New Perspectives on HTML5, CSS3, and JavaScript

Tutorial Three: Designing a Page Layout

A Guide to this Instructor's Manual:
We have designed this Instructor's Manual to supplement and enhance your teaching experience through classroom activities and a cohesive chapter summary.

This document is organized chronologically, using the same headings in **blue** that you see in the textbook. Under each heading you will find (in order): Lecture Notes that summarize the section, Figures and Boxes found in the section (if any), Teacher Tips, Classroom Activities, and Lab Activities. Pay special attention to teaching tips and activities geared toward quizzing your students, enhancing their critical thinking skills, and encouraging experimentation within the software.

In addition to this Instructor's Manual, our Instructor Companion Site also contains PowerPoint Presentations, Test Banks, and other supplements to aid in your teaching experience.

Table of Contents

Tutorial Objectives

Students will have mastered the material in Tutorial Three when they can:

Session 3.1
- Create a reset style sheet
- Explore page layout designs
- Center a block element
- Create a floating element
- Clear a floating layout
- Prevent container collapse

Session 3.2
- Explore grid-based layouts

- Create a layout grid
- Format a grid
- Explore the CSS grid styles

Session 3.3
- Explore positioning styles
- Work with relative positioning
- Work with absolute positioning
- Work with overflow content

Introducing the `display` Style

LECTURE NOTES
- Define the syntax to display style for a page element.
- Discuss the different values of the **`display`** property.

BOXES
- Tip: You also can hide elements by applying the style `visibility: hidden;`, which hides the element content but leaves the element still occupying the same space in the page (HTML 172).

FIGURES
- Figure 3-1

TEACHER TIP
Using the Figure 3-1, discuss with the students the various values of the `display` property. Be sure to point out that students can prevent browsers from displaying an element by setting its display property to none.

CLASSROOM ACTIVITIES
- Quick Quiz:
  - True/False: When the `display` property of an element is set to `none`, the element is not a part of the document structure. (Answer: False)
  - True/False: The display value `list-item` displays the elements as a list item along with a bullet marker. (Answer: True)

LAB ACTIVITY
- None

Creating a Reset Style Sheet
LECTURE NOTES
- Explain the usage of a reset style sheet.
- Show how to create a reset style sheet.

BOXES
- Tip: The reset style sheet should always be the first style sheet listed before any other style sheets to ensure that your default styles are applied first (HTML 175).

FIGURES
- Figure 3-2, Figure 3-3, Figure 3-4

TEACHER TIP
Students may not be familiar with reset style sheets. Be sure to discuss the benefits of resetting the styles. Also, emphasize the fact that some style sheets may contain more style rules than actually needed. In order to speed up the website, one should pare down the reset style sheet to use only the elements needed for the website.

CLASSROOM ACTIVITIES
- Class Discussion: Ask students about the benefits and the drawbacks of resetting a style sheet.

LAB ACTIVITY
- Student Lab:
  - Ask students to refer to Figure 3-4 and design a home page for an e-commerce company. The page layout of their final product should look similar to Figure 3-4.

Exploring Page Layout Designs
LECTURE NOTES
- Discuss the challenges of designing web page layouts which will be viewed at different screen resolutions.
- Explain the following web page layouts:
  - Fixed layout
  - Fluid layout
  - Elastic layout
- Explain the concept of responsive design.

BOXES
- None

FIGURES

- Figure 3-5

CLASSROOM ACTIVITIES
- Class Discussion: Ask students to compare between fixed and fluid layouts. Ask them to list the advantages and disadvantages of all the layouts.

LAB ACTIVITY
- None

Working with Width and Height
LECTURE NOTES
- Define the syntax to set the width and height of an element.
- Define the syntax to set the maximum and minimum limits on the width and height of a block element.
- Explain the procedure and syntax to set block elements in the center, both vertically and horizontally.

BOXES
- Reference: Setting Width and Heights (HTML 179)
- Insight: Working with Element Heights (HTML 182)

FIGURES
- Figure 3-6, Figure 3-7, Figure 3-8

CLASSROOM ACTIVITIES
- Class Discussion: Ask students about the several methods to overcome the challenges of vertical centering an element.

