



Covers:
HTML5/CSS3,
Static and Liquid Layouts,
Media Queries,
Animations,
Transformations, and
Transitions

Includes:

10 Units,
200+ code listings,
30+ Exercises for practice,
15 Hands-on Exercises, and
Self-Evaluation Test *

* You will receive a certificate of completion.

Less theory more practical approach saves many hours of struggle and pain

CSS3 Crash Course for Web Artists

Design and Style Websites with CSS/CSS3

www.htmlceous.com

✓ @htmlceous

www.htmlceous.com

www.htmlceous.com

www.htmlceous.com

CSS3 Crash Course for Web Artists

Design and Style Websites with CSS/CSS3

Team Htmleceous (Babeeta K)



CSS3 Crash Course for Web Artists

Copyright © 2016 **Htmleceous**

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews. No patent liability is assumed with respect to the use of information contained herein. Although every precaution has been taken in the preparation of this book, neither the author, nor Htmleceous, and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Htmleceous cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Book Code: **H00R2**

<http://www.htmleceous.com>

To all coders!

About Htmleceous

Htmleceous was established with the aim to provide quality learning and training solutions to students, professionals, and educators. The Htmleceous team has years of proficiency and expertise in the field of web design and development. The team specializes in website design/development, mobile website/ mobile application development, graphic design, logo & corporate branding, search engine optimization, and search engine marketing.

Preface

Why This Book

CSS (Cascading Style Sheet), the styling language used for websites, since its inception in 1996 has come a long way. Its latest incarnation is CSS3 that is completely compatible with the earlier specifications of CSS. This book is aimed at those who just started working on web design, CSS programming in particular. After completing this book, you would be able to style any website.

What You Will Learn

This book will teach you to create complex layouts, effects, animations, transitions, transformations, and much more with CSS code. The CSS properties are explained with examples and screen captures. You will gain skills by completing the exercises provided in the book.

What You Need

You need a computer, internet connection, and a text editor to complete the examples and exercises given in the book.

How This Book Is Structured

This book contains 10 units, 200+ code listings, and 15 hands-on exercises. Exercises [30+] are given at the end of the each chapter for practice. The following table summarizes the content of the book.

Content	Description
Unit CS1: Getting Started with CSS	This unit starts with the introduction of CSS and CSS syntax. You will learn to create and apply CSS styles.
Unit CS2: Identifying what to Select – Selectors	This unit deals with selectors. In the Unit, core CSS selectors are explained.
Unit CS3: Working with the Box Model	This unit explains the box model and the properties that are used to style the box.
Unit CS4: Working with the Borders and Backgrounds	This unit explain the techniques to create borders around boxes to properly visualize the padding and margin spaces.
Unit CS5: Working with Colors	This unit deals with colors. You will learn to set foreground color, background color, and

with Color	opacity for the elements.
Unit Formatting Text CS6:	The formatting of text is very important to create a webpage that appeals. In this unit, you will learn to use of the properties that you can use to format text.
Unit CS7: Styling Lists, Table, Forms, and Cursors	In this unit, you will learn how to change appearance of the unit elements, format tables, and specify bullet point styles for the lists.
Unit CS8: Creating Layout	In this unit teaches you how to control the position of the element on a page and how to create different types of layouts using CSS including the multicolumn layout.
Unit Animation, Transformations, and Transitions CS9:	This unit deals with animation, transformations, and transitions. You will learn how to apply animation to the elements of a page using CSS.
Unit CS10: Media Queries	This units deals with presenting content to all devices such as desktops, mobile phones, and tablets using media-queries.
Unit HOECS1: CSS Hands-on Exercises	This unit contains 11 hands-on exercises on CSS3 styling.
Unit HOEHT1: HTML Hands-on Exercises	This unit contains 4 hands-on exercises on creating HTML layouts using CSS3.
Unit CSA: Self-Evaluation Test	The purpose of the evaluation test is to evaluate the skills you have gained during this course. Attempt the test and send answers to the author. On successful completion of the test, you will be awarded a certification of completion.

What are the main features of the book?

- Beautifully designed and formatted units with screen shots and illustrations to help you quickly learn CSS programming.
- Less theory more practical. Saves many hours of struggle and pain.
- The important terms are italicized and color coded so that you never miss them.
- Covers CSS3 using HTML5 template.
- Additional information in form of tip, and notes.
- Color PDF of images used in the book.
- 10 units written in an easy to understand language.

- 200+ code listings, and 15 hands-on exercises.
- 30+ exercises for practice.
- Full source code included.
- Self-Evaluation Test and certificate of completion.

Downloading the Resources

This book is sold via multiple sales channels. If you don't have access to the source code used in this book, you can place a request for the resources by visiting the following link: <http://bit.ly/htmlcontact>. Please mention "*Resources-H00R2*" in the subject line.

Customer Support

At Htmleceous we believe support is personal. Our technical team is always ready to take care of your technical query. If you have any problem with the technical aspect of the book, navigate to <http://bit.ly/htmlcontact> and let us know about your query. Please mention "*Technical Query-H00R2*" in the subject line. We will do our best to resolve your queries.

Reader Feedback

Your feedback is always welcome. Your feedback is critical to our efforts at Htmleceous and it will help us in developing quality titles in the future. To send the feedback, visit <http://bit.ly/htmlcontact>. Please mention "*Feedback-H00R2*" in the subject line.

Errata

We take every precaution while preparing the content of the book but mistakes do happen. If you find a mistake in this book general or technical, we would be happy that you report it to us so that we can mention it in the errata section of the book's online page. If you find any errata, please report them by visiting the following link: <http://bit.ly/htmlcontact>. This will help the other readers from frustration. Once your errata is verified, it will appear in the errata section of the book's online page. Please mention "*Errata-H00R2*" in the subject line.

Stay Connected

Stay connected with us through Twitter ([@htmleceous](https://twitter.com/htmleceous)) and Facebook (<https://www.facebook.com/Htmleceous>) to know the latest updates about our products, information about books, and other related information.

Unit CS1: Getting Started With CSS

CSS stands for *Cascading Style Sheets*. HTML was never intended to contain tags for formatting a web page. CSS styles feature was added in HTML 4.0 to address this issue. To understand how CSS works, imagine that there is an *invisible box* around every HTML element. CSS allows you to change the appearance of the box using the CSS rules. In this chapter, I will show you how to create and apply CSS styles.

Key Points & Concepts

- What you need
- Getting familiar with CSS Syntax
- Understanding the Box model
- Creating internal, external, and inline style sheets
- Understanding how style sheet cascade
- Understanding the inheritance model

What you need?

You can use any *text editor* (even *Windows Notepad*) to write the *HTML* and *CSS* code written in this book. There are many free and paid editors available on the internet that you can download and use. *NotePad++* is one such editor that you can use to write the code. It is *open source* and supports many languages including *HTML* and *CSS*. You can download it from the following link: <http://notepad-plus-plus.org>.

I personally like *Brackets* from *Adobe*. It is a community guided *open source* text editor and allows you to easily design a web page. It includes lots of visual tools that help in the web design process. You can download *Brackets*

from the following link: <http://brackets.io>.

Defining a Style

CSS declaration is a rule that tells a web browser how to *format something* on a web page. For example, create a *blue border* around an image, add a drop-shadow to the box, create gradients, and so forth. A style is made up of two elements: the web page element [*referred to as selector*] that the browser formats and the formatting instructions [*referred to as declaration block*].

Now, for example, if you want to set the *font color* and the *font size* for all paragraphs on a web document, you would write the following CSS rule:

```
p { color: blue; font-size: 14px;}
```

The above rule tells browser that change the *color* of the text in all paragraphs to *blue* and set the *size* of the text to *14 pixels*. Table cs1.1 shows the description of each element of the CSS rule.

Table cs1.1: Description of CSS rules

Element	Description
Selector	The selector [p in this case] tells the browser which element(s) on a page to style.
Declaration Block	The code between the curly brackets [{ and }] is a declaration block.
Declaration	Between the opening and closing curly brackets, you can add one or more declarations. Every declaration has two parts, a property and a value separated by a colon. The declaration ends with a semi-colon.
Property	The properties in CSS are used to create wide range of formatting options. A property can be a single word or it can be few hyphenated words, for example, <i>overflow</i> , <i>font-size</i> , <i>list-style-type</i> , and so forth.
Value	Values are assigned to the properties.

In the above example, *p* is a selector whereas *color:blue*; and *font-size:14px*; are declarations. In the first declaration, *color* is the property and *red* is the value. In the second declaration, *font-size* is the *property* and *14px* is the *value*.

The area inside { and } is *declaration block*.

To make CSS more readable, you can put declaration on *each line*. In the following code, the content in all `<p>` elements will be *center-aligned*, with the text color set to *blue*:

```
p {  
color: blue;  
text-align: center;  
}
```

CSS Comments

Comments are used to explain the code. They help you in understanding the code when you edit it at a later stage. If you are part of a team, comments help your teammates to understand the code you have written. You must make a habit of putting comments in your code.

A CSS comment starts with a /* and ends with */.

```
p {  
color: blue;  
/* Sets the color of the text to blue */  
text-align: center;}
```

The comments can also span into multiple lines.

```
/* This is a multiline  
comment */
```

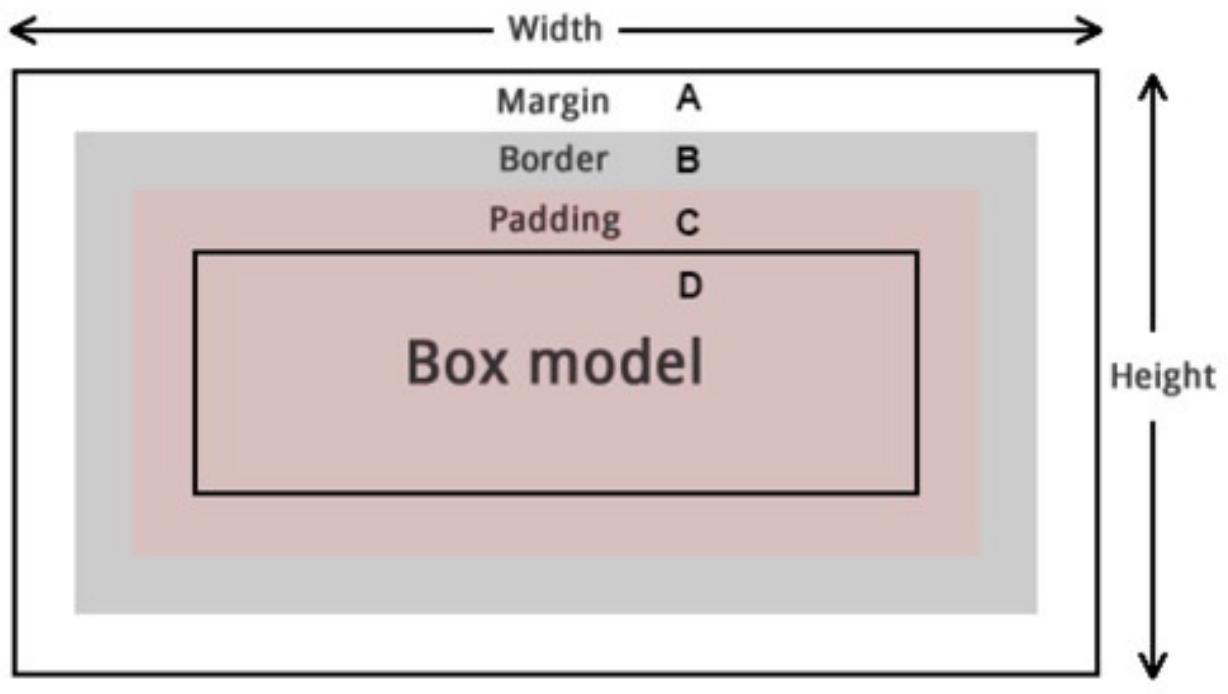


Comments

The comments are ignored by the browser.

Understanding the Box Model

The *CSS box model* is a *box* that wraps around the *HTML* elements. Each element in a web page is represented by a rectangular box. Each of these boxes are described via the *box* model. The box consists of *margins* [marked as *A* in Figure 1], *borders* [marked as *B*], *padding* [marked as *C*], and actual content [marked as *D*].



1

Table cs1.2 shows the different element of a box.

Table cs1.2: Elements of a box	
Element	Description
Content	The text and images appear here.
Padding	Clears a transparent area around the content.
Border	The border goes around the padding and content.
Margin	Like padding, margin clears a transparent area outside the border.

The *content* area is the area where the *real content* of the element lives. The *padding* area extends the *content* area and *add space* between the *content* and

the *eventual borders* surrounding the content. The *border* area extends the *padding* area. It controls the area *inside the border edge*. The *margin* area extends the *border* area and adds space to separate an *element* and its neighboring elements.

The *box* model is discussed in detail in a later chapter.

Using External CSS

An *external style sheet* is a *text file* containing all the *CSS rules*. The text file always ends with an extension *.css*. Once you create an external style sheet, you must *connect* it with the web page that you intend to format. You can attach a style sheet to a web page by using the *<link>* html element. The syntax of the *<link>* tags varies a bit depending on whether you are using *HTML5*, *HTML 4.01*, or *XHTML*. Table cs1.3 shows various link declarations.

Table cs1.3: Link declarations	
Type	Description
HTML 5	<code><link rel="stylesheet" href="css/global.css"></code>
HTML 4.01	<code><link rel="stylesheet" type="text/css" href="css/global.css"></code> HTML 4.01 is similar to the HTML5 but requires an additional type attribute.
XHTML	<code><link rel="stylesheet" type="text/css" href="css/styles.css" /></code> XHTML is like HTML 4.01 but it requires a closing slash at the end of the <i><link></i> tag.



href attribute

The *href* attribute points to the *location* where *external CSS file* is *stored*. It works similar to the *src* attribute of the ** tag.

You can also link a style sheet using CSS. CSS allows you to link to the external style sheets using the *@import* directive.

```
@import url(css/global.css);
```

You can also call the *global.css* from the web page by adding this directive to an HTML style tag:

```
<style>
@import url(css/global.css);
</style>
```



HTML 4.01

If you are using *HTML 4.01* or *XHTML* doctype, you need to add *type="text/css"* to the opening *<style>* tag.



Multiple External Style Sheet

You can also include *multiple* external style sheets by using more than one *@import* directive.

You can use the regular CSS after the *@import* directive. You must place all the import directives *before CSS rules*. For example:

```
<style>
@import url(css/global.css);
@import url(css/formelements.css);
p { color: blue;}
</style>
```

You can also specify *character encoding* for a style sheet. The only declaration that can come before an *@import* statement is a *@charset* directive. This directive specifies the encoding used by the stylesheet.

```
@charset "UTF-8";
@import "global.css"
p { color: blue;}
```

Using Internal CSS

An *internal* [or *embedded*] style sheet can be used when a single document that has a unique style. You can define the styles in the *head* section of the page using the `<style>` element. Listing cs1.1 shows the use of the `<style>` element.

Listing cs1.1: /cs1/embeddedcss.html

```
<!DOCTYPE html>
<html>
<head>
<title> Embedded CSS </title>

<style type="text/css">
p {
color: blue;
}

h1 {
color:red;
}
</style>
</head>
<body>
<h1> Embedded styles in Action </h1>
<p>
You can use the style element to define an embedded style sheet !
</p>
</body>
</html>
```

Inline Styles

The *inline* styles allow you to add style to *a specific element*. However, when you use the *inline* styles, you lose lots of advantages of a style sheet. Following is an example of the inline style:

```
<h1 style="color:red; margin-top: 2%;> This is a heading ! </h1>
```

How CSS rules Cascade

If there are two or more rules that apply to a single element, it is important to know that which rule will take *precedence*. Generally, all styles will *cascade* into a *new virtual style sheet*. Following is the *order of precedence* from lower to higher priority [the *inline* style has the highest priority]:

1. Browser default (the default styles applied by the browsers)
2. User styles (styles that have been defined by the user)
3. External style sheet (styles imported using the link element)
4. Internal style sheet in the head element (styles created using the style element)
5. Inline style (styles that are defined using the style global attribute)

If there are two or more rules in a style sheet, you should remember the following:

- Last Rule: If the two selectors are *identical*, the *last selector* will take *precedence*.
- Specificity: If one selector is more *specific* than the other, the *more specific* rule takes *precedence*.
- Important: You can also override the normal cascade order. You can add *!important* after any property value to indicate high priority.

Listing cs1.2 shows the specificity in action. Figure 2 shows the result of the

page in the browser.

Listing cs1.2:/cs1/cascade.html

```
<!DOCTYPE html>
<html>
<head>
<title>Cascade in Action</title>

<style>
*
{
font-family: verdana, sans-serif;
font-size: 120%;
}
p{
font-family: arial, sans-serif;
}
strong{
font-family: cursive;
color: green;
}

p > strong{
color:red !important;
}
p > strong{
color: blue;
}
.cacd{
background-color: #e7e7e7;
```

```
width: 480px;  
text-align: center;  
border: 1px dashed #000;  
}  
  
i{  
text-decoration: underline;  
}  
  
i{  
text-decoration: overline;  
</style>
```

```
</head>  
<body>  
<p class="cacd">  
This is the <strong> cascade </strong> in <i>action </i> !  
</p>  
</body>  
</html>
```

2

This is the **cascade** in action !

In Listing cs1.2:

- *p* is more specific than *.
- *p > strong* is more specific than *strong*.
- The second *i* selector takes precedence over the first *i*.
- In *p > strong* code block, *color:red !important*; has more precedence than the *color: blue; declaration*.

If you enter in a situation where there are two styles that can be applied to a single element, browser determines the specificity of a style by using the

following:

1. Number of *id values* in the *style selector*.
2. Number of *other attributes* and *pseudo-classes* in the *selector*
3. Number of *elements names* and *pseudo-classes* in the selector

To measure the *specificity*, in a selector, *count* the number of *inline styles* [say *a*], count the number of *IDs* [say *b*], count the number of *classes/attributes/pseudo-classes* [say *c*], and count the number of *elements* and *pseudo-elements* [say *d*]. Now, concatenate the four numbers [*a-b-c-d*] to get the specificity. Listing cs1.3 shows an example.

Listing cs1.3: /cs1/specification.html

```
<!DOCTYPE html>
<html>
<head>
<title> Specificity </title>

<style>
h1 {
color:red;
}
p {
color:blue;
}
p.theClass{
color:chocolate;
}
</style>
</head>
```

```
<body>
<h1> Specificity in Action </h1>
<p class="theClass">
Browser uses specificity when it counters a tie-break situation!
</p>
</body>
</html>
```

In the Listing cs1.3, the selector `p.theClass` has **0** inline styles, **0** ids, **1** class attribute, and **1** element, therefore, it has specificity of **0 0 1 1**. The other style `p` has **0** inline styles, **0** ids, **0** class attributes, and **1** elements, therefore, it has specificity of **0 0 0 1**. As a result, `p.theClass` has take precedence over the `p` declaration.



Calculating specificity

You can also use the following website to calculate the specificity:
<http://specificity.keegan.st.>

One other approach that is popular among the CSS programmers is to give a value of **100** to the **ID selector**, a value of **10** to **every class selector**, and a value of **1** to **every HTML selector**. Next, add them all to get the specificity value.

For example:

- a: has a specificity of **1** because it has **1 HTML selector**.
- #theID: has a specificity of **100**.
- .theClass: has a specificity of **10**.
- div p.theClass: has a **2** HTML selectors, and a class selector therefore a specificity **[1+1+10]** of **12**.

Considering the above rule and Listing cs1.3, the `p` style has a specificity of **1** whereas the `p.theClass` style has a specificity of **$1+10=11$** .

Inheritance

Inheritance is a process by which some CSS properties are inherited from the parent tags. In other words, the properties applies to one tag is passed on to the nested tags. For example, if you specify *font-family* or *color* properties for the `<body>` element, they will be inherited by the most of the child elements such as `<h1>`, ``, ``, and so forth. Not all properties are inherited by the *child elements*. For example, *background-color* and *border* properties.

Consider the following listing:

Listing cs1.4: /cs1/inheritance.html

```
<!DOCTYPE html>
<html>
<head>
<title>Inheritance in Action</title>

<style>
body{
font-family: verdana, arial, sans-serif;
color: darkblue;
padding: 10px;
}
.inheritance{
width:500px;
border: 1px dotted #000;
}

</style>

</head>
```

```
<body>
<div class="inheritance">
<h1> Inheritance in Action </h1>
<p>
In object-oriented programming, inheritance is the concept that when a class of objects
is defined,
any subclass that is defined can inherit the definitions of one or more general classes.
</p>
</div>
</body>
</html>
```

If you render the above listing in browser, you will get the output as shown in Figure 3.

Now, change the following code in the *.inheritance* code block and re-render *[inheritance1.html]*: *padding: inherit;*.

```
.inheritance{
width:500px;
border: 1px dotted #000;
padding: inherit;
}
```

Inheritance in Action

In object-oriented programming, inheritance is the concept that when a class of objects is defined, any subclass that is defined can inherit the definitions of one or more general classes.

3

Notice in Figure 4, the *.inheritance* class has *inherited padding value* of *10px* from the *body* code block.

Inheritance in Action

In object-oriented programming, inheritance is the concept that when a class of objects is defined, any subclass that is defined can inherit the definitions of one or more general classes.

4

Exercises

Exercise CS1-E1

Change the color of all `<p>` elements in Listing cs1-e1 to *red* using an *internal style sheet*.

Listing cs1-e1: /cs1/cs1-e1.html

```
<!DOCTYPE html>
<html>
<head>
<title> About Apple</title>
</head>
<body>

<h1> About Apple </h1>
<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>

</body>
</html>
```

Exercise CS1-E2

Add an *external style sheet* to the Listing cs1-e1 with link to *global.css*.

Exercise CS1-E3

Using the Listing cs1-e1, set the color of the *second paragraph* to *blue* using inline style.

Exercise CS1-E4

Remove all styles from the Listing cs1-e4 except the *inline style*.

Listing cs1-e4: /cs1/cs1-e4.html

```
<!DOCTYPE html>
<html>
<head>
<title> About Apple</title>
<style>
body{
font-family: helvetica, verdana, sans-serif;
padding: 10px;
}
h1{
color: red;
}
#appleMania{
width: 300px;
border: 1px dotted #000;
color: blue;
padding: inherit;
}
</style>
<link rel="stylesheet" href="global.css">
```

```
</head>

<body>

<h1> About Apple </h1>
<div id="appleMania">
<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p style="text-decoration:underline">Apple Inc. is an american company that sells consumer electronics, computer software, and online services. </p>
</div>
</body>
</html>
```

Summary

In this chapter, I have described how to *define* styles, how to connect to the *external* CSS style files, how CSS rules *cascade*, how *inheritance* works. You have also seen two different approaches you can take to calculate the specificity. In next chapter, we will talk about the selectors.

Unit CS2: Identifying

What to Select - Selectors

In the previous chapter, I have explained how to *define* CSS rules. If you recall from the previous chapter, I talked about, how to style HTML elements such as *a*, *h1*, *body*, and so forth. In CSS, these are referred to as *type selectors*. In this chapter, I will explain *core CSS3 selectors*. By the time you finish this chapter, you will have a firm understanding of making selections using various selectors.

Key Points & Concepts

- Using basic selectors
- Using pseudo-element selectors
- Using pseudo-class selectors

Using Basic CSS Selectors

A *selector* is the *most basic* component in CSS. In this section, I will explain some simple selectors.

Universal Selector (Asterisk)

The *universal selector (*)* is most fundamental selector of the CSS selectors. The *shorthand* for this selector is *asterisk*. It allows you to target every element in the document. However, it is used rarely because its scope is wide. For example, say you want all elements [*tags*] in the page to appear in *blue* color, your group selector would look like the following:

```
a, p, h1, h2, h3, h4, p, span, div ... ... and so forth... {
```

```
color: blue;  
}
```

The *universal selector* is a much shorter way to tell browser that you want all tags in the page to appear in *blue* color:

```
*
```

```
{
```

```
color: blue;
```

```
}
```

The syntax for the *universal selector* is given next:

```
* {
```

```
css declarations;
```

```
}
```

Listing cs2.1 shows an example that uses the *universal* selector.

Listing cs2.1: /cs2/universal-selector.html

```
<!DOCTYPE html>  
<html>  
<head>  
<title>Universal Selector</title>  
  
<style>  
*{  
border: 1px solid darkblue;  
}
```

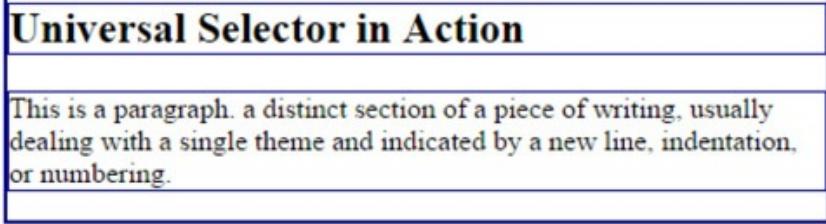
```

</style>

</head>
<body>
<div>
<h2> Universal Selector in Action </h2>
<p>
This is a paragraph. a distinct section of a piece of writing,
usually dealing with a single theme and indicated by a new line,
indentation, or numbering.
</p>
</div>
</body>
</html>

```

The style I have defined in Listing cs2.1 draws a *solid border* around the selected elements. You can see the effect of style in Figure 1.



If you omit *asterisk (*)* with simple selectors, it does not affect the result. For example, `*.warning` and `.warning` are same, `*[lang=it]` and `[lang=it]` are same, and `*#myid` and `#myid` are same. The *universal* selector can also select *everything inside an element*. In the following example, it selects all elements inside the `<div>` elements and set their *background color* to *blue*:

```

div * {
background-color: blue;
}

```

Since the *universal* selector doesn't specify a particular element, it is rarely used by the CSS programmers.

Type Selector

If you want to select a *specific element* in a web page, you can use the *element* type as a selector. It matches elements by node name. For example, if you want to target all of *p* elements, use *p* as the *selector*. Listing 2-2 shows the use of the *type* selector, *p*:

Listing cs2.2: /cs2/type-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> Type Selector </title>
<style>
p{
padding: 10px;
margin: 10px;
color: blue;
}
</style>
</head>
<body>
<h2> About Apple </h2>
<p>Apple is the round fruit of a tree of the rose family, which typically has
thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer
software,
and online services. </p>
</body>
</html>
```

Figure 2 shows the effect of the `p` selector used in Listing cs2.2.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc. is an American company that sells consumer electronics, computer software, and online services.

2

Class Selector

The `class` selector matches an element based on the contents of the `class attribute` of the element. When you want to style elements different from the related elements on a page, for example, say if you want to add `padding` to a paragraph and while leaving other untouched, you can use the `class` selector with that paragraph. The `class` attribute can contain `space separated list` of items, for example, `<p class="class1 class2 class3"> </p>`.

You can create a `class` selector by first giving it a `name` and then applying it to the elements that you want to style. The syntax for the `class` selector is as follows:

```
.class {  
    css declarations;  
}
```

Listing cs2.3 shows the demonstration of this selector.

[Listing cs2.3: /cs2/class-selector.html](#)

```
<!DOCTYPE html>  
<html>  
<head>  
<title> Class Selector </title>
```

```

<style>
.fruit{
color: blue;
}
.company{
color: red;
}
.company1{
font-family: Georgia, sans-serif;
}
</style>
</head>
<body>
<h2> About Apple </h2>
<p class="fruit">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p class="company company1">Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>
</body>
</html>

```

Figure 3 shows the effect of the *class* selector used in Listing cs2.3. In this listing, I have used three classes: *fruit*, *company*, and *company1*. The *fruit* class selector targets the *first paragraph* whereas the other two selectors target the *second paragraph*.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

3

Apple Inc. is an American company that sells consumer electronics, computer software, and online services.

You can also pin point a tag using a *class* selector. You can target a *single type of element* that has been assigned to a class. In Listing cs2.4, the *span* element

is targeted which has been assigned the *company1* class selector. Figure 4 shows the result of the style.

Listing cs2.4: /cs2/class-selector-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> Class Selector </title>
<style>
.fruit{
color: blue;
}
.company{
color: red;
}
span.company1{
font-family: Georgia, sans-serif;
font-weight: bolder;
}
</style></head>
<body>
<h2> About Apple </h2>
<p class="fruit">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p class="company"> <span class="company1"> Apple Inc </span> is an American company that sells consumer electronics, computer software, and online services. </p>
</body>
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services.

4

ID Selector

This selector allows you to select elements by using the *id* attribute. The value of the *id* attribute should be *unique* in a web page. It uses the following syntax:

```
#id {  
    css declarations;  
}
```

Listing cs2.5 shows the *id* selector in action.

Listing cs2.5: /cs2/id-selector.html

```
<!DOCTYPE html>  
<html>  
<head>  
<title> ID Selector </title>  
<style>  
#company{  
font-weight: 600;  
text-decoration: underline;  
color:red;  
}  
</style>  
</head>  
<body>
```

```
<h2> About Apple </h2>
<p id="fruit">Apple is the round fruit of a tree of the rose family, which typically has
thin green or red skin and crisp flesh.</p>
<p> <span id="company">Apple Inc</span> is an American company that sells
consumer electronics, computer software, and online services. </p>

</body>
</html>
```

In this example, I have targeted an element with *id* set to *company*. Figure 5 shows the result.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has
thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics,
computer software, and online services.

5

Attribute Selectors

The *attribute selectors* allows you to target the elements based on a given attribute or attribute value. The following is the syntax for the attribute selector:

```
[attribute] { css declarations; }
```

Listing cs2.6 shows the use of the *attribute* selector.

Listing cs2.6: /cs2/attribute-selector.html

```
<!DOCTYPE html>
<html>
<head>
```

```

<title> Attribute Selector </title>
<style>
[href] {
font-weight: 600;
color:blue;
text-decoration: none;
}
</style>
</head>
<body>
<h2> About Apple </h2>
<p><a href="www.apple.com"> Apple Inc. </a> is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p> Apple Inc is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs"> Steve Jobs </a> was one of the co-founder of the company. </p></body> </html>

```

In Listing cs2.6, I have used the simplest form of the *attribute* selector that selects all tags with *href* attribute irrespective of the value the attribute holds. The style code in Listing cs2.6 will select all tags with *href* attributes [the Apple and Steve Jobs links], as shown in Figure 6.

About Apple

[Apple Inc.](http://www.apple.com) is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](http://en.wikipedia.org/wiki/Steve_Jobs) was one of the co-founder of the company.

6

You can also use *conditions* to *match* attributes. The Table 2.1 shows various *conditions* you can use to *match attributes*.

Table 2.1: Conditions for the <i>attribute</i> selector		
Selector (Condition)	Example	Example Description
		Selects all elements with a target attribute irrespective of the value assigned to

[attr]	[target]	the target attribute.
[attr="val"]	[target=_blank]	Selects all elements with target set to _blank [see Listing cs2.7 and Figure 7]
[attr^="val"]	a[href^="https"]	Selects every <a> tag whose href attribute begins with the string https [see Listing cs2.8 and Figure 8]
[attr\$="val"]	p[class\$="-2"]	Selects every <p> tag whose class attribute ends with the string -2 [see Listing cs2.9 and Figure 9]
[attr*="val"]	p[class*="pple"]	Selects every <a> tag whose href attribute contains the string pple [see Listing cs2.10 and Figure 10]
[attr~="val"]	a[title~="Apple"]	Selects all tags with the title attribute containing space separated multiple values and one of them is Apple [see Listing cs2.11 and Figure 11]
[attr =“val”]	p[class =“apple”]	Selects all elements with a class attribute whose value is hyphen-separated list of values and the first of which is apple [see Listing cs2.12 and Figure 12]

Listing cs2.7: /cs2/attribute-selector-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> Attribute Selector </title>
<style>
[href] {
font-weight: 600;
color:blue;
text-decoration: none;
}
[target=_blank]{
color:red;
}
</style>
</head>
<body>
```

```
<h2> About Apple </h2>
```

```
<p> Apple Inc is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
```

```
<p> <a href="www.apple.com">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" target="_blank"> Steve Jobs </a> was one of the co-founder of the company. </p>
```

```
</body>
```

```
</html>
```

About Apple

Apple Inc is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. **Steve Jobs** was one of the co-founder of the company.

7

Listing cs2.8: /cs2/attribute-selector-2.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title> Attribute Selector </title>
```

```
<style>
```

```
[href]
```

```
{
```

```
text-decoration: none;
```

```
}
```

```
a[href^="https"]{
```

```
color: red;
```

```
}
```

```
</style>
</head>
<body>

<h2> About Apple </h2>

<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p> <a href="www.apple.com">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs"> Steve Jobs </a> was one of the co-founder of the company. Use <a href="https://www.google.com"> Google </a> to know more about Steve Jobs. </p>
</body>
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. Steve Jobs was one of the co-founder of the company. Use Google to know more about Steve Jobs.

8

Listing cs2.9: /cs2/attribute-selector-3.html

```
<!DOCTYPE html>
<html>
<head>
<title> Attribute Selector </title>
<style>
[href]
{
text-decoration: none;
```

```

    }
  p[class$="-2"]{
border: 1px dotted darkgreen;
padding: 2px;
}

</style>
</head>
<body>

<h2> About Apple </h2>
<p class="apple-1">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p class="apple-2" > <a href="www.apple.com">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs"> Steve Jobs </a> was one of the co-founder of the company. </p>
<p class="apple-2"> Use <a href="https://www.google.com"> Google </a> to know more about Steve Jobs. </p>

</body>
</html>

```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. **Steve Jobs** was one of the co-founder of the company.

Use [Google](https://www.google.com) to know more about Steve Jobs.

9

Listing cs2.10: /cs2/attribute-selector-4.html

<!DOCTYPE html>

```
<html>
<head>
<title> Attribute Selector </title>
<style>
[href]
{
text-decoration: none;
}
p[class*="pple"]{
border: 1px dotted green;
padding: 2px;
}

</style>
</head>
<body>

<h2> About Apple </h2>
<p class="aple">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p class="apple-2" > <a href="www.apple.com">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs"> Steve Jobs </a> was one of the co-founder of the company. </p>
<p class="apple-2"> Use <a href="https://www.google.com"> Google </a> to know more about Steve Jobs. </p>

</body>
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. Steve Jobs was one of the co-founder of the company.

Use Google to know more about Steve Jobs.

10

Listing cs2.11: /cs2/attribute-selector-5.html

```
<!DOCTYPE html>
<html>
<head>
<title> Attribute Selector </title>
<style>
[href]
{
text-decoration: none;
}
p[class~="apple2"]{
border: 1px dotted red;
padding: 2px;
}

</style>
</head>
<body>

<h2> About Apple </h2>
```

```
<p class="apple1 apple2"> Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
```

```
<p class="apple1" ><a href="www.apple.com">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs"> Steve Jobs </a> was one of the co-founder of the company. </p>
```

```
<p class="apple1 apple2"> Use <a href="https://www.google.com"> Google </a> to know more about Steve Jobs. </p>
```

```
</body>  
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. Steve Jobs was one of the co-founder of the company.

Use Google to know more about Steve Jobs.

11

Listing cs2.12: /cs2/attribute-selector-6.html

```
<!DOCTYPE html>  
<html>  
<head>  
<title> Attribute Selector </title>  
<style>  
[href]  
{  
text-decoration: none;  
}  
p[class |=“apple”]{
```

```
border: 1px dotted #000;  
padding: 2px;  
}  
  
</style>  
</head>  
<body>  
  
<h2> About Apple </h2>  
<p class="apple-a-fruit">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>  
  
<p class="apple-a-company"> <a href="www.apple.com" title="Apple the fruit">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" title="Apple the company"> Steve Jobs </a> was one of the co-founder of the company. </p>  
<p> Use <a href="https://www.google.com"> Google </a> to know more about Steve Jobs. </p>  
  
</body> </html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. Steve Jobs was one of the co-founder of the company.

12

Use [Google](#) to know more about Steve Jobs.

Combining Selectors

You can *combine* different selectors to create CSS rules. In this section, I will describe various combination of the selectors.

Styling group of tags

To style a group of tags, define a *comma separated list* of selectors. In Listing cs2.13, I have specified a type *selector(h1)*, and two attribute selectors (*p[class |=“apple”]* and *[href]*). You can see the effect of this union of selectors in Figure 13.

Listing cs2.13: /cs2/combing-selectors-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> Combining Selectors </title>
<style>

h1, p[class |=“apple”], [href] {
font-family: helvetica, verdana, sans-serif;
text-decoration: none;
}

</style>
</head>
<body>

<h2> About Apple </h2>
<p class=“apple-a-fruit”>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p class=“apple-a-company”> <a href=“www.apple.com” title=“Apple the fruit”>Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href=“http://en.wikipedia.org/wiki/Steve_Jobs” title=“Apple the company”> Steve Jobs </a> was one of the co-founder of the company. </p>
<p> Use <a href=“https://www.google.com”> Google </a> to know more about Steve Jobs. </p>
```

```
</body>
```

```
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Use [Google](#) to know more about Steve Jobs.

13

Descendant Selector

You can use the *decedent* selector to match all elements that are *decedent* (*child*, *grandchild*, and *beyond*) of the specified element. To define a selector, add a *whitespace* between two simple selectors. The *decedent* selector is similar to the *child* selector, however, the relation between the matched elements doesn't need to be strictly parent child. In Listing cs2.14, I have used a CSS rule that defines some CSS properties for all instances of *span* elements within the *p* element. Figure 14 shows the result of Listing cs2.14.

[Listing cs2.14: /cs2/combing-selectors-2.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> Combining Selectors </title>
<style>
body{
line-height: 1.5;
}
p span{
border: 1px dotted black;
padding: 3px;
```

```

margin: 2px;
}

</style>
</head>
<body>

<h2> About Apple </h2>

<p >Apple is the round <span class="apple-1"> fruit</span> of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p ><a href="www.apple.com" >Apple Inc</a> is an American <span class="apple-2"> company</span> that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" > Steve Jobs</a> was one of the co-founder of the company. </p>

<p> Use <a href="https://www.google.com"> Google</a> to know more about Steve Jobs. </p>

</body>
</html>

```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

14

Listing cs2.15 is another example of *descendant* selector. In this listing, I have defined two tables with id attributes. Then, I created two rules targeting all decedents of *th* elements within the tables. You can see the effect in Figure 15.

