

Web Page Creation (DreamWeaver)

Tak Auyeung, Ph.D.

March 7, 2004

- 20040201 2206 TA: found out how to run a local website with PHP support. Start to talk about site management in chapter 4, things are getting thicker now!
- 20040124 2205 TA: expanded basic text chapter, first assignment 2.6 will be assigned in the next class meeting, due on 2004/2/2
- 20040120 2133 TA: late, but let's see how much I can finish before the class.

Contents

0.1	Copyright Notice	5
1	Introduction	7
1.1	General Web Page Creation	7
1.1.1	Web page authoring approaches	7
1.1.2	Scope of this class	7
1.2	Logistics	8
2	Basic Text Elements	9
2.1	Starting with DW	9
2.2	Exercise: Can you make this?	9
2.3	Previewing and saving	9
2.4	Basic text formatting guidelines	14
2.5	Structural Text	14
2.5.1	Header	14
2.5.2	(Bulleted) Unordered Lists	14
2.5.3	(Numbered) Ordered Lists	16
2.6	Homework Assignment	16
2.6.1	Objectives of this Assignment	16
2.6.2	What to do?	17
2.6.3	Point breakdown	17
2.6.4	How to turn this in	17
3	Basic Tables	19
3.1	Components of a table	19
3.2	Creating a table	19
3.3	Changing Properties of a table	20
3.4	Selecting cells and changing their properties	21
3.5	Exercise (do not turn in)	21
4	Basic Web <i>Site</i> Design and Construction	23
4.1	Hyperlinks	23
4.1.1	Exercise (Do not turn in)	24
4.2	Site Management in DW	24
4.2.1	Give it a name	25
4.2.2	Choose a server technology	25
4.2.3	Development method	25
4.2.4	Testing Files URL	27
4.2.5	Remote Server	27
4.2.6	Done!	27
4.3	Working with a web site in DW	27

5	Graphics and Images	29
5.1	Basics	29
5.1.1	Local to documents	29
5.1.2	Subfolder among documents	29
5.1.3	Site-wide image folder	29
5.2	Finding images and using them	30
5.3	Creating your own images	30
5.3.1	You can still get into trouble!	30
5.3.2	Digitizing photos	30
5.3.3	Post digitization processing	31
5.4	Including pictures with DW	31
5.5	Properties of an image	31
6	Phase 2 of Your Project	33
6.1	Planning the web site	33
6.1.1	The Process	33
6.1.2	Presentation	34
6.2	What to turn in?	35
6.3	How to turn this in	36
7	Common Layout and Theme	37
7.1	Design a Template	37
7.1.1	Saving to Your Own Template File	37
7.1.2	The Layout View	37
7.1.3	Templates	37
7.2	Styles and Cascading Style Sheets	38
7.2.1	Where to Define Styles?	38
7.2.2	Handling Styles	38
7.2.3	Selecting a Style	39

0.1 Copyright Notice

All materials in this document are copyrighted. The author reserves all rights. Infringements will be prosecuted at the maximum extent allowed by law.

You are permitted to do the following:

1. add a link to the source of this document at www.mobots.com
2. view the materials online at www.mobots.com.
3. make copies (electronic or paper) for *personal* use only, given that:
 - (a) copies are not distributed by *any* means, you can always refer someone else to the source
 - (b) copyright notice and author information be preserved, you cannot cut and paste portions of this document without also copying the copyright notice

Chapter 1

Introduction

1.1 General Web Page Creation

Creating a web page, or a web site, is much like publishing an essay. There are many tools out there, some are free, some are expensive. In the end, however, it is not the tools, but the author, that makes the difference.

This class is about web page creation using DreamWeaver (hence DW). However, we will also cover a lot of topics that are not specific to DW. I think the non DW-specific topics are at least as important as the DW specific topics.

1.1.1 Web page authoring approaches

There are many ways to author web pages. For someone who knows HTML (hypertext markup language), even a plain text editor can be used to create web pages. Afterall, a web page is merely a text file with HTML content.

Although HTML is a fairly simple language and most people can learn it in a few days, it is very tedious to create web pages in HTML. I used to do it, so I definitely know.

To make web page authoring a little more convenient, there are some nice and free web page composers. For example, Netscape 7.x (Mozilla 1.x) has a composer component. These composers are often WYSIWYG (what-you-see-is-what-you-get), eliminating the need to learn HTML and all kind tags. Furthermore, you can also easily integrate graphical elements using these composers. Using the plain text file approach, it is difficult to “envision” how the product will look like.

The drawback of these composers is that they are typically page-oriented. In other words, they are great for constructing individual and static HTML files, but they do not have much to offer to structure a web site, let alone incorporating scripts (client side and server side) to make web pages more dynamic and interactive.

Next up are web site publishing software intended for end-users. Microsoft Frontpage, for example, is intended for end-users. Compared to Netscape/Mozilla Composer, Frontpage is more sophisticated, and it supports web *site* management. In other words, you can visually see how web pages are linked. Frontpage also offers scripting ability without needing the user to understand any scripting languages.

So, where is DW? DW is an expensive program (US\$399 retail, US\$99 with education discount) that is intended for commercial web site development. If text editors are bicycles, Composer is a moped, Frontpage is a passenger car, and DW is a camper (with the kitchen sink included).

What is “commercial” about DW is its extensive library of scripts and ability to interact with databases on different platforms. Yes, this is the kind of stuff that you need to author a web site that sells product, tracks customers and etc. As a I said, *commercial*.

1.1.2 Scope of this class

I am *guessing* that most students in this class are not, at least not immediately, going to create commercial web sites. As a result, this class will be taught at an end-user oriented fashion.

If you have any specific needs related to DreamWeaver, please contact me during my office hours or by email. I’d more than happy to explore features that are out of the scope of the class.

1.2 Logistics

DW is no light-duty software. Just the installation of the demo version (dated back to 2002) included on the CD in the book takes more than 300MB of disk space. The project files created by DW is also quite large compared to other tools. Floppy disks are *not* a viable medium for storing project files!

Only section A of room 152 is loaded with the full version of DW. This means you can only use the computers in section A to run DW. All of those computers are equipped with USB ports.

My recommendation is to get a USB flash storage device. If you intend to use USB devices to store all kinds of homework assignments and as a general portable storage device, the USB flash card reader are good options. This way, you only need to purchase additional flash media to expand the total storage ability. If you own digital cameras or other devices using flash media, a simple flash reader allows you to reuse the low capacity flash media (16MB or 32MB) that you have outgrown.

If you don't have a digital camera, and just want to get a USB mass storage device for this class, you should get the integrated ones. These are available from many places. Most stores selling computer peripherals will carry such devices. The advantages of these integrated units include the small size, ease to carry and ease of use. A quick survey reveals that, as of now (2004/01/20), a 256MB USB drive is about US\$40 from eBay. The retail price at a store is probably about US\$50 or so (remember, you usually have to pay for shipping and handling with online purchase).

Chapter 2

Basic Text Elements

If you have used a word processor before, you already know how to use DW for basic text elements. This is a relatively short chapter covering basic text elements.

2.1 Starting with DW

DW is not a simple program. Its user interface is a bit more complex than a word processor. In fact, you can compare the user interface of DW to that of programming environments like Microsoft Visual Studio.

Do not let this intimidate you. Let's start with just the most basic features of DW. Figure 2.1 is a screenshot of DW when you start it up.

Let's take a look at the various components of this screen. Figure 2.2 highlights the tabs that represent categories of components that you can insert into a web page. We can ignore this part for now.

Figure 2.3 highlights the buttons controlling the view. The default, Design View, is appropriate for now.

The large window in the middle is the editing area. This is where you edit a webpage in a WYSIWYG manner.

Figure 2.4 highlights the most important area for this chapter. This area lets you control properties of whatever element is selected. Since a web page is primarily a text document, it is usually displaying text formatting controls.

2.2 Exercise: Can you make this?

This is an exercise, not an assignment. Do this at the lab, but do not turn in anything. Not doing this exercise does not impact your grade directly, but if you do not use this chance to explore and learn how to DW, it can affect your ability to complete the projects (which determine your grade).

See if you can use just the properties control pane and the main editing area to create the web page as illustrated in figure 2.5.

Hints: use the font control, size control, color control, font style control, paragraph justification control, ENTER and shift-ENTER.

I'll show you how this is done in the following class meeting.

2.3 Previewing and saving

If you do not trust the WYSIWYG ability of DW, you can preview web pages created by DW using an external browser. To do this, access the File menu, then select Preview in Browser, then pick your favorite browser (usually Internet Explorer). This starts a new instance of Internet Explorer to view the HTML output of the current DW document.

For simple and mainly text web pages, I doubt that the preview will look any different from the editor. For complex web pages, especially those involving browser side scripts, you need to preview with a browser to test the interactive or dynamic elements.

When you work on your project, you should save it once in a while. Click the File menu, then select Save to save the document. Because of the complexity of DW, even saving file is not a simple operation. You need to choose an extension to save the file. For what we are doing now, a `.html` extension is appropriate.