LAB ACTIVITY
- Ask the students to open the home page file they had created in the previous lab. Set the width of the page body to 80% of the browser window ranging from 460 pixels to 1280 pixels. Ask them to design a company logo with its width set to 100%.
- Ask the students to work on the same file and center the home page horizontally.

Floating Page Content

LECTURE NOTES
- Explain the concept of floating an element.
- Discuss the syntax and role of the clear property in HTML.
- Discuss the concepts of the following box models:
  - Content box model
  - Border box model
- Describe the occurrence of container collapse situation with an example.
- Explain the purpose of after pseudo-element.

BOXES
- Reference: Floating an Element (HTML 184)
- Reference: Clearing a Float (HTML 187)
- Tip: Height values are similarly affected by the type of layout model used (HTML 192).
- Reference: Defining How Widths Are Interpreted (HTML 192)
- Tip: To find other ways to prevent container collapse, search the web using the keywords *CSS clearfix* (HTML 196).
- Reference: Keeping a Container from Collapsing (HTML 197)
- Proskills: Problem Solving: The Virtue of Being Negative (HTML 198)

FIGURES
- Figure 3-9, Figure 3-10, Figure 3-11, Figure 3-12, Figure 3-14, Figure 3-15, Figure 3-16, Figure 3-17, Figure 3-18, Figure 3-19, Figure 3-20, Figure 3-21, Figure 3-22, Figure 3-23, Figure 3-24, Figure 3-25, Figure 3-26, Figure 3-27, Figure 3-28

TEACHER TIP
Remind the students that the combined width of the elements to be placed within a single row cannot exceed the total width of their parent element. Have a discussion regarding the different scenarios that might occur, as a result of exceeding the boundaries of the container.

CLASSROOM ACTIVITIES
- Class Discussion: Ask students to note down the advantages and disadvantages of floating an element over positioning them.

LAB ACTIVITY
- Ask the students to create a navigation list with hypertext links. Then, ask them to change the links to block elements and center the link text within each block.
- Ask them to open the home page they had designed in the earlier lab. Ask them to create two sections named Left and Right. Set the width of the left section to 39% of the body width and set the width of the right section to 51%. Ask them to float the sections side-by-side on the left margin.

Introducing Grid Layouts

LECTURE NOTES
- Discuss the concept of grid-based layouts.
- Differentiate between fixed and fluid grids.
- Define framework and the most popular CSS frameworks.

BOXES
- None

FIGURES
- Figure 3-29, Figure 3-30

TEACHER TIP
Remind the students that the grid is not a part of the web page content. Have a discussion pointing out the several aesthetic and practical advantages of working using a grid.

CLASSROOM ACTIVITIES
- Class Discussion: Ask students to find out the advantages of using a CSS framework such as Bootstrap over manually designing grids.

LAB ACTIVITY
- Ask the students to create a grid system for a desktop layout without using any web-based framework. Later, ask them to design the same using a CSS framework.

Setting up a Grid

LECTURE NOTES
- Discuss the role of `div` element in creating a grid.
- Discuss the procedure and the syntax to design grid rows and columns.
- Discuss the procedure to add page content in a grid.

BOXES
- Tip: The class name row for grid rows is not mandatory; you can choose a different class name for your own grid rows (HTML 208).
- Tip: For a review of attribute selectors, and specifically `elem[att^="text"]`, refer to Figure 2-15 in Tutorial 2 (HTML 209).

- Tip: Choose percent values for the column widths so that the total width of all of the columns in the row does not exceed 100% (HTML 210).
- Insight: Generating Content with Lorem Ipsum (HTML 216)

FIGURES
- Figure 3-31, Figure 3-32, Figure 3-33, Figure 3-34, Figure 3-35, Figure 3-36, Figure 3-37, Figure 3-38, Figure 3-39, Figure 3-40, Figure 3-41, Figure 3-42, Figure 3-43

TEACHER TIP
Prepare few examples of websites using grids to design the web page. Have few other examples of web pages without using grid. Ask the students to compare the appearance of the page content and list out the difference found.

CLASSROOM ACTIVITIES
- Quick Quiz:
  - True/False: A column cannot contain its own grid of rows and columns. (Answer: False)
  - True/False: The actual column widths are defined in style sheets. (Answer: True)
- Class Discussion: Ask students their views on lorem ipsum and if it is important to use this tool.

