



Part III

Adding Advanced Design Features

IN THIS PART

Chapter 10

Working with Layouts

Chapter 11

Using Behaviors

Chapter 12

Setting Up Tables

Chapter 13

Interactive Forms

Chapter 14

Creating Lists

Chapter 15

Using Frames and Framesets

Chapter 16

Powering Ajax Pages with Spry

Chapter 17

Working with JavaScript Frameworks



Working with Layouts

IN THIS CHAPTER

Understanding how divs and AP elements work in Dreamweaver

Inserting `<div>` tags

Dreamweaver Technique: Applying a CSS Layout

Working with fluid grids for multiple screens

Modifying AP elements: Resizing, moving, and altering properties

Dreamweaver Technique: Inserting an AP Element

Creating interactive AP elements with Dreamweaver behaviors

For many years, page designers have taken for granted the capability to place text and graphics anywhere on a printed page—even enabling graphics, type, and other elements to bleed off a page. This flexibility in design has eluded web designers until recently. Lack of absolute control over layout has been a high price to pay for the universality of HTML, which makes any web page viewable by any system, regardless of the computer or the screen resolution.

Now, however, the integration of absolutely positioned elements (AP elements) within the Cascading Style Sheets specification has brought both relative and absolute positioning to the web. Page designers with a yen for more control welcome the precision offered with Cascading Style Sheets positioning. Positioning styles are frequently applied to `<div>` tags, which are used to separate a page into different areas or divisions.

The increase in the number of screen sizes found in mobile devices has brought another challenge to the web designer's door. How is a layout to be seen optimally under the wide range of conditions? One technique with a good deal of support among modern designers is called *fluid grids*. The fluid grid layout combines CSS media queries with a percentage-based layout for the ultimate in flexibility. As you'll see in this chapter, Dreamweaver makes it possible to build a fluid grid layout that adapts to phone, tablet, and desktop screens.

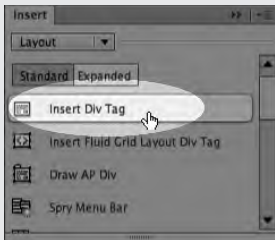
This chapter explores every aspect of how various layout elements work in web pages. With the fundamentals under your belt, you learn how to create, modify, and populate `<div>` tags, AP elements, and fluid grid layouts in your designs.



If you're already familiar with `<div>` tags and absolutely positioned elements, here's a quick overview of how to add each to the page.

To insert a `<div>` tag to your page, follow these steps:

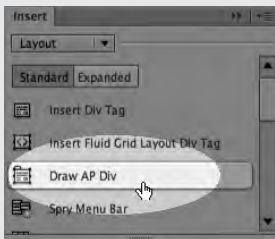
1. From the Common category of the Insert panel, click Insert Div Tag.



2. When the Insert Div Tag dialog box opens, choose where you'd like the tag to be inserted: at the current cursor position or before—or after—a particular tag.
3. Choose an already-defined class or ID to assign to the `<div>` tag. Alternatively, you can define a new CSS rule and use it.
4. Click OK to add the `<div>` tag and placeholder text to the page.

To draw an absolutely positioned element on the page, do this:

1. From the Layout category of the Insert panel, click Draw AP Div.



2. Drag out a rectangle the approximate size and shape of the desired AP div.
3. To reposition the AP div, drag its selection handle in the top-left corner to a new screen location.
4. For precise placement or dimensions, change the values in the Property inspector.

Divs and AP Elements 101

When the World Wide Web first made its debut in 1989, few people were concerned about the aesthetic layout of a page. In fact, because the web was a descendant of Standard Generalized Markup Language (SGML)—a multiplatform text document and information markup specification—layout was trivialized. Content and the capability to use hypertext to jump from one page to another were emphasized. After the first graphical web-browser software (Mosaic) was released, it quickly became clear that a page's graphics and layout could enhance a website's accessibility and marketability. Content was still king, but design was moving up quickly.

To relieve the woes of web designers everywhere, the W3C included a feature within the new Cascading Style Sheets specifications that allowed for absolute positioning of an element upon a page. Absolute positioning enables an element, such as an image or block of text, to be placed anywhere on the web page. Browser support for Cascading Style Sheets-Positioning (CSS-P) specification began with fourth-generation browsers and has grown steadily ever since.

The addition of the third dimension, depth, truly turned the positioning specs into AP elements. Now objects can be positioned side by side, and they have a z-index property as well. The z-index gets its name from the practice in geometry of describing three-dimensional space with x, y, and z coordinates; the z-index is also called the stacking order because objects can be stacked upon one another.

All these attributes, and others such as background color, can be assigned to a CSS style, as shown in the following code:

```
#header {
  position: absolute;
  z-index: 1;
  height: 115px;
  width: 400px;
  left: 100px;
  top: 50px;
  background: #FFCC33;
}
```

The CSS style is then applied to a <div> tag to represent an area on the page, like this:

```
<div id="header">Header content goes here.</div>
```

Dreamweaver calls <div> tags that are drawn with your mouse AP elements; the CSS style is automatically created and embedded in the page for you. Drawing out the same AP element results in the same CSS code, except the selector ID is automatically created for you (apDiv1, apDiv2, and so on) and the code is embedded in the page, like this:

```
<style type="text/css">
<!--
#apDiv11 {
  position: absolute;
  z-index: 1;
}
```

```
height: 115px;
width: 400px;
left: 100px;
top: 50px;
background: #FFCC33;
}
-->
</style>
```

The `<div>` code is also added for you, *sans content*, like this:

```
<div id="apDiv11"></div>
```

Although both approaches are valid, many designers prefer to keep the CSS information in an external style sheet rather than embedded. As you see in this chapter, Dreamweaver supports both methods fully.

If you don't define a unit of measurement for AP element positioning, Dreamweaver defaults to pixels. If you edit out the unit of measurement, the web browser defaults to pixels.

Positioning Measurement

The positioning of AP elements is determined by aligning elements on an x-axis and a y-axis. In CSS, the x-axis (defined as Left in CSS syntax) begins at the left side of the page, and the y-axis (defined as Top in CSS syntax) is measured from the top of the page down. As with many of the other CSS features, you have your choice of measurement systems for Left and Top positioning. All measurements are given in Dreamweaver as a number followed by the abbreviation of the measurement system (without any intervening spaces). The measurement system options are as follows:

Unit	Abbreviation	Measurement
Pixels	px	Relative to the screen
Points	pt	1 pt = 1/72 in
Inches	in	1 in = 2.54 cm
Centimeters	cm	1 cm = 0.3937 in
Millimeters	mm	1 mm = 0.03937 in
Picas	pc	1 pc = 12 pt
EMS	em	The height of the element's font
Percentage	%	Relative to the browser window

For the screen-oriented style sheets prevalent on the web, it's best to use pixels, ems, or percentage. You can, however, use points or other measurement units for print style sheets.

Placing <div> Tags

As noted earlier, CSS-P information can be defined in a style sheet or embedded. Defining the CSS rule in a style sheet (either external or internal) has the benefit of truly separating content from presentation, which, in turn, makes it easier to reshape the content via another style sheet for another medium. A block of text, for example, can be positioned on the right when viewed in a monitor and left when printed out. Moreover, many designers find that centralizing the layout information in a style sheet is a far more effective way to work. Often, the same layout is used on multiple pages of a site; with CSS-P rules in an external style sheet, you can modify the layout of all the related pages simply by altering the CSS in the style sheet. To accomplish the same change when the CSS is embedded would require extensive search-and-replace and the re-uploading of every altered file.

Dreamweaver recognizes the importance of the <div> tag in modern website design with the integration of the <div> object. Not only is the insertion of the <div> tag now possible in Design view, but Dreamweaver also provides visual feedback indicating placement and easy modification through the Property and Tag inspectors.

Defining a CSS rule for a <div> tag

When you are using a <div> object, the typical workflow is to first create the required CSS rules. For layouts, I prefer to use the ID selector because it is applied to only one <div> tag. It's a good idea to give your CSS rules descriptive IDs, such as #header, #footer, #mainContent, and #navigation.



If you're not familiar with creating CSS style rules, see Chapter 6 for more information.

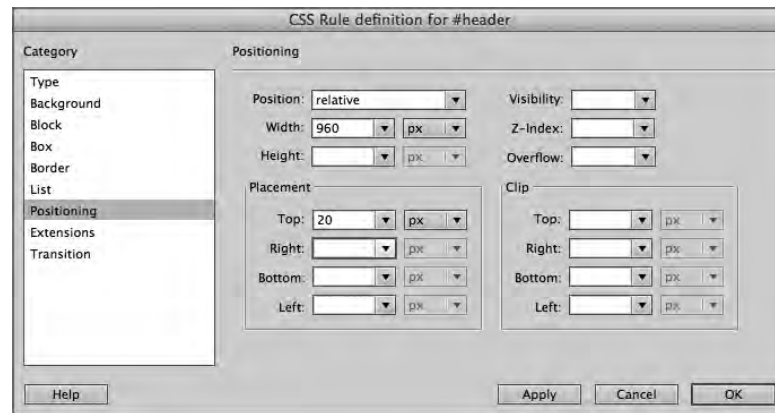
To create a CSS rule for use with <div> tags, follow these steps:

1. Choose Window ⇨ CSS Styles to open the CSS Styles panel.
2. From the CSS Styles panel, click the New CSS Rule button. This action opens the New CSS Rule dialog box.
3. From the New CSS Rule dialog box, set the Type option to ID.
4. Enter a name for your new style in the Selector field. It's a good idea to create the style for your layout <div> tag as an ID. To create an ID, preface the style name with a pound sign, as in #mainContent.
5. If you want to create your style in an external style sheet, use the Browse button to locate an existing style sheet.
6. If you want to add the style to the <head> region of the current document, select This Document Only from the Rule Definition list.
7. Click OK when you're finished to open the CSS Rule Definition dialog box.
8. Select the Positioning category.

- From the Positioning category (see Figure 10-1), enter desired values for the following attributes: Position, Width, Height, Visibility, Z-Index, Overflow, Placement (Top, Right, Bottom, and Left), and Clip settings (Top, Right, Bottom, Left). Overflow and Clip settings are optional.

FIGURE 10-1

Use the Positioning category of the CSS Rule Definition dialog box to set AP element attributes in an internal or external style sheet.



The Position attribute offers four options: Absolute, Fixed, Relative, and Static. An Absolute `<div>` uses the upper-left corner of its nearest positioned ancestor as the origin for the Left and Top measurements, whereas a box with Relative `<div>` tags originates from its current location in the normal document flow. Use Static when you don't want to place the `<div>` in a certain position but you still want to specify a rectangular block. Static `<div>` types ignore the Left and Top attributes.

- If appropriate, select other categories and enter any additional style sheet attributes. Click OK when you're finished.

Inserting the `<div>` tag

After you have defined your CSS rule, follow these steps to add a `<div>` tag to the page:

- Place your cursor where you want the `<div>` tag to appear. You can also select content on a page you'd like to wrap a `<div>` tag around.
- From the Layout category of the Insert panel, click the Insert Div Tag button. Alternatively, you can choose Insert ⇨ Layout Objects ⇨ Div Tag. Dreamweaver displays the Insert Div Tag dialog box, as shown in Figure 10-2.
- Choose the CSS rule from either the Class or the ID list. Dreamweaver shows only those IDs that have not been previously applied.

