Preston Prescott

HTML

HTML5

Preston Prescott

and informational purposes only

Contents

Chapter 1: Introduction

Creating a web page is as simple as eating breakfast. Virtually anyone can create or edit web pages, and you don't need any special skills or an advanced degree to become a web developer. All you need is a desire to learn, the motivation to succeed, and enough patience to go through the process of publishing a web page on the internet.

In this book, we are going to learn the basics of HTML 5, a markup language, and how it works. If you are already familiar with other versions of HTML, especially HTML 4, than learning HTML 5 should be a piece of cake for you. HTML 5 is the latest version of Hypertext Markup Language, yet it incorporates various features from HTML 4 and even XHTML. We will also shed some light all on these versions so that you can pick the style that meets your particular requirements. However, it is strongly recommended that you only use HTML 5 to create your web pages.

The book also has its own unique HTML examples in usable form. This is to help you create your own web page using HTML 5 with relative ease. The basic purpose of this book is to get your web page up and running on the World Wide Web so that you can amaze your relatives, friends, and colleagues with your new skills.

What is Hyper Text Markup Language?

People who know anything about HTML 4 or older versions of HTML might already know what Hypertext Markup

Language is and what role it plays on the web. As soon as you start exploring the internet, you come across a bevy of images, content, graphics, text, audio, and video. Every web page seems unique, but one thing is common between all of them: the Hypertext Markup Language, which works in the background to display those pages.

You have to use HTML to create your web page regardless of the type of content you intend to use. It is the force that keeps the content together and prevents your page from falling apart. HTML is a basic website building block and all other content acts like supporting bricks. It is a must to have full mastery over HTML before moving on to more complex programming languages, such as CSS and JavaScript.

It is also pertinent to note that HTML 5 web pages are simple text files. Text is a universal method of feeding data to computers and browsers. Anything you create using HTML 5 or other versions on your Personal Computer (PC) will work equally well on Mac, Linux, and other operating systems.

Hypertext:

HTML contains special instructions which enable lines of text to point to information on internet. Such pointers are normally called as “hyperlinks,” which are the lifeline of the World Wide Web. Without hyperlinks, there is no World Wide Web. Your web browser usually displays hyperlinks as blue, underlined text. If they are clicked, the hyperlink will take you to another page on the internet.

HTML 5

Now that you know what HTML is, it's time to come back to HTML 5. The World Wide Web Consortium extended the charter of HTML 5—which is the latest version of the standard and the 5th revision in total—to a full working draft on February 14, 2011. HTML 5 enables you to create web applications that can interact with your local data and servers more easily than ever before; additionally, it provides rich media support.

Initially, the World Wide Web Consortium developed HTML 5 to resolve compatibility issues which marred the previous version, HTML 4. The major difference between all previous versions and HTML5 is that the latter does not require APIs and proprietary plug-ins, which were responsible for creating compatibility issues. On the contrary, HTML makes loading easier and ensures cross browser compatibility by offering a common interface for all.

HTML 5 offers many new features which can totally change the way users interact with web. These features include but are not limited to:

- New Attributes
- New Parsing Rules
- Lack of redundant elements and attributes
- Detailed rules for parsing
- Offline editing
- Ability to store a MySQL data base using a common standard.

This book concentrates only on new elements, tags, and entities found in HTML 5.

Creating an HTML 5 Document:

Now it is time to create your first ever web page using HTML 5. You can use any professional text editor such as Notepad++, Sublime Text or Dreamweaver to edit HTML 5.

Let us begin. Open the text editor of your choice and write some HTML in it. Take a look at the following example.

Example 1:

```
<!DOCTYPE HTML>
<html>
<head>
<title>My First HTML 5 Document</title>
</head>
<body>
<h1>HTML 5 Tutorial</h1>
</body>
</html>
```

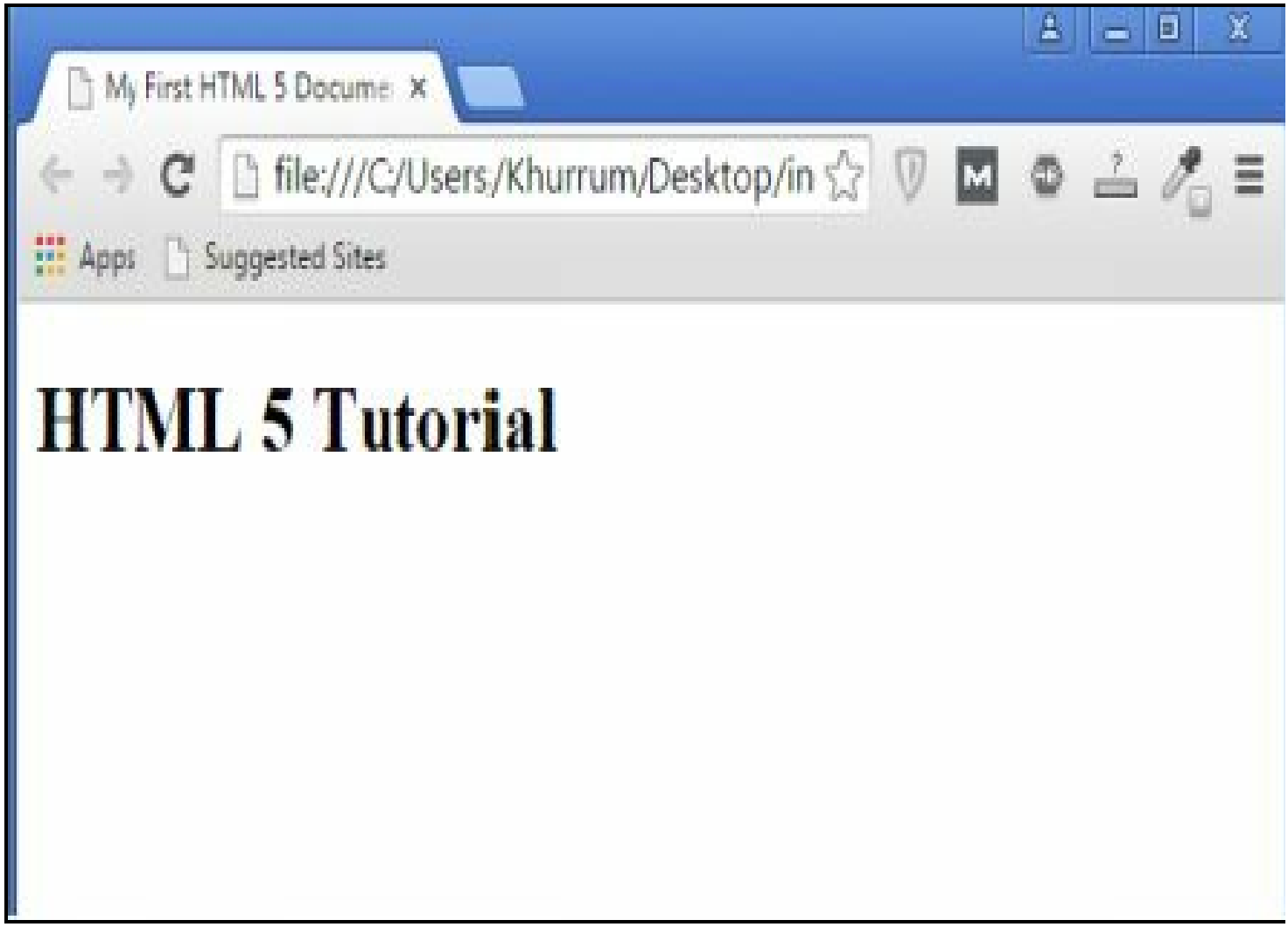
Remember that HTML 5 documents always start with <!DOCTYPE HTML> which tells the web browser that this is indeed an HTML 5 document.

Now you have to save the file on your computer. To do so, go to *File* and then click *Save As* in your editor's menu.

Name the file *index.html* (or give it any name of your choice, as long as it ends with .htm). Note that you can use both .htm and .html extensions.

Your HTML 5 document has been saved. Now you can

open the file in any browser to view it. The browser will display the file as follows:



Congratulations! You have built your first HTML5 document.

Text Editors to watch for:

It is up to you to decide which text editor you want to use. It basically depends on your particular requirements and taste. The following are some of the best text editors for both PC and Mac:

- Notepad++
- Sublime Text
- Text Wrangler

- Komodo Edit
- Text Edit
- Adobe Dreamweaver
- Microsoft Expression Web

All of these text editors are high quality and can help you edit your HTML 5 document smoothly and without any hassle. However, beginners are recommended to use Text Edit for Mac or Notepad++ for PC.

Summary:

In this chapter, we discussed:

- What this book will cover
- What Hypertext Markup Language is
- The difference between HTML 4 and HTML 5
- How to create an HTML 5 Document

Chapter 2: HTML 5 Document Structure

In Chapter 2, you will learn:

- The HTML 5 Document Structure
- How to Establish an HTML 5 Document Structure
- The HTML 5 Head Element
- The HTML 5 Body Element
- HTML 5 Headings and Paragraphs

The Document Structure:

At the end of the last chapter, we discussed how to create and save a simple HTML 5 document; however, we did not discuss much about the document structure. An HTML 5 document mainly consists of a Head and Body. The Head contains the data, which informs the browser and even web servers that it is an HTML 5 document. On the other hand, the Body contains content that web browsers actually display.

This chapter covers all the major elements necessary to craft a basic HTML 5 document.

Establishing a Document Structure:

Each and every HTML 5 document employs a unique combination of elements and content to define a page. The structure of all the properly documented pages is the same and contains:

- A declaration at the top, which indicates that it is an

HTML 5 document

- Adocument header
- Adocument body

A collection of HTML 5 elements constitutes an HTML 5 document. Some of these elements are essential while others are optional. However, you can always find the following three elements in every page in addition to the DOC Type declaration at the top.

- <!DOCTYPE> informs the browsers that it is actually an HTML 5 document. Although there are other types of DOC Types, this is the most commonly used declaration.
- The DOC Type Declaration is followed by <html> </html> opening and closing tags. These tags contain everything inside the document, including the Head and Body.
- <head></head> opening and closing tags follow the opening html tag. These tags contain information about the body, title of the page, definitions, labels, etc. You can only use certain markup elements in the HTML 5 head. Some of these elements include *style, title, base, link, script and meta*. In HTML 5, these elements are collectively known as HTML Head Elements.
- After the closing head tag is the <body></body> opening and closing Body tags. They contain all the content which appears on the browser, as well as the related HTML 5 codes. Theoretically, you can create an HTML 5 document without anything in the

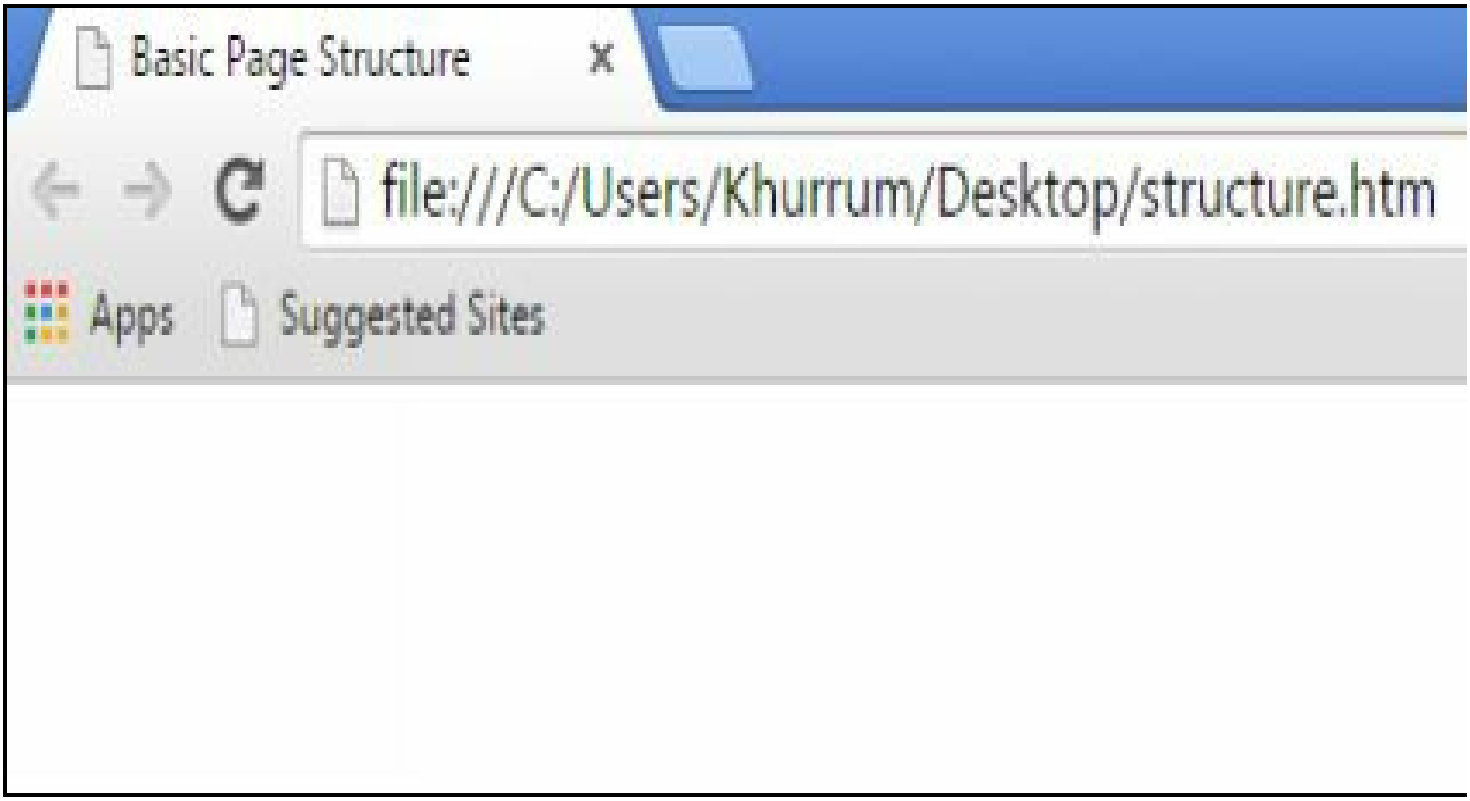
body, but you need to have a well-crafted Head and Body to index your page properly in the browser. The following example contains all the basic elements you need to create an HTML 5 page.

Example 1:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Basic Page Structure</title>
</head>
<body>
</body>
</html>
```

Notice that the head element also contains a Meta tag. Meta elements are used to specify other metadata, such as page author, description, keywords, last modified, etc.

If you save the above HTML 5 document and see it in the browser, it will display as:



Notice that the browser is showing nothing apart from the page title in the tab. This is because you have not done anything with the body so far.

The Body Element:

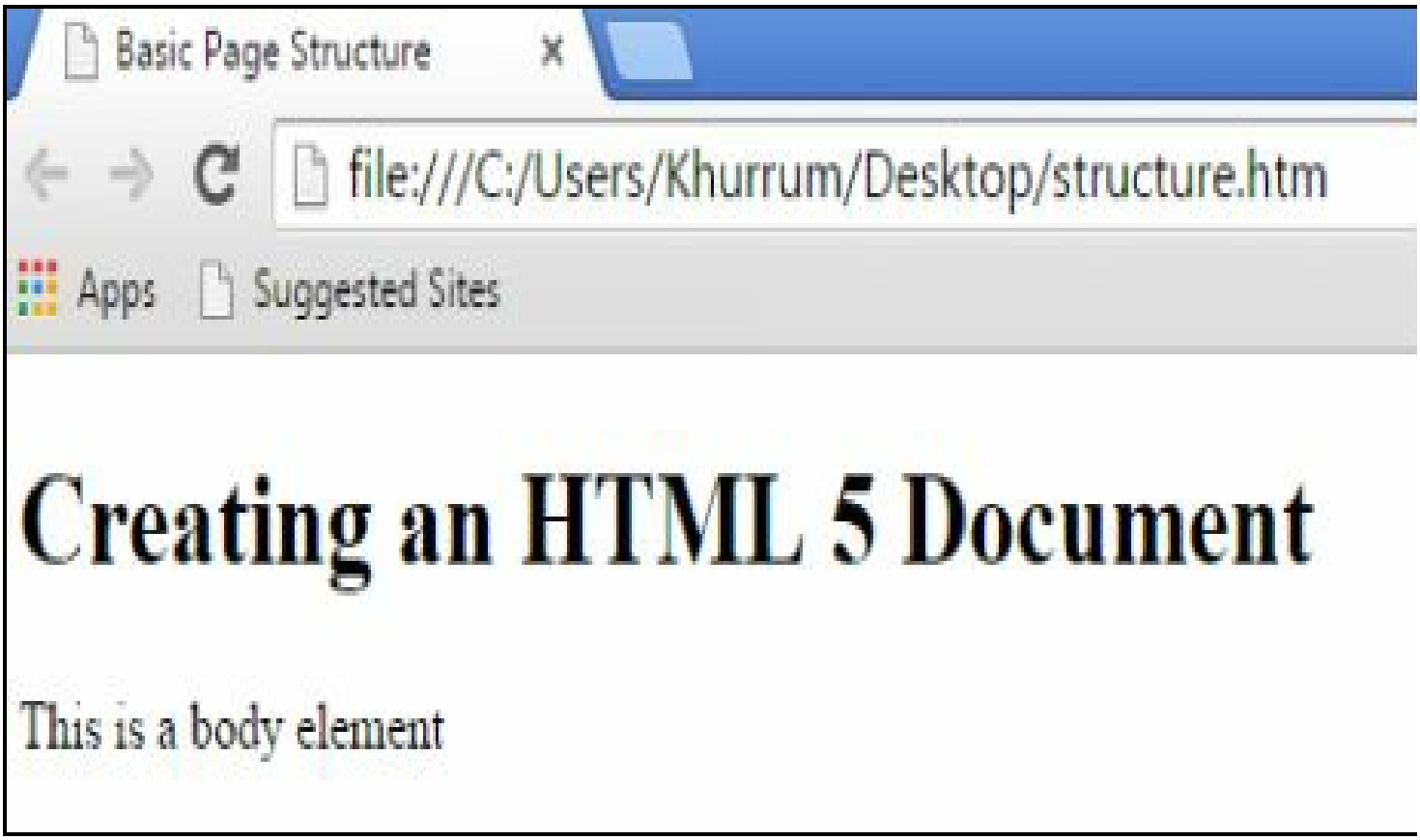
After successfully setting up your Head, including metadata and page title, it is time to create some HTML5 markup in the Body which will actually appear in the browser. The Body is like a big container which contains everything you want to see in the browser. Take a look at Example 2.

Example 2:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
```

```
<title>Basic Page Structure</title>
</head>
<body>
<h1>Creating an HTML 5 Document</h1>
<p>This is a body element</p>
</body>
</html>
```

You should see the following output in your browser:



That is fantastic. Now your browser is showing a heading at the top and some content below the heading as well. This is because you have added some HTML 5 markup in the Body element. Similarly, you can edit your HTML 5 document by adding any markup in the Body element.

Headings and Paragraphs:

Headings and paragraphs are two of the most important and fundamental elements used in HTML 5.

