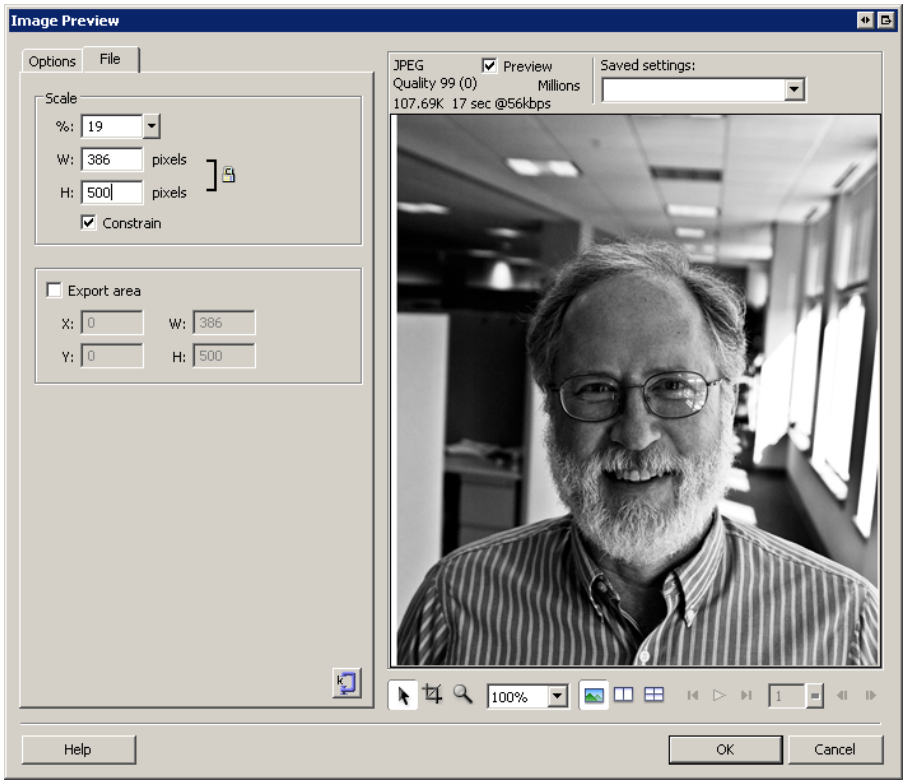


In the upper left of the Image Preview window, click on the *File* tab.

Make sure *Constrain* is checked. Change the longest dimension of your image to 500.

Hit the *Tab* key to get a preview of the change.

Click *OK*.