Listing cs2.15: /cs2/combing-selectors-3.html

```
<!DOCTYPE html>
```

```
<html>
<head>
<title> Combining Selectors </title>
<style>
body, h1{
font-family: verdana,san-sarif;
font-size: 90%;
}
h1{
width: 168px;
padding: 10px;
background-color: lightgreen;
}
#sales-table-1 th {
border-bottom: 3px solid red;
padding: 5px;
}
#sales-table-2 th{
border-bottom: 3px solid blue;
padding: 5px;
}
td{
color: saddlebrown;
border-bottom: 1px solid black;
padding: 5px;
}
</style>
</head>
<body>
<h1> [ January Sales Data ] </h1>
<table id="sales-table-1">
```

```
<tr><th>Salesman</th> <th>Appointments</th> <th>Sales</th> <th>Expanses</th>
</tr>

<tr><td>Phillip</td><td>110</td><td>57</td><td>2500</td></tr>
<tr><td>Brent</td><td>78</td><td>48</td><td>2000</td></tr>
<tr><td>Bill</td><td>100</td><td>75</td><td>3700</td></tr>
<tr><td>Fred</td><td>95</td><td>56</td><td>4000</td></tr>
<tr><td>Tim</td><td>99</td><td>45</td><td>6000</td></tr>
<tr><td>Graham</td><td>58</td><td>35</td><td>4500</td></tr>
</table>

<h1> [ February Sales Data ] </h1>

<table id="sales-table-2">

<tr><th>Salesman</th> <th>Appointments</th> <th>Sales</th> <th>Expanses</th>
</tr>

<tr><td>Phillip</td><td>57</td><td>60</td><td>3200</td></tr>
<tr><td>Brent</td><td>98</td><td>52</td><td>2700</td></tr>
<tr><td>Bill</td><td>77</td><td>80</td><td>4200</td></tr>
<tr><td>Fred</td><td>105</td><td>62</td><td>4800</td></tr>
<tr><td>Tim</td><td>88</td><td>55</td><td>6200</td></tr>
<tr><td>Graham</td><td>46</td><td>41</td><td>4800</td></tr>
</table>

</body>
</html>
```

[January Sales Data]

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham	58	35	4500

[February Sales Data]

Salesman	Appointments	Sales	Expenses
Phillip	57	60	3200
Brent	98	52	2700
Bill	77	80	4200
Fred	105	62	4800
Tim	88	55	6200
Graham	46	41	4800

15

Child Selector

The *child* selector allows you to select the *direct children* of a *parent ancestor*. To define a child selector, place a *>* sign (greater than sign) between two simple selectors. In Listing cs2.16, I am looking for the *li* element that is a direct child of *ul* element. You can see the effect in Figure 16.

[Listing cs2.16: /cs2/combing-selectors-4.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> Combining Selectors </title>
<style>
body, h1{
font-family: verdana,san-sarif;
font-size: 90%;
margin: 20px;
```

```

    }
ul > li{
color: blue;
}

</style>
</head>
<body>
<ul>
<li> Item 1</li>
<li> Item 2 </li>
<ol>
<li> Nested Item 1</li>
<li> Nested Item 2</li>
</ol>
<li> Item 3 </li>
</ul>

</body>
</html>

```

If you use the following descendant selector with the Listing cs2.16, all `` tags will be targeted:

```

ul li{
color: blue;
}


- Item 1
- Item 2
  - 1. Nested Item 1
  - 2. Nested Item 2
- Item 3

```

Selecting Adjacent Sibling Elements

This selector only selects the specified element that is immediately preceded by the former element. This selector is almost similar to the general sibling selector with the only difference that the targeted element must be an *immediate sibling*, not just a general sibling. To define adjacent sibling selector, put a *plus* sign between two simple selectors. For example, `li + li { border-left: 1px dotted #000; }` rule select a *list item* that is directly following another *li* element.

In Listing cs2.17, I am looking for all *p* elements that are immediately after the *div* element. Figure 17 shows the effect of the rule.

Listing cs2.17: /cs2/combing-selectors-5.html

```
<!DOCTYPE html>
<html>
<head>
<title> Combining Selectors </title>
<style>
body{
line-height: 1.5;
}
div + p{
color: blue;
}

</style>
</head>
<body>
<div>
<h1> About Apple </h1>
<p >Apple is the round <span class="apple-1"> fruit</span> of a tree of the rose family,
```

which typically has thin green or red skin and crisp flesh.</p>

<p >Apple Inc is an American company that sells consumer electronics, computer software, and online services. Steve Jobs was one of the co-founder of the company. </p>

<div class="search"> Search the Web </div>

<p> Use Google to know more about Steve Jobs. </p>

</div>

</body>

</html>

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

17

[Use Google to know more about Steve Jobs.](#)

Figure 18 shows the result if you use the following rule:

```
p + div{  
color: blue;  
}
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

18

Use [Google](#) to know more about Steve Jobs.

Figure 19 shows the result if you use the following rule:

```
p + p{  
color: blue;  
}
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

19

Use [Google](#) to know more about Steve Jobs.

Selecting General Sibling Selector

This selector matches elements that are *sibling* of the *specified element*. To create a general sibling selector, place a *tilde character* (~) between the simple selectors. The combinator matches the second element only if preceded by the first. Also, both should share a *common parent*. In Listing cs2.18, I am looking to match all *p* elements if it is a sibling of a *p* element. In other words, I am looking to select all paragraphs that are at the same hierarchy level. Figure 20 shows the render of Listing cs2.18.

[Listing cs2.18:/cs2/combing-selectors-6.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> Combining Selectors </title>
<style>
body{
line-height: 1.5;
}
p~p{
color: blue;
}
</style>
</head>
<body>
<div>
<h1> About Apple </h1>
<p ><a href="www.apple.com" >Apple Inc</a> is an American <span class="apple-1"> fruit</span> of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p ><a href="http://en.wikipedia.org/wiki/Steve_Jobs" >Steve Jobs</a> was one of the co-founder of the company. </p>
<div class="search"> Search the Web </div>
<p> Use <a href="https://www.google.com">Google</a> to know more about Steve Jobs. </p>
</div>
</body>
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

20

[Use Google](#) to know more about Steve Jobs.

Figure 21 shows the result, if you use the following rule:

```
div~p{  
color: blue;  
}
```

Here, I am looking for all the *p* elements that come after the *div* element.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

21

[Use Google](#) to know more about Steve Jobs.

Using Pseudo-Element Selectors

The *pseudo elements* allow you to style certain parts of a document. For example, the *::first-line* pseudo-element is used to target the *first line* of an element.

Following is the syntax for pseudo elements:

```
selector::pseudo-element {  
    property:value;  
}
```



: Vs ::

Sometimes, `:` is used in place of `::`. The *double-colon* `::` convention is an attempt to distinguish between *pseudo*-classes and *pseudo*-elements. However, most of the browser supports both conventions.



Selection pseudo-element

The *Selection* pseudo-element always starts with a double colons (`::`).

In this section, I will discuss various pseudo-elements.

Using the `::first-line` Selector

The `::first-line` selector applies styles only to the first line of an element, it does not target any real HTML element. The volume of text depends on many factors such as width of the element is one of them. In Listing cs2.19, I am selecting first line of each paragraph. Figure 22 shows the result.

[Listing cs2.19:/cs2/first-line-selector.html](#)

```
<!DOCTYPE html>  
<html>  
<head>  
<title> First Line Selector </title>  
<style>  
p::first-line{  
background-color: grey;  
color: black;
```

```
}

</style>

</head>

<body>

<h2> About Apple </h2>

<p>Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.</p>

<p>Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>

</body>

</html>
```

About Apple

Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.

22

Apple Inc. is an American company that sells consumer electronics, computer software, and online services.



::first-line Selector

You can use the `::first-line` selector only with the *block-level* elements.

Using the ::first-letter Selector

The `::first-letter` selector allows you to select *first letter* of the first line of block. In Listing cs2.20, the selector applies *red* color to the *first letter* of the *paragraph* assigned with `id1` selector. Figure 23 shows the result of Listing cs2.20.

[Listing cs2.20: /cs2/first-letter-selector.html](#)

```
<!DOCTYPE html>

<html>

<head>
```

```
<title> First Letter Selector </title>
<style>
p#id1::first-letter{
color: red;
font-size: 200%;
}
</style>
</head>
<body>

<h2> About Apple </h2>
<p id="id1">Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>

</body>
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.

23

Apple Inc. is an American company that sells consumer electronics, computer software, and online services.

Using the :before and :after Selectors

These two selectors are used to add *cosmetic content* to the page using the *content* property. These elements are *inline* by default. In Listing cs2.21, I have placed two images before and after the *h2* element. Figure 24 shows the result.

[Listing cs2.21: /cs2/before-after-selector.html](#)

```

<!DOCTYPE html>
<html>
<head>
<title> Class Selector </title>
<style>
h2::before{
content: url("images/aapple1.png");
}
h2::after{
content: url("images/apple2.png");
}
</style>
</head>
<body>

<h2> About Apple </h2>
<p id="id1">Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>
</body></html>

```



About Apple

Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.

24

Apple Inc. is an American company that sells consumer electronics, computer software, and online services.

Using the ::Selection Selector

You can use this selector to a portion of the document that is selected by the user in the browser using mouse or any other pointing device. In Listing

cs2.22, this selector applies *chocolate color background* and *white colored text* for the selected text in the browser. Figure 25 shows the result.

About Apple

Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.

25

Apple Inc. is an American company that sells consumer electronics, computer software, and online services.

Listing cs2.22: /cs2/selection-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> ::selection Selector </title>
<style>
::selection {
background: chocolate;
color: white;
}
</style>
</head>
<body>

<h2> About Apple </h2>
<p>Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>

</body>
</html>
```

Using the Pseudo-Classes Selectors

A **pseudo-class** is a keyword that is used with selectors to target a special state of the element. For example, the `:hover` pseudo-class will apply style to an element when the user hovers over the element. Pseudo-classes together with pseudo-elements give you ability to not only select elements from the page hierarchy but also from the history of content. For example, the `:visited` pseudo class allows you to select links that have been visited by the user. In this section, I will explain different pseudo-classes.

Following is the syntax for declaring a pseudo-class:

```
selector:pseudo-class {  
    property: value;  
}
```

Using the `:root` Selector

The `:root` pseudo-class selects the first element in the document tree. You are not going to use this selector very often. The root in an HTML document will be always the `html` element. Listing cs2.23 shows the root selector in use. You can see the effect of this selector in Figure 26.

[Listing cs2.23:/cs2/root-selector.html](#)

```
<!DOCTYPE html>  
<html>  
<head>  
<title>Root Selector</title>  
<style>  
:root{  
    padding: 20px;  
    border: 1px solid blue;  
}  
</style>  
</head>
```

```
<body>
<div class="inheritance">
<h2> Root Selector in Action </h2>
<p>
This is a paragraph. a distinct section of a piece of writing,
usually dealing with a single theme and indicated by a new line,
indentation, or numbering.
</p>
</div>
</body>
</html>
```

Root Selector in Action

This is a paragraph. a distinct section of a piece of writing,
usually dealing with a single theme and indicated by a new line,
indentation, or numbering.

26

Using the :first-child Selector

This class selects any element that is the *first child* element of its *parent*. Listing cs2.24 shows the *:first-child* selector in use.

[Listing cs2.24: /cs2/firstchild-selector.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> First Child Selector </title>
<style>
body{
line-height: 1.5;
}
```

```
p > a:first-child{  
color:red;  
}  
</style>  
</head>  
<body>  
<div id="desc">  
<h1> About Apple </h1>  
<p > Apple is the round <span class="apple-1"> fruit</span> of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>  
  
<p ><a href="www.apple.com" >Apple Inc</a> is an American <span class="apple-2"> company</span> that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" > Steve Jobs</a> was one of the co-founder of the company. </p>  
<div class="search"> Search the Web </div>  
<p> Use <a href="https://www.google.com"> Google </a> to know more about Steve Jobs. </p>  
</div>  
</body>  
</html>
```

The *:first-child* selector in the Listing cs2.23 will match any *a* element that is *first child* of the *p* element. You can see the match in Figure 27.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

Using the :last-child Selector

This pseudo-class selects any element that is the *last child* of its *parent*. Listing cs2.25 shows this selector in use.

Listing cs2.25:/cs2/lastchild-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> Last Child Selector </title>
<style>
body{
line-height: 1.5;
}
p > a:last-child{
color:red;
}

</style>
</head>
<body>
<div id="desc">
<h1> About Apple </h1>
<p > Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p > <a href="www.apple.com" >Apple</a> Inc is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" >Steve Jobs</a> was one of the co-founder of the company. </p>
<div class="search"> Search the Web </div>
<p> Use Google to know more about Steve Jobs. </p>
```

```
</div>
```

```
</body>  
</html>
```

You can see in Figure 28 the elements this pseudo class matches. Notice there are two *a* elements in the middle paragraph but the *:last-child* selector affected the second *a* element in the paragraph because it is last child of its parent (*p*).

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple](#) Inc is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

28

Use Google to know more about Steve Jobs.

Using the *:only-child* Selector

This class represents any element which is the *only child* of its parent. Listing cs2.26 shows the selector in use.

[Listing cs2.26: /cs2/onlychild-selector.html](#)

```
<!DOCTYPE html>  
<html>  
<head>  
<title> Only Child Selector </title>  
<style>  
body{  
line-height: 1.5;  
}  
p > a:only-child{
```

```

color:red;
}
</style>
</head>
<body>
<div id="desc">
<h1> About Apple </h1>
<p > Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p ><a href="www.apple.com" >Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" > Steve Jobs</a> was one of the co-founder of the company. </p>
<div class="search"> Search the Web </div>
<p> Use <a href="https://www.google.com">Google</a> to know more about Steve Jobs. </p>
</div>
</body>
</html>

```

Figure 29 shows the result of the Listing cs2.26. Notice the selector ignores the first two paragraphs because there is no *a* element in the first paragraph and there are two *a* elements in the second paragraph. It matches the *a* element in the third paragraph as the *a* element is the only child in the third paragraph.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

Using the :only-of-type Selector

This pseudo-class allows you to match any element that has *no siblings* of the *given type* (define by their parent). Listing cs2.27 shows the use of this class.

Listing cs2.27: /cs2/onlyoftype-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> Only of Type Selector </title>
<style>
body{
line-height: 1.5;
}
p > span:only-of-type{
color:red;
}
</style>
</head>
<body>
<div id="desc">
<h1> About Apple </h1>
<p >Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p ><a href="www.apple.com" >Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" > Steve Jobs</a> was one of the co-founder of the company. </p>
<div class="search"> Search the Web </div>
<p> Use <a href="https://www.google.com"> Google</a> to know more about <span class="class1">Steve Jobs.</span></p>
```

```
</div>
</body>
</html>
```

Notice the result in Figure 30, this selector matches the *span* element in the third paragraph because the *p* element has no sibling of type span..

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

30

Use [Google](#) to know more about [Steve Jobs](#).

Using :first-of-type Selector

This pseudo-class matches the *first sibling* of its *type* from the children of its parent. In other words, the *:first-of-type* selector allows you to target the first occurrence of an element within its container. Listing cs2.28 shows the use of this selector.

[Listing cs2.28: /cs2/firstoftype-selector.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> First of Type Selector </title>
<style>
body{
line-height: 1.5;
}
p > span:first-of-type{
```

```

color:red;
}
</style>
</head>
<body>
<div id="desc">
<h1> About Apple </h1>
<p >Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p ><a href="www.apple.com" >Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" > Steve Jobs</a> was one of the co-founder of the company. </p>
<div class="search"> Search the Web </div>
<p>Use <span class="class1">Google</span> search to know more about <span class="class1">Steve Jobs.</span></p>
</div></body> </html>

```

Figure 31 shows the effect of this selector. Notice the class matches the *first span* element in the last paragraph *ignoring* the second span element.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

31

Use [Google](#) search to know more about Steve Jobs.

Using :last-of-type Selector

This pseudo-class matches the *last sibling* of its *type* from the *children* of its

parent. Listing cs2.29 shows the use of this selector.

Listing cs2.29: /cs2/last-of-type-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> Last of Type Selector </title>
<style>
body{
line-height: 1.5;
}
p > span:last-of-type{
color:red;
}
</style>
</head>
<body>
<div id="desc">
<h1> About Apple </h1>
<p >Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p ><a href="www.apple.com" >Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" > Steve Jobs</a> was one of the co-founder of the company. </p>
<div class="search"> Search the Web </div>
<p>Use <span class="class1">Google</span> search to know more about <span class="class1">Steve Jobs.</span></p>
</div>
</body>
```

```
</html>
```

Notice in Figure 32, the selector matches the last span element in the last paragraph.

Using :nth-of-type(n) Selector

The *[nth-of-type\(n\)](#)* selector targets [based on their source order] every element that is the *nth* child of a specific element. *n* can be a *number*, a *keyword*, or *formula*. Listing cs2.30 shows the selector in use.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Search the Web

32

Use Google search to know more about [Steve Jobs](#).

[Listing cs2.30: /cs2/nthoftype-selector.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> nth of Type Selector </title>
<style>
body{
line-height: 1.5;}
div p:nth-of-type(2){
color:red;}
</style>
</head>
```

```
<body>
<div id="desc">
<h1> About Apple </h1>
<p >Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p ><a href="www.apple.com" >Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" >Steve Jobs </a> was one of the co-founder of the company. </p>
<div class="search"> Search the Web </div>
<p>Use <span class="class1">Google</span> search to know more about <span class="class1">Steve Jobs.</span></p>
</div>
</body>
</html>
```

Notice in Figure 33, this selector targets the *middle* paragraph which is the *second child* of the *div* element.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. Steve Jobs was one of the co-founder of the company.

Search the Web

33

Use Google search to know more about Steve Jobs.

Listing cs2.31 shows another use of the selector. You can use *even* and *odd* keywords to select the even and odd children, respectively, of an element. Figure 34 shows the result. Notice that I have used the selector to change the color of the alternate rows.

Listing cs2.31: /cs2/nth-of-type-selector1.html

```
<!DOCTYPE html>
<html>
<head>
<title> nth of Type Selector </title>
<style>
h1{
font-size: 20px;
padding: 10px;
background-color: lightgreen;
width: 200px;
}
#sales-table-1 tr:nth-of-type(odd) {
background-color: slategray;
}
#sales-table-1 tr:nth-of-type(even) {
background-color: lightblue;
}
</style>
</head>
<body>
<h1> [ January Sales Data ] </h1>
<table id="sales-table-1">
<tr><th>Salesman</th> <th>Appointments</th> <th>Sales</th> <th>Expenses</th>
</tr>
<tr><td>Phillip</td><td>110</td><td>57</td><td>2500</td></tr>
<tr><td>Brent</td><td>78</td><td>48</td><td>2000</td></tr>
<tr><td>Bill</td><td>100</td><td>75</td><td>3700</td></tr>
<tr><td>Fred</td><td>95</td><td>56</td><td>4000</td></tr>
```

```

<tr><td>Tim</td><td>99</td><td>45</td><td>6000</td></tr>
<tr><td>Graham</td><td>58</td><td>35</td><td>4500</td></tr>
</table>
</body>
</html>

```

[January Sales Data]

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham	58	35	4500

34

You can also use a *formula*, $(an+b)$ with this selector. In this formula, *n* is the *counter* which starts at *0*, *a* is the *cycle size* and *b* is an *offset value*. In the Listing cs2.32, I have use the formula, $3n+0$. It will select all *tr* elements whose index is multiple of *3*. This selector will target every *third row* of the table, refer to Figure 35.

[Listing cs2.32: /cs2/nth-of-type-selector.html](#)

```

<!DOCTYPE html>
<html>
<head>
<title> nth of Type Selector </title>
<style>
h1{
font-size: 20px;
padding: 10px;
background-color: lightgreen;
width: 200px;
}

```

```

#sales-table-1 tr:nth-of-type(3n+0){
background-color: lightgray;
}
</style>
</head>
<body>
<h1> [ January Sales Data ] </h1>
<table id="sales-table-1">
<tr><th>Salesman</th> <th>Appointments</th> <th>Sales</th> <th>Expenses</th>
</tr>
<tr><td>Phillip</td><td>110</td><td>57</td><td>2500</td></tr>
<tr><td>Brent</td><td>78</td><td>48</td><td>2000</td></tr>
<tr><td>Bill</td><td>100</td><td>75</td><td>3700</td></tr>
<tr><td>Fred</td><td>95</td><td>56</td><td>4000</td></tr>
<tr><td>Tim</td><td>99</td><td>45</td><td>6000</td></tr>
<tr><td>Graham</td><td>58</td><td>35</td><td>4500</td></tr>
</table>
</body>
</html>

```

[January Sales Data]

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham	58	35	4500

35

In the formula $an+b$, a is the integer value, n is the literal letter n , $+$ is an operator [can be $+$ or $-$], and b is an integer. The equation iterates through each sibling element. If you want to select every second element, you will use the $2n$ equation. Each time an element is checked n is increased by 1 [as in arrays]. The index starts from 0 . If the order of the element matches the result of the equation, the element is selected. The equation $2n$ will select 2^{nd} [2x1], 4^{th} [2x2], 6^{th} [2x3], and so forth elements.

You can use the following link to test the structural pseudo-classes:
<http://lea.verou.me/demos/nth.html>.

Using :nth-child(n) Selector

This class matches the elements that are *nth* child of their parent. In the Listing cs2.32, the *span:nth-child(2)* style targets every *span* element that is *second child* of its parent. Figure 36 shows the result.

Listing cs2.33: /cs2/nthchild-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> nth of Type Selector </title>
<style>
body{
line-height: 1.5;
}
span:nth-child(2){
color: red;
}
</style>

</head>

<body>
<div id="desc">
<h1> About Apple </h1>
<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
```

```
<p><a href="www.apple.com" >Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" > Steve Jobs</a> was one of the co-founder of the company. </p>
```

```
<p>Use <span class="class1">Google</span> search to <span class="class2">know more </span> about <span class="class3">Steve Jobs.</span></p>
```

```
</div>
```

```
</body>
```

```
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

36

Use Google search to **know more** about Steve Jobs.

Using :nth-last-child(n) Selector

In the Listing cs2.33, the *span:nth-last-child(3)* style selects every *span* element that is *third child* of its *parent* counting from the *last child*. Figure 37 shows the result.

Listing cs2.34: /cs2/nthlastchild-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> nth of Type Selector </title>
```

```
<style>
body{
line-height: 1.5;
}
span:nth-last-child(3){
color: red;
}

</style>
</head>
<body>
<div id="desc">
<h1> About Apple </h1>
<p >Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p ><a href="www.apple.com" >Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" > Steve Jobs</a> was one of the co-founder of the company. </p>

<p>Use <span class="class1">Google</span> search to <span class="class2">know more </span> about <span class="class3">Steve Jobs</span> and <span class="class3">Apple Inc.</span></p>
</div>

</body>
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

37

Use Google search to [know more](#) about Steve Jobs and Apple Inc.

Using :nth-last-of-type(n) Selector

This class selector matches every element that is *nth* child of *a particular type* counting from the last child. In Listing cs2.35, this selector matches third *span* element [counting from last child of parent, *p*] of the last paragraph. Figure 38 shows the result.

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

38

Use Google search to [know more](#) about Steve Jobs and Apple Inc.

[Listing cs2.35: /cs2/nthoflasttype-selector.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> nth of Type Selector </title>
<style>
body{
```

```

line-height: 1.5;
}

span:nth-last-of-type(3){
color: green;
}


```

</style>

</head>

<body>

<div id="desc">

<h1> About Apple </h1>

<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

<p>Apple Inc is an American company that sells consumer electronics, computer software, and online services. Steve Jobs was one of the co-founder of the company. </p>

<p>Use Google search to know more about Steve Jobs and Apple Inc.</p>

</div>

</body>

</html>

Using the :enabled, :disabled, :checked, and :default Selectors

The *enabled* and *disabled* classes represent an enabled and disabled elements, respectively. The checked CSS selector represents any *radio*, *check box*, or *option* element that is *checked* or is in the *on* state. The *default* selector targets an element that is default [such as the *Submit* button]. Listing cs2.36 shows the use of these classes.

[Listing cs2.36: /cs2/enableddisabled-selector.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> :enabled, :disabled Selector </title>
<style>
body{
line-height: 1.5;
padding: 10px;
}
input:enabled{
background-color:#e5dded;
}
input:disabled{
background-color: lightblue;
}
input:checked+span{
color:#931505;
}
:default{
color: red;
}
</style>
</head>
<body>
<form action="example.com">
<p>
<label for="userid">User ID:</label>
<input type="text" id="userid"><br />
<label for="emailID">Email ID:</label>
<input type="text" id="emailid" disabled="disabled"><br/>
```

```

<label for="pw">Password:</label>
<input type="password" id="pw"><br/>

<label for="rememberme">Remember me:</label>
<input type="checkbox" id="rememberme" checked>
<span> Ensure cookies are enabled !</span> <br/>
</p>
<input type="submit" value="Submit" />
<a href="example.com"> Reset Password </a>
</form>
</body>
</html>

```

Figure 39 shows the result of the Listing cs2.36. Notice the color of the *Submit* button is *red* because it is the *default* element in the *form* element. There isn't much styling you can use with check boxes, therefore, I have added a rule *input:checked+span* that targets the text attached to check box using the *span* element.

User ID:

Email ID:

Password:

Remember me: Ensure cookies are enabled !

39

Submit | [Reset Password](#)

Using the :valid, :invalid, and :required Selectors

The *:valid* class targets those input elements whose content validates correctly whereas the *:invalid* class targets the elements whose content fails to validate. The *:required* targets those *input* elements whose required attribute is turned on. Listing cs2.37 shows the use of these classes. Figure 40 shows the result.

[Listing cs2.37: /cs2/validinvalid-selectors.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> :valid, :invalid, and :required Selectors </title>
<style>
body{
line-height: 1.5;
padding: 10px;
}
input:invalid{
background-color: #807b7b;
}
input:valid{
background-color: lightgreen;
}
input:required{
background-color: chocolate;
}
</style>
</head>
<body>
<form action="example.com">
<p>
<label for="userid">User ID:</label>
<input type="text" id="userid" required><br />
<label for="emailID">Email ID:</label>
<input type="email" id="emailid" ><br/>
<label for="pw">Password:</label>
<input type="password" id="pw"><br/>
<label for="rememberme">Remember me:</label>
```

```

<input type="checkbox" id="rememberme" checked>
<span> Ensure cookies are enabled !</span> <br/>
</p>
<input type="submit" value="Submit" />
<a href="example.com"> Reset Password </a>
</form>
</body>
</html>

```

User ID:

Email ID:

Password:

Remember me: Ensure cookies are enabled !

40

[Reset Password](#)

Using the :optional Selector

This class selects an *input element* that does not have the *required* option turned on. Listing cs2.38 demonstrate the use of this class. Figure 41 shows the result.

Listing cs2.38: /cs2/optional-selector.html

```

<!DOCTYPE html>
<html>
<head>
<title> :optional Selector </title>
<style>
body{
line-height: 1.5;
padding: 10px;
}
input:optional{
background-color: yellow;

```

```

    }
</style>
</head>
<body>
<form action="example.com">
<p>
<label for="userid">User ID:</label>
<input type="text" id="userid" required><br />
<label for="emailID">Email ID:</label>
<input type="email" id="emailid" ><br/>
<label for="pw">Password:</label>
<input type="password" id="pw"><br/>

<label for="rememberme">Remember me:</label>
<input type="checkbox" id="rememberme" checked>
<span> Ensure cookies are enabled !</span> <br/>
</p>
<input type="submit" value="Submit" />
<a href="example.com"> Reset Password </a>
</form>
</body>
</html>

```

User ID:

Email ID:

Password:

Remember me: Ensure cookies are enabled !

41

[Submit](#) [Reset Password](#)

Using :link and :visited Selectors

The **:link** class selects the links inside elements. It selects any link that has not visited yet. The **:visited** class selects those links that have already been visited. Listing cs2.39 shows the demonstration of these classes. Figure 42 shows the result.

Listing cs2.39: /cs2/linkvisited-selectors.html

```
<!DOCTYPE html>
<html>
<head>
<title> Attribute Selector </title>
<style>

a:link {
border: 1px solid black;
background-color: lightgrey;
padding:4px;
margin:4px;
}
a:visited{
background-color: lightgreen;
}
</style>
</head>
<body>

<h1> About Apple </h1>
<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p ><a href="www.apple.com">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs">Steve Jobs</a> was one of the co-founder of the company. </p>

</body>
</html>
```

About Apple

42

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Using the :hover Selector

This class selector matches elements when you *hover* your mouse over them. Listing cs2.40 shows the demonstration of this class. Figure 43 shows the result.

Listing cs2.40: /cs2/hover-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> Hover Selector </title>
<style>
a:link {
border: 1px solid black;
background-color: lightgrey;
padding:4px;
margin:4px;
}
a:hover{
background-color: lightgreen;
}
</style>
</head>
<body>
<h1> About Apple </h1>
```

```
<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
```

```
<p ><a href="www.apple.com">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs">Steve Jobs</a> was one of the co-founder of the company. </p>
```

```
</body>
```

```
</html>
```

About Apple

Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

43

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.



:hover Selector

You can use the `:hover` selector on all elements, not just links.

Using :active Selector

This class targets an element when it is being *activated* by the user. If you are using *mouse*, the active state is when you *press* the *mouse button* and then *release* it. Listing cs2.41 and Figure 44 demonstrate the use of this selector.

[Listing cs2.41: /cs2/active-selector.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> :active Selector </title>
<style>
body{
```

```

line-height: 1.5;
padding: 10px;
}

a:active{
border: 3px dotted red;
}

</style>
</head>
<body>
<form action="example.com">
<p>
<label for="userid">User ID:</label>
<input type="text" id="userid" required><br />
<label for="emailID">Email ID:</label>
<input type="email" id="emailid" ><br/>
<label for="pw">Password:</label>
<input type="password" id="pw"><br/>

<label for="rememberme">Remember me:</label>
<input type="checkbox" id="rememberme" checked>
<span> Ensure cookies are enabled !</span> <br/>
</p>
<input type="submit" value="Submit" />
<a href="example.com"> Reset Password </a>
</form>
</body>
</html>

```

User ID:

Email ID:

Password:

Remember me: Ensure cookies are enabled !



Styling Links

In order to style links properly, you need to define the `:active` rule after all link related rules. You should style the links using the **LVHA** order: `:link`, `:visited`, `:hover`, and `:active`.

Using :focus Selector

This class is used to style elements when they *receive focus*. Listing cs2.42 demonstrates the use of this selector. Figure 45 shows the result.

Listing cs2.42: /cs2/focus-selector.html

```
<!DOCTYPE html>
<html>
<head>
<title> :focus Selector </title>
<style>
body{
line-height: 1.5;
padding: 10px;
}
:focus{
border: 3px dotted red;
}
</style>
</head>
<body>
<form action="example.com">
<p>
<label for="userid">User ID:</label>
<input type="text" id="userid" required><br />
<label for="emailID">Email ID:</label>
```

```

<input type="email" id="emailid"><br/>
<label for="pw">Password:</label>
<input type="password" id="pw"><br/>

<label for="rememberme">Remember me:</label>
<input type="checkbox" id="rememberme" checked>
<span> Ensure cookies are enabled !</span> <br/>
</p>
<input type="submit" value="Submit" />
<a href="example.com"> Reset Password </a>
</form>
</body>
</html>

```

User ID:

Email ID:

Password:

Remember me: Ensure cookies are enabled !

[Reset Password](#)

45

Using :empty Selector

This selector matches the elements that define *no children*. Listing cs2.43 shows the use of this selector. Notice in Figure 46 the *empty td* element has the *red background color*.

[Listing cs2.43: /cs2/empty-selector.html](#)

```

<!DOCTYPE html>
<html>
<head>
<title> :empty Selector </title>
<style>

```

```

body{
line-height: 1.5;
padding: 10px;
}

td:empty{
background-color: red;
}

</style>
</head>
<body>
<table>
<tr><td>1</td><td>One</td></tr>
<tr><td>2</td><td>Two</td></tr>
<tr><td>3</td><td></td></tr>
</table>
</body></html>

```

1 One

2 Two

3 

46

Using :lang() Selector

This class styles elements based on the *lang* attribute. Listing cs2.44 shows the use of this selector. Figure 47 shows the result.

Listing cs2.44: /cs2/lang-selector.html

```

<!DOCTYPE html>
<html>
<head>
<title> Lang Selector </title>
<style>

```

```

:lang(en){color:red}
:lang(de){color:green}
:lang(fr){color:blue}

</style>
</head>
<body>

<p lang="en"> Write text in English here !</p>
<p lang="de"> Write text in German here !</p>
<p lang="fr"> Write text in French here !</p>

</html>

```

Write text in English here !

Write text in German here !

47

Write text in French here !

Using :target Selector

This class styles a unique element with an *id* matching the fragment identifier of the *url* of the document. To see the use of this selector, run the Listing cs2.45 using the following url: [target-selector.html#id1](#) or [target-selector.html#id2](#). Figure 48 shows the result.

[Listing cs2.45: /cs2/target-selector.html](#)

```

<!DOCTYPE html>
<html>
<head>
<title> Target Selector </title>
<style>
:target{
border: 1px solid #222222;
padding: 10px;
}

```

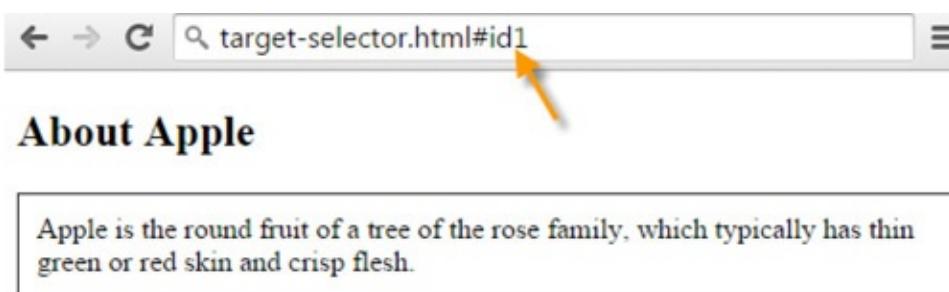
```

</style>
</head>
<body>

<h2> About Apple </h2>
<p id="id1">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p id="id2" ><a href="www.apple.com">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" target="_blank">Steve Jobs</a> was one of the co-founder of the company. </p>

</body>
</html>

```



48

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

Using :not() Selector

The `:not` selector also known as *negation* selector. It uses the following syntax: `not(x)`. It matches an element that is not represented by the `x`. Listing cs2.46 shows the use of this selector and Figure 49 shows the result.

Listing cs2.46: /cs2/negation-selector.html

```

<!DOCTYPE html>
<html>
<head>
<title> Negation Selector </title>
<style>

```

```

p:not(.class1){
border: 1px dotted #CCC;
color:red;
}

body :not(p){
color:blue;
}

</style>
</head>
<body>

<h2> About Apple </h2>
<p class="class1"><a href="www.apple.com"> Apple Inc</a> is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p class="class2"> Apple Inc is an American company that sells consumer electronics, computer software, and online services. <a href="http://en.wikipedia.org/wiki/Steve_Jobs" target="_blank">Steve Jobs</a> was one of the co-founder of the company. </p>

</body>
</html>

```

About Apple

[Apple Inc](#) is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.

Apple Inc is an American company that sells consumer electronics, computer software, and online services. [Steve Jobs](#) was one of the co-founder of the company.

49



Negation Selector

X must not contain another *negation selector*.

Exercises

[Exercise CS2-E1](#)

Using the template shown in Listing cs2-e1, change the color of the *first letter* of the paragraph to *blue* using the paragraph defined by *id1*. Also, scale font size to *130%*.

[Listing cs2-e1: /cs2/cs2-e1.html](#)

```
<!DOCTYPE html>
<html>

<head>
<title> First Letter Selector </title>
<style>
</style>
</head>
<body>

<h2> About Apple </h2>
<p id="id1">Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>

</body>

</html>
```

[Exercise CS2-E2](#)

Using the template shown in Listing cs2-e2, change the *color* of the *h1* element to *blue* that is *first child* of its parent.

[Listing cs2-e2: /cs2/cs2-e2.html](#)

```
<!DOCTYPE html>
<html>

<head>
<title> Child Selector </title>
</head>

<body>
<div>
<h1> About Apple </h1>
<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>
</div>
</body></html>
```

Exercise CS2-E3

Use template shown in Listing cs2-e3. Create a rule for the *h1* and *p* elements when they are *active* using the following declarations:

1. Set *background color* to *light yellow*.
2. Set *1px dotted* border with *black* color.
3. Assign *padding* of *10px*.

Listing cs2-e3: /cs2/cs2-e3.html

```
<!DOCTYPE html>
<html>

<head>
<title> Active Elements </title>
```

```
<style>
</style>
</head>

<body>
<div>
<h1> About Apple </h1>
<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer software, and online services. </p>
</div>
</body>

</html>
```

Exercise CS2-E4

Use the template shown in Listing cs2-e4. Change the *color* of the *Apple Inc* link to *red* when use hovers mouse over it. Change it to *blue* when the link is *active*.

Listing cs2-e4: /cs2/cs2-e4.html

```
<!DOCTYPE html>
<html>

<head>
<title> Psuedo Classes</title>
<style>
</style>
</head>
```

```
<body>
<div>
<h1> About Apple </h1>
<p>Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p><a href="http://www.apple.com" class="apple">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. </p>
</div>
</body>

</html>
```

Exercise CS2-E5

Use the template shown in Listing cs2-e5. Place the *underline* under the *first line* of the paragraph defined by *id1*.

Listing cs2-e5: /cs2/cs2-e5.html

```
<!DOCTYPE html>
<html>
<head>
<title> First Line Selector </title>
<style>
</style>
</head>

<body>

<h2> About Apple </h2>
<p id="id1">Apple is the round fruit of a tree of the rose family. It typically has thin green or red skin and crisp flesh.</p>
<p>Apple Inc. is an American company that sells consumer electronics, computer
```

software, and online services. </p>

</body>

</html>

Summary

In this chapter, I explained various CSS selectors such as basic selectors, pseudo-element selectors, and pseudo-class selectors. The knowledge you have gained in this chapter will help you immensely in the later chapters.

Unit CS3: Working With the Box Model

In the previous chapter, I have explained various *class* and *element* selectors that you can use to effectively select and style elements. In this chapter, I will cover those properties that are used to style the box.

Key Points and Concepts

- Understanding the Box model
- Assigning padding to the elements
- Assigning margin to the elements
- Centering content on a page
- Working with the box dimensions
- Setting minimum and maximum sizes of the box
- Dealing with the overflowing content
- Hiding elements
- Switching between the inline and block-level display
- Creating floating boxes
- Creating box shadows

Padding

The *padding* properties allow you to *add space* between the *content* and *its border*. You can set padding for *individual edges* or you can use the *padding shorthand* to declare padding in one line. Generally, padding is specified using *pixels*. However, if padding is specified in *percentages*, the padding is a percentage of the browser window or of the containing box. Listing cs3.1

shows how you can apply padding to an element.



Width of the Box

If a `width` is specified for the box, the padding is added to the `width` of the box.

