

ABSOLUTE POSITIONING

Advantages of Dynamic XHTML

- (1) Dynamic XHTML makes documents dynamic. Dynamic documents :
 - o Allow the designer to control how the HTML displays Web pages' content.
 - o React and change with the actions of the visitor.
 - o Can exactly position any element in the window, and change that position after the document has loaded.
 - o Can hide and show content as needed.
- (2) Dynamic XHTML allows any HTML element (any object on the screen that can be controlled independently using JavaScript) to be manipulated at any time, turning plain HTML into dynamic HTML
- (3) With DHTML, changes occur entirely on the client-side
- (4) Using DHTML gives the author more control over how the page is formatted and how content is positioned on the page..

Positioning Elements

Cascading Style Sheets (CSS) Positioning defines the placement of elements on a page and is an extension of cascading style sheets as specified in the W3C on Positioning HTML with CSS. By default, elements flow one after another in the same order as they appear in the HTML source, with each element having a size and position that depends on the type of element, the contents of the element, and the display context for the element as it will render on the page. This default flow model for HTML layout doesn't allow a high level of control over the placement of elements on the page. By applying a small set of CSS attributes to the elements that are defined for the page, CSS can control the precise position of elements by giving exact coordinates. It is also possible to specify placement relative to the position of other objects on the page.

Just like any other HTML or CSS attribute, the CSS attributes used to control an element's position are available for scripting. The position of these elements on the page can thus be dynamically changed with script. As a result, the position of these elements can be recalculated and redrawn after the document is loaded without reloading the page from the server. It usually involves using JavaScript to change a positioning style properties of an HTML elements. `position`, `top`, `left` are the three properties that dictate the position of the elements. `position` specifies the reference point for the placement of the elements. `top` and `left` specify the distance from top and left of reference point where element is to appear. `absolute`, `relative` and `static` are the three possible values for the position property.

Absolute Positioning

A element can be placed at specific position in the document using absolute value for the position styling property. Absolute positioning defines the x and y coordinates of

an element with reference to the top left corner of the browser page or the containing block and the position attribute is set to absolute. With absolute positioning elements are placed without regard to the positions of other elements. For example, if you want place an image 100 pixels from the top and 100 pixels from the left of the document display window, it can be placed as following statements:

```

```

Use of Absolute positioning

- Places elements at specific position in the document display.
- Can be used superimpose text over the ordinary text to create effect similar that watermark on page.

The following example illustrates the usage of absolute positioning to position five elements at specific different positions in the document display.

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//w3c//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<! - - abpos.html - - >
<html xmlns = "http://www.w3.org/1999/xhtml">
  <head>
    <title> Absolute positioning </title>
    <style type = "text/css">
      .s1 {position: absolute; top: 45px; left: 50px; }
      .s2 {position: absolute; top: 45px; left: 300px; }
      .s3 {position: absolute; top: 175px; left: 50px; }
      .s4 {position: absolute; top: 175px; left: 300px; }
      .s5 {position: absolute; top: 100px; left: 175px; }
    </style>
  </head>
  <body>
    <p>Positioning 5 instances same image at 5 different positions
    
    
    
    
    
  </p>
  </body> </html>
```

Output

When an element is absolutely positioned inside another positioned element, the top and left property values are measured from the upper-left corner of the enclosing element.

The following example illustrate the nested element placement. When an element is absolutely positioned inside another positioned element, the top and left property values are measured from the upper-left corner of the enclosing element.

The following example illustrate the nested element placement

Output

Hello. And now it's time to say goodbye

Hello, again

!!!!.

In the above example we insert a `span` element inside the `div` element. Positioning attributes for the `span` element place it 10 pixels in from the left and 30 pixels down from the top of its positioning context—the `div` element in this case.

Source : <http://elearningatria.files.wordpress.com/2013/10/cse-vii-programming-the-web-10cs73-notes.pdf>