

Web Programming Paper Solution (Chapter wise)

HTML 5.0

HTML 5.0 and the difference from HTML 4.

HTML5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and current version of the HTML standard.

It was published in October 2014 by the World Wide Web Consortium (W3C) to improve the language with support for the latest multimedia, while keeping it easily readable by humans—and consistently understood by computers and devices such as web browsers, parsers, etc. HTML5 is intended to subsume not only HTML 4, but also XHTML 1 and DOM Level 2 HTML.

HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and application programming interfaces (APIs) for complex web applications. For the same reasons, HTML5 is also a candidate for cross-platform mobile applications, because it includes features designed with low-powered devices such as smartphones and tablets.

Many new syntactic features are included. To natively include and handle multimedia and graphical content, the new <video>, <audio> and <canvas> elements were added, and support for scalable vector graphics (SVG) content and MathML for mathematical formulas. To enrich the semantic content of documents, new page structure elements such as <main>, <section>, <article>, <header>, <footer>, <aside>, <nav> and <figure>, are added. New attributes are introduced, some elements and attributes have been removed, and others such as <a>, <cite> and <menu> have been changed, redefined or standardized.

It comes with a new introductory line that looks like an SGML document type declaration, <!DOCTYPE html>. In addition to specifying markup, HTML5 specifies scripting application programming interfaces (APIs) that can be used with JavaScript such as Canvas, Timed Media Playback, Offline, Editable content, Drag-and-drop, History, MIME type and protocol handler registration, Microdata, Web Messaging, Web Storage.

Web Programming Paper Solution (Chapter wise)

HTML5 v/s HTML4?

Topic	HTML 4	HTML 5
Year of proposal	1997	2012
Doctype	based on SGML <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">	<!doctype HTML>
Audio and video	Not part of HTML4 specification, external plug ins required	Use of <audio> and <video> tags
Vector graphics	With use of external plug ins like Flash, Silverlight	SVG support and <canvas> tag
User location	Impossible to get true location	Possible using API JS GeoLocation
Cache	Only browser cache	Application cache, SQL cache, web storage accessible via JS
Web sockets	Not available	Available as full duplex communication for streaming of data
JS runs as	Runs as separate thread	JS can run in background
browser support	By new and old browsers	Updated new browsers
href in <a>	Can be blank	Can't be blank
<acronym>	Not supported	Supported to create acronym in web page
<applet>	Not supported	Supports applet in web page
	Used as emphasized text	Defines important text
<body>	All body attributes were deprecated in HTML 4.1 onwards	Attributes not supported
<hr>	Used for horizontal rule	Used to create thematic break to separate content
<map>	No name map is allowed	Map should have name
<meta>		
<script>	Requires type attribute	Considers JS
<table> border	Can have any value	Only "" and "1" supported
Structural elements	Not supported by tags, uses <table>, <frame> tags	Uses <article>, <aside>, <command>, <details>, <header>, <footer>, <nav>, etc
Media tags	Not supported	<audio>, <video>, <source>, <embed>, <track>
Draw graphics on run	Not supported	Using <canvas>
New form elements	Not applicable	<datalist>, <keygen>, <ouput>
Framing	<frame>, <frameset>, <noframe>	Not supported

Web Programming Paper Solution (Chapter wise)

HTML code to display class timetable.

```
<!doctype HTML>
<html>
  <head>
    <title>Class Timetable</title>
  </head>
  <body>
    <table border="1">
      <caption align="top">Class Timetable</caption>
      <tr>
        <td></td>
        <th>Day</th>
        <th>Saturday</th>
        <th>Sunday</th>
      </tr>
      <tr>
        <th rowspan="2">Time</th>
        <th>9.00 to 11.00</th>
        <td colspan="2">WP</td>
      </tr>
      <tr>
        <th>12.00 to 1.00</th>
        <td>CN</td>
        <td>AM4</td>
      </tr>
    </table>
  </body>
</html>
```

Class Timetable

	Day	Saturday	Sunday
Time	9.00 to 11.00	WP	
	12.00 to 1.00	CN	AM4

Web Programming Paper Solution (Chapter wise)

Links in HTML.