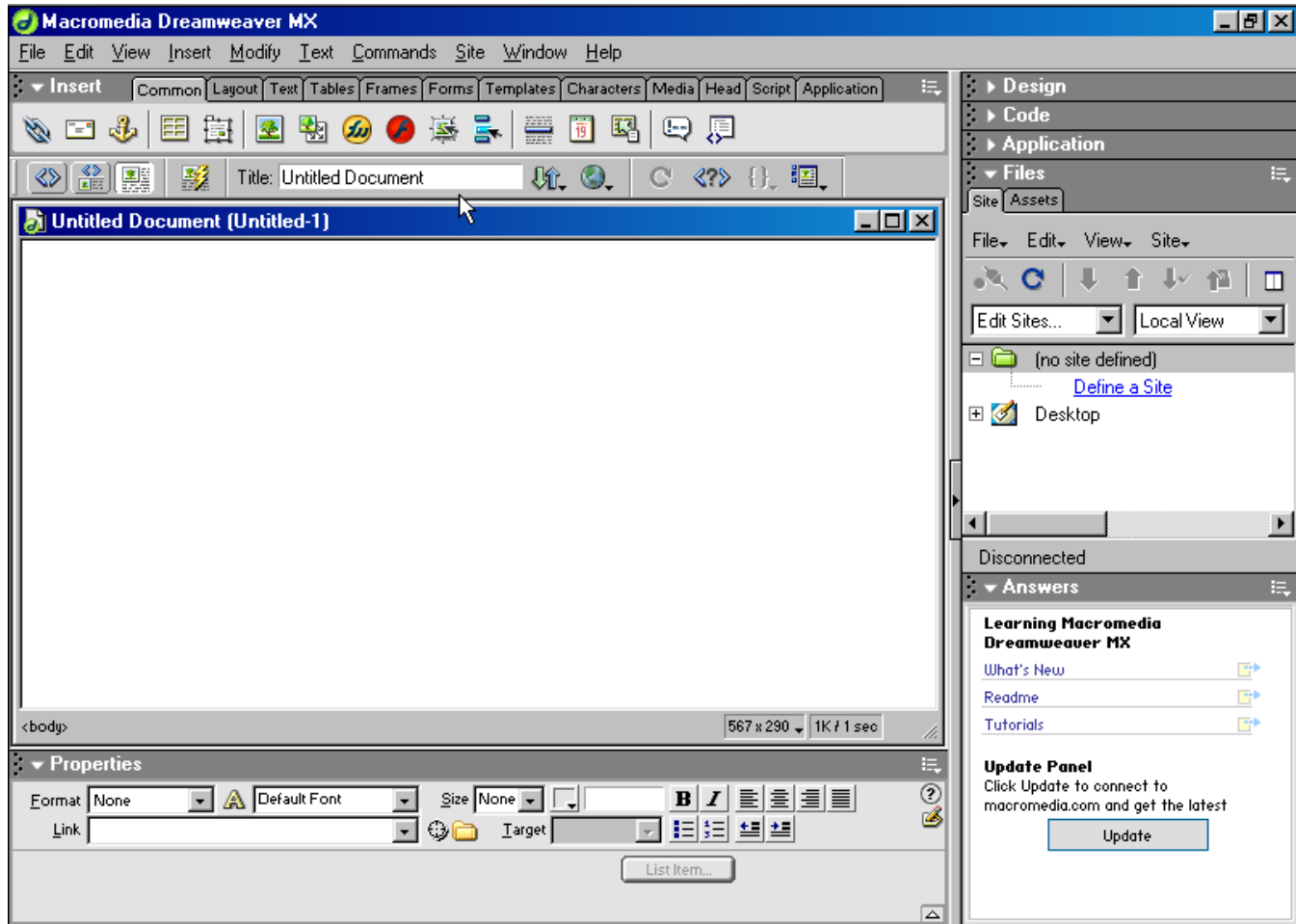


Figure 2.1: The start up screen of DW.

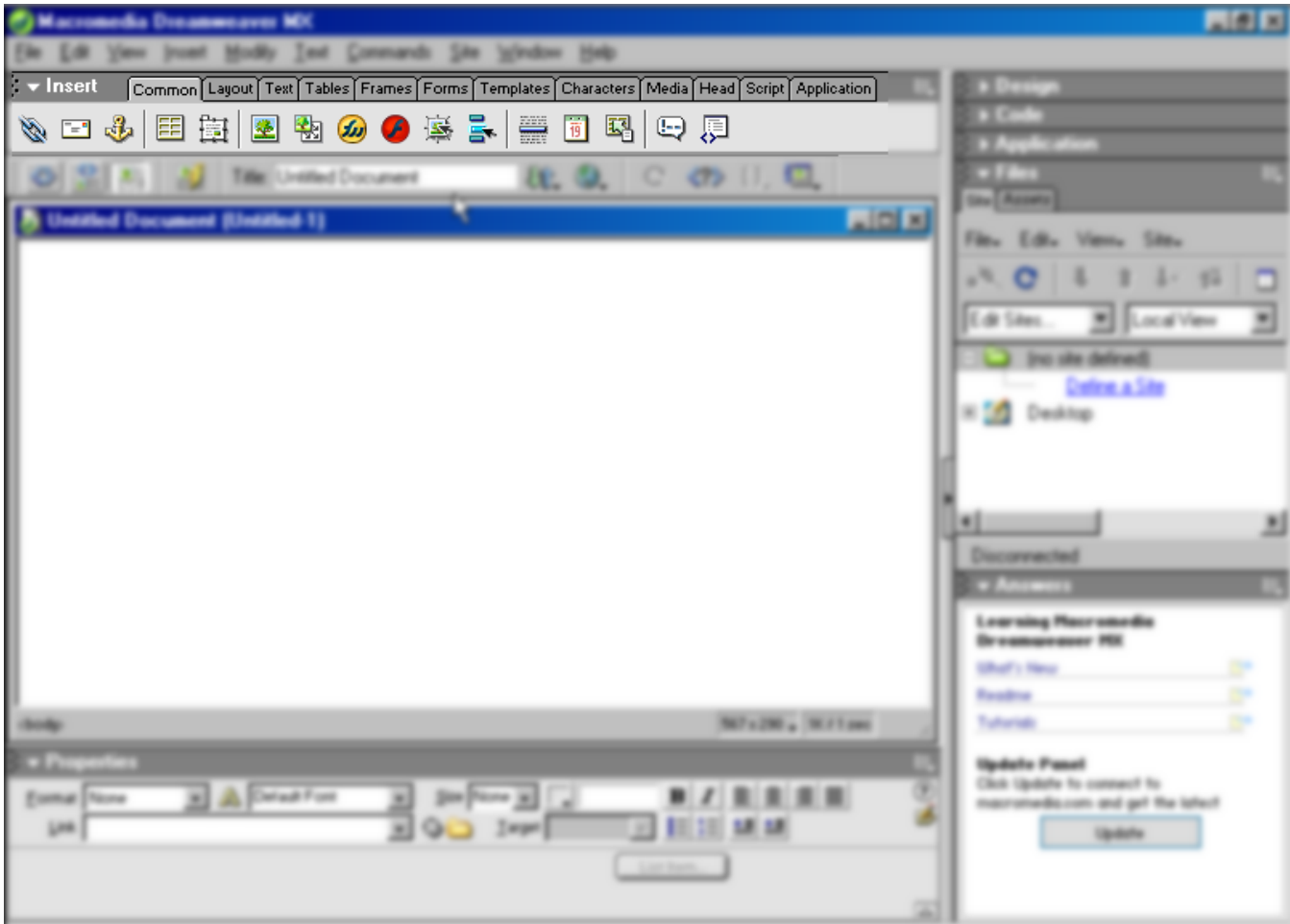


Figure 2.2: Component tabs of DW.

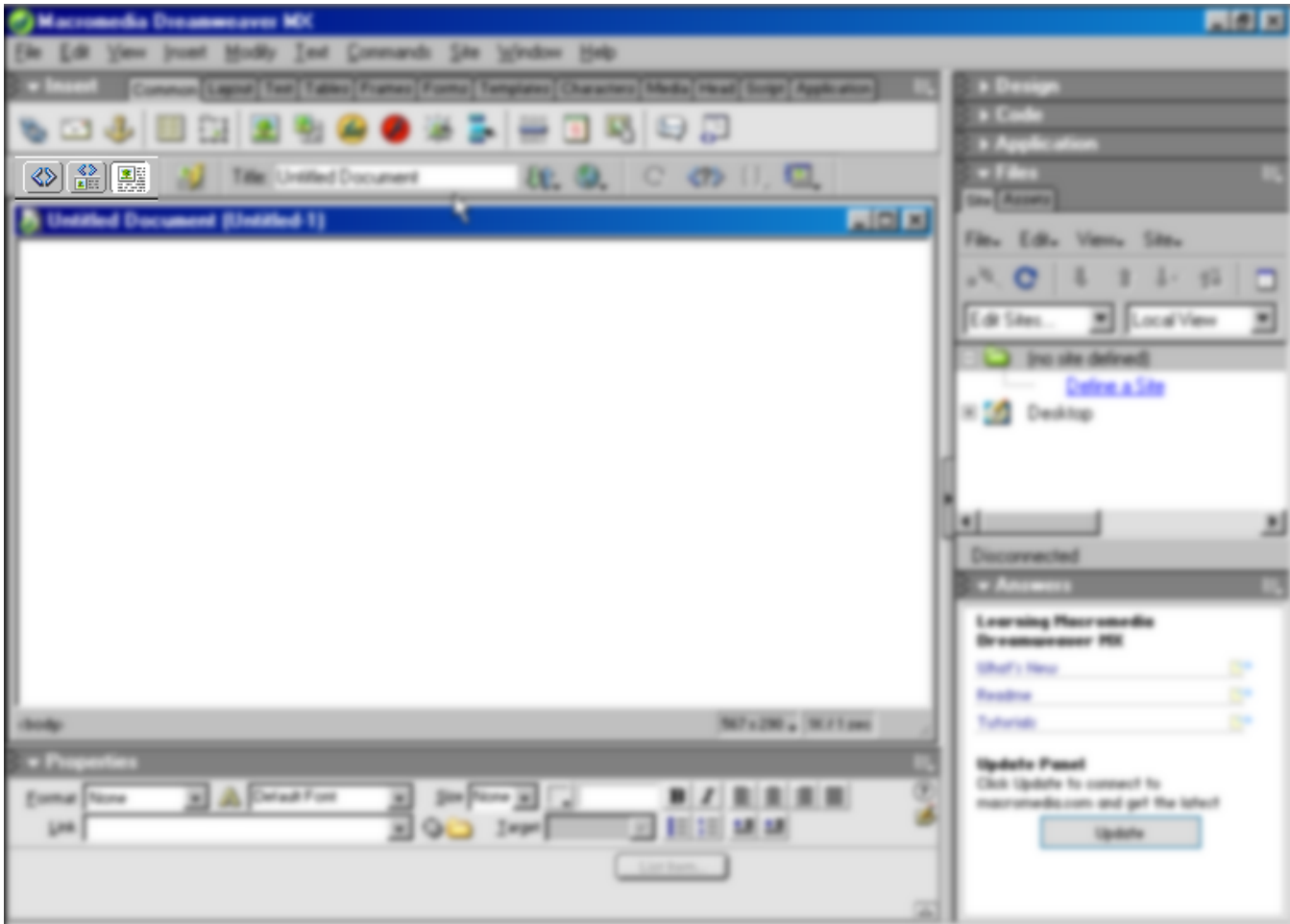


Figure 2.3: View control buttons of DW.

