Workbook

Web Engineering

(CT – 362)



**Name:Haseebullah**

**Roll No:CT-19003**

**Batch:2019**

**Year:3rd year**

**Department:CS&IT**

Workbook

Web Engineering

(CT – 362)

Prepared by

Mrs. Nazish Irfan

I.T Manager,

Department of Computer Science and Information Technology

Approved by

Chairman

Department of Computer Science and Information Technology

Table of Contents

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No** | **Object** | **Page No** | **Signatures** |
| 1. | Introduction to HTML. | 04 |  |
| 2. | Exploring Basic HTML Elements. | 07 |  |
| 3. | Introduction to HTML5. | 18 |  |
| 4. | Exploring HTML5 Elements. | 21 |  |
| 5. | Introduction to Cascading Style Sheets. | 29 |  |
| 6. | Introduction to Javascript. | 41 |  |
| 7. | Introduction to ASP.NET. | 60 |  |
| 8. | Exploring some controls and the concept of field validation in ASP.NET. | 64 |  |
| 9. | Exploring the working of Login controls and some basic Postback controls in ASP.Net. | 67 |  |
| 10. | Working with Databases in ASP.Net | 69 |  |
| 11 | Introduction to PHP | 72 |  |
| 12 | PHP Variables, Control Structures and Loops | 74 |  |
| 13 | MySQL and PHP | 81 |  |
| 14 | Select, Insert, Update and Delete in PHP/MYSQL | 84 |  |

**Lab # 1**

**Object:**

Introduction to HTML.

**Theory:**

Hyper Text Markup Language (HTML) is the main markup language for displaying web pages and other information that can be displayed in a web browser.

HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>), within the web page content. HTML tags most commonly come in pairs like <h1> and </h1>, although some tags, known as empty elements, are unpaired, for example <img>. The first tag in a pair is the start tag, the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, tags, comments and other types of text-based content.

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts in languages such as JavaScript which affect the behavior of HTML webpages.

**HTML Page Structure.**

<html>

<head>

<body>

</body>

</head>

</html>

**HTML Elements.**

HTML documents are defined by HTML elements. An HTML element is everything from the start tag to the end tag. The element content is everything between the start and the end tag. Some HTML elements have empty content Empty elements are closed in the start tag. Most HTML elements can have attributes. HTML Document can also contain nested elements.

The **<html> tag** tells the browser that this is an HTML document. The <html> element is also known as the root element. The <html> tag is the container for all other HTML elements

The **<head> element** is a container for all the head elements. The <head> element must include a title for the document, and can include scripts, styles, meta information, and more.

CT-362: Web Engineering Page 4

The **<body> tag** defines the document's body. The <body> element contains all the contents of an HTML document, such as text, hyperlinks, images, tables, lists, etc.

**HTML Attributes.**

Attributes provide additional information about an element Attributes are always specified in the start tag Attributes come in name/value pairs like: name="value". Attribute values should always be enclosed in quotes. Double style quotes are the most common, but single style quotes are also allowed.

Some attributes that are standard for most HTML elements:

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| class | Classname | Specifies a classname for an element. |
| id | Id | Specifies a unique id for an element. |
| style | style\_definition | Specifies an inline style for an element. |
| title | tooltip\_text | Specifies extra information about an element (displayed as a tool tip). |

**Exercise:**

1. Which of the following is a properly formed HTML document?

C. <html>

<head><title>My document</title></head>

<body>This is my web page</body>

</html>

2. What are the advantages and disadvantages of HTML.

**Advantages:**

* HTML helps to build structure of a website and is a widely used Markup language.
* It is easy to learn.
* Every browser supports HTML Language.
* HTML is light weighted and fast to load.

**Disadvantages:**

* it cannot produce dynamic output alone, since it’s a static language.
* Making the structure of HTML documents becomes tough to understand.

Errors can be costly.

* It is the time consuming as the time it consume to maintain on the color scheme of a page and to make lists, tables and forms.

**Lab # 2**

**Object:**

Exploring Basic HTML Elements.

**Theory:**

**HTML Headings.**

Headings are defined with the <h1> to <h6> tags. <h1> defines the most important heading. <h6> defines the least important heading. Browsers automatically add some empty space (a margin) before and after each heading. Use HTML headings for headings only. Don't use headings to make text BIG or bold. Search engines use your headings to index the structure and content of your web pages.

*<h1>This is a heading</h1>*

*<h2>This is a heading</h2>*

*<h3>This is a heading</h3>*

|  |  |
| --- | --- |
| **Attribute** | **Value** |
| align. | left.  right.  center.  justify. |

**HTML Lines.**

The <hr/> tag creates a horizontal line in an HTML page. The hr element can be used to separate content.