LAB ACTIVITY
- Ask the students to open the company home page they had designed in the earlier labs and perform the following tasks:
  - Design a layout of three main rows. The first row contains the page title and the third row contains the page footer.
  - The second row consists of two columns—the first column displays information about the company, and the second column displays a list of frequently asked questions.
  - Within the first row, they should design a nested 2 x 2 grid containing short articles about their company. Their final product should look something similar to Figure 3-32.
- Ask the students to work on the file they had created in the last lab and ask them to modify it by adding page content to each of its rows and columns.

Outlining a Grid
LECTURE NOTES
- Discuss the concept of outlining a grid.
- Explain the properties of outlining a grid.

BOXES
- Tip: Outlines can also be applied to inline elements such as inline images, citations, quotations, and italicized text (HTML 217).
- Reference: Adding an outline (HTML 217)
- Insight: Creating Drop Caps with CSS (HTML 218)

FIGURES
- Figure 3-44, Figure 3-45

CLASSROOM ACTIVITIES
- Quick Quiz:
  - True/False: Outlines cannot be applied to inline elements. (Answer: False)
  - True/ False: An outline always surrounds an entire element. (Answer: True)

LAB ACTIVITY
- Ask the students to open the company's home page that they had designed in the earlier lab. Then, ask them to modify the web page by outlining every `div` element in the grid.

Introducing CSS Grids

LECTURE NOTES
- Define the general syntax to create a CSS grid.
- Discuss the various ways to place an element within a grid cell.

BOXES
- Reference: Defining Grids with CSS (HTML 221)
- Proskills: Written Communication: Getting to the Point with Layout (HTML 222)

FIGURES
- None

CLASSROOM ACTIVITIES
- Discuss the problems one faces while using grids.

LAB ACTIVITY
- None

Positioning Objects

LECTURE NOTES
- Discuss the CSS positioning style properties.
- Explain the different types of positioning supported by CSS.

- Explain with an example how to use positioning styles in an HTML document.

BOXES
- Tip: To place an element at the bottom right corner of its container, use absolute positioning with the right and bottom values set to 0 pixels (HTML 227).
- Tip: You can work with an interactive demo of positioning styles using the demo_positioning.html file from the demo folder (HTML 228).
- Tip: If all of the objects within a container are placed using absolute positioning, the container will have no content and will collapse (HTML 229).
- Reference: Positioning Objects with CSS (HTML 230)
- Insight: Creating an Irregular Line Wrap (HTML 240)

FIGURES
- Figure 3-46, Figure 3-47, Figure 3-48, Figure 3-49, Figure 3-50, Figure 3-51, Figure 3-52, Figure 3-53, Figure 3-54, Figure 3-55, Figure 3-56, Figure 3-57, Figure 3-58

TEACHER TIP
Provide several visual aids to help students understand the difference between each positioning styles. Use cutouts on a flipchart. Move the cutouts around to demonstrate what effect different types of positioning will have on an object.

CLASSROOM ACTIVITIES
- Quick Quiz:
  - To fix an object within the browser window so that it doesn't scroll, the position property must be set to _____. (Answer: fixed)
  - Which positioning property is the same as not using any CSS positioning at all? (Answer: Static)
- Class Discussion: Positioning gives you a great deal of control over how your web pages look. Do some websites overuse positioning? Will the students like a website that is filled with eye-popping boxes of text or something that flows in a more linear style? Debate the pros and cons of excess positioning. Show an example of a website that has excess of positioning styles.

LAB ACTIVITY
- Ask the students to design a web page with three Sports as headings. All the headings must be horizontally placed against each other. Ask them to display an image and write an advantage and disadvantage of that sport. Ask them to use CSS positioning styles to vertically align the images.

Handling Overflow
LECTURE NOTES
- Discuss the role of overflow property in controlling the excess content in a browser.
- Discuss the different types of overflow properties:
  - visible
  - hidden

- o   scroll
- o   auto

BOXES
- Reference: Working with Overflow (HTML 241)
- Insight: Managing White Space with CSS (HTML 243)

FIGURES
- Figure 3-59, Figure 3-60, Figure 3-61

TEACHER TIP
Share with the students the overflow properties by CSS3 to handle horizontal and vertical directions specifically.