FIGURE 10-2

Dreamweaver lists all the available CSS rules that can be applied to a new <div> tag, as either as a Class or an ID.

**TIP**

If the CSS rule is not available from either of the lists—perhaps because the style sheet is dynamically applied—you can enter the name directly into either the Class or the ID field. However, unless Design Time Style Sheets are used to show the styles, the layout won't render properly in Dreamweaver.

4. Select where you'd like the tag placed from the Insert list. Dreamweaver provides different options depending on the makeup of the page and whether content is already selected. Only tags with assigned IDs are listed, along with the <body> tag. Here are the Insert options you can choose from:
 - **At Insertion Point:** Inserts the <div> tag at the current cursor position. This option is available only if no content is selected.
 - **Wrap Around Selection:** Wraps the <div> tag around the currently selected content. Available only if a selection was made prior to inserting the <div> tag.
 - **Before Tag:** Puts the tag before the tag selected in the adjacent field.
 - **After Tag:** Inserts the <div> tag after the tag selected in the adjacent field.
 - **After Start Of Tag:** Places the <div> tag immediately following the opening tag in a tag pair, before any content within the tag.
 - **Before End Of Tag:** Inserts the tag right before the closing tag in a tag pair, after any content within the tag.
5. Click OK when you're finished to insert the tag.

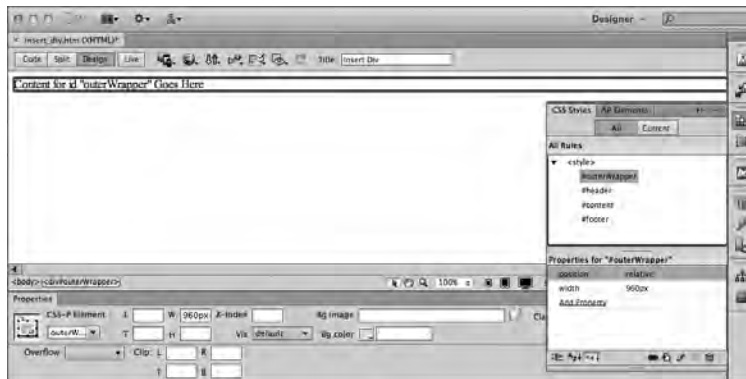
CAUTION

Be sure not to insert the <div> tag in the middle of an empty element. Empty elements, you may remember, are elements that have no corresponding closing tag and contain no content, such as or
.

If the `<div>` tag was not wrapped around previously selected content, Dreamweaver adds placeholder text to help identify the tag and its class or ID. As another helpful aid to identification, a red outline appears when your cursor crosses the outer boundary of the `<div>`. This highlight is controlled by the Mouse-Over option found in the Highlighting category of the Preferences. The red outline is replaced by a thick blue one when the `<div>` tag is selected, as shown in Figure 10-3.

FIGURE 10-3

A thick, blue outline appears around the div when selected.



Select the `<div>` in Design view and the Property inspector displays all the current attributes. If you make any changes in the Property inspector, such as adding or altering the background color, the change is written into the associated CSS style rule. If the `<div>` tag's position property is set to absolute, the outline is supplemented with positioning and sizing handles. The interaction among object, Property inspector, and style sheet holds true if you drag the selected `<div>` around the page or resize it using the sizing handles; see the later section "Modifying an AP Element" for more details. You can also modify `<div>` tag properties by editing the style rule directly or by altering its properties on the Tag inspector panel.

CAUTION

If your layout is controlled by an external style sheet that also controls the layout of other pages in your site, be careful when adjusting the properties of a `<div>` tag. When you make changes, Dreamweaver modifies the CSS style rule in the external style sheet, potentially altering the layout of other pages using the same style sheet.

Visualizing `<div>` tags

The `<div>` tag is a structural element, not intended to be apparent when viewed through the browser at runtime. Design time, however, is another matter. Designers often need to be able to see the underlying structure to craft their layouts; they also need to be able to hide the structure at any point so they can see a browser-like view while designing.

Dreamweaver provides a full slate of visualization options for CSS layouts. Each of the options, found under the Visual Aids menu button on the Document toolbar or the View ⇄ Visual Aids menu item, can be toggled in and out of view at will. There are three different visualizations that can be used singly or in combination: CSS Layout Backgrounds, CSS Layout Box Model, and CSS Layout Outlines. The three CSS layout visualization options apply to other page elements in addition to `<div>` tags. Any page element with the CSS declaration of `display:block`, `position:absolute`, or `position:relative` is considered a block layout element and is affected as well. For example, if an `<a>` tag style was set to `display:block`—a common method used when developing CSS navigation buttons—it would be rendered with the same visualizations as `<div>` tags, including sizing handles, and so on.

CSS Inspect

Live View makes it easy for the Dreamweaver designer to accurately preview the current web page as it appears in a modern browser. Unfortunately, by itself, Live View obscures the underlying CSS structure: There's no easy way to see how the various elements fit together. However, Dreamweaver includes an option called Inspect that exposes certain properties of any CSS block-level elements on the page.

NOTE

With Inspect mode enabled, you can quickly view the margins, paddings, border, and content area of any block-level element in Live View. Just click Inspect and hover over any `<div>`, `<p>`, ``, or other tag to display these properties as solid color blocks (see Figure 10-4). In addition, Inspect can highlight the targeted area in both Code view and the CSS Styles panel.

FIGURE 10-4

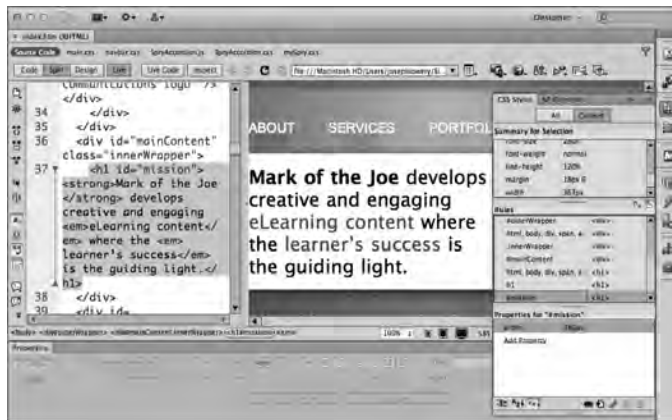
When Inspect mode is toggled on in Live View, you can reveal the margins, padding, and border of any element by hovering over it.



Inspect mode is also available in other Dreamweaver views of the document. If you have Code view or Live Code enabled, the code for the element under the cursor in Live View is highlighted. When the CSS Styles panel is open and Current mode engaged, the CSS rule that applies to the targeted element is shown. You can see both of these effects at once—as well as the box model color highlights in Live View—in Figure 10-5. Better still, any changes made to the CSS Styles panel are instantly reflected in Live View and Inspect mode.

FIGURE 10-5

Inspect mode can also reveal the code and CSS related to a targeted element.



Inspect mode is great for getting a clearer picture of how your page is structured and very useful when debugging a CSS issue.

CSS layout backgrounds

In the early stages of laying out a page, it's often helpful to see your basic building blocks clearly depicted. When you invoke the CSS Layout Background options from the Visual Aids menu, Dreamweaver clears any background image or color previously defined in the CSS styles and replaces them with a different solid color for each `<div>` tag. The resulting patchwork shows at a glance how the page is structured (see Figure 10-6). This option is also useful when debugging layouts because it clearly shows which `<div>` tags—if any—overlap.

TIP

The colors assigned to each of the `<div>` backgrounds are random and can't be predefined. A new set of colors is used every time you toggle CSS Layout Backgrounds into view.

CSS Layout Box Model

All CSS block elements are rendered in the browser according to the CSS box model. The box model, established by the W3C CSS standards body, determines how much room a block element

actually takes up on the page. The amount of space required for a block element, such as a `<div>` tag, is a combination of the declared width, plus the padding, border, and margin settings. For example, say a `<div>` tag has the following style declared:

```
#myBox {
  width:200px;
  padding:10px;
  border:5px;
  margin:10px;
}
```

FIGURE 10-6

Turn on CSS Layout Backgrounds when first creating your CSS layout or debugging it.



Although nominally, the `myBox` style appears to be 200 pixels wide, CSS specifications indicate that it will actually take up 250 pixels of space horizontally. Here's how the space requirement is figured:

```
200 pixels content area width
 10 pixels padding-left
 10 pixels padding-right
  5 pixels border-left
  5 pixels border-right
 10 pixels margin-left
 10 pixels margin-right
250 pixels width total
```

To make it easy for you to design with the box model in mind, Dreamweaver provides the CSS Layout Box Model visual aid. When enabled, any selected `<div>` tag or otherwise affected block element depicts all the contributing elements: content area (the width), padding, borders, and margins. Both the padding and the margins are shown with colored diagonal lines, although in opposing directions, as shown in Figure 10-7.

FIGURE 10-7

The CSS Layout Box Model visual aid reveals the unseen reserved space around layout elements.



Dreamweaver not only reveals how the layout element is constructed visually but also provides you a wealth of information from tooltips that appear as you move your cursor around the element. The information in the tooltip varies according to cursor's position:

- Hover over the content area to see all the CSS properties, including those related to the box model.
- Move your cursor over the padding or margin areas to see their respective values—for example, `Margin: 10px`.
- With the cursor over the border, the tooltip reveals the current values for the margin, border, and padding properties.

I recommend turning on the CSS Layout Box Model feature in the fine-tuning and debugging stages of your web page development; it's a great tool for understanding exactly why elements on your page are positioned the way they are.

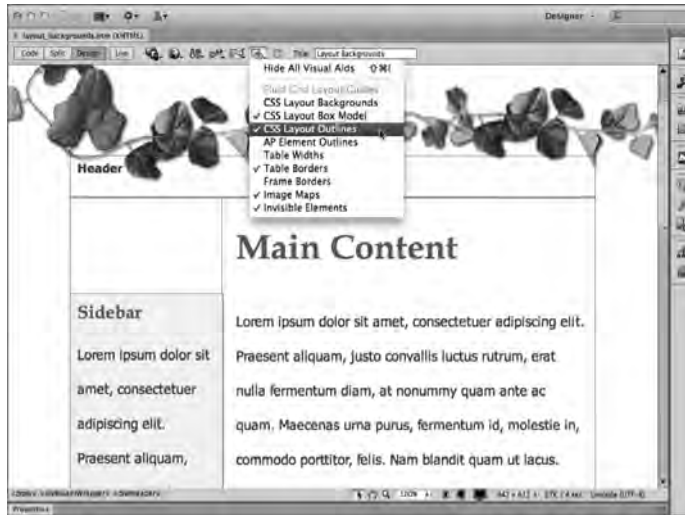
CSS Layout Outlines

CSS Layout Outlines, when enabled, place a border around `<div>` tags and other block layout elements (see Figure 10-8). The outline is a dashed style for inserted `<div>` tags and a solid border for drawn Dreamweaver AP elements.

