UNIT 2

HTML (Hyper Text Markup Language)

Introduction to HTML

HTML stands for Hypertext Markup Language and it is a widely used programming language used to develop web pages.

HTML Stands for HyperText Markup Language, where;

HyperText stands for Link between web pages.

Markup Language means Text between tags that define the structure.

HTML is a markup language that is used to create web pages. It defines how the web page looks and how to display content with the help of elements. It forms or defines the structure of our Web Page, thus it forms or defines the structure of our Web Page.

HTML Documents

All HTML documents must start with a document type declaration: <!DOCTYPE html>. The HTML document itself begins with <html> and ends with </html>. The visible part of the HTML document is between <body> and </body>.

Structure of an HTML Document:

```
<!DOCTYPE html>
<html>
<head>
    <title>this is title page</title>
</head>
<body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
</body>
</html>
```

An HTML Document is mainly divided into two parts:

HEAD: This contains the information about the HTML document. For Example, the Title of the page, version of HTML, Meta Data, etc.

BODY: This contains everything you want to display on the Web Page.

HTML Elements and HTML Attributes:

HTML Elements

An HTML element is what we created from a tag start to its end.

<tag> This is an HTML element </tag>

Example: This is a simple paragraph

However, not all the HTML elements have closing words. They go by the names void, singleton, or empty elements.

Examples include: This text contains
a line break.

*Types of HTML Elements: Two broad categories of HTML elements. They are the **inline-level** as well as the **block-level** HTML elements.*

*The **block-level** elements make up the structure of the document. It takes up the entire width of the page.*

Examples include , , <h1>, <h6>, <form>, <ol>, <ul> and <li>, e.t.c.

*The **inline-level** elements are mainly in the contents of a block.* They include <img>, <a>, , , <b>, , <i>, <code>, <input> and <button> among others.

HTML Attribute:

HTML attributes provide additional information about HTML elements. It is usually in the form of unique words that one inserts inside the opening tag. They control the behavior of the element that follows. Attributes usually come in name/value pairs like: name="value"

Structure of an attribute is as follows:

<element attribute="value">element content</element>

Examples:

Visit W3Schools

HTML Headings:

The HTML heading tags (<h1> to <h6>) are used to add headings to a webpage. The <h1> to <h6> HTML elements represent six levels of section headings. <h1> is the highest section level and **<h6> is the lowest.**

Example:

<h1>This is heading 1.</h1>

```
<h2>This is heading 2.</h2>
<h3>This is heading 3.</h3>
<h4>This is heading 4.</h4>
<h5>This is heading 5.</h5>
<h6>This is heading 6.</h6>
```

HTML Paragraphs

The HTML <p> element defines a paragraph. A paragraph always starts on a new line, and browsers automatically add some white space (a margin) before and after a paragraph.

Example:

```
<p>This is a paragraph.</p>
```

HTML Tag

The HTML element is the generic container for flow content. The HTML tag is used to group the large section of HTML elements together.

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
.myDiv {
  border: 5px outset red;
  background-color: lightblue;
  text-align: center;
}
</style>
</head>
<body>
<h1>The div element</h1>
<div class="myDiv">
  <h2>This is a heading in a div element</h2>
```

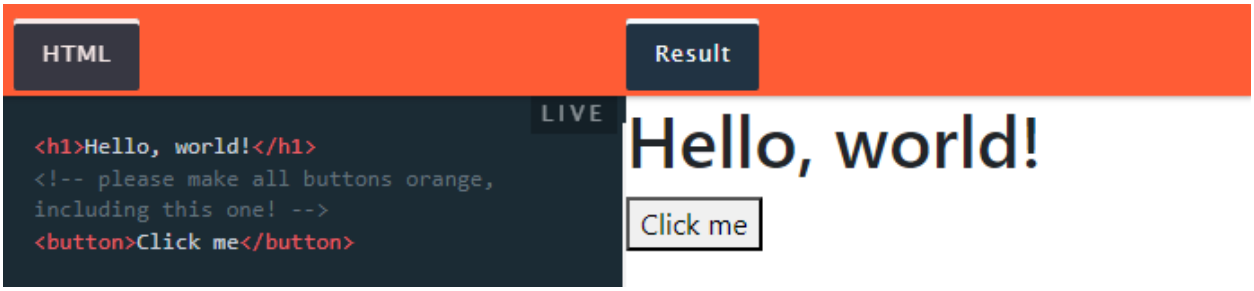
```
  <p>This is some text in a div element.</p>
</div>
<p>This is some text outside the div element.</p>
</body>
</html>
```

HTML Comments:

What is a comment in HTML?

In HTML, a comment is a section of text that is not processed by the web browser. Comments are enclosed in <!-- --> tags. These tags tell the browser that the text inside them is a comment and should not be rendered on the front end.

For Example:



Note:

You can also quickly comment a line of code with the keyboard shortcut Ctrl + / on PC or Command + /. This method is much faster than typing the tags manually.

HTML Text Formatting

What is HTML Formatting?

HTML Formatting is a process that allows us to format text to increase its visual appeal. Various HTML tags can change how text appears on a web page and make the text attractive. We can use HTML text formatting tags to bold, italicize, underlined text, and do more.

HTML and Tags

In HTML, tag and tag display the text in bold front.

HTML tag

The HTML tag specifies bold text, without any logical importance. It opens with and ends with tag.

HTML Tag

The HTML tag is a logical tag that specifies text with importance. The content inside the element appears in the bold text but with semantic importance. It opens with and ends with tag.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Strong Text Example</title>
  </head>
  <body>
    <p>This is normal text.</p>
        <p><b>This is bold text.</b></p>
    <p><strong>This is strong text with logical importance.</strong></p>
  </body>
</html>
```

HTML Tag

The HTML tag displays the text in italics. It begins with tag and ends with tag.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Italics Text Example</title>
  </head>
  <body>
    <p>This is normal text.</p>
    <p><i>This is italics text.</i></p>
  </body>
</html>
```

HTML Tag

The HTML tag emphasizes the text with semantic importance. It begins with tag and ends with tag.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Emphasized Text Example</title>
  </head>
  <body>
    <p>This is normal text.</p>
    <p><em>This is emphasized text.</em></p>
  </body>
</html>
```