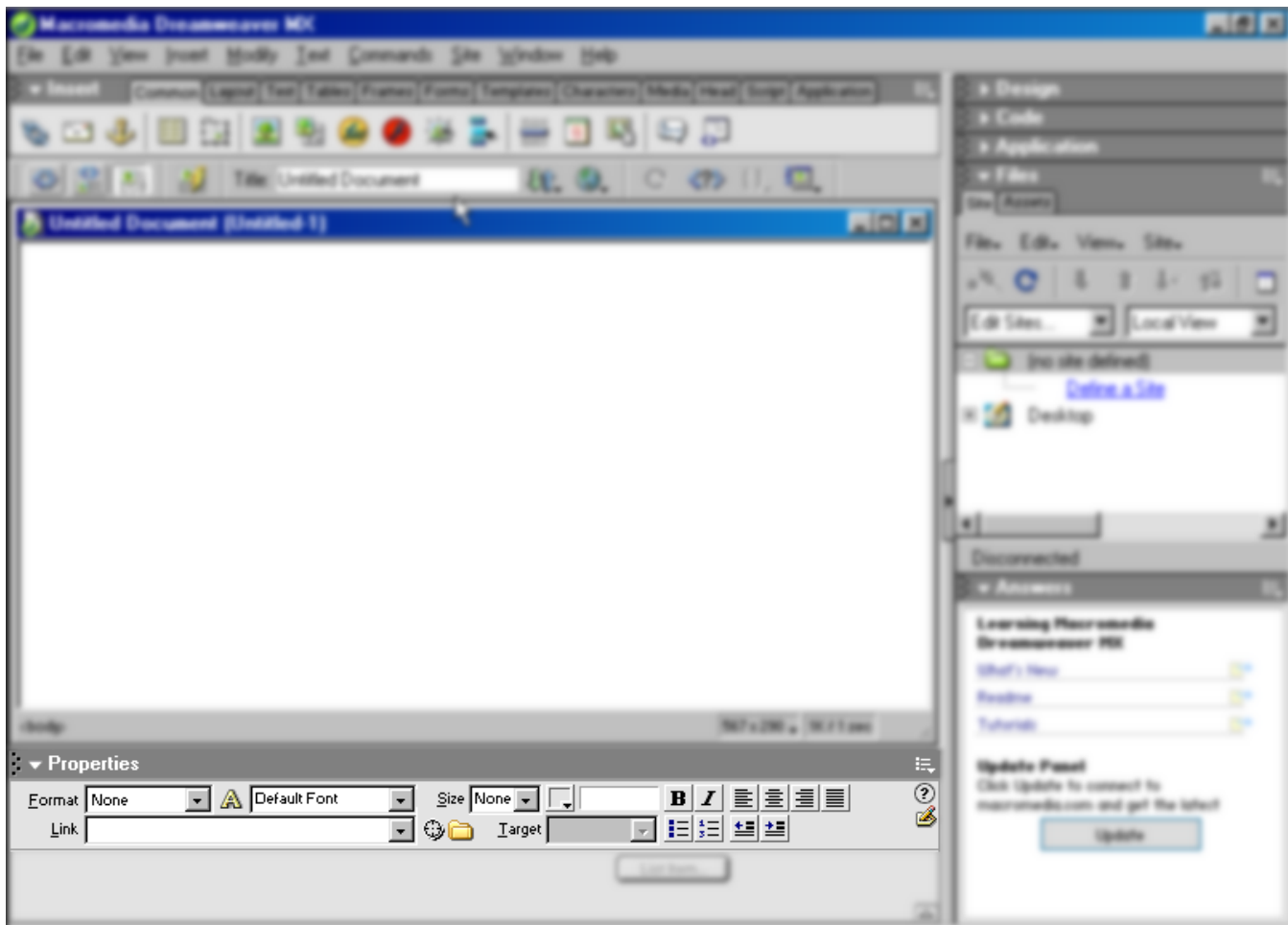


Figure 2.4: Properties control pane of DW.

Creating a web page can be
as boring as typing.

But it can also be as **creative** as
painting.

The sky isn't the limit,
you are!

Figure 2.5: Can you make this?

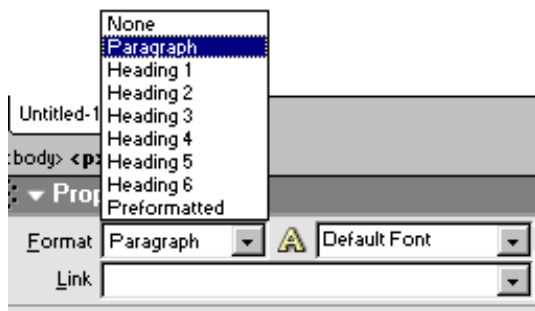


Figure 2.6: Drop down box to change paramgraph format.

2.4 Basic text formatting guidelines

There is nothing wrong with a web page that only has text. Splashy graphics and fancy formatting are not always necessary. In fact, for low bandwidth connections, plain-and-simple web pages with mostly just text elements are more appropriate! A picture is worth a thousand words, but a picture also has the download time of a thousand words!

At this point, I have shown you that you can format text in HTML much like you can format text in a word processor. Is it a good idea to type a document like the one in figure 2.5?

The answer depends on the message you are trying to get across to your readers. Changing the font, color or font size every other word demonstrates your mastery of font control, but it can be very distracting to the reader. As a result, your message may not get across effectively because the reader spends too much time appreciating the formatting (not!).

Furthermore, many people do not have color printers. Without a color printer, everything prints as black, gray and white. What is attention grabbing (red, yellow and other high-saturation colors) may become merely gray, or worse yet, light gray, when the document is printed.

2.5 Structural Text

Although text doesn't sound very structured, HTML has certain structural text elements so an author can prepare text in a structured manner.

2.5.1 Header

At the top level, HTML has 6 levels of headers. For those who know Microsoft Word, header styles have special meanings when it come to formatting and the generation of the table of contents. In HTML, header styles do not have as much importance. Nonetheless, they are available for the formatting of a sectioned document.

Figure 2.6 shows how you can change the header style of a paragraph in the Properties pane.


Note that when you change the format to a header style, you apply the change to the entire paragraph. In other words, header styles are applicable to entire paragraphs.

2.5.2 (Bulleted) Unordered Lists

An unordered list looks like the following in a browser:

- this is an item
- this is another item
- this is the last item

When you need to list a number of items, you can use the unordered list feature of HTML. Each paragraph is considered a list item in an unordered list.

To use this feature from DW, click the button that looks like . Once this mode is selected (the corresponding button looks depressed), all paragraphs are unordered list items. You can also select a block of paragraphs, then click this button to turn them into a series of unordered list items.

Hierarchical Lists

An item within a list can be, by itself, another list! This feature is useful when you need to describe something that is structured in layers. The following is an example (about sci-fi series):

- Farscape:
 - Puppets by Jim Henson & Co.
 - mostly Australian
 - mostly actors and actresses unknown to Americans
- Stargate SG1
 - lots of CGI
 - definitely American
 - has Richard Dean Anderson (McGuyver)


To make lists that are parts of another list, you have to options.

First, you can type everything as a single list.

- Farscape:
- Puppets by Jim Henson & Co.
- mostly Australian
- mostly actors and actresses unknown to Americans
- Stargate SG1
- lots of CGI
- definitely American
- has Richard Dean Anderson (McGuyver)

Highlight the portion that should become a sublist (note that you don't need to completely select each paragraphs, portions are fine),

- Farscape:
- Puppets by Jim Henson & Co.
- mostly Australian
- mostly actors and actresses unknown to Americans
- Stargate SG1
- lots of CGI
- definitely American
- has Richard Dean Anderson (McGuyver)

then click the Text Indent button (). The result is as follows:


- Farscape:
 - Puppets by Jim Henson & Co.
 - mostly Australian
 - mostly actors and actresses unknown to Americans
- Stargate SG1
- lots of CGI
- definitely American
- has Richard Dean Anderson (McGuyver)

Alternatively, you can type until you need to use a sublist:


- Farscape:
 - |

Then click the Text Indent button to make a sublist:

- Farscape:
 - puppets by |


If you need to go from a sublist back to the containing list level, use the Text Outdent button ()

Advanced Features

Whenever you are editing a list item, the List Item button () appears in the Properties pane. Click this button to see properties specific to a list item. For unordered lists, the options are fairly limited. You can change the “Style” from using a circular bullet to a square bullet.

Be careful that the List Properties dialog box is divided into two sections. The main section applies to the *entire* list, whereas the subarea titled “List Item” applies only to the specific list item. In other words, you can change the entire list to use square bullets, or just change one item to use a square bullet.

2.5.3 (Numbered) Ordered Lists

Ordered lists are similar to unordered lists, except that ordered lists use a numeric value instead of a bullet at the beginning of each list item. Use the Ordered List button () to make a paragraph (or a selection of paragraphs) an ordered list.

Advanced Features

Just like with an unordered list item, when you edit an ordered list item, the List Item button appears. Click on this button to see advanced features.

For ordered list items, there are more options. For example, you can change the numbering to a number of different styles. You can also manually change the counter.

Just like with unordered lists, the List Properties dialog box has items that apply to the whole list, and items that apply only to the list item that you are editing.

2.6 Homework Assignment

This homework assignment is phase 1 of your project. Be sure to read this section completely!

2.6.1 Objectives of this Assignment

This assignment gets you started to think about your project. Your project is going to be a website, and its content can be anything that you are interested in telling others (unless it is offensive or illegal).

Besides getting you to *think* about your website, this assignment also gets you started with DW and basic text features.

2.6.2 What to do?

In this homework assignment, describe the topic and scope of the website that you will construct as your project. Use DW to write this proposal as an HTML file.