*<p>This is a paragraph</p>*

*<hr />*

*<p>This is a paragraph</p>*

*<hr />*

**HTML Comments.**

Comments can be inserted into the HTML code to make it more readable and understandable. Comments are ignored by the browser and are not displayed. *<!-- This is a comment -->*

**HTML Paragraphs.**

Paragraphs are defined with the <p> tag.

*<p>This is a paragraph</p>*

*<p>This is another paragraph</p>*

|  |  |
| --- | --- |
| **Attribute** | **Value** |
| align. | left.  right.  center.  justify. |

**HTML Line Breaks.**

Use the <br /> tag if you want a line break (a new line) without starting a new paragraph.

CT-362: Web Engineering Page 7

*<p>This is<br />a para<br />graph with line breaks</p>*

**HTML Formatting Tags.**

HTML uses tags like <b> and <i> for formatting output, like bold or italic text. ***HTML Text Formatting Tags.***

|  |  |
| --- | --- |
| **Tag** | **Description** |
| <b> | Defines bold text. |
| <big> | Defines big text. |
| <em> | Defines emphasized text. |
| <i> | Defines italic text. |
| <small> | Defines small text. |
| <strong> | Defines strong text. |
| <sub> | Defines subscripted text. |
| <sup> | Defines superscripted text. |
| <ins> | Defines inserted text. |
| <del> | Defines deleted text. |

***HTML "Computer Output" Tags.***

|  |  |
| --- | --- |
| **Tag** | **Description** |
| <code> | Defines computer code text. |
| <kbd> | Defines text. |
| <samp> | Defines sample computer code. |
| <tt> | Defines teletype text. |
| <var> | Defines a variable. |
| <pre> | Defines preformatted text. |

**HTML Hyperlinks.**

A hyperlink (or link) is a word, group of words, or image that you can click on to jump to a new document or a new section within the current document. When we move the cursor over a link in a Web page, the arrow will turn into a little hand.

Links are specified in HTML using the <a> tag.

The <a> tag can be used in two ways:

• To create a link to another document, by using the href attribute.

• To create a bookmark inside a document, by using the name attribute. The HTML code for a link is simple. It looks like this:

*<a href="url">Link text</a>*

The href attribute specifies the destination of a link.

*<a href="http://www.neduet.edu.pk">Visit NEDUET</a>*

CT-362: Web Engineering Page 8

***HTML Links - The target Attribute.***

The target attribute specifies where to open the linked document. The example below will open the linked document in a new browser window or a new tab:

*<a href="http://www.neduet.edu.pk" target="\_blank">Visit NEDUET!</a>*

***HTML Links - The name Attribute.***

The name attribute specifies the name of an anchor. The name attribute is used to create a bookmark inside an HTML document. Bookmarks are not displayed in any special way. They are invisible to the reader.

A named anchor inside an HTML document:

*<a name="tips">Useful Tips Section</a>*

Create a link to the "Useful Tips Section" inside the same document: *<a href="#tips">Visit the Useful Tips Section</a>*

Or, create a link to the "Useful Tips Section" from another page:

*<a href="http://www.w3schools.com/html\_links.htm#tips">Visit the Useful Tips Section</a>*

***HTML Links –Email Address.***

Anchors can also link to email addresses. When someone clicks on this type of anchored link, their default email program initiates an email message to the linked address.

*<a href="mailto:myemail@neduet.edu.pk">myemail@neduet.edu.pk</a>*

**HTML Images - The <img> Tag and the Src Attribute.**

In HTML, images are defined with the <img> tag. The <img> tag is empty, which means that it contains attributes only, and has no closing tag. To display an image on a page, you need to use the src attribute. Src stands for "source". The value of the src attribute is the URL of the image you want to display. Syntax for defining an image:

*<img src="url" alt="some\_text"/>*

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| alt. | text. | Specifies an alternate text for an image. |
| src. | url. | Specifies the URL of an image. |
| align. | top.  bottom.  middle.  left.  right. | Specifies the alignment of an image according to surrounding element. |
| border. | pixels. | Specifies the width of the border around an image. |
| height, width. | pixels % | Specifies the height and width of an image. |
| hspace. | pixels. | Specifies the whitespaces on left & right side of an image. |
| vspace. | pixels. | Specifies the whitespaces on top & bottom of an image. |
| usemap. | #mapname. | Specifies an image as a client side image map. |

CT-362: Web Engineering Page 9

***HTML Images - The Alt Attribute.***

The required alt attribute specifies an alternate text for an image, if the image cannot be displayed. The value of the alt attribute is an author-defined text:

*<img src="computer.gif" alt="Computer" />*

The alt attribute provides alternative information for an image if a user for some reason cannot view it (because of slow connection, an error in the src attribute, or if the user uses a screen reader).

***HTML Images - Set Height and Width of an Image.***

The height and width attributes are used to specify the height and width of an image. The attribute values are specified in pixels by default:

*<img src="computer.jpg" alt="Computer" width="304" height="228" />*

***HTML Image Map.***

The <map> tag is used to define a client-side image-map. An image-map is an image with clickable areas. The name attribute of the <map> element is required and it is associated with the <img>'s usemap attribute and creates a relationship between the image and the map.