The HTML Anchor Element (<a>) defines a hyperlink to a location on the same page or any other page on the Web. It can also be used (in an obsolete way) to create an anchor point—a destination for hyperlinks within the content of a page, so that links aren't limited to connecting simply to the top of a page.

Example:

```
<a href="https://xyz.com">XYZ</a>
```

Attributes:

- **download:** This attribute, if present, indicates that the author intends the hyperlink to be used for downloading a resource so that when the user clicks on the link they will be prompted to save it as a local file. If the attribute has a value, the value will be used as the pre-filled file name in the Save prompt that opens when the user clicks on the link .
- **href:** This was the single required attribute for anchors defining a hypertext source link, but is no longer required in HTML5. Omitting this attribute creates a placeholder link. The href attribute indicates the link target, either a URL or a URL fragment.
- **rel:** For anchors containing the href attribute, this attribute specifies the relationship of the target object to the link object.
- **target:** This attribute specifies where to display the linked resource. In HTML4, this is the name of, or a keyword for, a frame.

_self: Load the response into the same HTML4 frame (or HTML5 browsing context) as the current one. This value is the default if the attribute is not specified.

_blank: Load the response into a new unnamed HTML4 window or HTML5 browsing context.

_parent: Load the response into the HTML4 frameset parent of the current frame or HTML5 parent browsing context of the current one. If there is no parent, this option behaves the same way as **_self**.

_top: In HTML4: Load the response into the full, original window, canceling all other frames. In HTML5: Load the response into the top-level browsing context (that is, the browsing context that is an ancestor of the current one, and has no parent). If there is no parent, this option behaves the same way as **_self**.

obsolete attributes in HTML5

charset defines the character encoding

coords for use with object shapes

name required in an anchor defining a target location within a page.

shape used to define a selectable region for hypertext source links associated with a figure to create an image map

Web Programming Paper Solution (Chapter wise)

<form>, <table>, <iframe>, <!doctype> tags (XML DTD)

- **<table>**

The HTML Table Element (<table>) represents tabular data: information expressed via two dimensions or more.

<caption>, <tr>, <th>, <td>, <thead>, <tbody> are tags used to define internal structure and data of a table.

Attributes

align(deprecated): This enumerated attribute indicates how the table must be aligned in regard of the containing document. (left/ right/ center) insted use margin in CSS

bgcolor(deprecated): This attribute defines the background color of the table and its content. Specified in hexa or color name, use background-color in CSS instead.

border: specifies table is bordered or not, can have value either "" or "1"

cellpadding(deprecated): This attribute defines the space between the content of a cell and the border,instead use border-collapse in CSS.

cellspacing(deprecated): This attribute defines the size, in percentage or in pixels, of the space between two cells (both horizontally and vertically), between the top of the table and the cells of the first row, the left of the table and the first column, the right of the table and the last column and the bottom of the table and the last row. Use border-spacing instead in CSS

Ex

```
<table>
<tr>
  <td>John</td>
  <td>Doe</td>
</tr>
<tr>
  <td>Jane</td>
  <td>Doe</td>
</tr>
</table>
```

- **<form>** tag

The HTML <form> element represents a document section that contains interactive controls to submit information to a web server.

<label>, <input>, <textarea>, <select> are some of the tags used to define interactive controls in form.

Attributes:

action: The URI of a program that processes the form information. This value can be overridden by a formaction attribute on a <button> or <input> element.

autocomplete: Indicates whether input elements can by default have their values automatically completed by the browser. This setting can be overridden by an autocomplete attribute on an element belonging to the form. Possible values are on/off

enctype: When the value of the method attribute is post, enctype is the MIME type of content that is used to submit the form to the server. Possible values are:

application/x-www-form-urlencoded: The default value if the attribute is not specified.

multipart/form-data: The value used for an <input> element with the type attribute set to "file".

text/plain (HTML5)

Web Programming Paper Solution (Chapter wise)

method : The HTTP method that the browser uses to submit the form. Possible values are:

post: Corresponds to the HTTP POST method ; form data are included in the body of the form and sent to the server.

get: Corresponds to the HTTP GET method; form data are appended to the action attribute URI with a '?' as separator, and the resulting URI is sent to the server. Use this method when the form has no side-effects and contains only ASCII characters.

