

# HTML Frames

HTML frames are used to divide your browser window into multiple sections where each section can load a separate HTML document. A collection of frames in the browser window is known as a frameset. The window is divided into frames in a similar way the tables are organized: into rows and columns.

## Disadvantages of Frames

There are few drawbacks with using frames, so it's never recommended to use frames in your webpages:

- Some smaller devices cannot cope with frames often because their screen is not big enough to be divided up.
- Sometimes your page will be displayed differently on different computers due to different screen resolution.
- The browser's *back button* might not work as the user hopes.
- There are still few browsers that do not support frame technology.

## Creating Frames

To use frames on a page we use `<frameset>` tag instead of `<body>` tag. The `<frameset>` tag defines how to divide the window into frames. The **rows** attribute of `<frameset>` tag defines horizontal frames and **cols** attribute defines vertical frames. Each frame is indicated by `<frame>` tag and it defines which HTML document shall open into the frame.

## Example

Following is the example to create three horizontal frames:

```
<!DOCTYPE html>

<html>
<head>
<title>HTML Frames</title>
</head>
<frameset rows="10%,80%,10%">
    <frame name="top" src="/html/top_frame.htm" />
    <frame name="main" src="/html/main_frame.htm" />
```

```
<frame name="bottom" src="/html/bottom_frame.htm" />
<noframes>
<body>
    Your browser does not support frames.
</body>
</noframes>
</frameset>
</html>
```

This will produce following result:



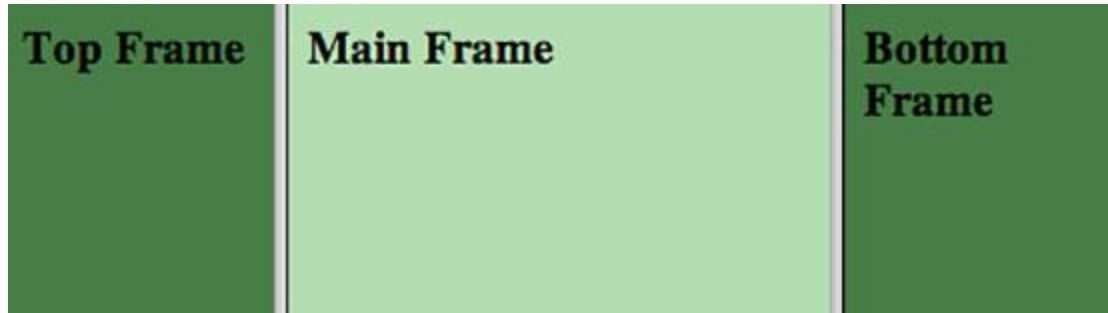
## Example

Let's put above example as follows, here we replaced rows attribute by cols and changed their width. This will create all the three frames vertically:

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Frames</title>
</head>
<frameset cols="25%,50%,25%">
    <frame name="left" src="/html/top_frame.htm" />
    <frame name="center" src="/html/main_frame.htm" />
    <frame name="right" src="/html/bottom_frame.htm" />
<noframes>
<body>
```

```
        Your browser does not support frames.  
  