Listing cs3.1: /cs3/padding-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> Padding Properties </title>
<style>
p{
border: 1px dotted blue
}
p.example{
padding-top: 10px;
padding-right: 5px;
padding-bottom: 10px;
padding-left: 5px;
border: 1px dotted blue;
}
</style>
</head>
<body>
<div>
<h1> About Apple </h1>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family</p>
<p class="example"><a href="http://www.apple.com" class="apple">Apple Inc</a> is an American company that sells consumer
```

```
electronics, computer software, and online services. </p>
</div>
</body>
</html>
```

In Listing cs3.1, I have used different padding for *top/bottom* and *left/right* edges of the box. It is evident from the Figure 1 that the second paragraph on which I have applied the padding is more readable than the first paragraph. You can also apply equal padding to all four edges of the box using the *padding* property. For example, if you want to assign *10px* padding to all sides, use the following CSS rule: `padding: 10px;`

About Apple

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family

[Apple Inc](#) is an American company that sells consumer electronics, computer software, and online services.

1

You can also use the *shorthand* where the values will be in clock-wise order: *top*, *right*, *bottom*, and *left*. For example, `padding: 10px, 5px, 3px, 6px;`. The padding property can use any of the syntax shown in Table cs3.1.

Table cs3.1: The *padding* property syntax

Property	Description
<code>padding: 10px 5px 8px 12px;</code>	Top: 10px, Right: 5px, Bottom: 8px, Left: 12px
<code>padding: 10px 5px 8px;</code>	Top: 10px, Bottom: 8px, Left & Right: 5px
<code>padding: 10px 15px;</code>	Top & Bottom: 10px, Left & Right: 15px
<code>padding: 10px</code>	All four edges of the box will have 10px padding



padding Property

The value of the *padding* property is not *inherited automatically*. You need to specify padding for *each element* that you need to use.

Margin

The *margin* properties allow you to adjust *space around an element* [outside the border]. It controls the gap between the boxes. Generally, the margin is specified using *pixels* however you can also use *percentages* and *ems*. You can specify the margin using the *margin-top*, *margin-right*, *margin-bottom*, and *margin-left* properties. You can also use a *shorthand* margin value to change all margin at once.

The *margin* property can use any of the syntax shown in Table cs3.2.

Table cs3.2: The <i>margin</i> property syntax	
Property	Description
margin: 10px 5px 8px 12px;	Top: 10px, Right: 5px, Bottom: 8px, Left: 12px
margin: 10px 5px 8px;	Top: 10px, Bottom: 8px, Left & Right: 5px
margin: 10px 15px;	Top & Bottom: 10px, Left & Right: 15px
margin: 10px	All four edges of the box will have 25px margin



margin property

The value of the *margin* property is *not inherited* automatically. You need to specify *margin* for each element that you need to use.

Listing cs3.2 shows the use of the *margin* property. You can see the effect in Figure 2. Notice that the *20px* margin is applied to the *top* and *bottom* edges whereas *30px* margin is applied to the *left* and *right* edges.

Listing cs3.2: /cs3/margin-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> About Apple</title>
```

```

<style>
p{
border: 1px dotted blue;
}
p.example{
margin: 20px 30px;
border: 1px dotted blue;
}
</style>
</head>
<body>
<div>
<h1> About Apple </h1>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family</p>
<p class="example"><a href="http://www.apple.com" class="apple">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. </p>
</div>
</body>
</html>

```

About Apple

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family

[Apple Inc](http://www.apple.com) is an American company that sells consumer electronics, computer software, and online services.

2

Centering Content

You can use the *margin* property to center a box on the page or center it in its containing box. To do so, set the *left-margin* and *right-margin* properties to *auto*. In order to center the box, you need to specify width of the box or it will

take entire width available to it. Listing cs3.3 shows the described method in action. Figure 3 shows the box defined by the *example* class at the center of the page.

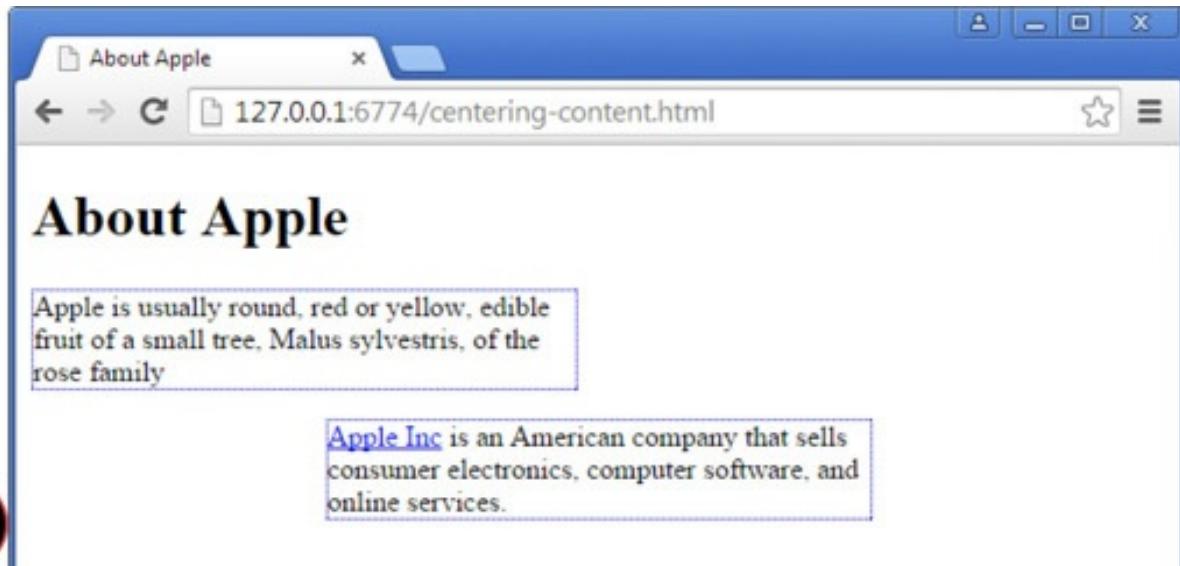
Listing cs3.3: /cs3/centering-content.html

```
<!DOCTYPE html>
<html>
<head>
<title> About Apple</title>

<style>
p{
border: 1px dotted blue;
width: 300px;
}
p.example{
margin: 20px 30px;
border: 1px dotted blue;
margin: 10px auto;
}
</style>
</head>
<body>
<div>
<h1> About Apple </h1>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family</p>
<p class="example"><a href="http://www.apple.com" class="apple">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. </p>
</div>
```

```
</body>
```

```
</html>
```



Box Dimensions

You can use the *height* and *width* properties to set the dimensions of a box. You can set these properties in *pixel*, *percentages*, or *ems*. When you use percentages, the size of the box is *relative* to the browser or to the containing box. When you use *ems*, the size is dependent on the *size of text* within it. While calculating the *width* and *height* of a box, you should also include the *padding*, *border*, and *margin* dimensions. The total width and height should be calculated as follows:

Total Width = Width + Left Padding + Right Padding + Left Border + Right Border + Left Margin + Right Margin

Total Height = Height + Top Padding + Bottom Padding + Top Border + Bottom Border + Top Margin + Bottom margin.

Consider the following rule:

```
div{  
    width=300px;  
    padding:10px;  
    margin-right:10px;  
}
```

The total width of the box will be *340px* [*Width+Left Padding+Right Padding+Margin Right*].

Listing cs3.4 shows the use of the *width* and *height* properties. Figure 4 shows the result. Notice the size of the *div* element defined by the *class2* class is *75%* of the size of the *div* element defined by the *class1* class. In other words, the size of the *div* element defined by the *class2* class is *300 pixel* wide and *225 pixels* high.

Listing cs3.4: /cs3/box-dimensions.html

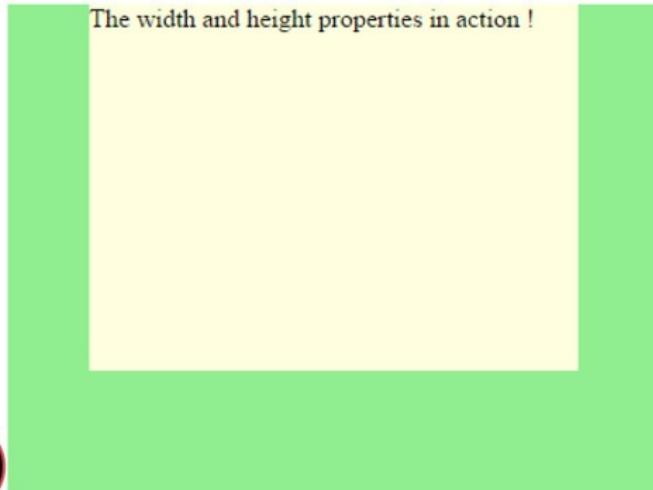
```
<!DOCTYPE html>
<html>
<head>
<title> Box Dimensions </title>

<style>
div.class1 {
width: 400px;
height: 300px;
background-color: lightgreen;
}

div.class2 {
width: 75%;
height: 75%;
background-color: lightyellow;
margin: 10px auto;
}

</style>
</head>
```

```
<body>
<div class="class1">
<div class="class2">
The width and height properties in action !
</div>
</div>
</body></html>
```



Setting Minimum and Maximum Sizes

You can use the *min-* and *max-* properties to set the limits for *width* and *height*. These properties allow browser to freely adjust the content within the limits. The web pages with the responsive design *shrink* to *fit* the screen size. You can control how small a box can be displayed using the *min-width* property. The *max-width* property allows you to set the maximum width a box can stretch. In Listing cs3.5, I have used the *min-width* and *max-width* properties to control the width of the table cells. Figure 5 shows the result. Resize your browser to see the effect.

Listing cs3.5: /cs3/limiting-dimensions.html

```
<!DOCTYPE html>
<html>
<head>
<title> Limiting Dimensions </title>
```

```
<style>
.desc{
min-width: 250px;
max-width: 350px;
}
td, tr{
border: 1px dotted blue;
}
</style>
</head>
<body>
<table class="tble">
<tr>
<td> Icon </td>
<td> Description </td>
</tr>
<tr>
<td></td>
<td class="desc"> Apple is usually round, red or yellow, edible fruit of a small tree,  
Malus sylvestris, of the rose family</td>
</tr>
<tr>
<td></td>
<td class="desc"> Apple is an American company that sells consumer electronics,  
computer software, and online services. </td>
</tr>
</table>
</body>
</html>
```

Icon	Description
	Apple is usually round, red or yellow, edible fruit of a small tree, <i>Malus sylvestris</i> , of the rose family
	Apple is an American company that sells consumer electronics, computer software, and online services.

5

Dealing with Overflowing Content

The *overflow* property allows you to control the behavior of the browser when the content of a box is larger than the box. Table cs3.3 shows the *overflow* properties. Table cs3.4 shows the allowed values for this property.

Table cs3.3: The *Overflow* properties

Property	Description
overflow-x	Sets the horizontal overflow style
overflow-y	Sets the vertical overflow style
overflow	Shorthand property

Table cs3.4: The *Overflow* property values

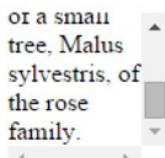
Value	Description
visible	This is the default value. The overflow is not clipped.
hidden	The overflow is clipped and rest of content is hidden.
scroll	The overflow is clipped, scroll bar appears to view the clipped content.
auto	If the overflow is clipped, scroll bar appears to see the rest of the content.
initial	Sets property to its default value.
inherit	Inherits this property from the parent element.

Listing cs3.6 shows the use of the *overflow* property. In Figure 6, you can see two boxes whose contents expand beyond the box size. For the first box, the

[overflow](#) property is set to *scroll*. For the second box, it is set to *hidden*.

overflow:scroll

or a small
tree, Malus
sylvestris, of
the rose
family.



ovreflow:hidden

Apple Inc is an
American
company that
sells consumer
electronics,
computer

6

[Listing cs3.6: /cs3/overflow.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> Overflow </title>
<style>
p.scroll{
width:100px;
height:100px;
overflow: scroll;
}
p.hidden{
width:100px;
height:100px;
overflow: hidden;
}
</style>
</head>
<body>
```

```

<h2>overflow:scroll</h2>
<p class="scroll">Apple is usually round, red or yellow, edible fruit of a small tree,  

Malus sylvestris, of the rose family.</p>
<h2>ovreflow:hidden</h2>
<p class="hidden">Apple Inc is an American company that sells consumer electronics,  

computer software, and online services.</p>
</body>
</html>

```

Hiding Elements

You can hide elements using the [visibility](#) property. This property takes three values: [collapse](#), [hidden](#), and [visible](#). The [default](#) value is [visible](#) which makes the element visible on the page. The [collapse](#) value hides the element but element doesn't occupy space in the page layout. The [hidden](#) value hides the element but element occupies the space on the page.



overflow property

This property is not particularly useful when used on its own however you can create some cool effects with [JavaScript](#).

Element Box Type

The [display](#) property lets you turn an [inline](#) element into a [block-level](#) element or vice versa. It supports many values, the commonly used values this property takes are listed in Table cs3.5.

Table cs3.5: The [display](#) properties

Value	Description
inline	Turns a block-level element to an inline element
block	Turns a inline element to a block-level element
inline-block	Turns a block-level element to an inline element while retaining other features of a block-level element.
none	Hides the elements from the page and it does not take any space in the page.

Listing cs3.7 shows the use of this property. In Figure 7, notice that the list items are displayed as *block-level* elements and are appearing in a *single* row.

Listing cs3.7: /cs3/display.html

```
<!DOCTYPE html>
<html>
<head>
<title> Display </title>
<style>
li {
display: inline;
padding-right: 10px;
}
</style>
</head>
<body>
<ul>
<li>Item-1</li>
<li>Item-2</li>
<li>Item-3</li>
<li>Item-4</li>
<li>Item-5</li>
</ul>
</body>
</html>
```

The `float` property allows you to create the floating boxes. The boxes are shifted to one side of the page until the `left` or `right` edge touches the containing box or another floating box in the page. This property takes three values: `left`, `right`, and `none`, which are self-explanatory.

In Listing cs3.8, I have floated content of the six paragraphs to the `left`. These six paragraphs are contained within the body element whose width is set to `750px`. Figure 8 shows the result.

Listing cs3.8: /cs3/float.html

```
<!DOCTYPE html>
<html>
<head>
<title> Float </title>
<style>
body{
width: 750px;
}
p{
float:left;
width: 200px;
border: 1px dotted blue;
padding: 10px;
margin: 10px;
}
</style>
</head>
<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris,
```

of the rose family. </p>

<p>Apple Inc is an American company that sells consumer electronics, computer software, and online services.</p>

<p>To know more about Apple, use Google search.</p>

<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p>Apple Inc is an American company that sells consumer electronics, computer software, and online services.</p>

<p>To know more about Apple, use Google search.</p>

</body>

</html>

In Figure 8, notice that the fourth paragraph sits below the third paragraph. It did not move below the first paragraph because there is some space below the third paragraph. To overcome this issue, use the *clear* property. I have used this property in Listing cs3.9. Figure 9 shows the result.

Listing cs3.9: /cs3/float-clear.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title> Float </title>
```

```
<style>
```

```
body{
```

```
width: 750px;
```

```
}
```

```
p{
```

```
float: left;
```

```
width: 200px;
```

```
border: 1px dotted blue;  
padding: 10px;  
margin: 10px;  
}  
.
```

```
.clear{  
clear: left;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. </p>
```

```
<p>Apple Inc is an American company that sells consumer electronics, computer software, and online services.</p>
```

```
<p>To know more about Apple, use Google search.</p>
```

```
<p class="clear">Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
```

```
<p>Apple Inc is an American company that sells consumer electronics, computer software, and online services.</p>
```

```
<p>To know more about Apple, use Google search.</p>
```

```
</body>
```

```
</html>
```

8

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Apple Inc is an American company that sells consumer electronics, computer software, and online services.

To know more about Apple, use Google search.

Apple Inc is an American company that sells consumer electronics, computer software, and online services.

To know more about Apple, use Google search.

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Apple Inc is an American company that sells consumer electronics, computer software, and online services.

To know more about Apple, use Google search.

9

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Apple Inc is an American company that sells consumer electronics, computer software, and online services.

To know more about Apple, use Google search.

Creating Box Shadow

The *box-shadow* property allows you to add *drop-shadow* to a box. It takes the following format:

box-shadow: hoffset voffset blur spread color inset;

Table cs3.6 shows the values of the *box-shadow* property.

Table cs3.6: Values of the <i>box-shadow</i> property	
Value	Description
hoffset	Horizontal offset of the shadow. Negative values position the shadow to the left of the box.
voffset	Vertical offset of the shadow. Negative values position the shadow to the top of the box.
blur	[Optional] Blurs the shadow. If omitted, the shadow is a solid line like a border
spread	[Optional] The spread of the shadow. Positive values make the shadow expand in all directions.
color	[Optional] The color of the shadow.

inset [Optional] Makes the shadows to be inset inside the box.

Listing cs3.10 shows the use of the *box-shadow* property. Figure 10 shows the effect of the style.

Listing cs3.10: /cs3/box-shadow-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> Box Shadow </title>
<style>
p{
padding: 10px;
margin: 10px;
box-shadow: 6px 10.392px 4.95px 0.05px #7e8325;
}
</style>
</head>
<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple Inc is an American company that sells consumer electronics, computer software, and online services. </p>
</body>
</html>
```

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple Inc is an American company that sells consumer electronics, computer software, and online services.

10

You can also define multiple shadows in a *single box-shadow declaration*. Listing cs3.11 shows the use. Here, *two shadows* are defined one of which is

inset. Figure 11 shows the result.

Listing cs3.11: /cs3/box-shadow-2.html

```
<!DOCTYPE html>
<html>
<head>
<title> Box Shadow </title>
<style>
p{
padding: 10px;
margin: 10px;
background-color: #7b7f24;
color: white;
box-shadow: 6px 10.392px 4.95px 0.05px #7e8325, inset 10px 5.321px 66.88px 9.12px
#000;
}
</style>
</head>
<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple Inc is an American company that sells consumer electronics, computer software, and online services. </p>

</body>
</html>
```

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple Inc is an American company that sells consumer electronics, computer software, and online services.

The browser vendors regularly add extensions to the existing CSS specifications. These extensions may not be part of the formal specification and generally proprietary to their browsers. In most cases, these features are developed according to the [W3C](#) drafts that have not reached the [*Candidate Recommendation*](#) status. The browser prefixes are used to test the extension before they become part of the regular specification.

Following are some browser prefixes:

Android: -webkit-

Chrome: -webkit-

Firefox: -moz-

Internet Explorer: -ms-

iOS: -webkit-

Opera: -o-

Safari: -webkit-

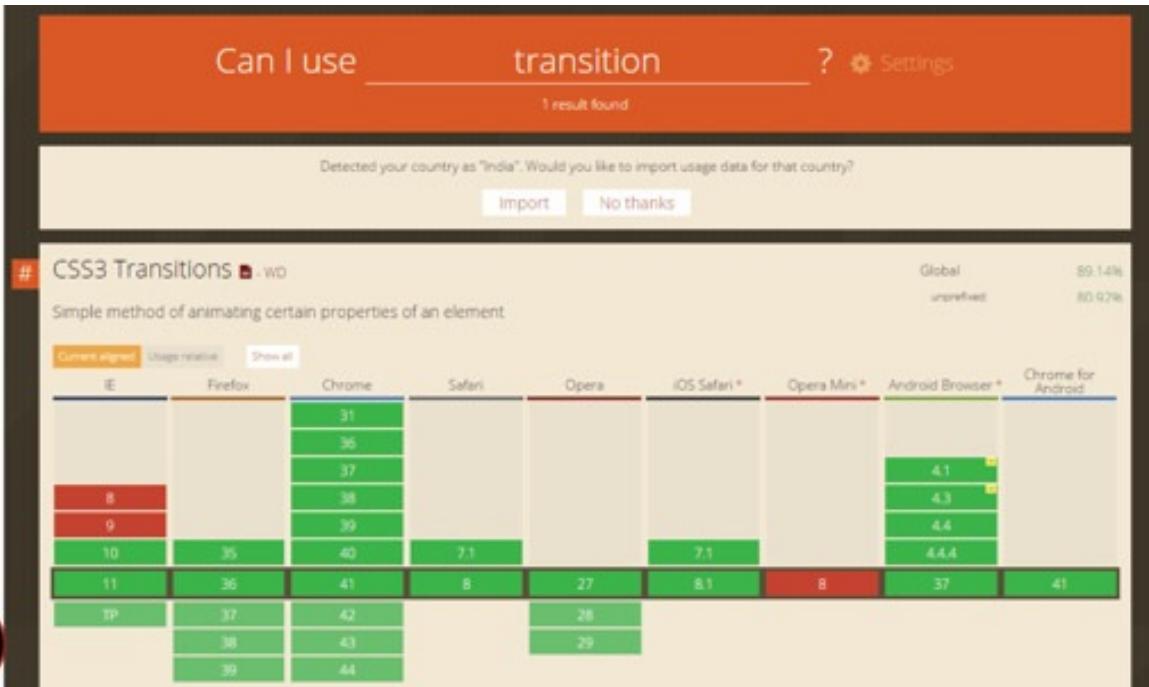
To use the prefix, take the standard CSS property and then add prefix for each browser. For example, if you want to use the prefixes with the [*transition*](#) property, first write the standard syntax and then use the prefixes. For example:

```
transition: all 4s ease;  
-webkit-transition: all 4s ease;  
-moz-transition: all 4s ease;  
-ms-transition: all 4s ease;  
-o-transition: all 4s ease;
```

Note that I have added prefixes in the following format: [*-prefix-*](#).

I personally use the following service to check the Browser compatibility:

<http://caniuse.com>. For example, if you type *transition* in the *Can I use* field, the results with the browser compatibility are displayed below the field (see Figure 12).



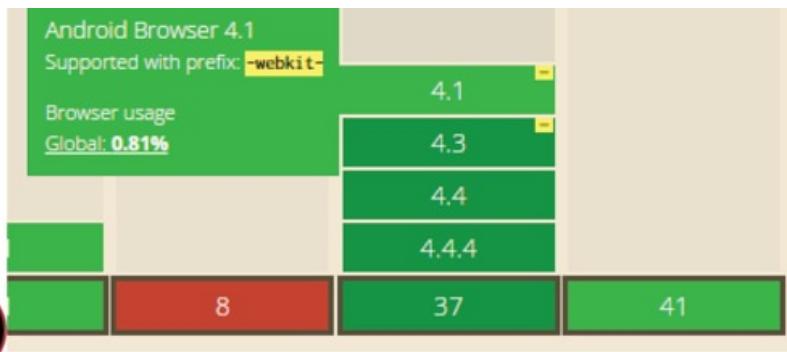
12

Notice in Figure 13, the *transition* property is not supported by *version 8* of the *Opera Mini* browser. Also, if you want to use this property with versions **4.1** and **4.3**, you have to use the prefix **-webkit-**.



Google Chrome

I have used version 41 of the *Google Chrome* browser to test the code in this book.



13

Exercises

Exercise CS3-E1

Using the Listing cs3-e1, assign **20%** padding to the bottom edge of the paragraph defined by the example class.

Listing cs3-e1: /cs3/cs3-e1.html

```
<!DOCTYPE html>
<html>
<head>
<title> About Apple</title>

<style>
</style>
</head>
<body>
<div>
<h1> About Apple </h1>
<p class="example">Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family</p>
<p ><a href="http://www.apple.com" class="apple">Apple Inc</a> is an American company that sells consumer electronics, computer software, and online services. </p>
</div>
</body>
</html>
```

Exercise CS3-E2

Consider the following rule and calculate the *total width* of the element.

```
div {
width: 300px;
padding: 20px;
border: 5px;
margin: 0;
}
```

Exercise CS3-E3

Using the template shown in Listing cs3-e3, specify the following settings:

Apply *padding* of *5px* and *margin* of *10px* to the paragraph. Create a *3px border* around the element with *dotted* style and *#4557bc* color. Change the *border style* to *solid* and *color* to *red* when user hovers mouse over it.

Listing cs3-e3: /cs3/cs3-e3.html

```
<!DOCTYPE html>
<html>

<head>
<title> Border and Backgrounds </title>
<style>
</style>
</head>

<body>
<p class="brdr">Apple is usually round, red or yellow, edible fruit of a small tree,  
Malus sylvestris, of the rose family.</p>

</body>

</html>
```

Summary

In this chapter, I have explained the *box* model to you. You learned about setting the *dimensions* of the box as well as how to adjust *space around the box* and *its contents* using the *padding* and *margin* properties. I also discussed about controlling the *overflowing* content, *hiding* elements, *creating floating* boxes, and switching between the *block* and *inline* display. You also learned about creating *box shadows* for the boxes. The knowledge you gained about the box model in this chapter will help you immensely when you will work on

some examples and exercises in the coming chapters.

Unit CS4: Working With Borders and Backgrounds

In the last chapter, you learned about the *box* model and properties that affect the box such as *padding* and *margin*. These properties are invisible to the naked eye. In this chapter, I introduce the properties that you can use to create *borders* around boxes to properly visualize the *padding* and *margin* spaces. I will also explain how to apply images in the background.

Key Points and Concepts

- Applying a border to an element
- Defining the border width and type
- Border shorthand
- Creating rounded corners
- Using images as borders
- Setting background color and images
- Using the background shorthand
- Creating gradients

Applying a Border to an Element

There are three basic *border* properties that control the border around elements. Table cs4.1 describes these properties. Listing cs4.1 shows these properties in action.

Table cs4.1: Basic border properties	
Property	Description
border-width	Controls the width of the border.
border-style	Controls the style used to draw the border.

border-color	Controls the color of the border.
border-top-width border-top-style border-top-color	Controls the top border.
border-bottom-width border-bottom-style border-bottom-color	Controls the bottom border.
border-left-width border-left-style border-left-color	Controls the left border.
border-right-width border-right-style border-right-color	Controls the right border.

Listing cs4.1: /cs4/applying-border.html

```
<!DOCTYPE html>
<html>

<head>
<title> Applying Border </title>
<style>
p {
padding: 10px;
border-width: 2px;
border-style: double;
border-color: darkred;
}
</style>
</head>

<body>
```

```
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
```

```
<p>Apple Inc is an American company that sells consumer electronics, computer software, and online services.</p>
```

```
</body>
```

```
</html>
```

In Listing cs4.1, I have applied border to the paragraph elements using *border-width*, *border-style*, and *border-color* properties. Figure 1 shows the *border* around the *paragraphs*.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple Inc is an American company that sells consumer electronics, computer software, and online services.

1

You can set the border width using CSS measurement units such as *em*, *px*, or *cm*. You can also specify the *percentages*. When *percentage* is used, browser draws border with *percentage of the width* of the area around which the border is drawn. You can also use *preset widths*. There presets are: *thin*, *medium*, and *thick*.

Values of the border-style Property

In Listing cs4.1, I have used *double* as *border style* which causes border to appear as *two parallel lines* with some *gap* between them. The *border-style* property supports other styles as well. Table cs4.2 lists them.

Table cs4.2: Values of the *border-style* property

Value	Description
none	Displays no border.
hidden	Like none, displays no border. In case, there is conflicting border is set in a table, the hidden value has the highest priority. In other words, the conflicting border will not be displayed.
dotted	Displays border as a series of rounded dots.

dashed	Displays border as a series of short square ended dots.
solid	Displays border as a single, straight, solid line.
double	Displays border as two straight line with a gap between them.
groove	Displays border as leading to a carved effect. It is opposite of the ridge style.
ridge	Displays border with 3D effect.
inset	Displays border in a way that makes the box appear embedded.
outset	Displays border in a way that makes the box appear embossed.

Listing cs4.2 shows the demonstration of the *values* shown in Table cs4.2. Figure 2 shows the result.

Listing cs4.2: /cs4/border-style.html

```
<!DOCTYPE html>
<html>
<head>
<title> Border Style </title>
<style>
body {
width: 750px;
}
p {
float: left;
width: 150px;
height: 100px;
background-color: bisque;
padding: 5px;
margin: 5px;
}
.b1 {
```

```
border-style: none;  
}  
.b2 {  
border-style: hidden;  
}  
.b3 {  
border-style: dotted;  
}  
.b4 {  
border-style: dashed;  
}  
.b5 {  
border-style: solid;  
}  
.b6 {  
border-style: double;  
}  
.b7 {  
border-style: groove;  
}  
.b8 {  
border-style: ridge;  
}  
.b9 {  
border-style: inset;  
}  
.b10 {  
border-style: outset;  
}  
</style>  
</head>
```

<body>

<p class=“b1”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b2”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b3”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b4”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b5”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b6”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b7”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b8”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b9”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

<p class=“b10”>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

</body>

</html>

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Using the border Shorthand Property

You can also use the *border shorthand property* to set all border properties in one declaration. You need to write the values that can be used in the following order: *border-width*, *border-style*, and *border-color*.

In Listing cs4.3, I have used the *border shorthand property* to draw a 5 pixel, *dotted* border around a *paragraph* element. Figure 3 shows the result.

Listing cs4.3: \cs4\border-shorthand.html

```
<!DOCTYPE html>
<html>

<head>
<title> Applying Border </title>
<style>
p {
padding: 5px;
margin: 10px;
border: 3px dotted #4557bc;
```

```

    }
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

</body>

</html>

```

3

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.



border Shorthand Property

If you are using the *border shorthand* property, it does not matter if one of the values is missing. For example, the following example is valid: *border: solid #000;*

Creating Rounded Corners

The *border-radius* property gives you ability to create *rounder corners* on any box. The value of this property indicates the *size* of the *radius* in pixels. Table cs4.3 shows the list of properties for creating *rounder corners*. Notice in Figures 4 and 5, the *radii* values project an oval to create corner of the box.

Table cs4.3: The *Border radius* properties

Property	Description
border-top-left-radius	Controls the radius of the top left corner of the box.
border-top-right-radius	Controls the radius of the top right corner of the box.
border-bottom-left-radius	Controls the radius of the bottom left corner of the box.
border-bottom-right-radius	Controls the radius of the bottom right corner of the box.
border-radius	This is a shorthand property which sets all corners at once.

To define a *curved corner*, you need to specify *two radii values* for the corner. The first value specifies the *horizontal radius* whereas the second value specifies the *vertical radius*. If you are using *percentages*, the values are of the vertical and the horizontal size of the box of the element. Listing cs4.4 shows the use of the *border-bottom-right-radius* property. Figure 4 shows the result.

Listing cs4.4: /cs4/border-radius-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> Border Radius </title>
<style>

div {
width: 300px;
height: 100px;
padding: 20px;
margin: 10px;
border: 1px solid blue;
border-bottom-right-radius: 80px 50px;
}
</style>
</head>

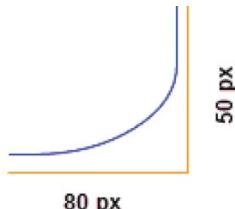
<body>
<div>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</div>

</body>
```

</html>

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

4



5

One Radius Value

If you supply only *one radius*, both the *horizontal* and *vertical* radii will use the *same value*.



Padding Property

If border is touching the text in the element due to any corner of the element, you can create the space between the content and corner by using the *padding* property as I have done in Listing cs4.4.

Using the Shorthand Property

The *border-radius* shorthand property is used to set the four *border-*radius* properties at *once*. For example:

border-radius: 10px;

is equivalent to:

border-top-left-radius:10px;

border-top-right-radius: 10px;

border-bottom-right-radius: 10px;

```
border-bottom-left-radius: 10px;
```

The syntax for the *first radius* allows one of the following *four* values:

```
border-radius: radius
```

```
border-radius: top-left-and-bottom-right top-right-and-bottom-left
```

```
border-radius: top-left top-right-and-bottom-left bottom-right
```

```
border-radius: top-left top-right bottom-right bottom-left
```

The syntax of the *second radius* also allows you to use one of the following *four* values:

```
border-radius: (first radius values) / radius
```

```
border-radius: (first radius values) / top-left-and-bottom-right top-right-and-bottom-left
```

```
border-radius: (first radius values) / top-left top-right-and-bottom-left bottom-right
```

```
border-radius: (first radius values) / top-left top-right bottom-right bottom-left
```

The declaration, *border-radius: 2em 1em 4em / 0.5em 3em;*, is equivalent to:

```
border-top-left-radius: 2em 0.5em;
```

```
border-top-right-radius: 1em 3em;
```

```
border-bottom-right-radius: 4em 0.5em;
```

```
border-bottom-left-radius: 1em 3em;
```

Listing cs4.5 demonstrate an example.

[Listing cs4.5: /cs4/border-radius-2.html](#)

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
<title> Border Radius </title>
<style>
p {
border: 1px solid firebrick;
padding: 10px;
}
p.cls1 {
border-radius: 20px / 15px;
}
p.cls2 {
border-radius: 2em 1em 4em / 0.5em 3em;
}
</style>
</head>

<body>
<p class="cls1">Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
<p class="cls2">Apple Inc is an American company that sells consumer electronics, computer software, and online services.</p>

</body></html>
```

In Listing cs4.5, there are two paragraphs represented by the `cls1` and `cls2` classes, respectively. The declarations for the `cls1` class define two values which are applied to all four corners, refer to Figure 6. Notice the horizontal values are separated from the vertical values using the `/` character.

6

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Apple Inc is an American company that sells consumer electronics, computer software, and online services.

Notice the declarations in the `cls2` class. The first four values represents the *horizontal radius values* followed by a `/` character and then the next four values are *vertical radius values*, refer Figure 6.

Using Images as Borders

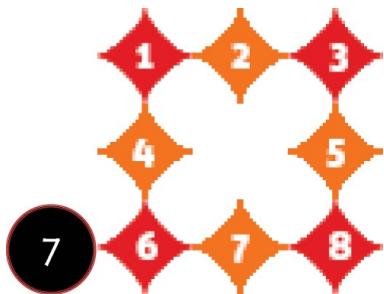
The `border-image` property allows you to draw an image on the borders of an element. It allows you to create beautiful looking borders. Table cs4.4 shows the `border-image` properties.

Table cs4.4: The `border-image` properties

Property	Description
<code>border-image-source</code>	Defines the source of the image.
<code>border-image-slice</code>	Controls the offsets for slicing the image
<code>border-image-width</code>	Controls the width of the border
<code>border-image-outset</code>	Defines the area outside the standard border that is used to show the border of the image.
<code>border-image-repeat</code>	Controls the repetition of the image along the border. It can take four values: <code>stretch</code> , <code>repeat</code> , <code>round</code> , or <code>space</code> .
<code>border-image</code>	This is the shorthand property to set all values at once.

Before you use any image as border, you need to *slice* the image. You use numeric values to specify offset values that browser uses to slice the image into nine parts. To demonstrate how browser slices an image, I have created an image as shown in Figure 7. Each diamond in Figure 7 is `30px` wide and `30px` tall that makes image `90px` wide and `90px` tall. The middle diamond in Figure 7 is transparent. Listing cs4.6 shows the use of the `border-image` properties.

Figure 8 shows the result.



[Listing cs4.6: /cs4/border-image-1.html](#)

```
<!DOCTYPE html>
<html>

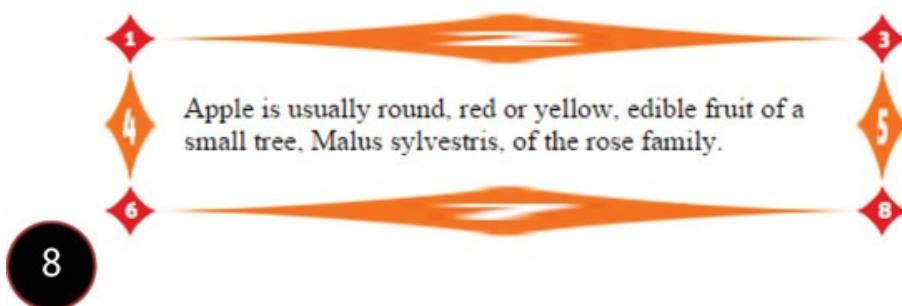
<head>
<title> Border Image </title>
<style>
p {
border: 1px solid firebrick;
padding: 10px;
}
p.cls1 {
border: 30px solid transparent;
border-image: url("images/diamond.png");
border-image-slice: 30;
border-image-outset: 5px;
}
</style>
</head>

<body>
<p class="cls1">Apple is usually round, red or yellow, edible fruit of a small tree,  
Malus sylvestris, of the rose family.</p>
```

```
</body>
```

```
</html>
```

Notice in Figure 8, the diamond marked as **1**, **3**, **6**, and **8** are used to draw the corners of the border whereas the diamond marked as **2**, **4**, **5**, and **7** are used to draw the border edges. Also, I have set **30px** inward offset for the image border as well as **5px** image outset that controls the amount by which the border image area extends beyond the border box.



Specifying Slice Size

Notice the declaration in **p.cls1** rule, while specifying the **slice** size, you don't have to provide the units. The units are assumed **pixels**, by default.

You can also use the **border-image** shorthand property. The syntax for this property is as follows:

border-image: source slice width outset repeat;

Listing cs4.7 shows the use of this property. Figure 9 shows the result. Listing cs4.8 and Figure 10 demonstrate another example.

Listing cs4.7: /cs4/border-image-2.html

```
<!DOCTYPE html>
<html>
<head>
<title> Border Image </title>
```

```

<style>
p {
border: 1px solid firebrick;
padding: 10px;
}
p.cls1 {
border: 15px solid transparent;
border-image: url("images/diamond.png") 30 30 stretch;
}
</style>
</head>

```

```

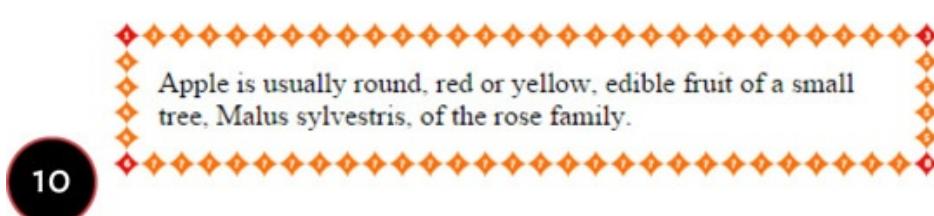
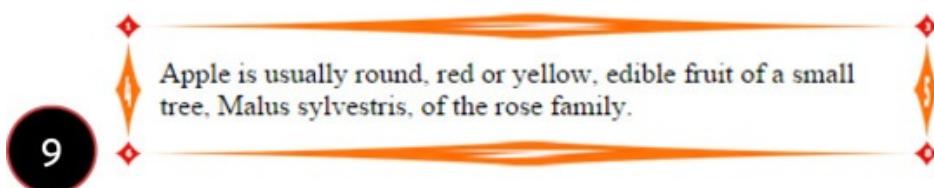
<body>
<p class="cls1">Apple is usually round, red or yellow, edible fruit of a small tree,  

Malus sylvestris, of the rose family.</p>

</body>

</html>

```



Listing cs4.8: /cs4/border-image-3.html

```

<!DOCTYPE html>
<html>

```

```

<head>
<title> Border Image </title>
<style>
p {
border: 1px solid firebrick;
padding: 10px;
}
p.cls1 {
border: 15px solid transparent;
border-image: url("images/diamond.png") 30 30 round;
}
</style>
</head>

<body>
<p class="cls1">Apple is usually round, red or yellow, edible fruit of a small tree,  

Malus sylvestris, of the rose family.</p>

</body>

</html>

```

Listing cs4.8a shows more examples of *border-radius* property. Figure 10a shows the result.

Listing cs4.8a: /cs4/border-radius-3.html