The <map> element contains a number of <area> elements that defines the clickable areas in the image map. The <area> element is always nested inside a <map> tag.

*<img src="planets.gif" width="145" height="126" alt="Planets" usemap="#planetmap" />*

*<map name="planetmap">*

*<area shape="rect" coords="0,0,82,126" href="sun.html" alt="Sun" /> <area shape="circle" coords="90,58,3" href="merc.html" alt="Mercury" /> <area shape="circle" coords="124,58,8" href="venus.html" alt="Venus" /> </map>*

**HTML Tables.**

Tables are defined with the <table> tag. A table is divided into rows (with the <tr> tag), and each row is divided into data cells (with the <td> tag). td stands for "table data," and holds the content of a data cell. A <td> tag can contain text, links, images, lists, forms, other tables, etc.

*<table border="1">*

*<tr>*

*<td>row 1, cell 1</td>*

*<td>row 1, cell 2</td>*

*</tr>*

*<tr>*

*<td>row 2, cell 1</td>*

*<td>row 2, cell 2</td>*

*</tr>*

*</table>*

CT-362: Web Engineering Page 10

***HTML Tables and the Border Attribute.***

If you do not specify a border attribute, the table will be displayed without borders. Sometimes this can be useful, but most of the time, we want the borders to show. To display a table with borders, specify the border attribute:

*<table border="1">*

*<tr>*

*<td>Row 1, cell 1</td>*

*<td>Row 1, cell 2</td>*

*</tr>*

*</table>*

***HTML Table Headers.***

Header information in a table are defined with the <th> tag. All major browsers display the text in the <th> element as bold and centered.

*<table border="1">*

*<tr>*

*<th>Header 1</th>*

*<th>Header 2</th>*

*</tr>*

*<tr>*

*<td>row 1, cell 1</td>*

*<td>row 1, cell 2</td>*

*</tr>*

*<tr>*

*<td>row 2, cell 1</td>*

*<td>row 2, cell 2</td>*

*</tr>*

*</table>*

***HTML Table Tags.***

|  |  |
| --- | --- |
| **Tag** | **Description** |
| <table> | Defines a table. |
| <th> | Defines a table header. |
| <tr> | Defines a table row. |
| <td> | Defines a table cell. |
| <caption> | Defines a table caption. |
| <colgroup> | Defines a group of columns in a table for formatting. |
| <col/> | Defines attribute values for one or more columns. |
| <thead> | Groups the header content in a table. |
| <tbody> | Groups the body content in a table. |
| <tfoot> | Groups the footer content in a table. |

**HTML Unordered Lists.**

An unordered list starts with the <ul> tag. Each list item starts with the <li> tag. The list items are marked with bullets (typically small black circles).

CT-362: Web Engineering Page 11

*<ul>*

*<li>Floppies</li>*

*<li>Harddisks</li>*

*</ul>*

*Output:*

*Floppies*

*Harddisks*

**HTML Ordered Lists.**

An ordered list starts with the <ol> tag. Each list item starts with the <li> tag. The list items are marked with numbers.

*<ol>*

*<li>Floppies</li>*

*<li>Harddisks</li>*

*</ol>*

*Output:*

*1. Floppies.*

*2. Harddisks*

**HTML Definition Lists.**

A definition list is a list of items, with a description of each item. The <dl> tag defines a definition list. The <dl> tag is used in conjunction with <dt> (defines the item in the list) and <dd> (describes the item in the list):

*<dl>*

*<dt>Keyboard</dt>*

*<dd>- an input device</dd>*

*<dt>Printer</dt>*

*<dd>- an output device</dd>*

*</dl>*

*Output:*

*Keyboard*

*- an input device*

*Printer*

*- an output device*

**HTML <div> and <span>.**

HTML elements can be grouped together with <div> and <span>.

***HTML Block Elements.***

Most HTML elements are defined as block level elements or as inline elements. Block level elements normally start (and end) with a new line when displayed in a browser. Examples: <h1>, <p>, <ul>, <table>.

***HTML Inline Elements.***

Inline elements are normally displayed without starting a new line.

Examples: <b>, <td>, <a>, <img>

***The HTML <div> Element.***

The HTML <div> element is a block level element that can be used as a container for grouping other HTML elements. The <div> element has no special meaning. Except that,

CT-362: Web Engineering Page 12

because it is a block level element, the browser will display a line break before and after it. When used together with CSS, the <div> element can be used to set style attributes to large blocks of content. Another common use of the <div> element, is for document layout. It replaces the "old way" of defining layout using tables. Using tables is not the correct use of the <table> element. The purpose of the <table> element is to display tabular data.

***The HTML <span> Element.***

The HTML <span> element is an inline element that can be used as a container for text. The <span> element has no special meaning. When used together with CSS, the <span> element can be used to set style attributes to parts of the text.