Headings:

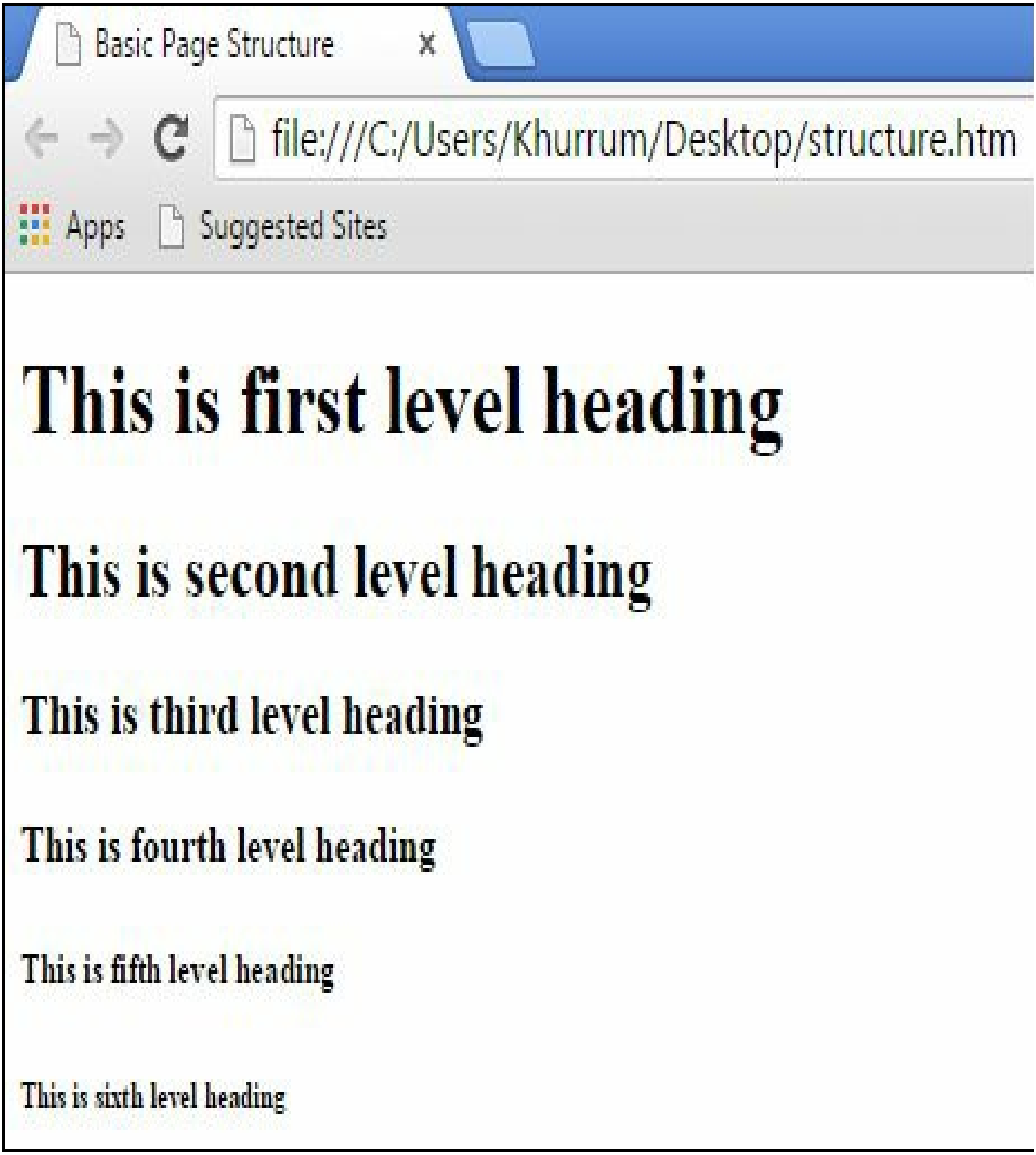
Denoted by the <h> tag, HTML 5 uses six levels of headings, from h1 to h6. In this regard, the first level heading <h1> is the largest whereas the last level heading <h6> is the smallest. The basic purpose of headings is to break documents into sections. They help you create an organized structure, breakup the text flow on the page, and provide visual cues about the grouping of the content. Example 3 demonstrates how to use headings in HTML 5.

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Basic Page Structure</title>
</head>
<body>
<h1>This is first level heading</h1>
<h2>This is second level heading</h2>
<h3>This is third level heading</h3>
<h4>This is fourth level heading</h4>
<h5>This is fifth level heading</h5>
<h6>This is sixth level heading</h6>
```

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

Different browsers will display the headings differently. Google Chrome displays them as the following:



Paragraphs:

Anyone who is familiar with HTML 4 or other version of HTML must also be familiar with the paragraph tag.

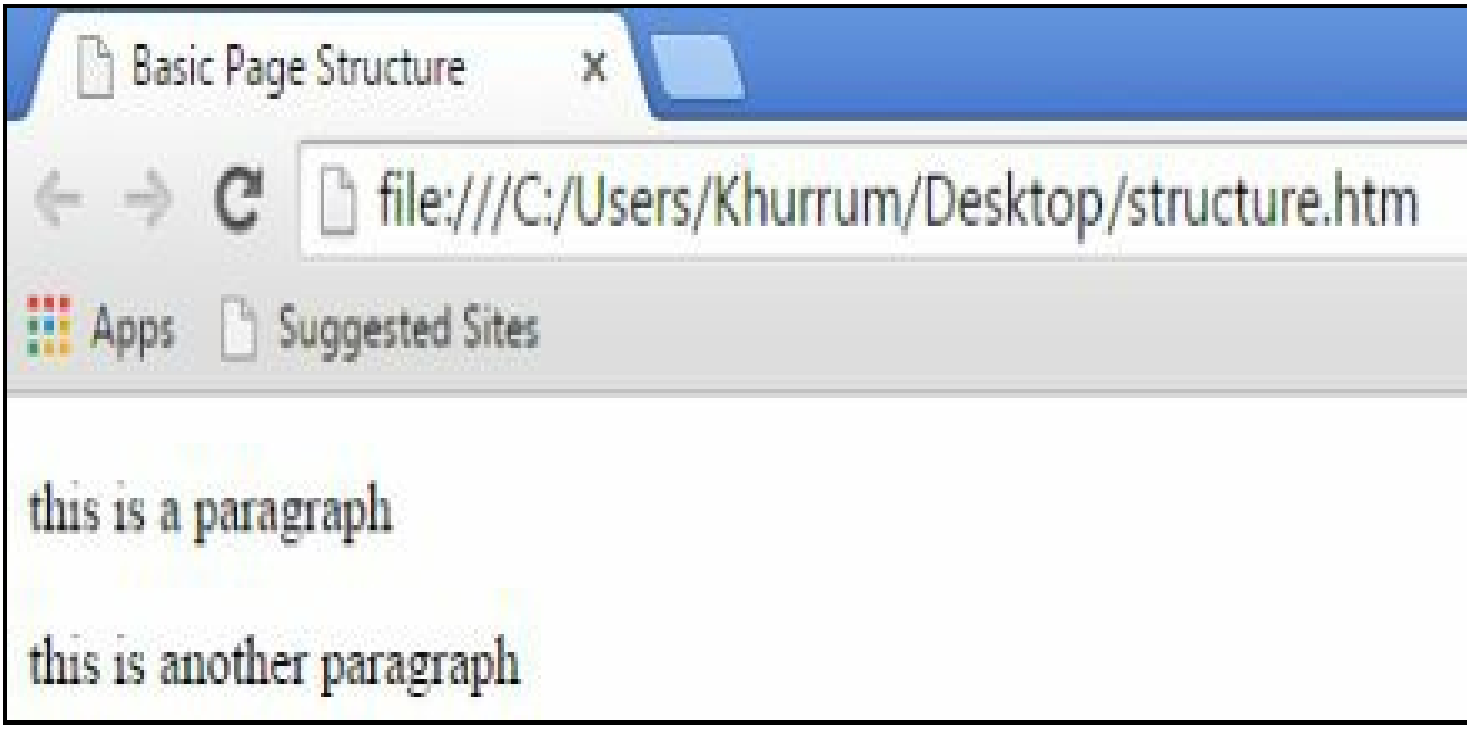
Paragraph is a text block which appears more than any other element on the web page. You have to use opening and closing tags in order to group your content in paragraphs because web browsers do not recognize hard returns inside HTML 5 editors.

It is very simple to create a paragraph in HTML 5. You have to put an opening tag inside the body, write some text and close the paragraph with a tag. Take a look at Example 4.

Example 4:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Basic Page Structure</title>
</head>
<body>
<p>this is a paragraph</p>
<p>this is another paragraph</p>
</body>
</html>
```

When you see the code in browser, you will get something like this:



Thanks to the paragraph tag, we have two different paragraphs on our HTML 5 web page. In the same way, you can create as many paragraphs as you require or desire on your web page. Just remember to properly close your paragraphs with ; otherwise it could cause a lot of rendering issues and your web page might not load properly.

Exercise 2

Task:

Create a basic HTML 5 page and title it Exercise. Also add an <h1> heading containing the text “Exercise 2” and a paragraph explaining what this exercise is about.

Solution:

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

<body>

<h1>Exercise 2</h1>
<p>Exercise 2 is about the headings and paragraphs tags learned in chapter 2.</p>

</body>
</html>
```

Chapter 3: Formatting Text & Creating Lists

In chapter 3, you will learn about:

- Formatting Text in HTML 5
- Organizing Text in Lists

Formatting Text or Text Decoration:

HTML 5 documents include a lot of text, images, videos, multimedia, and other types of content, making it important for you to format blocks of content to lay a strong foundation for your web page. In chapter 2, you learned how to organize text in HTML 5 using block level elements, such as paragraphs and headings. In this chapter, you will explore other important elements which will help you format text properly.

Bold and Italics:

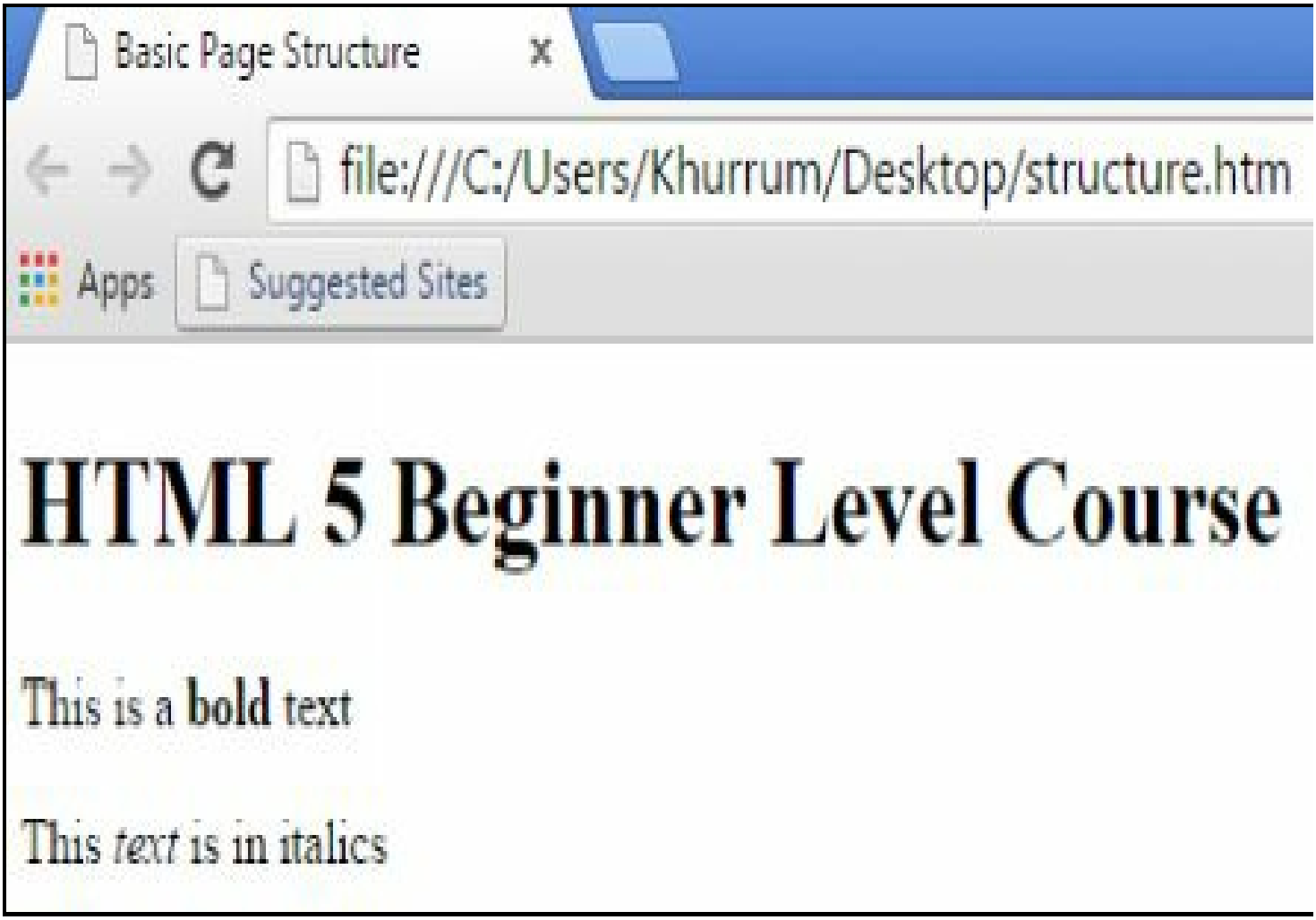
HTML 5 introduces some new formatting elements which were missing from previous versions. For instance, it is strongly recommended that you use opening and closing tags for making a chunk of text bold instead of using .

Similarly, HTML 5 experts prefer using emphasize element tags for italics rather than . This

effectively informs the browser that it is an HTML 5 document and that the content between the and tags is in italics.

Example 1:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Basic Page Structure</title>
</head>
<body>
<h1>HTML 5 Beginner Level Course</h1>
<p>This is a <strong>bold</strong>text</p>
<p>This <em>text</em> is in italics</p>
</body>
</html>
```



Line Breaks and Horizontal Rule:

HTML 5 normally ignores line breaks unless you use certain formatting elements or tags which break the line as needed. The most common of these elements is the line break element, commonly referred to as
. Look at example 2 to see what happens when we try to put some space between two words or lines within a paragraph without
.

Example 2:

```
<!DOCTYPE HTML>
<html>
```

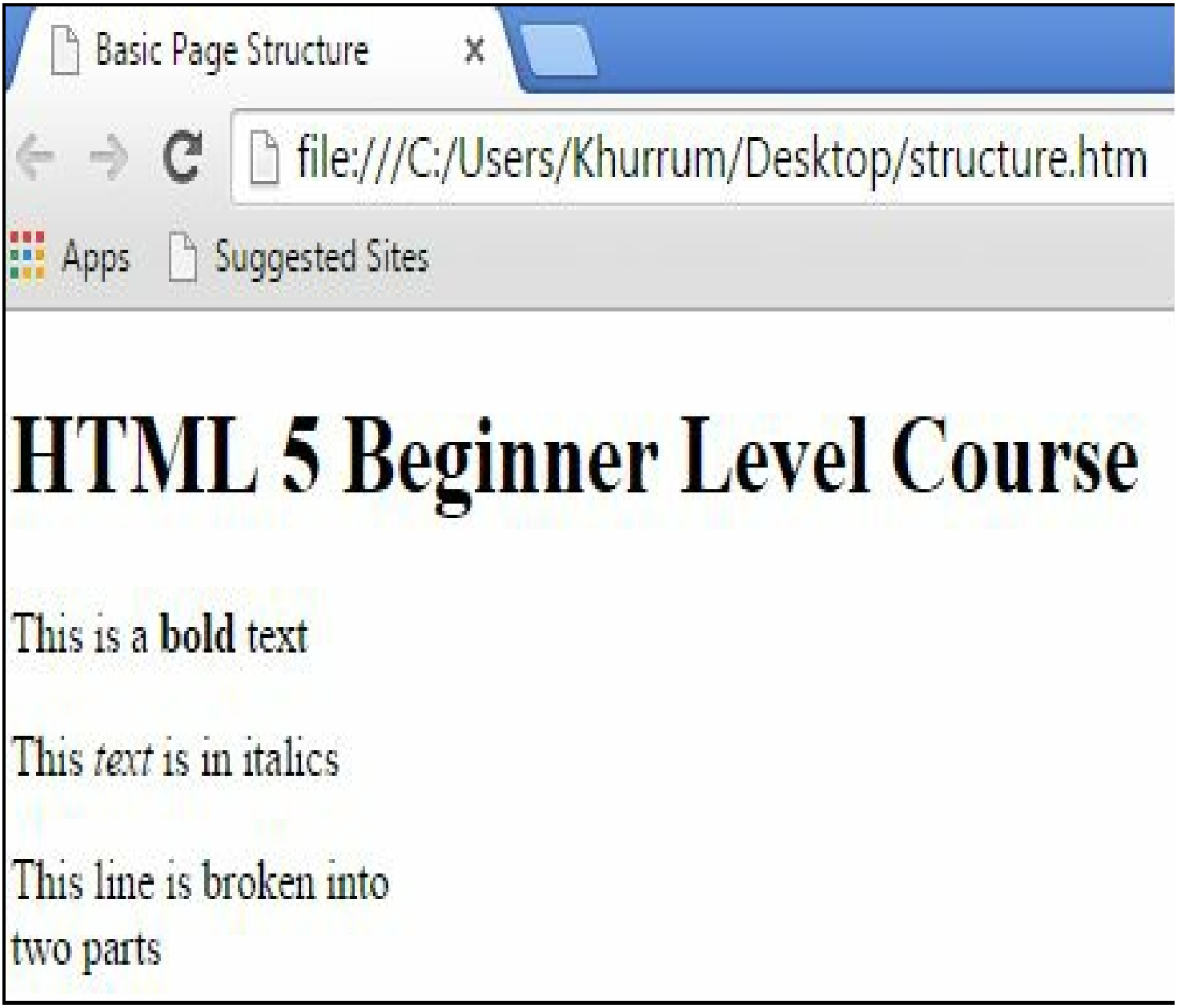
```
<head>
<meta charset= “utf-8”>
<title>Basic Page Structure</title>
</head>
<body>
<h1>HTML 5 Beginner Level Course</h1>
<p>This is a <strong>bold</strong>text</p>
<p>This <em>text</em> is in italics</p>

<p>This line is broken into
    two parts</p>
</body>
</html>
```

As you can see, HTML 5 has totally ignored the line breaks. Now we'll put in a
 to break the line and see what happens.

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= "utf-8">
<title>Basic Page Structure</title>
```

```
</head>
<body>
<h1>HTML 5 Beginner Level Course</h1>
<p>This is a <strong>bold</strong>text</p>
<p>This <em>text</em> is in italics</p>

<p>This line is broken into<br>
    two parts</p>
</body>
</html>
```



Now, HTML 5 has inserted the line break as we wanted it to. Remember that
 is a self-closing tag and does not require any separate closing tag to complete the code.

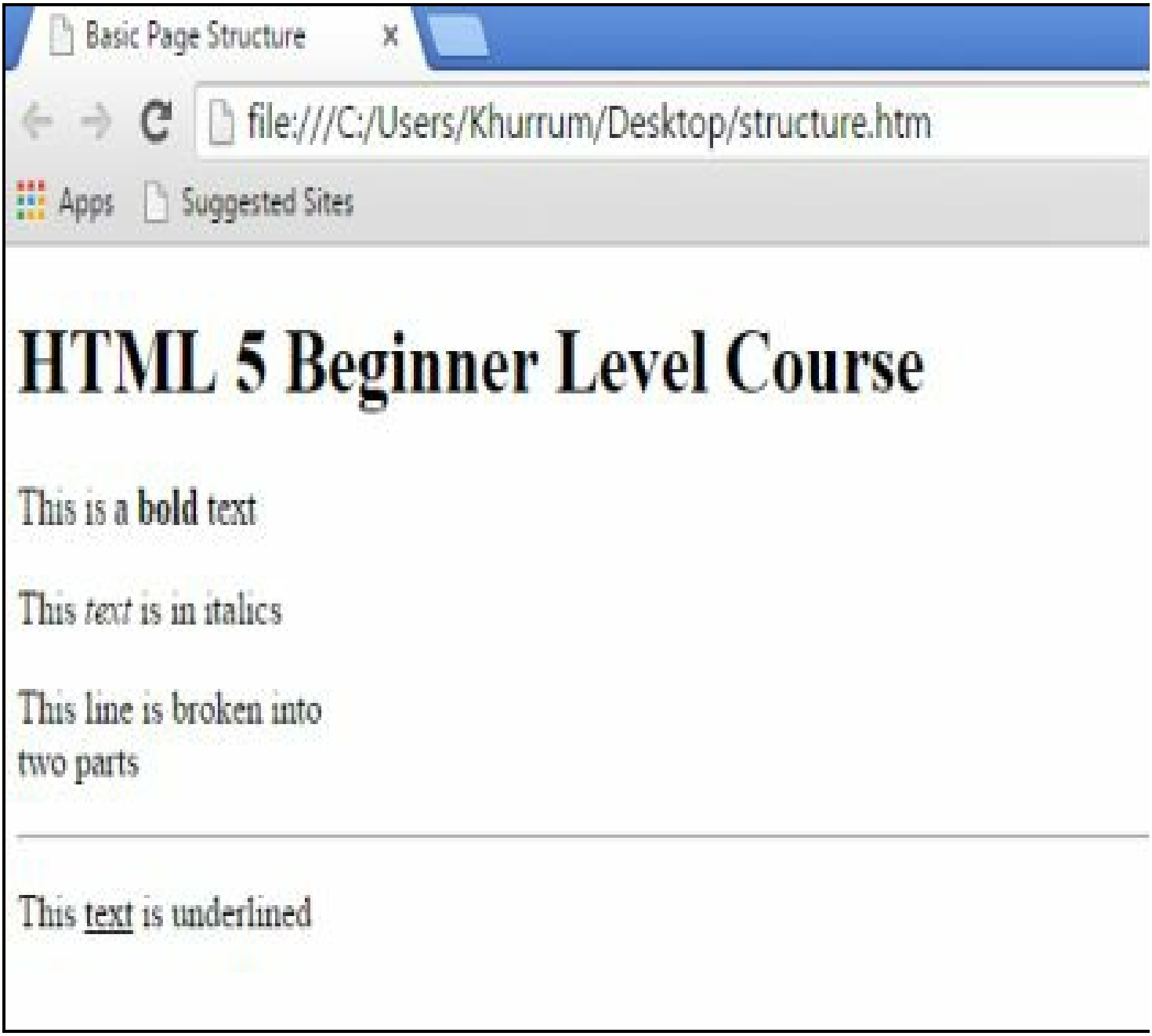
Just like
, horizontal rule, or <hr>, is also a self-closing tag; it is used to put solid straight lines, called rules, on your page. A horizontal rule is a good option to break a page into logical sections or to separate headers and footers from the rest of the page.

Another important tag that is commonly used in HTML 5 is

the <u></u> opening and closing tag. The <u> tag informs the browser that a specific piece of content is underlined. The following example demonstrates how to use these tags in HTML 5.

Example 4:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Basic Page Structure</title>
</head>
<body>
<h1>HTML 5 Beginner Level Course</h1>
<p>This is a <strong>bold</strong>text</p>
<p>This <em>text</em> is in italics</p>

<p>This line is broken into<br>
    two parts</p>

<hr>

<p>This <u>text</u> is underlined</p>

</body>
</html>
```



Organizing Text in Lists:

You will create lists if you are looking to group similar information on your page. Lists allow your visitors to easily go through groups of information. You can include anything in the list from sets of information to links. There are four main types of lists: the unordered list, the ordered or numbered list, the definition list, and the nested list.

