CHAPTER 8
BACKGROUND, FRAME AND LAYOUT

Learning Objectives:
After reading this INFORMATION SHEET, YOU MUST be able to:

- Create a good layout to give better look to your website.;
- Set background within a web page;
- Display a web pages in web page using FRAME;

This chapter demonstrates about the HTML Background, Frame and Layout which are used to give the better looking for the web pages

CHAPTER 8.1: HTML Background

By default, your webpage background is white in color. You may not like it, but no worries. HTML provides you following two good ways to decorate your webpage background.

- HTML Background with Colors
- HTML Background with Images

Now let's see both the approaches one by one using appropriate examples.

Html Background with Colors

The **bgcolor** attribute is used to control the background of an HTML element, specifically page body and table backgrounds.

Note − The *bgcolor* attribute deprecated in HTML5. Do not use this attribute.

Following is the syntax to use bgcolor attribute with any HTML tag.

```
<tagname bgcolor = "color_value"...>
```

This color_value can be given in any of the following formats −

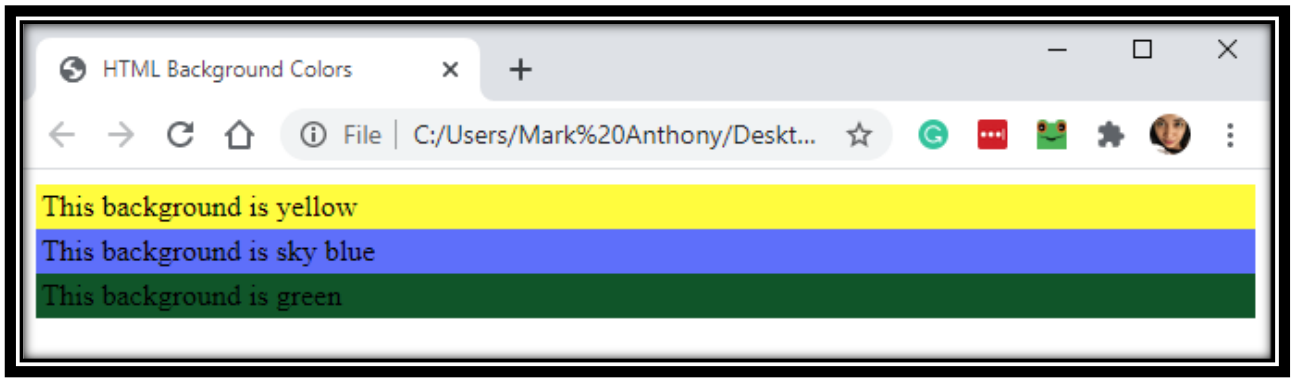
```
<!-- Format 1 - Use color name -->
<table bgcolor = "lime" >

<!-- Format 2 - Use hex value -->
<table bgcolor = "#f1f1f1" >

<!-- Format 3 - Use color value in RGB terms -->
<table bgcolor = "rgb(0,0,120)" >
```

Example

Here are the examples to set background of an HTML tag −

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Background Colors</title>
   </head>

   <body>
      <!-- Format 1 - Use color name -->
      <table bgcolor = "yellow" width = "100%">
         <tr>
            <td>
               This background is yellow
            </td>
         </tr>
      </table>

      <!-- Format 2 - Use hex value -->
      <table bgcolor = "#6666FF" width = "100%">
         <tr>
            <td>
               This background is sky blue
            </td>
         </tr>
      </table>

      <!-- Format 3 - Use color value in RGB terms -->
      <table bgcolor = "rgb(255,0,255)" width = "100%">
         <tr>
            <td>
               This background is green
            </td>
         </tr>
      </table>
   </body>

</html>
```

This will produce the following result –



Html Background with Images

The **background** attribute can also be used to control the background of an HTML element, specifically page body and table backgrounds. You can specify an image to set background of your HTML page or table.

Note − The *background* attribute deprecated in HTML5. Do not use this attribute.

Following is the syntax to use background attribute with any HTML tag.

Note − The *background* attribute is deprecated and it is recommended to use Style Sheet for background setting.