***HTML Layouts - Using <div> Elements.***

The div element is a block level element used for grouping HTML elements. The following example uses five div elements to create a multiple column layout:

*<!DOCTYPE html>*

*<html>*

*<body>*

*<div id="container" style="width:500px">*

*<div id="header" style="background-color:#FFA500;">*

*<h1 style="margin-bottom:0;">Main Title of Web Page</h1></div>*

*<div id="menu"*

*style="backgroundcolor:#FFD700;height:200px;width:100px;float:left;"> <b>Menu</b><br />*

*HTML<br />*

*CSS<br />*

*JavaScript</div>*

*<div id="content" style="background*

*color:#EEEEEE;height:200px;width:400px;float:left;">*

*Content goes here</div>*

*<div id="footer" style="background-color:#FFA500;clear:both;text-align:center;"> </div>*

*</div>*

*</body>*

*</html>*

***HTML Layouts - Using Tables.***

A simple way of creating layouts is by using the HTML <table> tag. Multiple columns are created by using <div> or <table> elements. CSS are used to position elements, or to create backgrounds or colorful look for the pages. The following example uses a table with 3 rows and 2 columns - the first and last row spans both columns using the colspan attribute:

CT-362: Web Engineering Page 13

*<!DOCTYPE html>*

*<html>*

*<body>*

*<table width="500" border="0">*

*<tr>*

*<td colspan="2" style="background-color:#FFA500;">*

*<h1>Main Title of Web Page</h1>*

*</td>*

*</tr>*

*<tr valign="top">*

*<td style="background-color:#FFD700;width:100px;text-align:top;"> <b>Menu</b><br />*

*HTML<br />*

*CSS<br />*

*JavaScript*

*</td>*

*<td style="background-color:#EEEEEE;height:200px;width:400px;text-align:top;"> Content goes here</td>*

*</tr>*

*<tr>*

*<td colspan="2" style="background-color:#FFA500;text-align:center;"> </td>*

*</tr>*

*</table>*

*</body>*

*</html>*

**HTML Forms and Input.**

HTML Forms are used to select different kinds of user input. HTML forms are used to pass data to a server. A form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons and more. A form can also contain select lists, textarea, fieldset, legend, and label elements. The <form> tag is used to create an HTML form:

*<form>*

*input elements*

*</form>*

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| action. | URL. | Specifies where to send the form-data when a form is submitted. |
| method. | get/post. | Specifies the HTTP method to use when sending form-data. |
| name. | name. | Specifies the name of a form. |
| target. | \_blank.  \_self.  \_parent.  \_top. | Specifies where to display the response that is received after submitting the form. |

CT-362: Web Engineering Page 14

***HTML Form - Event Attributes.***

HTML Forms tag supports event attributes. The value for all attributes is script. Some of these are given below:

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| onclick. | Script to be run on a mouse click. |
| onmousedown. | Script to be run when mouse button is pressed. |
| onmousemove. | Script to run when mouse pointer moves. |
| onmouseover. | Script to run when mouse pointer moves over an element. |
| onkeypress. | Script to be run when a key is pressed and released. |
| onsubmit. | Script to be run when a form is submitted. |

***HTML Forms - The Input Element.***

The most important form element is the input element. The input element is used to select user information. An input element can vary in many ways, depending on the type attribute. An input element can be of type text field, checkbox, password, radio button, submit button, and more. The most used input types are described below:

***Text Fields.***

<input type="text" /> defines a one-line input field that a user can enter text into:

*<form>*

*First name: <input type="text" name="firstname" /><br />*

*Last name: <input type="text" name="lastname" />*

*</form>*

***Password Field.***

<input type="password" /> defines a password field:

*<form>*

*Password: <input type="password" name="pwd" />*

*</form>*

***Radio Buttons.***

<input type="radio" /> defines a radio button. Radio buttons let a user select ONLY ONE of a limited number of choices:

*<form>*

*<input type="radio" name="sex" value="male" /> Male<br />*

*<input type="radio" name="sex" value="female" /> Female*

*</form>*

***Checkboxes.***

<input type="checkbox" /> defines a checkbox. Checkboxes let a user select ONE or MORE options of a limited number of choices.

CT-362: Web Engineering Page 15

*<form>*

*<input type="checkbox" name="vehicle" value="Bike" /> I have a*

*bike<br />*

*<input type="checkbox" name="vehicle" value="Car" /> I have a*

*car*

*</form>*

***Submit Button.***

<input type="submit" /> defines a submit button. A submit button is used to send form data to a server. The data is sent to the page specified in the form's action attribute. The file defined in the action attribute usually does something with the received input:

*<form name="input" action="html\_form\_action.asp" method="get"> Username: <input type="text" name="user" />*

*<input type="submit" value="Submit" />*

*</form>*

**Exercise:**

1. Write the HTML tags for :

a. A paragraph that is a description of a book, include the title of the book as well as its author. Names and titles should be underlined, italicized and bolded.