I tend to keep CSS Layout Outlines enabled during most of my design process; I find having the outlines visible allows me to quickly insert content in just the right places and, more important, select the `<div>` tags for quick refinement.

FIGURE 10-8

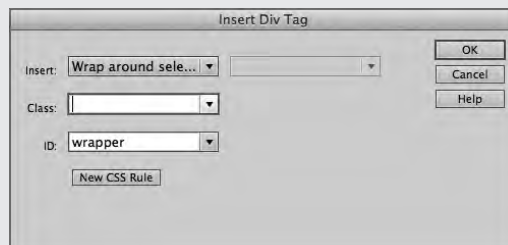
Highlight your <div> tags at design time with CSS Layout Outlines.



Applying a CSS Layout

In this Technique, you apply previously defined CSS styles to a variety of <div> tags to create a basic layout.

1. From the Techniques site, expand the 10_Divs folder and open `div_start.htm`.
The first style to apply acts as a container for the entire page; to apply this properly, you'll need to position the cursor properly.
2. Place your cursor in the series of links on the page and click the <p> tag in the Tag Selector.
3. From the Insert panel's Layout category, click Insert Div Tag.
4. When the Insert Div Tag dialog box opens, make sure the Insert list is set to Wrap Around Selection.
5. From the ID list, choose Wrapper and click OK.



Continues

continued

Next, add the first of three `<div>` tags, the header, so that it appears with the wrapper `<div>` tag:

1. From the Insert panel's Layout category, click Insert Div Tag.
2. From the Insert list, choose After Start Of Tag and then, when the adjacent list appears, select `<div id="wrapper">`.
3. From the ID list, choose Header and click OK.
4. Click Delete to remove the selected placeholder text.

Now you're ready to insert the middle `<div>` tag after the header `<div>` tag:

1. From the Insert panel's Layout category, click Insert Div Tag.
2. From the Insert list, choose After Tag and then, when the adjacent list appears, select `<div id="header">`.
3. From the ID list, choose Middle and click OK.

The final `<div>` wraps around navigation links and forms the bottom area, the footer:

1. Place your cursor in the series of links on the page and click the `<p>` tag in the Tag Selector.
2. From the Insert panel's Layout category, click Insert Div Tag.
3. When the Insert Div Tag dialog box opens, make sure the Insert list is set to Wrap Around Selection and, from the ID list, choose Footer; click OK when you're done.



4. Save your page.

Your new CSS-based layout is now ready to be filled out with content.

Designing fluid layouts with a grid

Anyone who has walked into a phone or electronics store recently will agree: Web-enabled screens now come in every size imaginable. While it's possible to keep the same design for all devices—whether desktop, tablet, phone, or whatever—that choice will typically result in a

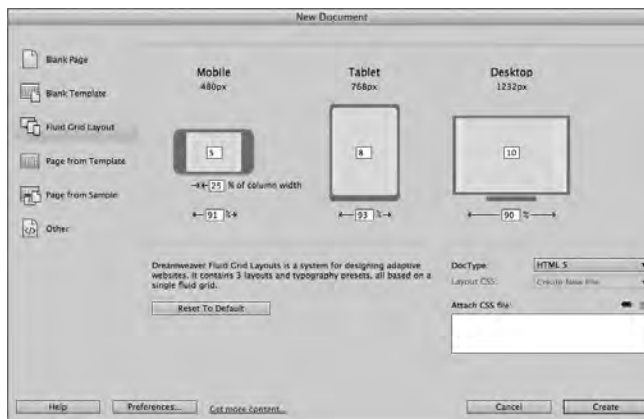
great deal of zooming and scrolling. A far better approach is to optimize the layout for a select number of devices and then switch from one layout to another using CSS media queries. What's even better is building those separate designs with a fluid grid that provides the best fit to myriad screen sizes while maintaining a uniform structure.

NEW FEATURE

Setting up a fluid grid involves determining the number of columns in a particular layout and the amount of space between each column—all in precise percentages of the overall screen target width. While it's entirely doable (given an industrial-grade calculator), it can be a little overwhelming. Dreamweaver now offers to take that chore off your hands with its support for fluid grid layouts. You can create a new fluid grid layout right from the New Document dialog box (see Figure 10-9) and immediately start designing for multiple screens: All your media queries and additional code are set up for you. As you build up your layout, you can easily switch between the three major screen sizes with just a click. When the design is finished, you can preview your work in Live View or a browser, where you'll notice that resizing the screen smoothly adjusts the design, but the relative layout is kept.

FIGURE 10-9

The Fluid Grid Layout dialog box defines three different screen sizes.



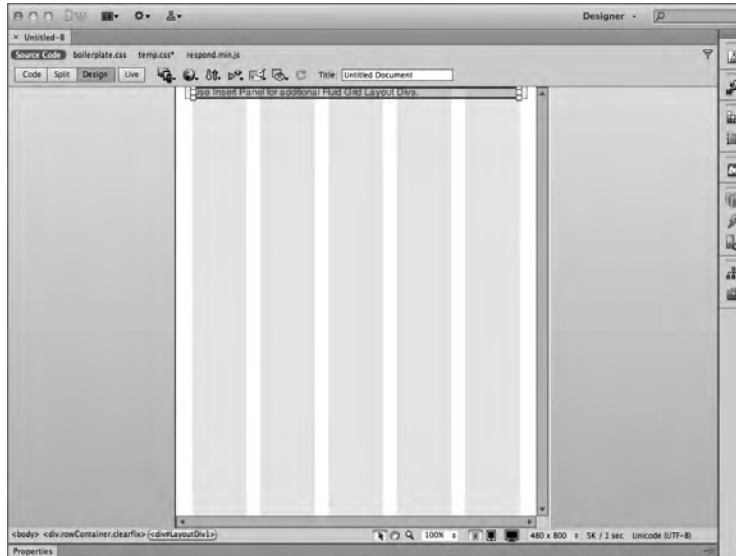
Working with a fluid grid layout

When the page is first created, you'll notice its distinct difference from the standard Document window: A series of columns each separated from the other by a uniform width are displayed in the document's background (see Figure 10-10).

The idea of the fluid grid layout is to use these columns as guides to determine the width of page elements such as `<div>` tags. The accompanying CSS allows these elements to then expand and contract gracefully when the browser window is within one set of media query conditions; this allows the web design to look its best on a large number of similar devices. When a fluid grid layout page is rendered on a different device, the media query statements apply a different, more suitable, set of CSS rules.

FIGURE 10-10

The number of layout columns displayed depends on the screen width; the mobile phone screen shows five columns.



The initial fluid grid layout document, by default, includes one `<div>` tag within another; the outer `<div>` tag has `.rowcontainer` and `.clearfix` classes for containing floats. The Document window size is automatically set to mobile phone dimensions. Here's what the starting code in the `<body>` tag looks like:

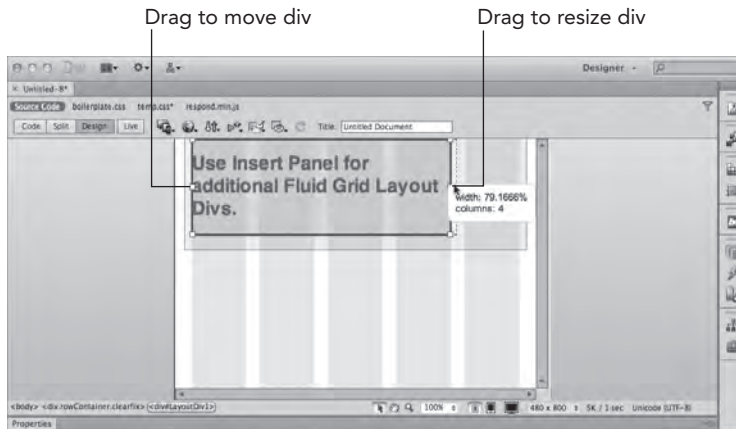
```
<div class="rowContainer clearfix">
  <div id="LayoutDiv1">Use Insert Panel for additional Fluid Grid
  Layout Divs.</div>
</div>
```

When either `<div>` is selected, you'll notice a series of handles surrounding it (see Figure 10-11). To resize a `<div>` tag, drag any of the handles on the right; to reposition the `<div>` tag, drag any handle on the left.

Dreamweaver provides an easy way to add new `<div>` tags to a fluid grid layout. To include a new element, expand the Layout category of the Insert panel and click Insert Fluid Grid Layout Div Tag. Dreamweaver will present a dialog box that allows you to specify an ID for the tag with an option to start a new row (see Figure 10-12). If you leave the Start New Row checkbox selected, the CSS rule for the new `<div>` tag will include a `clear: both` declaration; if the checkbox is not selected, `clear: none` is inserted instead.

FIGURE 10-11

Use the handles to resize and reposition the fluid grid <div> tags.

**FIGURE 10-12**

Add a new <div> tag to your fluid grid layout through the Insert panel object.



Let's walk through a typical fluid grid workflow so you can understand how to make the most of this new feature:

1. Create a new document by choosing File ⇨ New Fluid Grid Layout or, alternatively, File ⇨ New and then selecting the Fluid Grid Layout category.
2. Once the new page is created, save it and confirm Dreamweaver's request to copy the helper files to your site.
3. Use the Insert Fluid Grid Layout Div Tag object from the Layout category of the Insert panel to create the page structure.

TIP

If you want to place two or more columns on the same row (not recommended for mobile phone layouts), add `.rowcontainer` and `.clearfix` classes in a surrounding `<div>` tag.

4. Click Tablet Size in the Status bar.
5. Resize and reposition the existing `<div>` tags for an optimum tablet layout.
6. Click Desktop Size in the Status bar.
7. Resize and reposition the existing `<div>` tags for the best desktop layout.

Unlike most layout features in Dreamweaver, fluid grid `<div>` tags are editable in Live View and Design view. Select any `<div>` to reveal the surrounding border and draggable handles. To turn off the background grid, toggle off the Fluid Grid Layout Guides entry under the Visual Aids menu button.

Fluid grid layout helper files

When you first save your fluid grid layout page, Dreamweaver asks you to confirm copying two files—`boilerplate.css` and `respond.js`—to your site and then links to those pages from your source code. Let's take a quick look at what these helper files actually do and how you can use them to create better, more responsive sites.

Let's first look at the JavaScript file, `respond.js`. One of the major components of the fluid grid layout is the CSS3 media query statement. Because media queries are a relatively recent development, older browsers, like Internet Explorer 6 through 8, do not support them. The code in `respond.js` changes all that and makes it possible for those browsers to apply the styles found in a CSS style sheet conditionally chosen by a media query. The backwards compatibility brought by `respond.js` goes a long way toward robust cross-browser support for fluid grid layouts.