This value can be overridden by a formmethod attribute on a <button> or <input> element.

name: The name of the form (unique and non empty)

target: A name or keyword indicating where to display the response that is received after submitting the form. Possible values are _self, _blank, _parent, _top, iFrameName

Ex:

```
<form action="xyz.php" method="post">
  <label for="POST-name">Name:</label>
  <input id="POST-name" type="text">
  <input type="submit" value="Save">
</form>
```

- **<iframe> tag**

The HTML Inline Frame Element (<iframe>) represents a nested browsing context, effectively embedding another HTML page into the current page. In HTML 4.01, a document may contain a head and a body or a head and a frameset, but not both a body and a frameset. However, an <iframe> can be used within a normal document body. Each browsing context has its own session history and active document.

Attributes:

allowfullscreen: This attribute can be set to true if the frame is allowed to be placed into full screen mode by calling its Element.requestFullscreen() method

name: A name for the embedded browsing context (or frame). This can be used as the value of the target attribute of an <a> or <form> element, or the formtarget attribute of an <input> or <button> element.

src: The URL of the page to embed.

Ex

```
<iframe src="https://mdn-samples.mozilla.org/snippets/html/iframe-simple-contents.html"
width="400" height="300">
  <p>Your browser does not support iframes.</p>
</iframe>
```

- **<!doctype> tag / DTD in XML**

<!DOCTYPE> informs the browser which version of HTML (or XML) you used to write the document. Doctype is a declaration, not a tag; you can also refer to it as "document type declaration", or "DTD" for short.

A document type declaration, or DOCTYPE, is an instruction that associates a particular SGML or XML document (for example, a webpage) with a document type definition (DTD) (for example, the formal definition of a particular version of HTML). In the serialized form of the document, it manifests as a short string of markup that conforms to a particular syntax.

Web Programming Paper Solution (Chapter wise)

The HTML layout engines in modern web browsers perform DOCTYPE "sniffing" or "switching", wherein the DOCTYPE in a document served as text/html determines a layout mode, such as "quirks mode" or "standards mode". The text/html serialization of HTML5, which is not SGML-based, uses the DOCTYPE only for mode selection. Since web browsers are implemented with special-purpose HTML parsers, rather than general-purpose DTD-based parsers, they don't use DTDs and will never access them even if a URL is provided. The DOCTYPE is retained in HTML5 as a "mostly useless, but required" header only to trigger "standards mode" in common browsers.

Syntax:

```
<!DOCTYPE root-element PUBLIC "FPI" ["URI"] [  
<!-- internal subset declarations -->  
>
```

Ex

in HTML4

```
<!DOCTYPE html PUBLIC  
"-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

in HTML5

```
<!doctype html>
```

in XML

An XML document with correct syntax is called "Well Formed". An XML document validated against a DTD is both "Well Formed" and "Valid".

A "Valid" XML document is a "Well Formed" XML document, which also conforms to the rules of a DTD:

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE note SYSTEM "Note.dtd">  
<note>  
<to>Tove</to>  
<from>Jani</from>  
<heading>Reminder</heading>  
<body>Don't forget me this weekend!</body>  
</note>
```

Note.dtd

```
<!DOCTYPE note  
[  
<!ELEMENT note (to,from,heading,body)>  
<!ELEMENT to (#PCDATA)>  
<!ELEMENT from (#PCDATA)>  
<!ELEMENT heading (#PCDATA)>  
<!ELEMENT body (#PCDATA)>  
>
```

PCDATA means parse-able text data.

