# HTML5 and CSS3

7th Edition

INTRODUCTORY

## Tutorial 3

## Designing a Page Layout

Carey

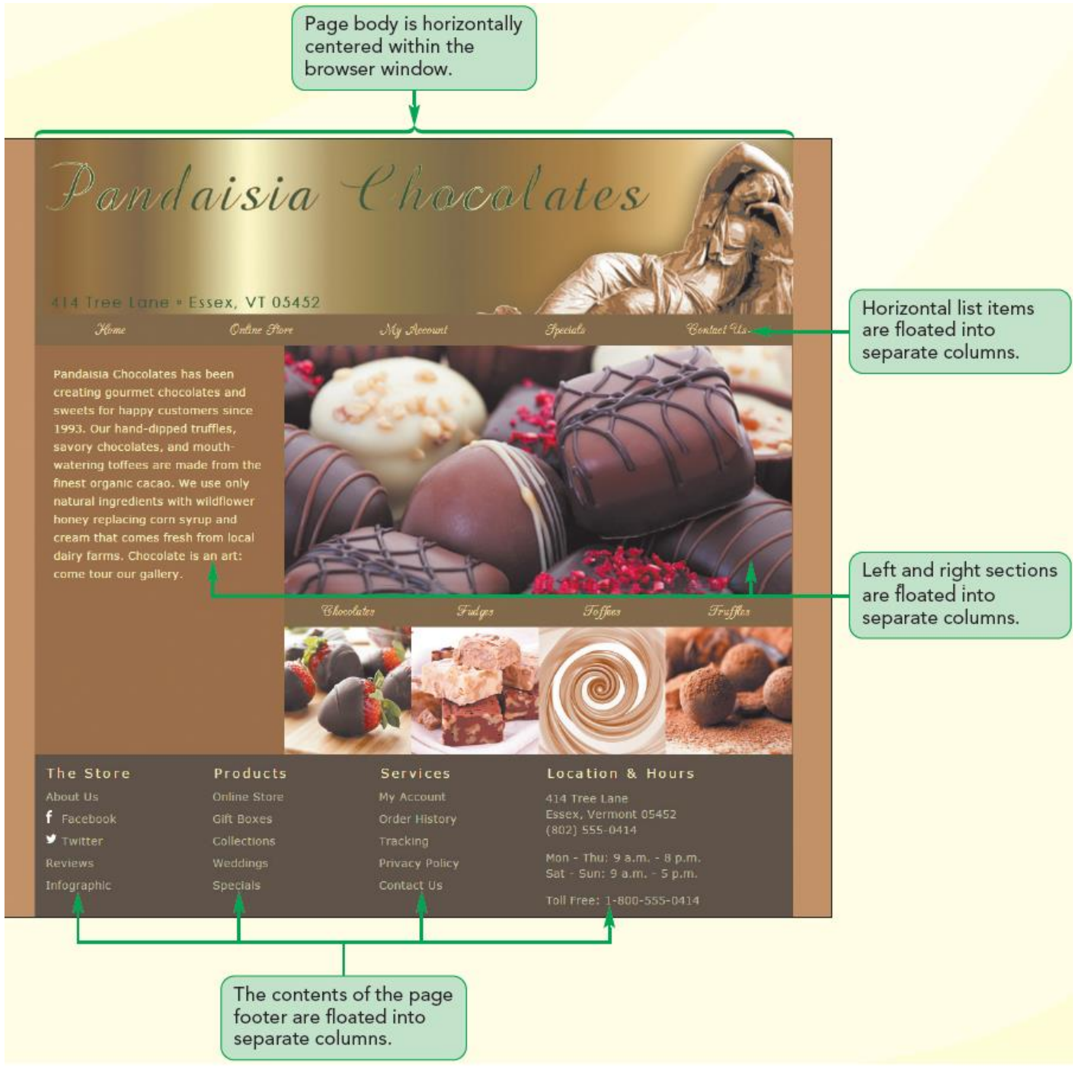
## Objectives

- Create a reset style sheet
- Explore page layout designs
- Center a block element
- Create a floating element
- Clear a floating layout
- Prevent container collapse
- Explore grid-based layouts

## Objectives (continued)

- Create a layout grid
- Format a grid
- Explore the CSS grid styles
- Explore positioning styles
- Work with relative positioning
- Work with absolute positioning
- Work with overflow content

# Page Layout with Floating Elements


Page body is horizontally centered within the browser window.
Horizontal list items are floated into separate columns.
Left and right sections are floated into separate columns.
The contents of the page footer are floated into separate columns.
Pandaisia Chocolates
414 Tree Lane • Essex, VT 05452
Home
Online Store
My Account
Specials
Contact Us
Chocolates
Fudges
Toffees
Truffles

Pandaisia Chocolates has been creating gourmet chocolates and sweets for happy customers since 1993. Our hand-dipped truffles, savory chocolates, and mouth-watering toffees are made from the finest organic cacao. We use only natural ingredients with wildflower honey replacing corn syrup and cream that comes fresh from local dairy farms. Chocolate is an art: come tour our gallery.

| The Store | Products | Services | Location & Hours |
|---|---|---|---|
| About Us | Online Store | My Account | 414 Tree Lane |
| Facebook | Gift Boxes | Order History | Essex, Vermont 05452 |
| Twitter | Collections | Tracking | (802) 555-0414 |
| Reviews | Weddings | Privacy Policy | Mon - Thu: 9 a.m. - 8 p.m. |
| Infographic | Specials | Contact Us | Sat - Sun: 9 a.m. - 5 p.m. |
| | | | Toll Free: 1-800-555-0414 |

## Introducing the `display` Style

- HTML elements are classified into
  - Block elements, such as paragraphs or headings
  - Inline elements, such as emphasized text or inline images
- The display style can be defined for any page element using

```
display: type;
```

where *`type`* defines the display type

## Introducing the `display` Style (continued)

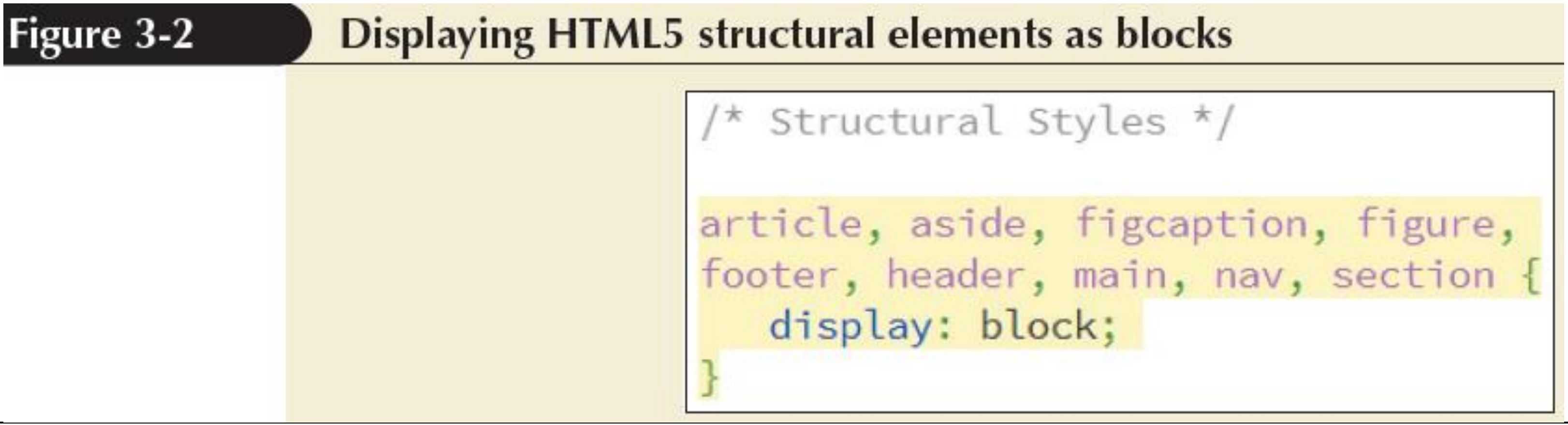
Figure 3-1 Some values of the display property

| Display Value | Appearance |
|---|---|
| `block` | Displayed as a block |
| `table` | Displayed as a web table |
| `inline` | Displayed in-line within a block |
| `inline-block` | Treated as a block placed in-line within another block |
| `run-in` | Displayed as a block unless its next sibling is also a block, in which case, it is displayed in-line, essentially combining the two blocks into one |
| `inherit` | Inherits the display property of the parent element |
| `list-item` | Displayed as a list item along with a bullet marker |
| `none` | Prevented from displaying, removing it from the rendered page |

## Creating a Reset Style Sheet

- **Reset style sheet** supersedes a browser's default styles and provides a consistent starting point for page design
- The first style rule in a sheet is the `display` property used to display HTML5 structural elements

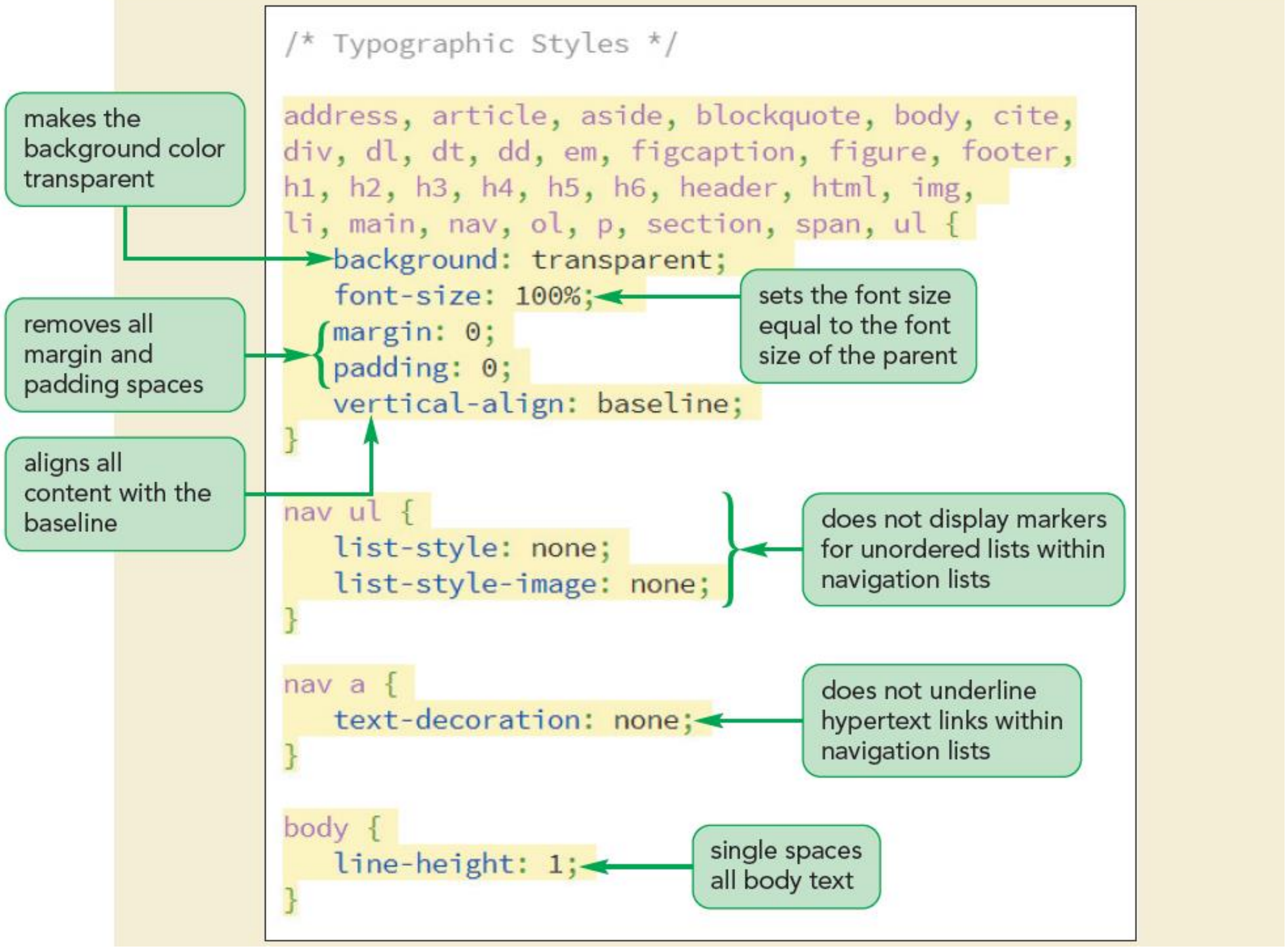
Figure 3-2 Displaying HTML5 structural elements as blocks



```css
/* Structural Styles */

article, aside, figcaption, figure,
footer, header, main, nav, section {
   display: block;
}
```

# Creating a Reset Style Sheet (continued)

Figure 3-3 Completing the reset style sheet



```
/* Typographic Styles */

address, article, aside, blockquote, body, cite,
div, dl, dt, dd, em, figcaption, figure, footer,
h1, h2, h3, h4, h5, h6, header, html, img,
li, main, nav, ol, p, section, span, ul {
   background: transparent;
   font-size: 100%;
   margin: 0;
   padding: 0;
   vertical-align: baseline;
}

nav ul {
   list-style: none;
   list-style-image: none;
}

nav a {
   text-decoration: none;
}

body {
   line-height: 1;
}
```

makes the background color transparent
removes all margin and padding spaces
aligns all content with the baseline
sets the font size equal to the font size of the parent
does not display markers for unordered lists within navigation lists
does not underline hypertext links within navigation lists
single spaces all body text

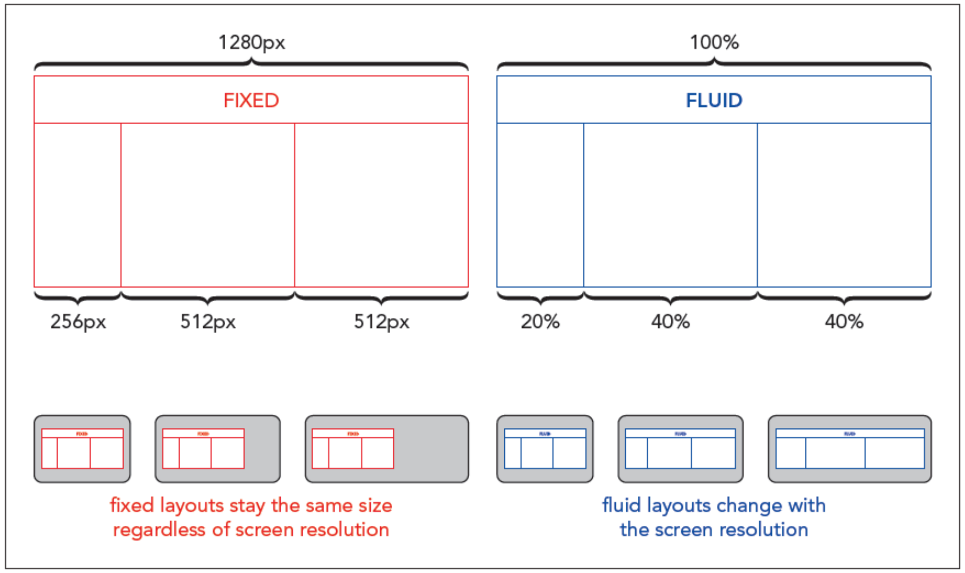
## Exploring Page Layout Designs

- Web page layouts fall into three categories:
  - **Fixed layout** – Size of the page and page elements are fixed, usually using pixels as the unit of measure
  - **Fluid layout** – The width of the page elements are set as a percent of the available screen width
  - **Elastic layout** – Images and text are always sized in proportion to each other in em units

## Exploring Page Layout Designs (continued)

- **Responsive design** – The layout and design of a page changes in response to the device that is rendering it