<p> Cheque book <strong> Celestial Bodies **centres around a family in the Omani village of al-Awafi**.</strong> <u> Vasdev Mohi </u> text.</p>

b. Print two lists with any information you want. One list should be an ordered list and the other should be unordered list.

<ol>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ol>

<ul>  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Milk</li>  
</ul>

c. Create a page with a link at the top of it when clicked will jump all the way to the bottom of the page and at the bottom of the page there should be a link to jump back to the top of the page.

Ans <p id="up">Top</p>

<a href="#bottom">Go to Bottom</a>

………….

<p id="bottom">Bottom</p>

<a href="#up">Go To Top</a>

**d. Create some links to various search engines.**

<a href ="https://www.google.com">Google</a><br>

<a href ="https://www.facebook.com">Facebook</a><br>

**e. Display an image that when clicked will link to a search engine of your choice and the page should open in a new window**.

<h1>Image As a Hyperlink</h1> <a href="https://www.google.com" target="\_blank"><img src="maclaren.jpg" alt="Image" width="400px" height="400px"></a>



2. Design a form to take input from the user to generate his / her CV.

<form class="cv">

<h1>CV</h1>

<label>Name: <input type="text" name="name" id="name" required ></label><br>

<label>Contact: <input type="text" name="contact" id="contact" required></label><br>

<textarea name="edu" rows="8" cols="80" id="edu" required>Education:</textarea><br>

<textarea name="addr" id="addr" rows="4" cols="40" required>Address:</textarea><br>

<label>Critical Thinking <input type="checkbox" name="skills" id="skills"></label>

<label>Teamwork <input type="checkbox" name="skills" id="skills" ></label>

<label>Leadership <input type="checkbox" name="skills" id="skills" ></label>

<label>Motivational <input type="checkbox" name="skills" id="skills" ></label>

<label>Flxible<input type="checkbox" name="skills" id="skills" ></label><br>

<label for="work">Work Experience</label>

<select class="work" >

<option>1</option><option>2</option><option>5</option><option>Or More</option>

</select><br>

<h4>Professional Skills:</h4>

<label>Tax Computations <input type="radio" name="prof" id="prof"></label>

<label>Financial Reporting<input type="radio" name="prof" id="prof"></label>

<label>Cost Analysis <input type="radio" name="prof" id="prof"></label>

<label>Payroll Forensic Accounting

<input type="radio" name="prof" id="prof"></label>

</form>

**Lab # 3**

**Object:**

Introduction to HTML5

**Theory:**

HTML5 is the new standard for HTML. The previous version of HTML, HTML 4.01, came in 1999. The web has changed a lot since then. HTML5 is still a work in progress. However, the major browsers support many of the new HTML5 elements and APIs.

HTML5 is cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG). WHATWG was working with web forms and applications, and W3C was working with XHTML 2.0. In 2006, they decided to cooperate and create a new version of HTML.

Some rules for HTML5 were established:

• New features should be based on HTML, CSS, DOM, and JavaScript • Reduce the need for external plugins (like Flash)

• Better error handling

• More markup to replace scripting

• HTML5 should be device independent

• The development process should be visible to the public

**The HTML5 <!DOCTYPE>.**

In HTML5 there is only one <!doctype> declaration, and it is very simple: *<!DOCTYPE html>*

**Minimum HTML5 Document.**

Below is a simple HTML5 document, with the minimum of required tags: *<!DOCTYPE html>*

*<html>*

*<head>*

*<title>Title of the document</title>*

*</head>*

*<body>*

*The content of the document......*

*</body>*

*</html>*

**HTML5 - New Features.**

Some of the most interesting new features in HTML5:

• The <canvas> element for 2D drawing

• The <video> and <audio> elements for media playback

• Support for local storage

• New content-specific elements, like <article>, <footer>, <header>, <nav>, <section> • New form controls, like calendar, date, time, email, url, search

CT-362: Web Engineering Page 18

**Browser Support for HTML5.**

HTML5 is not yet an official standard, and no browsers have full HTML5 support. But all major browsers (Safari, Chrome, Firefox, Opera, Internet Explorer) continue to add new HTML5 features to their latest versions.

**New Elements in HTML5.**

The internet has changed a lot since HTML 4.01 became a standard in 1999. Today, some elements in HTML 4.01 are obsolete, never used, or not used the way they were intended to. These elements are removed or re-written in HTML5. To better handle today's internet use, HTML5 includes new elements for better structure, better form handling, drawing, and for media content.