In this file, be sure to include the following items:

- your name
- a one-sentence summary of your proposed website
- a section describing what type of information you intend to include in the website
- a section describing how pictures will be used at the website

2.6.3 Point breakdown

This subsection discusses how your first assignment will be scored. Each requirement item is all-of-none. In other words, if you miss one requirement, you lose all the points for that requirement. If you have questions, be sure to ask me before you turn in the assignment!

- contents:
 - name: I don't anyone will miss this, 10 points.
 - one-sentence summary: it has to be a meaningful and complete sentence that answers the question, 10 points.
 - type of information section: it has to be complete enough to convince me you know enough about the subject and there is enough material for the website, 20 points.
 - how pictures will be used: again, it has to be complete enough, and I expect explanations (why pictures should be used and how they enhance the website), 20 points.
- layout and formatting:
 - Use header styles, and make your proposal structured to have at least two layers (use header1 and header2 styles). 20 points.
 - Use hierarchical lists (unordered or ordered). At least one list should have at least two layers (sublists). 20 points.
 - At least one font change in the document. 10 points.
 - At least one color change in the document. 10 points.
 - At least one font size change in the document. 10 points.

2.6.4 How to turn this in

Turn it in by email. You do not need to use a ZIP account to send the email. I reply with your score to whatever account you send the assignment.

Use the following as the subject line of your email:

CISC306 Phase 1 by *your name*

Substitute *your name* with your actual name.

I suggest you copy and paste the subject line from this document to make sure it conforms to the standard. I use sorting by subject in my mailbox to collect the assignments. If you do not follow the rules, your email will not show up with the rest of the class, and you may lose all the points for the homework assignment!

If I receive a submission that does not follow the standard subject line, 50 (yes, fifty) points will be deducted.

The due date is 2004/2/2.

Chapter 3

Basic Tables

Table is of the most important formatting components offered by HTML. Besides its obvious ability to display data in a tabular format, tables are also used, in conjunction with graphics components, to create splashy webpages with seemingly custom graphical backgrounds.

In this chapter, we will learn the basic features of tables, and how to access those features in DW.


3.1 Components of a table

A table has the following components:

- table header: this is an optional component to display the header for each column in a table
- table row: each table row represents a row in the table
- table data: each table data represents a cell in a row
- table caption: this is an optional component to display a description of the table

When you use the outline view in DW, you typically do not need to worry about these components of a table.

3.2 Creating a table

Creating a table in DW is easy. Using *either* the “Common” or “Tables” tab, click on the table icon () , and a dialog box pops up to ask you about the table you want to create. See figure 3.1 for an example of this dialog box.

Let's examine this dialog box.

- Rows: this is the number of rows in the table

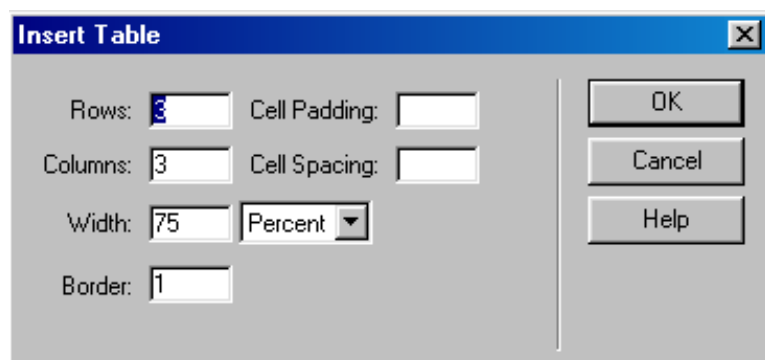


Figure 3.1: Insert table dialog box.

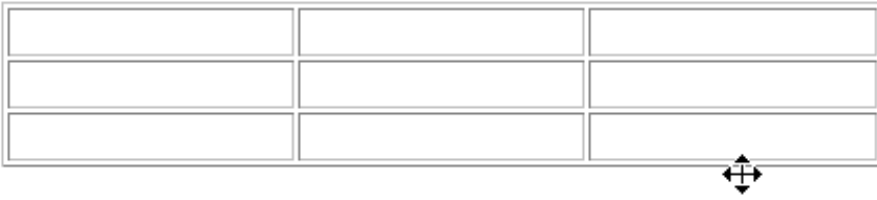


Figure 3.2: To select a table.

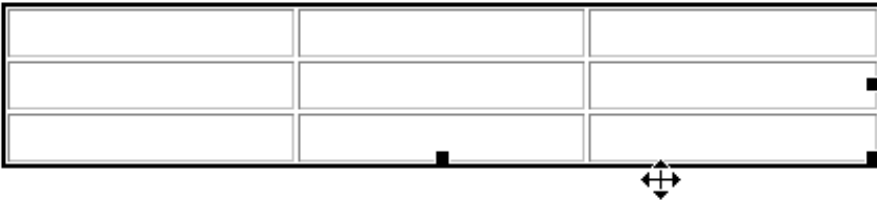


Figure 3.3: A table is selected.

- Cell Padding: this asks how many pixels you want to use to pad from a table data content to the cell’s boundaries (vertical and horizontal)
- Columns: this is the number of columns in the table
- Cell Spacing: this determines how you want to space out cells (i.e., number of pixels between boundaries of neighboring cells).
- Width: you can either enter a number as the number of pixels of the width of a table, or as a percentage of the width of the browser. If you use units of pixel, the browser display a horizontal scrollbar when the interior width is smaller than the width of the table. If you use units of percentage, the width of the table changes dynamically as you resize the browser.
- Border: the width of the borders of a table. This property controls the thickness of the boundary lines of cells, as well as the boundaries of the entire table. You can make the boundaries invisible by setting this property to zero.

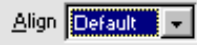
Don’t worry if you think you may need to change the formatting of a table, you can do that with ease after the table is created.

3.3 Changing Properties of a table

If you want to change the properties of a table, you need to select the table first. Move the mouse pointer close to the outside boundary of a table until it changes to an icon of four arrows. See figure 3.2 for an illustration.

Then, click to select the table. You should see the boundaries of the selected table turn black (see figure 3.3. At the same time, the Properties pane should change to display the properties of this table (see figure 3.4.

Once a table is selected, you can delete the entire table by pressing the “Del” key.

In the Properties pane, you can control even more properties than the dialog box for creating a table. For example, you can change the alignment of all the paragraphs in a table using the “Align” dropdown box (). Note that with the Properties pane, you can control the height of a table (find the label “H”), much like you can control the width (label “W”). Unless you are combining a table with tiled graphical components, it is best to leave the height control to defaults.

You can also change the background color (for the entire table) using the color tool next to “Bg Color”, or the border color using the color tool next to “Brdr Color”. Since we have not talked about graphical elements yet, we’ll defer the discussion of the background image (labeled “Bg Image”).

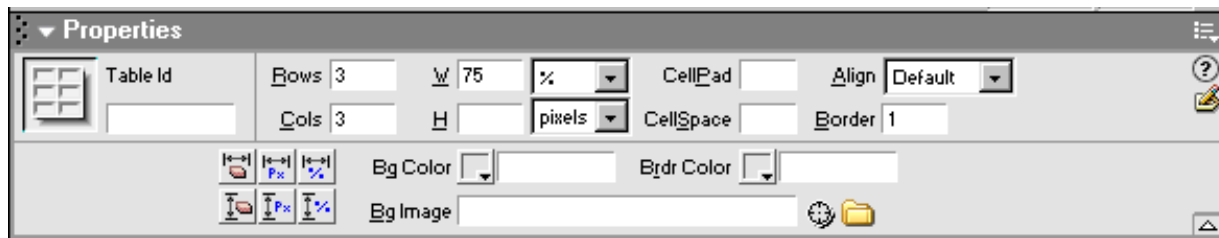


Figure 3.4: Properties pane for a table.




Figure 3.5: Row properties (in the Properties pane).

3.4 Selecting cells and changing their properties

From time to time, you may want to change properties of just one or a few cells of a table. You begin with selecting the cell(s) you wish to change. This is done by press-and-holding the left button when you are at the corner of the cells to be selected, then drag the mouse (without releasing the left button) to the opposite corner cell. You should see the selected cells with their boundaries blackened. Release the left button to finish the selection step.

Once a cell or cells is/are selected, the bottom of the Properties pane should change so you can format the cell(s). See figure 3.5 for an illustration.

If you select a whole row or a whole column, DW recognizes that and indicates that a row or column is selected. Besides some controls that we have seen with table properties, you also find some extra controls.

- : this control lets you merge the entire selection into one cell, or split it apart back to the default number of columns and rows. Use the left tool to merge, and the right tool to split.
- “Horz”: controls the horizontal alignment of contents within the cell(s)
- “Vert”: controls the vertical alignment of contents within the cell(s)
- “V” and “H”: controls the size of the selected cells. Note that you don’t have a drop down box to select “Pixel” or “Percent”. Enter a number implies it is a number of pixels, use the percent symbol (%) at the end to indicate it is a ratio (of the width of the entire table).
- “No Wrap”: checked means text contained in the selected cells do not wrap around, everything is on the same line.
- “Header” : checked means the selected cells become header cells. Header cells have default properties (boldface and centered). You typically only want the first row in a table marked as header cells, but HTML does not impose any limitations.

3.5 Exercise (do not turn in)

Using text and table controls only, create webpages that look like the following:

1		1
2	I can do this	2
3	without	3
4	using any	4
5	graphical component!	5

Chapter 4

Basic Web *Site* Design and Construction

Up to this point, we have only explored how to construct individual single web pages. On one hand, you *can* use HTML as a simplified document format and put *everything* in one document, as you would with most word processors. On the other hand, it is not easy to navigate a single HTML document that needs 50 page-down keys to get to the bottom.


Generally speaking, web browsing is best if the end user does not need to scroll the browser. There are few exceptions. For example, if the web page is a long list of items, the end user expects to scroll a page to view the contents. Examples include search results from a search engine, product listing from a eCommerce web site, transaction listing from online banking and etc.