Figure 3-5 Fixed layouts vs. fluid layouts


1280px
FIXED
256px
512px
512px
100%
FLUID
20%
40%
40%
fixed layouts stay the same size regardless of screen resolution
fluid layouts change with the screen resolution

## Working with Width and Height

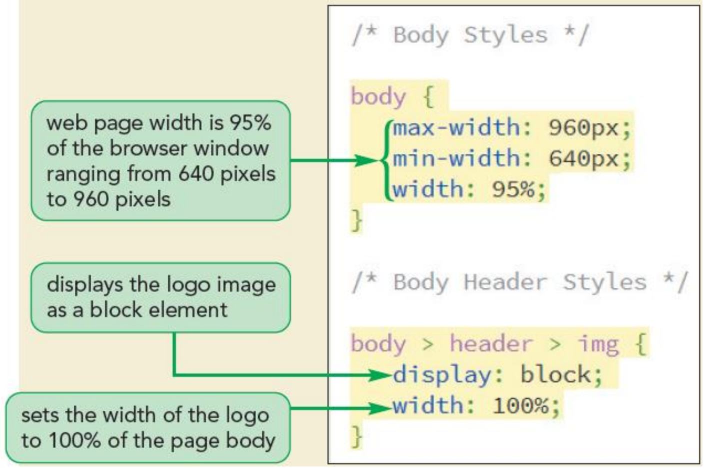
- The width and height of an element are set using the following properties:

```
width: value;
height: value;
```

where *value* is the width or height using one of the CSS units of measurement or as a percentage of the width or height of the parent element

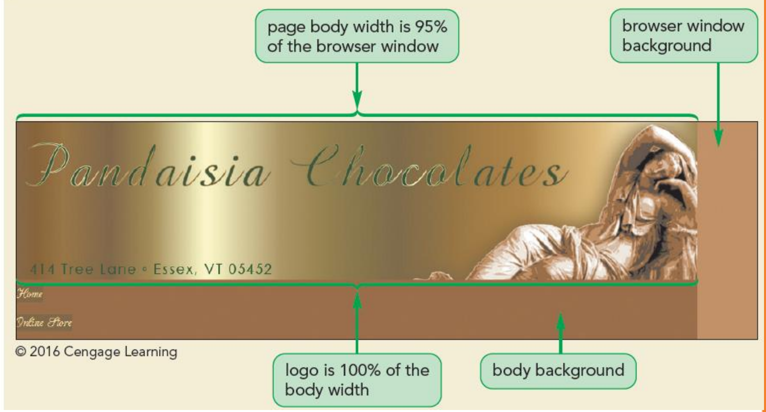
## Working with Width and Height (continued)

Figure 3-6 Setting the width of the page body and logo


web page width is 95% of the browser window ranging from 640 pixels to 960 pixels
displays the logo image as a block element
sets the width of the logo to 100% of the page body

```
/* Body Styles */

body {
   max-width: 960px;
   min-width: 640px;
   width: 95%;
}

/* Body Header Styles */

body > header > img {
   display: block;
   width: 100%;
}
```

Figure 3-7 Initial view of the body header


page body width is 95% of the browser window
browser window background
Pandaisia Chocolates
414 Tree Lane • Essex, VT 05452
Home
Online Store
logo is 100% of the body width
body background

## Centering a Block Element

- Block elements can be centered horizontally within their parent element by setting both the left and right margins to `auto`

```
body {
     margin-left: auto;
     margin-right: auto;
}
```

## Vertical Centering (continued)

- Centering an element vertically can be accomplished by displaying the parent element as a table cell and setting the `vertical-align` property to `middle`
- For example, to vertically center the following h1 heading within the `div` element:

```
<div>
    <h1>Pandaisia Chocololates</h1>
</div>
```

## Vertical Centering

- Apply the style rule

```
div {
     height: 40px;
     display: table-cell;
     vertical-align: middle;
}
```

Using this style rule, the h1 heading will be vertically centered

## Floating Page Content

- **Floating** an element takes it out of position and places it along the left or right side of its parent element
- To float an element, apply

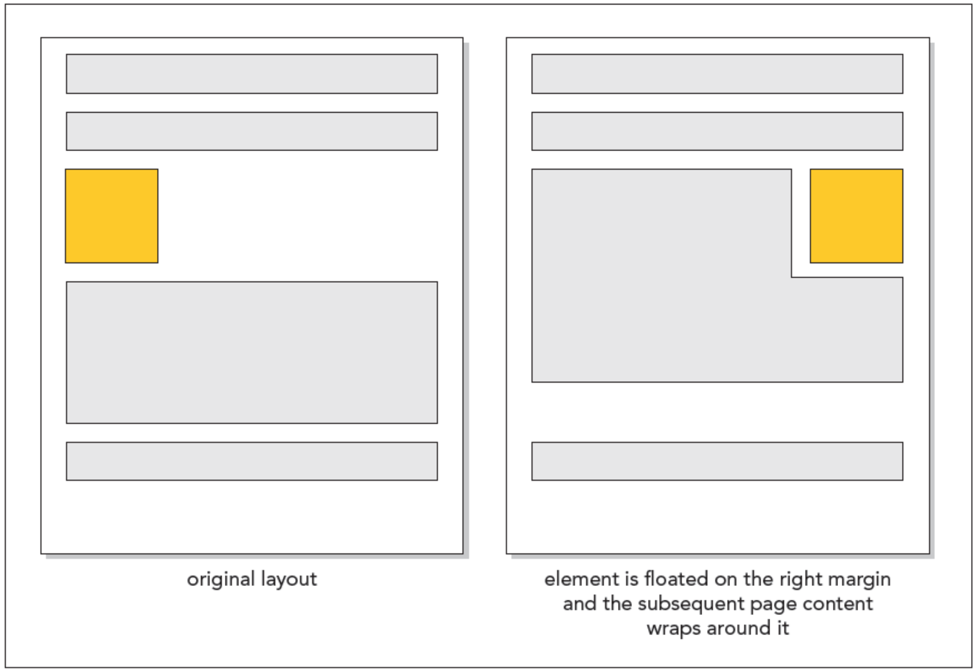
```
    float: position;
```

where *`position`* is `none` (the default), `left` to float the object on the left margin or `right` to float the object on the right margin

## Floating Page Content (continued 1)

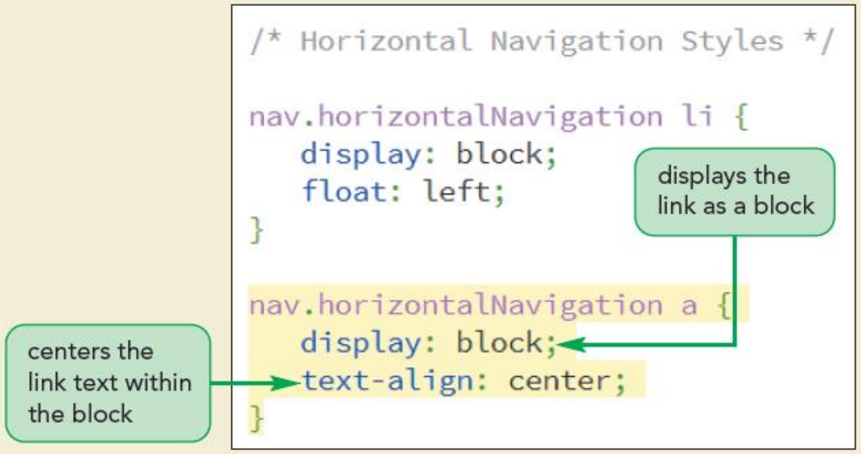
- For elements to be placed within a single row, the combined width of the elements cannot exceed the total width of their parent element

Figure 3-9 Floating an element


original layout
element is floated on the right margin and the subsequent page content wraps around it

# Floating Page Content (continued 2)

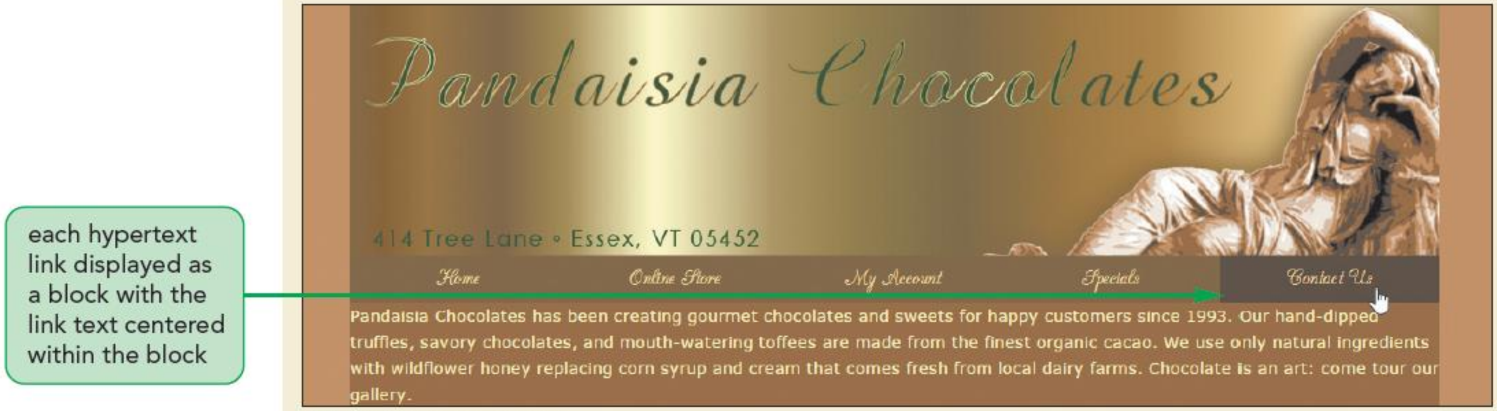
**Figure 3-13** Formatting hyperlinks in horizontal navigation lists



```
/* Horizontal Navigation Styles */

nav.horizontalNavigation li {
   display: block;
   float: left;
}

nav.horizontalNavigation a {
   display: block;
   text-align: center;
}
```

displays the link as a block
centers the link text within the block

**Figure 3-14** Links in the body header



each hypertext link displayed as a block with the link text centered within the block
Pandaisia Chocolates
414 Tree Lane • Essex, VT 05452
Home
Online Store
My Account
Specials
Contact Us

Pandaisia Chocolates has been creating gourmet chocolates and sweets for happy customers since 1993. Our hand-dipped truffles, savory chocolates, and mouth-watering toffees are made from the finest organic cacao. We use only natural ingredients with wildflower honey replacing corn syrup and cream that comes fresh from local dairy farms. Chocolate is an art: come tour our gallery.

## Clearing a Float

- To ensure that an element is always displayed below floated elements, use

  ```
  clear: position;
  ```

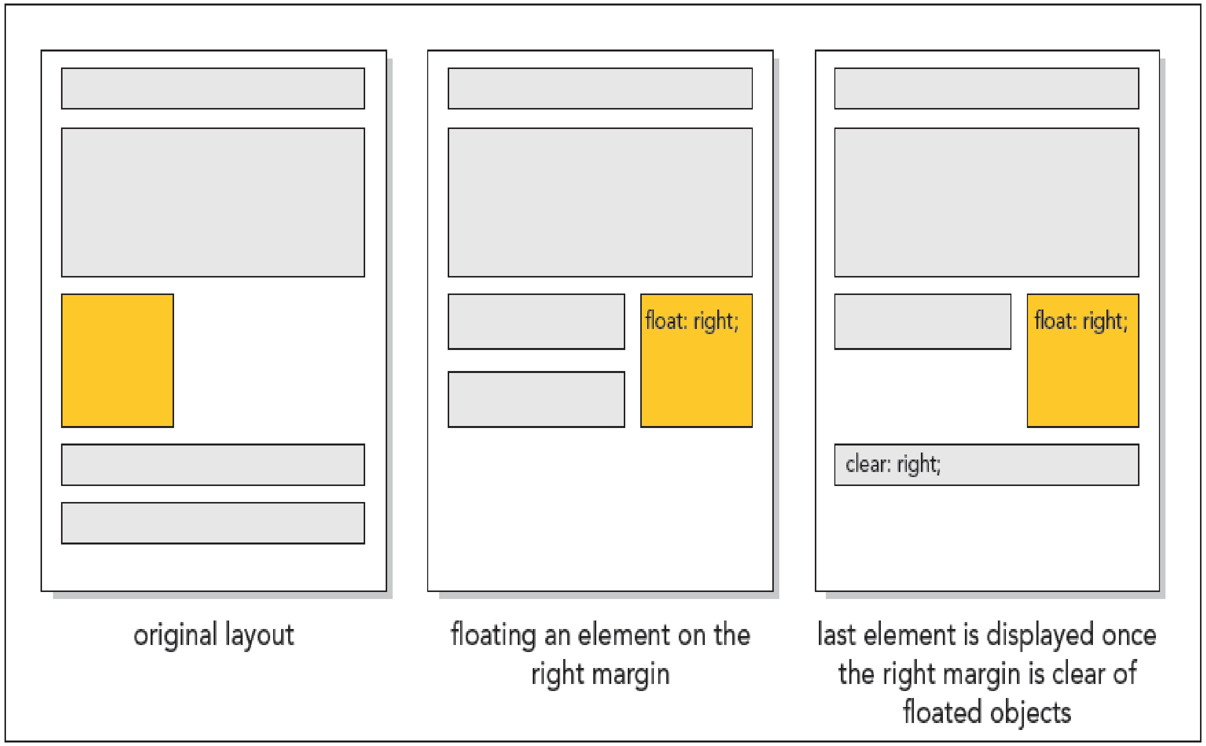
  where *`position`* is `left`, `right`, `both`, or `none`

## Clearing a Float (continued 1)

- `left` – Displays the element only when the left margin is clear of floating objects
- `right` – Displays the element only when the right margin is clear of floating objects
- `both` – Displays the element only when both margins are clear of floats
- `none` – Displays the element alongside any floated objects

# Clearing a Float (continued 2)

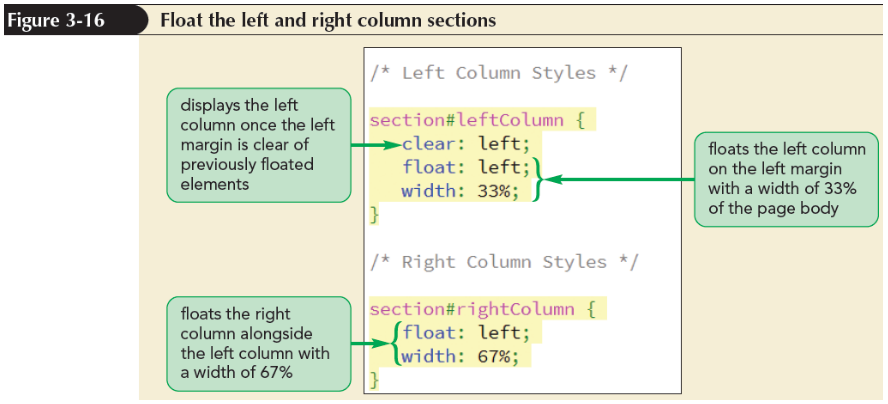
**Figure 3-15** Clearing a float


float: right;
float: right;
clear: right;
original layout
floating an element on the right margin
last element is displayed once the right margin is clear of floated objects