HTML <mark> Tag

The <mark> tag in HTML is used to highlight a text. It begins with a <mark> tag and a closes with a </mark> tag.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Highlighted Text Example</title>
  </head>
  <body>
    <p>This is normal text.</p>
    <p><mark>This is highlighted text.</mark></p>
  </body>
</html>
```

HTML <sup> Tag

The <sup> tag helps us to superscript a text. It has an opening ^{tag and a closing} tag. Superscript text displays with a raised baseline using a smaller font.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Superscript Text Example</title>
  </head>
  <body>
    <p>This is normal text.</p>
    <p>This is<sup>superscript</sup>text.</p>
  </body>
</html>
```

HTML <sub> Text

The <sub> element helps us to subscript a text on a web page. It has a starting tag _{and a closing tag}. The subscript text appears with a lowered baseline using a smaller font.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Subscript Text Example</title>
  </head>
  <body>
    <p>This is normal text.</p>
    <p>This is<sub> subscript</sub>text.</p>
  </body>
</html>
```

HTML <small> Tag

The HTML <small> tag specifies smaller text. It is mostly used for copyright and legal text on a web page. With this tag, the size of the text will reduce one font size more than the previous font size.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Smaller Text Example</title>
  </head>
  <body>
      <p>Small Font:<small> This is an example to show Smaller Text.</small></p>
  </body>
</html>
```

HTML <big> Tag

The HTML <big> tag helps us make the text's size larger than the surrounding text. Using this tag makes the font size larger than the previous one.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Big Text Example</title>
  </head>
  <body>
      <p>Big Font:<big> This is an example to show Larger Text.</big></p>
  </body>
</html>
```

HTML <del> Tag

The HTML <del> tag specifies that text that has been deleted from a document. The <del> element starts with a <del> tag and ends with a </del> tag. The text between these tags shows as deleted text with a line striking through it.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Delete Text Example</title>
  </head>
  <body>
    <p>Check out this example,<del>this text will be deleted.</del></p>
  </body>
</html>
```

HTML <ins> Element

The HTML <ins> tag specifies that text has been inserted into a document. It has a starting tag <ins> tag and a closing tag </ins>. The text between these tags appears as inserted text with an underline beneath it.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Insert Text Example</title>
  </head>
  <body>
     <p> Check out this example,<ins> to insert a new paragraph.</ins></p>
  </body>
</html>
```

HTML Non-Breaking Space ( )

The simplest way to add a space in HTML (besides hitting the spacebar) is with the non-breaking space entity, written as *** *** *or* *** ***. Multiple adjacent non-breaking spaces won't be

collapsed by the browser, letting you “force” several visible spaces between words or other page elements.

Example:

```
<p>There is one space between these words.</p>

<p>There are two spaces between these  words.</p>

<p>There are five spaces between these     words.</p>
```

HTML <Pre> tag

The <pre> tag defines preformatted text. Text in a <pre> element is displayed in a fixed-width font, and the text preserves both spaces and line breaks. The text will be displayed exactly as written in the HTML source code.

Example:

```
<pre>
In HTML, the pre tag
keeps all     spaces and
line breaks
intact.
                __oooooooo__
            oOOOOOOOOOOOOOOOOOOOo
</pre>
```

**HTML Break (
) Tag**

If you want to insert a line break, use the HTML break tag, written as
. You don’t need a closing tag here — just writing
 adds a line break.

Example:

```
<p>HubSpot, Inc.<br>25 First St.<br>Cambridge, MA 02141</p>
```

HTML Tag

A element which is used to color a part of a text.The tag is an inline container used to mark up a part of a text, or a part of a document.

Example:

```
<!DOCTYPE html>
```

```
<html>

<body>

<h1>The span element</h1>

<p>My mother has <span style="color:blue;font-weight:bold">blue</span> eyes and my father has <span style="color:darkolivegreen;font-weight:bold">dark green</span> eyes.</p>

</body>

</html>
```

HTML <tt> tag

HTML <tt> tag was used to define text in monospaced font or fixed-width fonts so that it would render as teletype, text-only screen, or line printer on the browser.

```
!DOCTYPE html>

<html>

<head>

<title>HTML tt tag</title>

</head>

<body>

<h2>Example of tt tag</h2>

<p>This is paragraph with default font</p>

<p><tt>This is teletype paragraph</tt></p>

</body>

</html>
```

HTML Images

What is HTML Image?

The HTML <img> tag is used to embed an image in a web page. Images are not technically inserted into a web page; images are linked to web pages. The <img> tag creates a holding space for the referenced image.

The <img> tag is empty, it contains attributes only, and does not have a closing tag.The <img> tag has two required attributes:

src - Specifies the path to the image

alt - Specifies an alternate text for the image

Syntax:

<img src="url" alt="alternatetext">

Example

<img src="img_chania.jpg" alt="Flowers in Chania">

Image Size - Width and Height

You can use the style attribute to specify the width and height of an image.

Example

<img src="img_girl.jpg" alt="Girl in a jacket" style="width:500px;height:600px;">

Image as a Link

To use an image as a link, put the <img> tag inside the <a> tag:

<img src="image destination">

Example:

```
<!DOCTYPE html>
<html>
<body>
<h2>Image as a Link</h2>
<p>The image is a link. You can click on it.</p>
<a href="https://www.w3schools.com/html/html_images.asp">
<img src="smiley.gif" alt="HTML tutorial" style="width:42px;height:42px;">
</a>
</body>
</html>
```

The Anchor element

The <a> HTML element (or anchor element), with its *href attribute*, creates a hyperlink to web pages, files, email addresses, locations in the same page, or anything else a URL can address.

Content within each <a> should indicate the link's destination. If the href attribute is present, pressing the enter key while focused on the <a> element will activate it.

Example

```
<a href="https://example.com">Website</a>
```

HTML List

What is an HTML List?

List in HTML helps to display a list of information semantically. There are three types of lists in HTML:

1. Unordered list or Bulleted list (ul)
2. Ordered list or Numbered list (ol)
3. Description list or Definition list (dl)

1. HTML Unordered List

In HTML unordered list, the list items have no specific order or sequence. An unordered list is also called a Bulleted list, as the items are marked with bullets. It begins with the <ul> tag and and closes with a tag. The list items begin with the <li> tag and end with tag.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Unordered List</title>
  </head>
  <body>
  <h2>List of Fruits</h2>
    <ul>
      <li>Apple</li>
      <li>Mango</li>
      <li>Banana</li>
      <li>Grapes</li>
      <li>Orange</li>
    </ul>
```

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

2. Ordered List

In HTML, all the list items in an ordered list are marked with numbers by default instead of bullets. An HTML ordered list starts with the <ol> tag and ends with the tag. The list items start with the <li> tag and end with tag.

