



Web Design and Programming

Lecture 6:

FRAMES

*Assoc. Prof.
Ali A. Al-Ani*

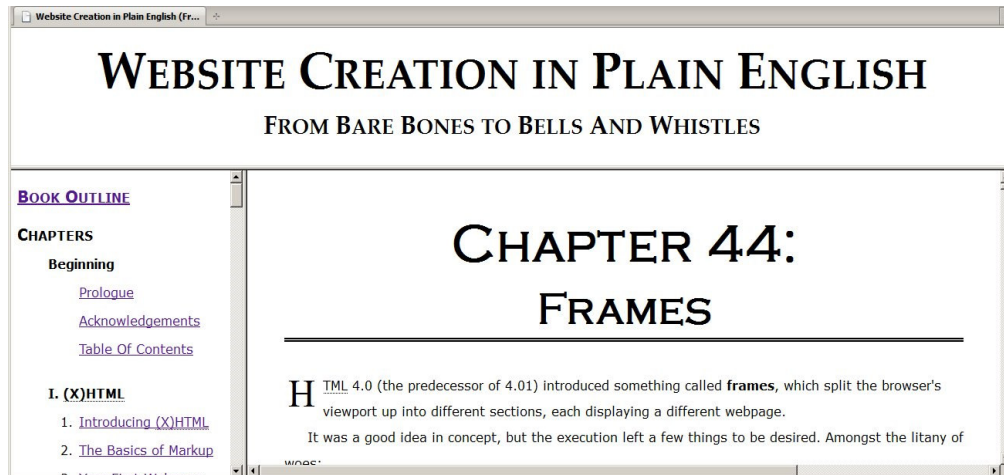


Objectives

- How to create a *frameset* document with multiple *frames*.
- How to deal with users whose browsers cannot use frames.
- How to create inline frames (*iframes*), which are single windows that sit within another page.



Objectives



XHTML: Frames

- **HTML frames** are used to divide our 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.
- One of the key advantages that frames offer is *that we can then load and reload single window without having to reload the entire contents of the browser window*.
- The simplest of framesets might just divide the screen into two rows, whereas a complex frameset could use several rows and columns.
- A collection of frames in the browser window is known as a **Frameset**.



XHTML: Frames

- It is quite rare to see frames in use these days. but in fact there are few circumstances that encourage us to use frames in a page such as.
 1. *When we want to display a lot of content in one single page and you cannot split the document into separate pages.*
 2. *When we have a lot of data in one part of the page that we do not want the user to have to reload while another part of the page changes.*



XHTML: Frames

- A couple of other drawbacks we should be aware of with frames are as follows:
 1. *Some browsers do not print well from framesets (print one frame at a time).*
 2. *The browser's Back button might not work as the user expects.*
 3. *Some smaller devices cannot cope with frames often because their screen is not big enough to be divided up.*
 4. *Sometimes the page will be displayed differently on different computers due to different screen resolution.*
 5. *There are still few browsers that do not support frame technology.*



XHTML: Frameset Element

- To use frames on a page we use `<frameset>` element instead of `<body>`. The `<frameset>` defines, how to divide the window into frames. . *The rows attribute of <frameset> defines horizontal frames and cols attribute defines vertical frames* and that specify how the browser window will be divided up into rows and columns:
 1. *cols specifies how many columns are in the frameset.*
 2. *rows specifies how many rows are in the frameset.*
- The `<frameset>` element contains a `<frame>` element for each frame of the document and a `<noframes>` element to indicate what should be displayed if the *user's browser does not load frames.*



XHTML: Frameset Element

- In addition to the *rows and cols attributes*, the frameset element can also take the following attributes:

Class, id, onload, onunload, rows, style, title, Onblur, onfocus, border, bordercolor, frameborder, framespacing

 1. *The cols Attribute:* The *cols attribute* specifies how many columns are contained in the frameset and the size of each column. We have to provide a value to indicate the width for each of the columns in our frameset, and the number of values we provide (*each separated by a comma*) indicates how many columns there are in the document.