```
<tagname background = "Image URL"...>
```

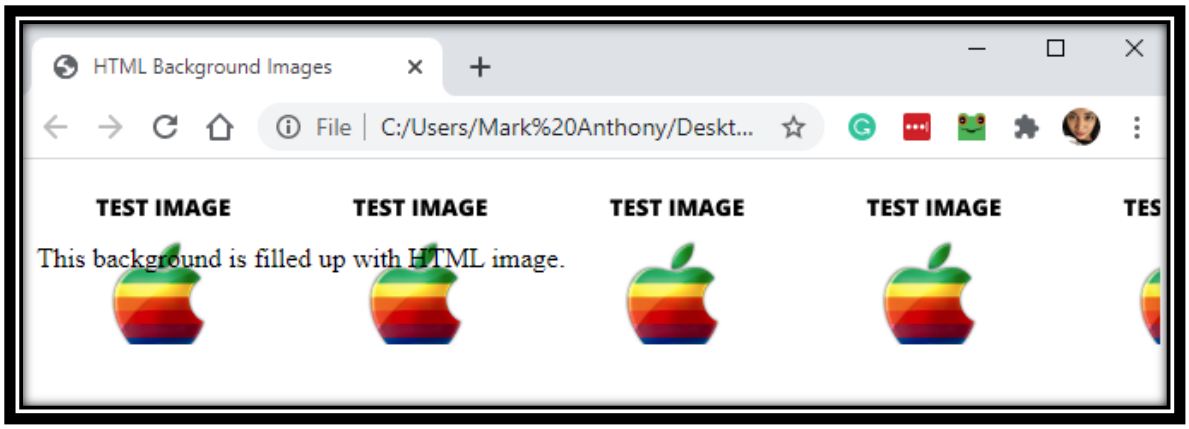
The most frequently used image formats are JPEG, GIF and PNG images.

Example

Here are the examples to set background images of a table.

```
<!DOCTYPE html>
<html>
   <head>
      <title>HTML Background Images</title>
   </head>

   <body>
      <!-- Set table background -->
      <table background = "/images/html.gif" width = "100%" height = "100">
         <tr><td>
            This background is filled up with HTML image.
         </td></tr>
      </table>
   </body>
</html>
```

This will produce the following result –



Patterned & Transparent Backgrounds

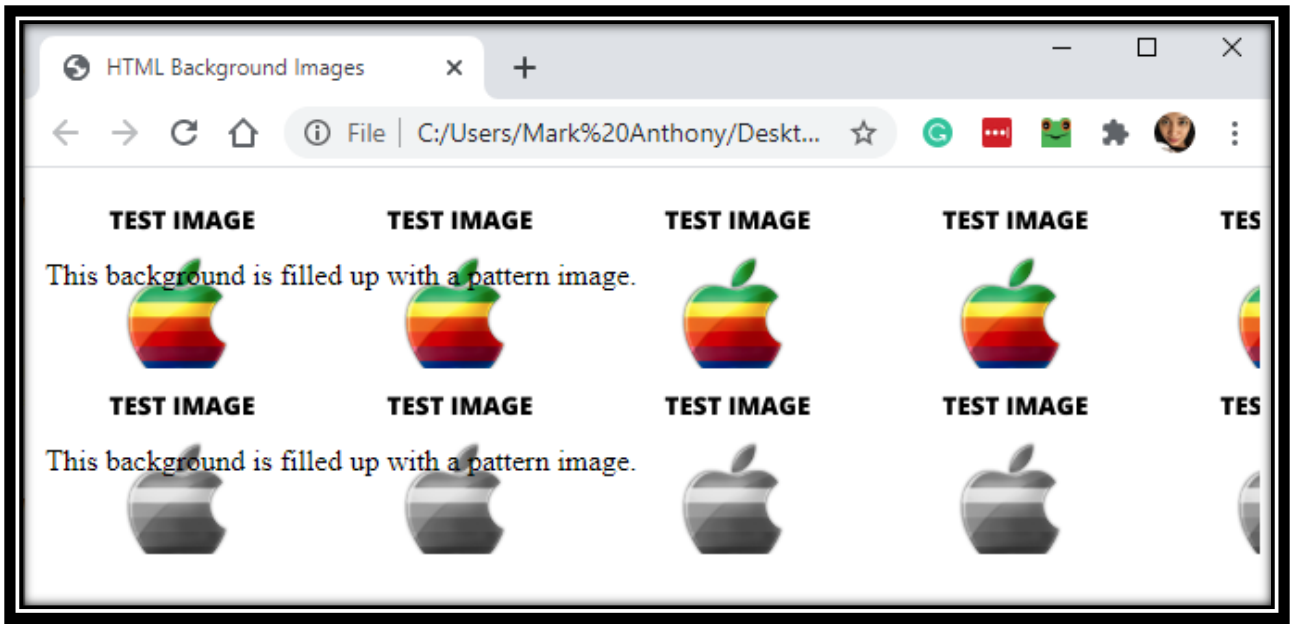
You might have seen many pattern or transparent backgrounds on various websites. This simply can be achieved by using patterned image or transparent image in the background.

It is suggested that while creating patterns or transparent GIF or PNG images, use the smallest dimensions possible even as small as 1x1 to avoid slow loading.

Example

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Background Images</title>
   </head>

   <body>
      <!-- Set a table background using pattern -->
      <table background = "/images/pattern1.gif" width = "100%" height = "100">
         <tr>
            <td>
               This background is filled up with a pattern image.
            </td>
         </tr>
      </table>

      <!-- Another example on table background using pattern -->
      <table background = "/images/pattern2.gif" width = "100%" height = "100">
         <tr>
            <td>
               This background is filled up with a pattern image.
            </td>
         </tr>
      </table>
   </body>

</html>
```