```

<!DOCTYPE html>
<html>

```

```
<head>
<title> Border and Backgrounds </title>
<style>
#box {
width: 200px;
height: 100px;
border: ridge lightblue;
background-color: lightcoral;
border-radius: 120px/36px;
float: left;
margin: 20px;
}
#box1 {
width: 200px;
height: 100px;
border: none;
background-color: lightcoral;
border-radius: 50px 20px;
float: left;
margin: 20px;
}
#box2 {
width: 200px;
height: 100px;
border: groove 8px darkred;
background-color: lightcoral;
border-radius: 50px;
float: left;
margin: 20px;
}
#box3 {
```

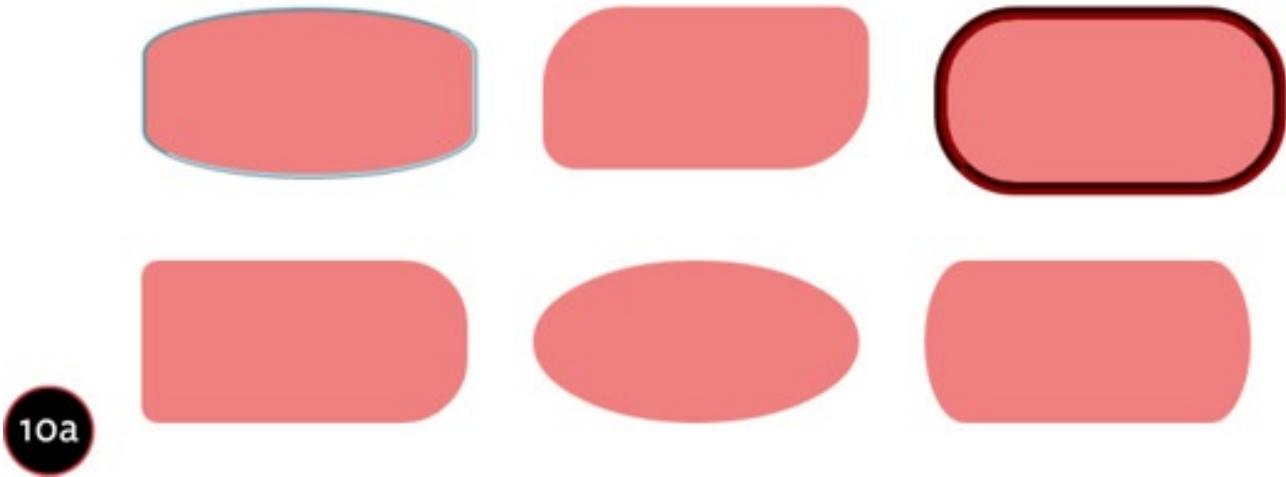
```
width: 200px;  
height: 100px;  
border: none;  
background-color: lightcoral;  
border-radius: 10px 40px 40px 10px;  
float: left;  
margin: 20px;  
}  
  
#box4 {  
width: 200px;  
height: 100px;  
border: none;  
background-color: lightcoral;  
border-radius: 70%;  
float: left;  
margin: 20px;  
}  
  
#box5 {  
width: 200px;  
height: 100px;  
border: none;  
background-color: lightcoral;  
border-radius: 20%/80%;  
float: left;  
margin: 20px;  
}  
</style>  
</head>
```

```
<body>  
<div id="box"></div>
```

```

<div id="box1"></div>
<div id="box2"></div>
<div id="box3"></div>
<div id="box4"></div>
<div id="box5"></div>
</body>
</html>

```



Setting Background Color and Images

You can use the *background* properties to style *background behind content*. The properties are summarized in Table cs4.5.

Table cs4.5: The *background* properties

Property	Description
background	The shorthand property for styling background.
background-color	Defines the background color of the content.
background-image	Sets the background image for element.
background-repeat	Sets the repeat style for the background image.
background-size	Sets the size of the background.
background-position	Sets the position of the background image.
background-	Sets the attachment style. Defines whether the background images are fixed or scrolls with

attachment	the rest of the page content.
background-clip	Defines the painting area of the background image.
background-origin	Sets the positioning area of the background image.

The *background-image* property allows you to place an image *behind* any HTML element. By default, the background image fills the *entire* page or box. In Listing cs4.9, I have used a pattern image (*pattern.jpg*) to apply it to the whole page because it is being used with the *body* element. Figure 11 shows the result.

Listing cs4.9: /cs4/background-image.html

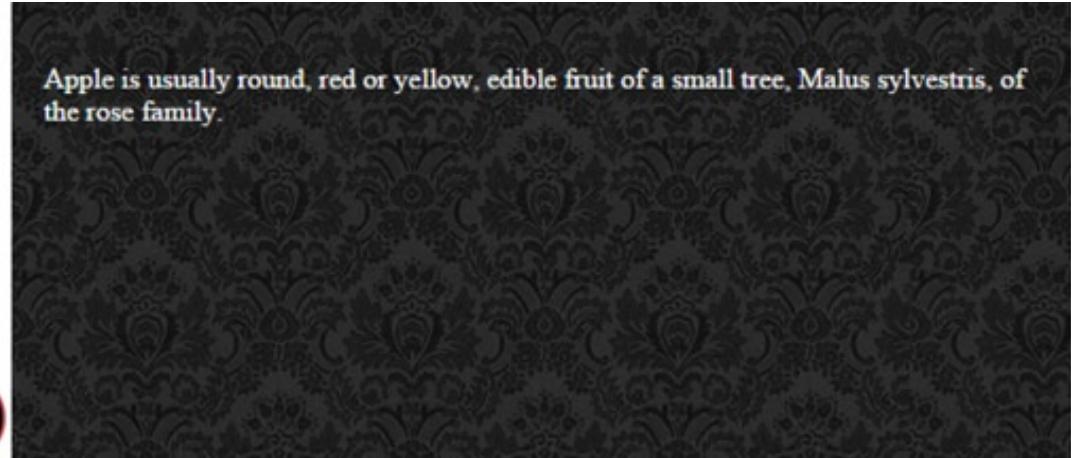
```
<!DOCTYPE html>
<html>

<head>
<title> Background Image </title>
<style>
Body {
Background-image: url(images/pattern.jpg);
padding: 10px;
color: white;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

</body>
```

```
</html>
```



In Listing cs4.10, I have applied the image just to a paragraph. Figure 12 shows the result.

Listing cs4.10: /cs4/background-image-1.html

```
<!DOCTYPE html>
<html>

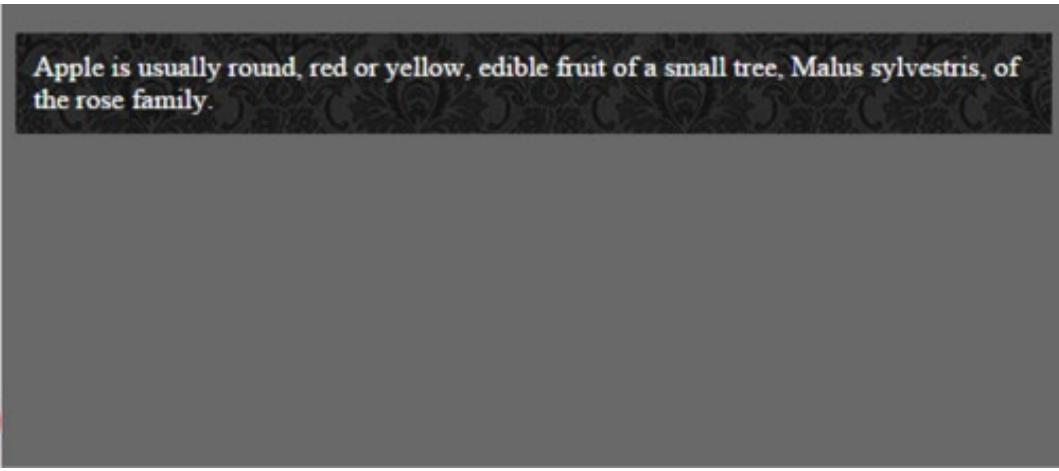
<head>
<title> Background Image </title>
<style>
body {
background-color: dimgray;
}
p {
Background-image: url(images/pattern.jpg);
padding: 10px;
color: white;
}
</style>
</head>
```

```
<body>
```

```
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris,  
of the rose family.</p>
```

```
</body>
```

```
</html>
```



Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

12

The *background-repeat* property sets the background *repeat style*. Table cs4.6 shows the values of this property.

Table cs4.6: The *background-repeat* values

Value	Description
repeat	The background image will be repeated vertically as well as horizontally.
repeat-y	The background image will be repeated vertically.
repeat-x	The background image will be repeated horizontally.
no-repeat	The background image will not be repeated.

In Listing cs4.10, I have set the *image repetition* to *horizontal* by using the *repeat-x* value. Figure 13 shows the result.

Listing cs4.10: /cs4/background-image-2.html

```
<!DOCTYPE html>
```

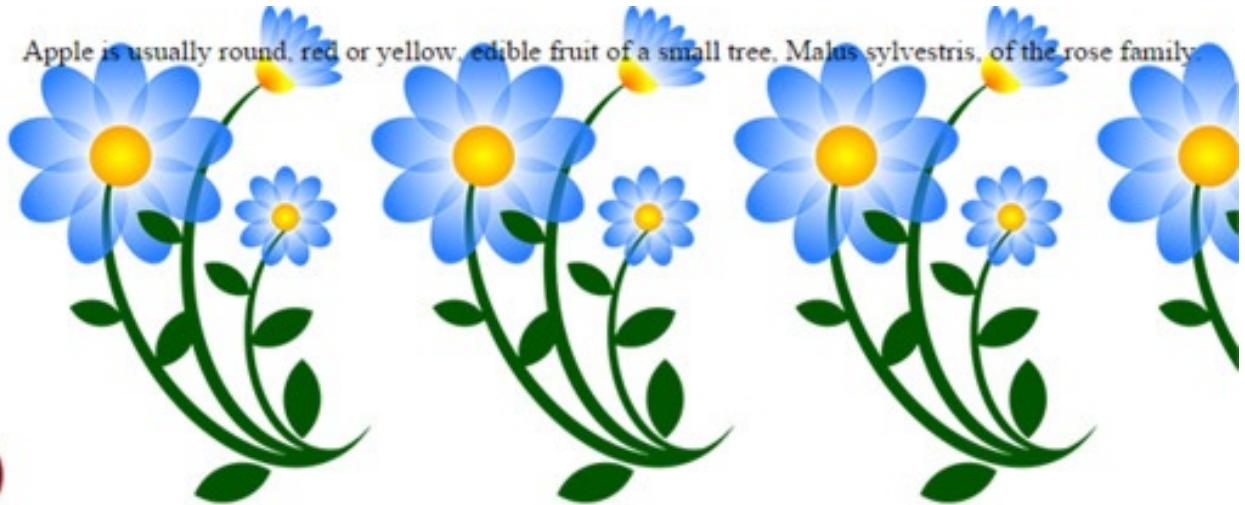
```

<html>

<head>
<title> Background Image </title>
<style>
body {
background-image: url(images/flower.png);
background-repeat: repeat-x;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
</body></html>

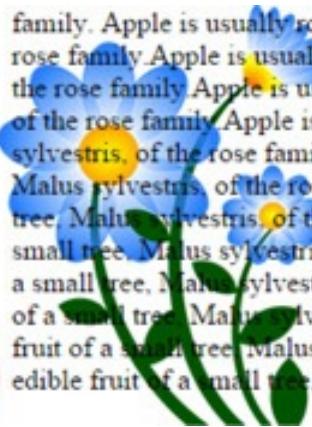
```



13

The *background-attachment* property lets you specify whether the background image should stay in one position or moves as the user scrolls the page. This property can have two values: *fixed* and *scroll*. When you use the fixed value, the image stays in the same position of the page. Listing 4.11 shows the use of this property. Figure 14 shows the result. Now, when you scroll the text the image should stay in the same position.

[Listing cs4.11: /cs4/background-image-3.html](#)



family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

You can also specify where in the browser window the background image should be placed. The ***background-position*** property allows you to specify the position. This property usually takes values in pair. The first value represents the horizontal value whereas the second represents the vertical value. It can take the following pairs of values:

- left top
- left center
- left bottom
- center top
- center center
- center bottom
- right top
- right center
- right bottom



The ***background-position*** Property

If you take one value, second will set to ***center***, by default.



The ***background-position*** Property

You can also use a pair of ***pixels*** or ***percentages***. The pair represents the distance from the ***top-left*** corner of the browser window. The top left corner is set to ***0% 0%***. If you set it to ***50% 50%***, the image will be centered horizontally and vertically.

Listing cs4.12 place the image at the ***center top*** position. Figure 15 shows the result.

[Listing cs4.12: /cs4/background-image-4.html](#)

```
<!DOCTYPE html>
<html>
<head>
<title> Background Image </title>
<style>
body {
background-image: url(images/flower.png);
background-repeat: no-repeat;
background-position: center top;
}
</style>
</head>
<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

</body></html>
```

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.



15

Using the Background Shorthand

The *background* property acts as a *shorthand* for all other background properties. The properties must be specified in the following order, however,

you can omit a property if you don't want to use it: *background-color*, *background-image*, *background-repeat*, *background-attachment*, and *background-position*.

Listing cs4.13 demonstrates the use of this property. Figure 16 shows the result.

Listing cs4.13: /cs4/background-image-5.html

```
<!DOCTYPE html>
<html>

<head>
<title> Background Image </title>
<style>
body {
background: #e7e7e7 url("images/flower.png") no-repeat fixed center;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

</body>

</html>
```

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.



16

Creating Gradients

You can create a gradient for a background using the *background-image* property. The gradients are created by creating a smooth transition between *two* or *more* specified colors. You can create two types of gradients: *Linear* gradients and *radial* gradients.

The *linear* gradient is most commonly used gradient. The gradient axis can go from left to right, top to bottom, or to any *angle* that you choose. If you don't declare the *angle*, the direction will be assumed *top to bottom*. For specifying colors, you can use *named colors*, *rgba*, or *hsla* values.

In Listing cs4.14, I have created a *linear gradient* from top to bottom using the *90* degrees angle. Figure 17 shows the result. If you change the angle to *180* degrees the output will look like something shown in Figure 18.

Listing cs4.14: /cs4/background-image-6.html

```
<!DOCTYPE html>
<html>

<head>
<title> Gradients </title>
<style>
p {
background-image: -moz-linear-gradient( 90deg, rgb(255, 109, 0) 0%, rgb(255, 182, 0)
26%, rgb(255, 255, 0) 50%, rgb(255, 110, 2) 100%);
```

```
background-image: -webkit-linear-gradient( 90deg, rgb(255, 109, 0) 0%, rgb(255, 182, 0) 26%, rgb(255, 255, 0) 50%, rgb(255, 110, 2) 100%);  
background-image: -ms-linear-gradient( 90deg, rgb(255, 109, 0) 0%, rgb(255, 182, 0) 26%, rgb(255, 255, 0) 50%, rgb(255, 110, 2) 100%);  
/*default syntax*/  
background-image: linear-gradient( 90deg, rgb(255, 109, 0) 0%, rgb(255, 182, 0) 26%,  
rgb(255, 255, 0) 50%, rgb(255, 110, 2) 100%);  
  
left: 81px;  
top: 150px;  
width: 365px;  
height: 187px;  
padding: 10px;  
}  
</style>  
</head>  
  
<body>  
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>  
  
</body>  
  
</html>
```

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Exercises

Exercise CS4-E1

Using the template shown in Listing cs4-e1, specify the following settings:

Apply *padding* of *5px* and *margin* of *10px* to the paragraph. Create a *3px* border around the element with *dotted* style and *#4557bc* color. Change the *border style* to *solid* and *color* to *red* when user hovers mouse over it.

Listing cs4-e1: /cs4/cs4-e1.html

```
<!DOCTYPE html>
<html>

<head>
<title> Border and Backgrounds </title>
<style>
</style>
</head>

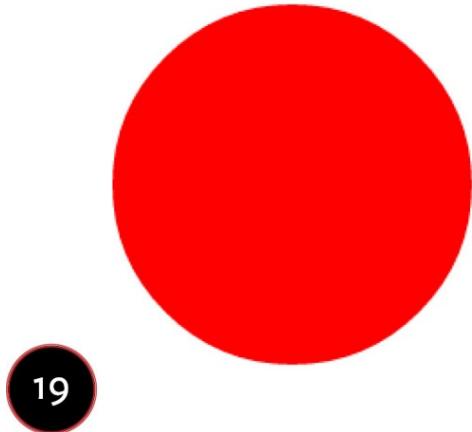
<body>
<p class="brdr">Apple is usually round, red or yellow, edible fruit of a small tree,  
Malus sylvestris, of the rose family.</p>

</body>
```

```
</html>
```

Exercise CS4-E2

Using the template shown in Listing cs4-e2 create a *circle* from a *200x200px* box (see Figure 19).



Listing cs4-e2: /cs4/cs4-e2.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title> Border and Backgrounds </title>
```

```
  <style>
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
  <div id="box">
```

```
  </div>
```

```
</body></html>
```

Summary

In this chapter, I have shown you the use of the properties that allow you to add *borders*, and *backgrounds* to the elements. You have seen how we can use the images to define borders for an element. I have also showed you how to create rounded corners using the *border* properties. In the end, I have showed you to create background using the gradients. In the next chapter, I will show you the use of the *color* properties.

Unit CS5: Working with Colors

In the previous chapters, I have used the *color*, *border-color*, and *background-color* properties in quite a few listings. In this chapter, you will see how to set *foreground color*, *background color*, and *opacity* for the elements.

Key Points and Concepts

- Setting foreground and background colors
- Specifying colors using color names and hex codes
- Specifying colors using rgb and hsl values
- Understanding various color terminologies
- Setting element's opacity

Setting the Foreground and Background Colors

The *color* property sets the *foreground* color for the element whereas the *background-color* property is used for the *background* color. If you don't specify a *background* color, the background is *transparent*. Listing cs5.1 shows the *color* and *background-color* properties in use.

[Listing cs5.1: /cs5/color.html](#)

```
<!DOCTYPE html>
<html>

<head>
<title> Applying Color </title>
<style>
p {
```

```

padding: 5px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
}

</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>

</body>

</html>

```

In Listing cs5.1, I have used two different styles for defining the color. Figure 1 shows the result. The background color of the paragraph is specified using the *rgba* property. This property allows you to specify a color just like the *rgb* property. However, with *rgba* you can add a fourth value which is known as the *alpha* value. The *alpha* value is a *number* between *0* to *1*. The value *1* represents *100%* opacity, a value of *0.5* represents *50%* opacity, and so on. I have also set the color of the text to *darkblue* using the *color* properties.

1

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

In CSS, you can specify a color using *RGB values*, *HSL values*, *HEX Codes*, and *color names*. Table cs5.1 summarizes these types and other terminology used.

Table cs5.1: The terminologies used for colors

Terminology	Description
RGB Values	These are values for red, green, and blue color components. These are expressed as numbers from 0 to 255. For example, rgb(255, 0, 0) represents red color.
HEX Codes	These values represent red, green, and blue values as hexadecimal code. These values are preceded by a pound sign (#). For example, #ff0000 represents red color.
Color Names	The colors are represented by the predefined names. For example, red can be used to represent red color.
Hue	It's another word for color. A color can also have saturation and brightness.
Saturation	Saturation is the intensity or purity of a hue. It represents amounts of gray in the color. At maximum saturation, there would be no gray.
Brightness	It is sometimes also referred to as value. It represents the amount of black present in the color.

Setting Element's Opacity

The *opacity* property allows you to specify the *opacity* of an element. It takes a number from *0* to *1* range. A value of *0.5* represents *50%* opacity, and a value of *0.2* represents *20%* value. Listing cs5.2 shows the use of this property.

Listing cs5.2: /cs5/color-1.html

```
<!DOCTYPE html>
<html>

<head>
<title> Applying Color </title>
<style>
.clr {
background-color: rgb(255, 0, 0);
position: absolute;
left: 28px;
top: 32px;
width: 134px;
height: 113px;
z-index: 1;
```

```
}

.clr1 {

background-color: rgb(255, 240, 0);

position: absolute;

left: 40px;

top: 60px;

width: 128px;

height: 87px;

z-index: 2;

opacity: 0.5;

}

</style>

</head>
```

```
<body>

<div class="clr">

<div class="clr1">

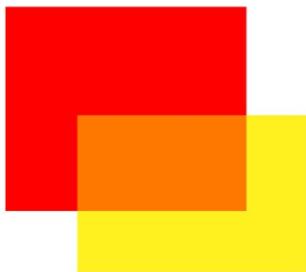
</div>

</div>

</body>

</html>
```

In Listing cs5.2, I have created two rectangles defined by the `clr` and `clr1` classes. The `clr1` class represents the *yellow* rectangle. I have set opacity of this rectangle to **50%** using the `opacity` property. Figure 2 shows the result.



2

Using the hsl and hsla Properties

These two properties are an alternate way to define colors. *HSL* stands for *hue*, *saturation*, and *lightness*. The *hue* value is represented as an angle from 0 to 360 degrees. The *saturation* and *lightness* are represented using *percentages*. For lightness, 0% is white whereas 100% is black. Listing cs5.3 and Figure 3 demonstrate the use of this property.



3

[Listing cs5.3: /cs5/color-2.html](#)

```
<!DOCTYPE html>
<html>

<head>
<title> Applying Color </title>
<style>
.clr {
background-color: hsl(0, 100%, 50%);
position: absolute;
left: 28px;
top: 32px;
width: 134px;
```

```
height: 113px;  
z-index: 1;  
}  
.clr1 {  
background-color: hsla(99, 93%, 31%, 0.77);  
position: absolute;  
left: 40px;  
top: 60px;  
width: 128px;  
height: 87px;  
z-index: 2;  
}  
</style>  
</head>
```

```
<body>  
<div class="clr">  
<div class="clr1">  
  
  
  
  
</div>  
</div>  
  
</body>  
  
</html>
```

Exercises

Exercise CS5-E1

Write all possible rules to set the *color* of the *paragraph* element to *red*.

Exercise CS5-E2

Write the rules to set the *color* of the *paragraph* element to *red* with *40%* translucency.

Exercise CS5-E3

Using the template shown in Listing cs5-e3, set the *opacity* of the *img* element to *50%*. Change it to *100%* when user hover mouse over the image.

Listing cs5-e3: /cs5/cs5-e3.html

```
<!DOCTYPE html>
<html>

<head>
<title> Applying Color </title>
<style>
</style>
</head>

<body>

</body>

</html>
```

Summary

In this chapter, I demonstrated the different ways for assigning colors to the elements. You learned setting colors by using *color names*, *hex codes*, *RGB*, and *HSL*. You also gained knowledge about various *terminologies* used for the color.

Unit CS6: Formatting Text

The *formatting* of text is very important to create a webpage that appeals. In this chapter, I will show the use of the properties that you can use to format text.

Key Points and Concepts

- Aligning and justifying text
- Specifying text direction
- Specifying letter and word spacing
- Working with word breaks
- Indenting text
- Decorating and transforming text
- Creating drop shadow
- Specifying typefaces
- Using the Google fonts

Aligning and Justifying the Text

The *text-align* and *text-justify* properties are used to align and justify text. Table cs6.1 summarizes these properties. Listing cs6.1 shows the use of the *text-align* property. The text in the paragraph is aligned to the *right* using the *right* value of the property. Figure 1 shows the result.

Table cs6.1: The properties to *align* and *justify* text

Property	Description
text-align	Controls the alignment for a block of text. It takes the following values: start, end, left, right, center, and justify.
text-justify	This property is used to justify the text. The text-justify property takes one of the following values: auto, none, inter-word, inter-ideograph, inter-cluster, distribute, and kashida.

Listing cs6.1: /cs6/text-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
text-align: right;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
</body></html>
```

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

1

To justify the text, you need to use *text-align* and *text-justify* properties, refer to Listing cs6.2. In this listing, the *text-justify* property is set to *inter-word* value which causes spacing to be *distributed between the words*. Figure 2 shows the result.

Listing cs6.2: /cs6/text-2.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
text-align: justify;
text-justify: inter-word;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
</body>
</html>
```

2

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Working with Whitespace

Whitespace such as multiple *spaces*, *line breaks*, and *tabs* is ignored in HTML. The whitespace is *collapsed* to a *single space character* and lines are wrapped.

However, you can control (*preserve*) the formatting of whitespace using the *white-space* property. Table cs6.2 shows the values corresponding to this property.

Table cs6.2: The *white-space* property values

Property	Description
normal	The default value. It collapses the whitespace and wraps the lines.
nowrap	The whitespace is collapsed but lines are not wrapped.
pre	The whitespace is preserved and wraps only on line breaks.
pre-line	The whitespace is collapsed to a single whitespace. When necessary, text will be wrapped on line breaks.
pre-wrap	The whitespace is preserved by the browser. The text will wrap on line breaks when necessary.

Listing cs6.3 shows the HTML code that I am using with the *white-space* property. Listing cs6.4 shows the *white-space* property in action. Figure 3 shows the effect of the *pre-line* value. Notice that text is *wrapped* and *line breaks* are *preserved*.

Apple is usually round, red or yellow, edible fruit of a small tree.

Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree.

Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree.

Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

3

Listing cs6.3:

<p>Apple is usually round, red or yellow, edible fruit of a small tree,

Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree,

Malus sylvestris, of the rose family.

Apple is usually round, red or yellow, edible fruit of a small tree,

Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

</p>

Listing cs6.4: /cs6/text-3.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title> Formatting Text </title>
```

```
  <style>
```

```
    p {
```

```
      padding: 10px;
```

```
      margin: 10px;
```

```
      border: 3px dotted #4557bc;
```

```
      background-color: rgba(183, 122, 7, 0.2);
```

```
      color: darkblue;
```

```
      white-space: pre-line;
```

```
    }
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
  <p>Apple is usually round, red or yellow, edible fruit of a small tree,
```

```
  Malus sylvestris, of the rose family.
```

```
  Apple is usually round, red or yellow, edible fruit of a small tree,
```

```
  Malus sylvestris, of the rose family.
```

```
  Apple is usually round, red or yellow, edible fruit of a small tree,
```

Malus sylvestris, of the rose family. Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

</p>

</body>

</html>

Specifying Text Direction

The *direction* property allows you to specify the *text direction/writing* direction. This property takes two values: *ltr* and *rtl*. Listing cs6.5 and Figure 4 shows use of this property.

Listing cs6.5: /cs6/text-4.html

```
<!DOCTYPE html>
<html>
<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
}
.cls1 {
direction: ltr;
}
.cls2 {
```

```

direction: rtl;
}

</style>
</head>

<body>

<p class="cls1">Left to right text: Apple is usually round, red or yellow, edible fruit of
a small tree, Malus sylvestris, of the rose family.

</p>

<p class="cls2">Right to left text: Apple is usually round, red or yellow, edible fruit of
a small tree, Malus sylvestris, of the rose family.

</p>

</body>

</html>

```

4

Left to right text: Apple is usually round, red or yellow, edible
fruit of a small tree, Malus sylvestris, of the rose family.

Right to left text: Apple is usually round, red or yellow, edible
fruit of a small tree, Malus sylvestris, of the rose family

Specifying Letter and Word Spacing

You can use the *letter-spacing*, *word-spacing*, and *line-height* properties to adjust space between the *letters*, *words*, and *lines*. Listing cs6.6 demonstrate the use of these properties.

Listing cs6.6: /cs6/text-5.html

```

<!DOCTYPE html>
<html>

```

```
<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-spacing: 5px;
letter-spacing: 2px;
line-height: 1.5em;
}
</style>
</head>

<body>
<p>Left to right text: Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.
</p>
</body>

</html>
```

The effect of these properties is shown in Figure 5.

5

Left to right text: Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Working with Word Breaks

The `word-break` property allows you to set the *line breaking rules*. This property supports three values: `normal`, `break-all`, and `keep-all`. The `normal` value breaks the words according to their usual roles. The `break-all` breaks the line between *any two* letters. The `keep-all` value prohibits the breaks between *pair of letters*. Figure 6.7 shows the use of this property.

Listing cs6.7: /cs6/text-6.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
float: left;
}
.cls1 {
width: 100px;
word-break: keep-all;
}
.cls2 {
width: 100px;
word-break: break-all;
}
</style>
</head>
```

```
<body>
```

```
<p class="cls1">Left to right text: Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
```

```
<p class="cls2">Left to right text: Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
```

```
</body>
```

```
</html>
```

There are two *p* elements in the above listing to which I have applied the *word-break* property. You can see the effect of the properties in Figure 6, the *break-all* value is breaking the line between any two characters.

6

Left to right text: Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Left to right text: Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

The *word-wrap* property allows to break a long word and wrap onto the next line. It takes two values: *normal* and *break-word*. The *break-word* value lets you break an *unbreakable* word. Listing cs6.8 shows the use of this value.

Listing cs6.8: /cs6/text-7.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```

<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 200px;
}
</style>
</head>

<body>
<p>Left to right text: Apple is usually round, red or yellow, edible fruit  

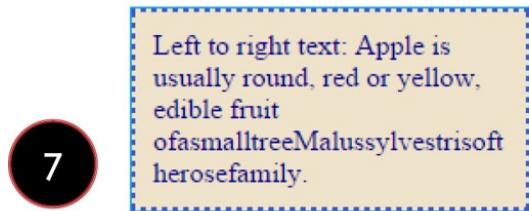
of a small tree Malus sylvestris of the rose family.</p>

</body>

</html>

```

The effect of *word-wrap* property is shown in Figure 7.



Indenting Text

The *text-indent* property lets you specify an *indentation* for the *first line* of block of text. You can specify indent as *length* or as a *percentage*. Listing

cs6.9 shows the use of this property.

Listing cs6.9: /cs6/text-8.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 200px;
text-indent: 10%;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
</body>

</html>
```

The *indent effect* is shown in Figure 8.

8

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Decorating and Transforming Text

The *text-decoration* and *text-transform* properties let you *decorate* and *transform* the text. Table cs6.3 summarizes the *text-decoration* and *text-transform* property values.

Table cs6.3: The text-decoration and text-transform properties values	
Property	Description
text-decoration	Allows you to decorate the text. It takes the following values: none underline overline line-through blink
text-transform	Allows you to apply transformations to a block of text. It accepts the following values: none capitalize uppercase lowercase

Listing cs6.10 shows the use of these values.

Listing cs6.10: /cs6/text-9.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
}
.test1 {
text-decoration: underline;
}
.test2 {
text-decoration: overline;
}
.test3 {
text-decoration: line-through;
}
.test4 {
text-transform: capitalize;
}
.test5 {
text-transform: uppercase;
}
.test6 {
text-transform: lowercase;
}
```

```

</style>
</head>

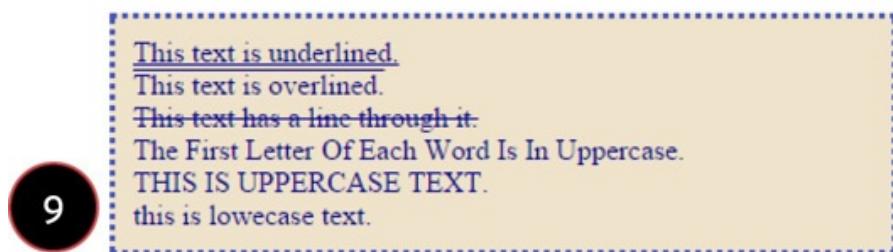
<body>
<p>
<span class="test1"> This text is underlined.</span>
</br>
<span class="test2"> This text is overlined.</span>
</br>
<span class="test3"> This text has a line through it.</span>
</br>
<span class="test4"> The first letter of each word is in uppercase.</span>
</br>
<span class="test5"> This is uppercase text.</span>
</br>
<span class="test6"> This is lowercase text.</span>
</br>
</p>

</body>

</html>

```

Figure 9 shows the effect of the *text-decoration* and *text-transform* properties.



9

In chapter 3, I showed how to create *drop-shadow* effect for a box. Similarly, you can create drop-shadow effect for a block of text. The *text-shadow*

property is used to create drop-shadow. It takes the following values: *h-shadow*, *v-shadow*, *blur-radius*, and *color*. The *h-shadow* and *v-shadow* values are required and control the position of the *horizontal* and *vertical* shadows, respectively. Listing cs6.11 shows the use of this property.



Color

The color can be specified before or after the offsets.

Listing cs6.11: /cs6/text-10.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 380px;
text-shadow: 2px 2px 4px #000000;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
```

```
</body>
```

```
</html>
```

Figure 10 shows the drop-shadow effect on a block of text.

10

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Working with Fonts

In this section, I will show you how to use *font* properties.

Selecting a Font

The *font-family* property allows you to select the font that you want to use in your design. This property can hold several font names. It works as a *fallback* system, if one font is not available, browser tries the next font. Use a *comma separated* font names with this property.



whitespace

If a font name contains whitespace such as *Times New Roman*, it must be *quoted*: "*Times New Roman*".

You can use a *family-name* [*times*, *courier*, *arial* etc] or *generic-family* name [*serif*, *sans-serif*, *cursive*, *fantasy*, and *monospace*]. Table cs6.4 shows the list of generic font family members.

Table cs6.4: The generic font family	
Family	Font
serif	Times
sans-serif	Helvetica
cursive	Zapf-Chancery
fantasy	Western

monospace courier

Listing cs6.12 shows the *font-family* property applies to a paragraph.

Listing cs6.12: /cs6/text-11.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 380px;
font-family: Helvetica, sans-serif;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
</body>

</html>
```

In Listing cs6.12, I have used *Helvetica* font for the *font-family* property. I have used generic *sans-serif* as *fallback* if the browser could not find *Helvetica* in the system. You can see the effect of this property in Figure 11.

11

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Setting the Font Size

The *font-size* property enables you to specify the size of the font. There are many ways to specify the font size. The most commons are *pixels*, *percentages*, and *ems*. The pixels are widely used with font size property as it gives you control to precisely how much space text takes. You can specify size in *percentages*. The default size of the text in a browser is *16px*. Therefore, if you want to use a pixel size of *12px*, you need to set the percentage to *75%*. A percentage of *200* would be equal to *32px*. You can also specify size in *ems*, an *em* is equivalent to the width of the letter *m*.

Listing cs6.13 shows the use of the *font-size* property.

Listing cs6.13: /cs6/text-12.html

```
<!DOCTYPE html>
<html>
<head>
<title> Formatting Text </title>
<style>
body {
  font-family: helvetica, sans-serif;
  font-size: 12px;
}
h1 {
  font-size: 200%
```

```

}

p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 380px;
font-size: 1.3em;
}

</style>
</head>

<body>
<h1>About Apple</h1>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
</body>

</html>

```

In Listing cs6.13, I have set font size for the `body` element to `12px`. The size for `h1` is `200%` which is equivalent to `24px` because `body` is the containing block for `h1`. The font size of the paragraph is set to `1.3em` which is equivalent to `18px`. Figure 12 shows the effect of this property.

About Apple

12

Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.

Setting Weight and Style for Font

The `font-weight` property allows you to create `bold` text. It takes two values: `normal` and `bold`. The `bold` value causes the text to appear in `bold`. Listing cs6.14 shows the use of this property. The effect is shown in Figure 13.

Listing cs6.14: /cs6/text-13.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 380px;
}
.test1 {
font-weight: bold;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, <span class="test1">Malus sylvestris</span>, of the rose family.</p>
</body>
```

```
</html>
```

13

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

The *font-style* property enables you to create *italic* or *oblique* text. It takes three values: *normal*, *italic*, and *oblique*. The *italic* value allows you to make text appear *italic*. The *oblique* value causes text to appear *oblique*. Listing cs6.14 shows the use of this property. The effect is shown in Figure 14.

Listing cs6.15: /cs6/text-14.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 380px;
}
.test1 {
font-style: italic;
}
</style>
</head>
```

```
<body>  
<p>Apple is usually round, red or yellow, edible fruit of a small tree, <span  
class="test1">Malus sylvestris</span>, of the rose family.</p>  
</body>  
</html>
```

14

Apple is usually round, red or yellow, edible fruit of a small tree, *Malus sylvestris*, of the rose family.

Using Webfonts

The `@font-face` rule enables you to specify a font, even if it is *not installed* in a system from where the user is accessing webpage. It eliminates the need to depend on the number of installed fonts that users might have installed on their systems.

The `font-family` property specifies the name of the font. This name can be used as a value for the `font-family` property in the rest of the document. The `src` property allows you to define the path of the font. Depending on your needs, you can specify multiple paths using this property. The `format` property specifies the format of the font.



Converting Fonts

You can convert a font to multiple formats using the service provided by *Font Squirrel*. You can access this service by navigating to the following link: <http://www.fontsquirrel.com/tools/webfont-generator>.

In Listing cs6.16, I have used the `chase.ttf` font that is inside the `/cs6/fonts` folder using the `src` property. I have first defined the name of the font family and then accessed it using `src`. Then, I have used the `font-family` name that I have defined in the `@font-face` rule code block for the `body` element to be able to use the font.

Font

Courtesy:

Anthony

James

(<https://www.behance.net/gallery/13895473/CHASE-Font>)

Listing cs6.16: /cs6/text-15.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
@font-face {
font-family: chase;
src: url("fonts/chase.ttf")
}
body {
font-family: chase, sans-serif;
font-size: 200%;
}
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 380px;
}
.test1 {
font-style: italic;
}
</style>
```

```
</head>
```

```
<body>
```

```
<p>Apple is usually round, red or yellow, edible fruit of a small tree, <span class="test1">Malus sylvestris</span>, of the rose family.</p>
```

```
</body>
```

```
</html>
```

Figure 15 shows the font applied to the block of text.

15

Apple is usually round, red or yellow, edible fruit
of a small tree. *Malus sylvestris*, of the rose
family.

Using Google Fonts

Google provides hundreds of *free, open-source* fonts optimized for the web that you can use in your projects. To use Google fonts, navigate to www.google.com/fonts and then browse through the fonts. Select font of your choice and then click *Quick-use* button, see Figure 16. From the page that is displayed, select the styles that you want to include and then add the font to page using *Standard*, *@import*, or *JavaScript* method, see Figure 17. Now, add font to the CSS using *font-family* property.

16

Normal 400
Grumpy wizards make toxic brew for the evil Queen and Jack.

Yanone Kaffeesatz, 4 Styles by Yanone



17

Standard @Import Javascript
3. Add this code to your website:

```
<link href="http://fonts.googleapis.com/css?family=Yanone+Kaffeesatz:400,700" rel="stylesheet" type="text/css">
```

In Listing cs6.17, I have incorporated the font using the *@import* directive and

then assigned this font to the *font-family* property.

Listing cs6.17: /cs6/text-16.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
@import url(http://fonts.googleapis.com/css?family=Yanone+Kaffeesatz:400,700);
body {
font-family: 'Yanone Kaffeesatz', sans-serif;
}
p {
padding: 10px;
margin: 10px;
border: 3px dotted #4557bc;
background-color: rgba(183, 122, 7, 0.2);
color: darkblue;
word-wrap: break-word;
width: 380px;
font-size: 120%;
}
</style>
</head>

<body>
<p>Apple is usually round, red or yellow, edible fruit of a small tree, Malus sylvestris, of the rose family.</p>
</body>
```

```
</html>
```

The effect of the *Yanone Kaffeesatz* font on the text in Figure 18.

18

Apple is usually round, red or yellow, edible fruit of a small tree,
Malus sylvestris, of the rose family.

Exercises

Exercise CS6-E1

Create four paragraphs with *Lorem Ipsum* text apply CSS styling to it [see Figure 19] using the following instructions:

1. Use the template shown in Listings cs6-e1.
2. Create four paragraphs with Lorem Ipsum text and assign the following *classes* in order: *left*, *right*, *center*, and *justify*.
3. Create a rule for the *p* element. Set its width to *450px*. Set border of the block with width *1px*, *dotted* style, and *blue* color. Finally set *padding* and *top* margin to *5px* and *10px*, respectively.
4. Add the following classes in the style block: *left*, *right*, *center*, and *justify*. Now, set the property to *align* the text to *left*, *right*, *center*, and *justify* in order using these classes.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.

19

Listing cs6-e1: /cs6/cs6-e1.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
</style>
</head>

<body>
</body>
```

</html>

The Lorem Ipsum text to be used:

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.

Exercise CS6-E2

Use the paragraph defined in Listing cs6-e2 and the write a CSS rule to preserve the *whitespace* [see Figure 20]. The lines should only break at newline characters in the source and at *
* elements.

Listing cs6-e2: /cs6/cs6-e2.html

-
<p>

 Lorem Ipsum is simply dummy text of the printing and typesetting industry.

 Lorem Ipsum has been the industry's standard dummy text ever since the 1500s,
 when an unknown
 printer took a galley of type and
 scrambled it to make a type specimen book.

</p>

 Lorem Ipsum is simply dummy text of the printing and typesetting industry.