CLASSROOM ACTIVITIES
- Quick Quiz:
  - o   Which overflow property is browser default? (Answer: `visible`)
  - o   Which type of overflow property adds scroll bars only as they are needed? (Answer: auto)
- Class Discussion: Ask students to come up with instances when they might want to crop an image on a web page rather than display the whole picture. Can they think of why they might want to crop it on the web page rather than using a photo editor? Have a discussion with students on how to handle overflow in horizontal and vertical directions.

LAB ACTIVITY
- Ask the students to open the file they have worked on the previous lab. Then, ask them to limit the size of the sections in such a way that the browser displays scroll bars as needed for the excess content.

Clipping an Element
LECTURE NOTES
- Define the syntax of `clip` property too view the content of an element.

BOXES
- Reference: Clipping Content (HTML 244)

FIGURES
- Figure 3-62

TEACHER TIP
Have a discussion with students regarding how to remove clipping when not required.

CLASSROOM ACTIVITIES
- Quick Quiz:
  - True/ False: `clip: auto.` This code is used to remove clipping completely. (Answer: True)
  - True/False: Clipping can only be applied when the object is placed using relative positioning. (Answer: False)

LAB ACTIVITY
- None

Stacking Elements
LECTURE NOTES
- Define the syntax of `z-index` style.
- Discuss the limitations of `z-index` property.

BOXES
- Proskills: Problem Solving: Principles of Design (HTML 245)

FIGURES
- Figure 3-63, Figure 3-64

TEACHER TIP
Provide few examples of how using the `z-index` style can be helpful to create layered effects. Remind the students that this property works only for elements which are placed with absolute positioning.

CLASSROOM ACTIVITIES
- Quick Quiz:
  - True/False: To alter the default stacking order, use the `z-index` style. (Answer: True)
  - True/False: `z-index` property works for all the elements, regardless of the type of positioning. (Answer: False)
- Class Discussion: Ask students to think about ways they might layer items on a page using the `z-index`.

LAB ACTIVITY
- Ask students to create a collage of pictures and text. Have them use the different positioning styles to layer the images across the page in a collage style. Have them display text boxes over different parts of the page.

End of Tutorial Material

- **Review Assignments:** Review Assignments provide students with additional practice of the skills they learned in the tutorial using the same tutorial case with which they are already familiar.
- **Case Problems:** A typical NP tutorial has four Case Problems following the Review Assignments. Short tutorials can have fewer Case Problems (or none at all); other tutorials may have five Case Problems. The Case Problems provide further hands-on assessment of the skills and topics presented in the tutorial, but with new case scenarios. There are three types of Case Problems:
  - **Apply**. In this type of Case Problem, students apply the skills that they have learned in the tutorial to solve a problem.
  - **Create**. In a Create Case Problem, students are either shown the end result, such as a finished website, and asked to create the document based on the figure provided or asked to create something from scratch in a more free-form manner.
  - **Challenge**. A Challenge Case Problem involves three or more Explore steps. These steps challenge students by having them go beyond what was covered in the tutorial, either with guidance in the step or by using online Help as directed.
- **ProSkills Exercises:** This feature is new for Office 2010 and Windows 7. ProSkills exercises integrate the technology skills students learn with one or more of the following soft skills: decision making, problem solving, teamwork, verbal communication, and written communication. The goal of these exercises is to enhance students' understanding of the soft skills and how to apply them appropriately in real-world, professional situations that also involve software application skills. ProSkills exercises are offered at various points throughout a text, encompassing the concepts and skills presented in a standalone tutorial or a group of related tutorials.

Glossary

- border box model (HTML 191)
- content box model (HTML 191)
- container collapse (HTML 195)
- drop cap (HTML 218)
- elastic layout (HTML 177)
- fixed (HTML 230)
- fixed grid (HTML 203)
- fixed layout (HTML 176)
- floating (HTML 183)
- fluid layout (HTML 177)
- fluid grid (HTML 203)
- framework (HTML 204)
- `fr` unit (HTML 219)
- grid cell (HTML 220)
- inherit (HTML 230)
- lorem ipsum (HTML 216)
- reset style sheet (HTML 172)
- responsive design (HTML 177)
- static positioning (HTML 226)

Top of Document

Glossary

- border box model (HTML 191)
- framework (HTML 204)