Example

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Ordered List</title>
  </head>
  <body>
  <h2>List of Fruits</h2>
    <ol>
      <li>Apple</li>
      <li>Mango</li>
      <li>Banana</li>
      <li>Grapes</li>
      <li>Orange</li>
    </ol>
  </body>
</html>
```

Different Types of Ordered Lists in HTML

Instead of numbers, you can mark your list items with the alphabet: A, B, C or a,b,c, or roman numerals: i, ii, iii, etc. You can do this by using the <ol> tag type attribute. To mark the list items with letters A, B, C, etc., you must specify A as the type attribute's value in the <ol> tag.

Example

```
<!DOCTYPE html>
```

```
<html>
  <head>
    <title>HTML Ordered List</title>
  </head>
  <body>
  <h2>List of Fruits</h2>
   <ol type="A">
      <li>Apple</li>
      <li>Mango</li>
     <li>Banana</li>
   </ol>
</body>
</html>
```

By default, an ordered list will start counting from 1. If you want to start counting from a specified number, you can use the start attribute:

Example

```
<ol start="50">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

3. **HTML Description List or Definition List**
   In an HTML Description list or Definition List, the list items are listed like a dictionary or encyclopedia. Each item in the description list has a description. You can use a description list to display items like a glossary. You will need the following HTML tags:
   <dl> (Definition list) tag – Start tag of the definition list
   <dt> (Definition Term) tag – It specifies a term (name)
   <dd> tag (Definition Description) – Specifies the term definition
   </dl> tag (Definition list) – Closing tag of the definition list

   Example:

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

```
<head>
  <title>HTML Description List</title>
</head>
 <body>
  <dl>
    <dt><b>Apple</b></dt>
    <dd>A red colored fruit</dd>
    <dt><b>Honda</b></dt>
    <dd>A brand of a car</dd>
    <dt><b>Spinach</b></dt>
    <dd>A green leafy vegetable</dd>
  </dl>
 </body>
</html>
```

4. **HTML Nested Lists:**

An HTML Nested list refers to a list within another list. We can create a nested ordered list, a nested unordered list, or a nested ordered list inside an unordered list.

Example:

```
<!DOCTYPE html>

<html>

<head>

   <title>HTML Nested Ordered List</title>

</head>

<body>

   <ol>

      <li>Banana</li>

      <li> Apple

         <ol>

            <li>Green Apple</li>

            <li>Red Apple</li>

         </ol>

      </li>

      <li>Pineapple</li>
```

```
    <li>Orange</li>
  </ol>
</body>
</html>
```

HTML Tables:

A table is a representation of data arranged in rows and columns. Really, it's more like a spreadsheet. In HTML, with the help of tables, you can arrange data like images, text, links and so on into rows and columns of cells.

Common HTML Table tags:

- <tr> - represents rows
- <td> - used to create data cells
- <th> - used to add table headings
- <caption> - used to insert captions
- <thead> - adds a separate header to the table
- <tbody> - shows the main body of the table
- <tfoot> - creates a separate footer for the table

HTML Table Syntax:

```
<table>
  <tr>
    <td>Cell 1</td>
    <td>Cell 2</td>
    <td>Cell 3</td>
  </tr>
  <tr>
    <td>Cell 4</td>
    <td>Cell 5</td>
    <td>Cell 6</td>
  </tr>
</table>
```

How to Add a Table Heading in HTML?

The <th> is used to add headings to tables.

Example:

```
<table>
  <tr>
    <th>First Name</th>
    <th>Last Name</th>
    <th>Email Address</th>
  </tr>
  <tr>
   <td>Hillary</td>
   <td>Nyakundi</td>
   <td>tables@mail.com</td>
  </tr>
  <tr>
    <td>Lary</td>
    <td>Mak</td>
    <td>developer@mail.com</td>
  </tr>
</table>
```

How to Add a Caption to a Table?

The main use of adding a caption to table is to provide a description about the data represented in the table. The caption can either be placed at the top of the table or the bottom and by default it will always be centered.

To insert a caption into a table, use the <caption> tag.

Example :

```
<table>
  <caption>Free Coding Resources</caption>
  <tr>
```

```
  <th>Sites</th>
  <th>Youtube Channels</th>
  <th>Mobile Appss</th>
 </tr>
 <tr>
  <td>Freecode Camp</td>
  <td>Freecode Camp</td>
  <td>Enki</td>
 </tr>
 <tr>
  <td>W3Schools</td>
  <td>Academind</td>
  <td>Programming Hero</td>
 </tr>
 <tr>
  <td>Khan Academy</td>
  <td>The Coding Train</td>
  <td>Solo learn</td>
 </tr>
</table>
```

How to Use Cell Spanning in an HTML Table?

colspan for horizontal spanning and rowspan for vertical spanning. The two attributes are assigned numbers greater that zero which shows the number of cells you wish to span.

Example:

```
<table>
 <tr>
  <th>Name</th>
  <th>Subject</th>
```

```
  <th>Marks</th>
 </tr>
 <tr>
  <td rowspan = "2">Hillary</td>
  <td>Advanced Web</td>
  <td>75</td>
 </tr>
 <tr>
  <td>Operating Syatem</td>
  <td>60</td>
 </tr>
    <tr>
  <td rowspan = "2">Lary</td>
  <td>Advanced Web</td>
  <td>80</td>
 </tr>
 <tr>
  <td>Operating Syatem</td>
  <td>75</td>
 </tr>
 <tr>
  <td colspan="3">Total Average: 72.5</td>
 </tr>
</table>
```

How to Add a Table Header, Body, & Footer in HTML Tables?

Just like how a website or any other document has three main sections – the header, body, and footer – so does a table.