The Unordered Lists:

An unordered or bulleted list contains one or more items prefaced by bullets. An unordered list requires at least three elements in the following order.

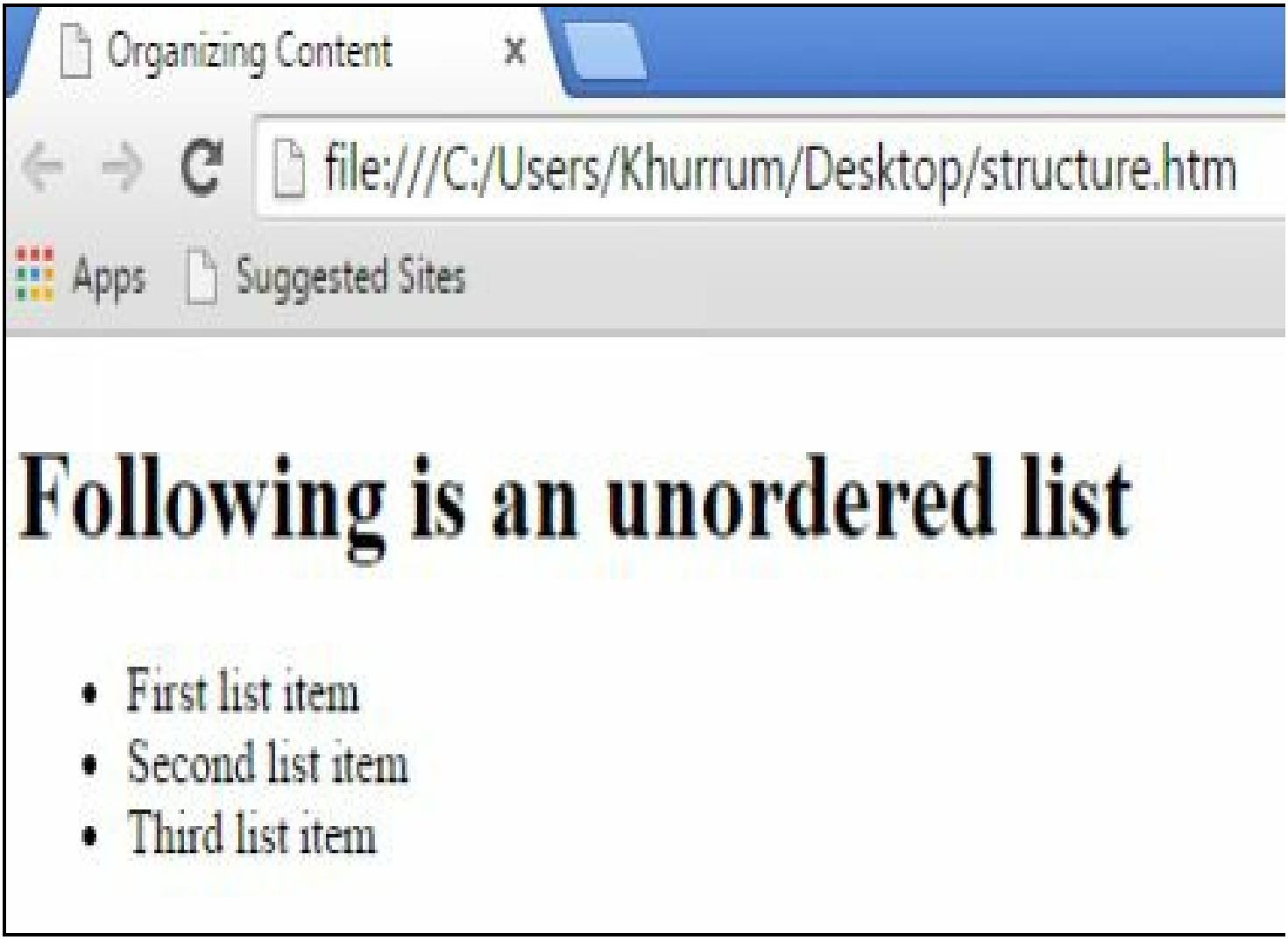
- the unordered list element
- which marks each element in the list
- the closing tag which indicates the unordered has ended

The following example contains the markup for unordered lists.

Example 5:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Organizing Content</title>
</head>
<body>
<h1>Following is an unordered list</h1>

<ul>

<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>

</ul>
```

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



Ordered or Numbered List:

Anumber prefaces all the items in an ordered or numbered list. An ordered list always has at least two items. Just like unordered lists, an ordered list contains the following three elements.

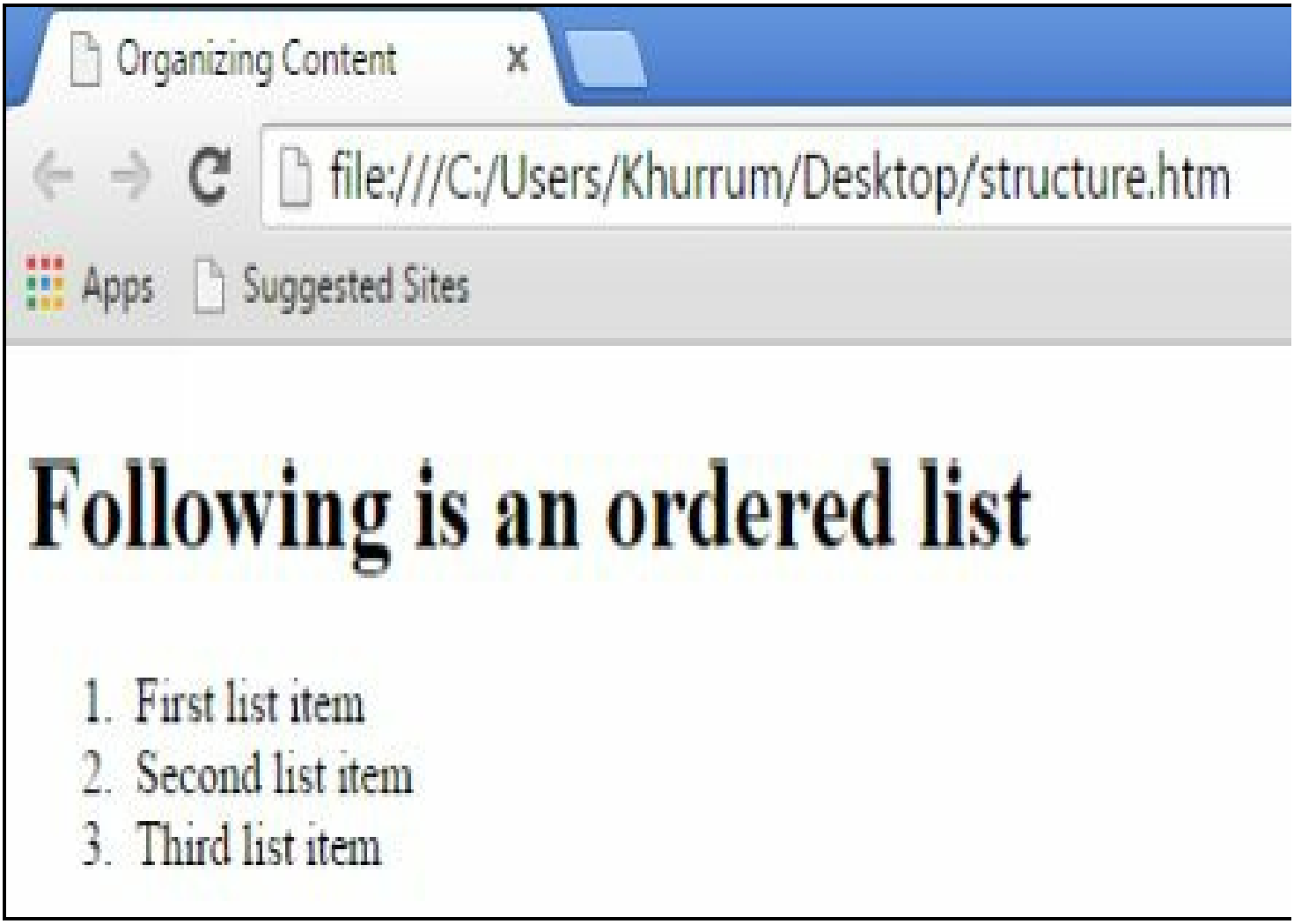
- the ordered list element
- which marks each elements in the list

- the closing tag indicating the ordered has ended

The following example contains the markup for an ordered list.

Example 6:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Organizing Content</title>
</head>
<body>
<h1>Following is an ordered list</h1>

<ol>

<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>

</ol>

</body>
</html>
```



The Definition List:

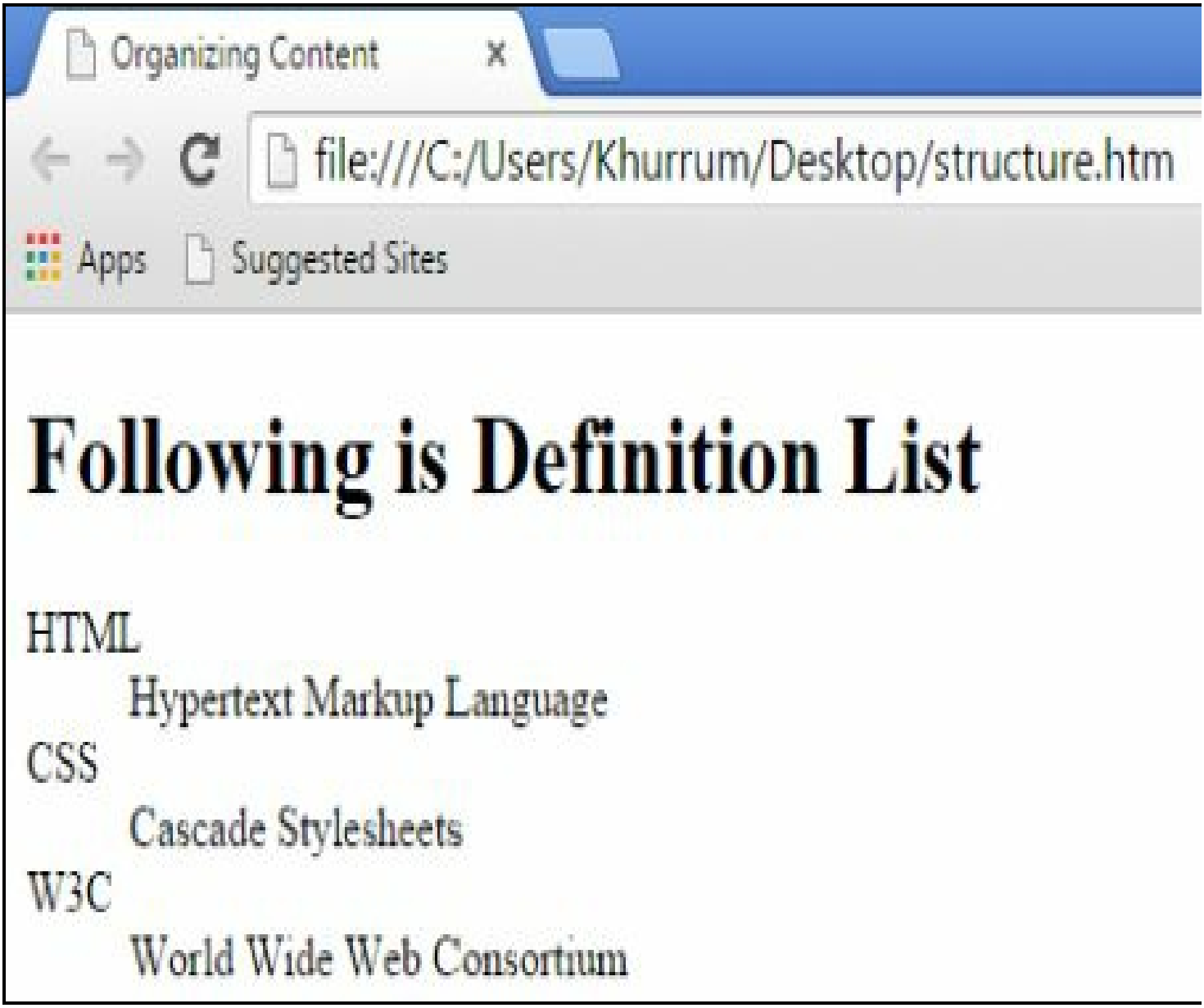
Definition lists are used to group definitions and terms into a single list on a page. Again, the definition list requires the following three elements.

- <dl> the element which defines the definition list
- <dt> the element which marks the term in the definition list
- <dd> defines the term in the definition list

The HTML for the definition list is as follows.

Example 7:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Organizing Content</title>
</head>
<body>
<h1>Following is Definition List</h1>

<dl>

<dt>HTML</dt>
        <dd>Hypertext Markup Language</dd>
<dt>CSS</dt>
        <dd>Cascade Stylesheets</dd>
<dt>W3C</dt>
        <dd>World Wide Web Consortium</dd>

</dl>

</body>
</html>
```



It is pertinent to note that different browsers display definition lists differently. Similarly, each search engine handles them in different ways, as do text-to-speech translators.

Nesting Lists:

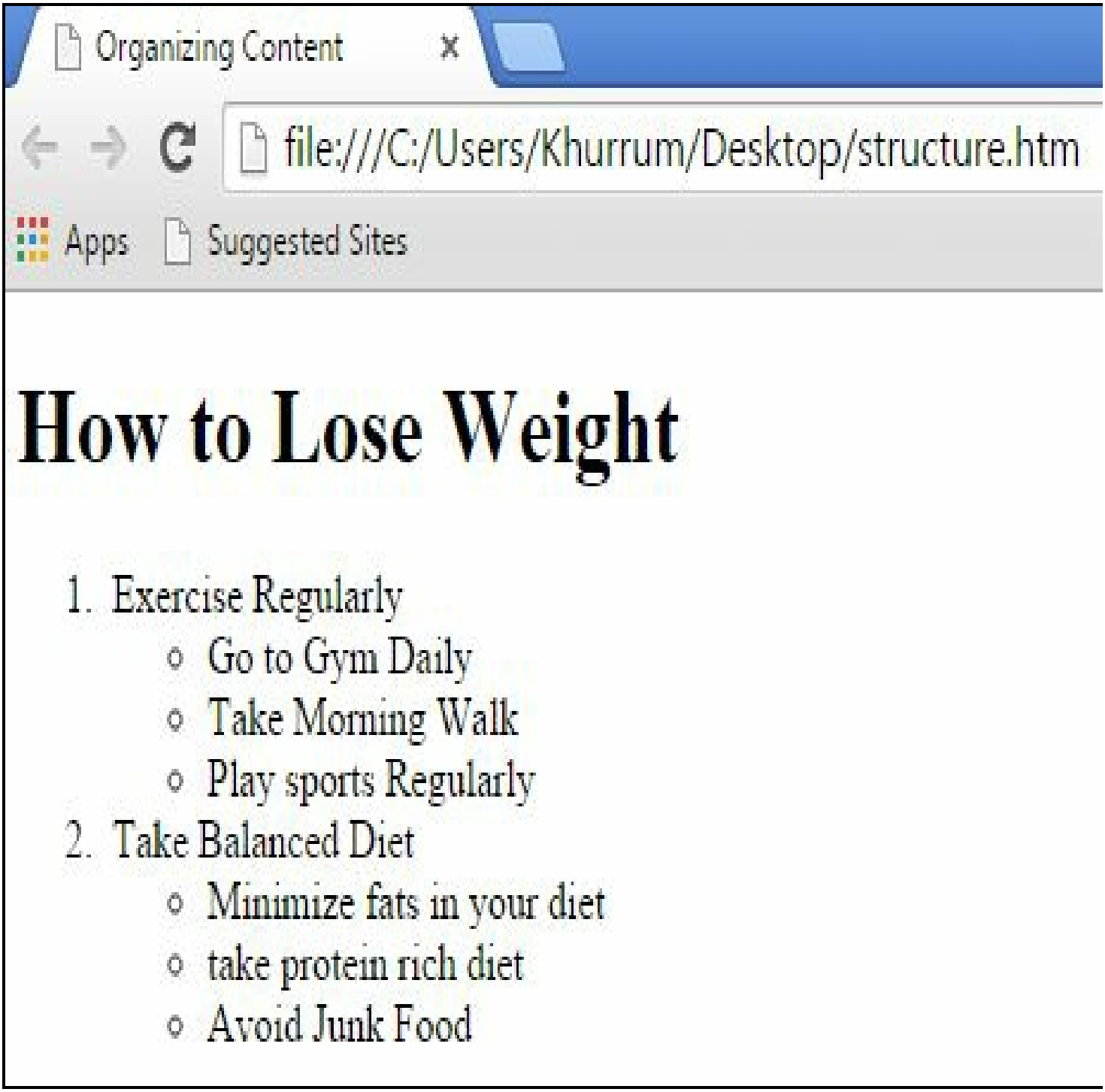
Finally, there are nesting lists which are actually subcategories within lists. They are mostly used for creating navigation tools, site maps, table-of-contents for online papers and books, and outlines.

You can combine any of the first three types of lists with each other to create a nested list. The following example starts with a numbered list, which defines things you need to do to lose weight, and further divides them into specific tasks.

Example 8:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Organizing Content</title>
</head>
<body>
<h1>How to Lose Weight</h1>

<ol>

<li>Exercise Regularly
<ul>
<li>Go to Gym Daily</li>
<li>Take Morning Walk</li>
<li>Play sports Regularly</li>
</ul>
</li>

<li>Take Balanced Diet
<ul>
```

```
<li>Minimize fats in your diet</li>
<li>take protein rich diet</li>
<li>Avoid Junk Food</li>
</ul>
</li>

</ol>

</body>
</html>
```



You can include as many items as you like in nested lists. However, when working with nested lists, you need to keep few things in mind:

- A complete second level list follows each list in top level ordered or unordered list.
- The second level list does not rest in the list items but in the top level list.

- Nested lists can become quite complicated; therefore, you should close all tags very carefully to avoid any rendering issues.

Exercise 3

Task:

Create an ordered list containing the names of any three states of the United States. Then, create three nested lists within the ordered list containing the names of two cities of each state.