© 2016 Cengage Learning

# Clearing a Float (continued 3)

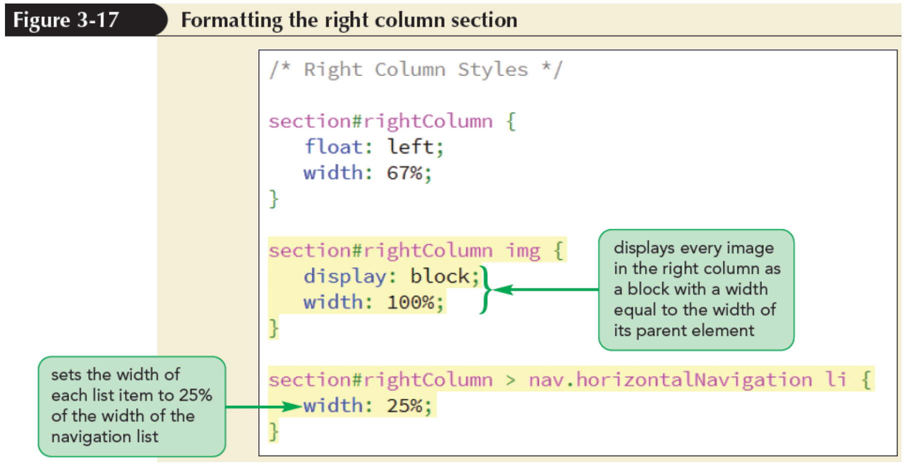
Figure 3-16 Float the left and right column sections



```
/* Left Column Styles */

section#leftColumn {
   clear: left;
   float: left;
   width: 33%;
}

/* Right Column Styles */

section#rightColumn {
   float: left;
   width: 67%;
}
```

displays the left column once the left margin is clear of previously floated elements
floats the left column on the left margin with a width of 33% of the page body
floats the right column alongside the left column with a width of 67%

Figure 3-17 Formatting the right column section

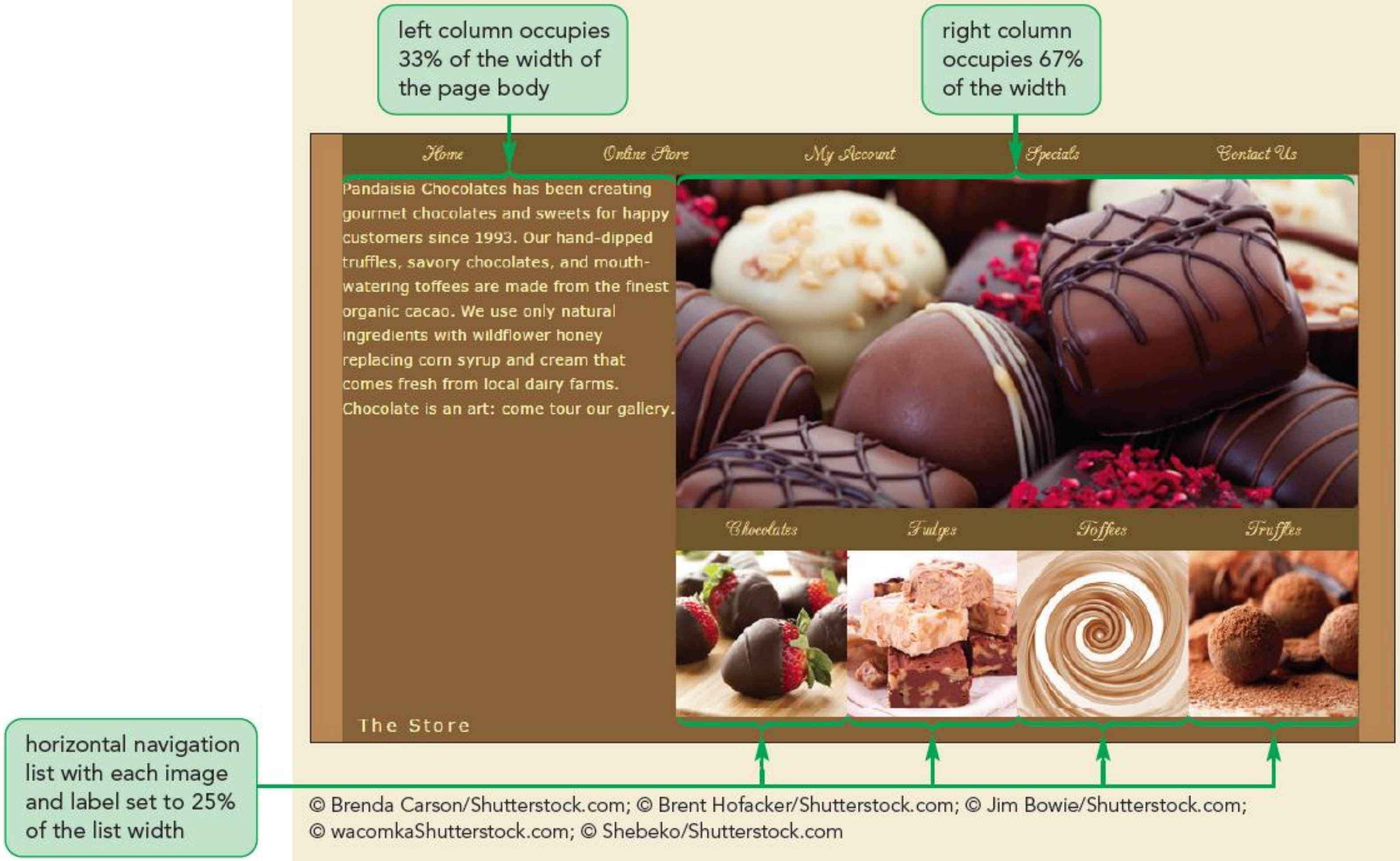


```
/* Right Column Styles */

section#rightColumn {
   float: left;
   width: 67%;
}

section#rightColumn img {
   display: block;
   width: 100%;
}

section#rightColumn > nav.horizontalNavigation li {
   width: 25%;
}
```

displays every image in the right column as a block with a width equal to the width of its parent element
sets the width of each list item to 25% of the width of the navigation list

# Clearing a Float (continued 4)

Figure 3-18 Layout of the left and right columns


left column occupies 33% of the width of the page body
right column occupies 67% of the width
Home
Online Store
My Account
Specials
Contact Us
Chocolates
Fudges
Toffees
Truffles
The Store
horizontal navigation list with each image and label set to 25% of the list width

Pandaisia Chocolates has been creating gourmet chocolates and sweets for happy customers since 1993. Our hand-dipped truffles, savory chocolates, and mouth-watering toffees are made from the finest organic cacao. We use only natural ingredients with wildflower honey replacing corn syrup and cream that comes fresh from local dairy farms. Chocolate is an art: come tour our gallery.

© Brenda Carson/Shutterstock.com; © Brent Hofacker/Shutterstock.com; © Jim Bowie/Shutterstock.com; © wacomkaShutterstock.com; © Shebeko/Shutterstock.com

## Refining a Floated Layout

- **Content box model** – The `width` property refers to the width of an element content only
  - Additional space include padding or borders
- **Border box model** – The `width` property is based on the sum of the content, padding, and border spaces
  - Additional space taken up by the padding and border is subtracted from space given to the content

## Refining a Floated Layout (continued 1)

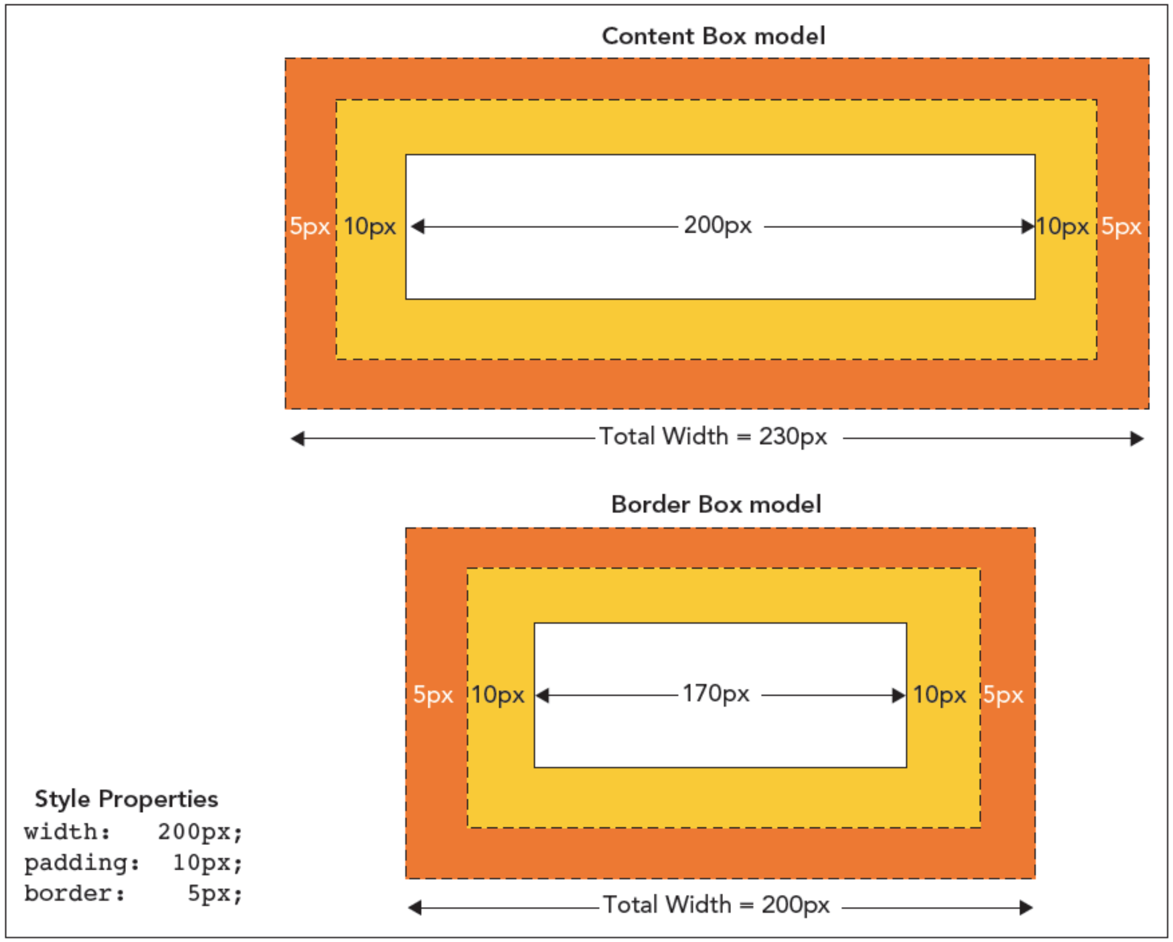
- The layout model can be chosen using

  ```
      box-sizing: type;
  ```

  where `type` is `content-box` (the default), `border-box`, or `inherit` (to inherit the property defined for the element’s container)

## Refining a Floated Layout (continued 2)

Figure 3-21 Comparing the Content Box and Border Box models



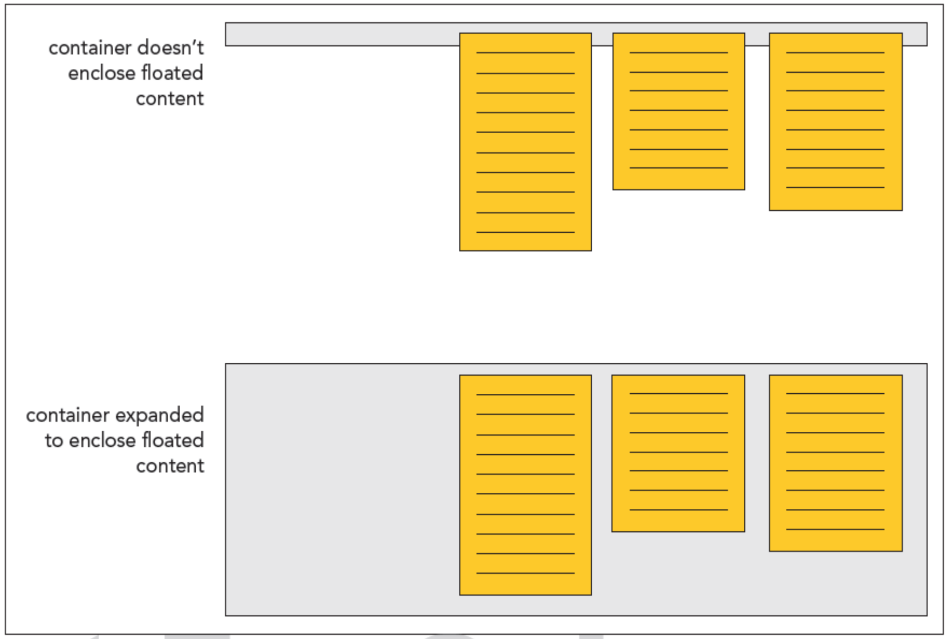
```
width:    200px;
padding:   10px;
border:     5px;
```

© 2016 Cengage Learning

## Working with Container Collapse

- **Container collapse** – An empty container with no content
  - Elements in the container are floated

Figure 3-26 Container collapse


container doesn't enclose floated content
container expanded to enclose floated content

## Working with Container Collapse (continued 1)

- Use the `after` pseudo-element to add a placeholder element after the footer
- The general style rule is

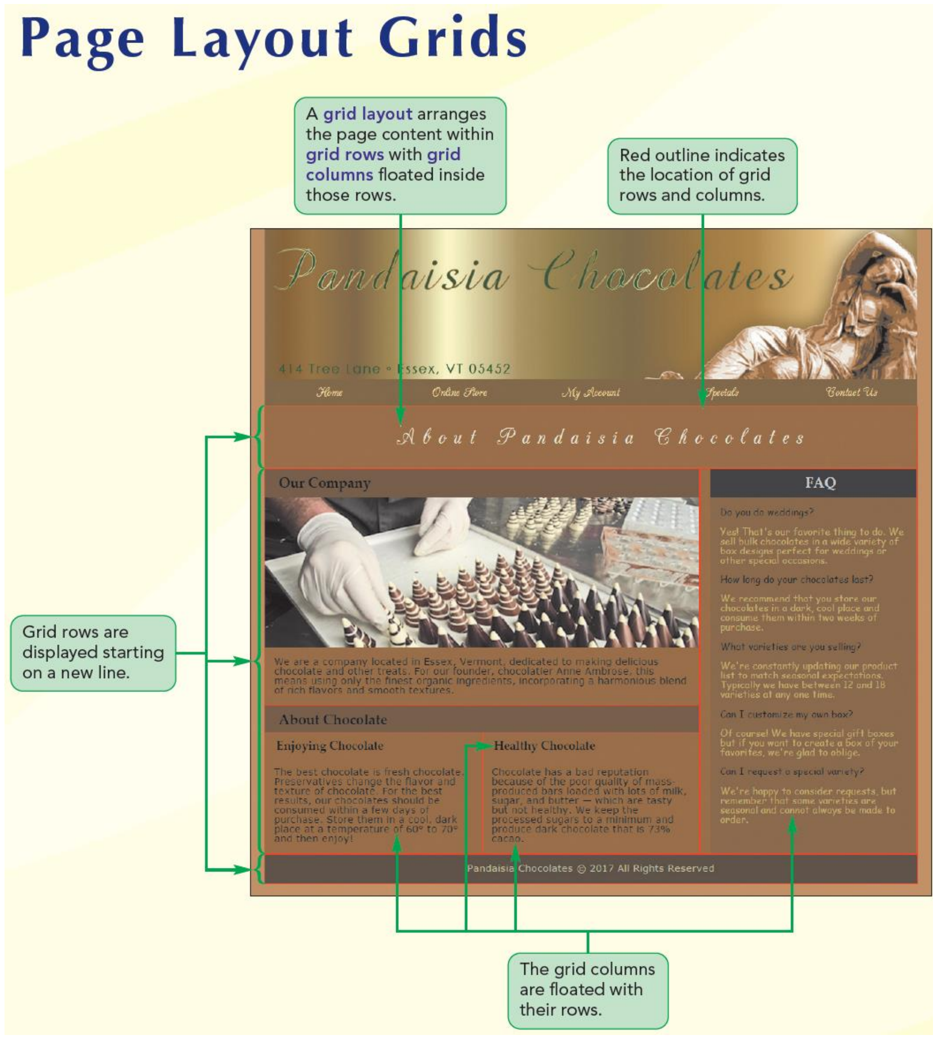
```
container::after {
      clear: both;
      content: “”;
      display: table;
}
```

where *`container`* is the selector for the element containing floating objects

## Working with Container Collapse (continued 2)

- The `clear` property keeps the placeholder element from being inserted until both margins are clear of floats
- The element itself is a web table and contains an empty text string

# Page Layout Grids


Page Layout Grids
A grid layout arranges the page content within grid rows with grid columns floated inside those rows.
Red outline indicates the location of grid rows and columns.
Grid rows are displayed starting on a new line.
The grid columns are floated with their rows.
Pandaisia Chocolates
414 Tree Lane • Essex, VT 05452
Home
Online Store
My Account
Specials
Contact Us

About Pandaisia Chocolates

Our Company

We are a company located in Essex, Vermont, dedicated to making delicious chocolate and other treats. For our founder, chocolatier Anne Ambrose, this means using only the finest organic ingredients, incorporating a harmonious blend of rich flavors and smooth textures.