- <thead> - provides a separate haeder for the table
- <tbody> - conatins main content of the table

- <tfoot> - creates a separete footer for the table

Example:

```
<table>
 <thead>
  <tr>
   <th colspan="2">October</th>
   <th colspan="2">November</th>
  </tr>
 </thead>

 <tbody>
  <tr>
   <td>Sales</td>
   <td>Profit</td>
   <td>Sales</td>
   <td>Profit</td>
  </tr>
  <tr>
   <td>$200,00</td>
   <td>$50,00</td>
   <td>$300,000</td>
   <td>$70,000</td>
  </tr>
 </tbody>

 <tfoot>
  <tr>
   <th colspan= "4">November was more produstive</th>
```

```
    </tr>
  </tfoot>
</table>
```

Frame (iframe):

An iFrame, also knowns as Inline Frame, is an element that loads another HTML element inside of a web page. They are commonly used to embed specific content like external ads, videos, tags, or other interactive elements into the page.

Example:

```
<iframe src="/default.asp" width="100%" height="300" style="border:1px solid black;">
    </iframe>
```

HTML Form

Forms are one of the most important parts of the web. Without them, there wouldn't be an easy way to collect data, search for resources, or sign up to receive valuable information.

HTML Forms are required, when you want to collect some data from the site visitor.

Basic HTML Form Syntax:

```
<form action = "Script URL" method = "GET|POST">
        form elements like input, textarea etc.
</form>
```

Following is a list of the most frequently used form attributes –

Action: Backend script ready to process your passed data.

Method: Method to be used to upload data. The most frequently used are GET and POST methods.

Usually, the form information is sent to a **server**.

HTML provides interactive **form controls**:

- **Text inputs (for one or multiple lines)**
- **Radio buttons**
- **Checkboxes**
- **Dropdowns**
- **Select Box Controls**
- **File Select boxes**
- **Clickable Buttons**
- **Reset Button**
- **Submit buttons**

Text Input Controls:

Almost all forms require **textual** input from users, in order for them to enter their name, email, password, address… Text form controls come in different variations:

Attributes: Following is the list of attributes for <input> tag for creating text field.

Sr.No	Attribute & Description
1	**type** Indicates the type of input control and for text input control it will be set to **text**.
2	**name** Used to give a name to the control which is sent to the server to be recognized and get the value.
3	**value** This can be used to provide an initial value inside the control.
4	**size** Allows to specify the width of the text-input control in terms of characters.
5	**maxlength** Allows to specify the maximum number of characters a user can enter into the text box.

Example

Text	`<input type="text">`		Allows any type of character
Email	`<input type="email">`		Might display a warning if an invalid email is entered
Password	`<input type="password">`		Shows dots as characters
Number	`<input type="number">`		Up/Down keyboard keys can be used
Telephone	`<input type="tel">`		Can trigger an autofill
Multiple line text	`<textarea></textarea>`		Can be resized

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Text Input Control</title>
  </head>
  <body>
    <form >
       First name: <input type = "text" name = "first_name" /> <br/>
       Last name: <input type = "text" name = "last_name" /><br/>
      Password: <input type = "password" name = "password" />
Description : <br />
        <textarea rows = "5" cols = "50" name = "description">
                Enter description here...
         </textarea>

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

Placeholders:

Text inputs can display a **placeholder** text that will disappear as soon as some text is entered.

```
<input type="text" placeholder="Enter your name">
```

HTML Label:

The <label> tag in HTML is used to provide a usability improvement for mouse user's i.e., if a user clicks on the text within the <label> element, it toggles the control.

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

- Firstly, use <label> tag by providing the <input> and id attribute. The <label> tag needs a *for* attribute whose value is the same as input id.

- Alternatively, <input> tag use directly inside the <label> tag. In this case, the *for* and id attributes are not needed because the association is implicit.

Syntax:

<label> form content... </label>

There are two ways to associate a text label and the form to which it belongs:

1. *Set the identifier (id) inside the <input> element and specify its name as a for attribute for the <label> tag.*

Example of the HTML <label> tag:

```
<!DOCTYPE html>
<html>
 <head>
  <title>Title of the document</title>
 </head>
 <body>
  <form>
   <label for="lfname">First name:</label>
   <input id="lfname" name="fname" type="text" />
  </form>
 </body>
</html>
```

2. *Insert the <input> into the <label> element.*

Example of the HTML <label> tag with an <input> element inside:

```
<!DOCTYPE html>
<html>
 <head>
  <title>Title of the document</title>
 </head>
 <body>
```

```
    <form>
      <label>Name
        <input id="User" name="Name" type="text" />
      </label>
    </form>
  </body>
</html>
```

Example of HTML <form> with a <label> tag used with radio type of an <input> tag:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title of the document</title>
  </head>
  <body>
    <form>
      <label for="barca">Barcelona</label>
      <input type="radio" name="team" id="barca" value="Barcelona">
      <br />
      <label for="real">Real Madrid</label>
      <input type="radio" name="team" id="real" value="Real Madrid">
      <br />
    </form>
  </body>
</html>
```

HTML | <input type="checkbox">:

The **HTML <input type="checkbox">** is used to define a checkbox field. The checkbox is shown as a square box that is ticked when it is activated. It allows the user to select one or more option among all the limited choices.

Syntax:

```
<input type="checkbox">
```

Example:

```
<form>
  <input type="checkbox" name="favorite_pet" value="Cats"
checked>Cats<br>
      <input type="checkbox" name="favorite_pet"
value="Dogs">Dogs<br>
      <input type="checkbox" name="favorite_pet"
value="Birds">Birds<br>
      <br>
      <input type="submit" value="Submit now" />

</form>
```

HTML <input type=”radio”>

The **HTML <input type=”radio”>** is used to define a Radio Button. Radio Buttons are used to let the user select exactly one option from a list of predefined options. Radio Button input controls are created by using the “input” element with a type attribute having value as “radio”.

Syntax:

```
<input type="radio">
```

Example:

```
<!DOCTYPE html>
<html>
<body>
 <h2>Welcome To GeeksforGeeks</h2>
 <p>Select a Technology Brand:</p>
 <div>
  <input type="radio" id="Netflix" name="brand" value="Netflix">
  <label for="Netflix">Netflix</label>
 </div>
 <div>
```

```
    <input type="radio" id="Audi" name="brand" value="Audi">
    <label for="Audi">Audi</label>
  </div>
  <div>
    <input type="radio" id="Microsoft" name="brand" value="Microsoft" checked>
    <label for="Microsoft">Microsoft</label>
  </div>
 </body>
</html>
```

Dropdown menus

If the number of options to choose from takes up too much space, you can use <select> dropdown menus. They work like radio buttons. Only their layout is different.

Example:

```
<select>
  <option>January</option>
  <option>February</option>
  <option>March</option>
  <option>April</option>
  <option>May</option>
  <option>June</option>
  <option>July</option>
  <option>August</option>
  <option>September</option>
  <option>October</option>
  <option>November</option>
  <option>December</option>
</select>
```

Multiple choice dropdown menus

If you add the multiple attribute, you can provide the ability to select multiple choices.

Example:

```
<label>Which browsers do you have?</label>
<select multiple>
  <option>Google Chrome</option>
  <option>Internet Explorer</option>
  <option>Mozilla Firefox</option>
  <option>Opera</option>
  <option>Safari</option>
</select>
```

Input Type File

This defines a field for file submission. When a user clicks it, they are prompted to insert the desired file type, which might be an image, PDF, document file, and so on.

Example:

```
<input type="file" />
```

Input Type Color

This is a fancy input type introduced by HTML5. With it, the user can submit their favorite color for example. Black (#000000) is the default value, but can be overridden by setting the value to a desired color.

Many developers have used it as a trick to get to select different color shades available in RGB, HSL and alphanumeric formats.

Example:

```
<input type="color" />
```

Input Type Search

Input with the type of search defines a text field just like an input type of text. But this time it has the sole purpose of searching for info. It is different from type text in that, a cancel button appears once the user starts typing.

```
<input type="search" />
```

Input Type Date

You might have registered on a website where you requested the date of a certain event. The site probably used an input with the type value set to date to acheive this.

```
<input type="date" />
```

Input Type Datetime-local

This works like the input type date, but it also lets the user pick a date with a particular time.

```
<input type="datetime-local" />
```

Input Type Week

The input type of week lets a user select a specific week.

```
<input type="week" />
```

Input Type Month

The input with the type of month populates months for the user to pick from when clicked.

```
<input type="month" />
```

Input Type URL

When the type attribute of an input tag is set to URL, it displays a field where users can enter a URL.

```
<input type="url" />
```

Meta Tag:

The <meta> tag defines metadata about an HTML document. Metadata is data (information) about data.

- <meta> tags always go inside the <head> element, and are typically used to specify character set, page description, keywords, author of the document, and viewport settings.
- Metadata will not be displayed on the page, but is machine parsable.
- Metadata is used by browsers (how to display content or reload page), search engines (keywords), and other web services.

Example:

```
<head>
  <meta charset="UTF-8">
```

```
<meta name="description" content="Free Web tutorials"/>
<meta name="keywords" content="HTML, CSS, JavaScript"/>
<meta name="author" content="John Doe"/>
<meta name="Revised" content="codeacademy , 3/7/2022"/>
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
</head>
```

Define keywords for search engines:

<meta name="keywords" content="HTML, CSS, JavaScript">

Define a description of your web page:

<meta name="description" content="Free Web tutorials for HTML and CSS">

Define the author of a page:

<meta name="author" content="John Doe">

Refresh document every 30 seconds:

<meta http-equiv="refresh" content="30">

Setting the viewport to make your website look good on all devices:

<meta name="viewport" content="width=device-width, initial-scale=1.0">

HTML | <audio>:

The **HTML <audio> controls attribute** is used to specify the control to play audio. It is the Boolean value. This attribute is new in HTML5. Currently there are three supported file format for HTML 5 audio tag.

1. mp3
2. wav
3. ogg

Attributes of HTML Audio Tag:

Attribute	Description
controls	It defines the audio controls which is displayed with play/pause buttons.

autoplay	It specifies that the audio will start playing as soon as it is ready.
loop	It specifies that the audio file will start over again, every time when it is completed.
muted	It is used to mute the audio output.
preload	It specifies the author view to upload audio file when the page loads.
src	It specifies the source URL of the audio file.

Syntax:

<audio controls>

HTML Audio Tag Example

```
<audio controls preload="none">
    <source src="koyal.mp3" type="audio/mpeg">
    Your browser does not support the html audio tag.
</audio>
```

HTML Video Tag

HTML 5 supports <video> tag also. The HTML video tag is used for streaming video files such as a movie clip, song clip on the web page.

Currently, there are three video formats supported for HTML video tag:

1. mp4
2. webM
3. ogg

Attributes of HTML Video Tag

Attribute	Description

controls	It defines the video controls which is displayed with play/pause buttons.
height	It is used to set the height of the video player.
width	It is used to set the width of the video player.
poster	It specifies the image which is displayed on the screen when the video is not played.
autoplay	It specifies that the video will start playing as soon as it is ready.
loop	It specifies that the video file will start over again, every time when it is completed.
muted	It is used to mute the video output.
preload	It specifies the author view to upload video file when the page loads.
src	It specifies the source URL of the video file.

HTML Video Tag Attribute Example

```
<video width="320" height="240" controls autoplay loop>
  <source src="movie.mp4" type="video/mp4">
  Your browser does not support the html video tag.
</video>
```

Canvas, Main, Section, Article, Header, Footer, Aside, Nav, Figure Tags

HTML Semantic Elements:

Semantic elements = elements with a meaning.

What are Semantic Elements?

A semantic element clearly describes its meaning to both the browser and the developer.

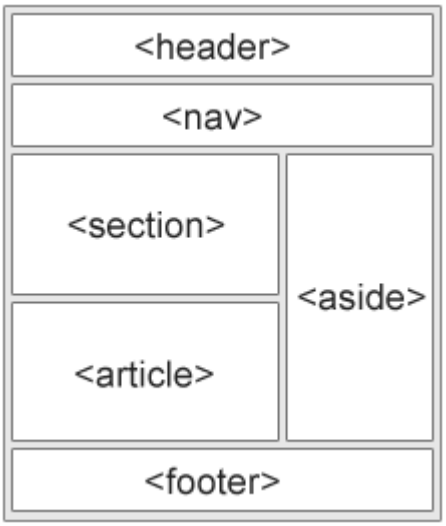
Examples of non-semantic elements: and - Tells nothing about its content.

Examples of semantic elements: <form>, <table>, and <article> - Clearly defines its content.

Semantic Elements in HTML

Many web sites contain HTML code like: to indicate navigation, header, and footer.

In HTML there are some semantic elements that can be used to define different parts of a web page:



HTML5 includes **new tags** or **new elements**. These tags or elements includes new media tags, **semantic tags**, svg, canvas etc. HTML5 is now more **semantic** than html4. Here is a **list of new tags in html5, with their meaning, use and example.**

HTML5 New Tags List:

- Article
- Section
- Aside
- Details
- Summary
- Time
- Figure
- Figcaption
- Picture

- Footer
- Header
- Mark
- Meter
- Nav
- Progress
- HTML5 Canvas
- HTML5 SVG
- Wbr

Article

An **article** defines a complete or an *Self Contained composition* in a webpage. An **Article** could be a *Blog post*, *forum*, *newspaper article*, *an independent content*, *user comment*.

An **article** must have an heading or subheading, i.e h1-h6