***New Semantic/Structural Elements.***

|  |  |
| --- | --- |
| **Tag** | **Description** |
| <article> | Defines an article. |
| <aside> | Defines content aside from the page content |
| <bdi> | Isolates a part of text that might be formatted in a different direction from other text outside it. |
| <command> | Defines a command button that a user can invoke |
| <details> | Defines additional details that the user can view or hide |
| <summary> | Defines a visible heading for a <details> element |
| <figure> | Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc |
| <figcaption> | Defines a caption for a <figure> element |
| <footer> | Defines a footer for a document or section |
| <header> | Defines a header for a document or section |
| <hgroup> | Groups a set of <h1> to <h6> elements when a heading has multiple levels. |
| <mark> | Defines marked/highlighted text |
| <meter> | Defines a scalar measurement within a known range (a gauge) |
| <nav> | Defines navigation links |
| <progress> | Represents the progress of a task |
| <ruby> | Defines a ruby annotation (for East Asian typography) |
| <rt> | Defines an explanation/pronunciation of characters (for East Asian typography) |
| <rp> | Defines what to show in browsers that do not support ruby annotations |
| <section> | Defines a section in a document |
| <time> | Defines a date/time |
| <wbr> | Defines a possible line-break |

**Removed Elements.**

The following HTML 4.01 elements are removed from HTML5:

<acronym>, <applet>, <basefont>, <big>, <center>, <dir>, <font>, <frame>, <frameset>, <noframes>, <strike>, <tt>

CT-362: Web Engineering Page 19

**Exercise:**

1. Write down the advantages and disadvantages of HTML5

## Advantages

* Multimedia support
* Short and simple syntax
* Improved security features
* Inlcude semantic tags
* Cross-platform support

## Disadvantages

* Client-side rendering
* Local storage is less secure
* Different video supports for different browsers
* Media licensing cost
* Doesn’t support old browsers
* Tough to handle responsiveness on the range of devices

**Lab # 4**

**Object:**

Exploring HTML5 Elements.

**Theory:**

**HTML5 Video.**

There has not been a standard for showing a video/movie on a web page until now. Today, most videos are shown through a plug-in (like flash). However, different browsers may have different plug-ins.

HTML5 defines a new element which specifies a standard way to embed a video/movie on a web page: the <video> element. The control attribute adds video controls, like play, pause, and volume. It is also a good idea to always include width and height attributes. If height and width are set, the space required for the video is reserved when the page is loaded. However, without these attributes, the browser does not know the size of the video, and cannot reserve the appropriate space to it. The effect will be that the page layout will change during loading (while the video loads).

Text should be inserted between the <video> and </video> tags for browsers that do not support the <video> element.

The <video> element allows multiple <source> elements. <source> elements can link to different video files. The browser will use the first recognized format.

*<video width="320" height="240" controls="controls">*

*<source src="movie.mp4" type="video/mp4" />*

*<source src="movie.ogg" type="video/ogg" />*

*Your browser does not support the video tag.*

*</video>*

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| autoplay. | autoplay. | Specifies that the video will start playing as soon as it is ready. |
| controls. | controls. | Specifies that video controls should be displayed (such as a play/pause button etc). |
| height. | pixels. | Sets the height of the video player. |
| loop. | loop. | Specifies that the video will start over again, every time it is finished |
| muted. | muted. | Specifies that the audio output of the video should be muted. |
| poster. | URL. | Specifies an image to be shown while the video is downloading, or until the user hits the play button. |
| preload. | auto, metadata, none. | Specifies if and how the author thinks the video should be loaded when the page loads. |
| src. | URL. | Specifies the URL of the video file. |
| width. | pixels. | Specifies the width of the video player. |

CT-362: Web Engineering Page 21

**HTML5 Audio.**

There has not been a standard for playing audio files on a web page until now. Today, most audio files are played through a plug-in (like flash). However, different browsers may have different plug-ins.

HTML5 defines a new element which specifies a standard way to embed an audio file on a web page: the <audio> element. Text should be inserted between the <audio> and </audio> tags for browsers that do not support the <audio> element.

The <audio> element allows multiple <source> elements. <source> elements can link to different audio files. The browser will use the first recognized format.

*<audio controls="controls">*

*<source src="song.ogg" type="audio/ogg" />*

*<source src="song.mp3" type="audio/mpeg" />*

*Your browser does not support the audio element.*

*</audio>*

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| autoplay. | autoplay. | Specifies that the audio will start playing as soon as it is ready. |
| controls. | controls. | Specifies that audio controls should be displayed (such as a play/pause button etc). |
| loop. | loop. | Specifies that the audio will start over again, every time it is finished |
| preload. | auto, metadata, none. | Specifies if and how the author thinks the audio should be loaded when the page loads. |
| src. | URL. | Specifies the URL of the audio file. |

**HTML5 Drag and Drop.**

Drag and drop is a part of the HTML5 standard. Drag and drop is a very common feature. It is when you "grab" an object and drag it to a different location. In HTML5, drag and drop is part of the standard, and any element can be draggable.