Web Programming Paper Solution (Chapter wise)

HTML code to illustrate lists.

```
<!doctype html>
<html>
  <head>
    <title>Different types of list</title>
  </head>
  <body>
    <!-- numbered ordered list where order of each element is important -->
    <!-- type can be 1/ a/ A/ i/ I
         start will be starting index in number no matter what is type -->
    <ol type="1" start="2">
      <li>Two</li>
      <li>Three</li>
    <!-- value to specify new index value breaking sequence -->
      <li value="5">Five</li>
    </ol>
    <!-- alphabet (small/caps) ordered list -->
    <ol type="a">
      <li>xyz</li>
      <li>def</li>
      <li>cde</li>
      <li type="A">abc</li>
    </ol>
    <!-- roman numbered (small/big) ordered list -->
    <ol type="i">
      <li>Two</li>
      <li>Three</li>
      <li value="4" type="I">Five</li>
    </ol>

    <!-- unordered list where order is not important
         type can be circle/ disc/ square
         custom bullets can be obtained using CSS -->
    <ul>
      <li>disc</li>
      <li type="circle">circle</li>
      <li type="square">square</li>
    </ul>
    <!-- definition list
         dd to define definition part
         dt for term part -->
    <dl>
      <dt>Term</dt>
      <dd>definition</dd>
      <dt>WP</dt>
      <dd>Web Programming</dd>
    </dl>
  </body>
```

Web Programming Paper Solution (Chapter wise)

</html>

2. Two
3. Three
5. Five

- a. xyz
- b. def
- c. cde
- D. abc

- i. Two
- ii. Three
- IV. Five

- disc
- circle
- square

Term

definition

WP

Web Programming

Web Programming Paper Solution (Chapter wise)

HTML code to create customer registration form.

```
<!doctype html>
<html>
  <head>
    <title>Customer Registration Form</title>
    <style>
      label{
        display: inline-block; min-width: 100px; vertical-align: top;
      }
    </style>
  </head>
  <body>
    <form action="form_process.php" method="post" name="registration_form">
      <fieldset>
        <legend>Customer Registration Form</legend>
        <label>Name</label>
        <input type="text" name="custName" value="" placeholder="Enter your name"
required />
        <br />
        <label>Password</label>
        <input type="password" name="custPass" value="" placeholder="Enter 8 char
long password" required />
        <br />
        <label>Gender</label>
        Male <input type="radio" name="custGender" value="M" checked />
        &nbsp; Female <input type="radio" name="custGender" value="F" />
        <br />
        <label>What you own?</label>
        <input type="checkbox" name="custOwn[]" value="car">Car
        <input type="checkbox" name="custOwn[]" value="smartphone"
checked>Smart Phone
        <input type="checkbox" name="custOwn[]" value="computer"> PC/Laptop
        <br />
        <label>Address</label>
        <textarea name="custAddr" placeholder="Enter address for
communication"></textarea>
        <br />
        <label>Age group</label>
        <select name="custAge">
          <option selected value="adult">18+</option>
          <option value="child">below 18</option>
        </select>
        <br />
        <label></label>
        <input type="submit" name="submitForm" value="Register" />
      </fieldset>
    </form>
  </body>
```

Web Programming Paper Solution (Chapter wise)

</html>

Customer Registration Form

Name	<input type="text" value="Enter your name"/>
Password	<input type="text" value="Enter 8 char long password"/>
Gender	Male <input checked="" type="radio"/> Female <input type="radio"/>
What you own?	<input type="checkbox"/> Car <input checked="" type="checkbox"/> Smart Phone <input type="checkbox"/> PC/Laptop
Address	<input type="text" value="Enter address for communication"/>
Age group	<input type="text" value="18+"/>
<input type="button" value="Register"/>	

Web Programming Paper Solution (Chapter wise)

 v/s <div>

- Every HTML element has a default display value depending on what type of element it is. The default display value for most elements is block or inline. A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can). An inline element does not start on a new line and only takes up as much width as necessary.

<div> is block level element while is inline element.

- <div> is used as container to other HTML elements, is often used as a container for some text.
- <div> is basically equivalent to having a line-break before and after it, which is not present in
- HTML5 have define some alternatives for <div> such as <header>, <footer>, <nav>
- Ex

The <div> element is a block-level element.

The element is inline element.