That said, most webpages intended to present static information (as opposed to lists dynamically generated at a web site), should be contained in one page.

Of course, there is only so much information that you can fit on one page. As a result, web pages rely on “links” or “hyperlinks” so that a document can refer to another document. An end user only needs to click on a hyperlink to go to another document. This single feature of HTML gives an author the ability to *structure* contents in a way that is impossible with conventional documents.

4.1 Hyperlinks

Hyperlinks can be inserted into an HTML document easily in DW. Under the “Common” tab in the tool bar, the link

() icon allows you to insert a link any place in a document.

To create a hyperlink, click the link icon, then fill in the blanks in the dialog box. Refer to figure 4.1 for an example. In this dialog box, fill in the blanks as follows:

- **Text:** This is the text to be displayed in the document. It is typically blue (for unvisited links) and purple (for visited links) and underlined.

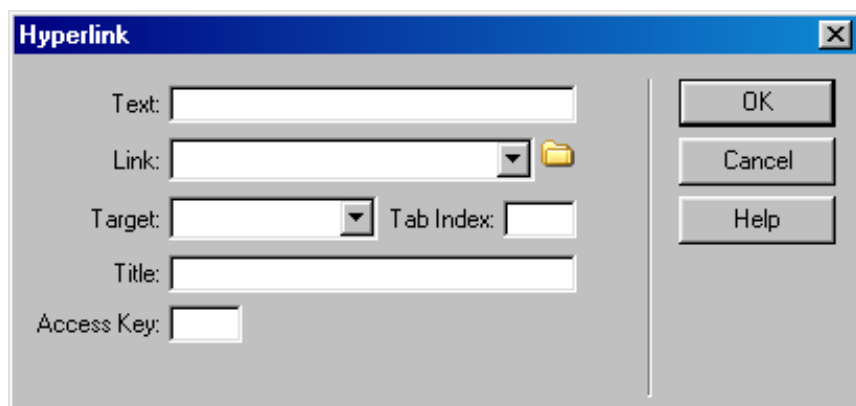


Figure 4.1: Insert Hyperlink dialog box.

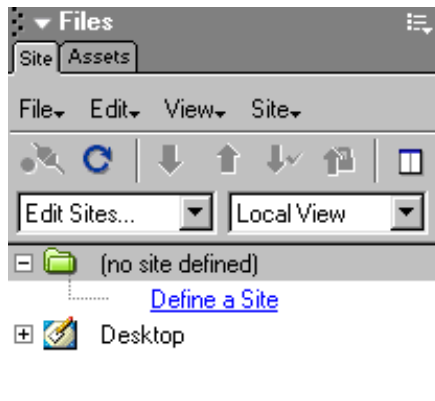


Figure 4.2: Files tab (right pane).

- **Link:** This is the destination page. You can type this in many different forms. The most common types include relative and absolute forms of an URL (universal resource location).
- **Target:** This controls how the destination will appear in a browser.
 - `_blank` opens a new browser. This is useful if you think the end user may want to read the referring document and the referred document at the same time.
 - `_parent` is only useful if you use frames. We’ll defer this until later.
 - `_self` is the default, it opens the destination document in the same browser window (as the hyperlink). As a result, as an end user clicks on a link, the original (referring) document will be replaced by the new (referred) document.
 - `_top` is only applicable for frames. We’ll defer its discussion to a later chapter.
- **Tab Index:** this controls how this link is positioned when an end user presses the tab key (to select links and fields). This feature is typically not useful for most end users who prefer to point and click. However, for users who cannot point and click, this property is very important. Use a smaller number for links that are important on a page, and a larger number for links that are relatively unimportant. You may also want to use an order that is implied by the position of the link on the page.
- **Title:** this is the title of the link. This is displayed when the mouse pointer hovers over the link. This is, generally speaking, not useful for text links. It is important for graphical links. We’ll talk about this again when we introduce graphical elements.
- **Access Key:** this allows you to specify one key to get to the link directly. Again, most end users point and click, making this property not very useful. However, for end users who can only operate from a keyboard, this property means much easier navigation (when compared to using the tab key). If you use ‘x’ for this property, it means the link will be selected (but not activated) when an end user types alt-x on the keyboard.

4.1.1 Exercise (Do not turn in)

Create a page with two hyperlinks to two of your favorite search engines. Try to specify something for all the properties of a hyperlink. Then use function key F12 to start a browser to test the page.

4.2 Site Management in DW

While you can create individual pages, then create hyperlinks among them and view the collection of pages as a website, DW offers a much better way to manage a web site.

On the right pane in DW, click the “Files” tab, you should see something similar to figure 4.2.

Click on “Define a site”, and DW starts a step-by-step dialog box to guide you through the process of creating a website. We’ll use the “Basic” tab in the dialog box.

4.2.1 Give it a name

The first step is simple, type in a name for the website. While you can use practically any name, I suggest that you use a name that only contains letters and digits (no space, no punctuations).

Click “Next” when you are done.

4.2.2 Choose a server technology

In this second step, DW asks which server technology you want to use, if any at all. You can choose “No” if you do not think you need to use any server-side technology.

If you think you may want to use server technology, choose “Yes”, then use the drop down box to pick one:

- **None:** this is obvious.
- **ASP JavaScript:** ASP stands for “Active Server Page”. JavaScript is a scripting language that *resembles* Java. To use *any* form of ASP, you will need a server that is ASP enabled. This means you will need a server using Microsoft technologies.
- **ASP VBScript:** similar to the option above, except scripts are written in an older version of Visual Basic.
- **ASP.NET C#:** similar to the option above, except scripts are written in “C#”, a Microsoft proprietary language similar to C++.
- **ASP.NET VB:** similar to the option above, except scripts are written in the current version of Visual Basic.
- **Coldfusion:** this is a proprietary language invented by Macromedia (publisher of DW). Note that unlike ASP variants of ASP, Coldfusion technology is available for platform other than Microsoft Windows.
- **JSP:** it stands for “Java Server Page”, invented and licensed by Sun Microsystems. Sun has recently relaxed the licensing of JSP technologies, so it may be “free” for you. Visit <http://www.sun.com> and read about the licensing of JSP before you decide to use it.
- **PHP MySQL:** this is actually a combination of two technologies. PHP is a CGI (common gateway interface) language that is free and available for almost any modern platforms. MySQL is a database server that is free and also available for practically all modern platforms. As a result, the combination of PHP and MySQL is quite universal, versatile and free.

The server that will host your web sites will be PHP enabled. So you can choose PHP MySQL.

One warning: web pages written in PHP will not be previewable until you upload the files to a PHP enabled machine, or have PHP enabled on the authoring machine.

Click “Next” after you have made your selection

4.2.3 Development method

This page asks you how you want to develop your website. There are four options:

- “Edit and test locally”: this means your authoring computer is a test platform. Whatever scripting option you have chosen in the previous step, it must be available on your authoring machine.
- “Edit locally, then upload to remote testing server”: this is the default setting. In this approach, your authoring machine only needs to run DW. However, to test your web site, you need to upload (transfer) the files to a testing server that has the necessary technologies. This approach is nice because you can use a minimally configured machine, and yet be able to edit pages most of the time.
- “Edit directly on remote testing server using local network”: this approach works if the hosting server can be accessed via a LAN (local area network). This approach assumes that you have Windows sharing available on the authoring machine and on the hosting server. Note that non-Windows platforms can utilize Windows sharing, making it possible to use this approach on Linux servers as well as Windows servers. With this approach, however, you need to have LAN access to the server to even edit the pages. It is best suited for commercial development where the testing server is local to the authoring machines.

- “Edit directly on remote testing server using FTP or RDS”: FTP stands for “File Transfer Protocol”, while RDS stands for “Remote Data Services”. This option makes it possible to transfer files between an authoring machine and a testing server via the internet (and not just a LAN). In other words, the authoring machine needs not be located near the testing server anymore, they can be half a world apart. Like the previous option, however, this approach requires network (internet) connectivity to even edit web pages because nothing is stored locally on the authoring machine.

So, what should you do? I recommend that you run your own server, then you can select the first option. The following subsection instructs you how to make a web server out of a Windows machine for free.

Be sure to take note where the files are stored. If you plan to use a USB jump drive to store your projects, you may want to consider changing the path to the jump drive. DW creates the folder if one does not exist already.

Run your own server

The main advantage of running your own server on the authoring machine is that you can edit *and* test the website on the same machine without any network access during development. I can even download the necessary files and burn you a CD so you don’t have to download the necessary software package.

Yes, you can do this legally without paying a dime.

If you do not have WinZIP or similar software installed, you can use 7-Zip. Follow these instructions to download and install 7-Zip:

- Go to <http://www.7-zip.org>
- Click “Download Now” to download the installer, click “Open” to get it installed
- After the installation, click “Start” button, then “Programs”, then “7-Zip”, then “7-Zip File Manager”
- In 7-Zip, click the “Tools” menu, then select “Options”
- In the “System” tab, check all the files that you want 7-Zip to open. Make sure “zip” is checked!
- Click OK, then close 7-Zip.


First, you need to download and install both Xitami and PHP. These packages are available from <http://www.xitami.org>. and <http://www.php.net>, respectively. However, I have already downloaded and setup both packages. There were some really obscure settings that were not documented at either web sites. You can save yourself a lot of trouble just by downloading a zipped file from my website at http://www.drtak.org/teaches/ARC/cisc306_dw/download/myserver.zip.

Click on the link to download, click “Open” to open the file with 7-Zip or other ZIP programs. Select both folders in the ZIP file (control-A), then click “Extract” and then type the following as the destination

C:\

to install both Xitami and PHP. Yes, we are simply unzipping the files to two folders on the C: drive. You need to customize Xitami just a little bit. I advise that you first create a web site in DW, then come back here to set up Xitami.