 Lorem Ipsum has been the industry's standard dummy text ever since the 1500s,
 when an unknown
 printer took a galley of type and

20

 scrambled it to make a type specimen book.

Exercise CS6-E3

Using the code shown in Listing cs6-e3, create a *blue* colored *drop-shadow* with the *horizontal* and *vertical* offsets set to *0* and *-2px*, respectively [see Figure 21].

Listing cs6-e3: /cs6/cs6-e3.html

```
<!DOCTYPE html>
<html>

<head>
<title> Formatting Text </title>
<style>
</style>
</head>

<body>
<p>
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

```
</p>
</body></html>
```

21

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

Exercise CS6-E4

Navigate to the *Google fonts site* and then find the font *Jolly Lodger*. Then, use this font to style the text [Figure 22] contained in the paragraph in Listing cs6-e4. Import the font using the *Standard* method and set the font size to *2em*.

Listing cs6-e4: /cs6/cs6-e4.html

```
<!DOCTYPE html>
<html>
```

```
<head>
<title> Formatting Text </title>
<style>
</style>
</head>

<body>
<p>
    Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
</p>
</body>

</html>
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

22

Summary

In this chapter, I showed the use of various properties that affects text on your web document and allow you to style the text. You learned about *font* and *text* alignment related properties. You have also learned how to use *custom fonts* and use them effectively. In the end, you learned to use the *Google web fonts*.

Unit CS7: Styling Lists, Table, Forms, and Cursors

In this chapter, I will explain how to change appearance of the *form elements*, *format tables*, and specify *bullet point styles* for the lists.

Key Points and Concepts

- Styling lists
- Styling tables (adding borders and backgrounds)
- Styling form elements
- Styling Cursor

Styling Lists

Table cs7-1 summarizes the properties that are specific to styling lists.

Table cs7.1: The properties specific to style lists

Property	Description
list-style-type	This property allows you to control the shape or style of a bullet point [also referred to as marker].
list-style-image	It allows you to use an image as a marker.
list-style-position	It controls position of the marker. It takes two values: inside and outside.
list-style	Shorthand for specifying all list characteristics in one line.

You can use the values summarized in Table cs7.2, with *list-style-type*

property.

Table cs7.2: The values of the *list-style-type* property

Property	Description
none	No bullet is drawn.
box, check, circle diamond, disc, dash square	The specified shape is used as a marker.
decimal	Use decimal numbers [1 2 3].
decimal-leading-zero	Use decimal numbers with padded zeros [01 02 03].
lower-roman	Use lowercase roman numerals [i ii iii].
upper-roman	Use uppercase roman numerals [I II III].
lower-alpha	Use lowercase alpha characters [a b c].
upper-alpha	Use uppercase alpha characters [A B C].



list-style-type Property

There are many more values that you can use with this property. Visit the following link to know more about these values: <https://developer.mozilla.org/en-US/docs/Web/CSS/list-style-type>.

Listing cs7.1 shows the use of these values.

Listing cs7.1: /cs7/lists-1.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title> Lists </title>
```

```
<style>
```

```
ol {  
list-style-type: lower-roman  
}  
  
ol#li1 {  
list-style-type: upper-alpha;  
}  
  
ul {  
list-style-type: square;  
}  
  
</style>  
</head>
```

```
<body>  
<h2> List-1 </h2>  
<ol>  
<li> Item 1</li>  
<li> Item 2</li>  
<li> Item 3</li>  
<li> Item 4</li>  
<li> Item 5</li>  
</ol>  
<h2> List-2 </h2>  
<ul>  
<li> Item 1</li>  
<li> Item 2</li>  
<li> Item 3</li>  
<li> Item 4</li>  
<li> Item 5</li>  
</ul>  
<h2> List-3 </h2>  
<ol id="li1">
```

```
<li> Item 1</li>
<li> Item 2</li>
<li> Item 3</li>
<li> Item 4</li>
<li> Item 5</li>
</ul>
```

```
</body>
```

```
</html>
```

Listing cs7.2 shows the use of the *list-style-image* property.

Listing cs7.2: /cs7/lists-2.html

```
<!DOCTYPE html>
<html>

<head>
<title> Lists </title>
<style>
ul {
list-style-image: url("images/bullet.png");
}
li {
margin: 10px 0px 0px 0px;
}
</style>
</head>

<body>
```

```
<h2> List-1 </h2>
<ul>
<li> Item 1</li>
<li> Item 2</li>
<li> Item 3</li>
<li> Item 4</li>
<li> Item 5</li>
</ul>
</body>

</html>
```

The effect of this property is shown in Figure 1. Notice in Listing cs7.2 that I have used *margin* property to add *vertical space* between the list items.

List-1

- Item 1
- Item 2
- Item 3
- Item 4
- Item 5

1

You can position the marker using the *list-style-position* property. It controls whether marker should appear on the *inside* or the *outside* of the box. This property takes two values: *inside* and *outside*. The *outside* value causes marker to sit to the left of the block of text whereas the *inside* value causes the marker to sit inside the block of text.

Listing cs7.3 shows the use of this property. Figure 2 shows the result.

Listing cs7.3: /cs7/lists-3.html

```
<!DOCTYPE html>
```

```
<html>

<head>
<title> Lists </title>
<style>
ul {
width: 50px;
}
.one {
list-style-position: inside;
list-style-image: url(images/bullet.png);
}
.two {
list-style-image: url(images/bullet.png);
list-style-position: outside;
}
</style>
</head>
```

```
<body>
<h2> List-1 </h2>
<ul class="one">
<li> Item 1</li>
<li> Item 2</li>
<li> Item 3</li>
<li> Item 4</li>
<li> Item 5</li>
</ul>
<h2> List-2 </h2>
<ul class="two">
<li> Item 1</li>
```

```
<li> Item 2</li>
<li> Item 3</li>
<li> Item 4</li>
<li> Item 5</li>
</ul>
</body>

</html>
```

List-1

- Item 1
- Item 2
- Item 3
- Item 4
- Item 5

List-2

- Item 1
- Item 2
- Item 3
- Item 4
- Item 5

2

You can also use the *list* shorthand to style lists using one declaration. For example, *list-style: inside circle;*:

Styling Tables

There are many properties that you can use to style the tables. In this section, I will talk about such properties.

Working with Empty Cells

The *empty-cells* property allows you to control the behavior of the empty cells. It takes three values: *show*, *hide*, and *inherit*. The *show* property shows border around the cells whereas the *hide* property hides the border. In Listing cs7.3, I have left two cells of the table empty. If you run this listing in the browser, you will get the result something like shown in Figure 3.

[Listing cs7.3: /cs7/empty-cells.html](#)

```
<!DOCTYPE html>
<html>

<head>
<title> Empty Cells </title>
<style>
body,
h1 {
font-family: verdana, san-sarif;
font-size: 90%;
}
h1 {
width: 168px;
padding: 10px;
background-color: lightgreen;
}
th,
td {
padding: 2px;
}
td {
border: 1px solid #000;
}
</style>
</head>

<body>
<h1> [ January Sales Data ] </h1>
<table id="sales-table-1">
```

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000

```

<td>Tim</td>
<td>99</td>
<td>45</td>
<td>6000</td>
</tr>
<tr>
<td>Graham</td>
<td></td>
<td></td>
<td>4500</td>
</tr>
</table>

```

</body></html>

[January Sales Data]

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham			4500

3

Notice that browser has drawn border around the empty cells that is not looking good. Now, lets hide the empty cells. Listing cs7.4 shows the code. Figure 4 shows the result.

Listing cs7.4: /cs7/empty-cells1.html

```

<!DOCTYPE html>
<html>

<head>

```

```
<title> Empty Cells </title>
<style>
body,
h1 {
font-family: verdana, san-sarif;
font-size: 90%;
}
h1 {
width: 168px;
padding: 10px;
background-color: lightgreen;
}
th,
td {
padding: 2px;
}
td {
border: 1px solid #000;
}
table {
empty-cells: hide;
}
</style>
</head>
<body>
```

```
<h1> [ January Sales Data ] </h1>
<table id="sales-table-1">
<tr>
<th>Salesman</th>
<th>Appointments</th>
```

Sales	Expanses
Phillip	110
57	2500
Brent	78
48	2000
Bill	100
75	3700
Fred	95
56	4000
Tim	99
45	

```

<td>6000</td>
</tr>
<tr>
<td>Graham</td>
<td></td>
<td></td>
<td>4500</td>
</tr>
</table>

```

</body>

</html>

[January Sales Data]

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham			4500

4

Controlling the Distance between the Adjacent Cells

The *border-spacing* property lets you control the distance between the adjacent cells. Notice in Figure 4, browser adds little bit space between the adjacent cells. The *border-spacing* property allows you to control that distance. You can also *collapse* the *adjacent borders* using the *border-collapse* property. It takes two values: *collapse* and *separate*. The *collapse* value collapses the borders to a *single* border. The *separate* value *detaches* borders from each other. In this case, the *border-spacing* and *empty-cells* properties will be considered.

Listing cs7.5 shows the use of these properties. Here, I have created two *class selectors* for the two tables. First table uses the *border-spacing* property. The borders of the second table are collapsed using the *border-collapse* property.

Figure 5 shows the result of this property.

Listing cs7.5: /cs7/border-distance.html

```
<!DOCTYPE html>
<html>

<head>
<title> Table Properties </title>
<style>
body,
h1 {
font-family: verdana, san-sarif;
font-size: 90%;
}
h1 {
width: 168px;
padding: 10px;
background-color: lightgreen;
}
th,
td {
padding: 2px;
}
td,
th {
border: 1px solid #000;
}
table.one {
border-spacing: 10px 5px;
}
```

```
table.two {  
border-collapse: collapse;  
}  
</style>  
</head>
```

```
<body>
```

```
<h1> [ January Sales Data ] </h1>
```

```
<table class="one">  
<tr>  
<th>Salesman</th>  
<th>Appointments</th>  
<th>Sales</th>  
<th>Expenses</th>  
</tr>  
<tr>  
<td>Phillip</td>  
<td>110</td>  
<td>57</td>  
<td>2500</td>  
</tr>  
<tr>  
<td>Brent</td>  
<td>78</td>  
<td>48</td>  
<td>2000</td>  
</tr>  
<tr>  
<td>Bill</td>  
<td>100</td>
```

```
<td>75</td>
<td>3700</td>
</tr>
<tr>
<td>Fred</td>
<td>95</td>
<td>56</td>
<td>4000</td>
</tr>
<tr>
<td>Tim</td>
<td>99</td>
<td>45</td>
<td>6000</td>
</tr>
<tr>
<td>Graham</td>
<td>50</td>
<td>30</td>
<td>4500</td>
</tr>
</table>
```

```
<table class="two">
<tr>
<th>Salesman</th>
<th>Appointments</th>
<th>Sales</th>
<th>Expenses</th>
</tr>
<tr>
```

Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham			

```
<td>50</td>  
<td>30</td>  
<td>4500</td>  
</tr>  
</table>
```

```
</body>
```

```
</html>
```

[January Sales Data]

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham	50	30	4500

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham	50	30	4500

5

Controlling the Position of the Caption

By default, the *caption* is shown at the *top* of the table. If you want to change this position, you can use the *caption-side* property. It takes two values: *top* and *bottom*. The *top* value is the default value. Listing cs7.6 shows the use of this property. Figure 6 shows the result.

[Listing cs7.6: /cs7/caption.html](#)

```
<!DOCTYPE html>  
<html>
```

```
<head>
```

```
<title> Table Properties </title>
```

```
<style>
```

```
body {
```

```
font-family: verdana, san-sarif;
```

```
font-size: 90%;
```

```
}
```

```
h1 {
```

```
width: 168px;
```

```
padding: 10px;
```

```
background-color: lightgreen;
```

```
}
```

```
th,
```

```
td {
```

```
padding: 2px;
```

```
}
```

```
td,
```

```
th {
```

```
border: 1px solid #000;
```

```
}
```

```
table {
```

```
border-collapse: collapse;
```

```
caption-side: bottom;
```

```
}
```

```
caption {
```

```
margin: 10px;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
<table class="one">
<caption> [ January Sales Data ]</caption>
<tr>
<th>Salesman</th>
<th>Appointments</th>
<th>Sales</th>
<th>Expenses</th>
</tr>
<tr>
<td>Phillip</td>
<td>110</td>
<td>57</td>
<td>2500</td>
</tr>
<tr>
<td>Brent</td>
<td>78</td>
<td>48</td>
<td>2000</td>
</tr>
<tr>
<td>Bill</td>
<td>100</td>
<td>75</td>
<td>3700</td>
</tr>
<tr>
<td>Fred</td>
<td>95</td>
<td>56</td>
```

```

<td>4000</td>
</tr>
<tr>
<td>Tim</td>
<td>99</td>
<td>45</td>
<td>6000</td>
</tr>
<tr>
<td>Graham</td>
<td>50</td>
<td>30</td>
<td>4500</td>
</tr>
</table>

</body>

```

</html>

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham	50	30	4500

6

[January Sales Data]

Working with the Layout of the Table

By default, browser sets the width of the table automatically. You can disable this behavior by using the *table-layout* property and using the value fixed for

it. When you use this value, the size of the table is set by the width values for the table and individual columns. In Listing cs7.7, I have set *width* to **100%** that causes table to occupy the available space. I have also set the layout to **fixed**.

Listing cs7.7: /cs7/fixed-layout.html

```
<!DOCTYPE html>
<html>

<head>
<title> Table Properties </title>
<style>
body {
font-family: verdana, san-sarif;
font-size: 90%;
}
h1 {
width: 168px;
padding: 10px;
background-color: lightgreen;
}
th,
td {
padding: 2px;
}
td,
th {
border: 1px solid #000;
}
table {
```

```
border-collapse: collapse;  
caption-side: bottom;  
table-layout: fixed;  
width: 100%;  
}  
caption {  
margin: 10px;  
}  
</style>  
</head>  
  
<body>  
<table class="one">  
<caption> [ January Sales Data ]</caption>  
<tr>  
<th>Salesman</th>  
<th>Appointments</th>  
<th>Sales</th>  
<th>Expanses</th>  
</tr>  
<tr>  
<td>Phillip</td>  
<td>110</td>  
<td>57</td>  
<td>2500</td>  
</tr>  
<tr>  
<td>Brent</td>  
<td>78</td>  
<td>48</td>  
<td>2000</td>
```

```
</tr>
<tr>
<td>Bill</td>
<td>100</td>
<td>75</td>
<td>3700</td>
</tr>
<tr>
<td>Fred</td>
<td>95</td>
<td>56</td>
<td>4000</td>
</tr>
<tr>
<td>Tim</td>
<td>99</td>
<td>45</td>
<td>6000</td>
</tr>
<tr>
<td>Graham</td>
<td>50</td>
<td>30</td>
<td>4500</td>
</tr>
</table>
```

```
</body>
```

```
</html>
```

Styling the Cursor

The `cursor` property allows you to specify the type of cursor to be displayed when pointing on an element. It takes many values such as `crosshair`, `help`, `progress`, `wait`, and so forth. Listing cs7.8 shows the use of this property. To show the cursor style, hover the cursor on the paragraphs.

Listing cs7.8: /cs7/cursor.html

```
<!DOCTYPE html>
<html>

<head>
<title> Cursor </title>
<style>
p.crosshair {
    cursor: crosshair;
}
p.progress {
    cursor: progress;
}
p.help {
    cursor: help;
}
</style>
</head>

<body>
<p class="crosshair">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
```

```
<p class="progress">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>
<p class="help">Apple is the round fruit of a tree of the rose family, which typically has thin green or red skin and crisp flesh.</p>

</body>

</html>
```

Exercises

Exercise CS7-E1

Using the template shown in Listing cs7-e1. Position the marker inside in *List1* and outside in *List2*. Generate the output shown in Figure 7.

This is list1

- Item 1
- Item 2
- Item 3
- Item 4
- Item 5

This is list2

- Item 1
- Item 2
- Item 3
- Item 4
- Item 5

7

Listing cs7-e1: /cs7/ex1.html

```
<!DOCTYPE html>
<html>

<head>
<title> Lists </title>
<style>
</style>
</head>
```

```

<body>
<ul id="id1">
This is list1
<li> Item 1</li>
<li> Item 2</li>
<li> Item 3</li>
<li> Item 4</li>
<li> Item 5</li>
</ul>

<ul id="id2">
This is list2
<li> Item 1</li>
<li> Item 2</li>
<li> Item 3</li>
<li> Item 4</li>
<li> Item 5</li>
</ul>
</body></html>

```

Exercise CS7-E2

Create a table using the values shown in Figure 8 [You can also use the Listing cs7-e2]. Then, format the table, see Figure 8. Below are some hints:

1. Width of the table is *600 px*.
2. *Letter spacing* is set to *0.5em* and text *transform* is set to *uppercase* for the *caption* element. It also uses *10px* margin.
3. A *padding* of *2px* is used for the *th* and *td* elements.
4. Use colors as per your choice.
5. Change *background color* of the *cell* to *#895050* when you hover mouse

over a cell.

6. Align text in the *Expenses* column to the right.

[J A N U A R Y S A L E S D A T A]

Salesman	Appointments	Sales	Expenses
Phillip	110	57	2500
Brent	78	48	2000
Bill	100	75	3700
Fred	95	56	4000
Tim	99	45	6000
Graham	32	68	4500

8

Listing Cs7-e2: /cs7/ex2.html

```
<!DOCTYPE html>
<html>

<head>
<title>January Sales Data</title>
<style>
body {
font-family: verdana, san-sarif;
font-size: 90%;
}
</style>
</head>

<body>
<table>
<caption> [ January Sales Data ] </caption>
<tr>
<th>Salesman</th>
```

<th>Appointments</th>

<th>Sales</th>

<th>Expanses</th>

</tr>

<tr>

<td>Phillip</td>

<td>110</td>

<td>57</td>

<td>2500</td>

</tr>

<tr>

<td>Brent</td>

<td>78</td>

<td>48</td>

<td>2000</td>

</tr>

<tr>

<td>Bill</td>

<td>100</td>

<td>75</td>

<td>3700</td>

</tr>

<tr>

<td>Fred</td>

<td>95</td>

<td>56</td>

<td>4000</td>

</tr>

<tr>

<td>Tim</td>

<td>99</td>

```
<td>45</td>
<td>6000</td>
</tr>
<tr>
<td>Graham</td>
<td>32</td>
<td>68</td>
<td>4500</td>
</tr>
</table>
```

```
</body>
```

```
</html>
```

Summary

In this chapter, I have described the properties that you can use to style lists. You have also seen the use of the properties that affects the appearance of the table. I also showed the use of cursor property to style the appearance of the cursor.

Unit CS8: Creating Layout

In this lesson, I will show you how to control the position of the element on a page and how to create different types of *layouts* using CSS including the *multicolumn* layout. I will also show you how to adjust the *position* of the elements using the *offset* properties.

Key Points and Concepts

- Normal positioning
- Relative positioning
- Absolute positioning
- Fixed positioning
- Drawing order of the elements
- Multicolumn layout

Positioning Content

You can *position content* of on a page using the *positioning* properties. Table cs8.1 summarizes the properties that you can use. Table cs8.2 shows the values that *position* property takes.

Table cs8.1: The *position* properties

Property	Description
position	Allows to set the positioning method.
left	
right	
top	
bottom	Sets the offset values for positioning elements.

z-index | Set the ordering of the overlapping elements.

Table cs8.2: The *position* property values

Property	Description
static	The default value. The element is laid out as normal.
relative	You can use this property to position element relative to its normal position.
absolute	The element is positioned relative to its ancestor.
fixed	The element is positioned relative to the browser window.

Normal Positioning

This is the default way in which browser places one block top of another block. Since it is the *default behavior* therefore you don't need a *CSS* property. However, the *syntax* for declaring *normal* flow is as follows:

```
position: static;
```

Listing cs8.1 shows the use of the static value.

Listing cs8.1: /cs8/normal-flow.html

```
<!DOCTYPE html>
<html>
<head>
<title> Positioning Content </title>
<style>
body {
width: 750px;
font-family: helvetica, verdana, sans-serif;
}
```

```
h1 {  
background-color: #a3a3a3;  
padding: 5px;  
}  
p {  
width: 500px;  
}  
</style>
```

```
</head>
```

```
<body>
```

```
<h1>What is Lorem Ipsum</h1>
```

```
<p class="class1">
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

```
</p>
```

```
<p class="class2">
```

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

```
</p>
```

```
<p class="class3">
```

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

```
</p>
```

```
</body>
```

```
</html>
```

Figure 1 shows the *normal* flow. Notice that paragraphs are appearing below heading in order. The *width* of the paragraph elements is *500px*. I have not provided *width* for the *h1* element that causes it to stretch to the full width [*750px*] specified using the *body* rule.

What is Lorem Ipsum

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

1

Relative Positioning

The *relative* value of the *position* property allows you to *offset* the element relative to its *normal* position. You need to use the *offset* properties [*top*, *bottom*, *left*, or *right*] to place element relative to its *normal* position. You can use *pixels*, *percentages*, or *ems* to assign values.

Listing cs8.2 shows the use of this property.

Listing cs8.2: /cs8/relative.html

```
<!DOCTYPE html>
<html>
```

```
<head>
<title> Positioning Content </title>
<style>
body {
width: 750px;
font-family: helveticaA, verdana, sans-serif;
}
h1 {
background-color: #a3a3a3;
padding: 5px;
}
p {
width: 500px;
}
.class2 {
position: relative;
top: 15px;
left: 150px;
margin-bottom: 50px;
}
</style>
</head>
```

```
<body>
<h1>What is Lorem Ipsum</h1>
<p class="class1">
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop
```

publishing software like Aldus PageMaker including versions of Lorem Ipsum.

```
</p>
```

```
<p class="class2">
```

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

```
</p>
```

```
<p class="class3">
```

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

```
</p>
```

```
</body>
```

```
</html>
```

Figure 2 shows the result of the Listing cs8.2. Notice the change in the *position* of the second paragraph. It moved *150px* from the *left* edge and *15px* from the *top* edge from the position where it would have been in the normal flow.

Absolute Positioning

When you use the *absolute* value, the element will be taken out of the flow and it no longer affects the position of the other elements on the page. The *offset* properties let you place the *box* relative to the browser. In the Listing cs8.3, the second paragraph is positioned *15px* away from the *top* edge and *150px* from the *left* edge of the browser. Figure 3 shows the result. I have set a *background color* for the second paragraph so that you can clearly see its position on the page.

Listing cs8.3: /cs8/absolute.html

```
<!DOCTYPE html>
<html>

<head>
<title> Positioning Content </title>
<style>
body {
width: 750px;
font-family: helveticaA, verdana, sans-serif;
}
h1 {
background-color: #a3a3a3;
padding: 5px;
}
p {
width: 500px;
}
.class2 {
position: absolute;
background-color: antiquewhite;
top: 15px;
left: 150px;
}
</style>
</head>

<body>
<h1>What is Lorem Ipsum</h1>
<p class="class1">
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

</p>

<p class="class2">

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

</p>

<p class="class3">

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

</p>

</body>

</html>

What is Lorem Ipsum

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

2

What is L

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

3

Fixed Positioning

The *fixed* value of the *position* property can be used to *position* element *relative* to the browser. The position of the elements is fixed therefore when a user scrolls the page, the elements stays in its place. The placement of the element is controlled by the *offset* properties. Listing cs8.4 shows use of this value.

[Listing cs8.4: /cs8/fixed.html](#)

```
<!DOCTYPE html>
<html>

<head>
<title> Positioning Content </title>
<style>
body {
width: 750px;
font-family: helveticaA, verdana, sans-serif;
}
h1 {
background-color: #a3a3a3;
padding: 5px;
position: fixed;
margin: 0;
top: 0px;
left: 400px;
}
p {
width: 500px;
}
</style>
</head>

<body>
<h1>What is Lorem Ipsum</h1>
<p class="class1">
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting,
```

remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

</p>

<p class="class2">

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

</p>

<p class="class3">

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

</p>

</body>

</html>

Notice in Figure 4, the heading stays in its place when page is *scrolled* and the paragraph content is obscured by the heading box. The paragraphs are in the normal flow and therefore ignore the heading element.

more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

What is Lorem Ipsum

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

The **`z-index`** property allows you to set the *front-to-back* order of the element. It takes a number as value. You can also use the *negative* numbers. The smaller the value, further to the back the element will be placed. Listing cs8.5 shows use of this property.

Listing cs8.5: /cs8/z-order.html

```
<!DOCTYPE html>
<html>

<head>
<title> Overlapping Elements </title>
<style>
.clr {
background-color: rgb(255, 0, 0);
position: absolute;
left: 28px;
top: 32px;
width: 134px;
height: 113px;
z-index: 1;
}
.clr1 {
background-color: rgb(255, 240, 0);
position: absolute;
left: 40px;
top: 60px;
width: 128px;
height: 87px;
z-index: 2;
}
```

```

</style>
</head>

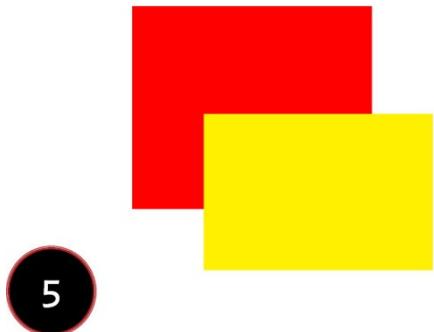
<body>
<div class="clr">
<div class="clr1">

</div>
</div>

</body>
</html>

```

Notice in Figure 5, the *red* square is placed behind the *yellow* square because the *red* square's *z-index* value is *smaller* than that of the *yellow* square.



Creating Multicolumn Layout

The CSS *multi-column* layout allows you to define multiple column of text as you see in newspapers. Table cs8.3 summarizes the multicolumn properties.

Table cs8.3: The <i>multicolumn</i> properties	
Property	Description
column-count	Specifies the number of columns.
column-gap	Controls the distance between columns.

column-rule-color	Sets color of the rule between columns.
column-rule-style	Sets the style of the rules.
column-rule-width	Sets the width of the rules.
column-rule	Shorthand property of setting column rules.
column-width	Specifies width of the columns.

Listing cs8.6 creates a three column layout. Figure 6 shows the result.

Listing cs8.6: /cs8/three-columns.html

```
<!DOCTYPE html>
<html>

<head>
<title> Creating Layout </title>
<style>
body {
width: 750px;
font-family: helvetica, verdana, sans-serif;
}
.news {
-webkit-column-count: 3;
/* Chrome, Safari, Opera */
-moz-column-count: 3;
/* Firefox */
-webkit-column-rule: 1px solid blue;
-webkit-column-gap: 18px;
/* Chrome, Safari, Opera */
-moz-column-rule: 1px solid blue;
-moz-column-gap: 18px;
```

```
/* Firefox */  
column-count: 3;  
column-gap: 18px;  
column-rule: 1px solid blue;  
}  
</style>  
</head>  
  
<body>  
<h1>What is Lorem Ipsum</h1>  
<p class="news">  
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.  
</p>  
</body>  
  
</html>
```

What is Lorem Ipsum

6

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the

literature, discovered the undoubtable source. Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the

Exercises

Exercise CS8-E1

Using the Listing cs8-e.1, create a *three column* newspaper style layout. Set the *column gap* of *40* pixels and set *rule color* to *red* and make it *2px* wide.

[Listing cs8.1: /cs8/ex1.html](#)

```
<!DOCTYPE html>
<html>

<head>
<title> Creating Layout </title>
<style>
</style>
</head>

<body>
<h1>What is Lorem Ipsum</h1>
<p class="news">
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem
Ipsum has been the industry's standard dummy text ever since the 1500s, when an
```

unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum. Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

</p>

</body>

</html>

Exercise CS8-E2

Create three squares, refer Figure 7, and then use the `z-index` property to decide their `front-to-back` order. The size of the boxes is `250x250px`. Following are the values of the `top` and `left` offset properties.

1. Red Square: 39px, 43px
2. Green Square: 102px, 117px
3. Blue Square: 164px, 190px



7

Summary

In this chapter, you explored different ways to position elements using *normal*, *relative*, and *absolute* positioning. You have also learned the use of the *offset* properties to place the content relative to the containing box and browser.

Unit CS9: Animation,

Transformations, and

Transitions

In this lesson, I will talk about applying animation to the elements of a page using CSS. You can use *animation* to add *interactivity* to your webpage. You can also apply these animations using *JavaScript* libraries but CSS has performance advantage over the *JavaScript*.

Key Points and Concepts

- Creating Transitions
- Creating Animations
- Using Transforms

Creating Transitions

When you apply a CSS property to an element, browser immediately applies it to the element. However, you can create a transition to gradually apply that effect to the element. Table cs9.1 shows the properties that you can use to control the transitions.

Table cs9.1: The *transition* properties

Property	Description
transition-delay	Controls the delay after which the transition will start.
transition-duration	Controls the time of the transition.
transition-property	Specifies property that the transition applies to. You can specify multiple properties.
transition-timing-function	Controls the speed curve of the transition effect. It allows you to change speed over time.
transition	This is a shorthand property for declaring all four properties in one sentence.



transition-duration Property

Always apply the *transition-duration* property because the *default* value of this property is *zero*. As a result, no *transition* is applied.

In Listing cs9.1, I have set a *100x100px* box and set its *background color* to *red*. The properties that I want to apply transition to are *width* and *background-color*. When user hover over the box, I want to change width of the box to *300px* and also want to change its *background color* to *blue*. I have set *transition delay* to *2* seconds and I want transition to last *5* seconds that I am controlling through *transition-duration* property.

Listing cs9.1: /cs9/transition-delay.html

```
<!DOCTYPE html>
<html>

<head>
<title> Animation </title>
<style>
div {
width: 100px;
height: 100px;
background-color: red;
transition-property: width, background-color;
transition-duration: 5s;
transition-delay: 2s;
transition-timing-function: linear;
}
div:hover {
width: 300px;
background-color: blue;
```

```

    }
</style>
</head>

<body>
<div></div>
</body>

</html>

```

Render the listing in the browser and hover the cursor on the *red* square, the transition should start after *2* seconds. Notice in Listing cs9.1, I have set *speed change curve* to *linear*. It sets the transition effect with the same speed from stat to end. Table cs9.2 describes various values.

Table cs9.2: The *transition-timing-function* values

Value	Description
ease	This is the default value. It causes transition to start slow, then fast, and then again end slowly.
linear	Same speed from start to end.
ease-in	Starts slow.
ease-out	Ends slow.
ease-in-out	Slow start and end.
cubic-bezier(n, n, n, n)	You can use it to define your own values.

Creating Animations

Animations are like *transitions* however they offer more control and more flexibility. Table cs9.3 summarizes the *animation* properties.

Table cs9.3: The *animation* properties

Property	Description
animation-name	Sets the name of the keyframe you want to bind to a selector.
animation-duration	Sets the total time of the animation.
animation-timing-function	Controls the speed curve of the animation.
animation-delay	Controls the time before the animation start.
animation-iteration-count	Controls how many times an animation is played.
animation-direction	Controls whether the animation should be played backward on alternate cycles.
animation-fill-mode	Controls the behavior when animation is not playing. For example, when it is finished or when it has a delay).
animation-play-state	Controls whether animation is playing or stopped.

@keyframes Rule

Before we work on the *animation* properties, let's first understand the *@keyframes* rule. It allows you to gradually switch from one set of CSS styles to another set of CSS styles. The style change happens in percentage where **0%** is the start of the animation and **100%** is the end of the animation. You can also use the *from* and *to* keywords where from represents **0%** and to represents **100%**.

The *@keyframes* rule comprised of three parts: a name of our choice [*moveBox* in Listing cs9.2], stages [0 to 100%], and CSS styles.

Listing cs9.2 shows the use of the *animation* properties. When you run the Listing cs9.2 in the browser, the *red* box animates *infinitely* because I have set the *animation-iteration-count* property to *infinite*. The positioning of the box is controlled by the relative value of the *position* property.

[Listing cs9.2: /cs9/animation-1.html](#)

<!DOCTYPE html>

```
<html>

<head>
<title> Animation </title>
<style>
div {
width: 100px;
height: 100px;
background-color: red;
position: relative;
-webkit-animation-duration: 5s;
-webkit-animation-iteration-count: infinite;
-webkit-animation-timing-function: ease-in-out;
-webkit-animation-name: moveBox;
}
@-webkit-keyframes moveBox {
0% {
top: 0px;
}
25% {
top: 200px;
}
75% {
top: 50px;
}
100% {
top: 100px;
}
}
</style>
</head>
```

```
<body>
<div></div>
</body>
</html>
```

Multiple Animations



You can add multiple animations to an element using a comma separator. For example:

```
.div{
    animation: boxFade 3s 1s infinite linear alternate,
               boxRotate 3s 1s infinite linear alternate;
}
@keyframes boxFade {
    to {
        opacity: 0;
    }
}
@keyframes boxRotate {
    to {
        transform: rotate(360deg);
    }
}
```

You can also change as many styles as you want in an animation. Listing cs9.3 shows an example. Run the listing and notice that the *background color* and the *width* are animated.

Listing cs9.3: /cs9/animation-2.html

```
<!DOCTYPE html>
<html>

<head>
<title> Animation </title>
<style>
div {
    width: 100px;
    background-color: red;
    height: 50px;
}
```

```
height: 100px;  
background-color: red;  
position: relative;  
-webkit-animation-duration: 5s;  
-webkit-animation-iteration-count: infinite;  
-webkit-animation-timing-function: ease-in-out;  
-webkit-animation-name: moveBox;  
}  
  
@-webkit-keyframes moveBox {  
0% {  
top: 0px;  
background-color: aqua;  
width: 50px;  
}  
100% {  
top: 100px;  
background-color: yellow;  
width: 100px;  
}  
}  
}  
</style>  
</head>  
  
<body>  
<div></div>  
</body>  
  
</html>
```

Listing cs9.4 shows the Listing cs9.3 with the *form* and *to* keywords.

Listing cs9.4: /cs9/animation-3.html

```
<!DOCTYPE html>
<html>

<head>
<title> Animation </title>
<style>
div {
width: 100px;
height: 100px;
background-color: red;
position: relative;
-webkit-animation-duration: 5s;
-webkit-animation-iteration-count: infinite;
-webkit-animation-timing-function: ease-in-out;
-webkit-animation-name: moveBox;
}
@-webkit-keyframes moveBox {
from {
top: 0px;
background-color: greenyellow;
width: 50px;
}
to {
top: 100px;
background-color: lawngreen;
width: 100px;
}
}
</style>
```

```
</head>
```

```
<body>  
<div></div>  
</body>  
</html>
```

Using Transforms

You can apply **2D** and **3D** transformations to an element using the **Transform** property. This property lets you *rotate*, *scale*, *move*, *skew*, and so on. The **transform** property specifies the *transform* function to apply. The **transform-origin** property specifies the *origin* or *pivot* of the transform.

Table cs9.4 describes the **transform** property values.

Table cs9.3: The <i>transform</i> properties	
Value	Description
none	No transformation.
matrix(n,n,n,n,n,n)	Defines the 2D transforms. A matrix of six values is used.
matrix3d (n,n,n,n,n,n,n,n,n,n,n,n,n,n)	Defines the 3D transforms, A matrix of sixteen values is used.
translate(x,y)	Defines the 2D transform.
translate3d(x,y,z)	Defines the 3D transform.
translateX(x) translateY(y) translateZ(z)	Defines a transform using the value for x, y, or z axis, respectively.
scale(x,y)	Defines a 2D scale transformation.
scale3d(x,y,z)	Defines a 3D scale transformation.
scaleX(x) scaleY(y) scaleZ(z)	Defines a scale transform using the value for x, y, or z axis, respectively.
rotate(angle)	Defines a 2D rotation. Accepts angle as parameter.

rotate3d(x, y, z, angle)	Defines a 3D rotation.
rotate X rotate Y rotate Z	Defines a rotation along the x, y, or z axis, respectively.
skew(x-angle, y-angle)	Defines a 2D skew transformation along the x and y axes.
skewX(angle) skewY(angle)	Defines a 2D skew transformation along the x or y axis, respectively.
perspective(n)	Defines a perspective view for a 3D element.

Let's start using the *transform* properties. The *rotate* value allows you to rotate an element by an *angle* expressed in degrees. The *scale* property lets you scale an element along the *x* and *y* axes. Listing cs9.5 shows the use of these two properties.

Listing cs9.5: /cs9/transform-1.html

```
<!DOCTYPE html>
<html>

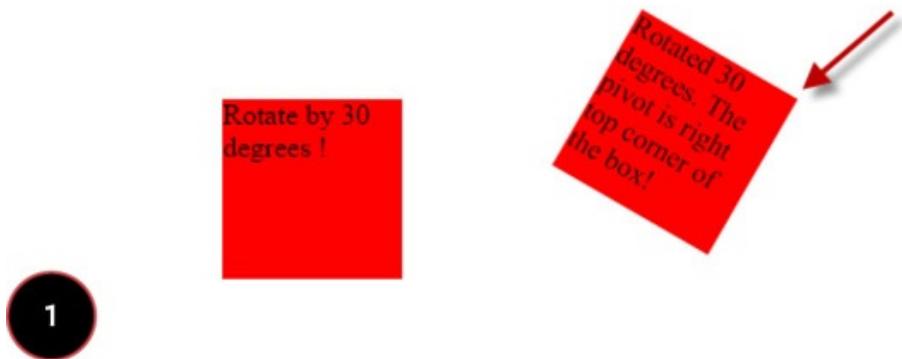
<head>
<title> Transform </title>
<style>
div {
float: left;
}
div.original {
margin: 60px;
width: 100px;
height: 100px;
background-color: red;
}
div.transformed {
```

```

margin: 60px;
width: 100px;
height: 100px;
background-color: red;
transform: rotate(10deg) scale(1.5);
}
</style>
</head>
<body>
<div class="original">Rotate by 10 degrees, scale by 1.5.</div>
<div class="transformed">Rotated 10 degrees and scaled by 1.5.</div>
</body></html>

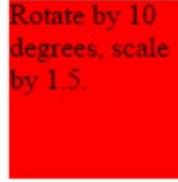
```

Notice that the original `div` element is *100px* wide and *100px* tall. I have used the `rotate` and `scale` values to `transform` the element. Notice the effect of `transform` show in Figure 1.

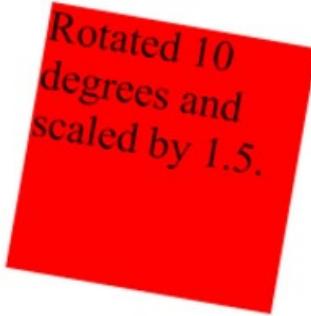


1

The `transform-origin` value lets you specify the `pivot point` or the `origin` of the transformation. Listing cs9.6 shows use of this value. In this example, I am trying to rotate the cube around the `top-right` corner of the box. The effect is shown in Figure 2.



Rotate by 10
degrees, scale
by 1.5.



2

Listing cs9.6: /cs9/transform-2.html

```
<!DOCTYPE html>
<html>

<head>
<title> Transform </title>
<style>
div {
float: left;
}
div.original {
margin: 60px;
width: 100px;
height: 100px;
background-color: red;
}
div.transformed {
margin: 60px;
width: 100px;
height: 100px;
background-color: red;
transform: rotate(30deg);
transform-origin: right top;
```

```

    }
</style>
</head>
<body>
<div class="original">Rotate by 30 degrees !</div>
<div class="transformed">Rotated 30 degrees. The pivot is right top corner of the box!
</div>
</body></html>

```

The *transform-origin* property uses the following syntax:

`transform-origin: x-axis y-axis z-axis|initial|inherit;`

Table cs9.4 describes the *transform-origin* properties values.

Table cs9.4: The <i>transform-origin</i> properties	
Value	Description
x-axis	Defines the X coordinate. Possible values are: left, center, right, length, and %.
y-axis	Defines the Y coordinate. Possible values are: top, center, bottom, length, and %.
z-axis	Defines the z axis location. The possible value is length.

Exercises

Exercise CS9-E1

Use the Listing cs9-ex1. Assign six values shown in Table cs9.2 to the six *div* boxes of the Listing cs9-ex1 and then see the effect in the browser.

Listing Cs9-ex1: /cs9/ex1.html

```
<!DOCTYPE html>
<html>

<head>
<title> Animation </title>
<style>
div {
width: 100px;
height: 30px;
background-color: beige;
transition-property: width, background-color;
transition-duration: 1s;
transition-delay: 1s;
transition-timing-function: linear;
}
div:hover {
width: 300px;
background-color: bisque;
}
</style>
</head>
```

```
<body>
<div class="one">Linear</div>
<div class="two">Ease</div>
<div class="three">Ease-In</div>
<div class="four">Ease-Out</div>
<div class="five">Ease-In-Out</div>
<div class="six">Cubic-Bezier</div>
</body>
```

```
</html>
```

Exercise CS9-E2

Use Listing cs9.3. Make changes to the `div` rule in CSS to ensure that the animation is played *two* times. Also, set the direction to alternate reverse.

Exercise CS9-E3

Use Listing cs9-ex2. Rotate the box by *360* degrees when the user hover mouse over the box. Also, change the *background color* to *lightcoral* and add a border of *10px* border with *lightgreen* color and *ridge* style.

Listing cs9-ex2: /cs9/ex3.html

```
<!DOCTYPE html>
<html>
<head>
<title> Transform </title>
<style>
div {
float: left;
}
div.original {
margin: 60px;
width: 100px;
height: 100px;
background-color: red;
}
</style>
</head>

<body>
```

```
<div class="original">Rotate it by 360 degrees !</div>
</body>
</html>
```

Summary

As you have seen, CSS gives absolute freedom to create animations. You have full control over the animation, transitions, and transformations. In this chapter, I demonstrated use of animation and transition properties.

Unit CS10-Media Queries

One of the challenge that modern day web designers/developers face is to *present content* to all devices such as desktops, mobile phones, and tablets. The CSS3 answer to this challenge is *media queries*. A media query consists of a *media type* and atleast one *expression*. Expression limits the *scope* of a stylesheet by using media features such as *height*, *width*, and *color*. In other words, media queries are used to *define style rules* for different media types and devices.

You can use media queries to check the media features [*width* and *height* of the browser window/device, *Orientation* of the display, *resolution*, and so forth] of the devices and then define style rules. Below is the syntax for the media queries:

```
@media not|only mediatype and (media feature) {  
    //CSS declarations here  
}
```

Writing Your First Media Query

Let's work on a simple example: Set the *background color* of the page to *lightgray* and when size of the browser is less than *400px*, change the *background color* to *lightcoral*. Below is the code for this example:

```
<style>  
body {  
    background-color: lightgray;  
}  
  
@media screen and (max-width: 400px) {
```

```
body {  
background-color: lightcoral;  
}  
}  
</style>
```

In the code above, I am targeting a *screen* device. When the *width* of the *browser* is greater than *400px*, the color of the background will be *lightgray*, see Figure 1. The color will change to *lightcoral* when the size is less than *400px*, see Figure 2.

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

1

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

2

Run the code shown in Listing cs10.1 and then try to make the width of the browser less than *400px*.

Listing cs10.1: /cs10/brow-size.html

```
<!DOCTYPE html>  
<html>
```

```
<head>
<style>
body {
background-color: lightgray;
}
@media screen and (max-width: 400px) {
body {
background-color: lightcoral;
}
}
</style>
</head>
```

```
<body>
```

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>

```
</body>
```

```
</html>
```

Using the Logical Operators

You can create complex media queries using the logical operators such as *not*, *and*, and *only*. The *and* operator is used to combine multiple features into a *single query*. Each combined feature must be *true* in order for the media query to be *true*. Consider the following example:

```
@media (min-width: 400px) and (orientation: landscape) { rules }
```

In this example, the code inside the curly brackets will only be effective if *minimum width* of the *display* area [*viewport*] is less than equal to *400px* and *orientation* of the display is *landscape*. If you want to add media type to the query, you have to combine it with the *and* operator as well. See below:

```
@media tv and (min-width: 400px) and (orientation: landscape) { rules }
```

In the above two examples, the style rules are *only* applied when all combined features are *true*. However, you can use a *comma separated list* if you want to apply styles when *any one* of the features is *true*. Below is an example:

```
@media (min-width: 400px), handheld and (orientation: landscape) { rules }
```

In this example, the style will be applied if *either* minimum width of the viewport of is *400px* *or* if it is a handheld device with the *landscape* orientation. Below is another example:

```
@media (max-width: 600px), (min-width: 800px) {  
  html { background: blue; }  
}
```

The *not* operator is applied to the entire media query. You cannot use it to negate individual features. Below is an example:

```
@media not screen and (color) {...}
```

In the above example, the *not* operator negates the query. Here, the expression applies to any device that does not have a *color screen*. This expression will be

applied to the *black & white* or *monochrome* screens.



The not Operator

If you specify a media feature and it does not apply to the device, the expressions involving that media feature always return false.

The *only* logical operator can be used to hide style sheets from the older user agents. This operator is not recognized by the user agents using HTML4 algorithm. The styles are hidden from the devices and browsers that don't support media queries.

In the following example, the expression is applied to the only screens in a portrait orientation for the devices capable of rendering media queries.