The CSS helper file, `boilerplate.css`, is much more broad based. This file is a part of a larger effort hosted at <http://html5boilerplate.com/> that has created a full starting point for HTML5 sites to work with the full spectrum of browsers, from Internet Explorer 6 to the most recent mobile-ready programs. The `boilerplate.css` copied by Dreamweaver to your site is well documented, so it's easy to see what's included. Here are a few highlights:

- HTML5 displays definitions that add backwards compatibility for the latest web language version.
- Numerous tweaks for improving readability in the browser.
- Many CSS rules that promote uniformity across browsers in tables, form elements, media tags.
- Helper classes to hide page elements from both screenreaders and browsers (`.hidden`) or just browsers (`.visuallyhidden`).

The fluid grid layout source code integrates one of the HTML5 Boilerplate methods throughout, but notably right up front, as a replacement for the standard technique for using conditional comments. Rather than include a range of style corrections, the fluid grid layout uses conditional comments to set up a series of conditional classes, such as `.ie6`, `.ie7`, and `.oldie`, right in the

opening `<html>` tag. This method allows designers to create targeted CSS rules within their standard style sheets, like this:

```
.ie6 aside { width: 120px; } //standard aside width 123px;
```

All in all, the fluid grid layout helper files make it far easier for the modern web designer to work efficiently and effectively.

Creating AP Elements with Dreamweaver

Dreamweaver enables you to drag out absolutely positioned `<div>` tags, also known as AP elements, creatively and precisely—and without coding. You can drag out an AP element, placing and sizing it by eye, or choose to do it by the numbers—it's up to you. Moreover, you can combine the methods, quickly eyeballing and roughing out an AP element layout and then aligning the edges precisely. For web design that approaches conventional page layout, Dreamweaver even includes rulers and a grid to which you can snap your AP elements. Creating AP elements in Dreamweaver can be handled in one of three ways:

- You can drag out an AP element after clicking the Draw AP Div button on the Insert panel.
- You can add an AP element in a predetermined size by choosing Insert ⇄ Layout Objects ⇄ AP Div.
- You can create an AP element with mathematical precision through the CSS Styles panel.

The first two methods are quite intuitive and are explained in the following section. The CSS Styles panel method is examined later.

Inserting an AP element object

When you want to draw out your AP element quickly, use the object approach. If you come from a traditional page-designer background and are accustomed to using a program such as QuarkXPress or InDesign, you're already familiar with drawing out frames or text boxes with the click-and-drag technique. Dreamweaver uses the same method for placing and sizing new AP element objects. To draw out an AP element as an object, follow these steps:

1. From the Layout category of the Insert panel, click the Draw AP Div button. Your pointer becomes a crosshairs cursor. (If you decide not to draw out an AP element, you can press Esc at this point or just click once without dragging to abort the process.)
2. Click anywhere in your document to position the AP element and drag out a rectangle. Release the mouse button when you have an approximate size and shape that is satisfactory (see Figure 10-13).

After you've dragged out your AP element, notice several changes to the screen. First, the AP element now has a small box on the outside of the upper-left corner. This box, shown in Figure 10-14, is the Selection handle, which you can use to move an existing AP element around the web page. When you click the selection handle, eight resize handles appear around the perimeter of the AP element.

FIGURE 10-13

After selecting the Draw AP Div object in the Insert panel (Layout category), the pointer becomes crosshairs when you are working on the page. Click and drag to create the AP element.

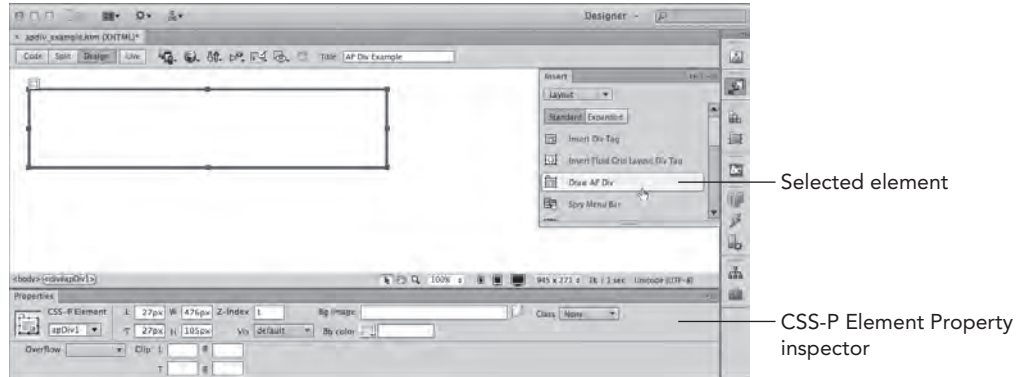


FIGURE 10-14

After an AP element is created, you can move it by dragging the selection handle and size it with the resize handles.



Another subtle but important addition to the screen is the AP element icon. Like the other Invisible Element icons, the AP element icon (when enabled in Preferences) can be cut, copied, pasted, and repositioned. When you move the AP element icon, however, its corresponding AP element does not move—you are actually only moving the code for the AP element to a different place in the HTML source. Generally, the location of the actual AP element code in the HTML is immaterial—however, you may want to locate your AP element source in a specific area to be appropriately placed for accessibility purposes. Dragging and positioning AP element icons one after another is a quick way to achieve this task.

Using the Insert ⇨ Layout Objects ⇨ AP Div option

The second method for creating an AP element is to use the menus. Instead of selecting an object from the Insert panel, choose Insert ⇨ Layout Objects ⇨ AP Div. Unlike the click-and-drag method, inserting an AP element through the menu automatically creates an AP element in the upper-left corner; the default size is 200 pixels wide by 115 pixels high.

Although the AP element is by default positioned in the upper-left corner of the Document window, it does not have any coordinates listed in the Property inspector. The position coordinates are added when you drag the AP element into a new position. If you repeatedly add new AP elements

through the menus without moving them to new positions, each AP element stacks directly on top of the previous one, with no offset.

CAUTION

It's important to assign a specific position (left and top) to every AP element. Otherwise, the browser displays all AP elements directly on top of one another. To give an AP element measurements, after you've inserted it through the menu, be sure to drag the AP element, even slightly, or manually type coordinates in the Property inspector.

Setting default characteristics of an AP element

You can designate the default size—as well as other features—of the inserted AP element with Insert ⇨ Layout Objects ⇨ AP Div. Choose Edit ⇨ Preferences (Dreamweaver ⇨ Preferences) or use the keyboard shortcut Ctrl+U (Command+U) to open the Preferences dialog box. Select the AP Elements category. The AP Elements Preferences category (see Figure 10-15) helps you set the AP element attributes described in Table 10-1.

FIGURE 10-15

If you're building AP elements to a certain specification, use the AP Elements Preferences category to designate your options.

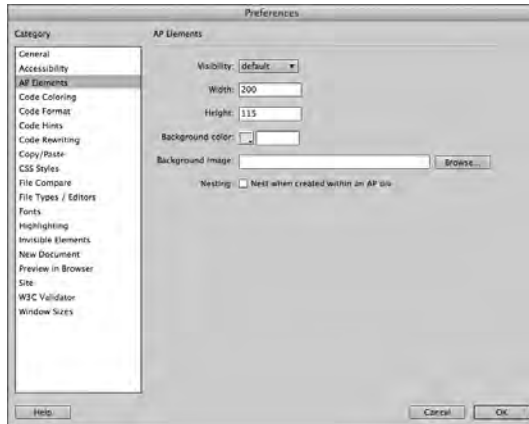


TABLE 10-1 AP Elements Preferences

AP Element Attribute	Description
Visibility	Determines the initial state of visibility for an AP element. The options are Default, Inherit, Visible, and Hidden.
Width	Sets the width of the AP element in the measurement system of your choice. The default is 200 pixels.

Continues

TABLE 10-1 (continued)

AP Element Attribute	Description
Height	Sets the height of the AP element in the measurement system of your choice. The default is 115 pixels.
Background Color	Sets a color for the AP element background. Select the color from the color palette of web-safe colors.
Background Image	Sets an image for the AP element background. In the text box, enter the path to the graphics file or click the Browse button to locate the file.
Nesting	If you want to nest AP elements when one AP element is placed in the other automatically, check the Nest When Created Within An AP Div checkbox.

Choosing relative instead of absolute positioning

In many cases, absolute positioning uses the top-left corner of the web page—the position at which the `<body>` tag begins—as the point of origin for positioning the AP elements. You can also specify measurements relative to other objects, such as `<div>`. Dreamweaver offers several methods to accomplish relative positioning.

Using the Relative attribute

In the first method for handling relative positioning, you select Relative as the Type attribute in the Style Sheet Positioning category. Relative positioning does not force a fixed position; instead, the positioning is guided by the HTML tags around it. For example, you may place a list of some items within a `<div>` tag and set the positioning relative to the table. You can see the effect of this sequence in Figure 10-16. Notice that Dreamweaver does not display sizing handles or a selection handle for relative AP elements.

FIGURE 10-16

The selected AP element is positioned relative to the bottom of the surrounding `<div>` tag.



Relative attributes can be useful, particularly if you want to place the positioned objects within free-flowing HTML. Free-flowing HTML repositions itself based on the size of the browser. When you're using this technique, remember to place your relative AP elements within AP elements. Otherwise, when the end user resizes the browser, the relative AP elements position themselves relative to the browser and not to the AP elements. This situation can produce messy results—use relative positioning with caution when mixed with absolute AP elements.

Using nested AP elements

The second technique for positioning AP elements relatively uses nested AP elements. After you nest one AP element inside another, the inner AP element uses the upper-left corner of the outer AP element as its orientation point. One approach for creating a nested AP element is to position your cursor in the outer AP element and press Alt (Option) after you start to drag out your AP element. For more details about nesting layers, refer to the section “Nesting with the AP Elements panel” later in this chapter.

Modifying an AP Element

Dreamweaver helps you deftly alter AP elements after you have created them. Because of the complexity of managing AP elements, Dreamweaver offers a tool in addition to the usual Property inspector: the AP Elements panel. This tool enables you to select any of the AP elements on the current page quickly, change AP element relationships, modify their visibility, and adjust their stacking order. You can also alter the visibility and stacking order of a selected AP element in the Property inspector, along with many other attributes. Before any modifications can be accomplished, however, you have to select the AP element.

Selecting an AP element

You can choose from several methods to select an AP element for alteration (see Figure 10-17). The selection method you choose generally depends on the complexity of your page layout:

- When you have only a few AP elements that are not overlapping, just click the selection handle of an AP element to select it.
- When you have AP elements placed in specific places in the HTML code (for example, an AP element embedded in a table using relative positioning), click the AP Element icon.
- When you have many overlapping AP elements that are being addressed by one or more JavaScript functions, use the AP Elements panel to choose an AP element by ID.
- When you're working with invisible AP elements, click the `<div>` (or ``) tag in the Tag Selector to reveal the outline of the AP element.