- Configure Xitami to point to your document folder:
 - Right click on the Xitami icon in the notification area and select Admin
 - Type the password for Admin (defaulted to “help” if you download `myserver.zip` from my download area), then click ”Ok”
 - In the main Configuration screen, you may want to change the directories for your set up:
 - * **Main HTML directory:** the default is a folder in the Xitami directory. You may want to change this to one that has your homework assignment. DW defaults your projects to `C:\My Documents\projname` (replace `projname` with your actual project name).
 - * **Main CGI directory:** same as the previous item
 - * **CGI URLs start with:** leave the default as is
 - Click “Save” (at the top left corner)

- Click “Restart” (at the top right corner)
- Wait a few seconds, then type in the URL address as follows:
`http://127.0.0.1`
- You should see the default Xitami start up screen.
- Start up Xitami if it is not started yet:
 - Use File Explorer to navigate to where Xitami was installed (C:\Xitami-25)
 - Select the “app” folder
 - Double-click “xigui32”
 - Nothing seems to happen, but you should see a new icon in the notification area (lower right corner) like 

This is it! You have got yourself a small web server that handles PHP scripts.

4.2.4 Testing Files URL

The next step requires you to enter an URL to test the website. If you use the local method (as suggested), you should type the following:

`http://localhost/`

in the text box for the URL. Don’t bother clicking the “Test URL” button unless you have already set up Xitami (or other servers).

4.2.5 Remote Server

The following step asks whether you want to use a remote server. If you use the local approach, you do not need to set up a remote server for submitting your project. On the other hand, if you plan to run your website (you need to find someone to host it), you may want to select “yes”.

If you select “no”, there is no further question for this step. Otherwise, you need to indicate how to connect to the remote server and which folder on the server to store the files.

If you select “yes” to remote server, you also need to indicate whether you’ll be sharing the files and use “check in and check out” methods. Unless you use some form of version control mechanism, answer “no”.

4.2.6 Done!

The next step lets you preview the settings. Remember, you can still change most of the properties using the Advanced Tab.

4.3 Working with a web site in DW

In the “Files” tab on the right pane, select the “Site” tab to get a structural view of the web site. If you start from scratch, there is no file associated.

This is a good time to set up Xitami to point to this folder containing your web site. Review section 4.2.3 for this purpose.

Most web sites start with a file called `index.htm`. This document is the first page that pops up when someone visit a web site. Right click on the site folder, select “New File” to create this first file (be sure to change the default name to `index.htm`). Once this file is created, double click it to edit it.

Create a simple page, then save it. Next, make sure Xitami is running. If the Xitami icon turns red, right click and select “Running” again.

Then, open Internet Explorer, and change the URL address to

`http://localhost`

You should see your index page!

Chapter 5

Graphics and Images

Websites with just text can be well structured and informative. However, there are times when a single image can replace an infinite amount of text. HTML allows the inclusion of images onto an HTML document. DW has extensive WYSIWYG support for images as well.

This chapter deals with the basics of including pictures in an HTML document.

5.1 Basics

While HTML allows the inclusion of images, the images are actually never a part of the HTML document. Remember, an HTML document is merely a plain text document. Instead, images are displayed in a browser because an HTML document has a “link” to it.

A “link” in this case is a reference. In other words, within an HTML document, it can refer to image files that are placed elsewhere in the same folder, on the same server, or practically anywhere the network reaches.

Because your images are external to HTML documents, it is important that you organize the image files in a structured manner. Otherwise, you’ll lose track of the images very soon, and it’ll be difficult to maintain your website. This is why we cover “site management” first.

Except for images that do not belong to you, you can organize pictures in the following ways:

- in the same folder where the referring HTML document is
- in a subfolder where the referring HTML document is
- in a site-wide folder

5.1.1 Local to documents

The nice thing about this approach is that references to the images are easy to read. However, you are mixing HTML files and image files in the same folder, making it slightly more difficult to find a particular file. If you have relatively few image files and HTML documents in a folder, this option is easy to set up and use.

5.1.2 Subfolder among documents

This approach is best if you have relatively few HTML documents, but a whole lot of images. This way, you don’t have to find HTML documents among many image files.

Generally speaking, this approach results in folders that are more “neat”, but links to images require the name of the image folder.

5.1.3 Site-wide image folder

This approach is usually combined with one of the above. If you have any logo image or other images that are “global” to the entire site, you can place those image files in a site-wide folder.

5.2 Finding images and using them

Let's start off with some references online. <http://www.patents.com/weblaw.sht> (Copyright 1996-1999 Oppedahl & Larson) has some pointers that are officially not legal advice. However, you can still get some ideas of what is permitted and what is not.

Another useful page is <http://people.bu.edu/rthomas/copyright.htm> from Boston University.

On the flip side, others can also link to or copy from your images. You can visit <http://www.2learn.ca/copyright/images.html> for more information. You should write some copyright statements for your website to make sure you can limit, at least legally, how others can use your images.

What is the bottomline? If you own the images (i.e., they are your pictures, paintings and etc.), you can safely include the images as a part of your documents. If you do not own the images, it is safer to use a hyperlink to the origin. Yes, a hyperlink requires the reader to click first to get to the image, but you can use the “_blank” option so the image pops up a new browser window. The hyperlink approach, while still may not be 100% safe, is much better than copying images or including images in your document without permission.

Of course, you can also contact an originating website and actually get a written permission/licence to include images. This may be free in some cases, especially if your website is non-profit or educational, and that you make the original copyright notice obvious.

5.3 Creating your own images

As mentioned above, although it is usually easier to include or copy images from existing sources, the legal implications can be overwhelming. Instead of risking legal actions against you over a few image files, it is often safer to create your own images for your website.

5.3.1 You can still get into trouble!

If you scan a page from a comic book, or capture a frame from DVD or VHS, you can still get into serious trouble when you publish the images. This is because most books, DVDs and video tapes have copyright notices! As a general rule, assume that you *cannot* use any scanned or captured materials.

5.3.2 Digitizing photos

There are now quite a few ways to digitize photos originally taken with film.

Scan the print

This is one of the most common methods to digitize a picture. You take a print, and scan it with a regular scanner.

Compared to other approaches, this is cheap and often produces images of sufficient quality for web publication. A low-end scanner is about US\$50 (or less!), and it still has at least 300dpi (effective resolution, not advertised resolution). This means a 4R (4x6) picture results in a 1200x1800 image. You'll have to scale it down to at the most 600x900 or 300x450 for a webpage.

If you start with prints, this method is often the most cost-effective way to get your photos digitized.

Scan the slide/negative with a flatbed scanner

Many of the newer flatbed scanners (for scanning prints and other reflective material) can also scan negatives, slides and other transparent material. A reasonable beginner set up costs about US\$150 to US\$200. However, you get the abilities to scan both prints and also negatives/slides.

The ability to scan from slides is important if you do not have any prints. It is quite expensive to get prints from slides. Even the ability to scan from negatives has its value. A frame of negative typically contains more information than can be presented on print paper. As a result, scanning negatives can result in better quality. This gained quality is usually the extension of tonal range for post-scanning manipulation (like contrast adjustment and etc.)

A low-end negative/slide capable scanner has native resolutions of 1600 dpi or 2400dpi. This may sound like a lot, but the effective resolution is often down to 800 dpi. A 35mm frame, therefore, scans to an 800x1200 image. You can then rescale to 400x600 for web publication purposes.

The bottom line is that you still have enough resolution.

Scan the slide/negative with a film scanner

A film scanner is a dedicated device just for slides/negatives. You cannot use it to scan reflective material. A low-end film scanner costs US\$300 or so, with a resolution of 2800dpi or more.

The main difference between a dedicated film scanner and a flatbed scanner is effective resolution. A dedicated film scanner has its effective resolution very close to the optical resolution. This is mainly because there is no glass in between the film and the sensor. There are also other factors that make dedicated film scanners superior to flatbed scanners.

Do you need the extra edge? It depends on whether you need to do extreme cropping or other forms of image manipulation. For example, if you intend to do a lot of post processing in terms of color and tone adjustment, or to add special effects, a high resolution scan from a film scanner gives you a lot more information to work with.


Dedicated film scanners are often required for digital photo printing because modern printers can print a lot of details. However, for web publication, the resolution (both pixel wise and color wise) is limited by monitors. As a result, the added value of a dedicated film scanner is not justified for just web authoring purposes.

5.3.3 Post digitization processing

It is beyond the scope of this class to discuss photo editing. If you are interested to “play with” it, I strongly recommend the GIMP. You can download this application for free at <http://www2.arnes.si/sopjsimo/gimp/stable.html> (following the instructions on the page). Alternatively, you can give me a CD-R and I’ll give you a copy of the GIMP and much more of free software.

What you can do with image manipulation is virtually unlimited.

5.4 Including pictures with DW

This is easy! Make sure the tool bar is at the “Common” tab, then use the image tool .

Once you click on the tool button, dialog box shows up. The following is a description of each control in the dialog box:

- **Select File Name From:** Unless you are planning to use scripting, use “File system”. The other option, “Data Sources” is useful if the image is going to be retrieved from a database via scripting. The rest of this list assumes you are using “File system” from this option.
- **Look in:** This is a navigation area that allows you quickly locate the image file from your folders. This means that this property is not useful if the image is at another website. When you click on an image, the “Image Preview” area should display a shrunk version of the image. If you use the GUI approach, you can let DW fill in the properties **File name**.
- **URL:** This is the complete URL path to the image file. If the image you wish to display is not local to your folders, you have to enter the URL here.
- **Relative To:** You have two options. Select “Document” if the image is in the same folder as the document, or if it is in a subfolder under the folder containing the document. Select “Site Root” if the image is at a folder that is not under the document’s own folder.

Click OK when you are all done.

5.5 Properties of an image

After you insert a picture into an HTML document, the real fun begins!

Click on an image, you should see the properties pane updated for images.

Let’s go through some of these properties:

- at the upper left corner, under the image and its size, you can enter a name for the image. This is only useful if you need to refer to the image in a script.