This will produce the following result –



CHAPTER 8.2: Layout

A webpage layout is very important to give better look to your website. It takes considerable time to design a website's layout with great look and feel.

Now-a-days, all modern websites are using CSS and JavaScript based framework to come up with responsive and dynamic websites but you can create a good layout using simple HTML tables or division tags in combination with other formatting tags. This chapter will give you few examples on how to create a simple but working layout for your webpage using pure HTML and its attributes.

HTML Layout - Using Tables

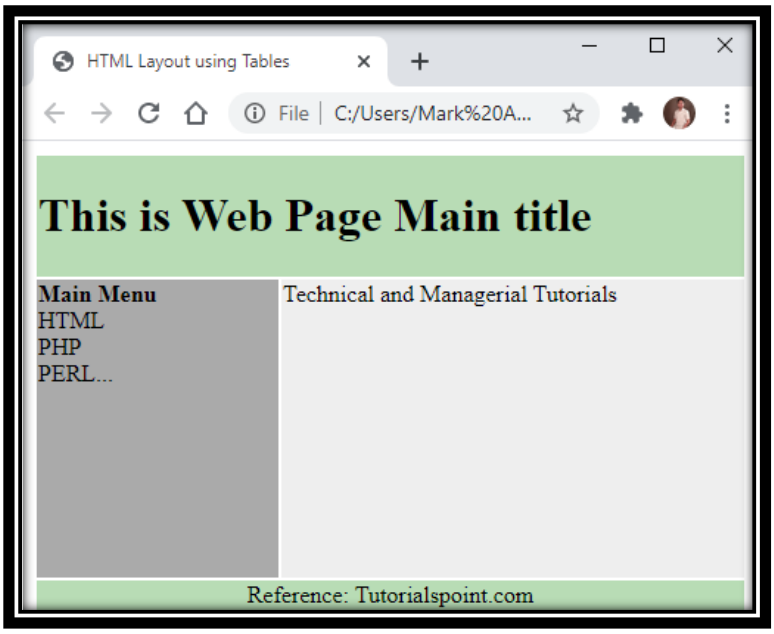
The simplest and most popular way of creating layouts is using HTML <table> tag. These tables are arranged in columns and rows, so you can utilize these rows and columns in whatever way you like.