Solution:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Exercise 3</title>
</head>
<body>
<h1>States of USA</h1>

<ol>
```

```
<li>California
<ul>
<li>Sacramento</li>
<li>Los Angeles</li>
</ul>
</li>

<li>Florida
<ul>
<li>Miami</li>
<li>Orlando</li>
</ul>
</li>

<li>Texas
<ul>
<li>Austin</li>
<li>Houston</li>
</ul>
</li>



</ol>

</body>
</html>
```

Chapter 4: Creating Tables in HTML 5

In this chapter, you will learn about:

- The capabilities and benefits of tables
- Creating tables in HTML 5

Tables in HTML 5:

If you are looking for an easy way to lay text, data, content, and images in a grid, try creating a table. Tables are perfect for displaying numerical data, but they can also help you present any information that naturally falls into columns and rows in a manner that's easy to understand. They can also help you save a lot of precious white space which would otherwise have been occupied by terms, definitions, and similar data which covers a lot of space when presented in paragraphs or other elements.

Markup in an HTML 5 Table:

Compared to lists, tables in HTML 5 contain a lot of markup. However, the *table* element is the primary markup container for creating tables. You use an opening table tag <table> and a closing table tag </table> to denote the opening and closing of a table in HTML 5. The table row <tr> and table data <td> are other primary elements used in tables in HTML 5.

You can use these three elements to create a basic table. However, if you are looking to create a rather complex table

in HTML 5, you can use other elements, such as <caption> or <colgroup> within these three primary elements. Moreover, there are <thead> and <tbody> elements which define the heading and actual content of the table, respectively.

The basic markup for a simple table in HTML 5 is as follows.

Example 1:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Tables</title>
</head>
<body>
<h1>How to Create Tables in HTML 5</h1>

<table>
<tr><th>Name</th><th>Age</th><th>Gender</th></tr>
<tr><td>David</td><td>25</td><td>Male</td></tr>
<tr><td>William</td><td>30</td><td>Male</td></tr>
<tr><td>Elizabeth</td><td>22</td><td>Female</td></tr>
</table>

</body>
</html>
```

Remember that you will have to use <tr> inside all other table containers.

You can add a border and a caption to the same table by inserting a caption <caption> element and a border entry, as shown in the following figure.

Example 2:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Tables</title>
</head>
<body>
<h1>How to Create Tables in HTML 5</h1>

<table border=“1”>
<caption>Markup for HTML 5 Table</caption>
<tr><th>Name</th><th>Age</th><th>Gender</th></tr>
<tr><td>David</td><td>25</td><td>Male</td></tr>
<tr><td>William</td><td>30</td><td>Male</td></tr>
<tr><td>Elizabeth</td><td>22</td><td>Female</td></tr>
</table>

</body>
</html>
```

Our table will look like this in the browser:

The Table Border:

Every table you create must have a border; you can apply one by using the "border" entry which we have used in the previous example. It is important for you to define the value of the table as "1" for the browser to validate it properly. However, browsers will display the border regardless of the value you use. Earlier versions of HTML use the table="n" entry to determine the width of the table. HTML 5 only uses the entry to display or not display the table if you remove it from your code. It has nothing to do with the width of the

table in HTML 5.

Managing Table Layouts:

You can make your tables complex, and you can also do a lot of interesting things with them. HTML 5 gives you a lot of power in this regard. However, for the sake of this book, it is recommended that you stick to the basics to better understand how HTML 5 works as far as tables are concerned.

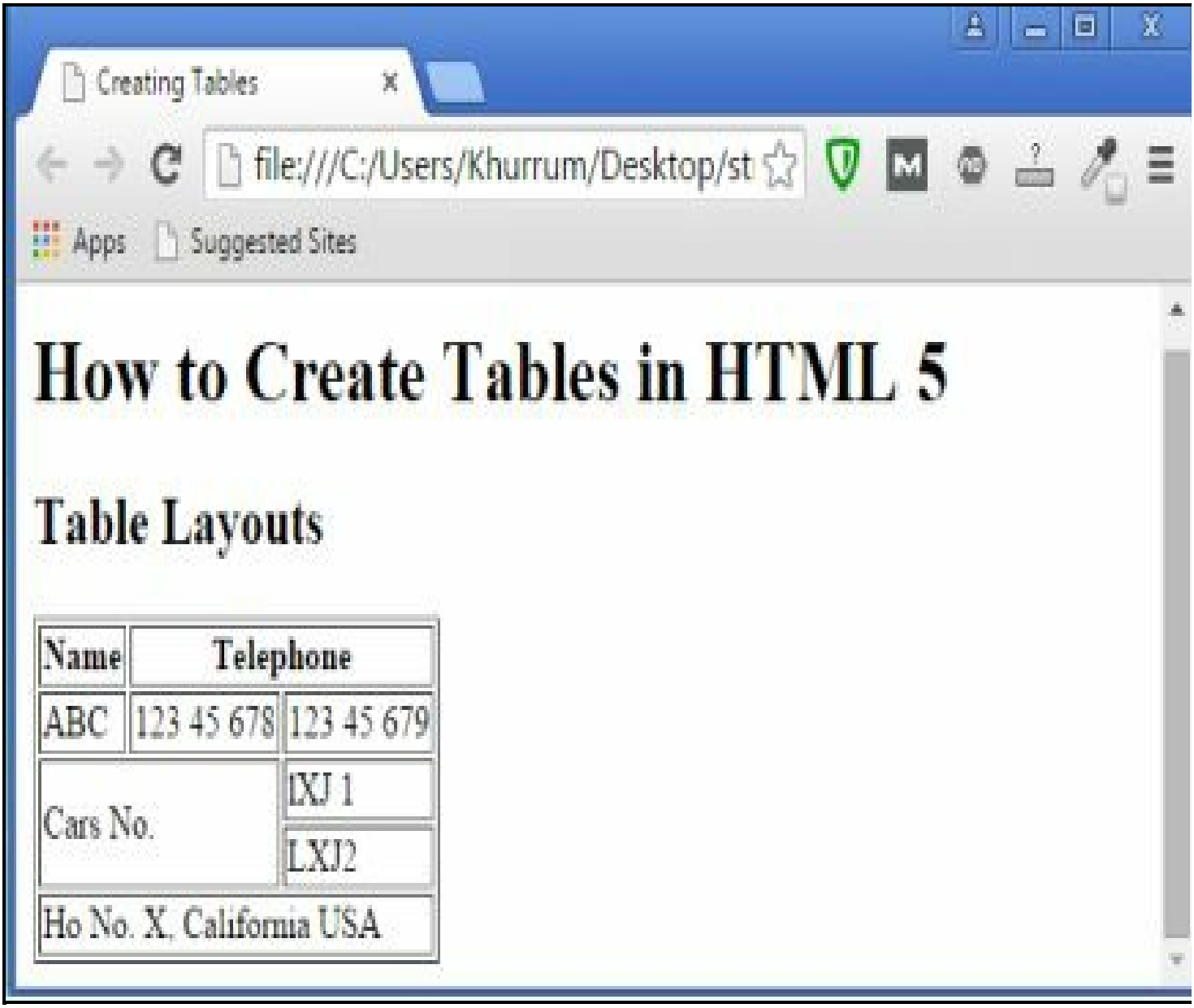
When it comes to the layout of the tables, there are only three main elements in HTML 5 which enable you to design a layout of your choice. These elements are as follows:

- **Cells** are the basic building blocks of tables and contain everything including data, text, images, and whatever else is within the borders of table.
- **Borders** are actually four lines on all four sides of the table that form a rectangle or square.
- **Cell Spans** help you add or delete cell walls within the borders of the table. Cell spans enable you to accommodate different cell layouts, making the table much more flexible.

Cell spans actually combine or merge cells, or remove walls between them to change the cell space. You can make your table much more attractive by spanning different rows, cells, and columns. In the following example, we will try to span cells and rows in the table and see what happens.

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Tables</title>
</head>
<body>
<h1>Table Layouts</h1>

<table>
<tr><th>Name</th><th colspan="2">Telephone</th></tr>
<tr><td>ABC</td><td>123 45 678</td><td>123 45
679</td></tr>
<tr><td rowspan="2" colspan="2">Cars No.</td><td>IXJ
1</td></tr>
<tr><td colspan="2">LXJ2</td></tr>
<tr><td colspan="3">Ho No. X, California USA</td></tr>
</table>

</body>
</html>
```



With this example, we conclude this chapter. Try to follow the instructions in this book to master HTML 5 basics before moving on to more complex table layouts and designs.

Exercise 4

Task:

Create a 3 column table with the first heading of Car, second heading of Manufacturer, and third heading of

Price. Put the names of your three favorite cars, their manufacturers, and their prices under each heading by creating three rows. Also add a border to the table and give it the value of 1.

Solution:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Exercise</title>
</head>
<body>
<h1>Exercise 4</h1>

<table border= “1”>
<tr><th>Car</th><th>Company</th><th>Price</th></tr>
<tr><td>i8</td><td>BMW</td><td>$13,6650</td></tr>
<tr><td>Fiesta</td><td>Ford</td><td>$13,965</td></tr>
<tr><td>XF</td><td>Jaguar</td><td>$94,975</td></tr>
</table>

</body>
</html>
```

Chapter 5: HTML 5 Forms

In this chapter, you will learn about:

- Creating and using forms on web pages
- Working with form data

You can present information and data on the web page for users with the help of HTML 5. In addition to providing static information on your page, you can also collect data from users. There are a plethora of *HTML form elements* and *tags* which give you the freedom to collect valuable information from the visitors coming to your site.

Types of forms in HTML:

Before we experiment with the markup for the forms, we will briefly discuss the types of forms in HTML 5. Irrespective of their shape, size, or use, all HTML 5 forms fall into two main categories which are as follows.

Search Forms:

As the name suggests, a search form helps your visitor obtain the required information from your website. You can search almost anything you like on the internet, depending on the genre of the website and type of search form they are using. There are search methods that require only a keyword, as well as search forms that are more sophisticated. The main form on the home page of the IRS is considered a search form.

Data-Collecting Forms:

As compared to search forms, data-collecting forms help you gather the necessary data and information from your visitors. Your form will be simple or complex depending on the amount and type of information you are collecting. For instance, if you need just a little bit of information the form can by fairly simple. Most forms that require your email address or name are data collection forms.

Creating Forms:

You can use HTML forms to receive information from users and vice versa. However, forms also offer various other methods to present information to users, such as:

- **Check boxes**, which allow you to choose multiple options from a group of choices.
- **Data selection tools**, such as *radio buttons*, which enable you to choose one option from a list.
- **Text Input Fields**, which can help you create tables extending to single line, double lines, or even multiple lines.

Structure of a Table:

The main element for creating tables is <form>, which also acts a content and input container much like paragraph <p>. The <form> element creates a logical document section in HTML 5, further incorporating sub-elements. It contains all the elements and tags associated with a single form which are also processed by the same form handler.

Input Tags and Fields:

The bulk of any form is composed of input tags, which you use to obtain information or *input* from your visitors. The major input element when it comes to getting information from users through HTML 5 forms is <input>. Within the input element, you define the kind of input or information you want to receive.

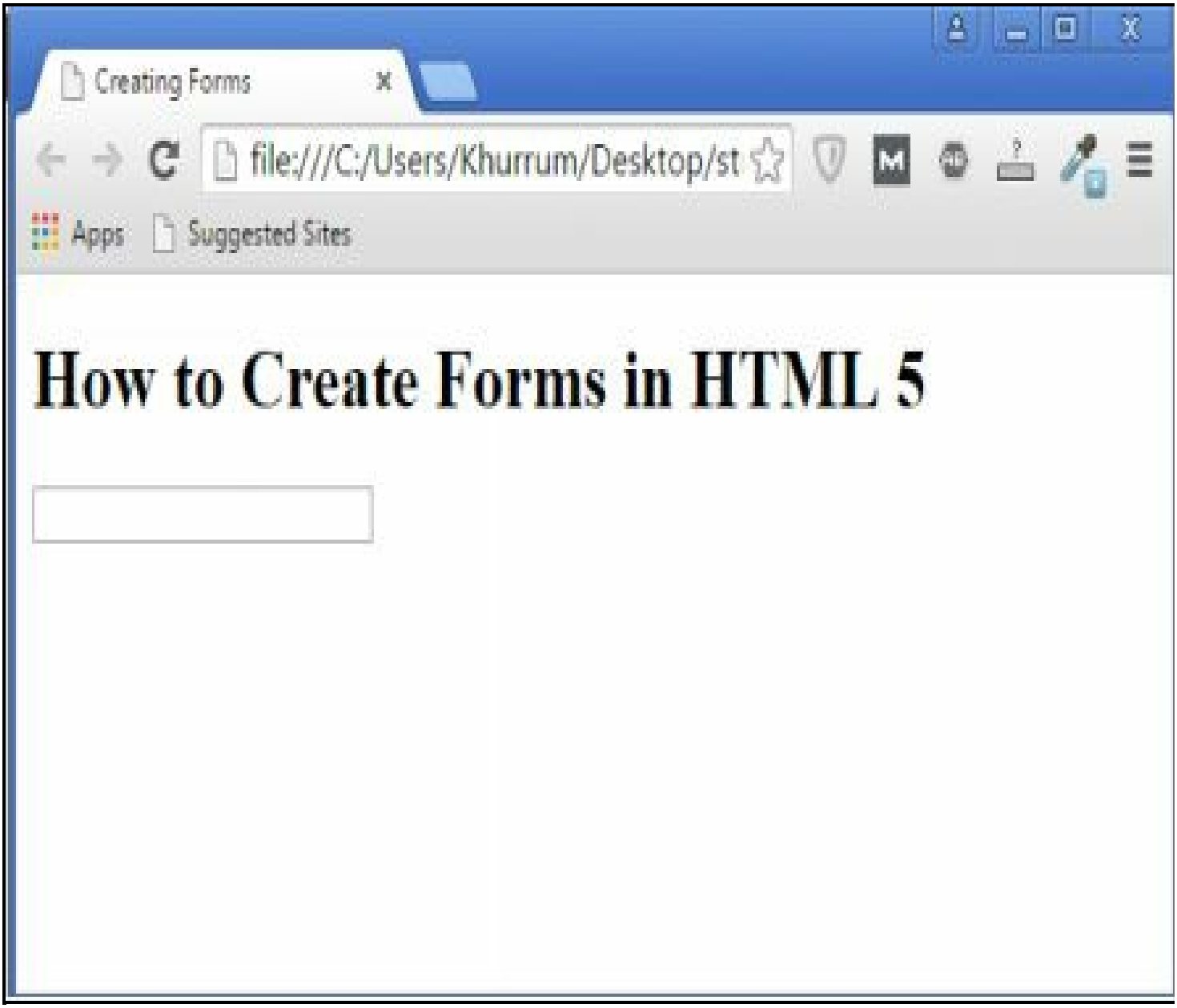
Similarly, you can use different types of input *fields or types* in your forms including, but not limited to, *radio, text, password, search, hidden, checkbox, submit, and email.*

The following example contains the markup for a simple form.

Example 1:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Forms</title>
</head>
<body>
<h1>How to Create Forms in HTML 5</h1>

<form action=”bin/guestbook.php”>
<input type=“text”/>
</form>
```

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

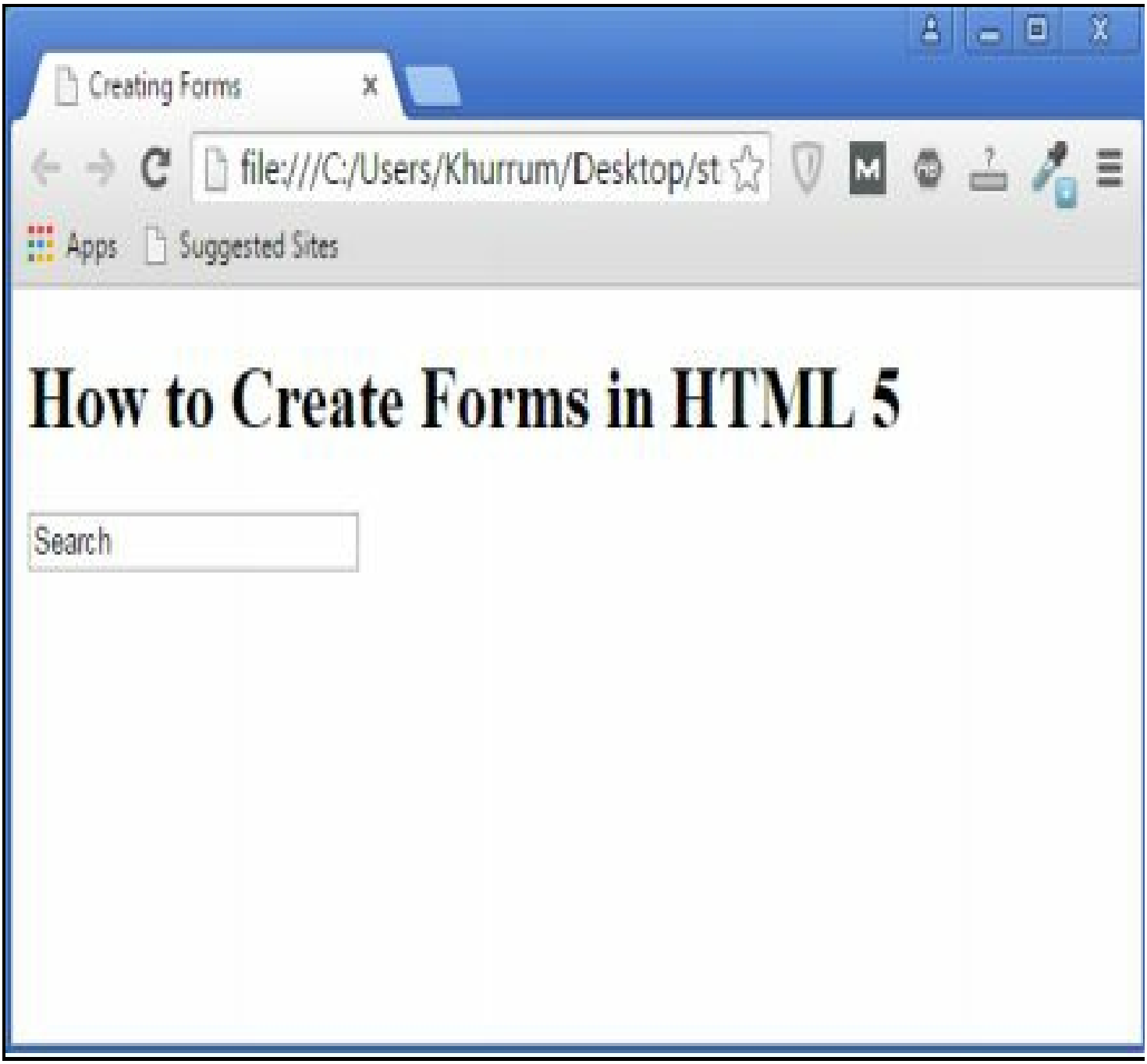


The above mentioned is the simplest of form you can create in HTML 5. Note that the <form> element contains a *section* attribute. A section attribute is an integral part of the table which informs browser about the destination *url* of the table.

We can do many more things to make tables more attractive or functional. For instance, we can add *value* attributes to the table, as the following example demonstrates.