```
<article>
   <h2>Heading for Article</h2>
</article>
```

Section

An **Section** represents an generic section of a document. **Section** could be various sections of an article with heading.

An section can include various chapters of a book or various webpage sections, like *introductions*, *content*, *contact info* etc.

An **section** must have an heading. h2-h6

```
<article>
  <h1>Article Heading</h1>
<section>
```

```
    <h2>Subheading 1 for Section</h2>
</section>
<section>
    <h2>Subheading 2 for Section</h2>
</section>
</article>
```

Aside

Aside is the sidebar of a container. The content inside **aside** is related to content next to it. Aside could be sibling of div, section or article. But the content inside **aside** is relevant to adjacent sibling.

aside is use for links, sidebars, for ads and other content we want to put aside.

```
<aside>
  <p>Aside</p>
</aside>
```

Details

Defines additional **details** that the user can view or hide. Content inside <details> is hidden. Only <summary> is visible to user. User can click on summary to view **details** .

Summary

Summary is the visible part of details. Except summary, everything details is hidden.

```
<details>
    <summary>Click To See</summary>
    <p>Hello</p>
</details>
```

Details Example with open

An additional **open attribute** can open details on page load..

```
<details open>
   <summary>Click To See</summary>
   <p>Hello</p>
</details>
```

Time

Time tag is an inline level element, used to represents a time or exact date in Gregorian calender.

```
<p> New Batch Timings are <time>10:00</time> AM.</p>
```

Time Tag with datetime

An additional **datetime** attribute can write **datetime** in *ISO String*.

```
<p> New Batch Timings are <time datetime="2017-12-08T10:00">10:00</time> AM.</p>
```

Figure

Figure tag Specifies **self-contained content**, like images, illustrations, diagrams, code listings, etc. Figure can have **figcaption** child to explain what **figure** is showing.

Figcaption

Figcaption is the caption of figure element.

```
<figure>
   <img src="img/path" alt="...">
   <figcaption> caption for image</figcaption>
</figure>
```

Picture

HTML5 **picture tag** is used to show either high or low resolution image for Desktop, Mobile or particular device. We can set two or more different images for different screens or resolutions, and browser will load a single resource from server, based on criteria. Like High quality images for Macbook and Full HD screens, medium quality images for normal screens and small image for mobiles.

Picture tags can enhance performance of a website.

```
<picture>
   <source srcset="img/logo.webp" type="image/webp">
   <source srcset="img/logo.jpg" type="image/jpeg">
   <img src="img/logo.jpg" alt="Tech Altum Logo">
</picture>
```

Header

Defines a **header** for a document, section or an article. **Header** can be used more than once on a single webpage. But try to use single **header** in a section, or an article.

```
<section>
   <header>
      <h2>Header for Section<h2>
   </header>
   <p>Section Content Goes Here...</p>
</section>
```

Footer

Footer tag defines footer of an document, section or an article. A webpage can have multiple footers, but in separate sections.

```
<!DOCTYPE html>
<html>
<body>
<footer>
  <p>Author: Hege Refsnes</p>
  <p><a href="mailto:hege@example.com">hege@example.com</a></p>
```

```
</footer>

</body>

</html>
```

Mark

Mark is and inline level element used as an *highlighter* for text. Default background-color of **mark tag** is yellow, and default font color is black.

```
<p>This text is <mark>marked<>/mark</p>
```

Nav

Nav Tag is used to group *navigational links*. Nav can be used twice or more in a single webpage.