</body>  
  
</noframes>  
  
</frameset>  
  
</html>
```

This will produce following result:



## The <frameset> Tag Attributes

Following are important attributes of the <frameset> tag:

Attribute	Description
cols	<p>specifies how many columns are contained in the frameset and the size of each column. You can specify the width of each column in one of four ways:</p> <ul style="list-style-type: none"><li>• Absolute values in pixels. For example to create three vertical frames, use <code>cols="100, 500, 100"</code>.</li><li>• A percentage of the browser window. For example to create three vertical frames, use <code>cols="10%, 80%, 10%"</code>.</li><li>• Using a wildcard symbol. For example to create three vertical frames, use <code>cols="10%, *, 10%"</code>. In this case wildcard takes remainder of the window.</li><li>• As relative widths of the browser window. For example to create three vertical frames, use <code>cols="3*, 2*, 1*"</code>. This is an alternative to percentages. You can use relative widths of the browser window.</li></ul>

	Here the window is divided into sixths: the first column takes up half of the window, the second takes one third, and the third takes one sixth.
rows	This attribute works just like the cols attribute and takes the same values, but it is used to specify the rows in the frameset. For example to create two horizontal frames, use <code>rows="10%, 90%"</code> . You can specify the height of each row in the same way as explained above for columns.
border	This attribute specifies the width of the border of each frame in pixels. For example <code>border="5"</code> . A value of zero means no border.
frameborder	This attribute specifies whether a three-dimensional border should be displayed between frames. This attribute takes value either 1 (yes) or 0 (no). For example <code>frameborder="0"</code> specifies no border.
framespacing	This attribute specifies the amount of space between frames in a frameset. This can take any integer value. For example <code>framespacing="10"</code> means there should be 10 pixels spacing between each frames.

## The <frame> Tag Attributes

Following are important attributes of <frame> tag:

Attribute	Description
Src	This attribute is used to give the file name that should be loaded in the frame. Its value can be any URL. For example, <code>src="/html/top_frame.htm"</code> will load an HTML file available in html directory.
name	This attribute allows you to give a name to a frame. It is used to indicate which frame a document should be loaded into. This is especially important when you want to create links in one frame that load pages into an another frame, in which case the second frame needs a name to identify itself as the target of the link.
frameborder	This attribute specifies whether or not the borders of that frame are shown; it overrides the value given in the frameborder attribute on the <frameset> tag if

	one is given, and this can take values either 1 (yes) or 0 (no).
marginwidth	This attribute allows you to specify the width of the space between the left and right of the frame's borders and the frame's content. The value is given in pixels. For example marginwidth="10".
marginheight	This attribute allows you to specify the height of the space between the top and bottom of the frame's borders and its contents. The value is given in pixels. For example marginheight="10".
noresize	By default you can resize any frame by clicking and dragging on the borders of a frame. The noresize attribute prevents a user from being able to resize the frame. For example noresize="noresize".
scrolling	This attribute controls the appearance of the scrollbars that appear on the frame. This takes values either "yes", "no" or "auto". For example scrolling="no" means it should not have scroll bars.
longdesc	This attribute allows you to provide a link to another page containing a long description of the contents of the frame. For example longdesc="framedescription.htm"

## Browser Support for Frames

If a user is using any old browser or any browser which does not support frames then `<noframes>` element should be displayed to the user.

So you must place a `<body>` element inside the `<noframes>` element because the `<frameset>` element is supposed to replace the `<body>` element, but if a browser does not understand `<frameset>` element then it should understand what is inside the `<body>` element which is contained in a `<noframes>` element.

You can put some nice message for your user having old browsers. For example *Sorry!! your browser does not support frames.* as shown in the above example.

# Frame's name and target attributes

One of the most popular uses of frames is to place navigation bars in one frame and then load main pages into a separate frame.

Let's see following example where a test.htm file has following code:

```
<!DOCTYPE html>

<html>

<head>

<title>HTML Target Frames</title>

</head>

<frameset cols="200, *">

    <frame src="/html/menu.htm" name="menu_page" />

    <frame src="/html/main.htm" name="main_page" />

</frameset>

<body>

    Your browser does not support frames.

</body>

</frameset>

</html>
```

Here we have created two columns to fill with two frames. The first frame is 200 pixels wide and will contain the navigation menubar implemented by **menu.htm** file. The second column fills in remaining space and will contain the main part of the page and it is implemented by **main.htm** file. For all the three links available in menubar, we have mentioned target frame as **main\_page**, so whenever you click any of the links in menubar, available link will open in main\_page.

Following is the content of menu.htm file

```
<!DOCTYPE html>

<html>

<body bgcolor="#4a7d49">

<a href="http://www.google.com" target="main_page">Google</a>

<br /><br />
```

```

<a href="http://www.microsoft.com" target="main_page">Microsoft</a>
<br /><br />
<a href="http://news.bbc.co.uk" target="main_page">BBC News</a>
</body>
</html>

```

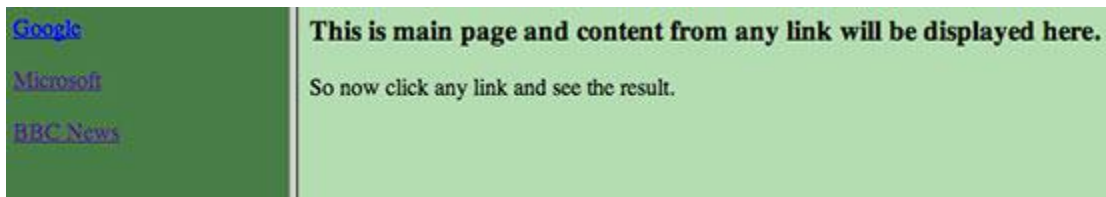
Following is the content of main.htm file:

```

<!DOCTYPE html>
<html>
<body bgcolor="#b5dcb3">
<h3>This is main page and content from any link will be displayed here.</h3>
<p>So now click any link and see the result.</p>
</body>
</html>

```

When we load **test.htm** file, it produces following result:



Now you can try to click links available in the left panel and see the result. The *target* attribute can also take one of the following values:

Option	Description
_self	Loads the page into the current frame.
_blank	Loads a page into a new browser window.opening a new window.
_parent	Loads the page into the parent window, which in the case of a single frameset is the main browser window.

_top	Loads the page into the browser window, replacing any current frames.
targetframe	Loads the page into a named targetframe.



HTML lets you specify metadata - additional important information about a document in a variety of ways. The META elements can be used to include name/value pairs describing properties of the HTML document, such as author, expiry date, a list of keywords, document author etc.

The **<meta>** tag is used to provide such additional information. This tag is an empty element and so does not have a closing tag but it carries information within its attributes.

You can include one or more meta tags in your document based on what information you want to keep in your document but in general, meta tags do not impact physical appearance of the document so from appearance point of view, it does not matter if you include them or not.

## Adding Meta Tags to Your Documents

You can add metadata to your web pages by placing <meta> tags inside the header of the document which is represented by **<head>** and **</head>** tags. A meta tag can have following attributes in addition to core attributes:

Attribute	Description
Name	Name for the property. Can be anything. Examples include, keywords, description, author, revised, generator etc.
content	Specifies the property's value.
scheme	Specifies a scheme to interpret the property's value (as declared in the content attribute).
http-equiv	Used for http response message headers. For example http-equiv can be used to refresh the page or to set a cookie. Values include content-type, expires, refresh and set-cookie.

## Specifying Keywords

You can use <meta> tag to specify important keywords related to the document and later these keywords are used by the search engines while indexing your webpage for searching purpose.

## Example

Following is an example where we are adding HTML, Meta Tags, Metadata as important keywords about the document.

```
<!DOCTYPE html>

<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

This will produce following result:

Hello HTML5!

## Document Description

You can use <meta> tag to give a short description about the document. This again can be used by various search engines while indexing your webpage for searching purpose.

## Example

```
<!DOCTYPE html>

<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

# Document Revision Date

You can use <meta> tag to give information about when last time the document was updated. This information can be used by various web browsers while refreshing your webpage.

## Example

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="revised" content="Tutorialspoint, 3/7/2014" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

# Document Refreshing

A <meta> tag can be used to specify a duration after which your web page will keep refreshing automatically.

## Example

If you want your page keep refreshing after every 5 seconds then use the following syntax.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="revised" content="Tutorialspoint, 3/7/2014" />
<meta http-equiv="refresh" content="5" />
</head>
```

```
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

## Page Redirection

You can use `<meta>` tag to redirect your page to any other webpage. You can also specify a duration if you want to redirect the page after a certain number of seconds.

### Example

Following is an example of redirecting current page to another page after 5 seconds. If you want to redirect page immediately then do not specify *content* attribute.

```
<!DOCTYPE html>
<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta name="revised" content="Tutorialspoint, 3/7/2014" />
<meta http-equiv="refresh" content="5" url=http://www.tutorialspoint.com" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

## Setting Cookies

Cookies are data, stored in small text files on your computer and it is exchanged between web browser and web server to keep track of various information based on your web application need.

You can use `<meta>` tag to store cookies on client side and later this information can be used by the Web Server to track a site visitor.

## Example

Following is an example of redirecting current page to another page after 5 seconds. If you want to redirect page immediately then do not specify *content* attribute.

```
<!DOCTYPE html>

<html>

<head>

<title>Meta Tags Example</title>

<meta name="keywords" content="HTML, Meta Tags, Metadata" />

<meta name="description" content="Learning about Meta Tags." />

<meta name="revised" content="Tutorialspoint, 3/7/2014" />

<meta http-equiv="cookie" content="userid=xyz; expires=Wednesday, 08-Aug-15 23:59:59 GMT;" />

</head>

<body>

<p>Hello HTML5!</p>

</body>

</html>
```

If you do not include the expiration date and time, the cookie is considered a session cookie and will be deleted when the user exits the browser.

**Note:** You can check [PHP and Cookies](#) tutorial for a complete detail on Cookies.

## Setting Author Name

You can set an author name in a web page using meta tag. See an example below:

### Example

```
<!DOCTYPE html>

<html>

<head>

<title>Meta Tags Example</title>

<meta name="keywords" content="HTML, Meta Tags, Metadata" />

<meta name="description" content="Learning about Meta Tags." />

<meta ame="author" content="Neelam " />

</head>
```

```
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

## Specify Character Set

You can use <meta> tag to specify character set used within the webpage.

### Example

By default, Web servers and Web browsers use ISO-8859-1 (Latin1) encoding to process Web pages. Following is an example to set UTF-8 encoding:

```
<!DOCTYPE html>

<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
<meta ame="author" content="Neelam " />
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```

To serve the static page with traditional Chinese characters, the webpage must contain a <meta> tag to set Big5 encoding:

```
<!DOCTYPE html>

<html>
<head>
<title>Meta Tags Example</title>
<meta name="keywords" content="HTML, Meta Tags, Metadata" />
<meta name="description" content="Learning about Meta Tags." />
```

```
<meta ame="author" content="Neelam" />
<meta http-equiv="Content-Type" content="text/html; charset=Big5" />
</head>
<body>
<p>Hello HTML5!</p>
</body>
</html>
```