Example 2:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Forms</title>
</head>
<body>
<h1>How to Create Forms in HTML 5</h1>

<form action=”bin/guestbook.php”>
<input type=“text” value=“Search”/>
</form>

</body>
</html>
```

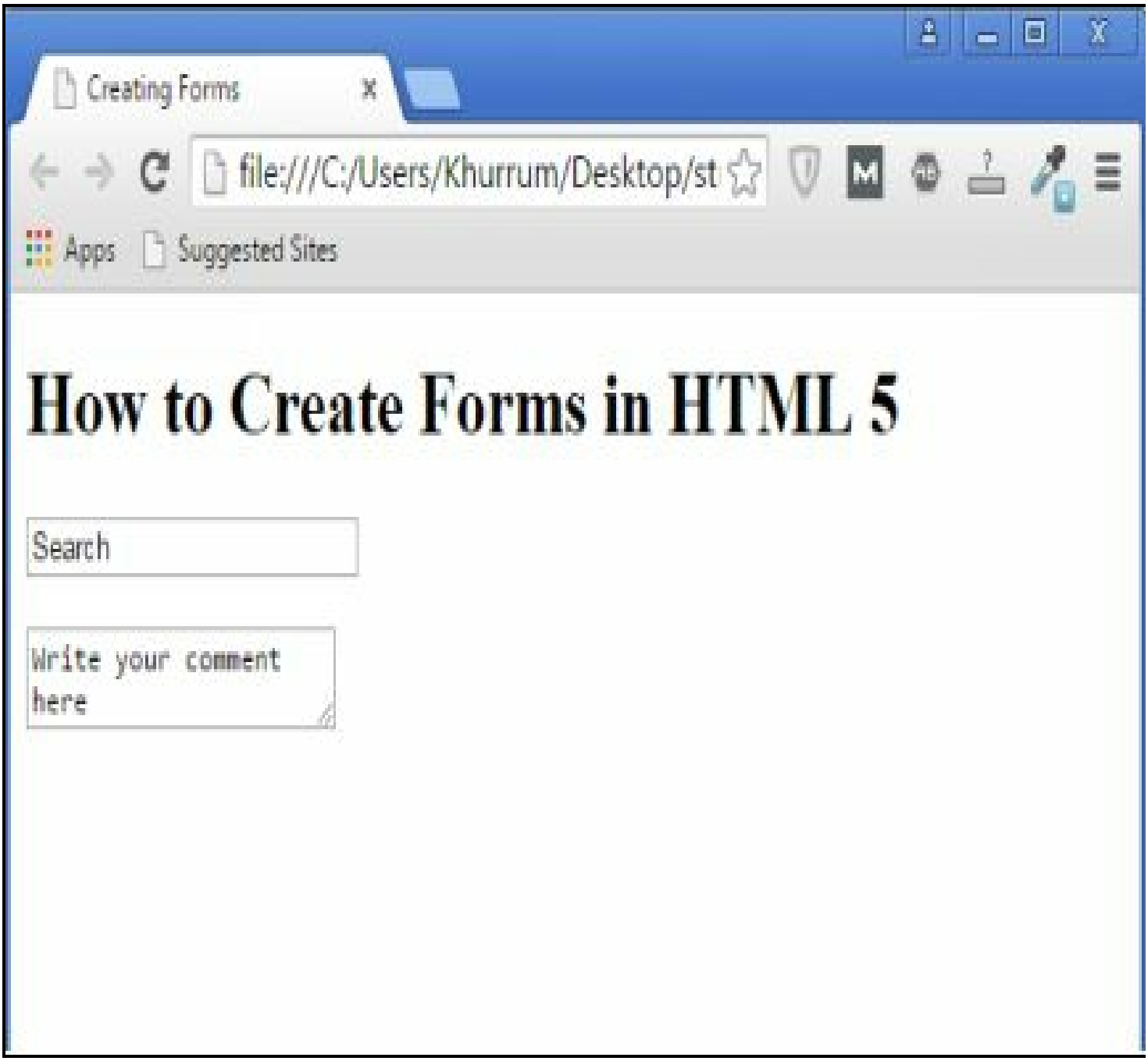


Now, you are practically telling your visitors to use this form to search for their required information. As a matter of fact, you can use any value inside the form, such as example, type here, or email.

What if you want users to put a larger piece of content in your form? You can do so by creating a Comment section by using a <textarea> tag.

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Forms</title>
</head>
<body>
<h1>How to Create Forms in HTML 5</h1>

<form action=”bin/guestbook.php”>
<input type=“text” value=“Search”/>
<br><br>

<textarea>Write your comment here</textarea>
</form>

</body>
</html>
```



The <textarea> tag is different from <input> in the sense that it is not self-closing. You have to put an opening tag and closing tag, and write your HTML within those tags.

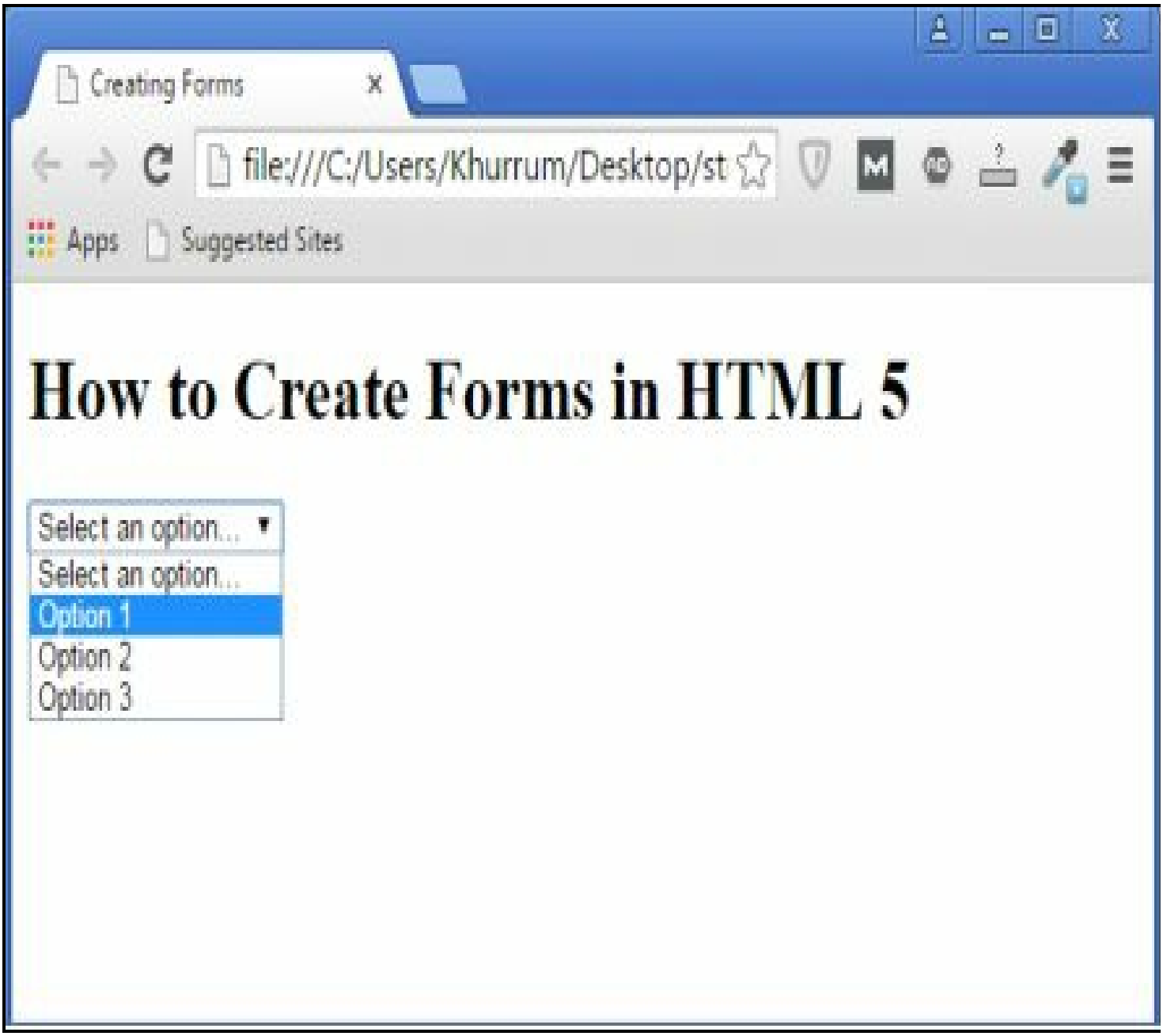
The Dropdown Box:

Dropdown boxes, menus, or lists are one of the most common types of tables featured on websites these days and, therefore, it requires special mention here. In order to

create a dropdown box, you have to use a <select> opening and closing tag within the form <form> element. Within the select element, you list all the options with the help of an option <option> tag.

Example 4:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Forms</title>
</head>
<body>
<h1>How to Create Forms in HTML 5</h1>

<form action=”bin/guestbook.php”>
<select>
<option>Select an option...</option>
<option>Option 1</option>
<option>Option 2</option>
<option>Option 3</option>
</select>
</form>
</body>
</html>
```



A dropdown menu is ready for you with an instruction at the top and three options to choose from. Remember that this is one of many types of dropdown menus you can create in HTML 5.

Checkboxes:

Checkboxes are perfect when you want your users to choose more than one option from a group of choices. You will use the <input> tag to create a checkbox, but the input

type will be *checkbox*.

Example 5:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Forms</title>
</head>
<body>
<h1>How to Create Forms in HTML 5</h1>

<form action=”bin/guestbook.php”>
<input type="checkbox" /> Apple
<input type="checkbox" /> Banana
<input type="checkbox" /> Orange
<input type="checkbox" /> Pineapple
</form>

</body>
</html>
```

What if you want to select only one option from the choices available? In that case, you are going to use a radio input type with any relevant attribute value.

Example 6:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
```

```
<title>Creating Forms</title>
</head>
<body>
<h1>How to Create Forms in HTML 5</h1>

<form action=”bin/guestbook.php”>
<input type="radio" name= “colour” /> Apple
<input type="radio" name= “colour” /> Banana
<input type="radio" name= “colour” /> Orange
<input type="radio" name= “colour” /> Pineapple
</form>

</body>
</html>
```

This is only the beginning of what you can use tables for in HTML 5. However, you need not worry because this is only a general overview of tables in HTML 5.

Exercise 5

Task:

Create a simple form with the value “Search this Website” and a simple comment section. Also, create a checkbox

containing four choices, but only allow one of those choices to be selected.

Solution:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Exercise</title>
</head>
<body>
<h1>Exercise 5</h1>

<form>
<input type=“text” value=“Search this Website”/>
<br><br>

<textarea>Give Your Feedback</textarea>
</form>

<form>
<input type="radio" name= “colour” />BMW
<input type="radio" name= “colour” />Ford
<input type="radio" name= “colour” />Jaguar
<input type="radio" name= “colour” />Volkswagen
</form>

</body>
```

```
</html>
```

Chapter 6: Understanding Hyperlinks

In this chapter, you will learn about:

- What a hyperlink is
- How to build hyperlinks
- How to use hyperlinks

Without hyperlinks, there would be no internet. Hyperlinks are what connect HTML and other resources on the internet. They are simple links which enable the users on your website to travel from one page to another on the same site or to another site. Your web page or website will be disconnected from the rest of the internet if it hasn't got any links.

You need the following to create a link:

- A Uniform Resource Locator (URL), commonly known as a web address. The web address usually starts with http://
- An anchor element <a> with an href attribute to make your link alive. The href attribute enables you to specify the URL you want your users to visit via the hyperlink.
- Apiece of text or content on the web page to link to.

You can use links to direct your users to any web page you want. For example, you may want the link to take your visitors to W3 Schools if you plan to refer to something about HTML 5 standards.

Types of Links:

There are two main types of links, absolute and relative, which are explained as follows.

Absolute Links:

Links which need the complete URL of the web page are called absolute links. They provide a standalone and complete path to the target web page. You can use an absolute link when you are linking to another website.

Relative Links:

Relatives link use a shortened URL or web address to point to the desired resource. For instance, the links used to point to a web page within same domain are relative links.

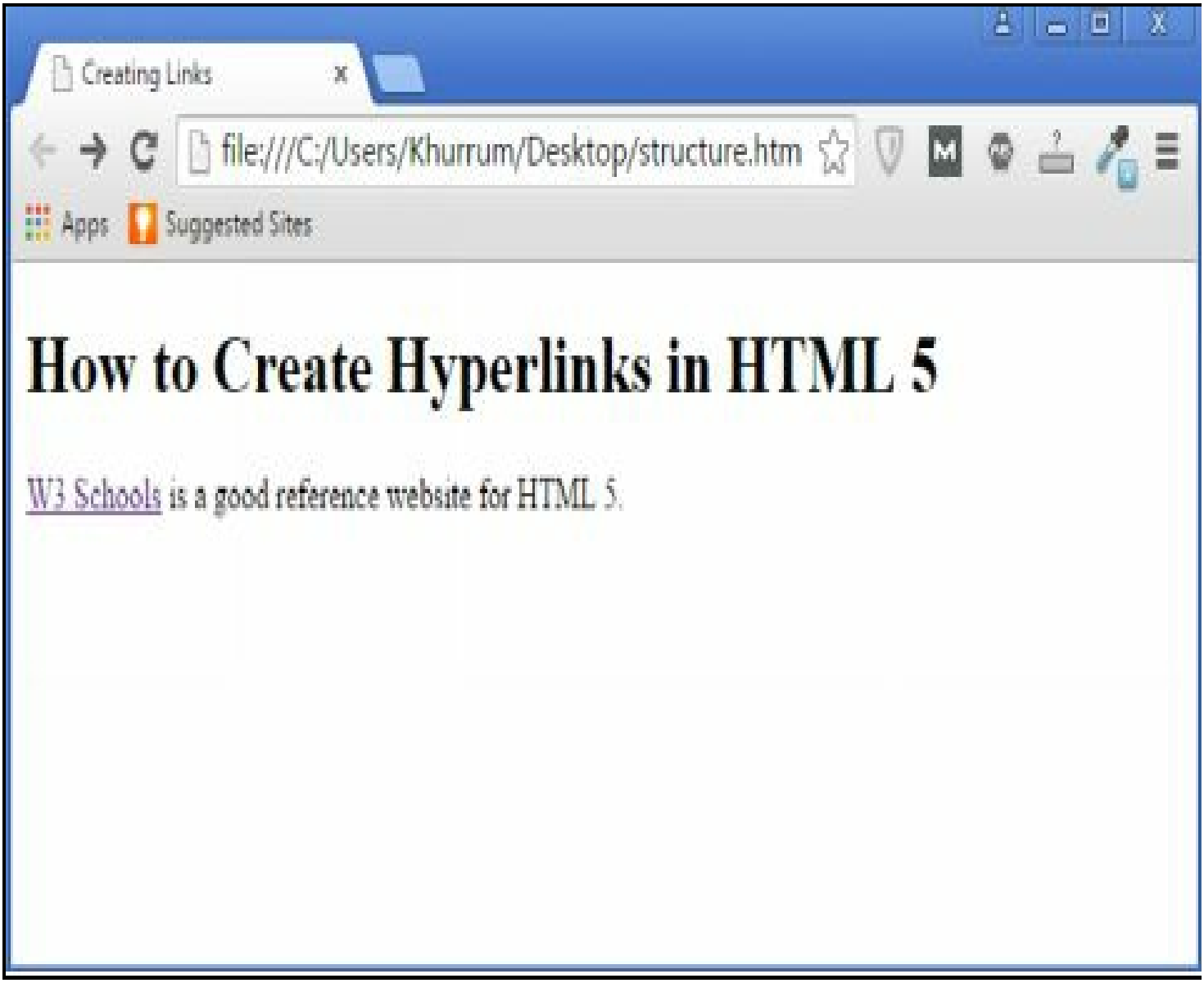
Creating a Link:

In the following example, you will specify the URL ***http://www.w3schools.com*** in the href attribute of the anchor element. Similarly, the text between the tags <a> will specify the link.

Example 1:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Links</title>
</head>
```

```
<body>
<h1>How to Create Hyperlinks in HTML 5</h1>

<p><a href= “http://www.w3schools.com”>W3 Schools</a> is a good reference website for HTML 5. </p>

</body>
</html>
```



In the above figure, W3 Schools is colored differently and underlined, which indicates that this is a hyperlink.

What if you want the link to open in a new tab or window? It's very simpla. Just add the ***target=_blank*** attribute to the target URL.

Example 2:

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

```
<meta charset= “utf-8”>
<title>Creating Links</title>
</head>
<body>
<h1>How to Create Hyperlinks in HTML 5</h1>

<p><a href=http://www.w3schools.com target="_blank">W3 Schools</a> is a good reference website for HTML 5.</p>

</body>
</html>
```

With this slight modification, the same link will open in the new tab.

Links to web pages on the same website:

As mentioned above, relative links can be used to point to a web page on the same website. Take a look at the following example:

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Links</title>
```

```
</head>
<body>
<h1>How to Create Hyperlinks in HTML 5</h1>

<p>This is an <a href="about.html"> about us </a>page of same website</p>

</body>
</html>
```

Now, if you click on “about us,” it will take you to the About Us page of the same website.

However, this method of inserting links will only work if all of your web pages are saved in the same directory. Sometimes pages reside in different directories. In that case, you will have to add additional information to the relative URL, as shown in the following example.

Example 4:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Creating Links</title>
</head>
<body>
```

```
<h1>How to Create Hyperlinks in HTML 5</h1>

<p>This is an<a href="../docs/about.html"> about us
</a>page of same website</p>

</body>
</html>
```

As you can see, we have added the address of the folder in which the about.html file resides. Notations in the example instruct the browser to take the following steps.

- Move up one folder from the folder in which the linking file resides.
- Find the folder named “docs.”
- Find the file named “about.html inside docs.”

Adding Links to information on the same page:

It is very easy to point to information on the same page. For instance, there are *Go to Top* tabs at the bottoms of some web pages which, once clicked, take you back to the top of the page. In that case, you will add an anchor element with a name attribute of “top.” Then, at the end of the page, you

will use the following URL to link back to top.

Back to Top

The hash sign (#) indicates that you're pointing to a location on the same page.

Example 5:

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

<title>Facts About United States</title>
<meta charset="utf-8" />

</head>
<body>
<a name="top" />

<h1>Hello, United States<h1>
<h2>Facts about a Beautiful Country <br /> Called the United States of America</h2>

<p>1. The United States gained independence from British rule on July 4, 1776. <br /> That was the day the United States of America emerged as an independent nation <br />
on the map.<br />
```

2. The United States has the largest economy in the world, followed by China and Japan.

3. The American flag’s official colors are “Old Glory Red,” “White,” and “Old Glory Blue.”</p>

<hr />

<h3>About the United States</h3>

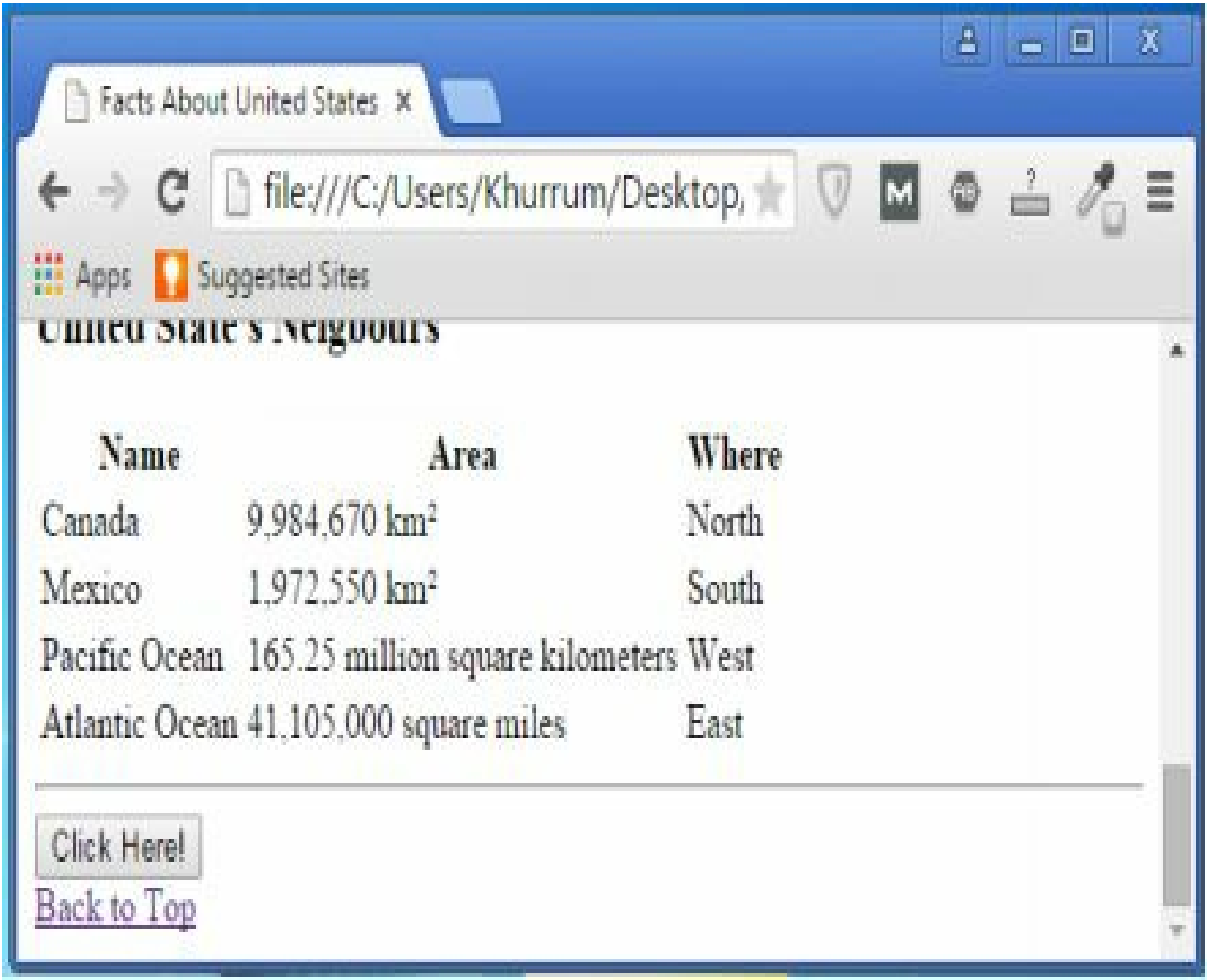
Founder:</Strong><a
href="http://en.wikipedia.org/wiki/George_Washington">Geo
Washington
Area:</Strong>9,857,306 km²
Population:</Strong>318.9 Million

<hr />

<h3>Top Three Facts</h3>

Total Number of States:</Strong>50
Primary Language:</Strong>English
National Sport:</Strong>Baseball


```
<hr />

<h3>The United State's Neigbours</h3>

<table>
<tr><th>Name</th><th>Area</th><th>Where</th></tr>
<tr><td>Canada</td><td>9,984,670 km²</td>
<td>North</td>
<tr><td>Mexico</td><td>1,972,550 km²</td>
<td>South</td>
<tr><td>Pacific Ocean</td><td>165.25 million square kilometers</td><td>West</td>
<tr><td>Atlantic Ocean</td><td>41,105,000 square miles</td><td>East</td>
</table>

<hr />

<form
action="http://en.wikipedia.org/wiki/George_Washington"
target="_blank">
<input type="submit" value="Click Here!" />
</form>

<a href="#top">Back to Top</a>

</body>
</html>
```



Now, if you click on the *Back to Top* tab, you will go straight back to the top of the page.