XHTML: Frameset Element

- **Ex: cols="20%, 60%, 20%"**, here there are three columns: the first takes up 20 percent of the width of the browser window, the second takes up 60 percent, and the third takes the last 20 percent. Because there are three values, the browser knows that there should be three columns. We can specify the width of each column in one of four ways:
 1. **Absolute values in pixels.**
 2. **A percentage of the browser window (or parent frame if you have one frameset sit inside another , which is known as a nested frame).**
 3. **Using a wildcard symbol (*).**



XHTML: Frameset Element

2. **The rows Attribute:** The *rows attribute* works just like the *cols attribute* and can take the same values, but it is used to specify the rows in the frameset. **Ex: rows="100, 80%, *"**.
3. **The border Attribute:** The border attribute specifies the width of the border of each frame in pixels. It was introduced in Netscape 3 and IE 4. For example **border="10"**. If we do not want a border, you can give this attribute a value of **"0"**.
4. **The framespacing Attribute:** The *framespacing attribute* specifies the amount of space between frames in a frameset. The value should be given in pixels and the default value is **2** if not otherwise specified. For example **framespacing="25"**.



XHTML: Frame Element

- The `<frame>` *element* indicates what goes in each frame of the frameset. This *element should always carry one attribute, src*, to indicate the page that should represent that frame. The `<frame>` element can carry any of the universal attributes, and the following attributes:

Frameborder, marginwidth, marginheight, noresize, scrolling, longdesc, src, name

```
<frameset cols="200, *">
```

```
<frame src="frames/linksNav.html" />
```

```
<frame src="frames/linksMain.html" name="main_page" />
```

```
</frameset>
```



XHTML: Frame Element

- 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, *src = "/html/top_frame.htm"*.
- 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 we want to create links in one frame that load pages into an another frame.*
- scrolling:** This attribute controls the appearance of the scrollbars that appear on the frame. This takes values either "yes", "no" or "auto".
- 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"*



XHTML: Noframe Element

- If a user's browser does not support frames (which is very rare these days), the contents of the `<noframes>` element should be displayed to the user.
- In XHTML, we 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 the `<frameset>` element, it should ignore these elements and the `<noframes>` element and understand what is inside the `<body>` element contained in the `<noframes>` element.



Frames: Creating Links Between Frames

- One of the most popular uses of frames is to place navigation bars in one frame and then load the pages with the content into a separate frame. This is particularly helpful in three situations:
 1. *When our navigation bar is rather large in size (such as thumbnails of photographs in a gallery). By using frames, the user does not need to reload the navigation bar each time he or she views a new page.*
 2. *When our main document is very long and the navigation bar provides shortcuts to parts of the main document (acting like a table of contents that is always in view).*
 3. *When we do not want to reload the whole page.*



Frames: Creating Links Between Frames

- Each `<frame>` element can carry the name attribute to give each frame a name. This name is used in the links to indicate which frame the new page should load into. For example:

```
<frameset cols="200, *">  
<frame src="frames/linksNav.html" />  
<frame src="frames/linksMain.html" name="main_page" />  
</frameset>
```

- The links on the left-side navigation bar will load pages into the right-side main page. The links in the linksNav.html file look like this:

```
<a href="http://www.yahoo.com" target="main_page">Wrox Press</a><br /><br />
```



Frames: Nested Framesets

- We create a nested frameset by using a new `<frameset>` element in the place of one of the `<frame>` elements. Take a look at the following example:

```
<frameset rows=" 300, *">  
<frame src="frames/top_frame.html" />  
  <frameset cols=" 400, *">  
    <frame src="frames/blank.html" />  
    <frame src="frames/main_frame.html" />  
  </frameset>  
<frame src="frames/bottom_frame.html" />  
</frameset>
```

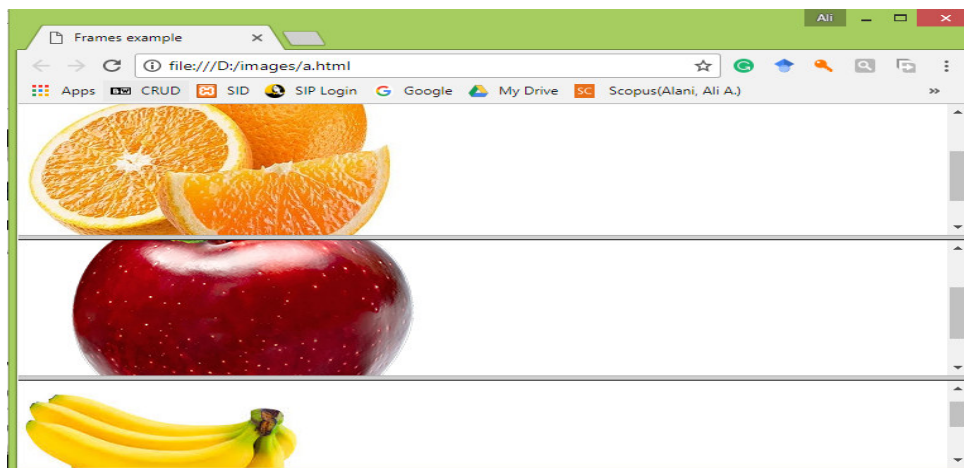



XHTML: Frames

```
<html>
<head> <title>Frames example</title> </head>
<frameset rows="150, *, 100">
<frame src="images/orange.jpg" />
<frame src="main_frame.html" />
<frame src="bottom_frame.html" />
<noframes> <body>
<p>This site uses a technology called frames. Unfortunately, your browser does not support this
technology. Please upgrade your browser and visit us again. </p> </body> </noframes>
</frameset> </html>
```