*<!DOCTYPE HTML>*

*<html>*

*<head>*

*<script type="text/javascript">*

*function allowDrop(ev)*

*{*

*ev.preventDefault();*

*}*

*function drag(ev)*

*{*

*ev.dataTransfer.setData("Text",ev.target.id);*

*}*

CT-362: Web Engineering Page 22

*function drop(ev)*

*{*

*ev.preventDefault();*

*var data=ev.dataTransfer.getData("Text");*

*ev.target.appendChild(document.getElementById(data));*

*}*

*</script>*

*</head>*

*<body>*

*<div id="div1" ondrop="drop(event)"*

*ondragover="allowDrop(event)"></div>*

*<img id="drag1" src="img\_logo.gif" draggable="true"*

*ondragstart="drag(event)" width="336" height="69" />*

*</body>*

*</html>*

***Make an Element Draggable.***

First of all: To make an element draggable, set the draggable attribute to true: *<img draggable="true" />*

***What to Drag - ondragstart and setData().***

Then, specify what should happen when the element is dragged. In the example above, the ondragstart attribute calls a function, drag(event), that specifies what data to be dragged. The dataTransfer.setData() method sets the data type and the value of the dragged data:

*function drag(ev)*

*{*

*ev.dataTransfer.setData("Text",ev.target.id);*

*}*

*In this case, the data type is "Text" and the value is the id of the draggable element ("drag1").*

***Where to Drop – ondragover.***

The ondragover event specifies where the dragged data can be dropped. By default, data/elements cannot be dropped in other elements. To allow a drop, we must prevent the default handling of the element. This is done by calling the event.preventDefault() method for the ondragover event:

*event.preventDefault()*

***Do the Drop – ondrop.***

When the dragged data is dropped, a drop event occurs. In the example above, the ondrop attribute calls a function, drop(event):

CT-362: Web Engineering Page 23

*function drop(ev)*

*{*

*ev.preventDefault();*

*var data=ev.dataTransfer.getData("Text");*

*ev.target.appendChild(document.getElementById(data));*

*}*

***Code explained:***

• Call preventDefault() to prevent the browser default handling of the data (default is open as link on drop).

• Get the dragged data with the dataTransfer.getData("Text") method. This method will return any data that was set to the same type in the setData() method. • The dragged data is the id of the dragged element ("drag1").

• Append the dragged element into the drop element.

**HTML5 Canvas.**

The HTML5 <canvas> element is used to draw graphics, on the fly, via scripting (usually JavaScript). The <canvas> element is only a container for graphics; you must use a script to actually draw the graphics.

A canvas is a draw able region defined in HTML code with height and width attributes. Canvas has several methods for drawing paths, boxes, circles, characters, and adding images.

***Create a Canvas.***

A canvas is specified with the <canvas> element. Specify the id, width, and height of the <canvas> element:

*<canvas id="myCanvas" width="200" height="100"></canvas>*

***Draw With JavaScript.***

The <canvas> element has no drawing abilities of its own. All drawing must be done inside a JavaScript:

*<script type="text/javascript">*

*var c=document.getElementById("myCanvas");*

*var ctx=c.getContext("2d");*

*ctx.fillStyle="#FF0000";*

*ctx.fillRect(0,0,150,75);*

*</script>*

JavaScript uses the id to find the <canvas> element:

*var c=document.getElementById("myCanvas");*

Then, create a context object:

*var ctx=c.getContext("2d");*

The getContext("2d") object is a built-in HTML5 object, with many methods to draw paths, boxes, circles, characters, images and more. The next two lines draws a red rectangle:

*ctx.fillStyle="#FF0000";*

*ctx.fillRect(0,0,150,75);*

CT-362: Web Engineering Page 24

The fillStyle attribute makes it red, and the fillRect attribute specifies the shape, position, and size.

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| height. | pixels. | Specifies the height of the canvas. |
| width. | pixels. | Specifies the width of the canvas. |

**HTML5 New Input Type.**

HTML5 has several new input types for forms. These new features allow better input control and validation.

• Color.

• Date.

• Datetime.

• datetime-local. • Email.

• Month.

• Number.

***Input Type: color.***

• range. • search. • tel.

• time. • url.

• week.

The color type is used for input fields that should contain a color.

*Select your favorite color: <input type="color" name="favcolor" />*

***Input Type: date.***

The date type allows the user to select a date.

*Birthday: <input type="date" name="bday" />*

***Input Type: datetime.***

The datetime type allows the user to select a date and time (with time zone). *Birthday (date and time): <input type="datetime" name="bdaytime" />*

***Input Type: datetime-local.***

The datetime-local type allows the user to select a date and time (no time zone). *Birthday (date and time): <input type="datetime-local" name="dtime" />*

***Input Type: month.***

The month type allows the user to select a month and year.

*Birthday (month and year): <input type="month" name="bdaymonth" />*

***Input Type: number.***

The number type is used for input fields that should contain a numeric value. You can also set restrictions on what numbers are accepted.