Exercise 6

Task:

Write a one line description of Washington DC and link the city name to a corresponding page on Wikipedia. Also add an image to your web page, make it a hyper link, and open

the link in a new tab.

Solution:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Exercise</title>
</head>
<body>
<h1>Exercise 6</h1>

<p><a
href="https://en.wikipedia.org/wiki/Washington,_D.C.">Wash
D.C.</a> is the capital of United States of America</p>

<br />

<a href="http://www.history.com/topics/us-
states/washington-dc" target="_blank"><img
src="washington.jpg" /></a>
</body>
</html>
```

Chapter 7: Working with Images

In this chapter, you will learn about:

- The role of images in HTML 5
- How to add images to web pages

Images not only contribute to the overall feel and look of the website, but can also be used to direct site navigation or deliver important information. The use of images requires careful consideration and selection; otherwise they lose their effectiveness.

Images can be anything from a logo to a clickable navigation aid. They may make your site beautiful or contain important information. There are many websites that make perfect use of images to deliver their messages, such as http://now.lincoln.com/. In fact, well used images are one of the most important components of any web design.

Image Types:

Before we discuss how to add an image to a web page, it is imperative to mention the type of images you can use in HTML 5. You can use many types of images, but we recommend using the following in your web pages.

- Joint Photographic Experts Group (JPEG)
- Graphics Interchange Format (GIF)
- Portable Network Graphics (PNG)

You also need to maintain a balance between image size and type in order for your website to load and render

properly.

Adding images to a Web Page:

When you have your image ready, use image <img>, an empty element, to determine where your images will go on the website. Also remember that <img> is a self-closing element and does not require separate opening and closing tags.

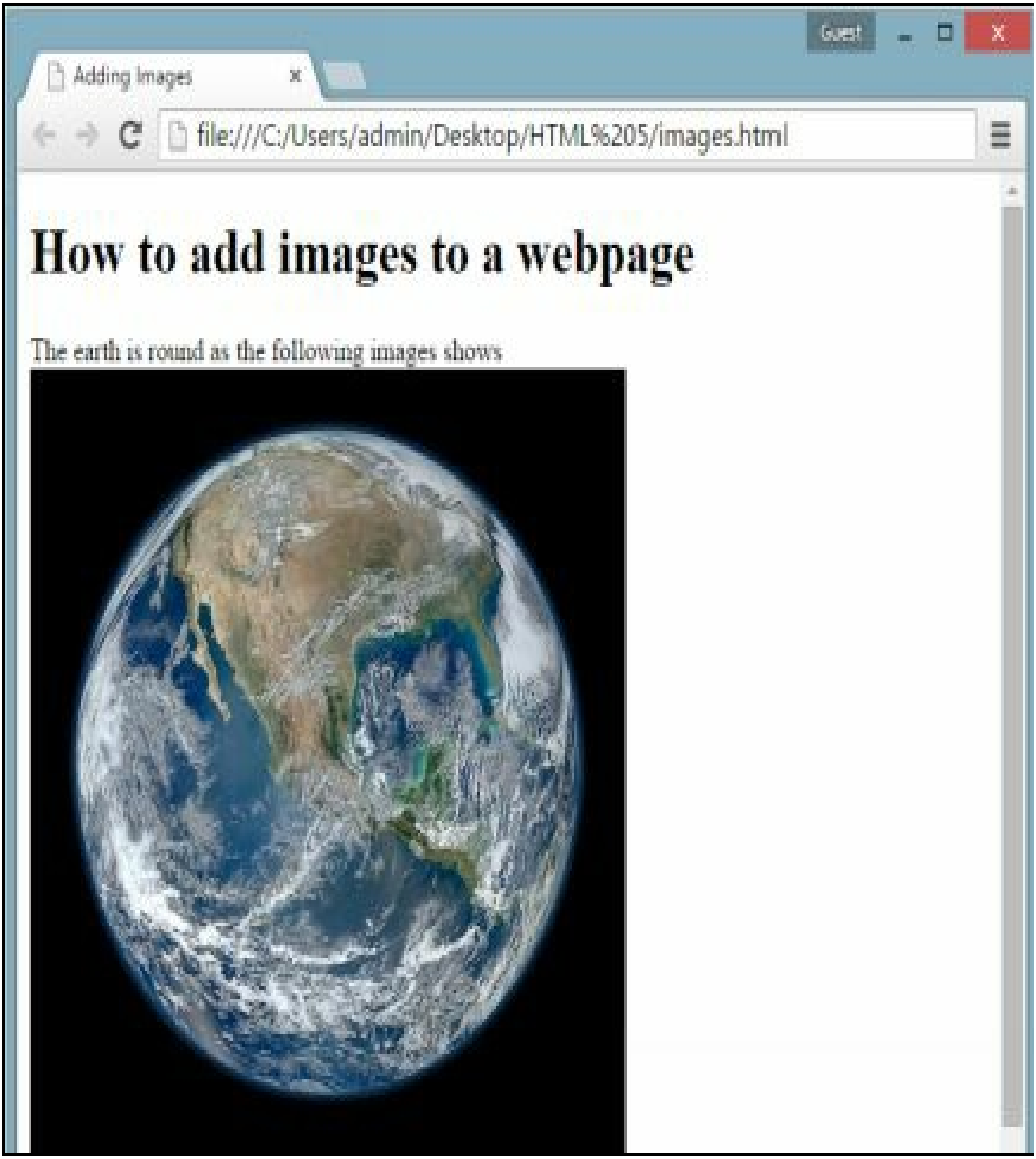
The following markup shows how you can add an image which resides in the same directory as that of the HTML 5 file to a web page.

Example 1:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Adding Images</title>
</head>
<body>
<h1>How to add images to a webpage</h1>

<p> The earth is round as the following images shows
<br>
<img src="roundworld.jpg" alt="The Earth is Round" />

</body>
</html>
```

In this example, the *source (src)* attribute specifies the location of the image to be displayed in the web page. Similarly, the *alt* attribute offers alternate information for the image if the user cannot somehow view it. It is good practice to add alt text because it is helpful for visitors who are visually impaired, have slow internet connections, or are accessing the website from a browser with limited visual capabilities. Some search engines also index images with the help of alt text.

The browser will display the above HTML 5 markup as follows:



Just like links, sometimes web developers store images in different directories. In this case, you have to mention the proper path of the image for the browser to display the image correctly. The following examples demonstrate how

you can do so.

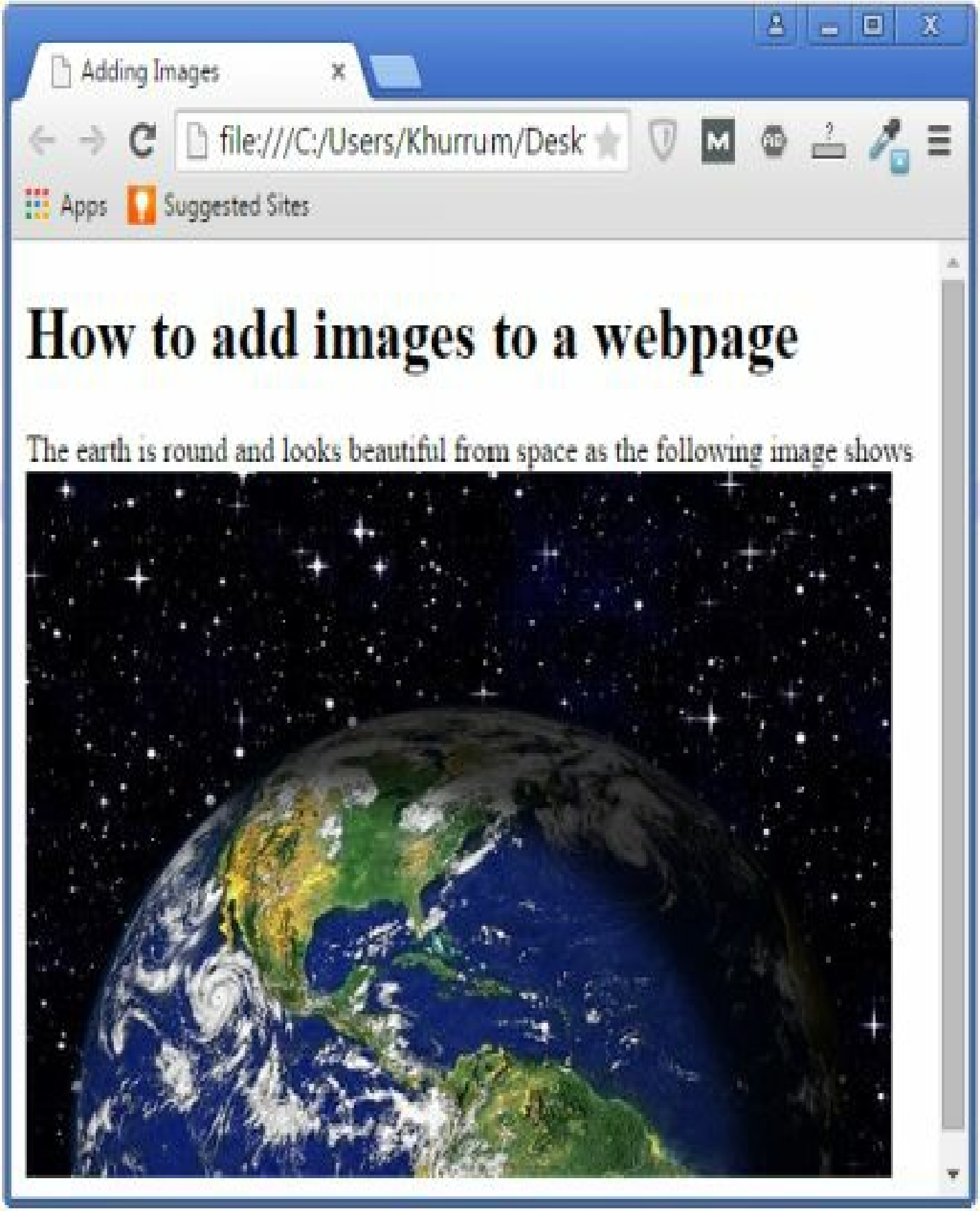
Example 2:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Adding Images</title>
</head>
<body>
<h1>How to add images to a webpage</h1>

<p>The earth is round and looks beautiful from the space as the following image shows <br>
<img src="images/earth.jpg" alt="The Earth is Round" title="Earth from Space"/>

</body>
</html>
```

In this example, the src attribute is instructing the browser to:

- Find the folder named “images”
- Find the “earth.jpeg” image in the images folder.

In this case, the browser will display the following image:



The Titled Image:

In addition to alt text, you should always use give images a title. A title is a text which all search engines, including Chrome and Firefox, display when you hover the mouse over the image. You can add a title through a title attribute in <img> tag, just like src and alt. In the following example, we will add a title to an image.

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Adding Images</title>
</head>
<body>

<h1>How to add images to a web page</h1>

<p> The earth is round as the following image shows
<br>
<img src="roundworld.jpg" alt="The Earth is Round" />

</body>
</html>
```

Guest

Adding Images

file:///C:/Users/admin/Desktop/HTML%205/images.html

How to add images to a webpage

The earth is round as the following images shows

Making Images Clickable:

It is possible to make any image a link. This is an easy task as you will be using the same <a> element to add links to images. However, in this case, you will replace the text between <a> with an image. Take a look at the following example:

Example 4:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Adding Images</title>
</head>
<body>
<h1>How to add images to a webpage</h1>

<p> The earth is round and looks beautiful from space as the following image shows <br>
<a href="http://science.nationalgeographic.com/science/earth/" target="_blank"><img src="images/earth.jpg" alt="The Earth is Round" title="Earth from Space"/></a>

</body>
</html>
```

This markup has made our image a link which, once clicked, will open a particular page of National Geographic's website in a new tab.

Adding Images

Earth, Earth Sciences Info

file:///C:/Users/Khurrum/Desk

Apps Suggested Sites

How to add images to a webpage

The earth is round and looks beautiful from space as the following image shows

...nationalgeographic.com/.../ea...

Resizing Images:

Finally, you can change the width and height of images in HTML 5 by using the *height* and *width* attributes in <img> self-closing tags. The image in the above example has the width and height of 500 pixels and 300 pixels (500X300), respectively. We will change the dimensions to 600X400 pixels using our HTML 5 markup.

Example 5:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Adding Images</title>
</head>
<body>
<h1>How to add images to a webpage</h1>

<p> The earth is round and looks beautiful from space as the following image shows <br>
<img src="images/earth.jpg" alt="The Earth is Round" title="Earth from Space" width="600" height="400"/>

</body>
</html>
```

Adding Images

file:///C:/Users/Khurrum/Desktop/HTML%205/images.html

Apps Suggested Sites

How to add images to a webpage

The earth is round and looks beautiful from space as the following image shows

As you can see, we have resized our image using width and height attributes; you can do the same with all images you upload to your web page.

Exercise 7

Task:

Add an image which resides in a different directory from that of your HTML 5 file. Adjust the width and height to 400pixels each. Also include some alt text and a title for the image.

Solution:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Exercise</title>
</head>
<body>
<h1>Exercise 7</h1>
```

```
<img src="exercise/statue.jpg" alt="This is Statue of Liberty" title="Statue of Liberty" width="400" height="400"/>

</body>
</html>
```

Chapter 8: Working with Video and Audio

In this chapter, you will learn how to:

- Embed video and audio in HTML 5 from other web pages
- Embed video in HTML 5 from a local computer
- Embed audio in HTML 5 from a local computer

With the passage of time, different types of media, including video and audio, are becoming more and more important. Today, the web is not only a place to access content and images, but has also become a go-to resource to watch live videos, conduct video conferences and video calls, and so on.

Unlike older versions of HTML, which required the help of various plug-ins to play audio and video, HTML 5 adds standard ways of plugging media. HTML 5 has eliminated many problems associated with playing media in earlier versions, such as finding the right plug-ins for your browser to run a specific kind of media.

Embedding Media from another webpage:

It is ridiculously easy to add any kind of content from other web pages to your page, and the same is true for media (both audio and video). For that matter, we use an iframe <iframe> element. Just like <a> and <img> elements, we use an *src* attribute inside the <iframe> element in order to specify the URL of the video we want to play. Take a look at

the following example:

Example 1:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Adding Media</title>
</head>

<body>
<h1>How to add Media to a webpage</h1>

<p> Vidoe on Twinkle Twinkle Little Star<br>
<iframe
src="https://www.youtube.com/embed/yCjJyiqpAuU"
width="560" height="315" frameborder="0"
allowfullscreen></iframe>

</body>
</html>
```


Adding Media
file:///C:/Users/Khurrum/Desktop/media.html
Apps
Suggested Sites
How to add Media to a webpage
Vidoe on Twinkle Twinkle Litte Star
Twinkle Twinkle Little Star
Super
Simple
Learning

In this example:

- The *src* attribute specifies the source or URL of the video you are trying to put in.
- The *width* and *height* attributes specify the width and height of video frame.
- The *Allowfullscreen* attribute gives viewers the option to watch video on full screen.

You do not need to write all the boring code in HTML 5 to get your video. You can simply copy the entire code and paste it into your HTML 5 editor. For instance, open any YouTube video you want to embed, go to the ***Share*** tab below the title, and then click ***Embed***. You will get a code very similar to the one used in the above example. Just copy the code and put it where you want your video to play.

Embedding video and audio from your computer:

So far, we have learned how to embed video and audio from other web pages. Now we are going to see how to embed video and audio residing somewhere in your local computer. The process is somewhat the same, as you just have to replace the <iframe> element with the <video> and <audio> elements for embedding video and audio, respectively.

Embedding Video:

In the case of a video, you will use the following markup:

Example 2:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Adding Media</title>
</head>

<body>
<h1>How to add Media to a webpage</h1>

<p> Vidoe on Twinkle Twinkle Little Star<br>
<video src="rhymes.mp4" width="560" height="315"
controls></video>

</body>
</html>
```

Adding Media

file:///C:/Users/Khurrum/Desktop/media.html

Apps Suggested Sites

How to add Media to a webpage

Vidoe on Twinkle Twinkle Litte Star

We have added a video in HTML 5 without using any annoying plug-ins. Note that there is also a control attribute in the markup which allows you to add controls like play, pause, and volume. Although the only value for this attribute is *controls*, no value is strictly required in HTML5.

Other attributes you can use in place of *controls* are:

- ***autoplay***: plays the video as soon as it is loaded
- ***loop:*** replays the video automatically from the start once it has finished
- ***preload:*** tells the browser whether it should preload video or not. You don't need this attribute if the *controls* attribute is already there.

It is always better to add width and height; otherwise there can be rendering issues, depending on the browser you are using.

Embedding Audio:

While embedding audio files from your local computer, you will replace the video <video> element with the audio <audio>. The remaining markup will remain more or less the same.

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Adding Media</title>
```

```
</head>

<body>
<h1>How to add Media to a webpage</h1>

<p> The "Firefiles" from You Can Unlearn Guitar<br>
<audio src="audio.mp3" controls autoplay></audio>

</body>
</html>
```

Media Formats in HTML 5:

It is worth mentioning some of the most commonly used audio and video formats in HTML 5. We recommend using these formats, rather than others, to avoid rendering and incorrect loading issues.

Major Audio Formats:

The following are major audio formats you can use in HTML 5.

- Ogg Vorbis (.ogg)
- MP3 (.mp3)

- Waveform Audio File Format, or WAVE (.wav)

Major Video Formats:

The following are some of the major video formats which HTML 5 recognizes.

- Ogg Theora (.ogg)
- MP4 (.mp4)
- VP8/9 (.webm)

Exercise 8

Task:

Add a video from YouTube to your web page and adjust the screen width and height to 500 pixels and 300 pixels, respectively. Also download a video to your Desktop and add it to your web page. Add controls and autoplay to the second video.