Resizing an AP element

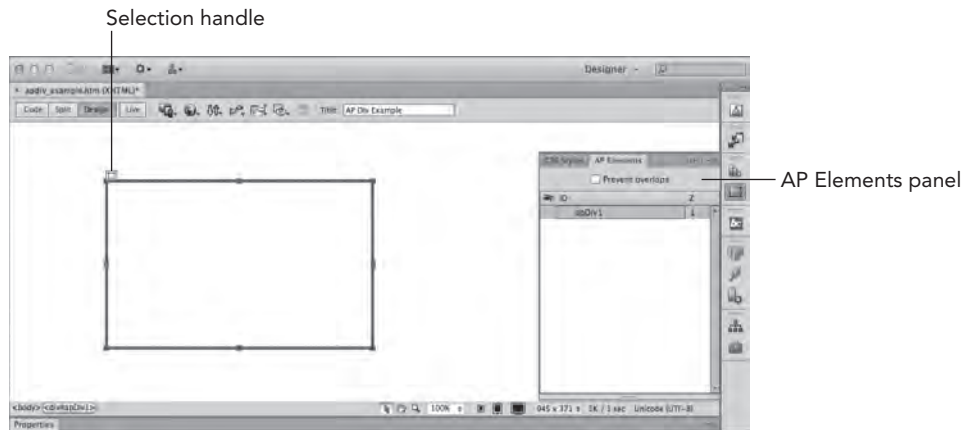
To resize an AP element, position the pointer over one of the eight resize handles surrounding the selected AP element. When over the handles, the pointer changes shape to a two- or four-headed arrow. Now click and drag the AP element to a new size and shape.

You can also use the arrow keys to resize your AP element with more precision. The following keyboard shortcuts change the width and height dimensions while the AP element remains anchored by the upper-left corner:

- When the AP element is selected, press Ctrl+arrow (Command+arrow) to expand or contract the AP element by 1 pixel.
- Press Ctrl+Shift+arrow (Command+Shift+arrow) to increase or decrease the selected AP element by 10 pixels.

FIGURE 10-17

There are a number of different methods for selecting an AP element to modify.



TIP

You can quickly preview the position of an AP element on a web page without leaving Dreamweaver in a couple of ways. One technique is to switch to Live View. Another method would be to deselect the View ⇄ Visual Aids ⇄ AP Element Borders option. This leaves the AP element outline displayed only when the AP element is selected; otherwise, it is not shown.

Moving an AP element

The easiest way to reposition an AP element is to drag the selection handle. If you don't see the handle on an AP element, click anywhere in the AP element. You can drag the AP element anywhere on the screen—or off the bottom or right side of the screen. To move the AP element off the left side or top of the screen, enter a negative value in the left and top (L and T) text boxes of the CSS-P Property inspector; whenever an AP element is selected, the CSS-P Property inspector is available.

TIP

To hide the AP element completely, match the negative value with the width or height of the AP element. For example, if your AP element is 220 pixels wide and you want to position it off-screen to the left (so that the AP element can slide onto the page at the click of a mouse), set the Left position at -220 pixels.

As with resizing AP elements, you can also use the arrow keys to move the AP element more precisely:

- Press any arrow key to move the selected AP element 1 pixel in any direction.
- Use Shift+arrow to move the selected AP element by 10 pixels.

Using the CSS-P Property inspector

You can modify almost all the CSS-P attributes for your AP element right from the CSS-P Property inspector, shown in Figure 10-18. Certain attributes, such as width, height, and background image and color are self-explanatory or recognizable from other objects. Other AP elements-only attributes, such as visibility and inheritance, require further explanation. Table 10-2 describes all the CSS-P properties, and the following sections discuss the features unique to AP elements.

FIGURE 10-18

The CSS-P Property inspector makes it easy to move, resize, hide, and manipulate all the visual elements of an AP element.

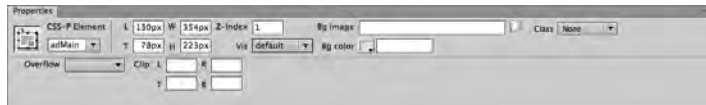


TABLE 10-2 CSS-P Property Inspector Options

CSS-P Attribute	Possible Values	Description
BgColor	Any hexadecimal or valid color name	Background color for the AP element.
BgImage	Any valid graphic file	Background image for the AP element.
Clip (Top, Bottom, Left, Right)	Any positive integer	Measurements for the displayable region of the AP element. If the values are not specified, the entire AP element is visible.

Continues

TABLE 10-2 (continued)

CSS-P Attribute	Possible Values	Description
H (Height)	Any positive integer measurement in pixels, centimeters, millimeters, inches, points, percentage, ems, or picas	Vertical measurement of the AP element.
ID	A unique ID without spaces or special characters	A label for the AP element so that it can be addressed by style sheets or JavaScript functions.
L (Left)	Any integer measurement in pixels, centimeters, millimeters, inches, points, percentage, ems, or picas	Distance measured from the origin point on the left.
Overflow	visible, scroll, hidden, or auto	An indication of how text or images larger than the AP element should be handled.
T (Top)	Any integer measurement in pixels, centimeters, millimeters, inches, points, percentage, ems, or picas	The distance measured from the origin point on the top.
Tag	span or div	Type of HTML tag to use for the AP element.
Vis (Visibility)	default, inherit, visible, or hidden	An indication of whether an AP element is displayed. If visibility is set to inherit, the AP element takes on the characteristic of the parent AP element.
W (Width)	Any positive integer measurement in pixels, centimeters, millimeters, inches, points, percentage, ems, or picas	The horizontal measurement of the AP element.
Z-Index	Any integer	Stacking order of the AP element relative to other AP elements on the web page. Higher numbers are closer to the top.

ID

IDs are important when working with AP elements. So you can refer to them properly for both CSS and JavaScript purposes, each AP element must have a unique ID attribute, unique among the AP elements and unique among every other object on the web page. Dreamweaver automatically names each AP element as it is created in sequence: apDiv1, apDiv2, and so forth. You can enter an ID that is easier for you to remember by replacing the provided ID in the text box on the far left of the Property inspector.



Inserting an AP Element

In this Technique, you add a Dreamweaver AP element to a CSS layout. The AP element is absolutely positioned within a relatively positioned `<div>` tag so that if the centered layout moves, the AP element stays in the proper place.

1. From the 10_Div folder, open `apDiv_start.htm`.
2. Place your cursor in the header `<div>`.
3. From the Insert panel's Layout category, click Draw AP Div.
4. Begin dragging out a rectangle on the right side of the header div and, while dragging, press and hold Alt (Option). When your rectangle is approximately 150 pixels wide by 50 pixels high, release the mouse and then Alt (Option). By pressing Alt (Option) while drawing the AP element, Dreamweaver puts the code for the AP element at the cursor position, inside the header `<div>`, effectively nesting the AP element.

CAUTION

As of this writing, there is a bug in the Windows implementation for nesting AP `<div>` tags: Pressing the Alt key while drawing an AP div does not insert a `<div>` tag and instead displays the Code Navigator. To work around this issue, first open Preferences and switch to the AP Element category. Then, enable the Nesting option. When the Nesting option is enabled, you won't need to press Alt while drawing an AP div to nest it. As long as your cursor is in another `<div>` tag, such as the header, it will nest automatically.



5. With your cursor inside the newly drawn AP element, switch to the Insert panel's Common category and click Images: Image.
6. In the Select Image Source dialog box, navigate to the images folder and select `new_properties.jpg`. Note that the Property inspector shows the image's dimensions as 166 × 55 pixels.
7. Select the AP element handle, and in its Property inspector, change the Width value to 166px and Height to 55px.

Continues

continued

- Precisely position the AP element by setting the Left value to 533px and Top to 50px.
- The last step is to rename the AP element and its accompanying style from the generic apDiv1 to something more precise. In the Property inspector's Name field, enter `new_properties`.



- Save your page.

If you test your page by previewing in the browser, you'll notice that the absolutely positioned AP element stays in the correct place even if the window is resized.

Tag attribute

The Tag drop-down list contains the HTML tags that can be associated with the AP element. By default, the positioned AP element has `<div>` as the tag, but you can also choose ``. As previously noted, the `<div>` and `` tags are endorsed by the World Wide Web Consortium group as part of its CSS standards.

Visibility

Visibility (`vis` in the Property inspector) defines whether you can see an AP element on a web page. Four values are available:

- **Default:** Enables the browser to set the visibility attribute. Most browsers use the `inherit` value as their default.
- **Inherit:** Sets the visibility to the same value as that of the parent positioned element, which enables a series of elements to be hidden or made visible by changing only one parent element.
- **Visible:** Causes the AP element and all its contents to be displayed.
- **Hidden:** Makes the current AP element and all its contents invisible.

Remember the following when you're specifying visibility:

- Whether or not you can see an AP element, remember that the AP element is still part of the page and demands some of the page-loading time. Hiding an AP element does not affect the layout of the page, and invisible graphics take just as long to download as visible graphics.
- When you are defining the visibility of a positioned object or AP element, do not use `default` as the visibility value. A designer does not necessarily know whether the site's end user has set the default visibility to `visible` or `hidden`. Designing an effective web page can be difficult without this knowledge. The common browser default is for visibility to be inherited, if not specifically shown or hidden.

Overflow

Typically, an AP element expands to fit the text or graphics inserted into it. You can, however, restrict the size of an AP element by changing the height and width values in the Property inspector. What happens when you define an AP element to be too small for an image, or when an amount of text depends on the setting of the AP element's `overflow` attribute? AP elements (the `<div>` and `` tags) support four different overflow settings:

- **Visible (Default):** All the overflowing text or image is displayed, and the height and width settings established for the AP element are ignored. It's possible for overflowing content to overlap other content on the page.
- **Hidden:** The portion of the text or graphic that overflows the dimensions is not visible.
- **Scroll:** Horizontal and vertical scroll bars are added to the AP element regardless of the content size or amount, and regardless of the AP element's measurements.
- **Auto:** When the content of the AP element exceeds the width and/or height values, horizontal and vertical scroll bars appear.

Most modern browsers render the `overflow` attribute correctly, as shown in Figure 10-19.

`<div>` Versus ``

The major difference between `<div>` and `` is that the `<div>` is a block-level element and the `` is inline.

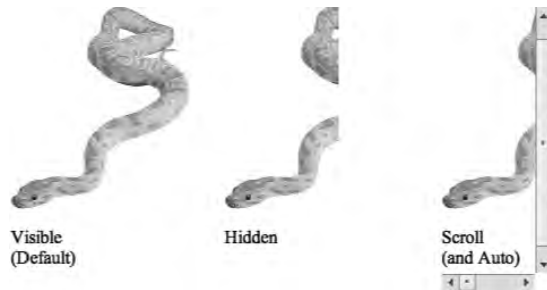
When you are positioning relatively (the elements are in the normal flow of the document), a `<div>` always causes the next element to appear on a new line. Block-level elements, such as `<h1>` and `<p>`, always create a new line unless the `display` property is set to `inline` using CSS.

The reverse is true of `` tags. The `` tag is an inline element and displays just like an image or link, without altering the text around it.

Generally, `<div>` tags are used for block-level elements that require positioning, and `` tags are more commonly used to apply inline formatting over positioning.

FIGURE 10-19

When your contents are larger than the dimensions of your AP element, you can regulate the results with the overflow attribute.



Clipping

If you're familiar with the process of cropping an image, you'll quickly grasp the concept of clipping AP elements. Just as desktop publishing software hides, but doesn't delete, the portion of the picture outside the crop marks, AP elements can mask the area outside the clipping region defined by the Left, Top, Right, and Bottom values in the Clip section of the CSS-P Property inspector.