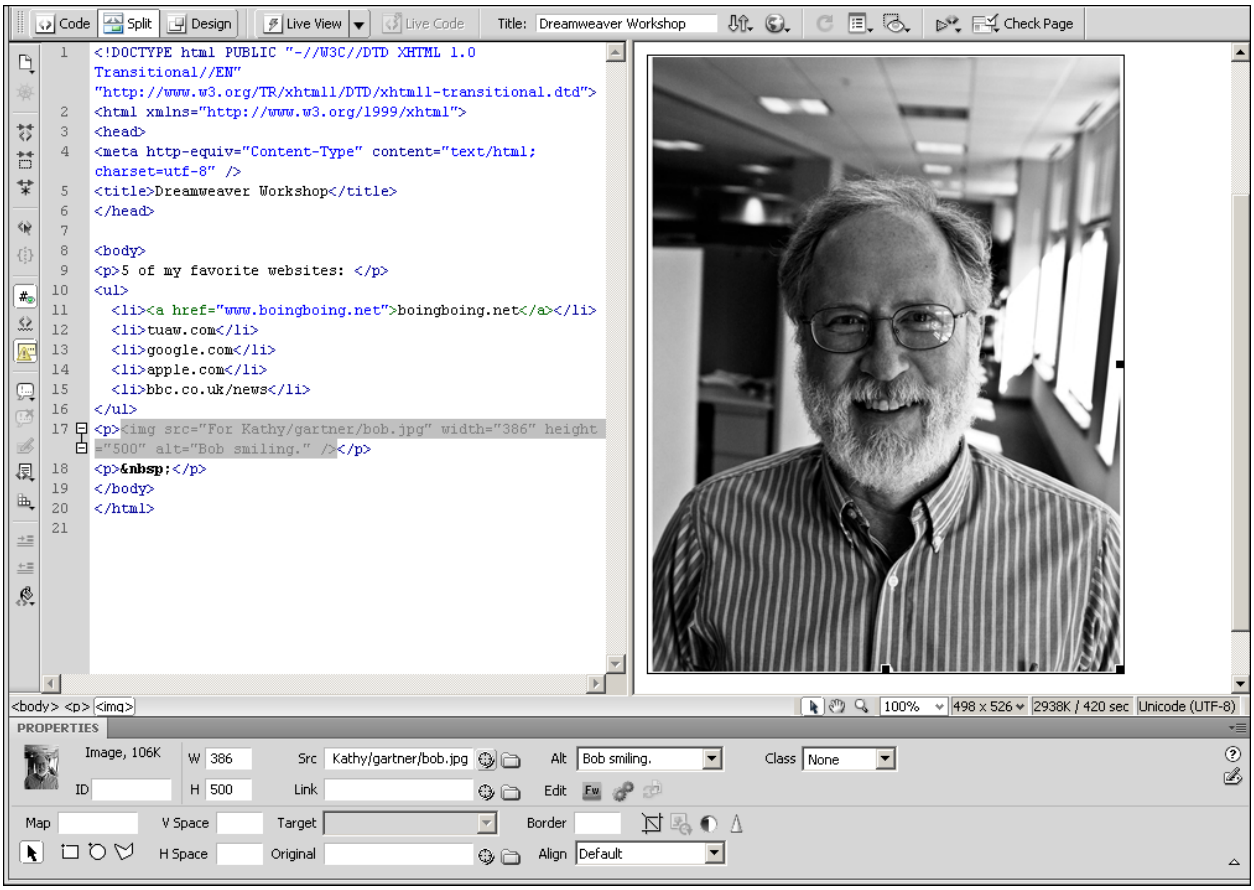


Your image should show up in Design View resized.

The HTML responsible for the image's insertion and alt text shows up on the Code side.

Note the change in the file size.

Your images should be smaller than 500 Kilobytes, as a general rule.