Solution:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Exercise</title>
</head>
<body>
<p>The Declaration of Independence</p>
```

```
<iframe
src="https://www.youtube.com/embed/jYyttEu_NLU"
width="500" height="300" frameborder="0"
allowfullscreen></iframe>

<br /><br />

<p>Johny Johny Yes Papa</p>
<video src="johny.mp4" width="500" height="300"
controls autoplay></video>
</body>
</html>
```

Chapter 9: Web Page Layout

In this chapter, you will learn the answers to:

- What is a navigation <nav> element?
- What is a header <header> element?
- What is a division element?
- What is a footer <footer> element?

So far, we have discussed how to add lists, tables, images, links, and media to web pages using HTML 5. However, we need to learn about a few more elements which are essential to HTML 5. These elements will help you to further boost the structure and semantics of your website, making it easier for web browsers to understand the markup and render your website correctly.

The Navigation <nav> Element:

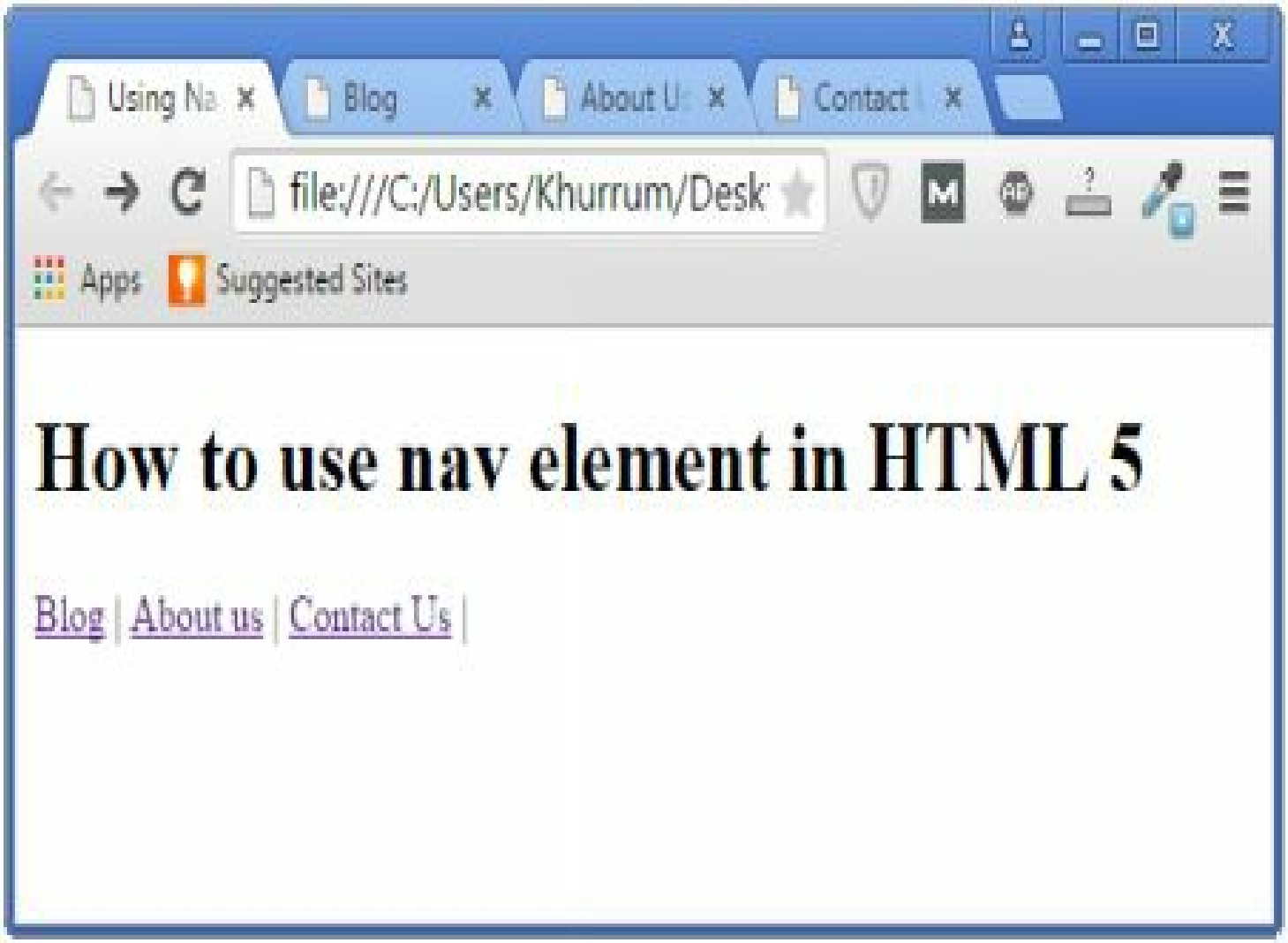
The <nav> element is exclusive to HTML 5 and allows you to group different links together, resulting in extra structure and stronger semantics which may help screen readers and web browsers. Although the definition of the <nav> element is pretty simple, no one has been able to fully explain how to use it properly. However, a simply markup using a <nav> element is as follows:

Example 1:

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

```
<meta charset= “utf-8”>
<title>Using Navigation Element</title>
</head>

<body>
<h1>How to use nav element in HTML 5</h1>

<nav>
<a href="blog.html">Blog</a> |
<a href="about.html">About Us</a> |
<a href="contact.html">Contact Us</a> |
</nav>

</body>
</html>
```

The web browser will represent the above markup as follows:



Web developers normally use navigation elements for main navigation bars or just above the footer. There are also other places where you can use this element, such as Table of Contents, Previous/Next Buttons, Search Forms, and Breadcrumbs.

The Header <Header> Element:

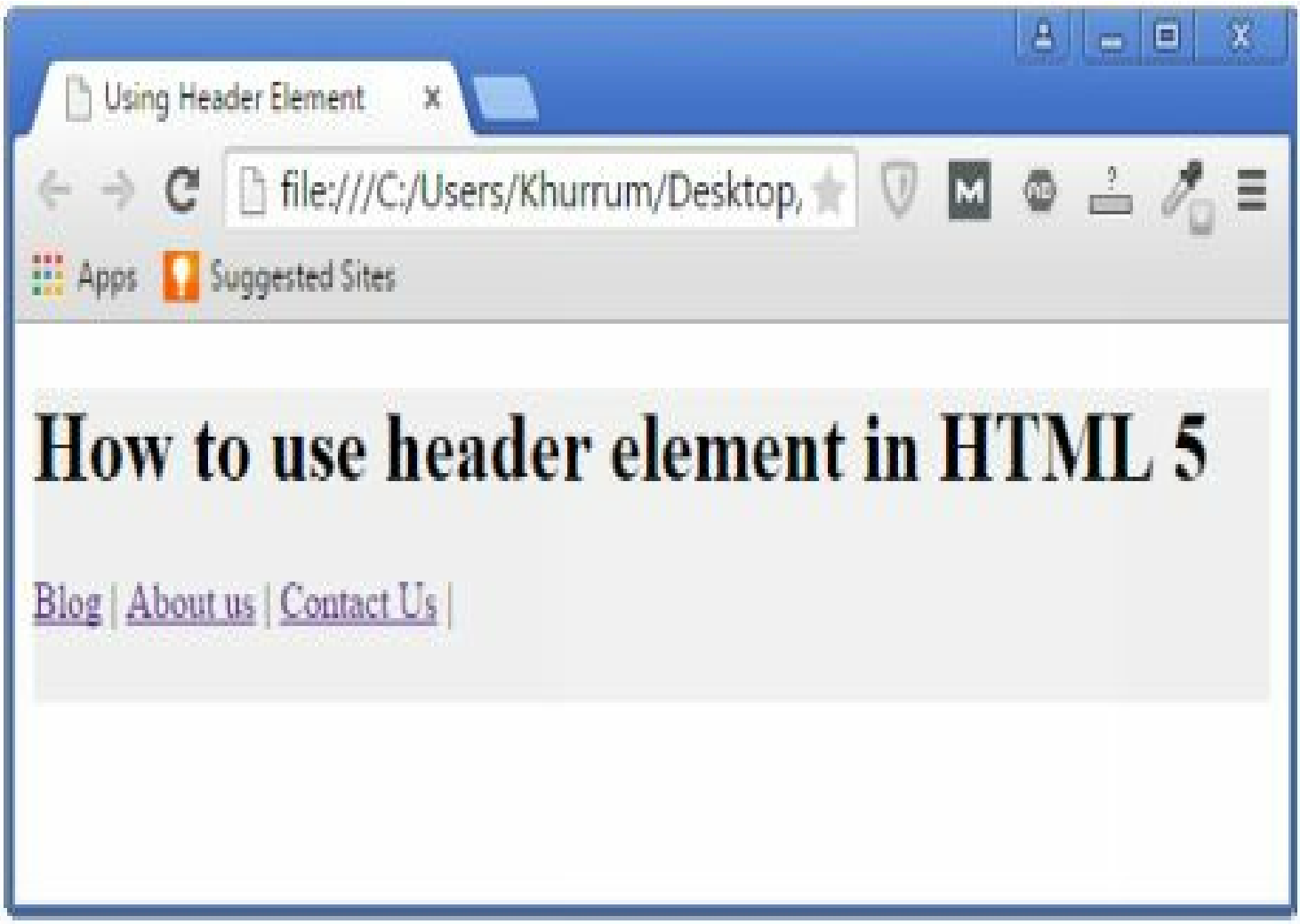
Just like the navigation element, the header element is exclusive to HTML 5. The header element is actually a "group of introductory or navigation aids." It is mostly used at the beginning or top of the web page, but you can also use it at the start of any new section or division. Similarly, it usually contains an h1 introductory heading, but you can

also insert logos, navigation menus, or a search form. A simple markup for a header element is shown in the following example:

Example 2:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Using Header Element</title>

<style>
#head {
        background-color: #F1F1F1;
        height: 100px;
        width: 100%;
}

</style>

</head>

<body>

<header id="head">
<h1>How to use header element in HTML 5</h1>
<nav>
<a href="blog.html">Blog</a> |
```

```
<a href="about.html">About us</a> |
<a href="contact.html">Contact Us</a> |
</nav>
</header>

</body>
</html>
```



Now we have a header at the top of our web page. The grey background, as well as the height and the width of the header, are a result of certain CSS codes which we shall

study in the CSS section of this course.

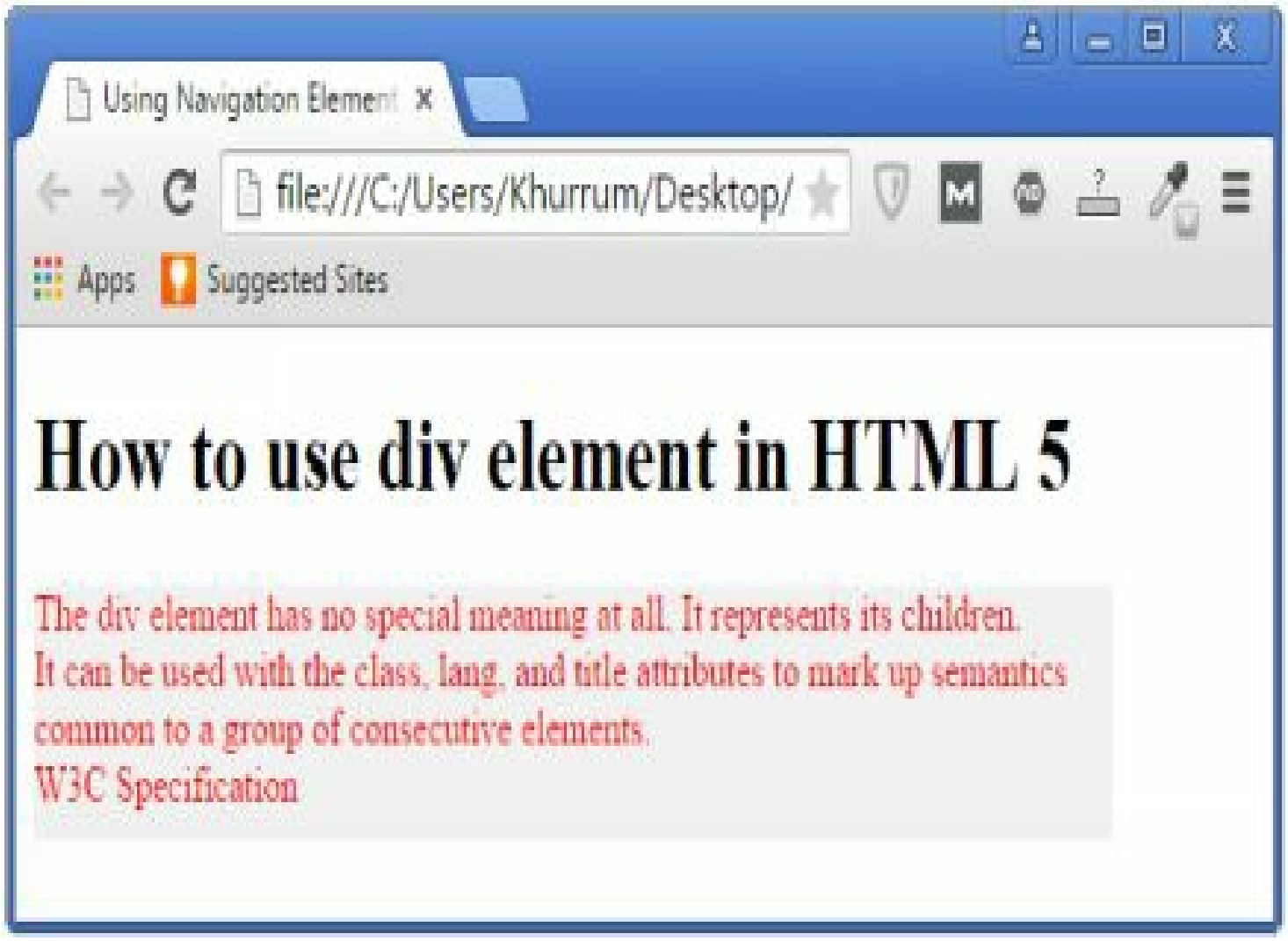
The Division Element:

The division element is perhaps one of the most important and widely used elements in HTML 5. The division element does not represent anything like the paragraph <p> or image <img> elements do; rather, it is a “generic container” which maintains the flow or content and provides extra structuring and semantics. For instance, if you want to style a particular chunk of content, the will do the job perfectly. Take a look at following example:

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Using Header Element</title>

<style>
#wrapper {
        background-color: #F1F1F1;
        height: 80px;
        width: 500px;
        color: red;
}
</style>
```

```
</head>

<body>
<h1>How to use div element in HTML 5</h1>

<div id="wrapper">

<p>The div element has no special meaning at all. It represents its children.<br />
It can be used with the class, lang, and title attributes to mark up semantics<br />
common to a group of consecutive elements.</br>
W3C Specification</p>

</div>

</body>
</html>
```



Similarly, you can use to create and style different sections, divisions, headers, footers, and almost anything else you like.

The Footer <footer> Element:

Finally, another important *semantic element* of HTML 5 is the footer <footer>. As the name suggests, this element defines the footer of a page or section. This particular element should contain as much information as possible about its containing element. Normally, it contains:

- Contact information
- Authorship and copyright information
- Back to top links (see previous chapter)

- Related documents
- Site maps

The following example demonstrates how to use footers in HTML 5.

Example 4:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Using Header Element</title>

<style>
#wrapper {
        background-color: #F1F1F1;
        height: 70px;
        width: 550px;
        color: red;
}
</style>

</head>

<body>
<h1>How to use footer element in HTML 5</h1>
```

```
<div id="wrapper">

<p>The footer element represents a footer for its nearest ancestor sectioning content.<br />
A footer typically contains information about its section such as who wrote it, links to<br />
, copyright data, postal and email addresses and the like.</p>

</div>

<footer>

<p>Witten by: David Sinclair<br />
Visit us at: xyz.com, </br />
mail to: <a href="mailto:david@xyz.com">david@xyz.com</a>.<br />
P.O. Box No. 111, New York, <br />
United States of America</p>

</footer>

</body>
</html>
```

Remember that footers will not magically go to the bottom of the page. You still have to style them using CSS as you would do for all other elements. It only has a semantic value, as it tells the browser that the data inside the element is footer data.

Exercise 9

Task:

Create a navigation bar containing four different links. The first link will lead you to Google, the second will go to the home page of www.example.com, the third one will go to Twitter, and the fourth one will take you to the home page of

the website of your choice.

Solution:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Exercise</title>
</head>
<body>
<h1>Exercise 9</h1>

<nav>
<a href="https://www.google.com">Google</a> |
<a href="http://example.com/">Example.com</a> |
<a href="https://www.facebook.com/">Facebook</a> |
<a href="http://www.imdb.com/">Internet Movie Database</a>
</nav>

</body>
</html>
```

Chapter 10: Understanding Quotations, iframes, and Entities

In this chapter you will learn about:

- HTML Quotation and Citation Elements
- iframe element
- HTML Entities

We have reached the last chapter of this book. We have covered a lot of things which you need to get going with HTML 5. In the last chapter, we will discuss a few more things which can help you make your web page a little more attractive, and the structure and semantics stronger and easier for web browsers to understand.

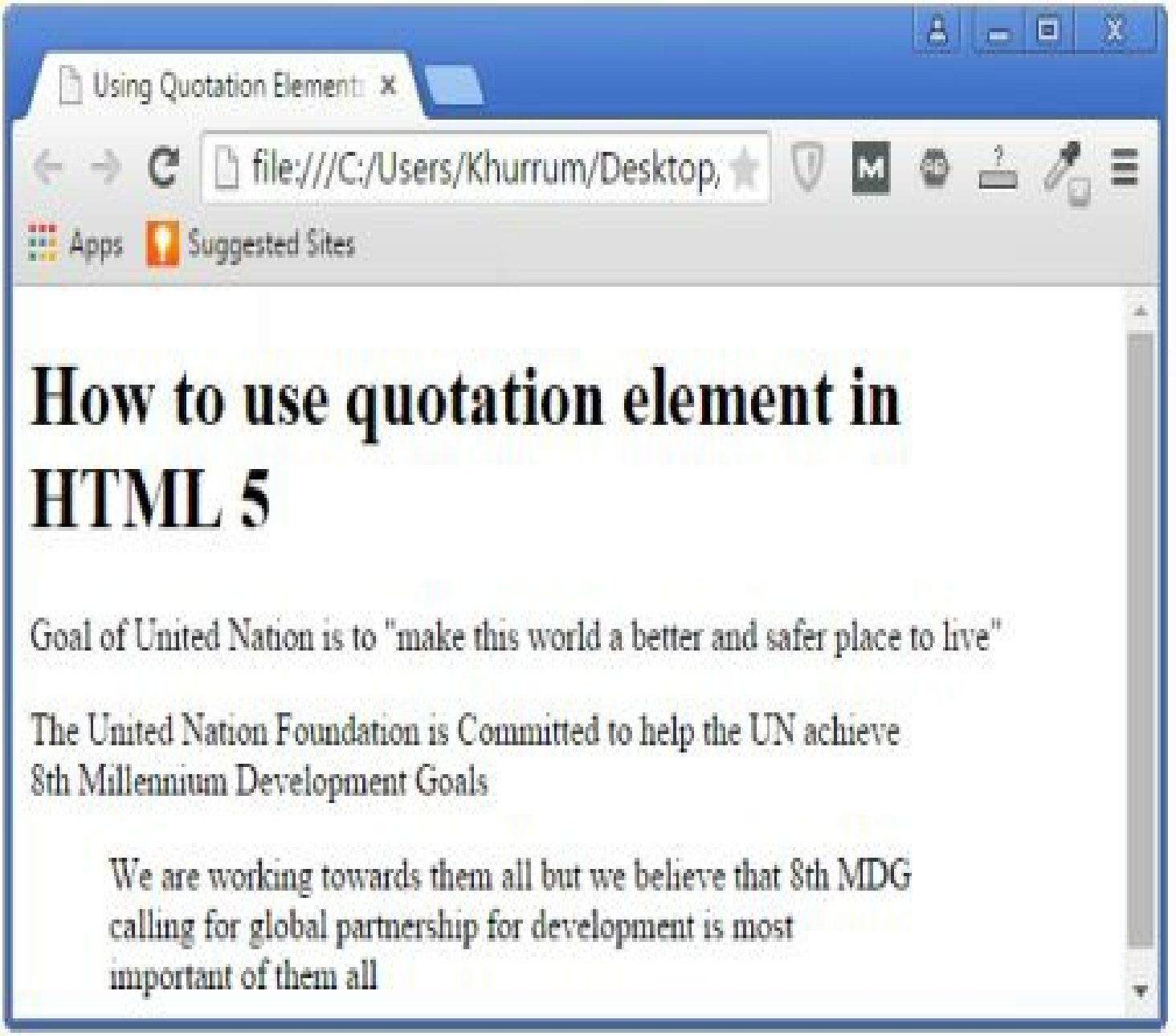
Quotation and Citation Elements:

Quoting has been well accommodated in HTML 5 given the deep roots it has in academics. The two most important quotation elements in HTML 5 are the q <q> and blockquote <blockquote> elements. The q and blockquote elements indicate short quotations within a non-quoted paragraph and represent a section which is quoted from another source, respectively. Let us see how to use these elements in HTML 5 markup.

Example 1:

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

```
<meta charset= “utf-8”>
<title>Using Quotation Elements</title>
</head>

<body>
<h1>How to use quotation element in HTML 5</h1>

<p>Goal of United Nation is to <q>make this world a better and safer place to live</p></p>

<p>The United Nation Foundation is Committed to help the UN achieve its<br/>
8th Millennium Development Goals
<blockquote>We are working towards them all but we believe that 8th MDG<br />
calling for global partnership for development is most<br />
important of them all</blockquote></p>