XHTML: Frames





XHTML: iFrames

- Another special kind of frame is commonly known as an *iframe* (although it is sometimes referred to as an *inline frame* or *floating frames* because it sits in the middle of a normal XHTML page); it does not need to appear either in a `<frameset>` element.
- The *inline frame* is created using the `<iframe>` element, and, rather like an image, the inline frame can have text flowing around it and you can set borders and margins around the *inline frame*.
- The core attribute it has to carry is the *src* attribute, whose value is the URL of the page to be included (wherever the `<iframe>` element is in the document), although it is also good to add the *height* and *width* attributes to control its size.



XHTML: iFrames

- In addition to the universal attributes, the `<iframe>` element can carry these attributes:

*Align, height, width, frameborder, longdesc, marginwidth,
marginheight, name, scrolling, src*

```
<body>
```

```
<h1>Inline frame</h1>
```

```
<iframe src="frames/iframe.html">
```

Error! You should be seeing iframe window. This site uses a technology called frames which is not supported by older browsers. If you are using a version of Internet Explorer older than version 3 or a version of Netscape older than version 6 you might need to upgrade your browser.

```
</iframe>
```

```
</body>
```



XHTML: iFrames

- In this example we create a simple page for children to learn about fruit. The page allows the child to click a link and load the corresponding image into the inline frame without the rest of the page changing.

1. **Create the skeleton of a standard Transitional XHTML document, as follows:**

```
<html>
<head>
<title>iframe example</title>
</head>
<body>

</body>
</html>
```



XHTML: iFrames

2. **Add a heading and then the <iframe> element, which must have a name attribute. In this case we should use the scrolling attribute to prevent the frame from having scrollbars. We should also set a size for the frame let's make it 150 pixels square:**

```
<h1>Learn your fruit</h1>

<iframe name="iframe" height="150" width="150" scrolling="no">

Pictures of fruit will appear here

</iframe>
```



XHTML: iFrames

3. Add the links that will load the images into the inline frame. As with the other type of frames, if we want the links to load the page into another frame, the links must carry the target attribute whose value is the name of the iframe.

`<p>Click on the name of the fruit to see an image of it appear.</p>`

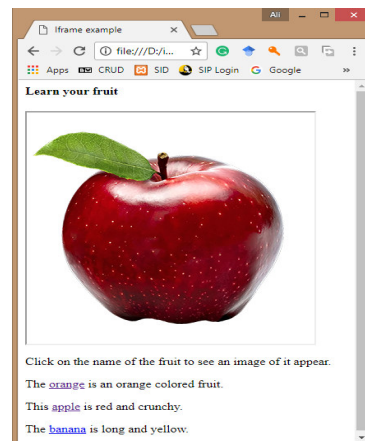
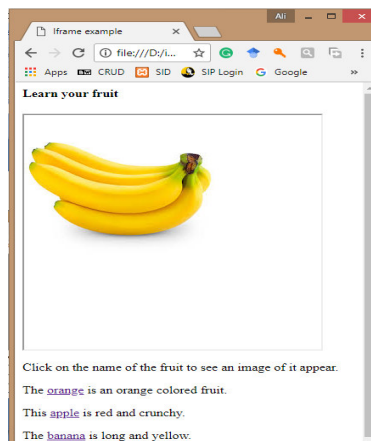
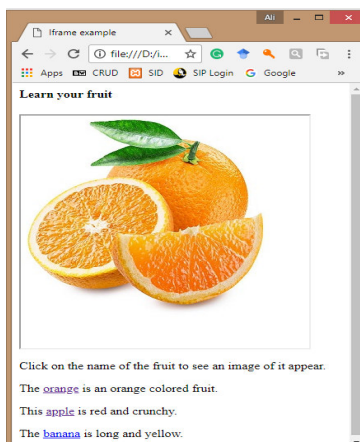
`<p>The orange is an orange colored fruit.</p>`

`<p>This apple is red and crunchy.</p>`

`<p>The banana is long and yellow.</p>`



XHTML: iFrames





The End