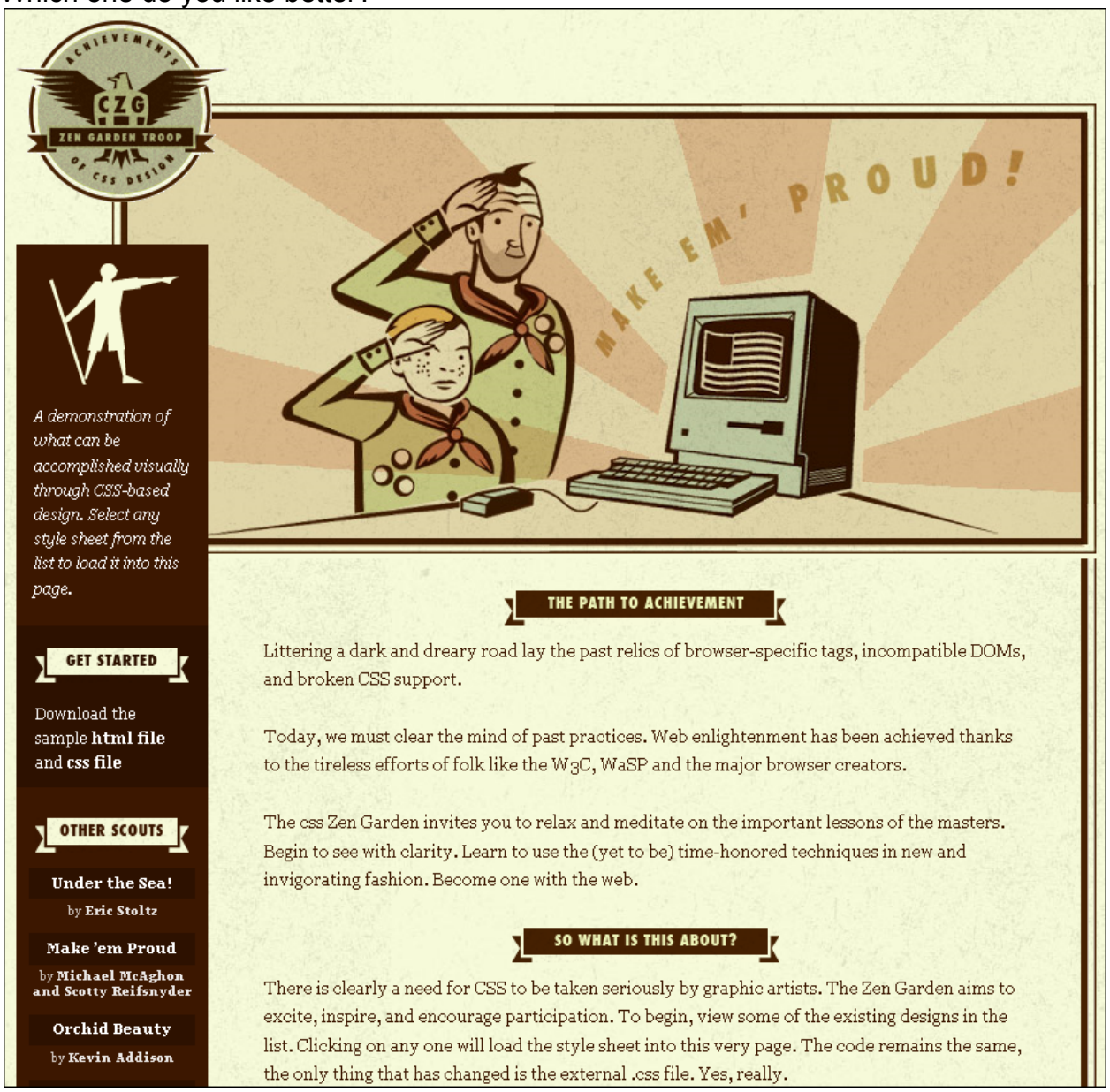
3 Pages, 1 CSS

http://www.csszengarden.com/

We won't cover the nuts and bolts of CSS in this workshop—that's for the later sessions. What we do need, however, is a good idea what CSS is and what it can do.