About Chocolate

Enjoying Chocolate

The best chocolate is fresh chocolate. Preservatives change the flavor and texture of chocolate. For the best results, our chocolates should be consumed within a few days of purchase. Store them in a cool, dark place at a temperature of 60° to 70° and then enjoy!

Healthy Chocolate

Chocolate has a bad reputation because of the poor quality of mass-produced bars loaded with lots of milk, sugar, and butter — which are tasty but not healthy. We keep the processed sugars to a minimum and produce dark chocolate that is 73% cacao.

FAQ

Do you do weddings?

Yes! That's our favorite thing to do. We sell bulk chocolates in a wide variety of box designs perfect for weddings or other special occasions.

How long do your chocolates last?

We recommend that you store our chocolates in a dark, cool place and consume them within two weeks of purchase.

What varieties are you selling?

We're constantly updating our product list to match seasonal expectations. Typically we have between 12 and 18 varieties at any one time.

Can I customize my own box?

Of course! We have special gift boxes but if you want to create a box of your favorites, we're glad to oblige.

Can I request a special variety?

We're happy to consider requests, but remember that some varieties are seasonal and cannot always be made to order.

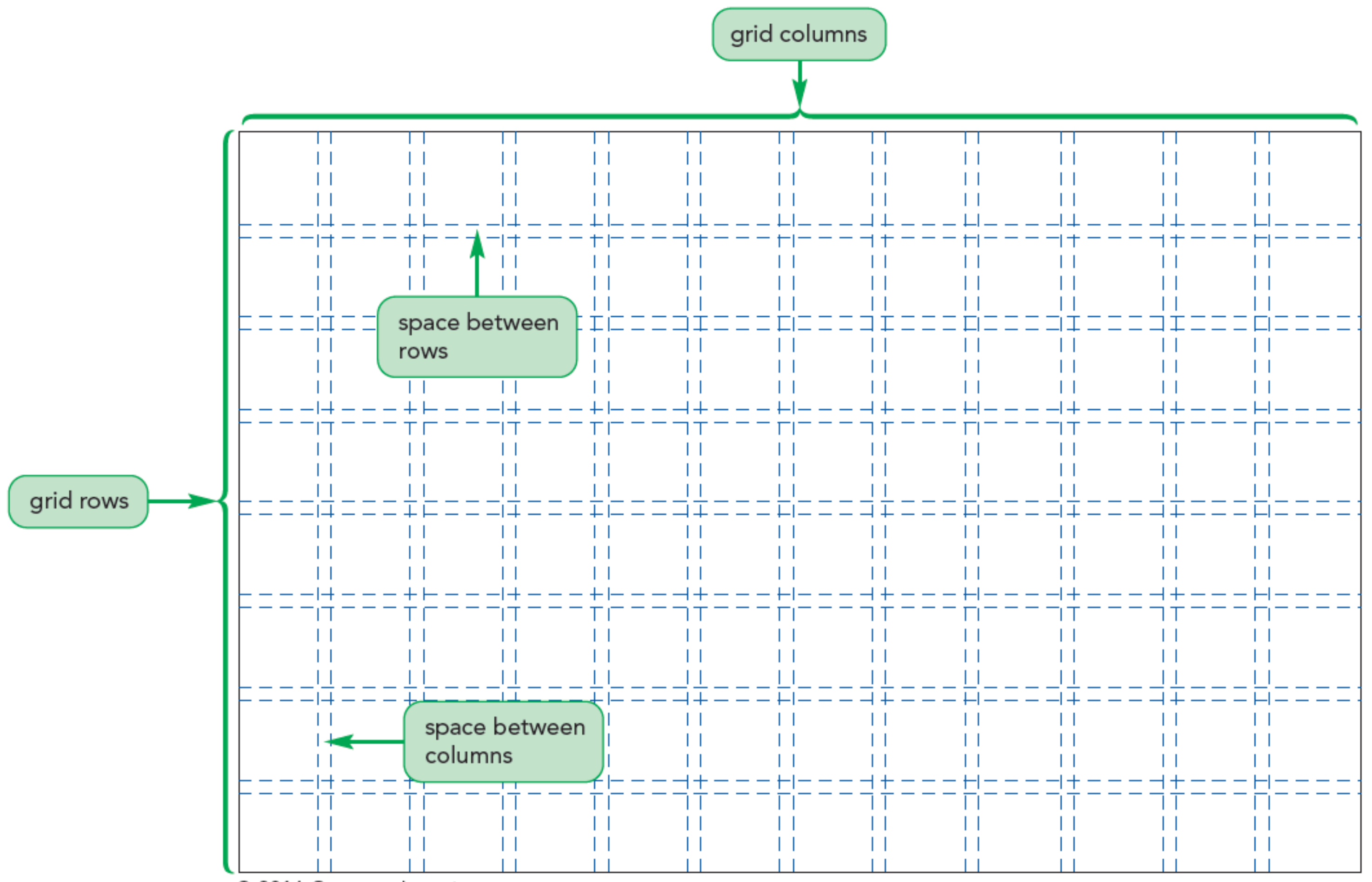
Pandaisia Chocolates © 2017 All Rights Reserved

## Overview of Grid-Based Layouts

- Rows and columns form a grid
  - The number of rows is based on the page content
  - The number of columns is based on the number that provides the most flexibility in laying out the page content

# Overview of Grid-Based Layouts (continued 1)

**Figure 3-29** Page grid



© 2016 Cengage Learning

# Overview of Grid-Based Layouts (continued 2)

- Advantages of using a grid:
  - Grids add order to the presentation of page content
  - A consistent logical design gives readers the confidence to find the information they seek
  - It is easily accessible for users with disabilities and special needs
  - It increases the development speed with a systematic framework for the page layout

## Fixed and Fluid Grids

- **Fixed grids** – Every column has a fixed position
  - Widths of the columns and margins are specified in pixels
- **Fluid grids** – Provides more support across different devices with different screen sizes.
  - Column width is expressed in percentages

## CSS Frameworks

- A **framework** is a software package that provides a library of tools to design a website
  - Includes style sheets for grid layouts and built-in scripts to provide support for a variety of browsers and devices
- Some popular CSS frameworks include
  - **Bootstrap**
  - **YAML4**
  - **960 Grid System**
  - **Foundation 3**

## Setting up a Grid

- A grid layout is based on rows of floating elements
- Each floating element constitutes a column
- The set of elements floating side-by-side establishes a row
- Many grid layouts use the `div` (or division) element to mark distinct rows and columns of the grid

## Setting up a Grid (continued 1)

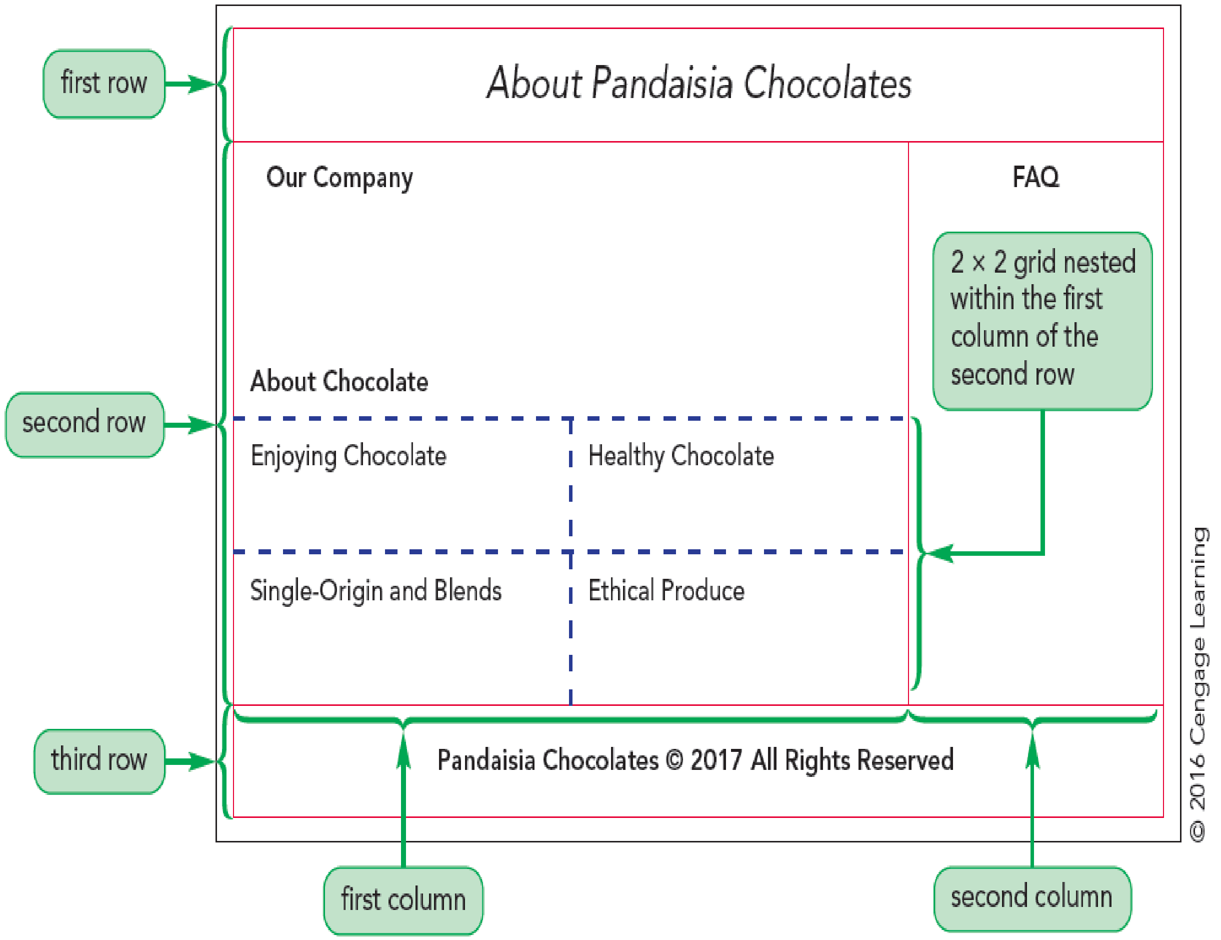
- This is an example of a simple grid consisting of a single row with two columns:

```
<div class=“row”>
     <div class=“column1”></div>
     <div class=“column2”></div>
</div>
```

  The page content is placed within the `div` elements

# Setting up a Grid (continued 2)

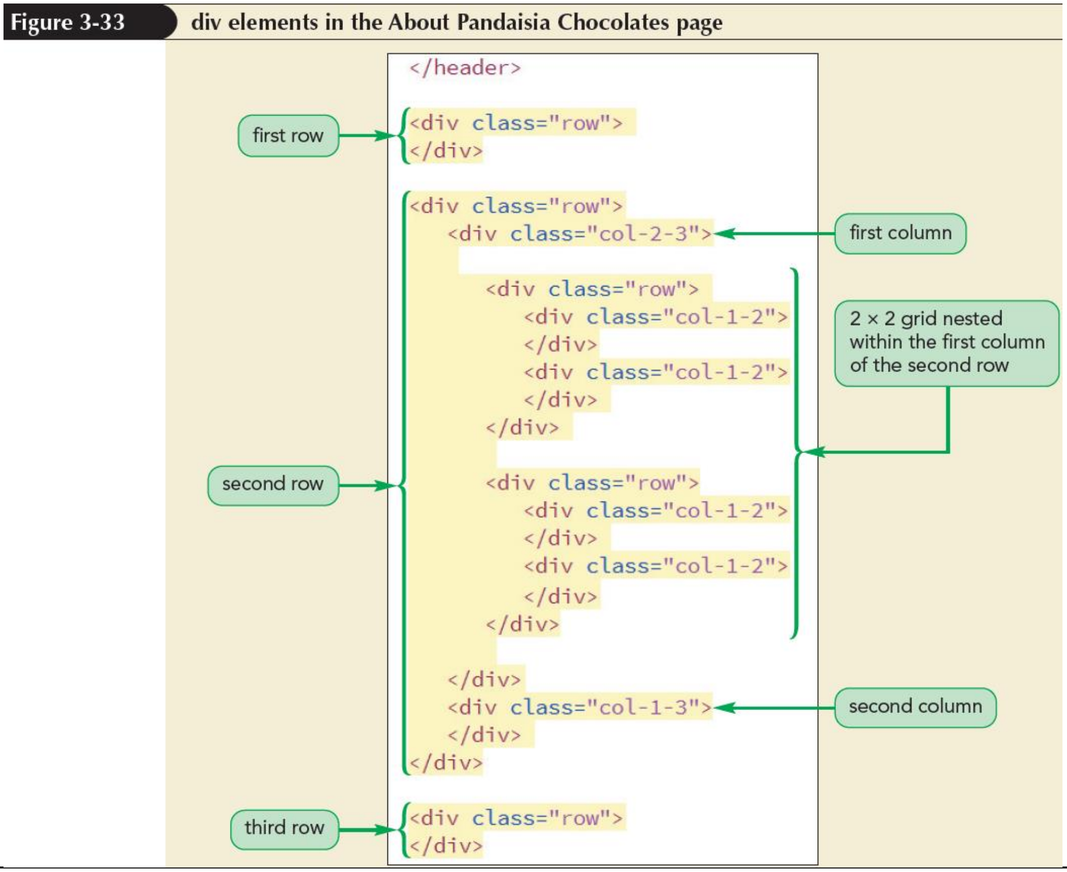
Figure 3-32 Proposed grid layout for the About Pandaisia Chocolates page



# Setting up a Grid (continued 3)

- The code for the grid layout for the Pandaisia Chocolates website is as follows:

Figure 3-33 div elements in the About Pandaisia Chocolates page

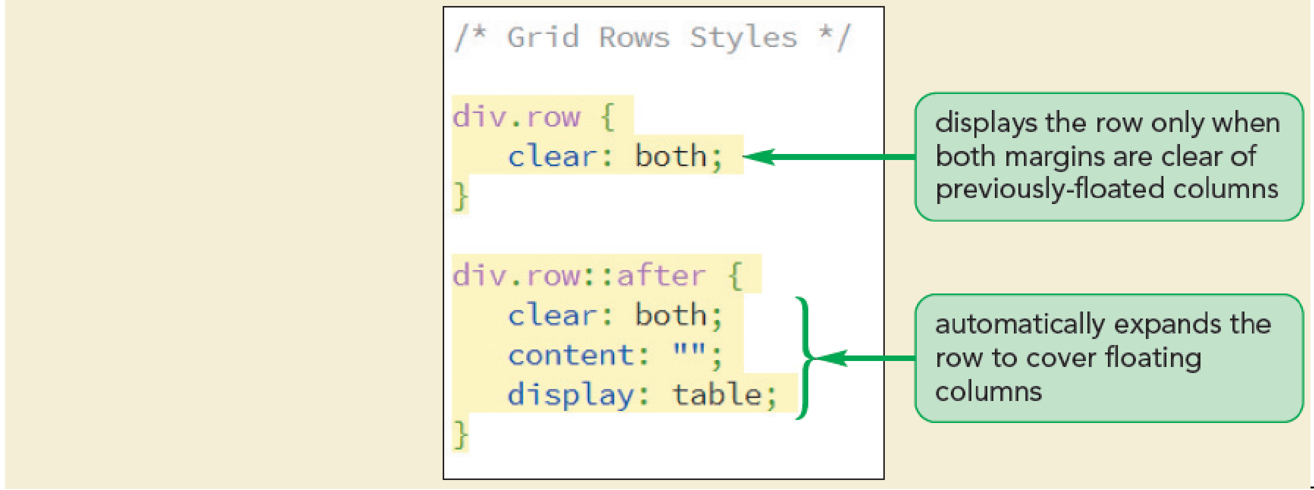


```
</header>

<div class="row">
</div>

<div class="row">
   <div class="col-2-3">

      <div class="row">
         <div class="col-1-2">
         </div>
         <div class="col-1-2">
         </div>
      </div>

      <div class="row">
         <div class="col-1-2">
         </div>
         <div class="col-1-2">
         </div>
      </div>

   </div>
   <div class="col-1-3">
   </div>
</div>

<div class="row">
</div>
```