All clipping values are measured from the upper-left corner of the AP element. You can use any CSS standard measurement system such as pixels (the default) or ems.

The current implementation of CSS supports only rectangular clipping. When you look at the code for a clipped AP element, you see the values you inserted in the CSS-P Property inspector in parentheses following the `clip` attribute, with the `rect` (for rectangular) keyword, as follows:

```
<div id="apDiv1" style="position:absolute; left:54px; top:24px; ;  
width:400px; height:115px; z-index:1; visibility:inherit; ;  
clip:rect(10px 100px 100px 10px)">
```

Generally, you specify values for all four criteria: Left, Top, Right, and Bottom. You can also leave the Left and Top values empty or use the keyword `auto`—which causes the Left and Top values to be set at the origin point: 0,0. If you leave any of the clipping values blank, the blank attributes are set to `auto`.

Z-index

One of an AP element's most powerful features is its capability to appear above or below other AP elements. You can change this order, known as the z-index, dynamically. Whenever a new AP element is added, Dreamweaver automatically increments the z-index—AP elements with higher z-index values are positioned above AP elements with lower z-index values. The z-index can be adjusted manually in either the CSS-P Property inspector or the AP Elements panel. The z-index must be an integer, either negative or positive.

TIP

Although some web designers use high values for the z-index, such as 3,000, the z-index is completely relative. The only reason to increase a z-index to an extremely high number is to ensure that a particular AP element remains on top.

Certain types of objects—including Java applets, plugins, and ActiveX controls—ignore the z-index setting when included in an AP element and appear as the uppermost AP element. However, certain plugin controls—most notably Flash—can be made to respect the z-index. If you need HTML content on top of active content, you can always hide the AP element containing the ActiveX control when necessary.

CAUTION

Working with the `above` and `below` attributes can be confusing. Notice that they determine which AP element is to appear on top of or underneath the current AP element, and not which AP element the present AP element is above or below.

Background image or color

Inserting a background image or color with the CSS-P Property inspector works like changing the background image or color for a table (as explained in Chapter 12). To insert an image, enter the path to the file in the Bg Image text box or select the folder icon to locate the image file on your system or network. If the AP element is larger than the image, the image is tiled, just as it would be in the background of a web page or table, unless you modify the CSS background property for the element.

To give an AP element a background color, enter the color name (in either its hexadecimal or its nominal form) in the Bg Color text box. You can also select the color box to pick your color from the color palette.

The AP Elements panel

Dreamweaver offers another tool to help manage the AP elements in your web page: the AP Elements panel. Although this tool doesn't display as many properties about each element as the Property inspector, the AP Elements panel gives you a good overview of all the AP elements on your page. It also provides a quick method of selecting an AP element—even when it's off-screen—and enables you to change the z-index and the nesting order.

The AP Elements panel, shown in Figure 10-20, can be opened either through the Window menu (Window ⇨ AP Elements) or by pressing the keyboard shortcut F2.

Modifying properties with the AP Elements panel

The AP Elements panel lists the visibility, name, and z-index settings for each AP element. You can modify all these properties directly through the AP Elements panel.

FIGURE 10-20

Use the AP Elements panel to quickly select—or alter the visibility or relationships of—all the AP elements on your page.



The visibility of a particular AP element is noted by the eye symbol in the first column of the inspector. Clicking the eye symbol cycles you through three different visibility states:

- **Eye closed:** Indicates that the AP element is hidden
- **Eye open:** Indicates that the AP element is visible
- **No eye:** Indicates that the visibility attribute is set to the default (which, for both Navigator and Internet Explorer, means inherit)

TIP

To change all your AP elements to a single state simultaneously, click the eye symbol in the column header. Unlike the individual eyes in front of each AP element name, the overall eye toggles between open and shut.

You can also change an AP element's name (in the second column of the AP Elements panel). Just double-click the current AP element name in the inspector; the name is highlighted. Type in the new name and press Enter (Return) to complete the change.

You can alter the z-index (stacking order) in the third column in the same manner. Double-click the z-index value, and then type in the new value and press Enter (Return). You can enter any positive or negative integer.

Nesting with the AP Elements panel

Another task managed by the AP Elements panel is nesting or unnesting AP elements. This process is also referred to as creating parent-child AP elements. To nest one AP element inside another through the AP Elements panel, follow these steps:

1. Choose Window ⇨ AP Elements or press F2 to open the AP Elements panel.
2. Press the Ctrl (Command) key, click the name of the AP element to be nested (the child), and drag it on top of the other AP element (the parent).

3. When you see a rectangle around the parent AP element's name, release the mouse. The child AP element is indented underneath the parent AP element, and the parent AP element has a minus sign (a downward-pointing triangle on the Mac) attached to the front of its name.
4. To hide the child AP element from view, click the minus sign (a downward-pointing triangle on the Mac) in front of the parent AP element's name. After the child AP element is hidden, the minus sign turns into a plus sign (a right-pointing triangle on the Mac).
5. To reveal the child AP element, click the plus sign (a right-pointing triangle on the Mac).
6. To undo a nested AP element, select the child AP element and drag it to a new position in the AP Elements panel.

You can use the nesting features of the AP Elements panel to hide many AP elements quickly. If the visibility of all child AP elements is set to the default—with no eye displayed—then by hiding the parent AP element, you cause all the child AP elements to inherit that visibility setting and also disappear from view.

TIP

You can also delete an AP element from the AP Elements panel. Just highlight the AP element to be removed and press the Delete key. Dreamweaver also enables you to delete nested AP elements as a group by selecting the parent AP element and pressing Delete. To remove a parent AP element but keep all children, use the Tag Selector. Select the parent tag, right-click (Control+click), and then choose Remove Tag.

Aligning AP elements

With the capability to position AP elements anywhere on a page comes additional responsibility and potential problems. In anything that involves animation, correct alignment of moving parts is crucial. As you begin to set up your AP elements, their exact placement and alignment become critical. Dreamweaver includes two tools to simplify layered web page design: the ruler and the grid.

Rulers and grids are familiar concepts in traditional desktop publishing. Dreamweaver's ruler shows the x-axis and y-axis in pixels, inches, or centimeters along the outer edge of the Document window. The grid crisscrosses the page with lines to support a visual guideline when you're placing objects. You can even enable a snap-to-grid feature to ensure easy, absolute alignment.

Using the ruler

With traditional web design, "eyeballing it" was the only option available for web page layout. The absolute positioning capability of AP elements remedied this deficiency. Now online designers have a more precise and familiar system of alignment: the ruler. Dreamweaver's ruler can be displayed in several different measurement units and with your choice of origin point.

To toggle the ruler in Dreamweaver, choose View ⇨ Rulers ⇨ Show or use the keyboard shortcut Ctrl+Alt+R (Command+Option+R). Horizontal and vertical rulers appear along the top and the left sides of the Document window, as shown in Figure 10-21. As you move the pointer, a light-gray line indicates the position on both rulers.

FIGURE 10-21

Use the horizontal and vertical rulers to assist your AP element placement and overall web page layout.



By default, the ruler uses pixels as its measurement system. You can change the default by choosing View ⇨ Rulers and selecting either inches or centimeters.

Dreamweaver also enables you to move the ruler origin to a new position. Normally, the upper-left corner of the page acts as the origin point for the ruler. On some occasions, it's helpful to start the measurement at a different location—at the bottom-right edge of an advertisement, for example. To move the origin point, select the intersection of the horizontal and vertical rulers and drag the crosshairs to a new location. When you release the mouse button, both rulers are adjusted to show negative values above and to the right of the new origin point. To return the origin point to its default setting, choose View ⇨ Rulers ⇨ Reset Origin, or you can simply double-click the intersection of the rulers.

TIP

You can access a ruler context menu by right-clicking (Control+clicking) the ruler itself. The context menu enables you to change the system of measurement, reset the origin point, or hide the rulers.

Working with guides

With the advent of CSS layouts, web designers found themselves needing a toolset more traditionally associated with graphic programs: guides. A guide is a single, thin, positionable line used at design time to align elements on a page. Guides are either horizontal or vertical and stretch from one edge of the browser window to the other.

Dreamweaver guides are wonderfully powerful and incorporate standard features found in graphics programs like Photoshop and Fireworks, as well as some unique web-centric options. Guides can be shown or hidden, set precisely, or varied in color. You can snap guides to the grid and/or objects; you can even do the reverse and have objects snap to guides. Dreamweaver designers will find guides a welcome addition to their working toolbox.

Before you can display a guide, you have to meet two conditions: The View ⇨ Guides ⇨ Show Guides option must be enabled along with View ⇨ Rulers ⇨ Show Rulers. The standard method for deploying a guide in Dreamweaver is the same as in those other programs: with rulers displayed, the user drags a guide onto the document (see Figure 10-22). Horizontal guides are dragged from the ruler on the top edge, and vertical guides are dragged from the left edge ruler. You can have as many guides on your page as you want.

Positioning and removing guides

To place a guide on the page, follow these steps:

1. Choose View ⇨ Rulers ⇨ Show to toggle the rulers into view.
2. Select View ⇨ Guides ⇨ Show Guides so that the option is checked.
3. To place a horizontal guide on the page, drag a guide from the ruler at the top of the page.
4. To place a vertical guide on the page, drag a guide from the ruler at the left of the page.

Once positioned, guides can be moved at any time. You can visually position guides by dragging them to a new location on the page; when your cursor is over the guide, you'll see a tooltip with the precise horizontal or vertical coordinate of the guide in pixels from the top or left of the page, respectively.

To remove a single guide, drag it back into the horizontal or vertical ruler. You can remove all guides on the page by selecting View ⇨ Guides ⇨ Clear Guides. If you just want to hide the guides, choose View ⇨ Guides ⇨ Show Guides again or use the keyboard shortcut Ctrl+; (Command+;).

FIGURE 10-22

Guides work well in a CSS-P layout design environment.



Displaying guide measurements

One of the slicker guide features implemented in Dreamweaver is the display of distance measurement. If you have a single horizontal or vertical guide onscreen and press Ctrl (Command),

Dreamweaver will show you the distance from the guide to the window's edge in your cursor area. Place your cursor between two guides of the same type (either horizontal or vertical) and press Ctrl (Command), and you'll see the distance that separates them, as shown in Figure 10-23.