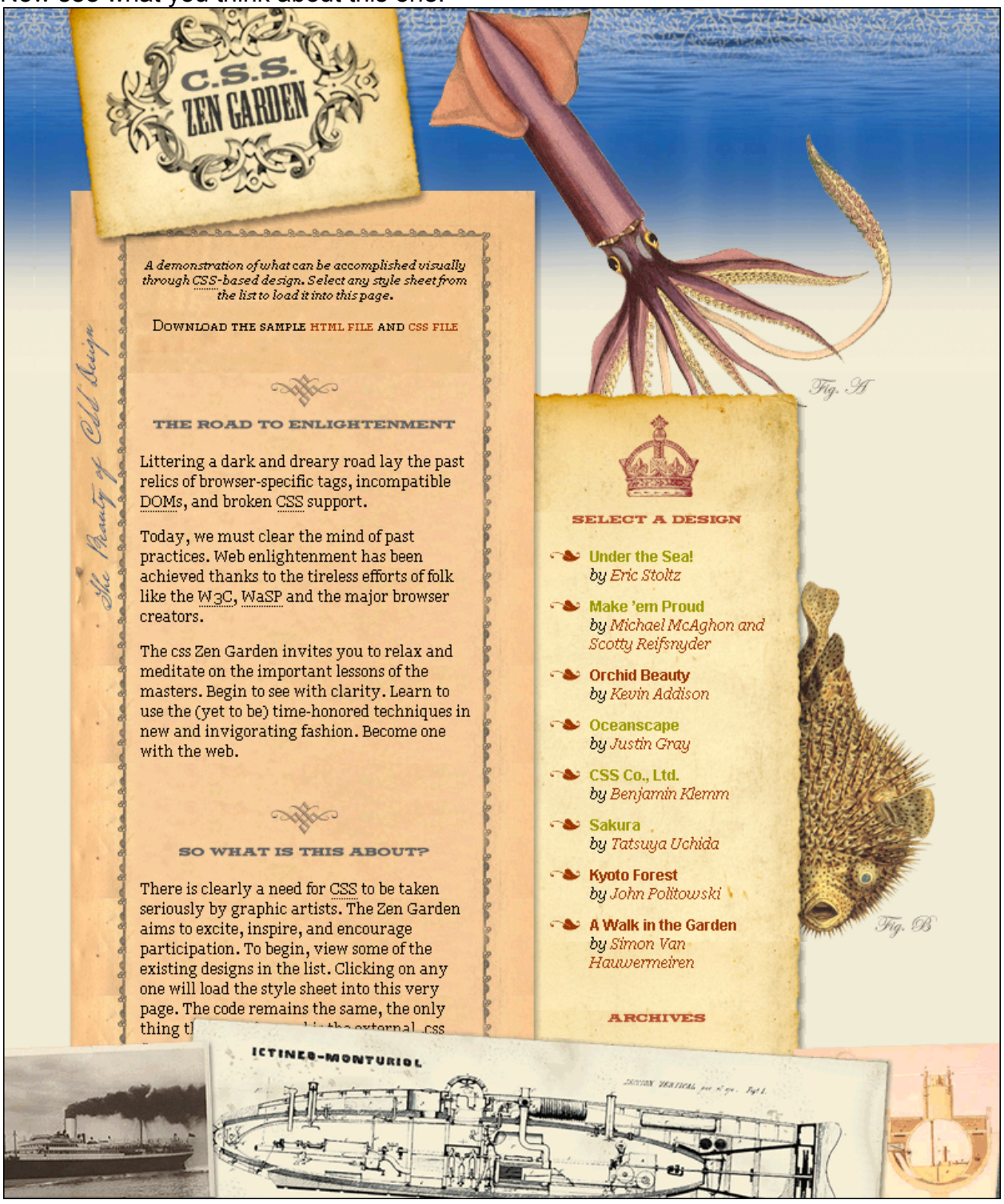
Check out this page:

Which one do you like better?



There isn't a right answer, but we want to give you some choices.
I'll put another for you to look at on the next page.

Now see what you think about this one:



These are examples of different cascading style sheets applied to the exact same data. The content is identical. The way it's presented is all that's changed.

CSS allows you to change the look and feel of an entire page or site by uploading 1 template. It's incredible, helps streamline redesigns and refreshes, and is definitely how people should think about constructing new web-based projects.