# Designing the Grid Rows

- Grid rows contain floating columns
- Since a grid row starts a new line within a page, it should only be displayed when both margins are clear of previously floated columns

**Figure 3-34** Styles for row div elements



```
/* Grid Rows Styles */

div.row {
   clear: both;
}

div.row::after {
   clear: both;
   content: "";
   display: table;
}
```

displays the row only when both margins are clear of previously-floated columns
automatically expands the row to cover floating columns

## Designing the Grid Columns

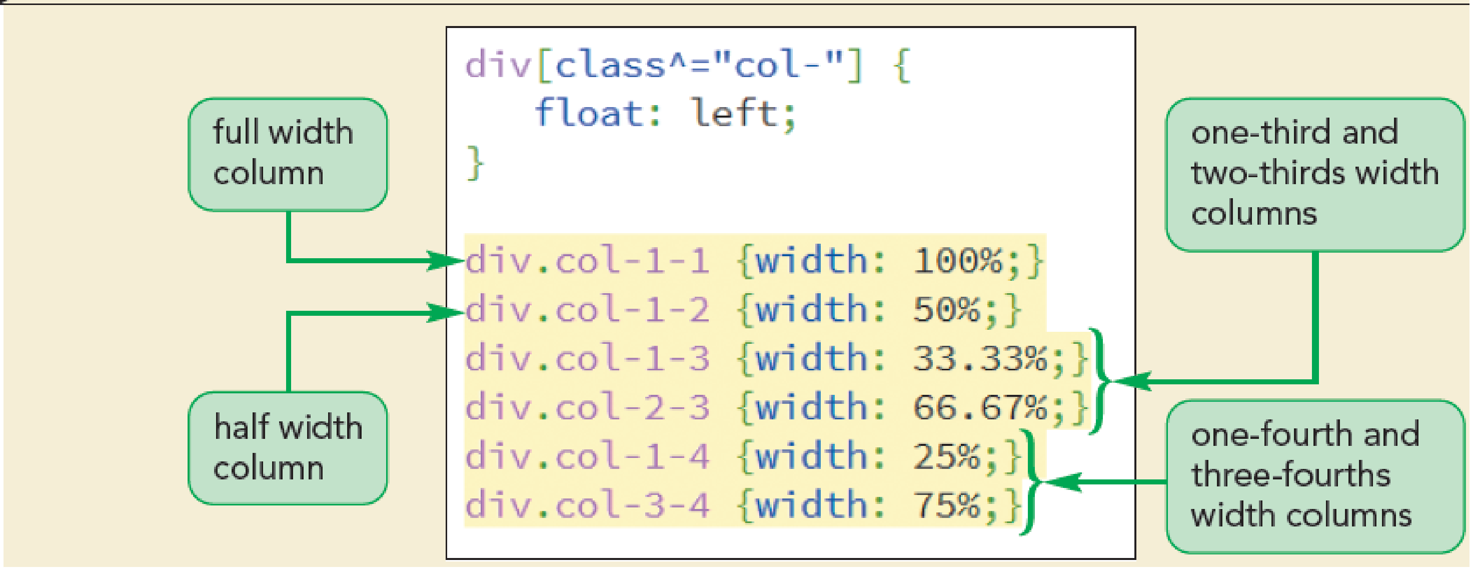
- Every grid column needs to be floated within its row
- Grid columns are placed within a `div` element having the general class name

  `class="col-`*`numerator-denominator`*`"`

  where *`numerator-denominator`* provides the fractional width of the column

# Designing the Grid Columns (continued)

Figure 3-36 Setting the column widths

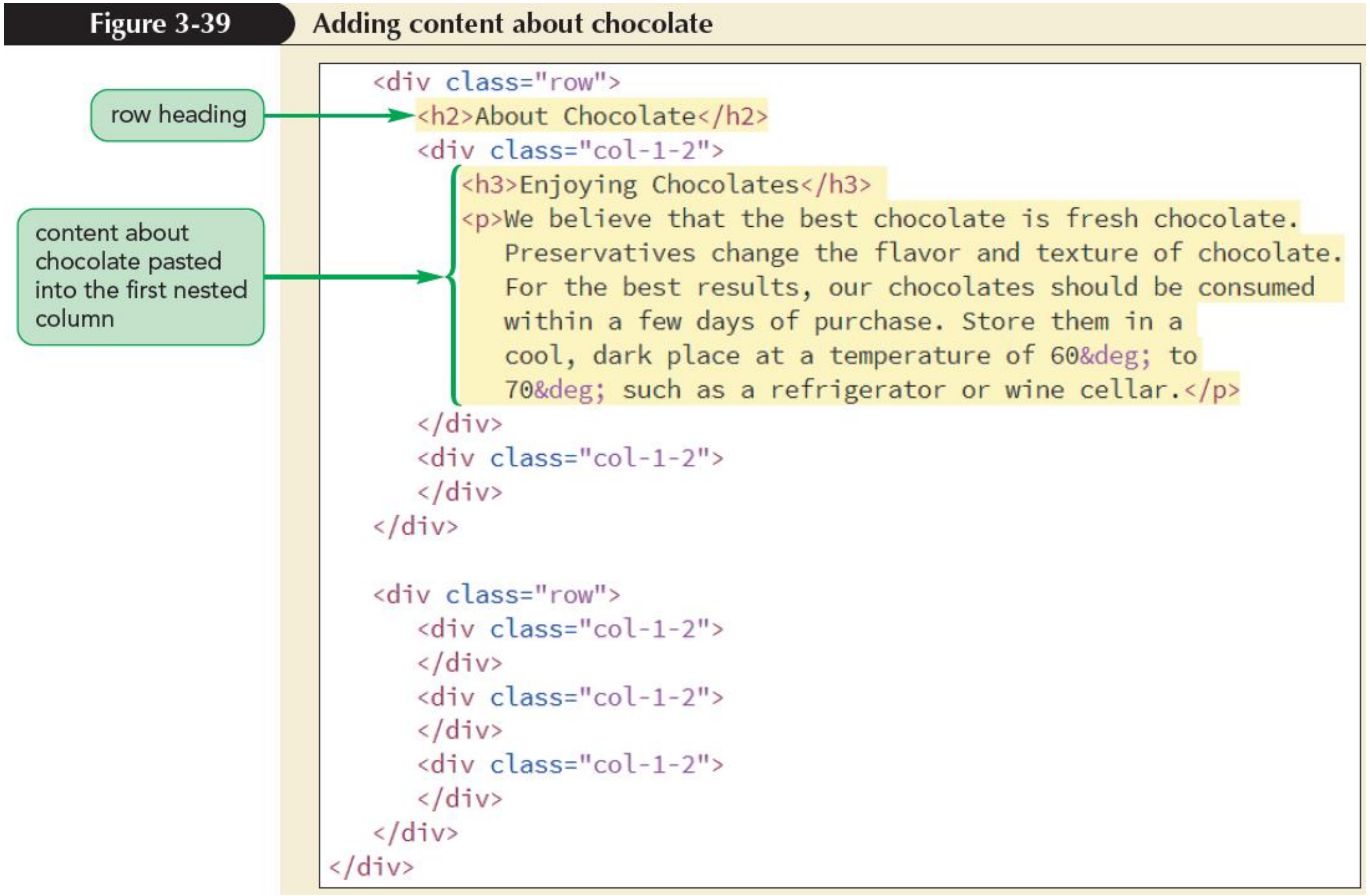


```
div[class^="col-"] {
   float: left;
}

div.col-1-1 {width: 100%;}
div.col-1-2 {width: 50%;}
div.col-1-3 {width: 33.33%;}
div.col-2-3 {width: 66.67%;}
div.col-1-4 {width: 25%;}
div.col-3-4 {width: 75%;}
```

full width column
half width column
one-third and two-thirds width columns
one-fourth and three-fourths width columns

# Adding the Page Content

**Figure 3-39** Adding content about chocolate



```
   <div class="row">
      <h2>About Chocolate</h2>
      <div class="col-1-2">
         <h3>Enjoying Chocolates</h3>
         <p>We believe that the best chocolate is fresh chocolate.
            Preservatives change the flavor and texture of chocolate.
            For the best results, our chocolates should be consumed
            within a few days of purchase. Store them in a
            cool, dark place at a temperature of 60&deg; to
            70&deg; such as a refrigerator or wine cellar.</p>
      </div>
      <div class="col-1-2">
      </div>
   </div>

   <div class="row">
      <div class="col-1-2">
      </div>
      <div class="col-1-2">
      </div>
      <div class="col-1-2">
      </div>
   </div>
</div>
```

## Outlining a Grid

- Outlines – Lines drawn around an element, enclosing the element content, padding, and border spaces
  - `Outline-width: `*`value;`* – Specifies the width of a line.
    - Properties of *`value`* are: `thin, medium,` or `thick`
  - `Outline-color: `*`color;`* – Specifies the color of a line.
    - Properties of *`color`* are: CSS color name or value

## Outlining a Grid (continued)

- `Outline-style:` *`style;`* – Specifies the design of a line
  - Properties of *`style`* are: `solid, double, dotted, dashed, groove, inset, ridge,` or `outset`

# Defining a CSS Grid

- To create a grid display without the use of `div` elements, use the following grid-based properties:

```
selector {
      display: grid;
      grid-template-rows: track-list;
      grid-template-columns: track-list;
}
```

  - *`grid`* – Selected elements in a grid
  - *`track-list`* – Space-separated list of row heights or column widths

# Defining a CSS Grid (continued)

- **fr unit** – Represents the fraction of available space left on the grid after all other rows or columns have attained their maximum allowable size
- For example, the following style creates four columns with the dimension specified in the style rule:

```
    grid-template-columns: 200px 250px
    1fr 2fr;
```

## Assigning Content to Grid Cells

- Elements in a CSS grid are placed within a **grid cell** at the intersection of a specified row and column
- By default, all of the specified elements are placed in the grid cell located at the intersection of the first row and first column

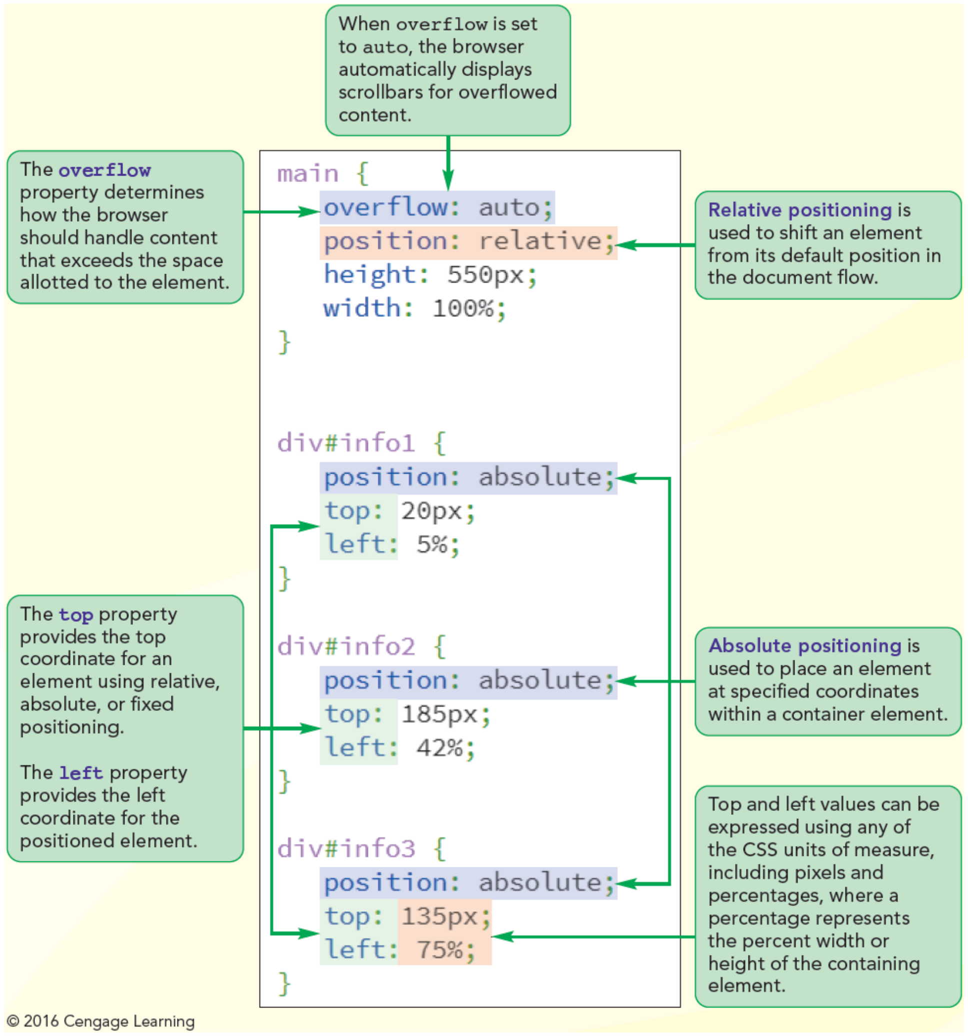
## Assigning Content to Grid Cells (continued)

- To place an element in a different cell, use

```
grid-row-start: integer;
grid-row-end: integer;
grid-column-start: integer;
grid-column-end: integer;
```

  where `integer` defines the starting and ending row or column that contains the content

# Layout with Positioning Styles



When overflow is set to auto, the browser automatically displays scrollbars for overflowed content.