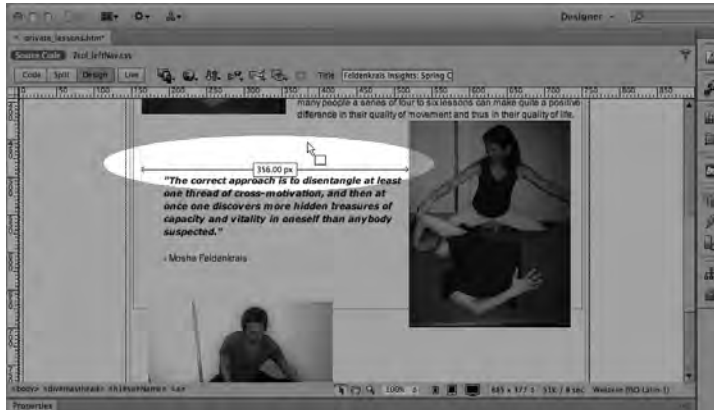
Locking and snapping guides

Dreamweaver gives you complete control over guides and positionable page elements. You can lock your guides so that they are not accidentally moved when repositioning elements. You can also snap your elements to your guides or guides to your elements—or both. Here's how it's all done:

- Prevent your guides from being moved by choosing View ⇨ Guides ⇨ Lock Guides or using the keyboard shortcut, Ctrl+Alt+; (Command+Option+;).
- Align your layout blocks to existing guides by enabling View ⇨ Guides ⇨ Snap Guides or using the keyboard shortcut, Ctrl+Shift+; (Command+Shift+;).
- Snap guides to the edges of layout blocks by selecting View ⇨ Guides ⇨ Guides Snap to Elements or using the keyboard shortcut, Ctrl+Shift+/(Command+Shift+/).

FIGURE 10-23

Bring up the distance between two guides in pixels by holding Ctrl on Windows or Command on a Mac.



TIP

Guides are retained when you save the page and restored when the page is re-opened.

Precise guide placement

Guides can be positioned precisely as well as by dragging. To set a guide's horizontal or vertical placement to a specific value, follow these steps:

1. Double-click the guide you want to move.
2. When the Move Guide dialog box opens, enter the desired position in the Location field.

3. Select the measurement system (pixels, centimeters, inches, or percentages) from the list.
To use the percentage measurement, enter a value between 0.00 and 100.00; Dreamweaver will reposition the guide according to the width or height of your Document window.
4. Click OK.

Editing guide settings

By default, the guides are colored a bright green, and the distance indicators are a dark blue. If these colors don't contrast enough with your layout to be seen clearly, you can adjust the colors through the Edit Guides dialog box. The Edit Guides dialog box also provides all-in-place access to basic guide controls: Show Guides, Snap To Guides, Lock Guides, Guides Snap To Elements, and Clear All (see Figure 10-24). To access the Edit Guides dialog box, choose View ⇨ Guides ⇨ Edit Guides.

FIGURE 10-24

Alter the color of your guides for maximum visibility.



Showing the browser window fold

One application in the guide feature set unique to Dreamweaver is especially useful for web designers: showing the browser window fold. When a browser displays a web page, the portion initially visible is said to be “above the fold.” The concept of a fold comes from the world of newspaper journalism where papers are divided into a top and bottom portion by the way they are folded. Dreamweaver can quickly display the fold of a browser window—that is, the viewable area—through guides.

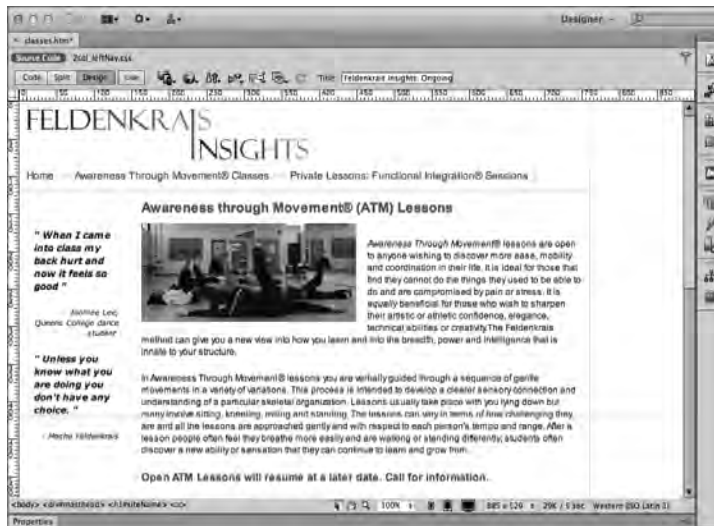
Five of the most common browser window configurations are available through the Guides submenu:

- 640 × 480, Default
- 640 × 480, Maximized
- 800 × 600, Maximized
- 832 × 624, Maximized
- 1,024 × 768, Maximized

You'll recognize the dimensions from the Window Size selector on the Status bar. When you select any of these options from the View ⇨ Guides menu, Dreamweaver inserts two guides—one horizontal and another vertical—to form the right and bottom edge of the browser window (see Figure 10-25). With these guides onscreen, designers can place their key content so that it will be visible immediately in the chosen screen resolution.

FIGURE 10-25

Make sure your most important content is above the fold with guides.



Aligning objects with the grid

Rulers and guides are generally good for positioning single objects, but a grid is extremely helpful when aligning one object to another. With Dreamweaver's grid facility, you can align elements visually or snap them to the grid. You can set many of the grid's other features, including grid spacing, color, and type.

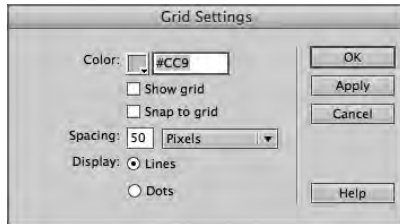
To turn on the grid, choose View ⇨ Grid ⇨ Show or press Ctrl+Alt+G (Command+Option+G). By default, the grid is displayed with tan lines set at 50-pixel increments.

The snap-to-grid feature is enabled by choosing View ⇨ Grid ⇨ Snap To or with the keyboard shortcut Ctrl+Alt+Shift+G (Command+Option+Shift+G). When activated, snap-to-grid causes the upper-left corner of an AP element to be placed at the nearest grid intersection when the AP element is moved.

Like most of Dreamweaver's tools, you can customize the grid. To alter the grid settings, choose View ⇨ Grid ⇨ Settings, and the Grid Settings dialog box opens. In the Grid Settings dialog box, shown in Figure 10-26, you can change any of the settings shown in Table 10-3 (just click OK when you've finished adjusting the settings).

FIGURE 10-26

Dreamweaver's grid feature is extremely handy for aligning a series of objects.

**TABLE 10-3 Grid Settings Dialog Box Options**

Grid Setting	Description
Color	Change the default color (tan) by selecting the color box (which brings up color palette) or by typing a new value in the text box.
Show Grid	Show or hide the grid with this checkbox toggle.
Snap To Grid	Toggle the checkbox to enable or disable the snap-to-grid feature.
Spacing	Adjust the distance between grid points by entering a numeric value in the text box.
Spacing Unit of Measure	Select pixels, inches, or centimeters from the Spacing drop-down list.
Display	Choose either solid lines or dots for the gridlines.

Adding elements to an AP element

After you have created and initially positioned your AP elements, you can begin to fill them with content. Inserting objects in an AP element is just like inserting objects in a web page. The same insertion methods are available to you:

- Position the cursor inside an AP element, choose Insert in the menu bar, and select an object to insert.
- With the cursor inside an AP element, select any object from the Insert panel. Note that you cannot select the Draw AP Div object.
- Drag an object from the Insert panel and drop it inside the AP element.

Forms and AP elements

When you're mixing forms and AP elements, follow only one rule: Always put the form completely inside the AP element. If you place the AP element within the form, all form elements after the AP element tags are ignored. With the form completely enclosed in the AP element, the form can safely be positioned anywhere on the page and all form elements still remain completely active.

Although this rule means that you can't split one form onto separate AP elements, you can set up multiple forms on multiple AP elements—and still have them all communicate to one final CGI or other program. This technique uses JavaScript to send the user-input values in the separate forms to hidden fields in the form with the Submit button. Suppose, for example, that you have three separate forms gathering information in three separate AP elements on a web page. Call them formA, formB, and formC on apDiv1, apDiv2, and apDiv3, respectively. When the Submit button in formC on layer3 is selected, a JavaScript function is first called by means of an `onClick` event in the button's `<input>` tag. The function, in part, looks like the following:

```
function gatherData() {  
    document.formC.hidden1.value = document.formA.text1.value  
    document.formC.hidden2.value = document.formB.text2.value  
}
```

Notice how every value from the various forms is sent to a hidden field in formC, the form with the Submit button. Now, when the form is submitted, all the hidden information gathered from the various forms is submitted along with formC's own information.

NOTE

To make the code cross-browser compatible, you can use an initialization function that allows for the differences, or you can build the code into the `onClick` function. (For more information about building cross-browser-compatible code, see Chapter 30.)

Using the Tracing Image Feature with AP Elements

Page-layout artists are often confronted with webpage designs that have been comped in a graphics program. Dreamweaver's Tracing Image function enables you to use such images to guide the precise placement of graphics, text, tables, and forms in your web page, enabling you to match the original design as closely as possible.

To use a Tracing Image, the graphic must be saved in JPEG, GIF, or PNG format. After the Tracing Image has been placed in your page, it is viewable only in Dreamweaver—it will never appear in a browser. A placed Tracing Image hides any background color or background graphic in your web page. Preview your page in a browser or hide the Tracing Image to view your page without it.

CAUTION

If you're concerned about your page validating, be sure to remove the Tracing Image after you've completed the page. The Tracing Image uses a number of attributes, such as `tracingsrc` and `tracingopacity`, none of which validate, inside the `<body>` tag.

Adding the Tracing Image to your page

To add a Tracing Image to your Dreamweaver page, choose `View ⇨ Tracing Image ⇨ Load`. This brings up a Select Image Source dialog box that enables you to select the graphic to use as a Tracing Image.

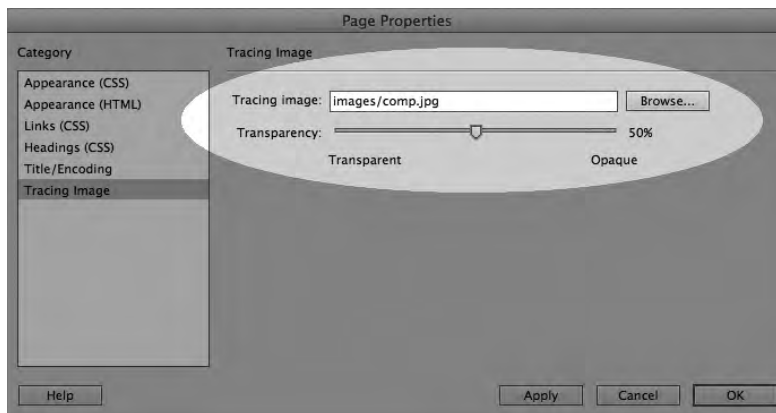
Clicking **Select** brings up the **Page Properties** dialog box, shown in Figure 10-27, in which you can specify the opacity of the **Tracing Image** in a range from **Transparent (0%)** to **Opaque (100%)**. You can change the **Tracing Image** or its transparency at any point by choosing **Modify ⇄ Page Properties** to bring up the **Page Properties** dialog box. You can toggle between hiding and showing the **Tracing Image** by choosing **View ⇄ Tracing Image ⇄ Show**. You can also enter the **Tracing Image** directly in the **Page Properties** dialog box by entering its path in the **Tracing Image** text box or by clicking the **Browse** button to locate the image.

NOTE

Even though the **Browse** dialog box for the **Tracing Image** enables you to choose from a data source, the image is not displayed on the page.