</body>
</html>
```

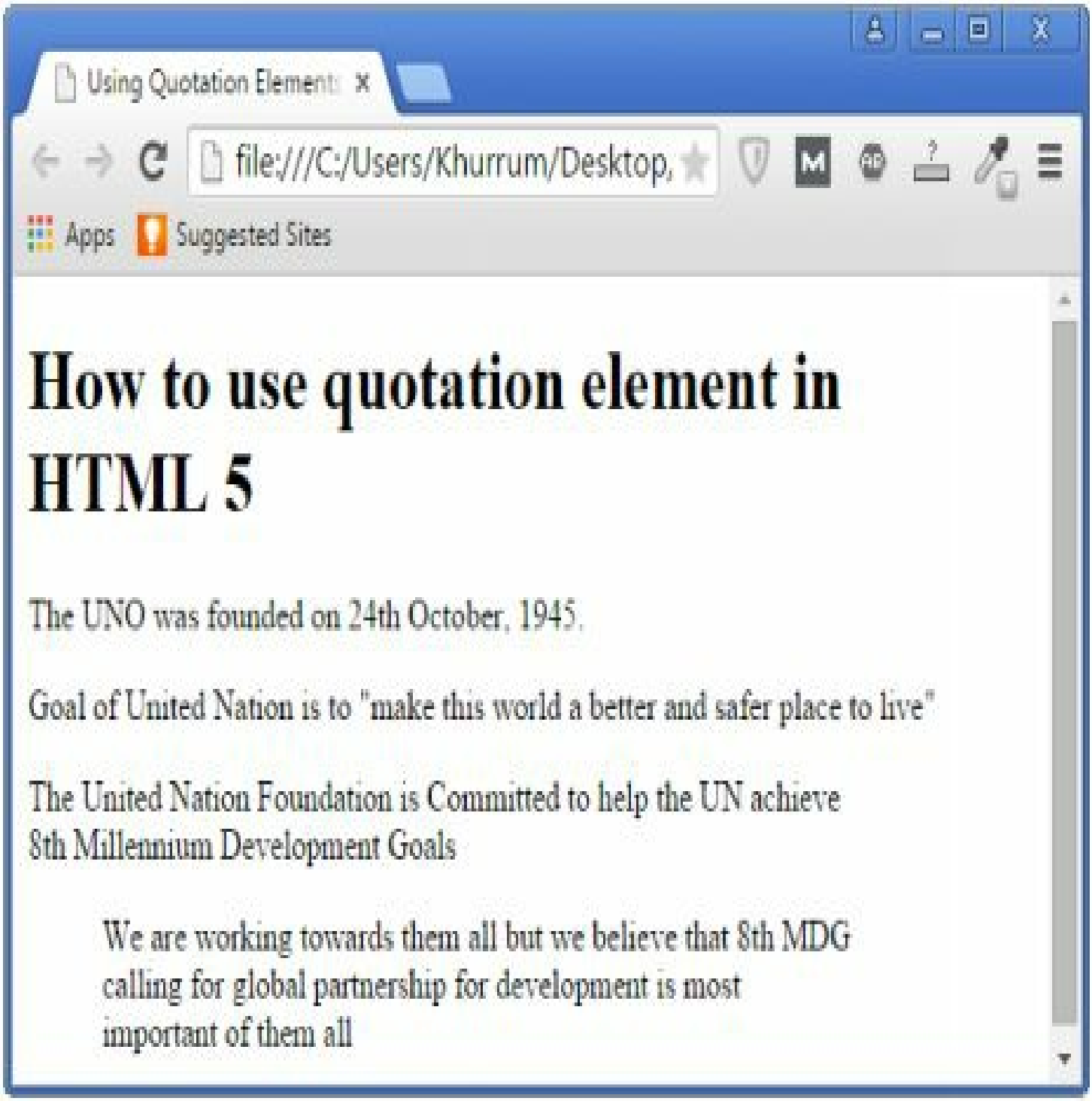


Similarly, there is an abbreviation <abbr> element which you can use to make abbreviations. Abbreviations are quite useful as they provide valuable information to the users and search engines as well as browsers. You have to always use an <abbr> element in combination with a title attribute as shown in the following example:

Example 2:

```
<!DOCTYPE HTML>
```

```
<html>
<head>
<meta charset= “utf-8”>
<title>Using Quotation Elements</title>
</head>

<body>
<h1>How to use quotation element in HTML 5</h1>

<p>The <abbr title= "The United Nations Organizations">UNO</abbr> was founded on 24th October, 1945.</p>

<p>Goal of United Nation is to <q>make this world a better and safer place to live</p></p>

<p>The United Nation Foundation is Committed to help the UN achieve<br/>
8th Millennium Development Goals
<blockquote>We are working towards them all but we believe that 8th MDG<br />
calling for global partnership for development is most<br />
important of them all</blockquote></ p>

</body>
</html>
```



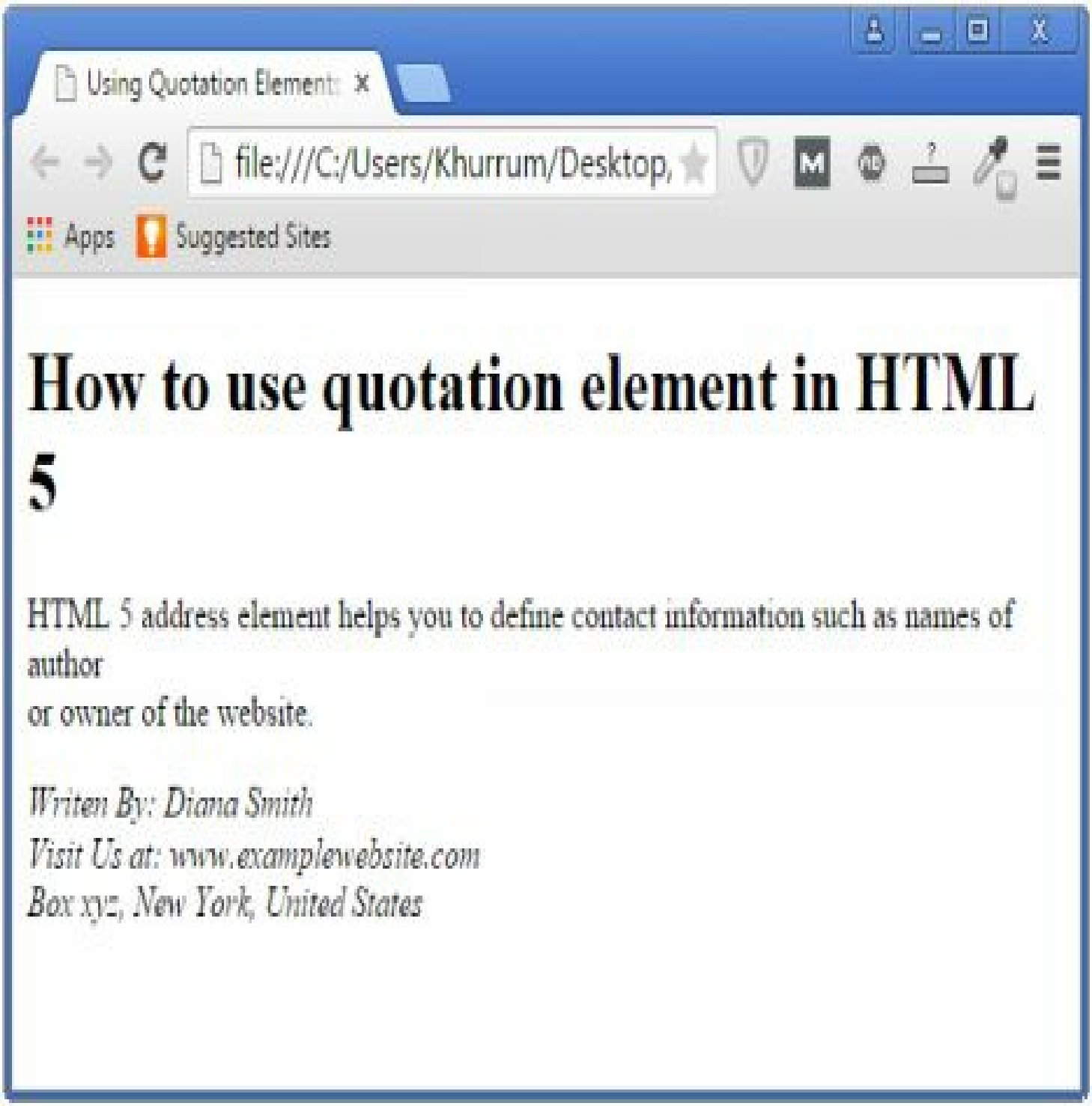
Adding an Address to a web page:

In the previous chapter, we added an address to the footer of our web page. The address element <address> is a perfect option for adding contact information on the

website, necessary for engaging your users and customers. It is rather simple to put in your contact information using address elements, as the following example demonstrates.

Example 3:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Using Quotation Elements</title>
</head>

<body>
<h1>How to use quotation element in HTML 5</h1>

<p>HTML 5 address element helps you to define contact information such as names of authors<br /> or owner of the website.</p>

<address>
Written By: Diana Smith <br />
Visit Us at: www.examplewebsite.com<br />
Box xyz, New York, United States
</address>

</body>
</html>
```

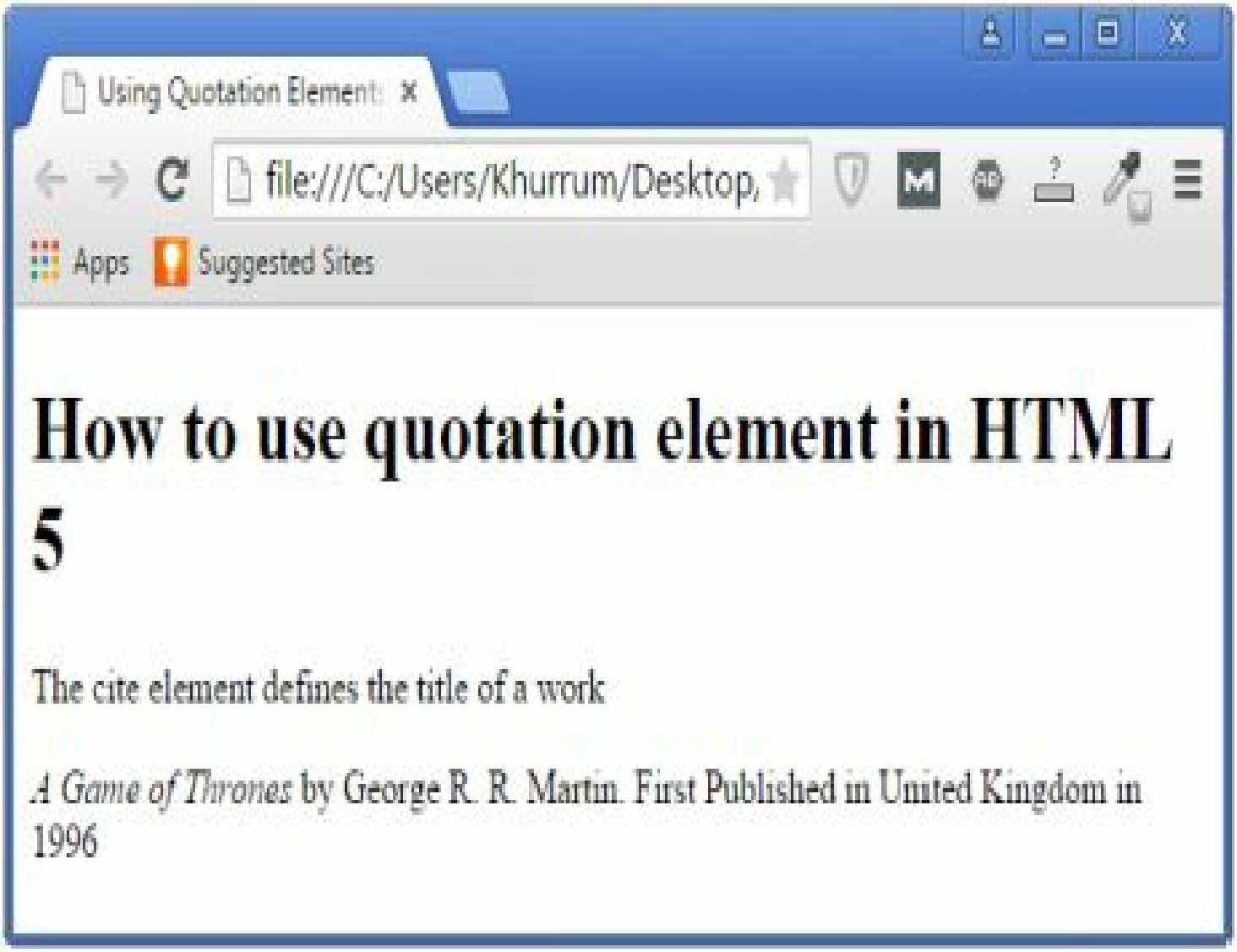


The last important element falling under the Quotation and Citation category is the Cite <cite> element. This element is mainly used to define the title of a literary work, but you can use it to define the title of anything you like. Most

browsers will display cite elements in italics.

Example 4:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Using Quotation Elements</title>
</head>

<body>
<h1>How to use quotation element in HTML 5</h1>

<p>The cite element defines the title of a work</p>

<p><cite>A Game of Thrones</cite> by George R. R. Martin. First Published in United Kingdom in 1996</p>
</body>
</html>
```



The iframe Element:

HTML 5 specifications refer to the iframe <iframe> element as a *nested browsing context*. In simple words, it helps you to pick the content from another source or web page and embed it in your own web page without the help of another document. You simply have to use a src attribute to define the location of the content you want to embed.

The following example explains how to add an inline frame using iframes.

Example 5:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Using iframe Element</title>
</head>

<body>
<h1>How to use iframe element in HTML 5</h1>

<p>iframe element helps you to embed content from another source</p>

<iframe src= "https://www.w3.org/wiki/HTML/Elements/iframe" width= "300" height= "300"</iframe>

</body>
</html>
```

Using iframe Element

file:///C:/Users/Khurrum/Desktc

Apps Suggested Sites

How to use iframe element in HTML 5

iframe element helps you to embed content from another source

You can use iframes to embed almost anything you like, such as videos. Similarly, you can also use the iframe element as target frames for links.

HTML 5 Entities:

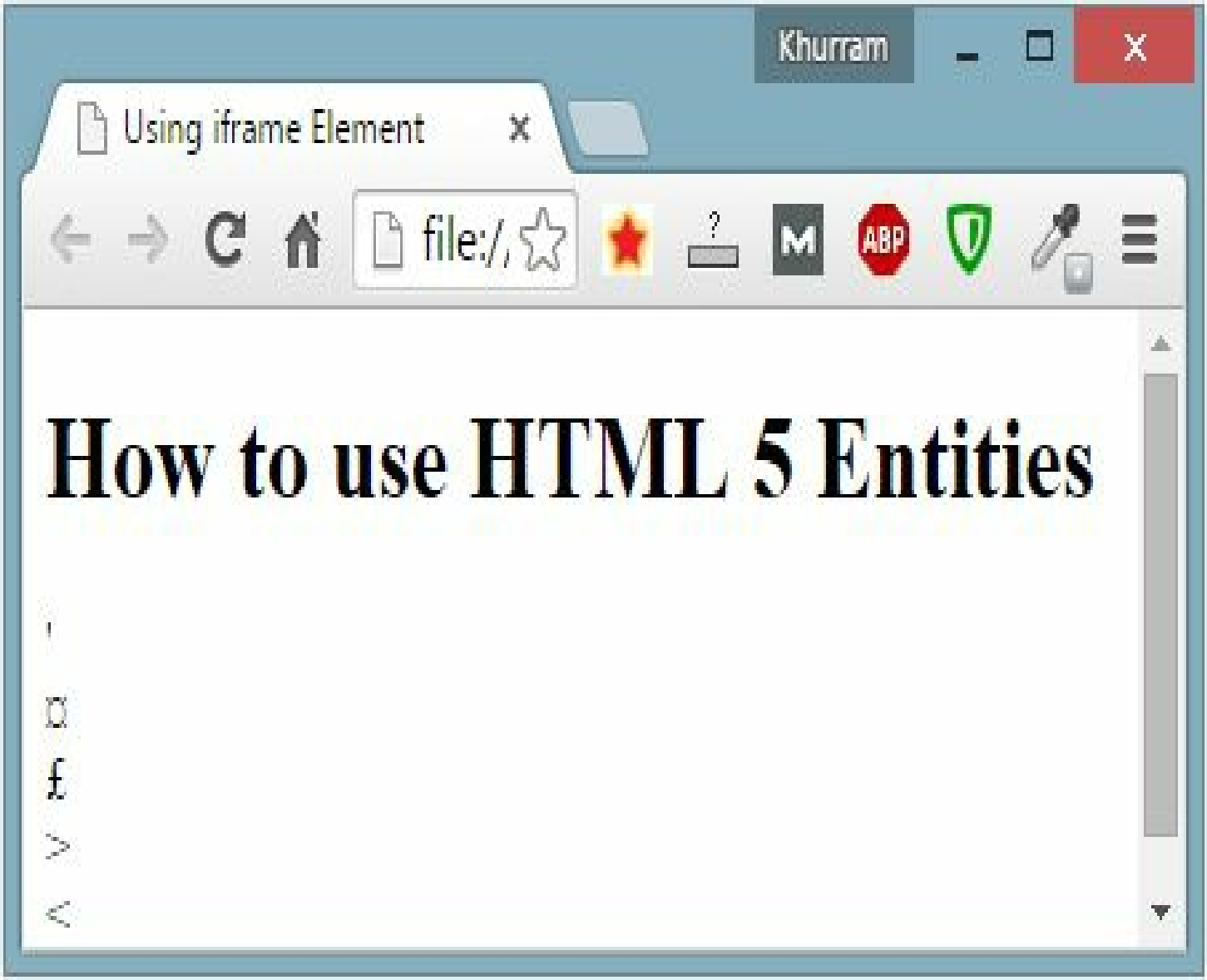
Finally, it is imperative to talk about HTML 5 entities. Entities are used to add those characters to your web page which you cannot add simply by typing text. For instance, you need to use certain entities to insert greater than > and less than < symbols because the browser can mistake them for markups. Such characters are also known as reserved characters and must be replaced with entities.

Remember that all entities start with an ampersand (&) followed by the entity name and a semicolon. For instance, you will use the following markup if you want to insert <, >, quotation marks, and currency or pound symbols in your web page.

Example 6:

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset= “utf-8”>
<title>Using Element Entities</title>
</head>

<body>
<h1>How to use HTML 5 Entities</h1>
```

```
<p>&amp;apos;<br />
&amp;curren;<br />
&amp;pound;<br />
&amp;gt;<br />
&amp;lt;</p>

</body>
</html>
```



You can also use numerical codes for the entities to appear on your web page. HTML 5 has countless reserve characters and it is impossible to remember the names or codes of all of them. In this regard, the following are some reference links to all of the HTML 5 entities.

http://dev.w3.org/html5/html-author/charref

http://www.w3schools.com/charsets/ref_html_symbols.asp

http://edoceo.com/notabene/xhtml-entities

Exercise 10

Task:

Embed a page from Wikipedia on Baseball in your web page using an iframe element.

Solution :

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

<body>

<h1>Exercise 10</h1>

<iframe src="https://en.wikipedia.org/wiki/Baseball"
width="300" height="300"></iframe>

</body>
</html>
```

Other Books by the Author

JavaScript Programming: ABeginners Guide to the Javascript Programming Language
http://www.linuxtrainingacademy.com/js-programming

If you've attempted to learn how to program in the past, but hadn't had much success then give *JavaScript Programming* a try. It will teach you exactly what you need to know about programming in the world's most widely used scripting language in existence today. It will start you at the beginning and allow you to build upon what you've learned along the way.

SQL: Learn the Structured Query Language for the Most Popular Databases including Microsoft SQL Server, MySQL, MariaDB, PostgreSQL, and Oracle
http://www.linuxtrainingacademy.com/sql-book

Additional Resources

Projects in HTML5
http://www.linuxtrainingacademy.com/html5-projects

In order to master a technique in web development or any artistic field, the best approach is one that's hands-on. This course allows you to not only learn concepts, but also apply them right away to build your own projects from scratch. That's why this course guides you through the process of actually creating a variety of games, apps, and sites using HTML5 and JavaScript.

Responsive Web Design AND Web Development with HTML5
http://www.linuxtrainingacademy.com/responsive

Are you wondering how the best web designers and web developers build powerful, modern and good-looking websites?

This is exactly the course you've been waiting for!

Learning to build your own websites with HTML5 and CSS3 is no rocket science — it's super easy, and you should start learning it today! In this course, you will learn web design AND web development step-by-step using the latest web technologies HTML5, CSS3 and jQuery.

Learn To Build Beautiful HTML5 And CSS3 Websites In 1 Month
http://www.linuxtrainingacademy.com/html5-in-a-month

This HTML & CSS is a project-based course that will show anyone, even a complete beginner how to build beautiful websites. Through a series of video lessons and tutorial you'll build 4 different websites - a landing page, blog template, magazine layout page, and your very own portfolio page to show off your work.