The overflow property determines how the browser should handle content that exceeds the space allotted to the element.

```
main {
   overflow: auto;
   position: relative;
   height: 550px;
   width: 100%;
}


div#info1 {
   position: absolute;
   top: 20px;
   left: 5%;
}

div#info2 {
   position: absolute;
   top: 185px;
   left: 42%;
}

div#info3 {
   position: absolute;
   top: 135px;
   left: 75%;
}
```

**Relative positioning** is used to shift an element from its default position in the document flow.

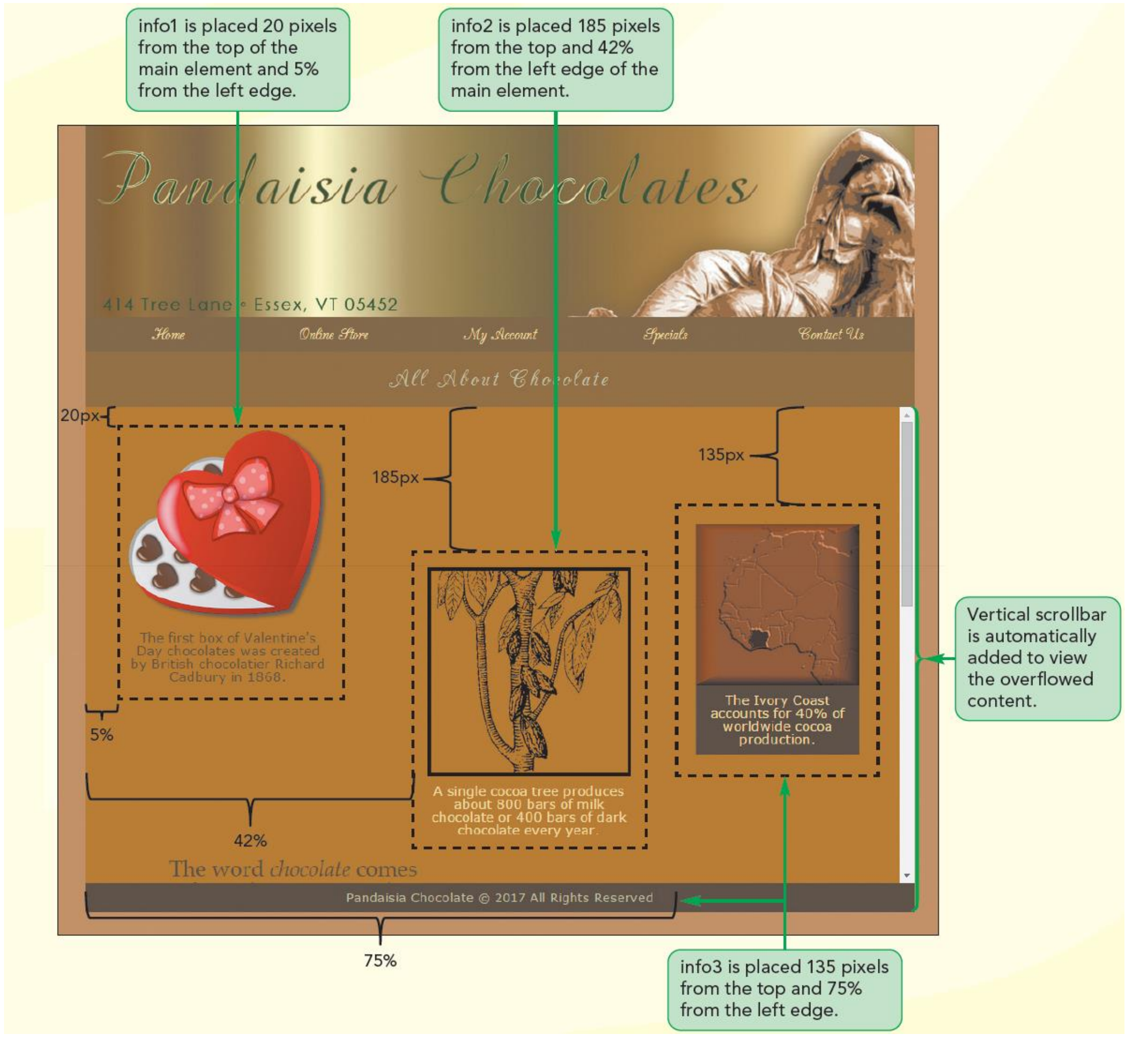
The top property provides the top coordinate for an element using relative, absolute, or fixed positioning.

The left property provides the left coordinate for the positioned element.

**Absolute positioning** is used to place an element at specified coordinates within a container element.

Top and left values can be expressed using any of the CSS units of measure, including pixels and percentages, where a percentage represents the percent width or height of the containing element.

# Layout with Positioning Styles (continued)


info1 is placed 20 pixels from the top of the main element and 5% from the left edge.
info2 is placed 185 pixels from the top and 42% from the left edge of the main element.
Pandaisia Chocolates
414 Tree Lane • Essex, VT 05452
Home
Online Store
My Account
Specials
Contact Us
All About Chocolate
20px
185px
135px
The first box of Valentine's Day chocolates was created by British chocolatier Richard Cadbury in 1868.
A single cocoa tree produces about 800 bars of milk chocolate or 400 bars of dark chocolate every year.
The Ivory Coast accounts for 40% of worldwide cocoa production.
5%
42%
The word chocolate comes
Pandaisia Chocolate © 2017 All Rights Reserved
75%
Vertical scrollbar is automatically added to view the overflowed content.
info3 is placed 135 pixels from the top and 75% from the left edge.

## The CSS positioning Styles

- To place an element at a specific position within its container, use

```
position: type;
top: value;
right: value;
bottom: value;
left: value;
```

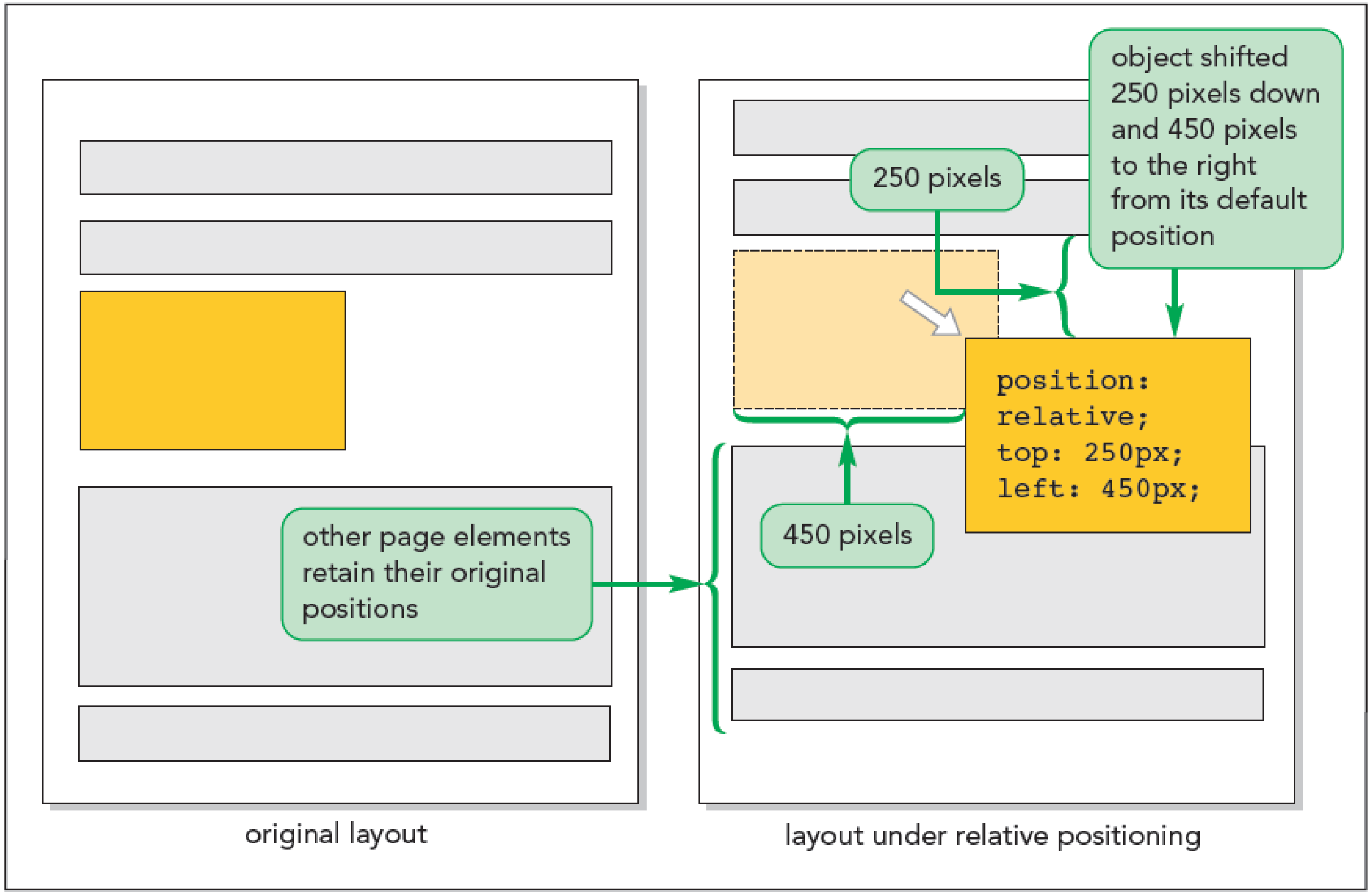
where `type` indicates the kind of positioning applied to the element and `top, right, bottom, and left` properties indicate the coordinates of the element

## The CSS Positioning Styles (continued 1)

- **Static positioning** – The element is placed where it would have fallen naturally within the flow of the document
- **Relative positioning** – The element is moved out of its normal position in the document flow
- **Absolute positioning** – The element is placed at specific coordinates within containers

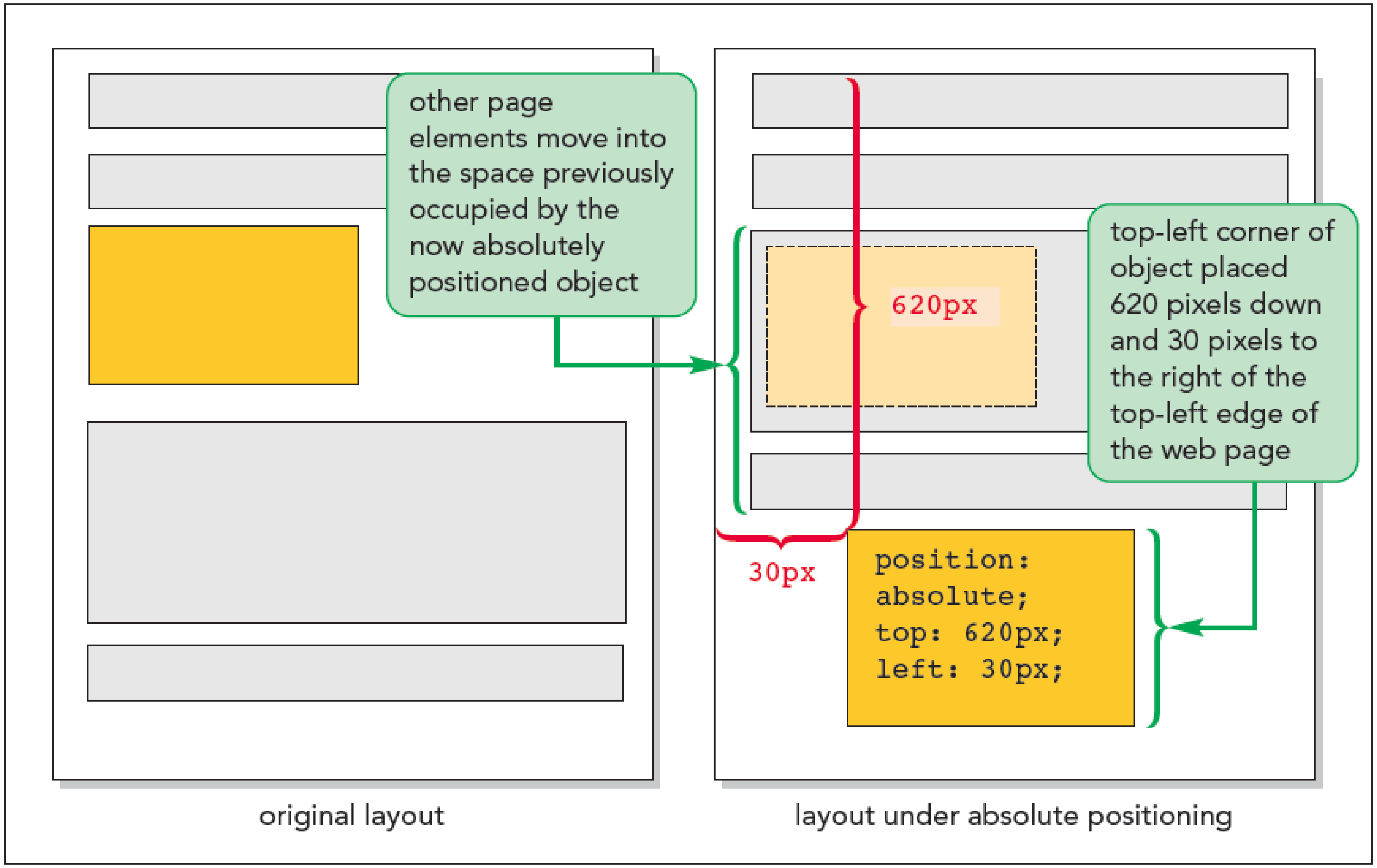
# The CSS Positioning Styles (continued 2)

**Figure 3-46** Moving an object using relative positioning


object shifted 250 pixels down and 450 pixels to the right from its default position
250 pixels
position: relative; top: 250px; left: 450px;
450 pixels
other page elements retain their original positions
original layout
layout under relative positioning

# The CSS Positioning Styles (continued 3)

**Figure 3-47** Moving an object using absolute positioning


other page elements move into the space previously occupied by the now absolutely positioned object
620px
top-left corner of object placed 620 pixels down and 30 pixels to the right of the top-left edge of the web page
30px
original layout
layout under absolute positioning

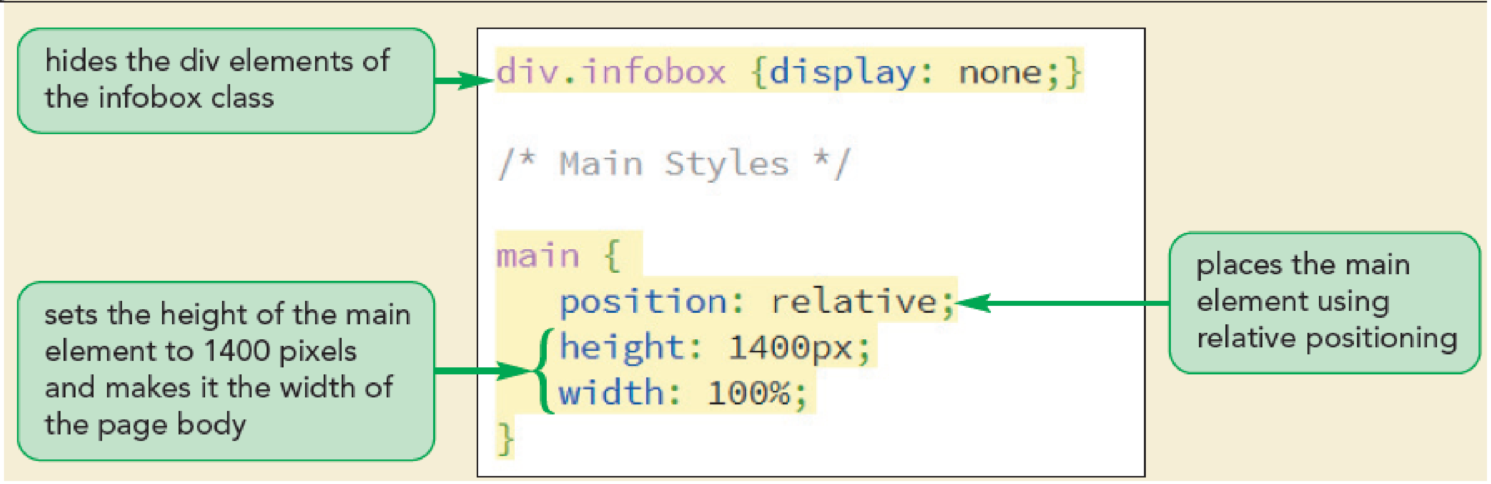
```
position:
absolute;
top: 620px;
left: 30px;
```

## Fixed and Inherited Positioning

- **Fixed positioning** – Fixes an object within a browser window to avoids its movement
- **Inherited positioning** – Allows an element to inherit the position value of its parent element

## Using the Positioning Styles

Figure 3-50 Setting the display styles of the main element


hides the div elements of the infobox class
sets the height of the main element to 1400 pixels and makes it the width of the page body
places the main element using relative positioning

```
div.infobox {display: none;}

/* Main Styles */

main {
   position: relative;
   height: 1400px;
   width: 100%;
}
```