```
@media only screen and (orientation: portrait) { ... }
```

Media Features

Media features provide information about the device that is being used to view the content. This information is used to *evaluate* the *expressions*. Let's explore some of the features.

width and height

The *width* media feature specifies the *width* of the display area. The basic syntax require you to enter a value:

```
@media (width: 400px) { rules }
```

In this example, the rules are applied only to browsers whose width is exactly *400px*. You can also use the *max-* and *min-* prefixes with the *width* feature to specify the *maximum* and *minimum* widths:

```
@media (max-width: 800px) { rules }
@media (min-width: 400px) { rules }
```

Let's look at a practical example:

```
@media handheld and (min-width: 25em), screen and (min-width: 25em) { rules }
```

In this example, I am specifying rules for *handheld devices* or *screen devices* with a *width* greater than *25em*.

The *height* media feature works in the same way but it targets browsers based on their *height*.

color

This feature specifies the number of *bits per color component* for the output device. If device has no color, it returns *false*. If the device has different numbers of bits per color component, the smallest number is used. For example, if device uses *5 bits* for blue and *6 bits* for red, then this feature considers device to be using *5 bits* per color component. Consider the following example:

```
@media all and (min-color: 4) {rules}
```

In this example, the style rules are applied to the devices with atleast *4 bits* per color component.

You can also use the *max-* and *min-* prefixes with the color feature.

aspect-ratio

This features allows you to query the *aspect ratio* of the targeted display device. It accepts two *integer* values separated by a *slash*. This feature represents the ratio of the *horizontal pixels* to the *vertical pixels*. Consider the following example:

```
@media screen and (min-aspect-ratio: 1/1) { rules }
```

This query targets screen devices with *minimum aspect ratio* of **1:1**.

device-aspect-ratio

This feature specifies the *ratio* between the *width* and the *height* of the device. Some manufactures define *widescreen* as **16:9**, some **16:10**, and some **15:10**. You can tackle this variations by using the *min-* and *max-* prefixes. Consider the following example:

```
@media screen and (device-aspect-ratio: 16/9), screen and (device-aspect-ratio: 16/10)  
{ rules }
```

In this example, styles are applied when the *aspect ratio* of the device is either **16:9** or **16:10**.

device-height, device-width

These features define the *width* and *height* of the *device* [not just the *rendering area*], respectively. In other words, you can target the physical screen size of the devices.

resolution

You can use this feature to query the *resolution* or *pixel density* of the device. The resolution can be specified using *dots per inch (DPI)* or *dots per centimeter (dpcm)*. Consider the following example:

```
@media print and (min-resolution: 300dpi) { rules }
```

This rule is applied to the devices with atleast **300 dots per inch** of resolution.

The modern *tablets* and *smartphones* have super high resolution screens. For clarity of content, manufactures use physical *pixels*. The ratio of the *physical*

pixels to the *CSS pixels* is known as *device pixel ratio (DPR)*. For example, iPhone 6 uses a *DPR* value of *2* which essentially means that *one* CSS pixel is equal to *four* physical pixels. Table 10.1 shows various features of iPhone 6.

Table 10.1: Some iPhone 6 features	
Property	Value
Display size	4.7"
Viewport's device-width (in CSS pixels)	375
Viewport's device-width on Android devices with similar display size	360
Device Pixel Ratio	2
Rendered Pixels (default viewport size * dpr)	750x1334
Physical pixels	750x1334

orientation

This feature specifies whether the display is in landscape mode or portrait mode. The value for this feature can be two keywords: *portrait* and *landscape*. Consider the following example:

```
@media all and (orientation: portrait) { rule }
```

In this example, the style are applied only when the device is in portrait orientation.

Mobile First Development

The one approach that web artists commonly takes is that they first develop content for the *mobile devices* and then add larger elements for the users accessing content using bigger devices. When you take the desktop first approach, the changes you make in the desktop styling will change the appearance of content on the mobile devices if you have not explicitly overwritten styles for the mobile version. The other drawback with desktop first approach is that mobile devices load page by first rendering all the

desktop styling.



CSS Rules

The CSS rules are rendered by a page in a *linear* fashion from *top* to *bottom*.

Dylan Baumann of [getflywheel.com](https://getflywheel.com/layout/start-practicing-mobile-first-development) has explained the mobile first approach in detail. To know more about it, visit the following link: <https://getflywheel.com/layout/start-practicing-mobile-first-development>.

Exercises

Exercise CS10-E1

Write a query that is usable when the viewport is between *500px* and *800px* wide.

Exercise CS10-E2

Write a query that specifies a style sheet that is applied to the printed media wider than *8* inches.

Exercise CS10-E3

Write a query that specifies minimum pixel density of *2* dots per centimeter.

Summary

In this chapter, you have seen the power of the media queries. You have also learned about the difference between the *mobile first* and *desktop first* approach. I showed the different media features of a device you can check and then write style rules so that website can be rendered seamlessly across all devices. The following table summarizes various media features:

Table 10.2: Media Features	
Feature	Description
width	Width of the display area
height	Height of the display area

device-width	Width of the device
device-height	Height of the device
max-device-height	Maximum height of the device
max-device-width	Maximum width of the device
min-device-height	Minimum height of the device
min-device-width	Minimum width of the device
max-height	Maximum height of the display area
max-width	Maximum width of the display area
min-height	Minimum height of the display area
min-width	Minimum width of the display area
device-aspect-ratio	Ratio between the width and height of the device
max-device-aspect-ratio	Maximum ratio between the width and height of the device
min-device-aspect-ratio	Minimum ratio between the width and height of the device
max-aspect-ratio	Maximum ratio between the width and height of the display area
resolution	Resolution of the device using dpi or dpcm
min-resolution	Minimum resolution of the device using dpi or dpcm
max-resolution	Maximum resolution of the device using dpi or dpcm
monochrome	Number of bits per color on a monochrome device
min-monochrome	Minimum number of bits per color on a monochrome device
max-monochrome	Maximum number of bits per color on a monochrome device
orientation	Display mode, Landscape or Portrait
color	Number of bits per color component for the output device
min-color	Minimum number of bits per color component for the output device
max-color	Maximum number of bits per color component for the output device
color-index	Number of colors the device can display
min-color-index	Minimum number of colors the device can display
max-color-index	Maximum number of colors the device can display
Grid	Checks whether the device is a grid device or not

Unit HOECS1: Hands-on Exercises [CSS-I]

Hands-on Exercise - 1: Animating a Square Shape into a Circular Shape

Let's first create a simple shape animation. In this exercise, we are going to transform a *square shape* into a *circular shape*.

Let's create a basic element that will hold the animation.

```
<!DOCTYPE html>
<html>

  <head>
    <title> Square to Circle Animation </title>
    <style>
    </style>
  </head>

  <body>
    <div>

    </div>
  </body>

</html>
```

Apply the following style to the div element:

```
width: 150px;  
height: 150px;  
margin: 30px;  
background-color: burlywood;
```

Add the following code to define a block for the `@keyframe` rule:

```
@keyframes sq-to-cir{  
0%{}  
25%{}  
50%{}  
75%{}  
100%{}  
}
```

Here, `sq-to-cir` is the name of the animation.

Define the following styles for the `@keyframe` rule.

```
@keyframes sq-to-cir {  
0% {  
border-radius: 0 0 0 0;  
}  
25% {  
border-radius: 50% 0 0 0;  
}  
50% {
```

```
border-radius: 50% 50% 0 0;  
}  
75% {  
border-radius: 50% 50% 50% 0;  
}  
100% {  
border-radius: 50%;  
}  
}
```

As you can see, we have defined *four* selectors for animation time at *0%*, *25%*, *50%*, *75%*, and *100%*. We are affecting the corners of the rectangle in the following order using the *border-radius* property: *top*, *right*, *bottom*, and at *100%* stage we are setting border radius to *50%* for all four corners.

If you want, you can apply different background color at each stage and rotate the shape a bit to add the visual appeal, as shown in code below:

```
@keyframes sq-to-cir {  
0% {  
border-radius: 0 0 0 0;  
background-color: #e6aff6a;  
}  
25% {  
border-radius: 50% 0 0 0;  
background-color: #dba35c;  
}  
50% {  
border-radius: 50% 50% 0 0;  
background-color: #d39546;  
}  
75% {
```

```
border-radius: 50% 50% 50% 0;  
background-color: #76460a;  
}  
100% {  
border-radius: 50%;  
transform: rotate(90deg); }  
}
```

Now, see animation in action, add the following style for the div tag:

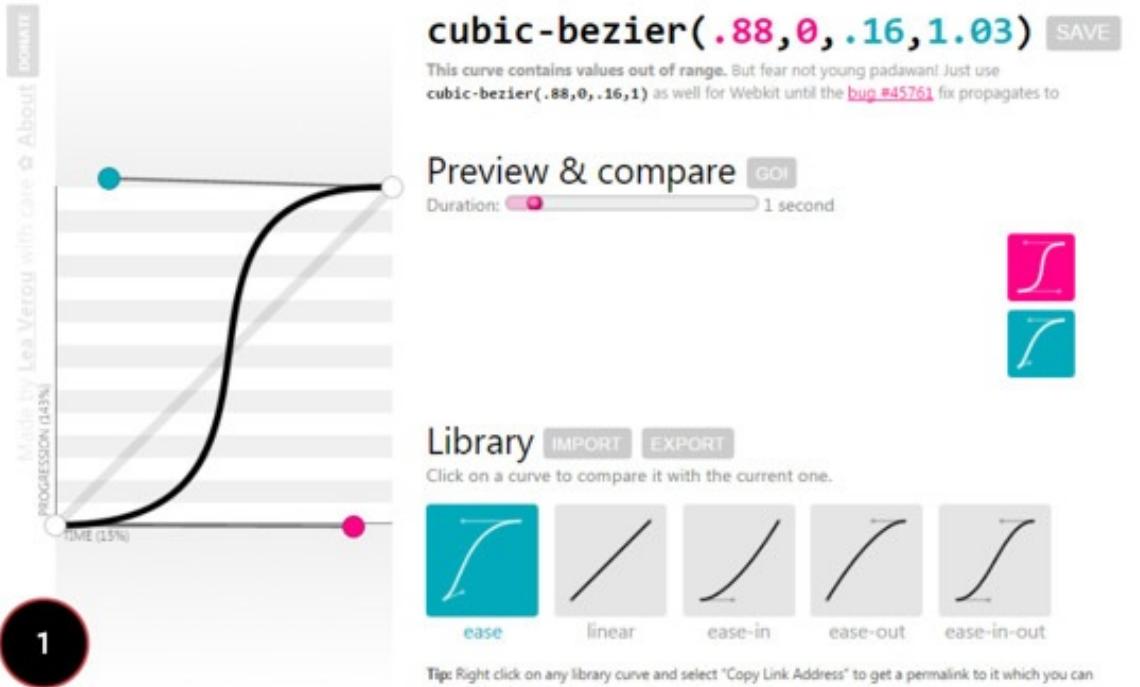
```
animation: sq-to-cir 2s 1s infinite alternate;
```

In the above rule, we are using the *sq-to-cir* name to connect with the *@keyframes* rule we defined earlier. Also, we have set animation delay of **1** seconds and animation duration of **2** seconds. In addition to this, we have set animation to loop indefinitely in the ping-pong playback style.

Let's now apply time function curve [using *animation-timing-function*] to the animation to make it believable. You can enter value manually or you can use any online tool to generate the time function values. Two such great tools are available at the following urls: <http://matthewlein.com/ceaser>, and <http://cubic-bezier.com>. Here. I am going to use the tool available at the cubic-bezier.com.

Navigate [to http://cubic-bezier.com](http://cubic-bezier.com). Set *Duration* to **1** second using the *Duration* slider and then adjust the curve located on the left of the *Duration* slider [see Figure 1]. Click on the *Go* button to preview the animation. At the top of the page, you will see the values generated for *cubic-bezier*. Copy the value and then add it the *div* style area, as shown below.

```
animation: sq-to-cir 2s 1s infinite cubic-bezier(.88,0,.16,1.03) alternate;
```



Here's the complete code:

Listing hoecs1.1: /hoecs1/hoecs1-1.html

```
<!DOCTYPE html>
<html>
<head>
<title> Square to Circle Animation </title>
<style>
div {
width: 150px;
height: 150px;
margin: 30px;
background-color: burlywood;
animation: sq-to-cir 2s 1s infinite cubic-bezier(.88, 0, .16, 1.03) alternate;
}
@keyframes sq-to-cir {
0% {
border-radius: 0 0 0 0;
```

```
background-color: #e6af6a;  
}  
25% {  
border-radius: 50% 0 0 0;  
background-color: #dba35c;  
}  
50% {  
border-radius: 50% 50% 0 0;  
background-color: #d39546;  
}  
75% {  
border-radius: 50% 50% 50% 0;  
background-color: #76460a;  
}  
100% {  
border-radius: 50%;  
transform: rotate(90deg);  
transform: scale(0%);  
}  
}  
</style>  
</head>  
<body>  
<div>  
</div>  
</body>  
</html>
```

Hands-on Exercise - 2: Creating a Bounce In Effect

Now, let's create a bounce in effect on a shape. This effect will be visible if mouse pointer is placed on the element.

Create a basic element that will hold the animation.

```
<!DOCTYPE html>
<html>
<head>
<title> Bounce In Animation </title>
<style>
</style>
</head>
<body>
<div>
</div>
</body>
</html>
```

Add the following code to the div style to define the shape:

```
width: 150px;
height: 150px;
margin: 30px;
background-color: burlywood;
```

We will now use the *scale* value of the *transform* property to create the effect. Define the *@keyframe* rule using the animation name *bounceIn*, as shown below:

```
@keyframes bounceIn {
0% {
transform: scale(.4);
}
```

```
50% {  
    transform: scale(1.1);  
}  
70% {  
    transform: scale(.9);  
}  
100% {  
    transform: scale(1);  
}  
}
```

Add the following style for hover pseudo class of div:

```
div:hover {  
    animation-name: bounceIn;  
    animation-duration: 1s;  
    animation-delay: 0.2s;  
    animation-timing-function: ease-in-out;  
    animation-fill-mode: forwards;  
}
```

The *animation-fill-mode* Property

The *animation-fill-mode* lets you specify if the animation styles are visible before or after the animation plays. By default, animation do not affect styles before the animation begins [if animation delay is applied] or after the animation is completed. You can use this property to override this behavior. You can choose one of the following values:

backwards: Before the animation i.e. during the animation delay, the style of *0%* keyframe step are applied to the element.

forwards: After the animation is completed, the styles defined by the *100%*

keyframe are retained by the element.

both: The animation will follow both rules defined by backwards and forwards.

normal: This is the default value. The animation is not applied to element before or after the animation.

Here's the complete code:

Listing hoecs1.2: /hoecs1/hoecs1-2.html

```
<!DOCTYPE html>
<html>
<head>
<title> Bounce In Animation </title>
<style>
div {
width: 150px;
height: 150px;
margin: 30px;
background-color: burlywood;
}
div:hover {
animation-name: bounceIn;
animation-duration: 1s;
animation-delay: 0.2s;
animation-timing-function: ease-in-out;
animation-fill-mode: forwards;
}
```

```
@keyframes bounceIn {  
0% {  
transform: scale(.4);  
}  
50% {  
transform: scale(1.1);  
}  
70% {  
transform: scale(.9);  
}  
100% {  
transform: scale(1);  
}  
}  
}  
</style>  
</head>  
<body>  
<div>  
</div>  
</body>  
</html>
```

Hands-on Exercise - 3: Creating a Fade In Effect

Let's now create a fade in up effect. We will use the same *div* element styling that is used in previous hands-on exercise. However, we will set *animation-name* to *fadeIn* and *animation-timing-function* to *ease-in*. Here's the code:

```
div {  
width: 150px;  
height: 150px;  
margin: 30px;
```

```
background-color: burlywood;  
animation-name: fadeIn;  
animation-duration: 1s;  
animation-delay: 0.2s;  
animation-timing-function: ease-in;  
animation-fill-mode: forwards;  
}
```

For the *fadeIn @keyframe* rule, we will use two stages: *0%* and *100%*. At *0%*, we will set *opacity* to *0* and set *translateY* value to *20px*. At *100%*, we will set *opacity* to *1* and set *translateY* value to *0px*. The shape will fade from completely transparent to the *100%* opaque state. Also, it will move *20px* up. It creates a nice looking effect. Here's the code:

```
@keyframes fadeIn {  
0% {  
    opacity: 0;  
    transform: translateY(20px);  
}  
100% {  
    opacity: 1;  
    transform: translateY(0);  
}
```

And, here's the complete code:

Listing hoecs1.3: /hoecs1/hoecs1-3.html

```
<!DOCTYPE html>  
<html>
```

```
<head>
<title> Fade In Animation </title>
<style>
div {
width: 150px;
height: 150px;
margin: 30px;
background-color: burlywood;
animation-name: fadeIn;
animation-duration: 1s;
animation-delay: 0.2s;
animation-timing-function: ease-in;
animation-fill-mode: forwards;
}
@keyframes fadeIn {
0% {
opacity: 0;
transform: translateY(20px);
}
100% {
opacity: 1;
transform: translateY(0);
}
}
</style>
</head>
<body>
<div>
</div>
</body>
</html>
```

Hands-on Exercise - 4: Create a Shake Effect

In this exercise, we are going to create a shake effect using the `transform` property. We will use the `transform` value to animate the shape. We will use the same `div` element that we have used in the previous hands-on exercise. Here's the code:

```
div {  
width: 150px;  
height: 150px;  
margin: 30px;  
background-color: burlywood;  
animation-name: shaky;  
animation-duration: 1s;  
animation-delay: 0.2s;  
animation-timing-function: ease-in;  
animation-fill-mode: forwards;  
}
```

The key to create the shaky effect is to animate the shape along the `positive` and `negative` axes at certain time intervals. At `0%` and `100%` stages, there will be no transformation. Add the following `@keyframe` rule and test the animation in the browser:

```
@keyframes shaky {  
0%, 100% {  
transform: translateX(0);  
}  
10%,  
30%,  
50%,
```

```
70%,  
90% {  
transform: translateX(-2px);  
}  
20%,  
40%,  
60%,  
80% {  
transform: translateX(2px);  
}  
}
```

Here's the complete code. In the code below, I have changed *animation-duration* to *0.2s*. Also, added the *:hover* pseudo class for the div element:

Listing hoecs1.4: /hoecs1/hoecs1-4.html

```
<!DOCTYPE html>  
<html>  
  
<head>  
<title> Shake Effect </title>  
<style>  
div {  
width: 150px;  
height: 150px;  
margin: 30px;  
background-color: burlywood;  
}  
div:hover {  
animation-name: shaky;
```

```
animation-duration: 0.2s;  
animation-delay: 0.2s;  
animation-timing-function: ease-in;  
animation-fill-mode: forwards;  
}  
  
@keyframes shaky {  
0%, 100% {  
transform: translateX(0);  
}  
10%,  
30%,  
50%,  
70%,  
90% {  
transform: translateX(-2px);  
}  
20%,  
40%,  
60%,  
80% {  
transform: translateX(2px);  
}  
}  
}  
  
</style>  
</head>  
<body>  
<div>  
</div>  
</body>  
</html>
```

Hands-on Exercise - 5: Create a Swing Effect

In this exercise, we are going to create a swing effect. We will use the *transform-origin* property to define the *pivot point* around which the shape will swing.

In the *div* styling block add the following rule:

```
transform-origin: top center;
```

For *div*'s *:hover* pseudo class, set *animation-timing-function* to *ease-in-out* and then add the following code for the *@keyframe* rule.

```
@keyframes swing {  
  20% {  
    transform: rotate(15deg);  
  }  
  40% {  
    transform: rotate(-10deg);  
  }  
  60% {  
    transform: rotate(5deg);  
  }  
  80% {  
    transform: rotate(-5deg);  
  }  
  100% {  
    transform: rotate(0deg);  
  }  
}
```

This code will give the shape pendulum like motion and at **100%** stage the shape will come to rest as we have set the angle to **0** degrees.

Here's the complete code:

Listing hoecs1.5: /hoecs1/hoecs1-5.html

```
<!DOCTYPE html>
<html>

<head>
<title> Swing Effect </title>
<style>
div {
width: 150px;
height: 150px;
margin: 30px;
background-color: burlywood;
transform-origin: top center;
}
div:hover {
animation-name: swing;
animation-duration: 1s;
animation-delay: 1s;
animation-timing-function: ease-in-out;
animation-fill-mode: forwards;
}
@keyframes swing {
20% {
transform: rotate(15deg);
}
```

```
40% {  
    transform: rotate(-10deg);  
}  
60% {  
    transform: rotate(5deg);  
}  
80% {  
    transform: rotate(-5deg);  
}  
100% {  
    transform: rotate(0deg);  
}  
}  
}  
</style>  
</head>  
<body>  
<div>  
</div>  
</body>  
</html>
```

Hands-on Exercise - 6: Creating the Wobble Effect

Here, we are going to create a wobble effect. The key to create the wobble effect is translate the shape along the *+ve* and *-ve x-axis* as well as rotate the shape clockwise or anti-clockwise. Here's the style code for the *div* tag.

```
div {  
    width: 150px;  
    height: 150px;  
    margin: 30px;  
    background-color: burlywood;
```

```
transform-origin: top center;  
}  
  
div:hover {  
animation-name: wobble;  
animation-duration: 1s;  
animation-delay: 1s;  
animation-timing-function: ease-in-out;  
animation-fill-mode: forwards;  
}
```

And here's the code for [*@keyframe*](#) rule:

```
@keyframes wobble {  
0% {  
transform: translateX(0%);  
}  
15% {  
transform: translateX(-25%) rotate(-7deg);  
}  
30% {  
transform: translateX(20%) rotate(5deg);  
}  
45% {  
transform: translateX(-15%) rotate(-7deg);  
}  
60% {  
transform: translateX(10%) rotate(5deg);  
}  
75% {  
transform: translateX(-5%) rotate(-2deg);  
}
```

```
100% {  
  transform: translateX(0%);  
}  
}
```

Here's the complete code:

Listing hoecs1.6: /hoecs1/hoecs1-6.html

```
<!DOCTYPE html>  
<html>  
  
<head>  
<title> Wobble Effect </title>  
<style>  
div {  
  width: 150px;  
  height: 150px;  
  margin: 30px;  
  background-color: burlywood;  
  transform-origin: top center;  
}  
div:hover {  
  animation-name: wobble;  
  animation-duration: 1s;  
  animation-delay: 1s;  
  animation-timing-function: ease-in-out;  
  animation-fill-mode: forwards;  
}  
@keyframes wobble {  
  0% {
```

```
transform: translateX(0%);  
}  
15% {  
transform: translateX(-25%) rotate(-7deg);  
}  
30% {  
transform: translateX(20%) rotate(5deg);  
}  
45% {  
transform: translateX(-15%) rotate(-7deg);  
}  
60% {  
transform: translateX(10%) rotate(5deg);  
}  
75% {  
transform: translateX(-5%) rotate(-2deg);  
}  
100% {  
transform: translateX(0%);  
}  
}  
</style>  
</head>  
<body>  
<div>  
</div>  
</body>  
</html>
```

Hands-on Exercise - 7: Card Flip Effect

Let's now create the card flip effect. We are going to reveal the text once card

is flipped.

Now, we will create a container that will hold the card and text containers. Add the following HTML code between the `<body>` and `</body>` tags:

```
<div class="wrapper">
<div class="card">
<div class="frontFace">

</div>
<div class="backFace">
The King of the Jungle !
</div>
</div>
</div>
```

Notice the `frontFace` and `backFace` classes hold the image and text, respectively. These classes will position the elements absolutely so that they can overlay each other. Also, we will set the `top` and `left` positions to `0` for these two classes. The `frontFace` element will have a higher `z-index`. Let's start styling:

Since, we are dealing with text, let's first define the `font family`. Add the following code to the `<style>` and `</style>` tags:

```
body {
font: 600 14px/24px Helvetica, Arial, "Lucida Grande", Sans-Serif;
margin: 30px;
}
```

Add perspective to the wrapper:

```
.wrapper {  
perspective: 1000;  
}
```

The *perspective* property controls the distance between the *z=0* plane and the user. As a result, the 3D positioned element receives some *perspective*. Each element with the *z* value greater than *0* becomes larger and each element with the *z* values less than *0* becomes smaller. The value specified using the *perspective* property controls the *strength* of the effect. The *vanishing point* is always at the *center* of the element, by default. However, you can set your own point using the *perspective-origin* property.

Add the following style:

```
.wrapper .card:hover {  
transform: rotateY(180deg);  
}
```

This code flips the card when hovered.

Add the following style:

```
.wrapper,  
.frontFace,  
.backFace {  
width: 300px;  
height: 159px;  
}
```

It sets the *width* and *height* of the elements according to the size of the card

image. Add the following style to set the transition speed:

```
.card {  
transition: all 1s ease-in-out;  
transform-style: preserve-3d;  
position: relative;  
}
```

Now, if you hover over the card in the browser, you will notice that image and text are flipping but we need to hide the back side of the elements. Let's work on that. Add the following code:

```
.frontFace,  
.backFace {  
backface-visibility: hidden;  
position: absolute;  
top: 0;  
left: 0;  
}
```

Now, hover the mouse pointer over the card, you would notice that the text is appearing on the image and also when the card is flipped, text does not appear. To fix this, we will rotate the *backFace* to *180* degrees so that it behaves like expected [as a *back* element, as if text is written on the back of the card]. Also, we will set *z-index* of the *frontFace* to *2* so that it appears in front of the text in the browser. Add the following styles:

```
.frontFace {  
z-index: 2;  
}  
.backFace {  
transform: rotateY(180deg);  
}
```

Test the code in the browser, it should work as expected. Now, add some more styles to beatify the card.

Here's the complete code:

Listing hoecs1.7: /hoecs1/hoecs1-7.html

```
<!DOCTYPE html>
<html>

<head>
<title> Card Flip </title>
<style>
body {
font: 600 14px/24px Helvetica, Arial, “Lucida Grande”, Sans-Serif;
margin: 30px;
}
.wrapper {
perspective: 1000;
}
.wrapper .card:hover {
transform: rotateY(180deg);
}
.wrapper,
.frontFace,
.backFace {
width: 310px;
height: 169px;
}
.card {
```

```
transition: all 1s ease-in-out;
transform-style: preserve-3d;
position: relative;
border-radius: 10px;
}

.frontFace,
.backFace {
backface-visibility: hidden;
position: absolute;
top: 0;
left: 0;
}
.frontFace img {
border-radius: 10px;
}
.frontFace {
z-index: 2;
}
.backFace {
display: block;
border-radius: 10px;
background: #d9d9e0;
color: #0c2c02;
line-height: 150px;
text-align: center;
transform: rotateY(180deg);
}
}

</style>
</head>
```

```
<body>
<div class="wrapper">
<div class="card">
<div class="frontFace">

</div>
<div class="backFace">
The King of the Jungle !
</div>
</div>
</div>
</body>

</html>
```

Hands-on Exercise - 8: Swapping One Image with Another

In this exercise, we are going to create a cross fading effect on two images. This has traditionally been done using JavaScript, however, using CSS transitions you can achieve this effect easily. This is done by placing one image on top of another and then changing the opacity of the top image on hover.

Add the following code between the `<body>` and `</body>` tags:

```
<div class="crossFade">


</div>
```

In this exercise, we are trying to reveal the color image when the user hovers the mouse over the gray image. Therefore, we have placed the color image

first in the `div` element defined by the `crossfade` class. We will add style to set the `opacity` to `0` of the gray image to `1` when user hovers mouse over it. Let's first style the container.

```
.crossFade {  
    position: relative;  
    height: 159px;  
    width: 300px;  
}
```

Add the following code to style the `img` tag:

```
.crossFade img {  
    position: absolute;  
    left: 0;  
    top: 0;  
    transition: opacity 0.3s ease-in-out;  
}
```

Here, we have set the position to `absolute` and `top` and `left` to `0` so that the images overlap each other at the same location. The `transition` property sets the speed of the fade.

Finally, add the following style that reveals the colored image:

```
.crossFade img.topImage:hover {  
    opacity: 1;  
}
```

Test the code in the browser. Here's the complete code:

Listing hoecs1.8: /hoecs1/hoecs1-8.html

```
<!DOCTYPE html>
<html>

<head>
<title> Image Swap </title>
<style>
.crossFade {
position: relative;
height: 159px;
width: 300px;
}
.crossFade img {
position: absolute;
left: 0;
top: 0;
transition: opacity 0.3s ease-in-out;
}
.crossFade img.topImage:hover {
opacity: 0;
}
</style>
</head>
<div class="crossFade">


</div>

<body>
```

```
</body>
```

```
</html>
```

If you want an image to be visible for *n* number of seconds before fading to another one, you can use the [`@keyframe`](#) rule. Here's the complete code:

Listing hoecs1.8B: /hoecs1/hoecs1-8-B.html

```
<!DOCTYPE html>
<html>

<head>
<title> Image Swap </title>
<style>
.crossFade {
position: relative;
height: 159px;
width: 300px;
}
.crossFade img {
position: absolute;
left: 0;
top: 0;
transition: opacity 0.3s ease-in-out;
}
.crossFade img.topImage {
animation-name: cf;
animation-timing-function: ease-in-out;
animation-iteration-count: infinite;
```

```
animation-duration: 5s;  
animation-direction: alternate;  
}  
  
@keyframes cf {  
0% {  
opacity: 1;  
}  
100% {  
opacity: 0;  
}  
}  
  
</style>  
</head>  
<div class="crossFade">  
  
  
</div>  
  
<body>  
  
</body>  
  
</html>
```

Hands-on Exercise - 9: Creating a Slideshow

Now, let's put the knowledge that we have gained from hands-on exercise 1 to hands-on exercise 8 and create a slideshow using the CSS animation. We have five images of the tiger for which we want to create the animation.

Add the following code between the `<body>` and `</body>` tags.

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

Add style for *crossfade* class:

```
.crossfade {  
  position: relative;  
  height: 175px;  
  width: 250px;  
}
```

Add following style to define duration of the animation:

```
.crossfade > img {  
  position: absolute;  
  top: 0px;  
  left: 0px;  
  opacity: 0;  
  color: transparent;  
  z-index: 0;  
  animation: iani 30s linear infinite 0s;  
}
```

The *position* is defined as *absolute* therefore all images will overlap each other. I have also set the *opacity* to *0*. I will animate it later using the *@keyframes* rule. I have also set the *duration* of the *animation* to *30* seconds.

Add the following style to define the `@keyframes` rule:

```
@keyframes iani {  
    0% {  
        opacity: 0;  
        animation-timing-function: ease-in;  
    }  
    10% {  
        opacity: 1;  
        animation-timing-function: ease-out;  
    }  
    20% {  
        opacity: 1;  
    }  
    30% {  
        opacity: 0;  
    }  
    100% {  
        opacity: 0;  
    }  
}
```

Now, to understand the percentages I came up with, we need to understand little bit of math behind it. We have **5** images and we want each image to be visible for **6** seconds, therefore, the total number of seconds required for a whole cycle is **30**. We need to define percentages for the `@keyframes` rule, therefore, we should know at what percentage value the second image will start appearing [the first image will fade out]. We know that the each image will last for **6** seconds, therefore, the percentage value will be **(6/30)*100=20%**. We will also prevent image from fading out for the whole duration. Therefore, we will allow fading for the first half of total animation time for the image i.e **20/2=10%**. According to the math we did, at **0%** the

opacity will be **0**, at **10%** opacity will be **1**, **30%** the image will disappear completely and for rest of the time the image will be transparent.



Percentage Calculations.xls

For easier math, I have provided an [MS Excel](#) sheet with the name *Precentge Calculations.xls*. Enter the number of images and duration for each image in the *B* and *C* columns, respectively. The sheet will calculate percentage steps for you.

Now, let's target each *img* tag we defined using the *nth-child()* selector.