Photoshop and the Zen of Online Images

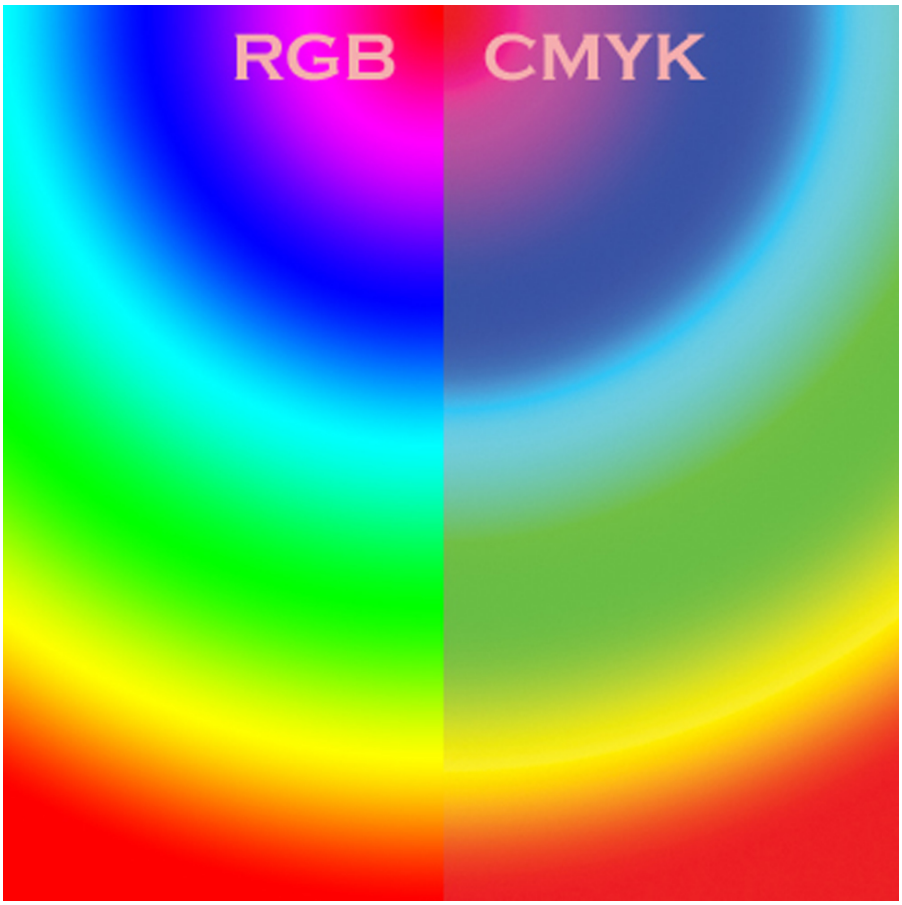
We put a section on graphics in this handout, so you might understand some terminology and techniques you'll undoubtedly come across, either in our workshops or on your own.

Color

Web-safe colors: Colors that are virtually guaranteed to display correctly on a monitor when viewed from a web browser.

RGB: Red, Green, Blue. How most monitors and printers reproduce all colors via combinations of these three basic colors.

CMYK: Cyan, Magenta, Yellow, and Black (or Key). A professional printing process that uses these 4 basic colors to approximate what you see in RGB.



Image/graphic file types

GIF: Smallest file size. Worst quality for images with complex colors, or images with lots of gradation.

JPG: File size larger than GIF, but well manageable for web use when files are saved correctly. Very good quality for images and heavy gradation.

TIFF: An uncompressed file format. Keeps layers. LARGE file sizes. Inappropriate for web use.

RAW/NEF: The highest quality image file available for digital cameras. H-U-G-E files. Totally unacceptable for web use.

PNG: Fireworks image/graphics file. File size between GIF and JPG. Excellent for anything GIF or JPG can do. Fast becoming the new favorite file type among Web Designers. Gives better image quality than a GIF, on par with jpg (some argue it's better), *and* it allows for transparency.

PSD: A Photoshop specific file. Keeps layers. Maintains high quality. HUMONGOUS files. T-totally, p-positively not for web use.

- Do you know the difference between lossy and lossless compression?