- **W**: the displayed width. If there is no unit, it is in pixels. You can use other units: `in`, `mm` and `cm` are common units. Yes, the browser can rescale an image on the fly!
- **H**: the displayed height. Similar to **W**.
- **Src**: where the image should be found. Drag and drop the “target sight” icon to a file, or use the folder icon to type in the location manually.
- **Link**: an image can serve as a hyperlink so a viewer can click on it. This property states where to jump to if someone clicks on the icon. Again, you can drag-and-drop the “target sight” or use the folder to enter the URL manually.
- **Alt**: this is text that appears when the mouse pointer hovers over the image. Note that for accessibility, you should *always* enter text that describes the content of the image for users who are visually impaired.
- **Edit** this is just a button to quickly launch whatever application is associated with image files. Unless you have an editor (such as the GIMP) associated with image file types, this button is fairly useless.
- **Reset Size** If you want to restore the image to its original size (based on pixels), click this button.
- **Map** this tool is very cool if the image is a link. Instead of using the entire image as a link, you can define a subarea in the image as a hotspot (that responds to mouse clicks). Choose either the rectangle, circle or polygon tool to enter the hotspot editing mode.

Note that a hotspot has its own link, target and alt text specifications. You can also define multiple hot spots on the same image.

You can visualize hot spots are spots on top of the image. This means if the image itself is a hyperlink, the hot spots are still accessible because they are on top of the image. Do not have hotspots overlapping each other (unless they have the same link) because the order of overlapping is not obvious.

- **V Space** and **H Space**: these two controls how much space to leave around the image.
- **Target**: just like with text hyperlinks, you can choose how to follow a link. Use “`_blank`” to open a new one, or “`_self`” to open in the existing browser.
- **Low Src**: this is similar to **Src**, but it is the specification of a lower resolution version of the image. If specified, this image is loaded before **Src** is loaded.
- **Border**: this property adds a border to the image. The default is 0. Note that if an image is a hyperlink, the border (if non-zero) is colored to indicate the link. By default, it’ll be blue if unvisited, and purple if visited.
- **Align**: You can control how the image is aligned with text on the same line. Top means top-to-top-of-whatever, middle means middle-to-baseline, and bottom means bottom-to-bottom-of-whatever. Absolute middle means middle-to-middle-of-text, absolute bottom means bottom-to-baseline-of-text. Left and Right means the image should be on the left or right, with text wrapped on the other side.

Chapter 6

Phase 2 of Your Project

Since March 18 is the last day we meet, we need to speed up the project a little.

In this phase of your project, we need to collect all relevant information, and start to structure the web site using hyperlinks.

6.1 Planning the web site

6.1.1 The Process

It may be obvious at first how you want to organize your website. As with any projects, start with your objectives. What is the main purpose of your website? Who is your target audience? What does you audience want to find on your website?

This subsection describes what you can do plan your website.

Your Objectives

What is the objectives of your website? For example, the main purpose of drtak.org is to distribute classnotes, sample programs, lecture recordings and other supportive material for courses that I teach. My target audience is, quite naturally, current, past and future students. My scope is somewhat limited to the courses that I have taught and courses that I will teach in the near future.

You can define your objects, target audience and scope the same way. Having a clear idea of who is browsing your web for what purpose expecting what helps the following steps.

Think from the Browser's Perspectives

We are all eager to put stuff on the web, but we need to think from the view of a potential viewer first.

In other words, what does a student want from my website? Check the classnotes? Download lecture recordings? What is the best way to help a student narrow down to the target page?

What type of information does your audience expect from your website?

Brainstorm

At this point, just toss everything into the salad bowl. Don't try to sort them out yet. Just brainstorm and jot down the kind of information and data that you think you should include in your website.

It is best to use a word processor to jot down your ideas. DW is not a bad tool for this purpose. This is because you will need to reorganize your ideas a lot in the next step. Using a word processor or DW makes it much easier.

Note that it is *impossible* to include everything in the first round. Just move on when you think there is enough material to start.

Categorize

In this step, we are not structuring yet. All we are doing is to group related ideas and material into chunks.

For example, if you plan to create a website to spread the word against DUI and educate the public, you may want to the following. Put all the data and statistics into one group, all the legal consequences into one group, all the real-life cases into one group, and all the additional resources into another group.

The key to categorize is to identify like or related items from the brainstorm session.

At this point, it is very likely that you realize that under a particular category, there are some additional topics to add. By all means, add the topics now.

Structure

Once you get “chunks” of topics from the previous step, it is time to structure these chunks so that a viewer can find each piece in each chunk easily.

Structuring is really categorizing, but this time categorizing chunks from the previous step. In other words, see each chunk as a single unit, what other chunks are closely related it? For example, in the against-DUI website, you can group the entire chunk of legal consequences and real-life cases into one bigger chunk, and then group the legal resources, support groups and other DUI victim resources together.

For very complex websites, you can do structuring repeatedly, relating bigger and bigger units and forming even bigger units.

In this step, it helps to use a multi-level list to track which chunk contains which other chunk. The indentation levels of each point visually displays how chunks are related.

Review, Revise and Repeat

For complex sites, you may need a few cycles to refine the structure. Remember to review from the perspective of a viewer, and see if it is easy for a viewer to find information. Think of *why* a viewer is browsing your site to begin with. With the DUI website, can an educator quickly get to the section that contains information that is useful for education? Can a victim or family of a victim quickly references to support groups? Can a person which drinking problems locate resources to control his/her drinking?

It is critical that you are satisfied with the structure *before* authoring any pages. While DW is a powerful WYSIWYG tool, it is still very cumbersome to perform structural changes after a website is created.

6.1.2 Presentation

Once the website structure is determined, you can think of ways to present the choices (hyperlinks). Unless there is a special reason, you should keep all options on the same page (without scrolling).

You have quite a few options.

Essay

You can write a short essay for each page, and insert hyperlinks to refer to other pages as necessary.

This approach gives viewers a “classic” feel. Due to the use of text only, webpages of this kind load very quickly. Needless to mention, webpages of this kind is also inherently friendly to viewers/browsers who have accessibility needs. A viewer can also quickly and easily change the size of text to his/her preference.

Depending on your target audience, this form of presentation may not be bad. However, *in general*, the web surfing crowd wants **instant** information delivery. Having to read through paragraphs is a bit time consuming.

A main drawback of this approach is that your hyperlinks (text-based) tend not to be very big compared to the rest of the page. This makes it more difficult for viewers to find the click on the links.

Lists

Instead of using complete sentences and paragraphs, you can consider using an outline mode using hierarchical lists (unordered or ordered). The main advantage of this approach is that it is easier for viewers to find the hyperlinks, and the one-link-per-line convention makes it easier for viewer to parse the content.

Using hierarchical lists also give you the ability to structure links within the same page. In other words, your website can be very organized and structured this way.

Lists are plain text in HTML. Webpages using hierarchical lists also load very quickly. On the other hand, since lists are still text based, your hyperlinks are still kind of small, although they are now easier to locate.

Button Bar

This is a concept borrowed from common Windows applications. With a button bar, you can use bitmaps for hyperlinks. This is an improvement over text because now you can make your buttons (links) very big and easy to spot. In addition, you can specify the displayed size of images, making it more convenient to control the look of a webpage in a browser.

On the negative side, images are slower to download. The ability to control image size is also a drawback because now a viewer cannot easily change the size of the buttons.

Link Column

I don't know the origin of this "convention", but it seems that most viewers are comfortable with or even expect a column on the left of a page listing most commonly used hyperlinks.

This design is useful if the main area of a page needs to display something else, as in most cases. The combination of a button bar on top and a link column on the left is a popular "template" (not related to frames in HTML) for many websites. This way, the main content of the page can be independent of the template, giving the site a consistent look.

While some sites use fancy buttons in the link column, most websites settle for text-based links in this column. You can reduce the font size and use hierarchical lists in the link column to keep all the advantages of using lists of links.

Button/Link Matrix

Some pages use a table of buttons for navigation. These pages tend to be picture oriented so that you select the link not by reading, but by viewing and interpreting pictures. This type of design is appropriate for, say, a car manufacturer.

You can replace buttons with just text hyperlinks. This approach, however, loses the eye-candy effect. Compared to a list of hyperlinks, a table of hyperlinks has the advantage of being able to fit more hyperlinks into the same area. You can also organize by rows and columns (instead of hierarchically).

6.2 What to turn in?

For phase 2, give me a structured list that represents the structure of your website. Give me a brief description for each item. For example:

- root: This is the home page, display a picture of a crashed car, an ambulance, sobbing friends or relatives
 - Site statements:
 - * Copyright statement: all pictures have copyright rights. Tell the viewer what restrictions apply.
 - * Mission statement: state the purpose of this website.
 - * Who creates this site: why the author creates this website. Contact info.
 - What is the problem?
 - * Statistics: use both text and charts. Show the proportions of accidents caused by DUI.
 - * Real stories: sublinks to individual stories.
 - Victim help
 - * Legal help references: how DUI cases are special, links to government documents, links to insurance company documents, links to court cases.
 - * Medical resources: in the case of uninsured DUI motorists, how to get sufficient medical help.
 - * Support groups: links to support groups, contact info to add support groups to the list
 - Help for drinkers: show a picture of a liquor bottle, unopened, being thrown into the trash can.
 - * Treatments: provide links to various treatments to stop drinking.
 - * Support groups: list support groups, contact info to add a group to the list, list by state
 - * Ex-drinkers speak up: list of links to feedback from ex-drinkers.

In addition, tell me how you plan to present the hyperlinks. If you plan to use the feel-and-look for all pages, describe the template. If you decide to use different methods, tell me what types of design you are considering.

6.3 How to turn this in

Send the document as an attachment by email to <mailto:tauyeung@drtak.org> with a subject line of
CISC306 Phase 2 by *your name*
The due date of phase 2 is 2/23/2004 (Monday).

Chapter 7

Common Layout and Theme

Although it is possible to layout each individual page in a site individually, it is far more effective to develop a “template”, then use the same template for all the pages, or most of the pages.

It is possible, and relatively easy, to do this in HTML. This is mostly due to the availability of Cascading Style Sheets (CSSes). This chapter covers some of the basics of template design and use.

7.1 Design a Template

While CSS helps you with certain aspects, it does not layout a overall structure of a webpage.

7.1.1 Saving to Your Own Template File

Without using frames, you can still design a general layout (using tables and images) that is common to all pages, and save it to an HTML file. This way, you can begin a new page using the template file to save yourself some work.