```
<!DOCTYPE html>

<html>

<body>

<nav>

  <a href="/html/">HTML</a> |

  <a href="/css/">CSS</a> |

  <a href="/js/">JavaScript</a> |

  <a href="/jquery/">jQuery</a>

</nav>

</body>

</html>
```

HTML Events:

In a web page or a website created using the HTML script (HyperText Markup Language), every action performed by the user and the web tool itself is termed an HTML Event.

The description contains six types of attributes for every Event: *window event attribute, form event attribute, keyboard event attribute, mouse event attribute, media event attribute, and clipboard event attribute.*

There are many event attributes available in HTML5, which are classified primarily into 6 different types. These attributes work using JavaScript language.

- **Windows Event Attributes:** This is provided for the action of the windows object. It works in body tag <body>.
- **Form Event Attributes:** If the user performs some action in the form like input data, cancel, submit, then these event attributes works.
- **Keyboard Event Attributes:** This keyboard event attributes used for keyboard action and user interaction.
- **Mouse Event Attributes:** Mouse event attribute used for mouse action that is move, click, wheel, etc.
- **Clipboard Event Attributes:** This event attributes used for clipboard action: example, cut, copy, and pest.
- **Media Event Attributes:** This event attribute works on media files like video etc.

1. Windows Event Attributes

Attribute	Description
onafterprint	This script act, though, after the document printed.
onbeforeprint	This script act, though, before a document printed.
onbeforeunload	Whereas before the unloaded document, this Script works.
onerror	In the document occurs the Error then this event executed.
onhashchange	The anchor part of URL change in the document that time event executed.

onload	When the first Web page is loaded completely, then this event executed.
onmessage	In the document, the message that occurred at that time event executed.
onoffline	If the network connection is unavailable and the browser says offline, then the event executed.
ononline	When the network available in the browser, then the event executed.
onpagehide	This script act; if the user not working on a current webpage, a then-current page can be hidden.
onpageshow	This script act at that time the current webpage is load.
onpopstate	This script automatically works on the browser for a history state change.
onresize	This script act when the browser of the window changes the size.
onstorage	When users web storage updated, then the event executed.

onunload	The user's current web page is not loaded or the window is closed, then the event is executed.

2. Form Event Attributes

Attribute	Description
onblur	Some form validation object loos the focus, then event fired.
onchange	The value change in the form, then event fired.
onfocus	In the form <input>, <a> , <select> object has focus. Working on this object then event fired.
oninput	The user gives input of value in the form then this event fired.
oninvalid	The event works on when the element does not satisfy its predefined constraints.
onreset	User reset the form information, then event fired.
onsearch	Users search the required field, then event fired.

onselect	The user selects the text or text area in form, then event fired.
onsubmit	The user submits the form at the end then the event fired.

3. Keyboard Event Attributes

Attribute	Description
onkeydown	Using a keyboard, the user press the key down at that point event works
onkeypress	Using the keyboard, users press the key and display characters at that point event works.
onkeyup	After the press, the key user releases the key then the event works.

4. Mouse Event Attributes

Attribute	Description

onclick	The user clicks the mouse on the button then an event occurred.
ondblclick	Users double click the mouse then the event occurred.
onmousedown	The user presses the mouse button on the element then the event occurred.
onmousemove	The user moves the mouse pointer over the element then the event occurred.
onmouseout	The user moves the mouse outside of the element then the event occurred.
onmouseover	The user moves the mouse over the element then the event occurred.
onmouseup	The user released the mouse button then the event occurred.
onmousewheel	Using the mouse wheel user rolls the up and down on element then the event occurred.
onwheel	Using a mouse wheel user roll them up and down then the event occurred.

5. Clipboard Event Attributes

Attribute	Description
oncopy	Using mouse users to copy the content, then the event occurred.
oncut	Using a mouse, users cut the content then the event occurred.
onpaste	Using a mouse user, paste the content, then an event occurred.

6. Media Event Attributes

Attribute	Description
onabort	When media files aborted for download and play again, then an event occurs.
oncanplay	When any media file ready for play, then this trigger is fired.
oncanplaythrough	Media file ready to play without buffering and loading.
oncuechange	Element changes the cue of <track> then event fired.

ondurationchange	The Media file changes the length of time then the trigger is fired.
onemptied	If the Media file unavailable and come fatal error, then the trigger is fired.
onended	The Media file comes on endpoint then the trigger is fired.
onerror	When an error occurred to get the media file, the trigger is fired
onloadeddata	The Media file loads the data then the trigger is fired.
onloadedmetadata	The Media file loads the metadata then the trigger is fired.
onloadstart	The Media file starts to load then the trigger is fired.
onpause	The Media file paused to play again then the trigger is fired.
onplay	Media file ready to play, then trigger is fired.
onplaying	The Media file starts to play when the trigger is fired.

onprogress	This script act when the browser is working on connecting with the media data.
onratechange	If the videos playback speed is changed, then the trigger is fired.
onseeked	Users completed moving; otherwise, skip the new position of video. this attribute set as false.
onseeking	The user wants to move; otherwise, skip the new position of the video. this attribute set as true.
onstalled	When the browser suddenly stops to the connection of data, then the event works.
onsuspend	When the web Browser on purpose does not get media data, then events work.
ontimeupdate	When a user changes the video play position like forward and backward.
onvolumechange	To change media volume low to high.

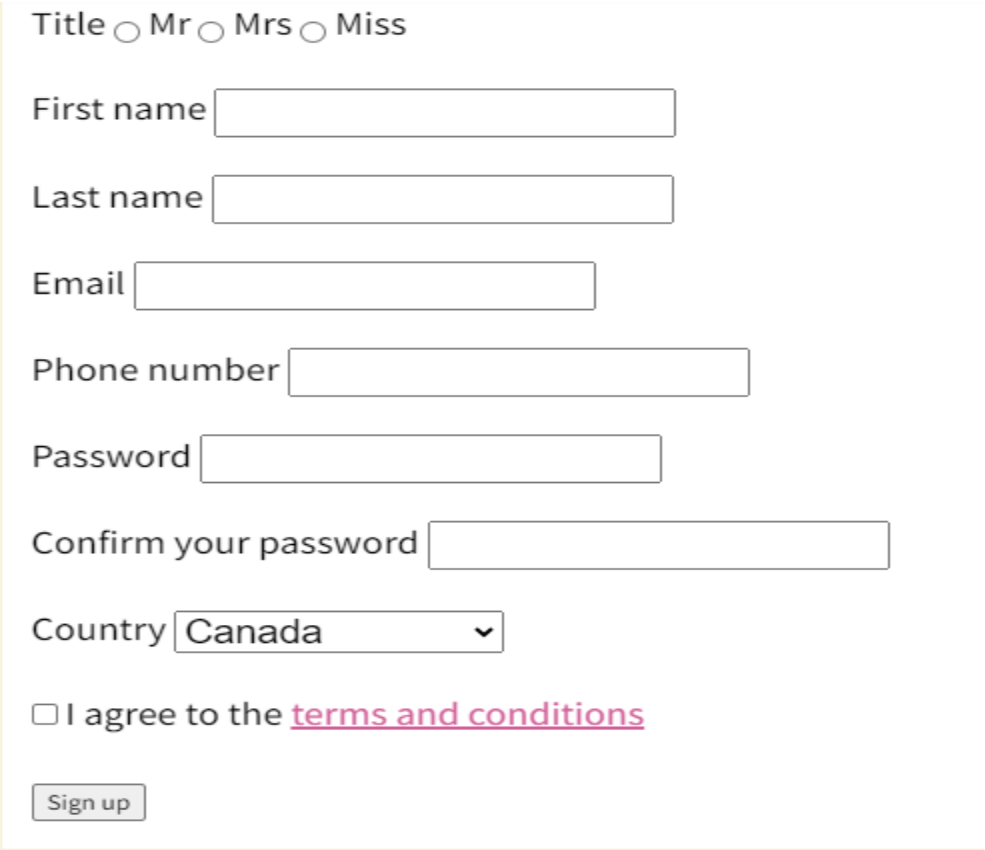
onwaiting	If the data load the information, current video stop with buffering then event works.

Html Events are an easy way to take action and interface between web browsers and users. Users can get the reaction of every action effortless. For example, cut and copy for the clipboard, move and click for mouse, buttons for form. Events make web applications real-time and make smart work systems.

THE END

Assignments-HTML

1. Write a HTML code for the following form-1.



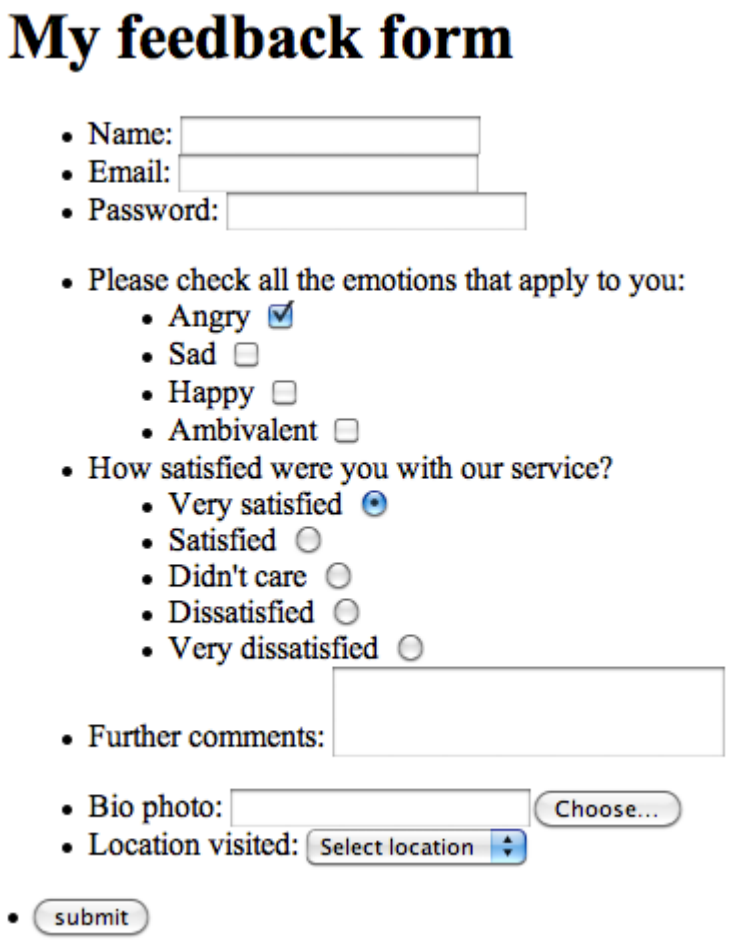
Solution:

```
<form action="/signup" method="POST">
 <p>
  <label>Title</label>
  <label>
   <input type="radio" name="title" value="mr">
   Mr
  </label>
  <label>
   <input type="radio" name="title" value="mrs">
   Mrs
  </label>
  <label>
   <input type="radio" name="title" value="miss">
   Miss
  </label>
 </p>
 <p>
  <label>First name</label>
  <input type="text" value="first_name">
 </p>
 <p>
  <label>Last name</label>
  <input type="text" value="last_name">
 </p>
 <p>
  <label>Email</label>
  <input type="email" value="email">
 </p>
 <p>
```

```
    <label>Phone number</label>
    <input type="tel" value="phone">
  </p>
  <p>
    <label>Password</label>
    <input type="password" value="password">
  </p>
  <p>
    <label>Confirm your password</label>
    <input type="password"
value="password_confirm">
  </p>
  <p>
    <label>Country</label>
    <select>
      <option>Canada</option>
      <option>France</option>
      <option>Germany</option>
      <option>Italy</option>
      <option>Japan</option>
      <option>Russia</option>
      <option>United Kingdom</option>
      <option>United States</option>
    </select>
  </p>
  <p>
    <label>
      <input type="checkbox" value="terms">
      I agree to the <a href="/terms">terms and
conditions</a>
```

```
    </label>
  </p>
  <p>
    <button>
      Sign up
    </button>
  </p>
</form>
```

2. Write a HTML code for the following form-2.



Solution:

```
<form id="contact-form" action="script.php"
method="post">
<ul>
```

```
<li><label for="name">Name:</label> <input type="text"
name="name" id="name" value="" /></li>
<li><label for="email">Email:</label> <input type="text"
name="email" id="email" value="" /></li>
<li><label for="pwd">Password:</label> <input
type="password" name="pwd" id="pwd" value="" /></li>
</ul>

<ul>
  <li>Please check all the emotions that apply to you:
    <ul>
      <li><label for="angry">Angry</label> <input
type="checkbox" name="angry" id="angry"
value="angry"></li>
      <li><label for="sad">Sad</label> <input
type="checkbox" name="sad" id="sad" value="sad"></li>
      <li><label for="happy">Happy</label> <input
type="checkbox" name="happy" id="happy"
value="happy"></li>
      <li><label for="ambivalent">Ambivalent</label> <input
type="checkbox" name="ambivalent" id="ambivalent"
value="ambivalent"></li>
    </ul>
  </li>
  <li>How satisfied were you with our service?
    <ul>
      <li><label for="vsat">Very satisfied</label> <input
type="radio" name="satisfaction" id="vsat"
value="vsat"></li>
```

```
      <li><label for="sat">Satisfied</label> <input
type="radio" name="satisfaction" id="sat" value="sat"></li>
      <li><label for="dcare">Didn't care</label> <input
type="radio" name="satisfaction" id="dcare"
value="dcare"></li>
      <li><label for="disat">Dissatisfied</label> <input
type="radio" name="satisfaction" id="disat"
value="disat"></li>
      <li><label for="vdisat">Very dissatisfied</label> <input
type="radio" name="satisfaction" id="vdisat"
value="vdisat"></li>
    </ul>
  </li>
  <li><label for="comments">Further comments:</label>
<textarea name="comments" id="comments" cols="25"
rows="3"></textarea></li>
</ul>

<ul>
  <li><label for="photo">Bio photo:</label> <input
type="file" name="photo" id="photo" value="" /></li>
  <li><label for="location">Location visited:</label>
  <select name="location" id="location">
    <option value="">Select location</option>
    <option value="nyork">New York</option>
    <option value="vancouver">Vancouver</option>
    <option value="atlantis">Atlantis</option>
    <option value="alpha">Alpha Centauri</option>
    <option value="blackpool">Blackpool</option>
    <option value="bognor">Bognor Regis</option>
```

```
  </select></li>
</ul>

<li><input type="submit" value="submit" /></li>
</form>
```

********THE-END********