*Quantity (between 1 & 5):<input type="number" name="quantity"*

*min="1" max="5" />*

***Input Type: range.***

The range type is used for input fields that should contain a value from a range of numbers. You can also set restrictions on what numbers are accepted. *<input type="range" name="points" min="1" max="10" />*

CT-362: Web Engineering Page 25

Use the following attributes to specify restrictions on number & range: max - specifies the maximum value allowed.

min - specifies the minimum value allowed.

step - specifies the legal number intervals.

value - Specifies the default value.

***Input Type: search.***

The search type is used for search fields (a search field behaves like a regular text field). *Search Google: <input type="search" name="googlesearch" />*

***Input Type: tel.***

*Telephone: <input type="tel" name="usrtel" />*

***Input Type: time.***

The time type allows the user to select a time.

*Select a time: <input type="time" name="usr\_time" />*

***Input Type: url.***

The url type is used for input fields that should contain a URL address. The value of the url field is automatically validated when the form is submitted.

*Add your homepage: <input type="url" name="homepage" />*

***Input Type: week.***

The week type allows the user to select a week and year.

Select a week: <input type="week" name="week\_year" />

**HTML5 Form Element.**

HTML5 has the following new form elements:

<datalist>.

<keygen>.

<output>.

***<datalist> Element.***

The <datalist> element specifies a list of pre-defined options for an <input> element. The <datalist> element is used to provide an "autocomplete" feature on <input> elements. Users will see a drop-down list of pre-defined options as they input data. Use the <input> element's list attribute to bind it together with a <datalist> element.

*<input list="browsers" />*

*<datalist id="browsers">*

*<option value="Internet Explorer">*

*<option value="Firefox">*

*<option value="Chrome">*

*<option value="Opera">*

*<option value="Safari">*

*</datalist>*

***<keygen> Element.***

The purpose of the <keygen> element is to provide a secure way to authenticate users. The <keygen> tag specifies a key-pair generator field in a form. When the form is

CT-362: Web Engineering Page 26

submitted, two keys are generated, one private and one public. The private key is stored locally, and the public key is sent to the server. The public key could be used to generate a client certificate to authenticate the user in the future.

*<form action="demo\_keygen.asp" method="get">*

*Username: <input type="text" name="usr\_name" />*

*Encryption: <keygen name="security" />*

*<input type="submit" />*

*</form>*

***<output> Element***

The <output> element represents the result of a calculation (like one performed by a script).

*<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">0*

*<input type="range" name="a" value="50" />100*

*+<input type="number" name="b" value="50" />*

*=<output name="x" for="a b"></output>*

*</form>*

**HTML5 Form Attribute.**

HTML5 has several new attributes for <form> and <input>.

New attributes for <form>:

• autocomplete.

• novalidate.

New attributes for <input>:

• autocomplete.

• Autofocus.

• Form.

• Formaction.

• Formenctype.

• Formmethod.

• Formnovalidate.

• Formtarget.

• height and width.

• List.

• min and max.

• multiple.

• pattern (regexp).

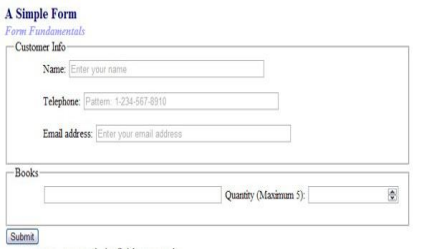
• Placeholder.

• Required.

• Step.

**Exercise:**

1. Create the form shown in the figure below.



When creating the code for your form, you must use the HTML5 tags that are appropriate to replicate the form and fulfill all the specifications listed.

a) Code the form with autocomplete active.

b) The Name field you create should have autofocus, placeholder text, and be required. Don't forget to select the appropriate type for this field as well as all the fields that follow.

c) The Telephone field should have placeholder text, a pattern to restrict entry, and be required.

d) The Email address field should have placeholder text and allow multiple entries. This field should also be required.

e) The Books field should have a data list. You can select the content you would like to list.

f) The Quantity (Maximum 5) field should have a minimum value of 1 and a maximum value of 5.

    <form action="">

        <h1>simple form</h1>

        <h1>contact info</h1>

        <div>

            <label for="name">Name</label>

            <input type="text" name="name" id="name">

        </div>

        <div>

            <label for="telephone">Telephone</label>

            <input type="number" name="telephone" id="telephone">

        </div>

        <div>

            <label for="email">Email</label>

            <input type="email" name="email" id="email">

        </div>

        <h1>books</h1>

        <select name="" id="">

<option value="0" selected>Select a book</option>

<option value="1">1</option>

<option value="2">2</option>

        </select>

        <div>

            <label for="Quantity">Quantity(Maximum:5)</label><input type="number" min="1" max="5" name="Quantity" id="Quantity"></div>

        </div>

        <div>

            <input type="submit" value="submit">

        </div>

    </form