```
.crossfade > img:nth-child(1) {  
    animation-delay: 0s;  
}  
  
.crossfade > img:nth-child(2) {  
    animation-delay: 6s;  
}  
  
.crossfade > img:nth-child(3) {  
    animation-delay: 12s;  
}  
  
.crossfade > img:nth-child(4) {  
    animation-delay: 18s;  
}  
  
.crossfade > img:nth-child(5) {  
    animation-delay: 24s;  
}
```

There will be no delay for *nth-child(1)* as it is the first image in the stack. For all other images the delay is in increments of **6** seconds.

Here's the complete code:

Listing hoecs1.9: /hoecs1/hoecs1-9.html

```
<!DOCTYPE html>
```

```
<html>

<head>
<title> Multiple Image Swap </title>
<style>
.crossfade {
position: relative;
height: 175px;
width: 250px;
}
.crossfade > img {
position: absolute;
top: 0px;
left: 0px;
opacity: 0;
color: transparent;
z-index: 0;
animation: iani 30s linear infinite 0s;
}
.crossfade > img:nth-child(1) {
animation-delay: 0s;
}
.crossfade > img:nth-child(2) {
animation-delay: 6s;
}
.crossfade > img:nth-child(3) {
animation-delay: 12s;
}
.crossfade > img:nth-child(4) {
animation-delay: 18s;
}
```

```
.crossfade > img:nth-child(5) {  
    animation-delay: 24s;  
}  
  
@keyframes iani {  
    0% {  
        opacity: 0;  
        animation-timing-function: ease-in;  
    }  
    10% {  
        opacity: 1;  
        animation-timing-function: ease-out;  
    }  
    20% {  
        opacity: 1;  
    }  
    30% {  
        opacity: 0;  
    }  
    100% {  
        opacity: 0;  
    }  
}  
</style>  
</head>
```

```
<body>  
<div class="crossfade">  
      
      
    
```

```



</div>
</body></html>

```

Hands-on Exercise -10: Styling Tables

Let's style some tables now. Before you start download the [hoecls1-10.html](#) and open it in the text editor.

Add the following code to between the style tags:

```

@charset "utf-8";
@import "tables.css";

```

Create a new file and save it as [tables.css](#). Now, let's add styles to create a table as shown in Figure 2.

Salesman	Appointments	Sales	Expenses
Phillip	NewYork	57	2500
Brent	Miami	48	2000
Bill	Washington	75	3700
Fred	Los Angeles	56	4000
Tim	Utah	45	6000
Graham	Shicago	50	450

2

Add the following code in the CSS file to define the [font](#) for the tables. Also, I will add bottom margin to the tables:

```

body {
  font-family: verdana, Sans-Serif;
}

```

```

table {
  margin-bottom: 25px;
}

```

```
}
```

Add the following code:

```
#style-1 {  
    font-size: 12px;  
    background: #d9d9d9;  
    width: 500px;  
    border-collapse: collapse;  
    text-align: left;  
}
```

Notice that I have set general properties for the table. This code fills the table with the *light gray* background and also *collapses* the *borders*.

Add the following code:

```
#style-1 th {  
    font-size: 13px;  
    font-weight: bold;  
    color: #039;  
    padding: 10px 8px;  
    border-bottom: 2px solid #011a6a;  
}
```

This code targets the column headers. It sets the *font* properties and *padding* properties for the headings. Also, it defines a *2* pixel bottom border for the cells. Using the same method, we will add padding and border to the rest of the rows. Add the following code:

```
#style-1 td {  
    border-bottom: 1px solid #a59a9a;
```

```
color: #46468d;  
padding: 6px 8px;  
}
```

Now, I want to change the color of the text when user hovers mouse pointers over a row. Add the following code:

```
#style-1 tbody tr:hover td {  
color: #030355;  
}
```

Here's the complete CSS code:

```
#style-1 {  
font-size: 12px;  
background: #d9d9d9;  
width: 500px;  
border-collapse: collapse;  
text-align: left;  
}
```

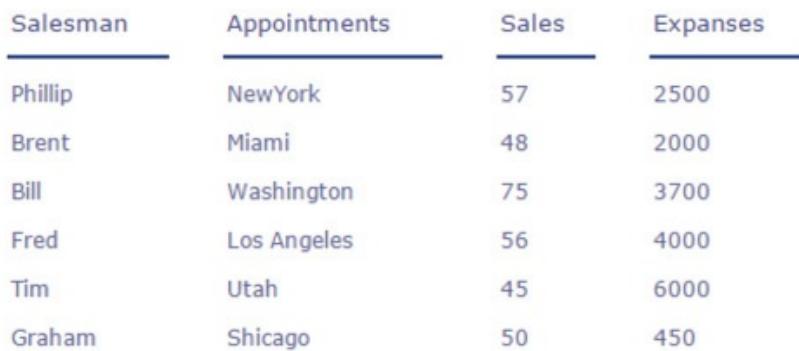
```
#style-1 th {  
font-size: 13px;  
font-weight: bold;  
color: #039;  
padding: 10px 8px;  
border-bottom: 2px solid #011a6a;  
}
```

```
#style-1 td {  
border-bottom: 1px solid #a59a9a;
```

```
color: #46468d;  
padding: 6px 8px;  
}
```

```
#style-1 tbody tr:hover td {  
color: #030355;  
}
```

Now, let's style the table as shown in Figure 3. I have given the id *style-2* to this table. Add the following code to define the general properties of the table:



Salesman	Appointments	Sales	Expanses
Phillip	NewYork	57	2500
Brent	Miami	48	2000
Bill	Washington	75	3700
Fred	Los Angeles	56	4000
Tim	Utah	45	6000
Graham	Shicago	50	450

3

```
#style-2 {  
font-size: 12px;  
background: #FFFFFF;  
width: 500px;  
border-collapse: collapse;  
text-align: left;  
}
```

Add the following code to style headings:

```
#style-2 th {  
padding: 10px 4px;  
font-weight: normal;  
font-size: 13px;
```

```
border-bottom: 2px solid #243a81;  
border-right: 25px solid red;  
border-left: 25px solid red;  
color: #46468d;  
}
```

If you render the html code in the browser, you will notice *25px* red colored border. The *border-left* and *border-right* properties are creating spacing on the left and right of the cells. The red color is only for demonstration purposes which I will change to match the background color [white]. Similarly, add the following code to add spaces on other rows of the table:

```
#style-2 td {  
padding: 12px 2px 0px 2px;  
border-right: 25px solid red;  
border-left: 25px solid red;  
color: #669;  
}
```

Now, change red to *#fff*.

Here's the complete style code:

```
#style-2 {  
font-size: 12px;  
background: #FFFFFF;  
width: 500px;  
border-collapse: collapse;  
text-align: left;  
}
```

```
#style-2 th {
```

```

padding: 10px 4px;
font-weight: normal;
font-size: 13px;
border-bottom: 2px solid #243a81;
border-right: 25px solid #fff;
border-left: 25px solid #fff;
color: #46468d;
}

```

```

#style-2 td {
padding: 12px 2px 0px 2px;
border-right: 25px solid #fff;
border-left: 25px solid #fff;
color: #669;
}

```

Now, let's style the table, as shown in Figure 4.

Salesman	Appointments	Sales	Expenses
Phillip	NewYork	57	2500
Brent	Miami	48	2000
Bill	Washington	75	3700
Fred	Los Angeles	55	4000
Tim	Utah	45	6000
Graham	Chicago	50	450

4

As usual, add the following code to define general properties of the table:

```

#style-3 {
font-size: 13px;

```

```
width: 500px;  
text-align: left;  
border-collapse: collapse;  
background: url('images/tbl.jpg') 190px 50px no-repeat;  
}
```

Add the following code to style *th* and *td* tags:

```
#style-3 th {  
padding: 10px;  
font-weight: normal;  
font-size: 13px;  
color: #060657;  
border-top: 2px solid #1e46b1;  
border-bottom: 2px solid #1e46b1;  
}
```

```
#style-3 td {  
padding: 12px 9px;  
color: #1d1d72;  
border-top: 1px solid #969696;  
}
```

To add red border at the bottom of the table, I will target the last *tr* element using the *last-child* selector. Add the following code:

```
#style-3 tr:last-child {  
color: red;  
border-bottom: 2px solid red;  
}
```

Add the following code to change the *font-weight* and *font color* of the text.

```
#style-3 tbody tr:hover td {  
color: darkslateblue;  
font-weight: bold;  
}
```

To add the background image to the table, add the following code in the **#style-3** id.

```
background: url('images/tbl.jpg') 180px 90px no-repeat;
```

Here's the complete style code:

```
#style-3 {  
font-size: 13px;  
width: 500px;  
text-align: left;  
border-collapse: collapse;  
background: url('images/tbl.jpg') 180px 90px no-repeat;  
}
```

```
#style-3 th {  
padding: 10px;  
font-weight: normal;  
font-size: 13px;  
color: #060657;  
border-top: 2px solid #1e46b1;  
border-bottom: 2px solid #1e46b1;
```

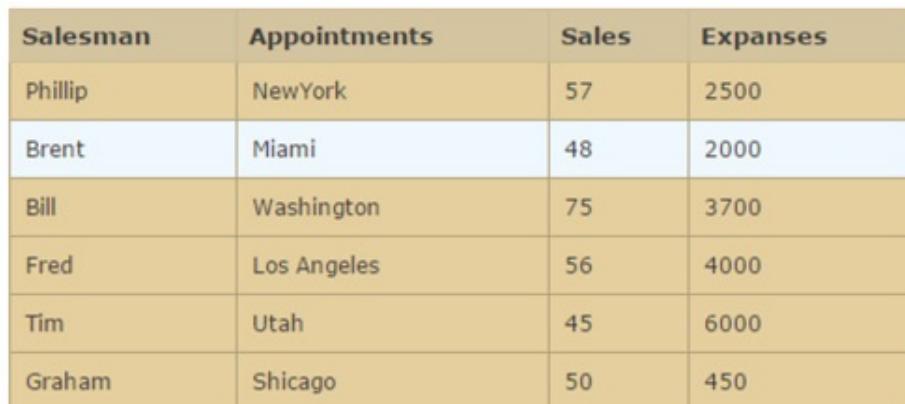
}

```
#style-3 td {  
padding: 12px 9px;  
color: #1d1d72;  
border-top: 1px solid #969696;  
}
```

```
#style-3 tr:last-child {  
color: red;  
border-bottom: 2px solid red;  
}
```

```
#style-3 tbody tr:hover td {  
color: darkslateblue;  
font-weight: bold;  
}
```

Below is the CSS code for one more style of the table [see Figure 5]. Experiment with the table and try changing properties of the table to create a different look. Here's the complete CSS code. I have assigned it to id [#style-4](#).



Salesman	Appointments	Sales	Expenses
Phillip	New York	57	2500
Brent	Miami	48	2000
Bill	Washington	75	3700
Fred	Los Angeles	56	4000
Tim	Utah	45	6000
Graham	Chicago	50	450

5

```
#style-4 {  
font-size: 12px;  
color: #3c3c3c;  
width: 500px;
```

```
border-width: 1px;  
border-color: #b9a986;  
border-collapse: collapse;  
}  
  
}
```

```
#style-4 th {  
font-size: 13px;  
background-color: #d4c4a0;  
border-width: 1px;  
padding: 6px;  
border-style: solid;  
border-color: #cbba95;  
text-align: left;  
vertical-align: middle;  
}
```

```
#style-4 tr {  
background-color: #e5cf9e;  
}
```

```
#style-4 td {  
font-size: 12px;  
border-width: 1px;  
padding: 8px;  
border-style: solid;  
border-color: #af9f7a;  
}
```

```
#style-4 tr:hover {  
background-color: aliceblue;}
```

Finally, let's create a table with rounded corners, as shown in Figure 6.

Salesman	Appointments	Sales	Expenses
Phillip	New York	57	2500
Brent	Miami	48	2000
Bill	Washington	75	3700
Fred	Los Angeles	56	4000
Tim	Utah	45	6000
Graham	Chicago	50	450

6

Add the following rule to define general properties of the table:

```
#style-5 {
border-collapse: separate;
border-spacing: 0;
width: 500px;
border: solid #c3c0c0 1px;
border-radius: 10px;
box-shadow: 0 1px 1px #ccc;
}
```

Notice that I have added *border-radius* property to create rounded corners. Also, a *drop-shadow* effect is created using the *box-shadow* property.

Add the following rules to the *style* tag to create *1px* border. Also, align text to the left:

```
#style-5 td,
th {
border-left: 1px solid #ccc;
border-top: 1px solid #ccc;
padding: 10px;
text-align: left;
```

```
}
```

Next, you will style the table header using `th` element.

Add the following code to the `style` element to set `background`, `drop-shadow`, and `text-shadow` effects:

```
#style-5 th {  
    background-color: #ECECEC;  
    background-image: linear-gradient(#f4ebeb, #e6fdfd);  
    box-shadow: 0 1px 0 #ffffff inset;  
    border-top: none;  
    text-shadow: 0 1px 0 #ffffff;  
}
```

Figure 7 shows the result. Notice that the corners are not round. Now, you will round the corners.

Salesman	Appointments	Sales	Expenses
Phillip	NewYork	57	2500
Brent	Miami	48	2000
Bill	Washington	75	3700
Fred	Los Angeles	56	4000
Tim	Utah	45	6000
Graham	Shicago	50	450

7

Add the following code to the `style` element:

```
#style-5 th:first-child {  
    border-radius: 10px 0 0 0;  
}
```

The `th:first-child` selector targets the first cell of the table.

Now, target the last cell of the first row. Add the following code:

```
#style-5 th:last-child {  
border-radius: 0 10px 0 0;  
}
```

Add the following code:

```
#style-5 tr:last-child td:last-child {  
border-radius: 0 0 10px 0;  
}
```

The `#style-5 tr:last-child td:last-child` rule targets the cell which has the *placeholder* as *Graham*. Next, you will target the right most cell of the last row.

Add the following code:

```
#style-5 tr:last-child td:last-child {  
border-radius: 0 0 10px 0;  
}
```

Figure 8 shows the result. If you zoom in on the cell [see Figure 9] that has the placeholder Graham, you will see an additional border line. Next, you will fix it.

Salesman	Appointments	Sales	Expenses
Phillip	NewYork	57	2500
Brent	Miami	48	2000
Bill	Washington	75	3700
Fred	Los Angeles	56	4000
Tim	Utah	45	6000
Graham	Shicago	50	450

8

Add the following code to the *style* tag:

```
#style-5 td:first-child,  
th:first-child {  
border-left: none;}
```



9

Finally, let's add a background transition to the rows when the user hovers mouse over them. Add the following code to the *style* tag:

```
#style-5 tr:hover {  
background: #e2dddd;  
transition: all 0.5s ease-in-out;  
}
```

Here's the complete CSS code:

```
#style-5 {  
border-collapse: separate;  
border-spacing: 0;  
width: 500px;  
border: solid #c3c0c0 1px;  
border-radius: 10px;  
box-shadow: 0 1px 1px #ccc;  
}  
  
#style-5 th,  
td {
```

```
border-left: 1px solid #ccc;  
border-top: 1px solid #ccc;  
padding: 10px;  
text-align: left;  
}
```

```
#style-5 th {  
background-color: #ECECEC;  
background-image: linear-gradient(#f4ebcb, #e6fdfd);  
box-shadow: 0 1px 0 #ffffff inset;  
border-top: none;  
text-shadow: 0 1px 0 #ffffff;  
}
```

```
#style-5 th:first-child {  
border-radius: 10px 0 0 0;  
}
```

```
#style-5 th:last-child {  
border-radius: 0 10px 0 0;
```

```
}
```

```
#style-5 tr:last-child td:last-child {  
border-radius: 0 0 10px 0;  
}
```

```
#style-5 tr:last-child td:last-child {  
border-radius: 0 0 10px 0;  
}
```

```
#style-5 td:first-child,  
th:first-child {  
border-left: none;  
}
```

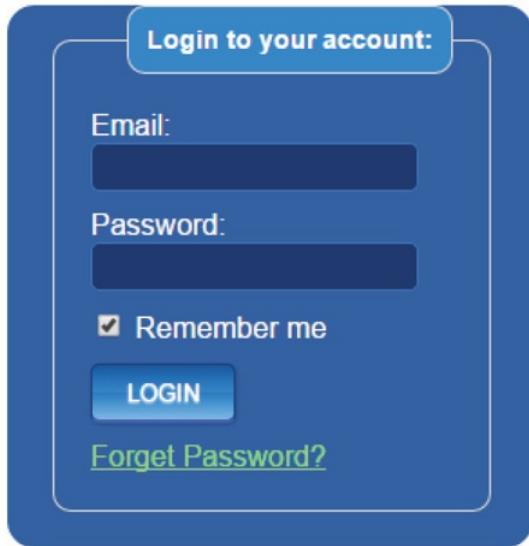
```
#style-5 tr:hover {  
background: #e2dddc;  
transition: all 0.5s ease-in-out;  
}
```



Hands-on Exercise -10 Files
Html: /hoecs1/hoecs1-10.html
CSS: /hoecs1/tables.css

Hands-on Exercise - 11: Styling a Login Form

In this exercise, you will style a login form [see Figure 10].



10

Open the code editor you are using and then create a new HTML document. Type following code and then save the html file:

```
<!DOCTYPE html>
<html>
<head>
<title> Styling a Table </title>
<style>
</style>
</head>

<body>

</body>

</html>
```

Add the following code between the `<body>` and `</body>` tags to create the HTML code for the exercise:

```
<div class="box">
```

```
<form action="example.php">
<fieldset>
<legend>Login to your account:</legend>
<div>
<label for="email" class="title">Email:</label>
<input type="email" id="email" name="email" />

</div>
<div>
<label for="pwd" class="title">Password:</label>
<input type="password" id="pwd" name="pwd" />

</div>
<div>

<input type="checkbox" name="reme" id="reme" checked />
<label for="reme" class="title">Remember me</label>

</div>
<div>
<input type="submit" value="LOGIN" id="submit" />
</div>
<div class="link">

<a href="example.php">Forget Password?</a>
</div>

</fieldset>
```

```
</form>
```

```
</div>
```

Next, you will start styling the form elements. Add the following code to specify the *font family*:

```
body {  
    font-family: helvetica, verdana, sans-serif;  
}
```

Add the following code to define the *base* of the form:

```
.box {  
    background: #3361a1;  
    border-radius: 15px;  
    width: 290px;  
    height: 300px;  
}
```

Figure 11 shows the result.



11

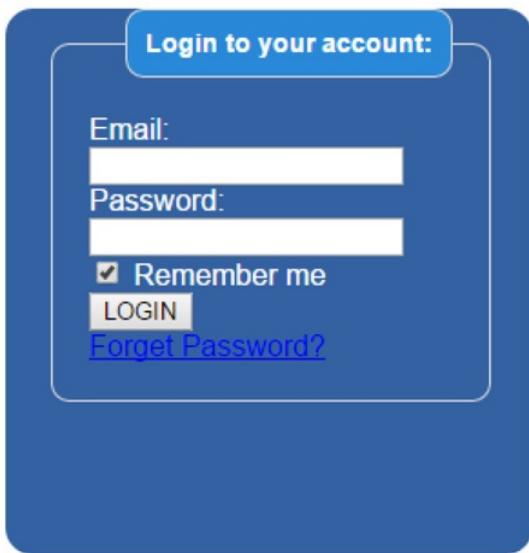
Add the following code to change the *color* of the text to *white*:

```
.title {  
color: white;  
}  
  
Add the following code to style fieldset and legend elements:  
  
fieldset {  
margin: 25px;  
width: 200px;  
border: 1px solid #dcdcdc;  
border-radius: 10px;  
padding: 20px;  
text-align: left;  
}  
  
legend {  
color: white;  
font-weight: bold;  
background-color: #2989d8;
```

```
border: 1px solid #dcdcdc;  
border-radius: 10px;  
padding: 10px 10px;  
text-align: right;  
font-size: 14px;  
}  
}
```

Figure 12 shows the result.

12



Add the following code to style the input elements of type *email* and *password*:

```
input[type="email"],  
input[type="password"] {  
border-style: solid;  
border-width: 1px;  
border-color: rgb(64, 114, 183);  
background-color: rgb(0, 52, 113);  
border-radius: 5px;  
padding: 5px 5px 5px 5px;  
margin-bottom: 10px;
```

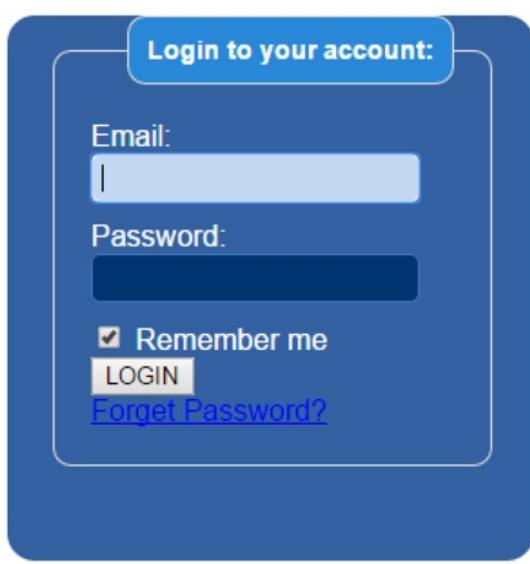
```
display: block;
```

```
}
```

Add the following code to define the behavior of the *input boxes* and when user *hovers* mouse over them:

```
input:focus {  
background-color: #c4d8f2;  
border: 1px solid #f6fbfc;}
```

Figure 13 shows the result.



Add the following code to style the *submit* button:

```
input[type="submit"] {  
width: 80px;  
height: 33px;  
margin-top: 10px;  
color: white;  
border-radius: 5px;  
border: 1px solid rgb(10, 61, 129);  
box-shadow: 0.5px 0.866px 2px 0px rgba(2, 23, 51, 0.259);  
text-shadow: 0px 1px 1px #ffffff;
```

```

background: -webkit-linear-gradient(top, #1e5799 0%, #2989d8 50%, #207cca 51%, #7db9e8 100%);

border-bottom: 2px solid #2a5be0;

}

input[type="submit"]:hover {

color: bisque;

border: 1px solid #a4a4a4;

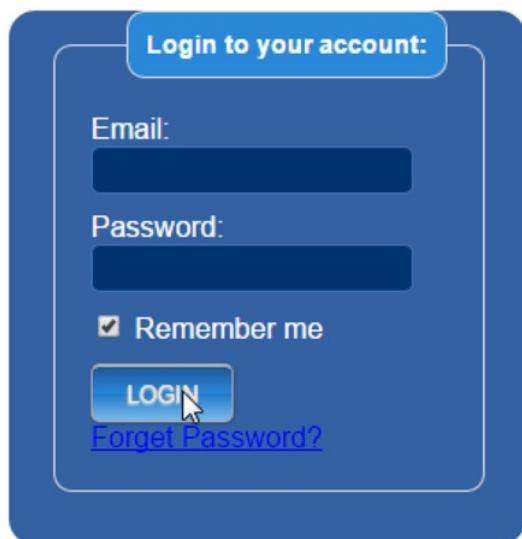
border-top: 2px solid #b2b2b2;

background: linear-gradient(top, #1e5799 0%, #2989d8 50%, #207cca 51%, #7db9e8 100%);

}

```

Figure 14 shows the result.



14

Finally, add the following code to style the Forget password link:

```

.link {

margin-top: 10px;

}

.link [href] {

margin-top: 25px;

color: #89f889;

}

```

Here's the complete code:

Listing hoecs1.10: /hoecs1/hoecs1-11.html

```
<!DOCTYPE html>
<html>

<head>
<title> Styling a Table </title>
<style>
body {
font-family: helvetica, verdana, sans-serif;
}

.box {
background: #3361a1;
border-radius: 15px;
width: 290px;
height: 270px;
}

.title {
color: white;
}

fieldset {
margin: 25px;
width: 200px;
border: 1px solid #dcdcdc;
border-radius: 10px;
padding: 20px;
text-align: left;
}

legend {
```

```
color: white;  
font-weight: bold;  
background-color: #2989d8;  
border: 1px solid #dcdcdc;  
border-radius: 10px;  
padding: 10px 10px;  
text-align: right;  
font-size: 14px;  
}  
}  
  
input[type="email"],  
input[type="password"] {  
border-style: solid;  
border-width: 1px;  
border-color: rgb(64, 114, 183);  
background-color: rgb(0, 52, 113);  
border-radius: 5px;  
padding: 5px 5px 5px 5px;  
margin-bottom: 10px;  
display: block;  
}  
  
input:focus {  
background-color: #c4d8f2;  
border: 1px solid #f6fbfc;  
}  
  
input[type="submit"] {  
width: 80px;  
height: 33px;  
margin-top: 10px;  
color: white;  
border-radius: 5px;
```

```
border: 1px solid rgb(10, 61, 129);
box-shadow: 0.5px 0.866px 2px 0px rgba(2, 23, 51, 0.259);
text-shadow: 0px 1px 1px #ffffff;
background: -webkit-linear-gradient(top, #1e5799 0%, #2989d8 50%, #207cca 51%, #7db9e8 100%);
border-bottom: 2px solid #2a5be0;
}

input[type="submit"]:hover {
color: bisque;
border: 1px solid #a4a4a4;
border-top: 2px solid #b2b2b2;
background: linear-gradient(top, #1e5799 0%, #2989d8 50%, #207cca 51%, #7db9e8 100%);
}

.link {
margin-top: 10px;
}

.link [href] {
margin-top: 25px;
color: #89f889;
}

</style>
</head>
```

```
<body>
<div class="box">

<form action="example.php">
<fieldset>
<legend>Login to your account:</legend>
<div>
<label for="email" class="title">Email:</label>
```

```
<input type="email" id="email" name="email" />

</div>
<div>
<label for="pwd" class="title">Password:</label>
<input type="password" id="pwd" name="pwd" />

</div>
<div>

<input type="checkbox" name="reme" id="reme" checked />
<label for="reme" class="title">Remember me</label>

</div>
<div>
<input type="submit" value="LOGIN" id="submit" />
</div>
<div class="link">

<a href="example.php">Forget Password?</a>
</div>

</fieldset>

</form>

</div>
</body>
```

</html>

Unit HOEHT1: Hands-on Exercises [HTML-I]

Hands-on Exercise -1: Creating Fixed Width Layout

In this exercise, you will create a *fixed* width layout [see Figure 1]. Figure 2 shows the *class* and *id* structure of the layout. You can use a grid system such as [960 Grid System](#) to create the fix width layout. The fixed width layout stays the same even if you resize the browser window. To create a fixed width layout, the width of the main element is generally specified using *pixels*.

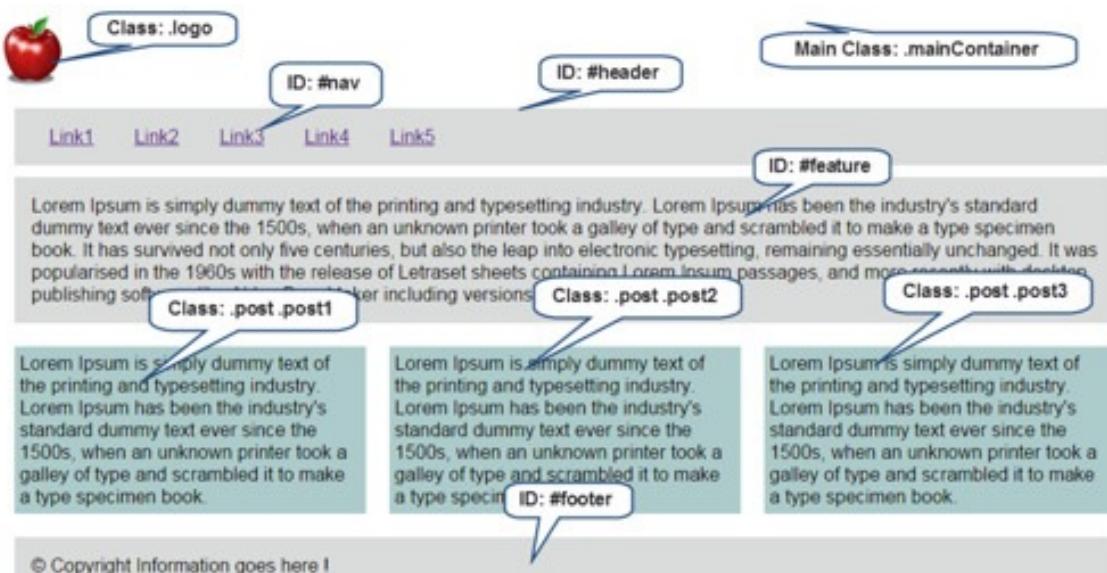
The screenshot shows a fixed-width layout with a total width of 960 pixels. It includes a header section with a logo and five navigation links (Link1, Link2, Link3, Link4, Link5). The main content area contains three columns of text. A footer section at the bottom left includes copyright information.

1
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

2
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

3
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

© Copyright Information goes here !





960 Grid System

For more information on this *Grid* system, visit the following link: <http://960.gs>.

Let's start styling the layout shown in Figure 1. Open the code editor of your choice and then create a new HTML document. Type following code and then save the html file:

```
<!DOCTYPE html>
<html>

<head>
<title> Fixed Width Layout </title>
<style>
</style>
</head>

<body>
</body>

</html>
```

This is the basic HTML template that you will be using to create the layout. Add the following code between the *<body>* and *</body>* tags to create the HTML code for the exercise:

```
<div class="mainContainer">
<div id="header">

<div id="nav">
<ul>
```

```
<li><a href="#">Link1</a> </li>
<li><a href="#">Link2</a> </li>
<li><a href="#">Link3</a> </li>
<li><a href="#">Link4</a> </li>
<li><a href="#">Link5</a> </li>
</ul>
</div>
</div>
<div id="feature">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>
</div>
<div class="post post1">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.</p>
</div>
<div class="post post2">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.</p>
</div>
<div class="post post3">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.</p>
</div>
</div>
```

```
<div id="footer">  
<p>&copy; Copyright Information goes here !</p>  
</div>
```

Next, you will *reset* the elements. You can use a reset stylesheet to get rid of the *browser inconsistencies*. It clears properties like the *default line heights*, *margins* and *font sizes of headings*, and so forth. I am using the *Reset CSS stylesheets* available at <http://meyerweb.com>.

Add the following code to the style element to *reset* the elements:

```
/* http://meyerweb.com/eric/tools/css/reset/  
v2.0 | 20110126  
License: none (public domain)  
*/
```

```
html, body, div, span, applet, object, iframe,  
h1, h2, h3, h4, h5, h6, p, blockquote, pre,  
a, abbr, acronym, address, big, cite, code,  
del, dfn, em, img, ins, kbd, q, s, samp,  
small, strike, strong, sub, sup, tt, var,  
b, u, i, center,  
dl, dt, dd, ol, ul, li,  
fieldset, form, label, legend,  
table, caption, tbody, tfoot, thead, tr, th, td,  
article, aside, canvas, details, embed,  
figure, figcaption, footer, header, hgroup,  
menu, nav, output, ruby, section, summary,  
time, mark, audio, video {  
margin: 0;  
padding: 0;
```

```
border: 0;  
font-size: 100%;  
font: inherit;  
vertical-align: baseline;  
}  
  
/* HTML5 display-role reset for older browsers */  
article, aside, details, figcaption, figure,  
footer, header, hgroup, menu, nav, section {  
display: block;  
}  
  
body {  
line-height: 1;  
}  
  
ol, ul {  
list-style: none;  
}  
  
blockquote, q {  
quotes: none;  
}  
  
blockquote:before, blockquote:after,  
q:before, q:after {  
content: '';  
content: none;  
}  
  
table {  
border-collapse: collapse;  
border-spacing: 0;  
}  
  
/* END RESET */
```

Add the following code to the `style` element:

```
body {  
    font-family: helvetica, verdana, sans-serif;  
    width: 960px;  
    margin: 0 auto;  
    line-height: 1.2;  
}
```

This code sets the fixed width of `960px` for the layout. The `margin: 0 auto;` declaration centers the content on the page. Also, this code defines `font family` and `height` of the line. Next, you will set the height of the main container to `100%` and `overflow` to `auto`.

Add the following code to the `style` element:

```
.mainContainer {  
    height: 100%;  
    overflow: auto;  
}
```

The `overflow` property lets you control the `overflowed content` that is not displayed to the user. Next, you will set the `background color` for the `nav`, `features`, and `footer` ids.

Add the following code:

```
#nav,  
#feature,  
#footer {  
    background-color: #dadbdb;  
    padding: 15px;
```

```
margin: 10px;
```

```
}
```

Figure 3 shows the result. Next, you will style the three columns.

Add the following code:

```
.post1,  
.post2,  
.post3 {  
background-color: #aecbcb;  
float: left;  
margin: 10px;  
width: 290px;  
padding: 5px;  
}
```



Link1
Link2
Link3
Link4
Link5

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

3
Copyright Information goes here !

3
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

This code floats the columns to the *left* and assigns *padding*, *margin*, and *background color* properties to them. The *width* of the box is set to *290px*. So the total width is equal to *960px*. Notice in Figure 4, the links are not *inline*. Next, you will make them *inline* and add a padding of *15px*.

Add the following code:

```
li {  
display: inline;  
padding: 15px;  
}
```

Finally, you will give some *margin* to the *logo*. Add the following rule for logo:

```
.logo {  
margin-bottom: 5px;  
margin-top: 10px;  
}
```



[Link1](#)
[Link2](#)
[Link3](#)
[Link4](#)
[Link5](#)

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

4
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

Copyright information goes here!
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

Copyright information goes here!
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

Below is the complete code for the layout:

Listing hoeht1.1: /hoeht1/hoeht1-1.html

```
<!DOCTYPE html>  
<html>
```

```
<head>
<title> Fixed Width Layout </title>
<style>
/* http://meyerweb.com/eric/tools/css/reset/
v2.0 | 20110126
License: none (public domain)
*/
html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code,
del, dfn, em, img, ins, kbd, q, s, samp,
small, strike, strong, sub, sup, tt, var,
b, u, i, center,
dl, dt, dd, ol, ul, li,
fieldset, form, label, legend,
table, caption, tbody, tfoot, thead, tr, th, td,
article, aside, canvas, details, embed,
figure, figcaption, footer, header, hgroup,
menu, nav, output, ruby, section, summary,
time, mark, audio, video {
margin: 0;
padding: 0;
border: 0;
font-size: 100%;
font: inherit;
vertical-align: baseline;
}
/* HTML5 display-role reset for older browsers */
article, aside, details, figcaption, figure,
footer, header, hgroup, menu, nav, section {
```

```
display: block;  
}  
body {  
line-height: 1;  
}  
ol, ul {  
list-style: none;  
}  
blockquote, q {  
quotes: none;  
}  
blockquote:before, blockquote:after,  
q:before, q:after {  
content: “”;  
content: none;  
}  
table {  
border-collapse: collapse;  
border-spacing: 0;  
}  
  
/* END RESET */
```

```
body {  
font-family: helvetica, verdana, sans-serif;  
width: 960px;  
margin: 0 auto;  
line-height: 1.2;  
}  
.mainContainer {  
height: 100%;
```

```
overflow: auto;  
}  
  
#nav,  
#feature,  
#footer {  
background-color: #dadbdb;  
padding: 15px;  
margin: 10px;  
}  
  
.post1,  
.post2,  
.post3 {  
background-color: #aecbcb;  
float: left;  
margin: 10px;  
width: 290px;  
padding: 5px;  
}  
  
li {  
display: inline;  
padding: 15px;  
}  
  
.logo {  
margin-bottom: 5px;  
margin-top: 10px;  
}  
  
</style>  
</head>
```

```
<body>
<div class="mainContainer">
<div id="header">

<div id="nav">
<ul>
<li><a href="#">Link1</a> </li>
<li><a href="#">Link2</a> </li>
<li><a href="#">Link3</a> </li>
<li><a href="#">Link4</a> </li>
<li><a href="#">Link5</a> </li>
</ul>
</div>
</div>
<div id="feature">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>
</div>
<div class="post post1">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.</p>
</div>
<div class="post post2">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.</p>
```

```

</div>

<div class="post post3">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
</p>
</div>
</div>

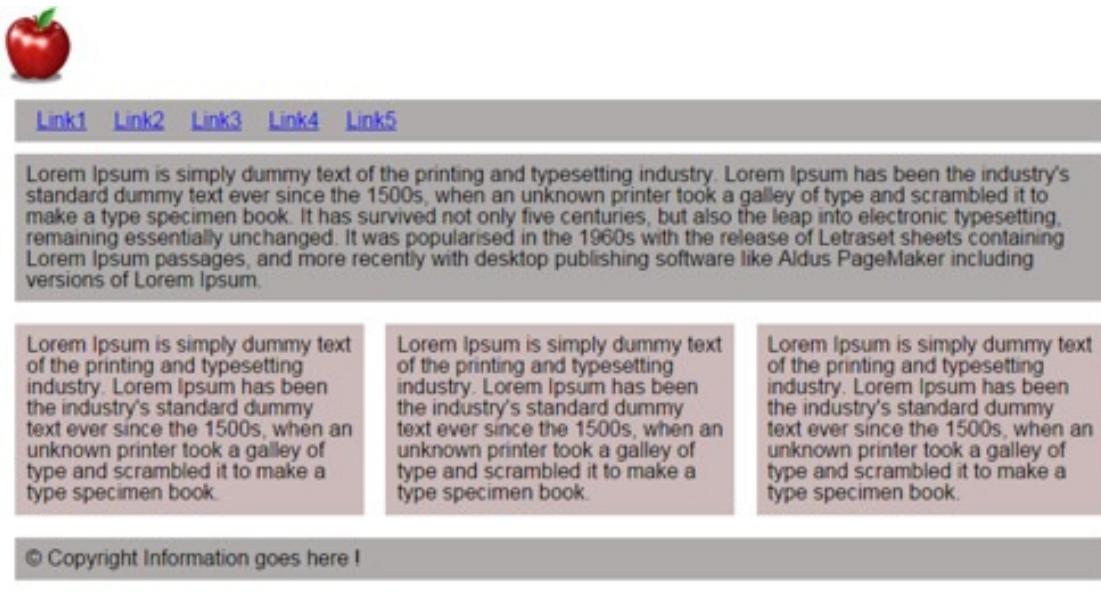
<div id="footer">
<p>&copy; Copyright Information goes here !</p>
</div>

</body></html>

```

Hands-on Exercise - 2: Creating a Liquid Layout

In this exercise, you will create a *liquid* layout, see Figure 5. In liquid layouts, the content is stretched and contracted as the user increases or decreases the size of the browser window. Generally, units are specified using *percentages* in liquid layouts.



Let's start styling the layout shown in Figure 5. Open the code editor of your choice and then create a new HTML document. Type the following code and then *save* the *html* file:

```
<!DOCTYPE html>
```

```
<html>

<head>
<title> Liquid Layout </title>
<style>
</style>
</head>

<body>

</body>

</html>
```

This is the basic HTML template that you will be using to create the layout. Add the following code between the `<body>` and `</body>` tags to create the HTML code for the exercise:

```
<div class="mainContainer">
<div id="header">

<div id="nav">
<ul>
<li><a href="#">Link1</a> </li>
<li><a href="#">Link2</a> </li>
<li><a href="#">Link3</a> </li>
<li><a href="#">Link4</a> </li>
<li><a href="#">Link5</a> </li>
</ul>
</div>
</div>
<div id="feature">
```

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

</p>

</div>

<div class="post post1">

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

</p>

</div>

<div class="post post2">

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

</p>

</div>

<div class="post post3">

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

</p>

</div>

</div>

<div id="footer">

<p>© Copyright Information goes here !</p>

</div>

This is the same code used in exercise 1. Next, you will [reset](#) the elements. Add the following code to the style element to reset the elements:

```
/* http://meyerweb.com/eric/tools/css/reset/
```

v2.0 | 20110126

License: none (public domain)

*/

```
html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code,
del, dfn, em, img, ins, kbd, q, s, samp,
small, strike, strong, sub, sup, tt, var,
b, u, i, center,
dl, dt, dd, ol, ul, li,
fieldset, form, label, legend,
table, caption, tbody, tfoot, thead, tr, th, td,
article, aside, canvas, details, embed,
figure, figcaption, footer, header, hgroup,
menu, nav, output, ruby, section, summary,
time, mark, audio, video {
margin: 0;
padding: 0;
border: 0;
font-size: 100%;
font: inherit;
vertical-align: baseline;
}
/* HTML5 display-role reset for older browsers */
article, aside, details, figcaption, figure,
footer, header, hgroup, menu, nav, section {
display: block;
}
body {
line-height: 1;
```

```
}

ol, ul {
list-style: none;
}

blockquote, q {
quotes: none;
}

blockquote:before, blockquote:after,
q:before, q:after {
content: “”;
content: none;
}

table {
border-collapse: collapse;
border-spacing: 0;
}

/* END RESET */
```

Add the following code to the `style` element:

```
body {

font-family: helvetica, verdana, sans-serif;
width: 90%;
margin: 0 auto;
}
```

This code defines width as **90%**. The remaining **10%** width creates a small gap between the **left** and **right** side of the browser window and main content of the page. The `margin: 0 auto;` declaration **centers** the content on the page.

Add the following style code:

```
.mainContainer {  
    overflow: auto;  
}  
}
```

This code sets the overflowing of the content to the *auto* mode. Next, you will specify width of the columns.

Add the following style code:

```
.post1,  
.post2,  
.post3 {  
    width: 29.3%;  
    margin: 1%;  
    padding: 1%;  
    float: left;  
    background-color: #ccbaba;  
}
```

The width of a post column is set to *31.3%* with a margin of *1%*. Therefore, the total width of the columns will be *29.3*3+1*6 (margin) + 1*6 (padding)=99.9%*. Figure 6 shows the result.



Link1
Link2
Link3
Link4
Link5

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

© Copyright Information goes here !

© Copyright Information goes here !

© Copyright Information goes here !

Add the following code to set *nav* list to *inline* display:

```
li {  
display: inline;  
padding: 0.5em;  
}
```

Add the following code to set the background colors and specify padding for *nav*, *feature*, and *footer* ids.