This approach works, but it is not flexible. If you decide to change the feel-and-look of the website, you’ll have to manually change all the individual pages!

DW supports a much more powerful feature called “Templates”. A template is not just an HTML document. It is an internal format specific to DW. In other words, other tools probably cannot use templates created by DW. Nonetheless, such templates are useful because they can propagate changes automatically to pages in a site automatically.

7.1.2 The Layout View

Click on the “Layout” tab and switch to the Layout View. In the layout view, you can draw cells of a table and move them around as free floating objects.

Select the Layout Table tool and draw an layout table. Within a layout table, you can add layout cells using the layout cell tool. Layout cells within a table are somewhat free floating, although there are restrictions to the height of a layout cell.

After you click inside a layout cell, you can immediately put contents into it. You can click near the border of a layout cell or a layout table to view and change its properties.

With the layout view, you can quickly and visually design the layout of your webpages. Leave a large cell for the contents.

7.1.3 Templates

DW supports a non-HTML feature called “templates”. This is a powerful concept, since DW takes on the responsibilities to make every page of a site look the same structurally.

To do this, after you create a file with the proper layout, simply use the “Save As Template” menu item from the “File” menu. This adds the template to your “Assets” under the “File” tab in the Design pane to the right of the screen. This method is useful if you already have created some webpages, and want to use them as templates for future pages.

You can double click on the template asset (in “Files” tab in the “Design” pane) to edit it.

Otherwise, if you are just beginning, you can use “File — New”, then select “Template Page” (under the “General” tab). Be sure to select the right Template Page option based on your scripting language. Most students in this class should choose “HTML Template” because there is no scripting for most.

A template is much like a normal HTML page. You can put any content into it, set the background image and etc.

Extra features for templates are found under the “Template” tab on the insert bar. Put the cursor in a cell that changes for each webpage, then click the insert region tool (the pen on folder icon) to make that cell editable for pages created from this template. Note that unless you mark a cell editable by inserting an “edit region”, pages created from a template cannot be modified!

You can create as many edit regions as you want to. You can even create edit regions outside layout cells.

To remove an editable region, click on the blue tab of the region, (look at the Properties pane to make sure you have selected the region), then use “Modify — Templates — Remove Template Markup” to remove an editable region.

Once you save a template file, you can now create new pages based on this template. Note that you can define multiple templates for each website.

If you already have pages created from a template, saving changes to the template will automatically pop up a dialog box to ask you to confirm the update of all affected webpages. In general, click “Update”. If you remove editable regions in a template, DW gives you a choice to put the content that used to be in the remove editable region to elsewhere in the document.

To create a file from a template, choose “File — New”, use the “Templates” tab to select the template you wish to use. Make sure “Update Page when Template Changes” is checked so that DW will do as much as it can to keep this page updated when the layout of the template changes.

Files created from templates have “Edit Regions” (blue tab and blue borders) that you can insert contents. You can *only* edit in these edit regions.

7.2 Styles and Cascading Style Sheets

The cascading stylesheet (CSS) standard supports styles. A style is, essentially, a user defined formatting element. A style is not only limited to text formatting. It is also applicable to many other elements in HTML.

The cool part of CSS is that you can change a single file and alter the entire web site. While CSS does not perform layout functions as DW templates, it does affect how elements appear. In other words, CSS is a complement to DW templates because CSS controls non-layout elements, while DW templates control layout elements.

7.2.1 Where to Define Styles?

Styles can be defined internal to an HTML document or external. Most sites use the external method so that multiple documents can link to the same style definitions.

7.2.2 Handling Styles

Click on the “CSS Styles” tab of the “Design” pane. If a document has no applicable styles, the list is empty when the “Apply Styles” radio button is pushed. Click on the “Edit Styles” radio button to handle styles.

For a document with no styles defined, the list displays “no styles defined” under the document. This is fine. Right click on the document name, then select “Edit”.

The next dialog box lets you select how you want to get styles into this document. Click the “Link” button to link an existing stylesheet document (with an extension of `.css`) to this document. Note that if the linked stylesheet file has any redefinition of HTML tags, the linking document inherits all those redefinitions automatically.

You can also click “New” if there are no existing stylesheets, or that you want to define a new stylesheet, or that you want the style to be specific only to the document.

After you click “New”, a new dialog box pops up. It asks you about the name of the style. Here, select the “Type” first! DW should have put the “Type” radio buttons on top of Name/Tag. You can refine existing HTML tags (don’t do this unless you know the global effects of it), or you can define your own styles.

Next, you can choose where to put this definition. Unless a style needs to be specific to a document, it is usually better to define it in a stylesheet file. If a document already links to at least one stylesheet file, you can specify one of these files for storing the new definition. You can also choose to create a new stylesheet file on-the-fly to store the style definition.

After you click “OK”, the following screen lets you define your style. A single style can control many categories of styles. Select and specify the appropriate categories. “Type” is for text formatting, which is the most commonly used category. You can specify your own style of text formatting. “Background” is for background control. “Border” is for specifying the border of cells and tables. “List” is for specifying properties of a list. Refer to the online help for a more thorough description of each category.

After you specify a style, you can click OK to add the definition (without applying it), or click “Apply” to apply the style to whatever is selected.

7.2.3 Selecting a Style

Once user-defined styles are available, selecting the “Apply Styles” button in the “CSS Styles” tab of the “Design” tab should let you apply styles to existing elements. Simply click on a user-defined style, and the selected section will use the user-defined style.

Chapter 8

Uploading Files

Once you are satisfied with the web site, you can upload the files to a file server. This chapter discusses the details of the upload process.

8.1 Concepts

When you use Dreamweaver, you are editing files that are local to the authoring machine. Although Dreamweaver can directly “edit” files on a remote machine, it is more practical to keep files local for most authors. It only makes sense to edit remote files directly when a server (see below) is connected to the authoring machine via a fast local area network.

Furthermore, keeping files local also has the advantage of being able to work on websites without any internet connection.

8.1.1 Web Server

A “web server” is formally called an HTTP (hypertext transport protocol) server. It is a computer that is connected to the internet, and it must run an HTTP server software.

When you connect to a web site, you connect to a web server. A web server is responsible of interpreting your request (URL) and figure out how to return with the requested contents. Although a web server can be set up to generate web pages on-the-fly using CGI (common gateway interface) programs, we will limit to static files in this class.

8.1.2 Account

In order to place your files on a web server, you must have an account. An account on a web server allows you to transfer files to the web server, as well as to delete and list files already on the web server. An account has a name and a password. We’ll discuss your account on our web server in the next section.

8.1.3 FTP

FTP (file transfer protocol) is an insecure method to send files between computers. It is insecure because a sniffer can, quite easily, sniff out usernames and passwords. Fortunately, this is not a problem at ARC because all the connections are switched, rendering sniffers useless.

We’ll discuss how to set up Dreamweaver so that it knows how to use FTP to synchronize your local files with a web server.

8.2 Your Web Server Account

An account is already set up for you so that you can put your files onto a web server.

8.2.1 Information

The name of the computer that hosts your file is

```
cis.arc.losrios.edu
```

Your account name (username) is your student ID. If you cannot remember it, you can go to <http://cis.arc.losrios.edu/cisc306.htm> and hover the mouse pointer over your name. Your student ID should appear on the status bar.

Your password is a string in the following format: MMDDYYYY. This is your birthdate that you used for enrolling the class. For example, if your birthdate is May 22nd, 1963, the password is

```
05221963
```

8.2.2 Setting up Dreamweaver

You can set up Dreamweaver so that you only need to enter information about FTP once.

Start up your project as usual. If you are doing this at the lab, you have to use “define a site” in the “Site” tab under “Files” on the right hand side.

If you already have the site set up, go to “Site — Edit Sites...”, click on the site that you want to edit, then click “Edit”. This should restart the same dialog sequence to fill in information about your web site.

When you are asked about “Sharing Files”, select “FTP” in the dropdown box for “How do you connect to your remote server?” Next, type in

```
cis.arc.losrios.edu
```

for “What is the hostname or FTP address of your Web server?”

Next, type

```
public_html/
```

for “What folder on the server do you want to store your files in?”

Type in your username next, note that you need to use the lower-case “w” letter as a prefix to your student number. In other words, if your student ID is 0024251, enter the following

```
w0024251
```

to answer “What is your FTP Login?”

Finally, type in the password. If your birthdate is March 12, 1963, type in the following:

```
03121963
```

Click “Test Connection” to see if Dreamweaver can connect. If you cannot connect, let me know ASAP! If the connection is successful, you should see a pop-up box indicating that the connection is successful.

Click “Next” to move on to the next dialog.

In the following dialog, titled “Sharing Files, Part 2”, say no to the question.

You are now done. Once the dialog box closes, click “Done”.

8.2.3 Synchronizing Your Files

Dreamweaver is smart enough to know that some times you have files more recent remotely, and most of the time your local files are more recent. To synchronize files on the server and your local system, right-click on the project, then select “Synchronize...” from the menu.

This should pop up a new dialog box. In the first drop-down box, select “Entire ‘your sitename’ Site”. In the second drop-down box, select “Put newer files to remote”. Check the checkbox captioned “Delete remote files not on local drive”. Then, click “Preview” so that Dreamweaver can let you know what it plans to do next.

In the following screen “Put” in the “Action” column means the file is going to be transmitted to the server. “Get” means the file is going to be transmitted to the local system. “Delete” means a file is going to be deleted. Most of the time, you only have items to “Put” to the remote server. It is okay to delete “index.html” on the remote system because it is a default root page that has precedence over your own “index.htm” file. If all looks good, click “OK” to continue.

If the file transfer is successful, click “Close” to close the dialog box.

In order to make your own root page the start up page, you can now delete “index.html” (don’t delete the wrong one!) from your local system. The next time you synchronize with the remote server, the remote one will be deleted as well.

8.3 Viewing the Site

Once you have synchronized your files, you can see how everyone else on the internet will see your web site. Start a web browser (Internet Explorer), then enter the following as the URL:

```
http://cis.arc.losrios.edu/~w0023042
```

Of course, this URL only works if your student ID is 0023042. Change that to match your own student ID to view your own website.