## Using the Positioning Styles (continued 1)

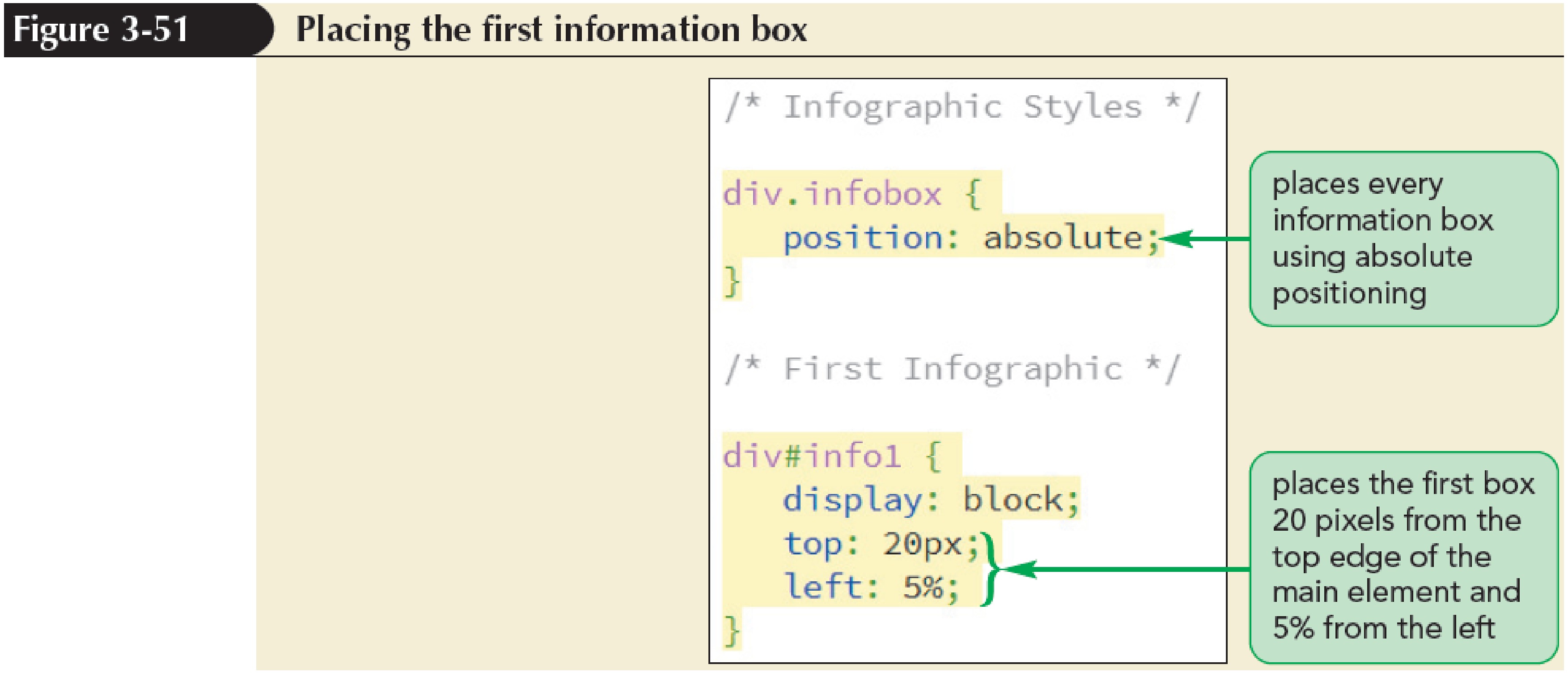


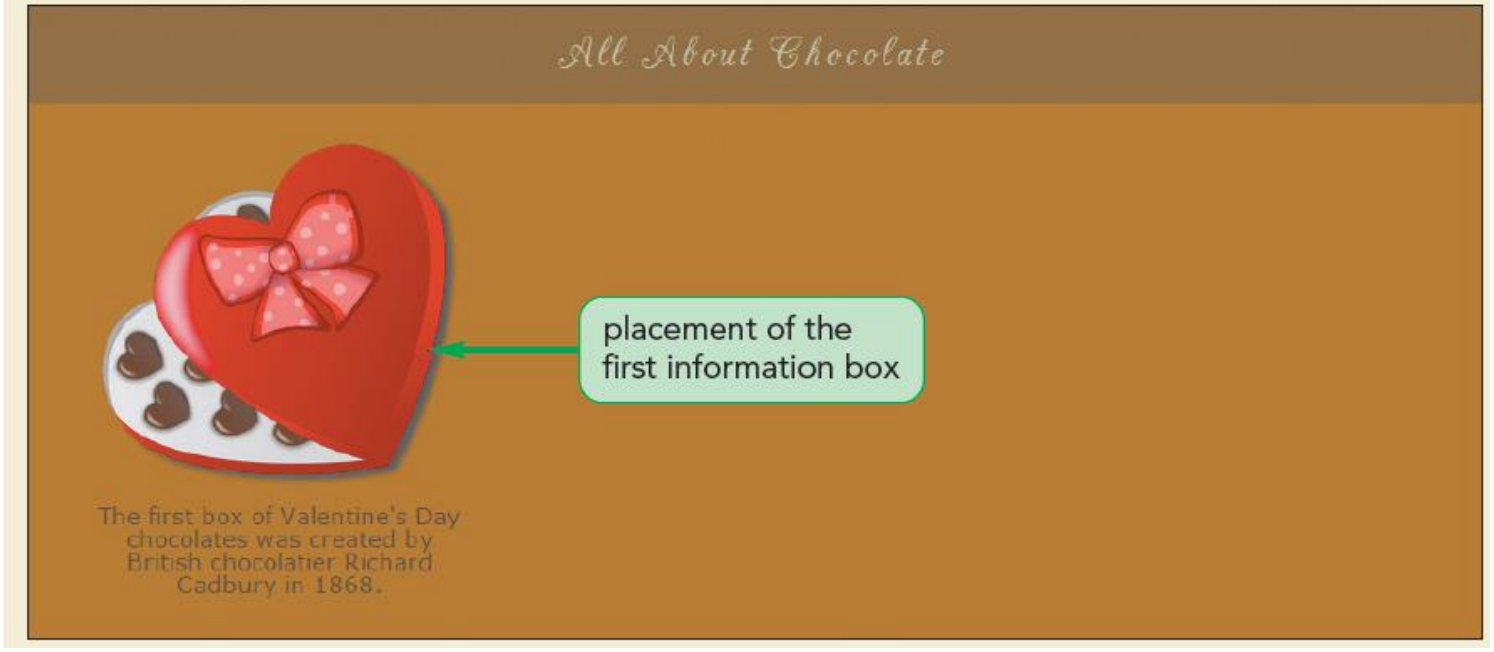
Figure 3-51 Placing the first information box

```
/* Infographic Styles */

div.infobox {
   position: absolute;
}

/* First Infographic */

div#info1 {
   display: block;
   top: 20px;
   left: 5%;
}
```

places every information box using absolute positioning
places the first box 20 pixels from the top edge of the main element and 5% from the left

# Using the Positioning Styles (continued 2)

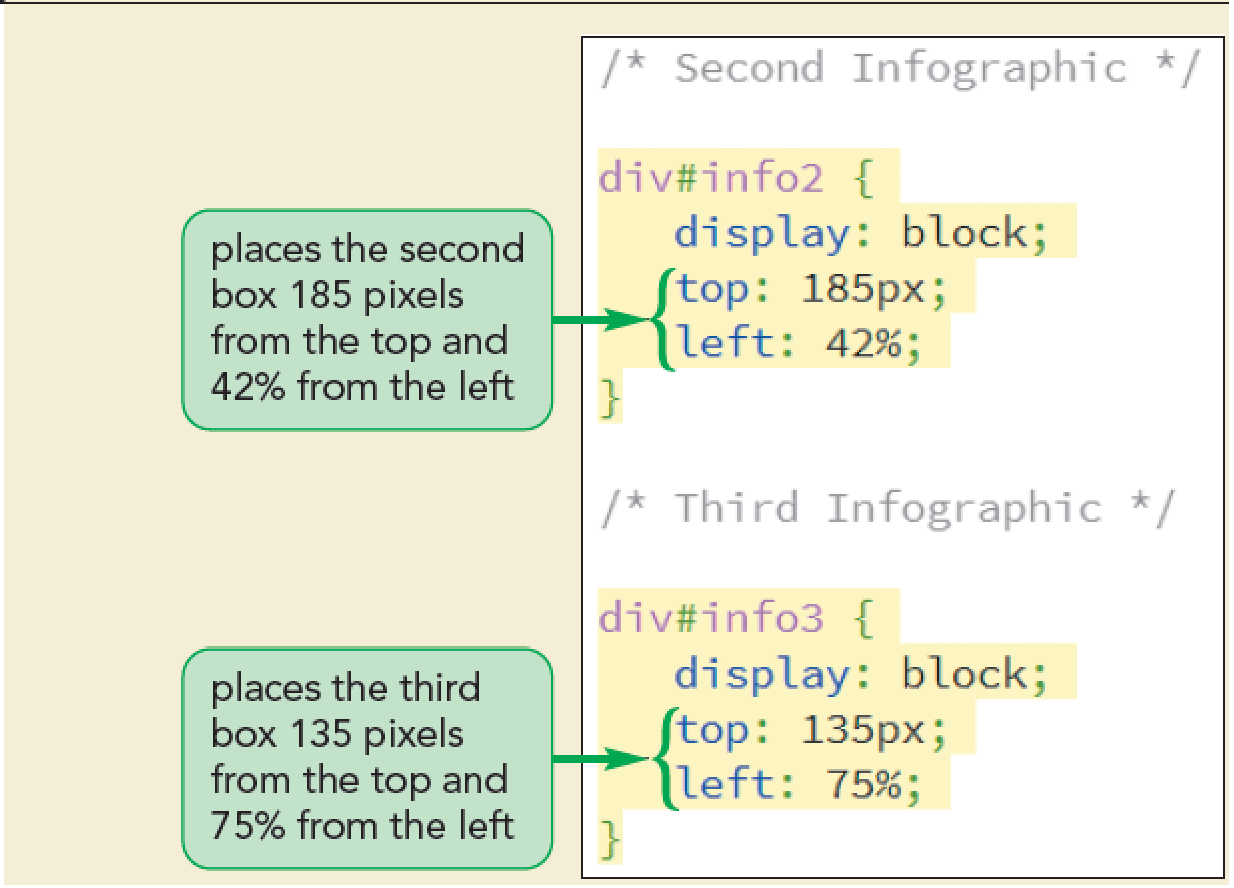
Figure 3-52 Appearance of the first information box



The first box of Valentine's Day chocolates was created by British chocolatier Richard Cadbury in 1868.

## Using the Positioning Styles (continued 3)

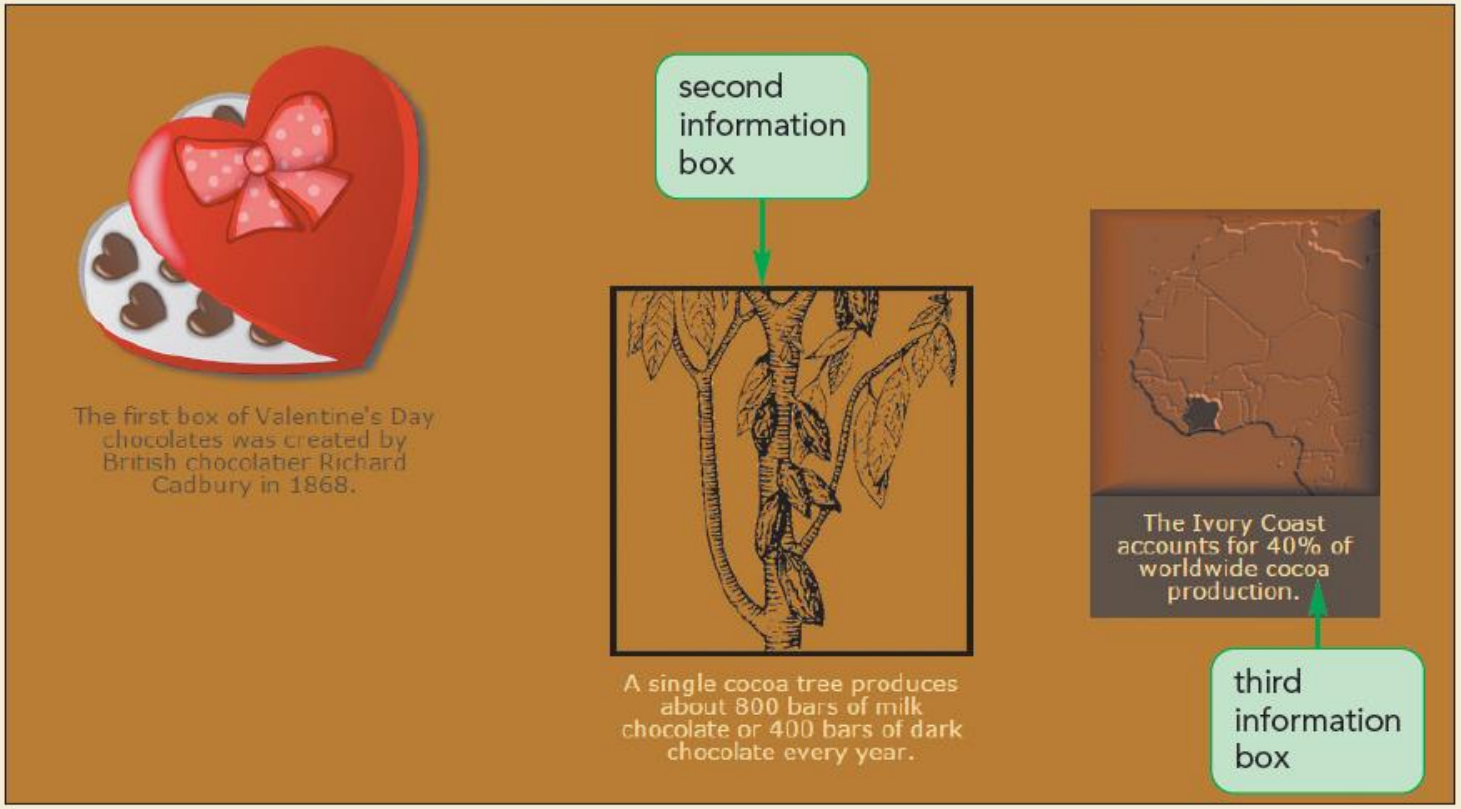
**Figure 3-53** Positions of the second and third boxes


places the second box 185 pixels from the top and 42% from the left
places the third box 135 pixels from the top and 75% from the left

```
/* Second Infographic */

div#info2 {
   display: block;
   top: 185px;
   left: 42%;
}

/* Third Infographic */

div#info3 {
   display: block;
   top: 135px;
   left: 75%;
}
```

# Using the Positioning Styles (continued 4)

Figure 3-54 Placement of the first three boxes



The first box of Valentine's Day chocolates was created by British chocolatier Richard Cadbury in 1868.

A single cocoa tree produces about 800 bars of milk chocolate or 400 bars of dark chocolate every year.

The Ivory Coast accounts for 40% of worldwide cocoa production.