```
#nav,  
#footer,  
#feature {  
background-color: #afabab;  
}
```

```
#nav,  
#footer,  
#feature {  
padding: 0.5em;  
}
```

Figure 7 shows the result. Notice that you need to add some space between the *nav* and *feature* ids. Also, you need to *align* the columns with rest of the content.



[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#)

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

7
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

© Copyright Information goes here !
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

7
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

Add the following code to achieve it:

```
#nav,  
#footer,  
#feature {  
margin: 1%;  
}
```

Now, try to resize the browser, the content within will stretch or contract as you resize the browser. Figures 8 and 9 show layout at different browser resolutions.



[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#)

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

8
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

9
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

10
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.



[Link1](#) [Link2](#) [Link3](#) [Link4](#) [Link5](#)

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

11
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

12
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

13
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

© Copyright Information goes here !

Below is the complete code for the exercise:

Listing hoeht1.2: /hoeht1/hoeht1-2.html

```
<!DOCTYPE html>
<html>

<head>
<title> Liquid Layout </title>
<style>
/* http://meyerweb.com/eric/tools/css/reset/
v2.0 | 20110126
License: none (public domain)
```

*/

html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code,
del, dfn, em, img, ins, kbd, q, s, samp,
small, strike, strong, sub, sup, tt, var,
b, u, i, center,
dl, dt, dd, ol, ul, li,
fieldset, form, label, legend,
table, caption, tbody, tfoot, thead, tr, th, td,
article, aside, canvas, details, embed,
figure, figcaption, footer, header, hgroup,
menu, nav, output, ruby, section, summary,
time, mark, audio, video {
margin: 0;
padding: 0;
border: 0;
font-size: 100%;
font: inherit;
vertical-align: baseline;
}
/* HTML5 display-role reset for older browsers */
article, aside, details, figcaption, figure,
footer, header, hgroup, menu, nav, section {
display: block;
}
body {
line-height: 1;
}
ol, ul {

```
list-style: none;  
}  
  
blockquote, q {  
quotes: none;  
}  
  
blockquote:before, blockquote:after,  
q:before, q:after {  
content: “”;  
content: none;  
}  
  
table {  
border-collapse: collapse;  
border-spacing: 0;  
}
```

```
/* END RESET */
```

```
body {  
font-family: helvetica, verdana, sans-serif;  
width: 90%;  
margin: 0 auto;  
}  
  
.mainContainer {  
overflow: auto;  
}  
  
.post1,  
.post2,  
.post3 {  
width: 29.3%;  
margin: 1%;  
padding: 1%;  
float: left;
```

```
background-color: #ccbaba;
}
li {
display: inline;
padding: 0.5em;
}
#nav,
#footer,
#feature {
background-color: #afabab;
}
#nav,
#footer,
#feature {
padding: 0.5em;
}
#nav,
#footer,
#feature {
margin: 1%;
}
</style>
</head>

<body>
<div class="mainContainer">
<div id="header">

<div id="nav">
<ul>
<li><a href="#">Link1</a> </li>
```

```
<li><a href="#">Link2</a> </li>
<li><a href="#">Link3</a> </li>
<li><a href="#">Link4</a> </li>
<li><a href="#">Link5</a> </li>
</ul>
</div>
</div>
<div id="feature">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.
</p>
</div>
<div class="post post1">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
</p>
</div>
<div class="post post2">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
</p>
</div>
<div class="post post3">
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
</p>
</div>
</div>
```

```
<div id="footer">  
<p>&copy; Copyright Information goes here !</p>  
</div>  
</body>  
  
</html>
```

Hands-on Exercise - 3: Creating a 3-Column Fixed Width Layout

In this exercise, you will create a *3-column fixed* width layout using semantic markup [see Figure 10].

In this exercise, I will place padding only on the *top* and *bottom* edges of the *block elements*. Avoiding padding on the left and right edges of the block elements prevents any redoing the *block level mathematics* because if you place padding on the left and/or right edges of the block elements the width of the block will be affected. Therefore, if you want to add margins to those block elements, place margin on the elements they hosts.

Wow!

- Link one
- Link two
- Link three
- Link four

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

Lorem Ipsum

What is Lorem Ipsum?



Caption-2

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book, it has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Lorem Ipsum

Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit...

Lorem Ipsum

Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit...

Where does it come from?



Caption-2

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance.

The first line of Lorem Ipsum, "Lorem ipsum dolor sit amet..", comes from a line in section 1.10.32.

Where do we use it?

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using 'Content here, content here', making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for 'lorem ipsum' will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).

10

(C) Wow Inc. Planet ZeroX, Universe

Create a new document and insert the following code in the document:

```
<!doctype html>
<html>

<head>
<title> Wow Inc </title>
<style type="text/css">

</style>
</head>

<body>
```

```
<div class="wrapper">
<header>

<a href="#">
</a>

</header>
<div class="sidebar1">
<nav>
<ul>
<li><a href="#">Link one</a>
</li>
<li><a href="#">Link two</a>
</li>
<li><a href="#">Link three</a>
</li>
<li><a href="#">Link four</a>
</li>
</ul>
</nav>
<aside>
<p> Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
<p> Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
</aside>
<!-- end .sidebar1 -->
</div>
<article class="content">
<h1>Lorem Ipsum</h1>
<section>
<h2>What is Lorem Ipsum?</h2>
```

```
<hgroup>
<figure> 
</figure>
<figcaption>Caption-2</figcaption>
</hgroup>

<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>
</section>

<section>
<h2>Where does it come from?</h2>
<hgroup>
<figure> 
</figure>
<figcaption>Caption-2</figcaption>
</hgroup>

<p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of “de Finibus Bonorum et Malorum” (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, “Lorem ipsum dolor sit amet..”, comes from a line in section 1.10.32.
</p>
</section>

<section>
<h2>Where do we use it?</h2>
<p>It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using ‘Content here, content here’, making it look like readable English. Many desktop publishing packages and web page
```

editors now use Lorem Ipsum as their default model text, and a search for ‘lorem ipsum’ will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).
</p>

</section>

<!-- end .content -->

</article>

<aside>

<h3>Lorem Ipsum</h3>

<p>Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit...</p>

</aside>

<aside>

<h3>Lorem Ipsum</h3>

<p>Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit...</p>

</aside>

<footer>

<address>

(C) Wow Inc.

Planet Zeroxi, Universe

</address>

</footer>

<!-- end .wrapper -->

</div>

</body>

</html>

Following is the HTML5 structure used to hold the page:

```
<div class="wrapper">
  <!-- The wrapper class wraps whole webpage -->
  <header>
    <!-- Contains the logo and Header. Total width of the page is 960px out of which 200px
        is reserved for the logo. The height of the header section is 150px. -->

  </header>
  <div class="sidebar1">
    <nav>
      <!-- It is the sidebar [width 200px] that contains an unordered list for
          navigation link. -->
    </nav>
    <aside>
      <!-- Holds two paragraphs -->
    </aside>
    <!-- end .sidebar1 -->
  </div>
  <article class="content">
    <!-- Holds main content of the page [width=560px] -->
    <section>
      <!-- Holds Post 1 -->
    </section>
    <section>
      <!-- Holds Post 2 -->
    </section>
    <section>
      <!-- Holds Post 3 -->
    </section>
    <!-- end .content -->
```

```
</article>

<aside>

<!-- Holds right column. --&gt;

&lt;/aside&gt;

&lt;footer&gt;

<!-- Holds footer --&gt;

<!-- end .wrapper --&gt;

&lt;/div&gt;</pre>
```

Let's start styling the page. Add the following code between the `<style>` and `</style>` tags to set font size, line height, font family, padding, margin, color of the text, and background color.

```
body {

font: 100%/1.2 Verdana, helvetica, sans-serif;

background-color: lightgray;

margin: 0;

padding: 0;

color: black;

}
```

Add the following code to remove padding and margins from the list. Due to variations in how browsers handles lists, it is best practice to get rid of the margins and paddings from the lists.

```
ul {

padding: 0;

margin: 0;

}
```

Add the following code to remove top margins from the headings and paragraphs. It fixes the issue of margin, escaping from the block for these elements.

```
h1,  
h2,  
h3,  
p {  
margin-top: 0;  
}
```

Add the following style to get rid of the blue borders that some browsers show around images:

```
a img {  
border: none;  
}
```

Add the following style to the *wrapper* class. These rules set the width of the page to *960px* and also center the content on the browser.

```
.wrapper {  
width: 960px;  
background-color: #ffffff;  
margin: 0 auto;  
}
```

Add the following code to define the background color and height of the header:

```
header {  
background-color: #d1eb4a;  
height: 150px;  
}
```

Add the following code for the `a` element that hosts the `img` tag for the logo:

```
header a {  
background-color: #ebebea;  
width: 200px;  
height: 100%;  
display: block;  
}
```

Add the following code to set the width and height of the logo. The `display: inline-block;`

`margin: 0 22px;` rules horizontally aligns the logo on it container.

```
header img {  
width: 150px;  
height: 150px;  
display: inline-block;  
margin: 0 22px;  
}
```

Add the following code to the `style` tag to assign sidebar a width of `200px` and float it to `left`.

```
.sidebar1 {  
float: left;
```

```
width: 200px;  
background-color: #5f450c;  
}
```

Add the following code to set width of the content, float it to the left, and assign padding of *10px* at the top and bottom edges of the container:

```
.content {  
padding: 10px 0;  
width: 560px;  
float: left;  
}
```

Notice that I did not assign the left and right padding for the content block. I will assign left and right paddings to the elements inside the blocks. Add the following code to add spaces between the blocks and element inside them:

```
h1,  
h2,  
h3,  
h4,  
p,  
figure,  
figcaption {  
padding-left: 15px;  
padding-right: 15px;  
}
```

Add the following code for the *aside* element to set its width to *200px*, float it to the left, assign padding of *10px* at the top and bottom edges, and *1px* bottom border:

```
aside {  
    float: left;  
    width: 200px;  
    background-color: #eadcae;  
    padding: 10px 0;  
    border-bottom: 1px solid #31302f;  
}
```

Add following code to select the *ul* element and get rid of the bullets:

```
nav ul {  
    list-style: none;  
    margin-bottom: 15px;  
}
```

Add the following code to set padding of *5px* for links, and set *display* as *block*.

```
nav a {  
    display: block;  
    padding: 5px 5px 5px 15px;  
    width: 180px;  
    text-decoration: none;  
    background-color: #a7c329;  
}
```

Add following code to add a bottom border to the links:

```
li {
```

```
border-bottom: 1px solid red;  
}
```

Add the following code to change the background color of the links when user hovers the mouse pointer over them:

```
nav a:hover {  
background-color: #ADB96E;  
color: #fff;  
}
```

Add following code to style the content of the footer:

```
footer {  
padding: 10px 0;  
background-color: #ccc49f;  
clear: both;  
}
```

```
footer address {  
padding-left: 15px;  
}
```

The ***clear: both*** rule makes sure that no floating element is allowed on either the left or right side. This property forces the footer element to sit at the bottom of the page.

Add the following code to style the ***figure*** and ***figcaption*** elements. The ***hrgroup*** tag contains ***figure*** and ***figcaption*** elements. The ***hrgroup*** tag is floated left.

```
hgroup {
```

```
float: left;  
}  
  
figure > img {  
width: 100px;  
height: 100px;  
}
```

```
figcaption {  
font-size: 90%;  
text-align: center;  
font-weight: bold;  
}
```

Below is the complete code for the exercise:

Listing hoeht1.3: /hoeht1/hoeht1-3.html

```
-<!doctype html>  
<html>  
  
<head>  
<title> Wow Inc </title>  
<style type="text/css">  
body {  
font: 100%/1.2 Verdana, helvetica, sans-serif;  
background-color: lightgray;  
margin: 0;  
padding: 0;  
color: black;  
}
```

```
ul {
padding: 0;
margin: 0;
}

h1,
h2,
h3,
p {
margin-top: 0;
}

a img {
border: none;
}

.wrapper {
width: 960px;
background-color: #ffffff;
margin: 0 auto;
}

header {
background-color: #d1eb4a;
height: 150px;
}

header a {
background-color: #ebebea;
width: 200px;
height: 100%;
display: block;
}

header img {
width: 150px;
height: 150px;
```

```
display: inline-block;
margin: 0 22px;
}
.sidebar1 {
float: left;
width: 200px;
background-color: #5f450c;
}
.content {
padding: 10px 0;
width: 560px;
float: left;
}
aside {
float: left;
width: 200px;
background-color: #eadcae;
padding: 10px 0;
border-bottom: 1px solid #31302f;
}
footer address {
padding-left: 15px;
}
nav ul {
list-style: none;
margin-bottom: 15px;
}
li {
border-bottom: 1px solid red;
}
nav a {
```

```
display: block;  
padding: 5px 5px 5px 15px;  
width: 180px;  
text-decoration: none;  
background-color: #a7c329;  
}  
  
nav a:hover {  
background-color: #ADB96E;  
color: #fff;  
}  
  
footer {  
padding: 10px 0;  
background-color: #ccc49f;  
clear: both;  
}  
  
h1,  
h2,  
h3,  
h4,  
  
p,  
figure,  
  
figcaption {  
padding-left: 15px;  
padding-right: 15px;  
}  
  
figure > img {  
width: 100px;  
height: 100px;  
}  
  
hgroup {  
float: left;
```

```
}

figcaption {
font-size: 90%;
text-align: center;
font-weight: bold;
}

</style>
</head>

<body>

<div class="wrapper">
<header>

<a href="#">
</a>

</header>
<div class="sidebar1">
<nav>
<ul>
<li><a href="#">Link one</a>
</li>
<li><a href="#">Link two</a>
</li>
<li><a href="#">Link three</a>
</li>
<li><a href="#">Link four</a>
</li>
</ul>
</nav>
```

```
<aside>
<p> Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
<p> Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
</aside>
<!-- end .sidebar1 -->
</div>

<article class="content">
<h1>Lorem Ipsum</h1>
<section>
<h2>What is Lorem Ipsum?</h2>
<hgroup>
<figure> 
</figure>
<figcaption>Caption-2</figcaption>
</hgroup>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>
</section>
<section>
<h2>Where does it come from?</h2>
<hgroup>
<figure> 
</figure>
<figcaption>Caption-2</figcaption>
</hgroup>
<p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going
```

through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of “de Finibus Bonorum et Malorum” (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, “Lorem ipsum dolor sit amet..”, comes from a line in section 1.10.32.

</p>

</section>

<section>

<h2>Where do we use it?</h2>

<p>It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using ‘Content here, content here’, making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for ‘lorem ipsum’ will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).

</p>

</section>

<!— end .content —>

</article>

<aside>

<h3>Lorem Ipsum</h3>

<p>Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit...</p>

</aside>

<aside>

<h3>Lorem Ipsum</h3>

<p>Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit...</p>

</aside>

<footer>

<address>

(C) Wow Inc.

Planet Zeroxi, Universe

</address>

```
</footer>  
<!-- end .wrapper -->  
</div>  
</body></html>
```



Nav on the Right

If you want to place `nav` on the right of the page, simply float the elements to the right and browser will render it for you in reverse order. Here's the code:

```
.sidebar {  
    float: right;  
    width: 200px;  
    background-color: #5f450c;  
}  
.content {  
    padding: 10px 0;  
    width: 560px;  
    float: right;  
}  
aside {  
    float: right;  
    width: 200px;  
    background-color: #eadcae;  
    padding: 10px 0;  
    border-bottom: 1px solid #31302f;  
}
```

Hands-on Exercise - 4: Creating a 2-Column Fixed Width Layout

In this exercise, you will create a two columns layout [see Figure 11]. You will also use a [Google Font](#) to style the header of the page.

LOREM IMPSUM

[Link \[1\]](#) [Link \[2\]](#) [Link \[3\]](#) [Link \[4\]](#) [Link \[5\]](#)

Links - I

[Other Link - 1](#)
[Other Link - 1](#)

Links - II

[Other Link - 1](#)
[Other Link - 1](#)

What is lorem ipsum?

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Where does it come from?

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, *consectetur*, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, "Lorem ipsum dolor sit amet..", comes from a line in section 1.10.32.

11

(c) 2015 Copyright information goes here !

I will be using the following skeleton to place elements of the page:

```
<div id="wrapper">  
  <header>  
  </header>  
  <nav>  
  </nav>  
  <aside id="sidebar">  
    <section class="sidesec">  
    </section>  
    <section class="sidesec">  
    </section>  
  </aside>  
  <section>  
    <article>
```

```
</article>
```

```
<article>
```

```
</article>
```

```
</section>
```

```
<footer>
```

```
</footer>
```

```
</div>
```

Here's the html of the page:

```
<!Doctype html>
```

```
<html>
```

```
<head>
```

```
<title> Two Column Layout </title>
```

```
<style type="text/css">
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div id="wrapper">
```

```
<header>
```

```
<!--Place Header Image here -->
```

```
<h1> Lorem Impsum</h1>
```

```
</header>
```

```
<nav>
```

```
<ul>
```

```
<li><a href="#" title=""> Link [1]</a>
```

```
</li>
```

```
<li><a href="#" title=""> Link [2]</a>
</li>
<li><a href="#" title=""> Link [3]</a>
</li>
<li><a href="#" title="">Link [4]</a>
</li>
<li><a href="#" title="">Link [5]</a>
</li>
</ul>
</nav>
<aside id="sidebar">
<section class="sidesec">
<p class="sdeco">Links - I</p>
<a href="#" title="">Other Link - 1</a>
<br>
<a href="#" title="">Other Link - 1</a>
<br>
</section>
<section class="sidesec">
<p class="sdeco">Links - II</p>
<a href="#" title="">Other Link - 1</a>
<br>
<a href="#" title="">Other Link - 1</a>
<br>
</section>
</aside>

<section>
<article>
<p class="artdeco"> What is lorem ipsum?</p>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an
```

unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>

</article>

<article>

<p class="artdeco"> Where does it come from?</p>

<p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of “de Finibus Bonorum et Malorum” (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, “Lorem ipsum dolor sit amet..”, comes from a line in section 1.10.32.</p>

</article>

</section>

<footer>

(c) 2015 Copyright information goes here !

</footer>

</div>

</body>

</html>

Render the code in the browser to see the result.

Let's start styling the page. We will first import a *Google Font* to create header text for the page. Navigate to the following link: <https://www.google.com/fonts>. Find the font “*Banger*” and then click on the *Quick Use* button. Scroll down to step 3 and choose the *@import* panel. Copy

the code and then place it between the *style* tags. It should be the first entry just after the opening style tag.

```
@import url(https://fonts.googleapis.com/css?family=Bangers);
```

Add the following code to set the width of the wrapper and center it on the page:

```
#wrapper {  
width: 800px;  
margin: 20px auto;  
margin-top: 10px;  
}
```

Add the following code to set the background color and font size of the page:

```
body {  
color: #3a3a3a;  
background-color: #6fabd1;  
font-family: verdana, sans-serif;  
font-size: 14px;  
line-height: 1.4em;
```

Add the following code to set the *header* properties:

```
header {  
width: 100%;  
height: 110px;  
margin-bottom: 10px;  
background-color: lightblue;  
border-radius: 10px;
```

```
}
```

Add the following code to style the *header* text:

```
h1 {  
    font-family: 'Bangers', cursive;  
    font-size: 80px;  
    display: inline-block;  
    margin: 42px 180px;  
}
```

The header text is styled using the *Bangers* font that we imported when started styling the page. The last two rules of the declaration block *center* the text on the header.

Add the following code to style the *nav* element:

```
nav {  
    background-color: lightblue;  
    height: 30px;  
    border-radius: 10px;  
    margin: 5px 0px;  
}  
  
nav ul {  
    margin: 0px;  
    padding: 3px 0px 5px 30px;  
    list-style-type: none;  
}  
  
nav li {  
    display: inline;  
    margin-right: 10px;  
}  
  
nav li a {
```

```
color: #272424;  
}  
  
nav li a:hover {  
color: #dd0b0b;  
}
```

Add the following code to set the width of the *sidebar* and float it to the left:

```
#sidebar {  
float: left;  
width: 170px;  
}
```

Add the following code to style elements of the *sidebar*:

```
.sidesec {  
width: 164px;  
background-color: azure;  
height: 100%;  
border-radius: 10px;  
margin-top: 5px;  
margin-bottom: 5px;  
padding: 5px;  
}
```

Add the following code to decorate the headings of the sidebar elements:

```
p.sdeco {  
border-top: 1px solid #6a6a6a;  
border-bottom: 1px solid #6a6a6a;  
padding: 2px 2px;
```

```
}
```

Add the following code to define the main section of the page:

```
section {  
    width: 630px;  
    float: left;  
}
```

Add the following code to define the articles in the main section:

```
article {  
    width: 615px;  
    background-color: azure;  
    height: 100%;  
    margin: 5px 0px 5px 15px;  
    border-radius: 10px;  
    float: left;  
}
```

Add the following code to place padding for the *p* elements inside the *article* elements:

```
article p {  
    padding: 0px 10px;  
}
```

Add the following code to decorate the *article* headings:

```
p.artdeco {  
    border-bottom: 1px solid black;
```

```
padding: 2px 2px;  
margin: 10px 10px 0px 10px;  
font-weight: bold;  
}
```

Finally, add the following code to style the *footer*:

```
footer {  
clear: both;  
background-color: lightblue;  
border-radius: 10px;  
padding: 6px 0px 6px 15px;  
}
```

Below is the complete code for the exercise:

Listing hoeht1.4: /hoeht1/hoeht1-4.html

```
-><!Doctype html>  
<html>  
  
<head>  
<title> Two Column Layout </title>  
<style type="text/css">  
@import url(https://fonts.googleapis.com/css?family=Bangers);  
#wrapper {  
width: 800px;  
margin: 20px auto;  
margin-top: 10px;  
}  
body {  
color: #3a3a3a;
```

```
background-color: #6fabd1;  
font-family: verdana, sans-serif;  
font-size: 14px;  
line-height: 1.4em;  
}  
  
header {  
width: 100%;  
height: 110px;  
margin-bottom: 10px;  
background-color: lightblue;  
border-radius: 10px;  
}  
  
h1 {  
font-family: ‘Bangers’, cursive;  
font-size: 80px;  
display: inline-block;  
margin: 42px 180px;  
}  
  
nav {  
background-color: lightblue;  
height: 30px;  
border-radius: 10px;  
margin: 5px 0px;  
}  
  
nav ul {  
margin: 0px;  
padding: 3px 0px 5px 30px;  
list-style-type: none;  
}  
  
nav li {  
display: inline;
```

```
margin-right: 10px;  
}  
  
nav li a {  
color: #272424;  
}  
  
nav li a:hover {  
color: #dd0b0b;  
}  
  
#sidebar {  
float: left;  
width: 170px;  
}  
  
.sidesec {  
width: 164px;  
background-color: azure;  
height: 100%;  
border-radius: 10px;  
margin-top: 5px;  
margin-bottom: 5px;  
padding: 5px;  
}  
  
p.sdeco {  
border-top: 1px solid #6a6a6a;  
border-bottom: 1px solid #6a6a6a;  
padding: 2px 2px;  
}  
  
section {  
width: 630px;  
float: left;  
}  
  
article {
```

```
width: 615px;  
background-color: azure;  
height: 100%;  
margin: 5px 0px 5px 15px;  
border-radius: 10px;  
float: left;  
}  
  
article p {  
padding: 0px 10px;  
}  
  
p.artdeco {  
border-bottom: 1px solid black;  
padding: 2px 2px;  
margin: 10px 10px 0px 10px;  
font-weight: bold;  
}  
  
footer {  
clear: both;  
background-color: lightblue;  
border-radius: 10px;  
padding: 6px 0px 6px 15px;  
}  
</style>  
</head>
```

```
<body>  
<div id="wrapper">  
<header>  
<!--Place Header Image here -->  
<h1> Lorem Impsum</h1>  
</header>
```

```
<nav>
<ul>
<li><a href="#" title=""> Link [1]</a>
</li>
<li><a href="#" title=""> Link [2]</a>
</li>
<li><a href="#" title=""> Link [3]</a>
</li>
<li><a href="#" title="">Link [4]</a>
</li>
<li><a href="#" title="">Link [5]</a>
</li>
</ul>
</nav>

<aside id="sidebar">
<section class="sidesec">
<p class="sdeco">Links - I</p>
<a href="#" title="">Other Link - 1</a>
<br>
<a href="#" title="">Other Link - 1</a>
<br>
</section>
<section class="sidesec">
<p class="sdeco">Links - II</p>
<a href="#" title="">Other Link - 1</a>
<br>
<a href="#" title="">Other Link - 1</a>
<br>
</section>
</aside>
```

```
<section>
<article>
<p class="artdeco"> What is lorem ipsum?</p>
<p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.</p>
</article>
<article>
<p class="artdeco"> Where does it come from?</p>
<p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintonck, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of “de Finibus Bonorum et Malorum” (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, “Lorem ipsum dolor sit amet..”, comes from a line in section 1.10.32.
</p>
</article>

</section>
<footer>
(c) 2015 Copyright information goes here !
</footer>
</div>
</body>

</html>
```


Appendix CSA: Evaluation Test

The purpose of the evaluation test is to *evaluate the skills* you have gained during this course. Attempt questions and send your answers to me, I will *check* your answers and award you *grade* and a certificate based on the marks you have earned. The following table shows the grade system.

Grade System	
Grade	Percentage Range
S	91-100
A	81-90
B	71-80
C	61-70
D	57-60
E	50-56
U	< 50

If you score “S”, that’s [S]upper awesome. If you score “U”, you have to reappear.

After completing the test, send the html files to me. Use the following format for naming the html files: *q1-ans.html*, *q2-ans.html*, and so forth. Compress the files in a *zip* file and send it to the following email address: teamhlmleceous@gmail.com.

I will check your answers and issue you a *Certification of Completion* if you score any grade between *S* through *E*. If you score “*U*”, you have to *reappear*

for the test.

Following is a sample certificate:



Question - 1 [5 Marks]

Using the Listing q1, set the background color of the *visited links* to *blue*. Set the *background color* of unvisited links to *lightcoral*. Also, set the *background color* of the *hover* and *active* links to *yellow*. Group the selectors to *minimize* the code.

Listing q1: /set/q-1.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
.lorem {
```

```
padding: 15px;
```

```
border: 1px dashed;
```

```
}
```

```
</style>
```

```
</head>

<body>
<h1>Lorem Ipsum</h1>
<div class="lorem">
<p>
    Lorem Ipsum is simply dummy text of the printing and <a href="https://www.google.co.in/search?q=typesetting+industry">typesetting industry</a>. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. To know more <a href="example.com">click here</a>.
</p>
</div>
</body>

</html>
```

Question - 2 [5 Marks]

Using Listing q2, hide the *h1* element so that it takes the same space what it was taking as before.

Listing q2: /set/q-2.html

```
<!DOCTYPE html>
<html>

<head>
<style>
</style>
```

```
</head>

<body>
<h1> Lorem Ipsum</h1>
<ul>
<li>Lorem</li>
<li>Ipsum</li>
<li>Lorem Ipsum</li>
</ul>
</body>

</html>
```

Question - 3 [5 Marks]

Using Listing q3, set the *indent* the *first line* by *25* pixels.

Listing q3: /set/q-3.html

```
<!DOCTYPE html>
<html>

<head>
<style>
.lorem {
padding: 15px;
border: 1px dashed;
}
</style>
```

```
</head>

<body>
<h1>Lorem Ipsum</h1>
<div class="lorem">
<p>
    Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. To know more <a href="example.com">Click Here</a>.
</p>
</div>
</body>

</html>
```

Question - 4 [5 Marks]

Using Listing q4, hide the *h1* element so that it does not consume any space on the page.

Listing q4: /set/q-4.html

```
<!DOCTYPE html>
<html>

<head>
<style>
</style>

</head>
```

```
<body>
<h1> Lorem Ipsum</h1>
<ul>
<li>Lorem</li>
<li>Ipsum</li>
<li>Lorem Ipsum</li>
</ul>
</body>

</html>
```

Question - 5 [5 Marks]

Using the Listing q5, set the *list style* for the *unordered list* to *square*. Also, set *list style* to *lower roman* for ordered list.

Listing q5: /set/q-5.html

```
<!DOCTYPE html>
<html>

<head>
<style>
</style>

</head>
```

```
<body>
Unorderd List:
<ul>
<li>Lorem</li>
<li>Ipsum</li>
```

```
<li>Lorem Ipsum</li>
</ul>
```

Ordered List:

```
<ul>
<li>Lorem</li>
<li>Ipsum</li>
<li>Lorem Ipsum</li>
</ul>
```

```
</body>
```

```
</html>
```

Question - 6 [5 Marks]

Using listing q6, and *margin* property, set the *top* and *bottom* margins of the *<h1>* element to *20 px*. Also, set left and right margins to *15px*. Use the *shorthand margin* property.

Listing q6: /set/q-6.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="lorem">
```

```
<h1> Lorem Ipsum</h1>
```

```
<p>
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

```
</p>
```

```
</div>
```

```
</body>
```

```
</html>
```

Question - 7 [5 Marks]

Using Listing q7, set the list marker to *star.png*. Ensure that the marker is displayed *inside* the content.

Listing q7: /set/q-7.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<ul>
```

```
<li>Lorem</li>
```

```
<li>Ipsum</li>
```

```
<li>Lorem Ipsum</li>
```

```
</ul>
```

```
</body>
```

```
</html>
```

Question - 8 [5 Marks]

Using Listing q8, *center align* the ***h1*** element on the page.

Listing q8: /set/q-8.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
.lorem {
```

```
padding: 15px;
```

```
border: 1px dashed;
```

```
background-color: bisque;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>Lorem Ipsum</h1>
```

```
<div class="lorem">
```

```
<p>
```

 Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

```
</p>
```

```
</div>
```

```
</body>
```

```
</html>
```

Question - 9 [5 Marks]

Using Listing 9, set the color of the *p* element to *blue*. Also, make the *text centrally aligned*.

Listing q9: /set/q-9.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
.lorem {
```

```
padding: 15px;
```

```
border: 1px dashed;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="lorem">
```

```
<p>
```

 Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

```
</p>
```

```
</div>
```

```
</body>
```

```
</html>
```

Question - 10 [5 Marks]

Using the Listing q10, create three classes [*big*, *medium*, and *small*] for the *img* element. Set the *height* to *auto*, *100px*, and *25px* for the *big*, *medium*, and *small* classes, respectively.

Listing q10: /set/q-10.html

```
<!DOCTYPE html>
<html>

<head>
<style>
</style>
</head>

<body>
<div class="lorem">
<h1> Lorem Ipsum</h1>
<p>Big</p>

<p>Normal</p>

<p>Small</p>

<p>
```

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

```
</p>
</div>
</body></html>
```

Question - 11 [5 Marks]

Using the listing q11, display the list item as *inline* elements.

Listing q11: /set/q-11.html

```
<!DOCTYPE html>
<html>
```

```
<head>
<style>
</style>
```

```
</head>
```

```
<body>
```

```
<ul>
<li>Lorem</li>
<li>Ipsum</li>
<li>Lorem Ipsum</li>
</ul>
</body>
```

```
</html>
```

Question - 12 [5 Marks]

Using Listing q12, remove the *underline* from the *click here* link.

Listing q12: /set/q-12.html

```
<!DOCTYPE html>
<html>

<head>
<style>
.lorem {
padding: 15px;
border: 1px dashed;
}
</style>
</head>

<body>
<h1>Lorem Ipsum</h1>
<div class="lorem">
<p>
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. To know more <a href="example.com"> Click Here </a>
</p>
</div>
</body>

</html>
```

Question - 13 [5 Marks]

Using Listing q13, change the text in the *h1* element to *upper case*.

Listing q13: /set/q-13.html

```
<!DOCTYPE html>
<html>

<head>
<style>
.lorem {
padding: 15px;
border: 1px dashed;
}
</style>
</head>

<body>
<h1>Lorem Ipsum</h1>
<div class="lorem">
<p>
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. To know more <a href="example.com"> Click Here </a>
</p>
</div>
</body>

</html>
```

Question - 14 [5 Marks]

Using Listing q14, float the *orange-star.png* to the right of the *p* element. Set *image border* to *1px*, *solid*, and *blue* in color. Also, set the *left* and *bottom* margin to *15px* and *10px*, respectively. Also, *justify* the text inside the *paragraph*.

Listing q14: /set/q-14.html

```
<!DOCTYPE html>
<html>

<head>
<style>
</style>
</head>

<body>
<div class="lorem">
<p> Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

</p>
</div>
</body>

</html>
```

Question - 15 [5 Marks]

Using the Listing q15, *hide* the *empty cells* from the table so that *border* is not displayed around them.

Listing q15: /set/q-15.html

```
<!DOCTYPE html>
<html>

<head>
<style>
th,
td {
padding: 5px;
}
td {
border: 1px solid;
}
</style>
</head>

<body>

<h1> Order Details</h1>
<table>
<tr>
<th>Order No</th>
<th>Product Name</th>
<th>Price</th>
<th>Quantity</th>
</tr>
<tr>
```

```
<td>I0001</td>
<td>Intel CPU</td>
<td>700</td>
<td>1</td>
</tr>
<tr>
<td>I0002</td>
<td>HDD 10TB</td>
<td>550</td>
<td>3</td>
</tr>
<tr>
<td>I0003</td>
<td>Samsung LCD</td>
<td>5000</td>
<td>10</td>
</tr>
<tr>
<td>I0004</td>
<td>Stylus</td>
<td></td>
<td></td>
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

Question - 16 [5 Marks]

Using the Listing q16 and *font* property, *italicize* the text in the *paragraph*. Also, set *font size* to *15px* and font to *Times New Roman*.

Listing q16: /set/q-16.html

```
<!DOCTYPE html>
<html>

<head>
<style>
.lorem {
padding: 15px;
border: 1px dashed;
}
</style>

</head>

<body>
<h1>Lorem Ipsum</h1>
<div class="lorem">
<p>
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. To know more <a href="example.com">Click Here</a>.
</p>
</div>
</body>
```

```
</html>
```

Question - 17 [5 Marks]

Using Listing q17, set border of *p* element to *3px*. Also, make *border dotted*.

Listing q17: /set/q-17.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="lorem">
```

```
<p>
```

 Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

```
</p>
```

```
</div>
```

```
</body>
```

```
</html>
```

Question - 18 [5 Marks]

Using Listing q18, break the *border shorthand* property into *three* individual

properties so that they show only bottom *border* of the *p* element.

Listing q18: /set/q-18.html

```
<!DOCTYPE html>
<html>

<head>
<style>
p {
border: 3px dotted blue;
}
</style>
</head>

<body>
<div class="lorem">
<p>
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

</p>
</div>
</body>

</html>
```

Question - 19 [5 Marks]

Using the Listing q19, *float* the image inside the *div* element to the *right*. Set image border to *1px, dotted*, and *blue* in color. Also, set the *left* and *bottom*

margins to `15px` and `10px`, respectively. Also, *justify* the text inside the *paragraph*.

Listing q19: /set/q-19.html

```
<!DOCTYPE html>
<html>

<head>
<style>
</style>
</head>

<body>
<div class="star">

<br> The Orange Star
</div>

<p> Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.
</p>

</body>

</html>
```

Question - 20 [5 Marks]

Using the Listing q20, set the *maximum width* of the *p* element to *30%* and background color of the element to *red* and foreground color to *yellow*.

Listing q20: /set/q-20.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="lorem">
```

```
<p>
```

 Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

```
</p>
```

```
</div>
```

```
</body>
```

```
</html>
```

Question - 21 [5 Marks]

Using q21, set change the color of *h1* and *p* elements to *red*. *Group* the selectors to minimize the code.

Listing q21: /set/q-21.html

```
<!DOCTYPE html>
<html>

<head>
<style>
.lorem {
padding: 15px;
border: 1px dashed;
}
</style>
</head>

<body>
<h1>Lorem Ipsum</h1>
<div class="lorem">
<p>
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

</p>
</div>
</body>

</html>
```

Question - 22 [5 Marks]

Using the Listing q22, get rid of the *space* around the *border* of the cells.

Listing q22: /set/q-22.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
th,
```

```
td {
```

```
padding: 5px;
```

```
}
```

```
td {
```

```
border: 1px solid;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1> Order Details</h1>
```

```
<table>
```

```
<tr>
```

```
<th>Order No</th>
```

```
<th>Product Name</th>
```

```
<th>Price</th>
```

```
<th>Quantity</th>
```

```
</tr>
```

```
<tr>
```

```
<td>I0001</td>
```

```
<td>Intel CPU</td>
```

```
<td>700</td>
```

```
<td>1</td>
</tr>
<tr>
<td>I0002</td>
<td>HDD 10TB</td>
<td>550</td>
<td>3</td>
</tr>
<tr>
<td>I0003</td>
<td>Samsung LCD</td>
<td>5000</td>
<td>10</td>
</tr>
<tr>
<td>I0004</td>
<td>Stylus</td>
<td></td>
<td></td>
</tr>

</table>
```

```
</body>
```

```
</html>
```

Question - 23 [5 Marks]

Using Listing q23, change the *background color* of the *div* element to *bisque*. Use the color name.

Listing q23: /set/q-23.html

```
<!DOCTYPE html>
<html>

<head>
<style>
.lorem {
padding: 15px;
border: 1px dashed;
}
</style>
</head>

<body>
<h1>Lorem Ipsum</h1>
<div class="lorem">
<p>
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

</p>
</div>
</body>

</html>
```

Question - 24 [5 Marks]

Using the Listing q24, set the *font size* of the page to *20px* and *h1* to *1.5em*.

Listing q24: /set/q-24.html

```
<!DOCTYPE html>
<html>

<head>
<style>
.lorem {
padding: 15px;
border: 1px dashed;
}
</style>

</head>

<body>
<h1>Lorem Ipsum</h1>
<div class="lorem">
<p>
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. To know more <a href="example.com">Click Here</a>.
</p>
</div>
</body>

</html>
```

Question - 25 [5 Marks]

An *img* element has been assigned a class *tilt*. To *rotate* the image by *5.2*

degrees, the *tilt* class is defined as follows:

```
img.tilt{  
    width: 200px;  
    height: 200px;  
    float: left;  
    —moz—transform: rotate(5.2deg);  
    —o—transform: rotate(5.2deg);  
    —ms—transform: rotate(5.2deg);  
    —webkit—transform: rotate(5.2deg);  
    transform: rotate(5.2deg);  
}
```

Is there any error in the declarations, if yes, mention the error.

Question - 26 [5 Marks]

Consider the following element:

```

```

This code generates the *left* image in q26. Write the style code using class *flip* to *horizontally flip* the tiger, see the right image in Figure q26.



q26