HTML & CSS Basics

Mohammad Zarif

Menu

- HTML Forms (Cont'd)
  - <label>
  - <button>
  - <textarea>
  - <select>
  - <fieldset>
  - <legend>
  - <datalist>

- Positions
  - Static
  - Relative
  - Fixed
  - Absolute
  - Sticky

  float

- <video>
  - Autoplay
  - Source

- <audio>
  - Controls
  - Source

HTML Forms

<label>:

- The <label> tag defines a label for many form elements. The <label> element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.
- The for attribute of the <label> tag should be equal to the id attribute of the <input> element to bind them together.

<button>

- The <button> tag defines a clickable button.
- Inside a <button> element you can put text (and tags like ,
, <img>, etc.). That is not possible with a button created with the <input> element!
- Always specify the type attribute for a <button> element, to tell browsers what type of button it is.

<textarea>

- The <textarea> element defines a multi-line input field (a text area)
- The rows attribute specifies the visible number of lines in a text area.
- The cols attribute specifies the visible width of a text area.

<fieldset> & <legend>

- The <fieldset> element is used to group related data in a form.
- The <legend> element defines a caption for the <fieldset> element.

<datalist>

- The <datalist> element specifies a list of pre-defined options for an <input> element.
- Users will see a drop-down list of the pre-defined options as they input data.
- The list attribute of the <input> element, must refer to the id attribute of the <datalist> element.

<datalist>

- Syntax:

```
<input list="browsers">
<datalist id="browsers">
  <option value="Internet Explorer">
  <option value="Firefox">
  <option value="Chrome">
  <option value="Opera">
  <option value="Safari">
</datalist>
```

Positions(Static)

position: static;

- HTML elements are positioned static by default.
- Static positioned elements are not affected by the top, bottom, left, and right properties.
- An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page

Positions(Relative)

position: relative;

- An element with position: relative; is positioned relative to its normal position.
- Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.

Positions(Fixed)

position: fixed;

- An element with position: fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.
- A fixed element does not leave a gap in the page where it would normally have been located.

Position(Absolute)

position: absolute;

- An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).
- However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.
- Note: Absolute positioned elements are removed from the normal flow, and can overlap elements.

Position(Sticky)

position: sticky;

- An element with position: sticky; is positioned based on the user's scroll position.
- A sticky element toggles between relative and fixed, depending on the scroll position. It is positioned relative until a given offset position is met in the viewport - then it "sticks" in place (like position:fixed).

Float

- The CSS float property specifies how an element should float.
- The CSS clear property specifies what elements can float beside the cleared element and on which side.
- The float property is used for positioning and formatting content e.g. let an image float left to the text in a container.

Float

The float property can have one of the following values:

- left - The element floats to the left of its container
- right - The element floats to the right of its container
- none - The element does not float (will be displayed just where it occurs in the text). This is default

Clear

- When we use the float property, and we want the next element below (not on right or left), we will have to use the clear property.
- The clear property specifies what should happen with the element that is next to a floating element.

Clear

The clear property can have one of the following values:

- none - The element is not pushed below left or right floated elements. This is default
- left - The element is pushed below left floated elements
- right - The element is pushed below right floated elements
- both - The element is pushed below both left and right floated elements

When clearing floats, you should match the clear to the float: If an element is floated to the left, then you should clear to the left. Your floated element will continue to float, but the cleared element will appear below it on the web page.

<video>

- The HTML <video> element is used to show a video on a web page.
- The controls attribute adds video controls, like play, pause, and volume.
- It is a good idea to always include width and height attributes. If height and width are not set, the page might flicker while the video loads.
- The <source> element allows you to specify alternative video files which the browser may choose from. The browser will use the first recognized format.
- The text between the <video> and </video> tags will only be displayed in browsers that do not support the <video> element.

<video> (autoplay)

- To start a video automatically, use the autoplay attribute
- Add muted after autoplay to let your video start playing automatically (but muted)

<audio>

- The HTML <audio> element is used to play an audio file on a web page.
- The controls attribute adds audio controls, like play, pause, and volume.
- The <source> element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.
- The text between the <audio> and </audio> tags will only be displayed in browsers that do not support the <audio> element

<audio> (autoplay)

- To start an audio file automatically, use the autoplay attribute
- Add muted after autoplay to let your audio file start playing automatically (but muted)