FIGURE 10-27

Setting the transparency of the **Tracing Image** to a setting such as 50 percent can help you differentiate between it and the content AP elements you are positioning.



Moving the Tracing Image

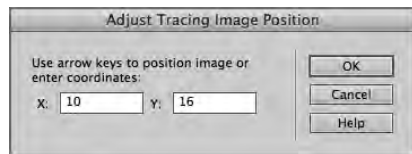
The **Tracing Image** cannot be selected and moved the same way as other objects on your page. Instead, you must move the **Tracing Image** using menu commands. You have several options for adjusting the **Tracing Image**'s position to better fit your design. First, you can align the **Tracing Image** with any object on your page by selecting the object and then choosing **View ⇄ Tracing Image ⇄ Align with Selection**. This procedure lines up the upper-left corner of the **Tracing Image** with the upper-left corner of the bounding box of the AP element you've selected.

To precisely or visually move the **Tracing Image** to a specific location, choose **View ⇄ Tracing Image ⇄ Adjust Position**. Enter the x and y coordinates into their respective boxes in the **Adjust Tracing Image Position** dialog box, as shown in Figure 10-28. For more hands-on positioning, click in the position fields and use the arrow keys to nudge the tracing AP element up, down, left, or right, one pixel at a time. Holding down the **Shift** key while pressing the arrow keys

moves the Tracing Image in 5-pixel increments. Finally, you can return the Tracing Image to its default location of 0 pixels down from the top and 0 pixels in from the left by choosing View ⇄ Tracing Image ⇄ Reset Position.

FIGURE 10-28

Use the Adjust Tracing Image Position dialog box to precisely place your graphic template.



Activating AP Elements with Behaviors

Although absolute positioning is a major reason to use AP elements, you may have other motives for using this capability. All the properties of an AP element—the coordinates, size and shape, depth, visibility, and clipping—can be altered dynamically and interactively. Normally, dynamically resetting an AP element's properties entails some fairly daunting JavaScript programming. Now, with one of Dreamweaver's core features—behaviors—activating AP elements is possible for nonprogrammers as well.



If you want to learn more about behaviors, Chapter 11 describes Dreamweaver's rich behaviors feature.

Behaviors consist of two parts: the event and the action. In Dreamweaver, three standard actions are designed specifically for working with AP elements:

- **Drag AP Element:** Enables the user to move the AP element and get a response to that movement.
- **Set Text of Container:** Enables the interactive alteration of the content of any AP element to include any HTML, not just text.
- **Show-Hide Elements:** Controls the visibility of AP elements, either interactively or through some preprogrammed action on the page.

You can find detailed information about these actions in their respective sections in Chapter 11. The following sections outline how to use these behaviors to activate your AP elements.

Drag AP Element

For the web designer, positioning an AP element is easy: Click the selection handle and drag the AP element to a new location. For the readers of your pages, moving an AP element is next to impossible—unless you incorporate the Drag AP Element action into the page's design.

With the Drag AP Element action, you can set up interactive pages in which the user can rearrange elements of the design to achieve an effect or make a selection. The Drag AP Element action includes an option that enables your application to execute a JavaScript command if the user drops the AP element on a specific target. In the example shown in Figure 10-29, each pair of shoes is in its own AP element. When the user drops a pair in the bag, a one-line JavaScript command opens the desired catalog page and order form.

After you've created all your AP elements, you're ready to attach the behavior. Because Drag AP Element initializes the script to make the interaction possible, you should always associate this behavior with the `<body>` tag and the `onLoad` event. Follow these steps to use the Drag AP Element action and to designate the settings for the drag operation:

1. Choose the `<body>` tag from the Tag Selector in the status bar.
2. Choose Window ⇨ Behaviors or press Shift+F4. The Behaviors panel opens.
3. Click the Add (+) button and choose Drag AP Element from the Add Action drop-down list.
4. In the Drag AP Element dialog box, select the AP element you want to make available for dragging.
5. To limit the movement of the dragged AP element, select Constrained from the Movement drop-down list. Enter the coordinates needed to specify the direction to which you want to limit the movement in the Up, Down, Left, and/or Right text boxes.
6. To establish a location for a target, enter coordinates in the Drop Target: Left and Top text boxes. You can fill these text boxes with the selected AP element's present location by clicking the Get Current Position button.
7. You can also set a snap-to area around the target's coordinates. When released in the target's location, the dragged AP element snaps to this area. Enter a pixel value in the Snap If Within text box.
8. Click the Advanced tab.
9. Designate the drag handle:
 - To enable the entire AP element to act as a drag handle, select Entire Element from the drop-down menu.
 - To limit the area to be used as a drag handle, select Area Within Element from the drop-down menu. Enter the Left and Top coordinates and the Width and Height dimensions in the appropriate text boxes.
10. To keep the AP element in its current depth and not bring it to the front, clear the checkbox for While Dragging: Bring Element To The Front. To change the stacking order of the AP element when it is released after dragging, select either Leave On Top or Restore Z-Index from the drop-down list.
11. To execute a JavaScript command when the AP element is dropped on the target, enter the code in the Call JavaScript text box. If you want the script to execute every time the AP element is dropped, enter the code in the When Dropped: Call JavaScript text box. If the code should execute only when the AP element is dropped on the target, make sure the Only If Snapped checkbox is selected.
12. To change the event that triggers the action (the default is `onLoad`), select an event from the drop-down list in the Events column.

FIGURE 10-29

On this interactive page, visitors can drop merchandise into the shopping bag; this feature is made possible with the Drag AP Element action.



Set Text of Container

You've seen how AP elements can dynamically move, and change their visibility and their depth—but did you know that you can also change an AP element's content dynamically? With Dreamweaver, you can do it easily (and you're not restricted to AP elements; any containing element will do). A standard behavior, Set Text Of Container, enables you to swap the entire contents of one AP element for whatever you'd like. You're not limited to exchanging just text, either. Anything you can put into HTML, you can swap—which is pretty much everything!

Targeted JavaScript Commands

You can enter the following simple yet useful JavaScript commands in the Snap JavaScript text box of the Drag AP Element dialog box:

- To display a brief message to the user after the AP element is dropped, use the `alert()` function:
`alert("You hit the target")`
- To send the user to another web page when the AP element is dropped in the right location, use the JavaScript location object:
`location = "http://www.yourdomain.com/yourpage.html"`

The location object can also be used with relative URLs.

This behavior is extremely useful for putting up context-sensitive help and other information. Rather than construct a series of AP elements that you show and hide, a single AP element is used, and just the contents change. To use Set Text Of Container, follow these steps:

1. Insert and name your AP elements as desired.
2. Select the graphic, button, or text link you'd like to act as the trigger for changing the content of the AP element.
3. Choose Window ⇨ Behaviors or press Shift+F4 to open the Behaviors panel.
4. Choose Set Text ⇨ Set Text of Container from the Add (+) drop-down list. The Set Text of Container dialog box (shown in Figure 10-30) shows a list of the available elements with an ID in the current web page and provides a space for the new content.
5. Select the ID element you want to alter from the Container drop-down list.
6. Enter the text or code in the New HTML text area. You can enter either plain text, which is rendered in the default paragraph style, or any amount of HTML code, including ``, `<table>`, or other tags.

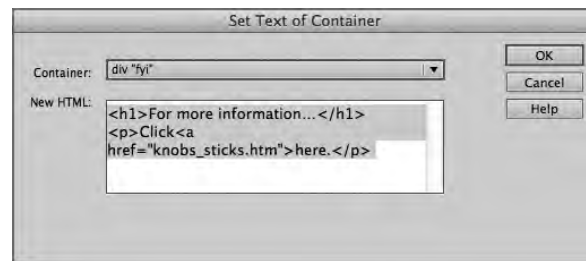
TIP

If you're entering a large amount of HTML, don't bother doing so by hand—Dreamweaver can do it for you. On a blank page, create your HTML content and then select it and switch to Code view to copy it. Then, in the Set Text of Container dialog box, paste the code using Ctrl+V (Command+V).

7. Click OK when you're finished.

FIGURE 10-30

Swap out all the contents of an ID element using the Set Text Of Container behavior.



If you want several ID elements to change when a single event is triggered, just add more Set Text Of Container behaviors to the same object.

NOTE

You may need to change the behavior event from its default; to do so, click the down arrow in between the Event and the Action columns on the Behaviors panel and choose a new event from the list.

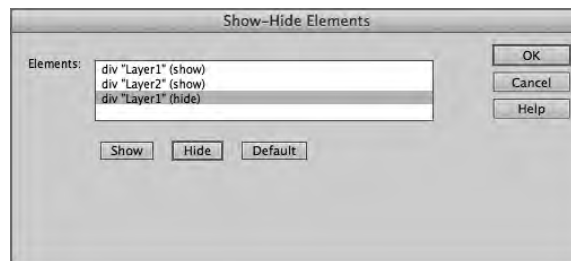
Show-Hide Elements

The capability to implement interactive control of an element's visibility offers tremendous potential to the web designer. The Show-Hide Elements action makes this implementation straightforward and simple to set up. With the Show-Hide Elements action, you can simultaneously show one or more elements while hiding as many other elements as necessary. Create your elements and give them a unique ID before invoking the Show-Hide Elements action. To use Show-Hide Elements, follow these steps:

1. Select an image, link, or other HTML tag to which to attach the behavior.
2. Choose Window ⇄ Behaviors or press Shift+F4 to open the Behaviors panel.
3. Choose Show-Hide Elements from the Add (+) drop-down list. The Show-Hide Elements dialog box (see Figure 10-31) shows a list of the available AP elements in the open web page.

FIGURE 10-31

With the Show-Hide Elements behavior attached, you can easily program the visibility of all the AP elements in your web page.



4. To cause a hidden AP element to be revealed when this event is fired, select the AP element from the list and click the Show button.
5. To hide a visible AP element when this event is fired, select its ID from the list and click the Hide button.
6. To restore an AP element's default visibility value when this event is fired, select the AP element and click the Default button.
7. Click OK when you are finished.
8. If the default event is not suitable, use the drop-down list in the Events column to select a different one.

Summary

AP elements are effective placement tools for developing the layout of a page. Anyone used to designing with desktop publishing tools can quickly learn to work with `<div>` tags and AP elements effectively. Keep these points in mind:

- A `<div>` tag can be styled for layout purposes using CSS.
- Dreamweaver calls a drawn out `<div>` tag with embedded CSS styling an AP element.
- AP elements are visible only on fourth-generation and later browsers.
- Dreamweaver includes a number of visual aids to help you create a `<div>`-based layout.
- The fluid grid layout makes it possible to design for multiple screens simultaneously.
- AP elements can be used to place HTML content anywhere on a web page.
- You can stack AP elements on top of one another. This depth control is referred to as the stacking order or the z-index.
- AP elements can be constructed so that the end user can display or hide them interactively, or alter their position, size, and depth dynamically.
- Dreamweaver provides guides, rulers, and grids to help with AP element placement and alignment.
- You can easily activate AP elements by using Dreamweaver's built-in JavaScript behaviors.

In the next chapter, you learn how to use Dreamweaver behaviors to enhance the interactivity of your sites.