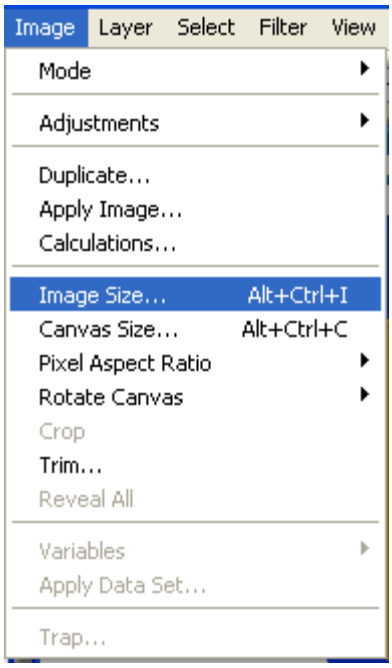
- Keep in mind the Adobe Publishing Triumvirate:
  - InDesign or Dreamweaver for publication
  - Photoshop for image and graphic manipulation
  - Illustrator and/or Fireworks for graphic creation

Image Size

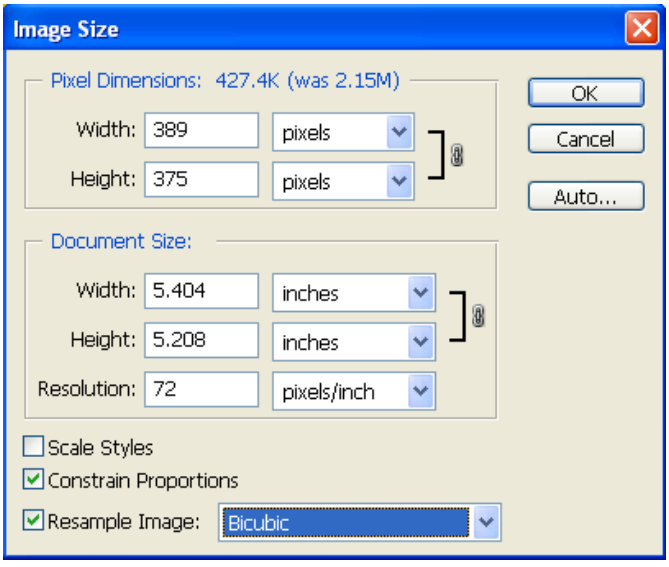
If you want greater control over image size and compression, Photoshop is the tool to use. It's more in-depth than resizing via Dreamweaver, but not required *if* you use Dreamweaver to adjust the size of your images.

Adjusting the size of your web image/graphic is incredibly important. Rarely do I display something greater than 500 pixels in a given dimension *unless* it is an integral part of the design of my page. Then I usually restrict the size of the image/graphic to no more than 1024x768 pixels to account for monitor sizes.

To change the size of your image/graphic in Photoshop, go to **Image**> **Image size**:



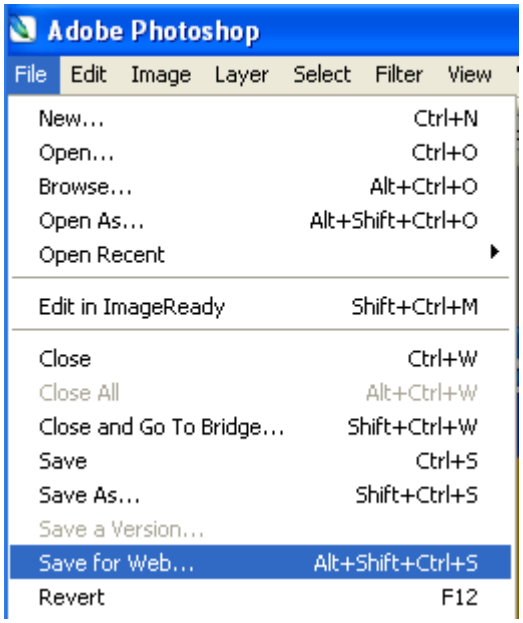
In the resulting window, you can change dimensional attributes of your file:



Save for Web

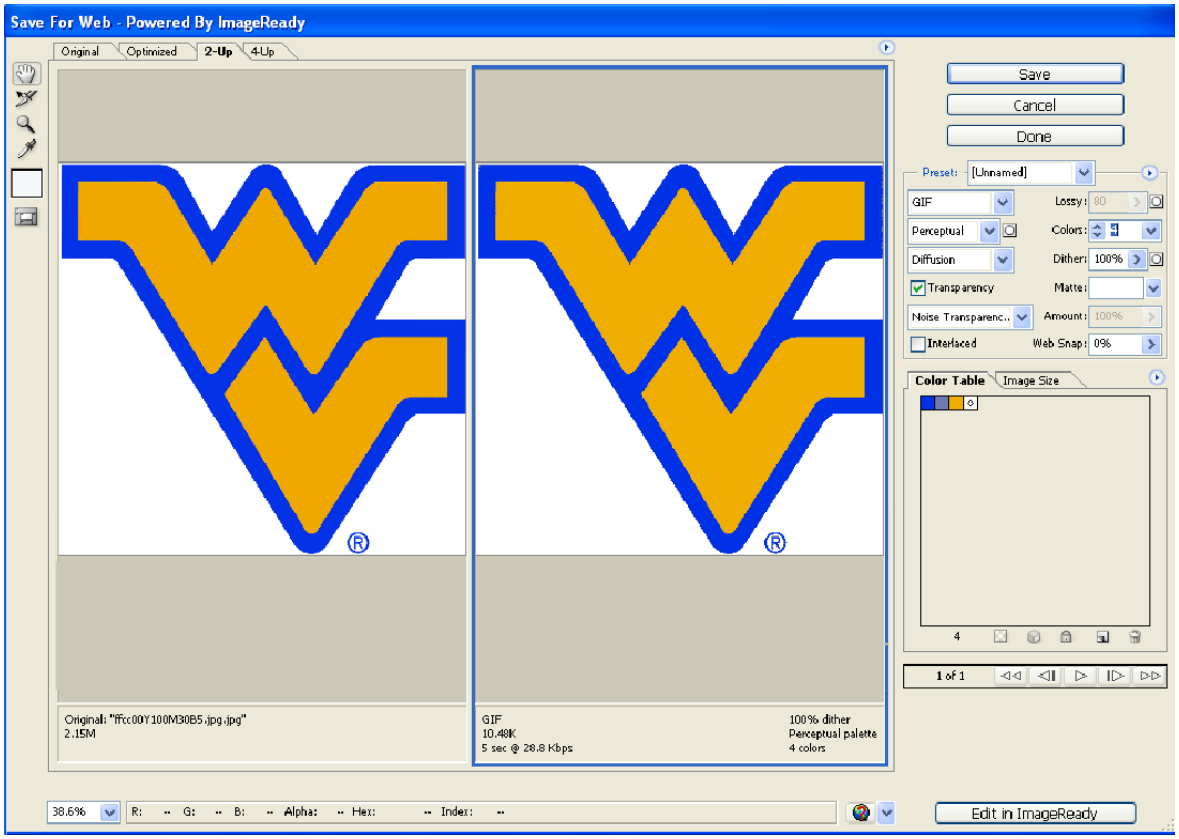
Photoshop makes it very easy to save your image to a smaller file size.

Simply go to **File**> **Save for Web**:



The Save for Web window allows you to choose file type and a myriad of other options to reduce the file size of your image/graphic.

This graphic went from 2MB to 10K by saving as a 4-color GIF:



More On Image Size and Saving for Web

- Make images a minimum of 72 pixels/inch, to a maximum of 96. The vast majority of modern monitors display images between 72 and 96 pixels/inch. Making your image more than this for web viewing is overkill, and a waste of bandwidth.

- Use Bicubic Sharper most of the time. If you have an image with a lot of subtle gradation, Bicubic Smoother may give better results.

- For general use, make sure Constrain Proportions and Resample Image are checked.

- People don't like to scroll. If you can't see an image in its entirety on a page, your viewer will feel cheated.

- Design 'above the fold'.

- Try to keep the average in mind at all times. You may have a 1600x1200 capable monitor, but laptops are starting to outsell desktops- and their resolution is usually much less than your deskbound 24 inch monitor.

- Web browsing isn't just on computers anymore. Think of all the people who surf the web on their phones.