## Handling Overflow

- `Overflow` – Controls a browser that handles excess content

  ```
      overflow: type;
  ```

  where *`type`* is `visible` (the default), `hidden`, `scroll`, or `auto`
- `visible` – Instructs browsers to increase the height of an element to fit overflow contents

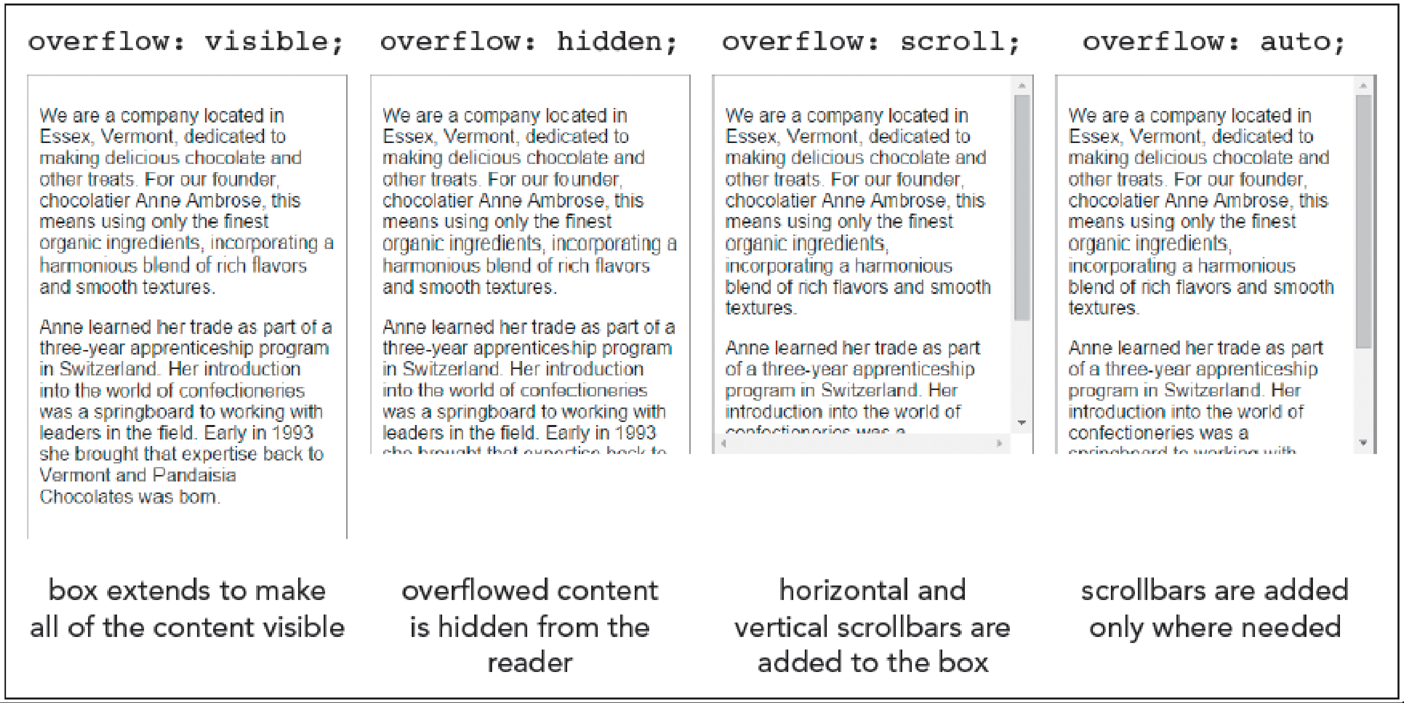
## Handling Overflow (continued 1)

- `hidden` – Keeps an element at the specified height and width, but cuts off excess content
- `scroll` – Keeps an element at the specified dimensions, but adds horizontal and vertical scroll bars
- `auto` – Keeps an element at the specified size, adding scroll bars when they are needed

# Handling Overflow (continued 2)

- CSS3 provides the `overflow-x` and `overflow-y` properties to handle overflow specially in the horizontal and vertical directions

**Figure 3-59** Values of the overflow property



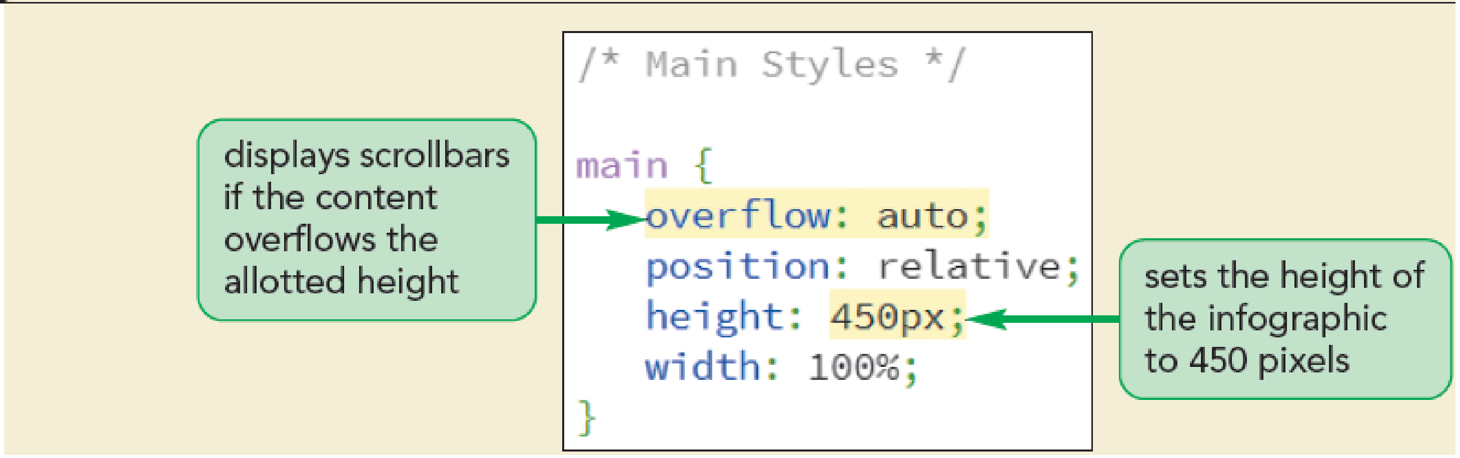
overflow: visible;
overflow: hidden;
overflow: scroll;
overflow: auto;
box extends to make all of the content visible
overflowed content is hidden from the reader
horizontal and vertical scrollbars are added to the box
scrollbars are added only where needed

We are a company located in Essex, Vermont, dedicated to making delicious chocolate and other treats. For our founder, chocolatier Anne Ambrose, this means using only the finest organic ingredients, incorporating a harmonious blend of rich flavors and smooth textures.

Anne learned her trade as part of a three-year apprenticeship program in Switzerland. Her introduction into the world of confectioneries was a springboard to working with leaders in the field. Early in 1993 she brought that expertise back to Vermont and Pandaisia Chocolates was born.

# Handling Overflow (continued 3)

**Figure 3-60** Setting the overflow property



```
/* Main Styles */

main {
   overflow: auto;
   position: relative;
   height: 450px;
   width: 100%;
}
```

displays scrollbars if the content overflows the allotted height
sets the height of the infographic to 450 pixels

## Clipping an Element

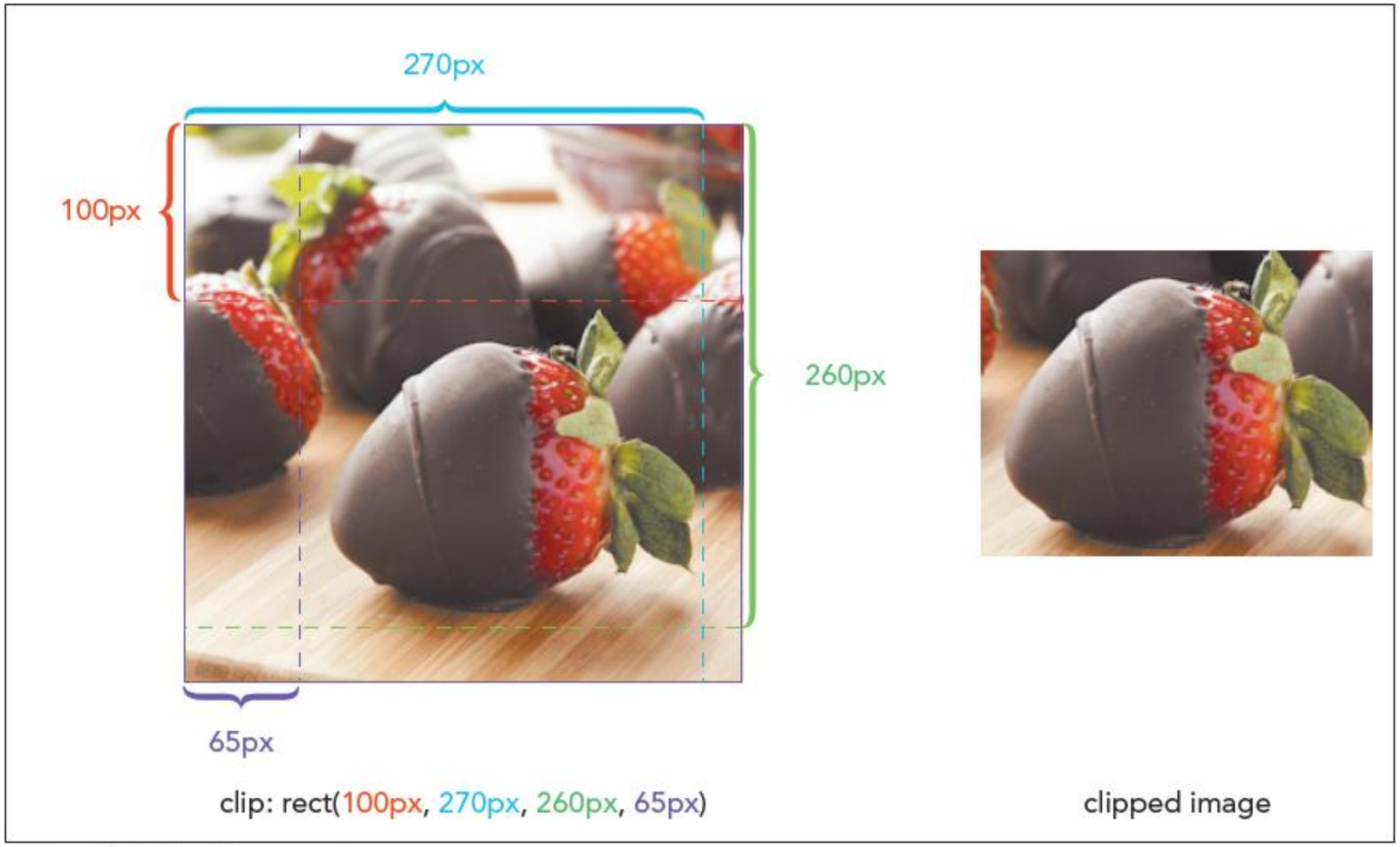
- **Clip** – Defines a rectangular region through which an element's content can be viewed
- Anything that lies outside the boundary of the rectangle is hidden
- The syntax of the `clip` property is

```
    clip: rect(top, right, bottom,left);
```

where *`top`*, *`right`*, *`bottom`*, and *`left`* define the coordinates of the clipping rectangle

# Clipping an Element (continued)

**Figure 3-62** Clipping an image



clip: rect(100px, 270px, 260px, 65px)

© Brent Hofacker/Shutterstock.com

## Stacking elements

- By default, elements that are loaded later by a browser are displayed on top of elements that are loaded earlier
- To specify different stacking order, use the following `z-index` property:

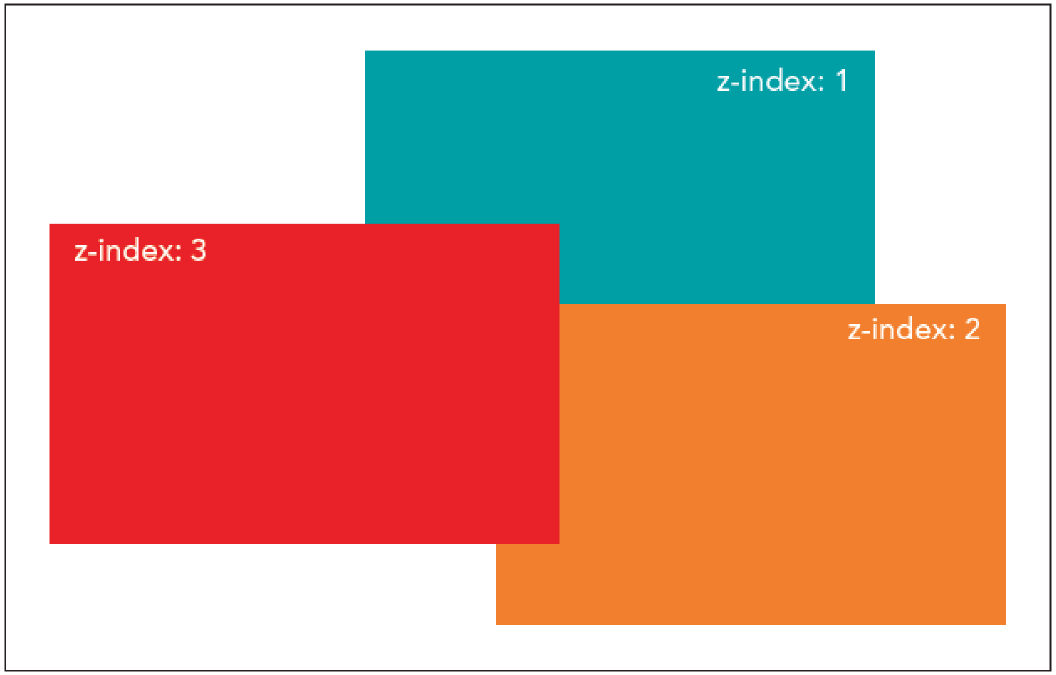
```
z-index: value;
```

where *`value`* is a positive or negative integer, or the keyword `auto`

## Stacking elements (continued 1)

- The `z-index` property works only for elements that are placed with absolute positioning

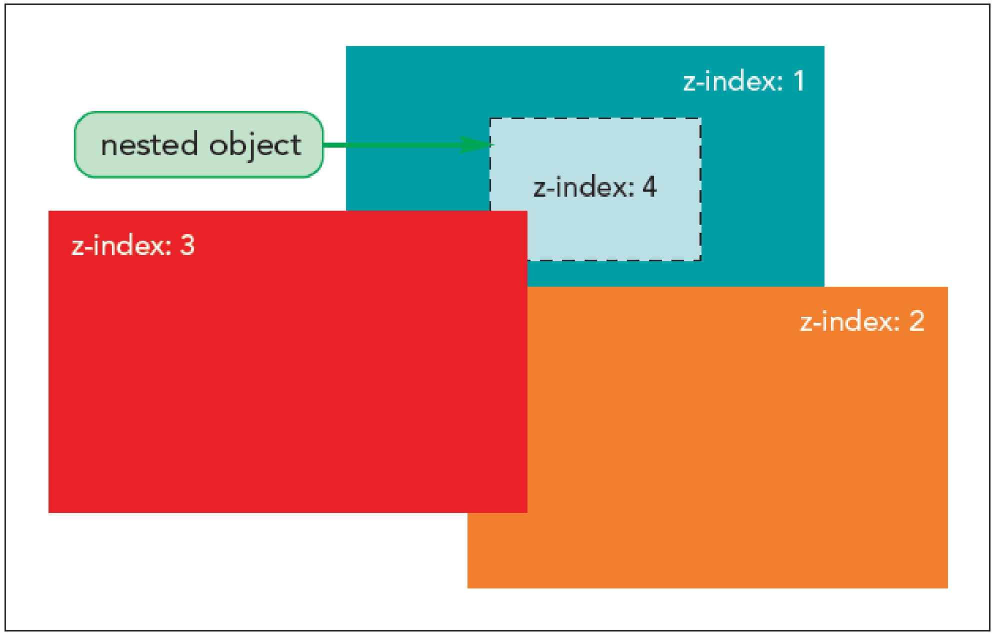
**Figure 3-63** Using the z-index property to stack elements



## Stacking elements (continued 2)

- An element's z-index value determines its position relative only to other elements that share a common parent

Figure 3-64 Stacking nested objects


z-index: 1
nested object
z-index: 4
z-index: 3
z-index: 2