Example

For example, the following HTML layout example is achieved using a table with 3 rows and 2 columns but the header and footer column spans both columns using the colspan attribute –

```
<!DOCTYPE html>
<html>
   <head>
      <title>HTML Layout using Tables</title>
   </head>
<body>
      <table width = "100%" border = "0">
         <tr>
            <td colspan = "2" bgcolor = "#b5dcb3">
               <h1>This is Web Page Main title</h1>
            </td>
         </tr>
         <tr valign = "top">
            <td bgcolor = "#aaa" width = "50">
               <b>Main Menu</b><br />
               HTML<br />
               PHP<br />
               PERL...
            </td>

            <td bgcolor = "#eee" width = "100" height = "200">
               Technical and Managerial Tutorials
            </td>
         </tr>
         <tr>
            <td colspan = "2" bgcolor = "#b5dcb3">
               <center>
                  Reference: Tutorialspoint.com
               </center>
            </td>
         </tr>
      </table>
</body>
</html>
```

This will produce the following result –



Multiple Columns Layout - Using Tables

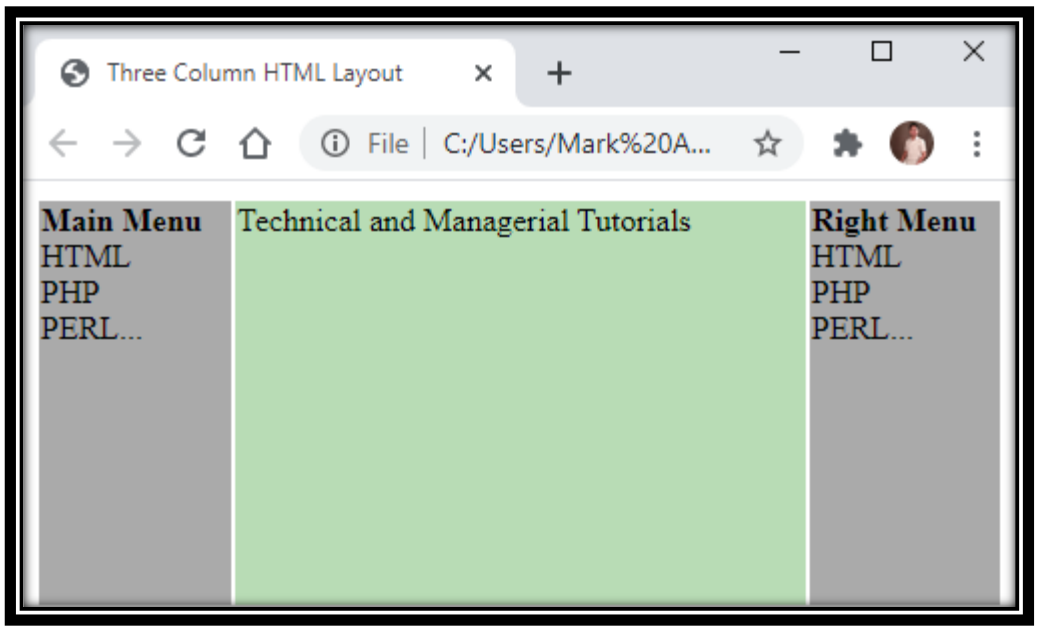
You can design your webpage to put your web content in multiple pages. You can keep your content in middle column and you can use left column to use menu and right column can be used to put advertisement or some other stuff. This layout will be very similar to what we have at our website tutorialspoint.com.

Example

Here is an example to create three column layout −

```
<!DOCTYPE html>
<html>

   <head>
      <title>Three Column HTML Layout</title>
   </head>

   <body>
      <table width = "100%" border = "0">

         <tr valign = "top">
            <td bgcolor = "#aaa" width = "20%">
               <b>Main Menu</b><br />
               HTML<br />
               PHP<br />
               PERL...
            </td>

            <td bgcolor = "#b5dcb3" height = "200" width = "60%">
               Technical and Managerial Tutorials
            </td>

            <td bgcolor = "#aaa" width = "20%">
               <b>Right Menu</b><br />
               HTML<br />
               PHP<br />
               PERL...
            </td>
         </tr>

      <table>
   </body>

</html>
```

This will produce the following result –



HTML Layouts - Using DIV, SPAN

The element is a block level element used for grouping HTML elements. While the tag is a block-level element, the HTML element is used for grouping elements at an inline level.

Although we can achieve pretty nice layouts with HTML tables, but tables weren't really designed as a layout tool. Tables are more suited to presenting tabular data.

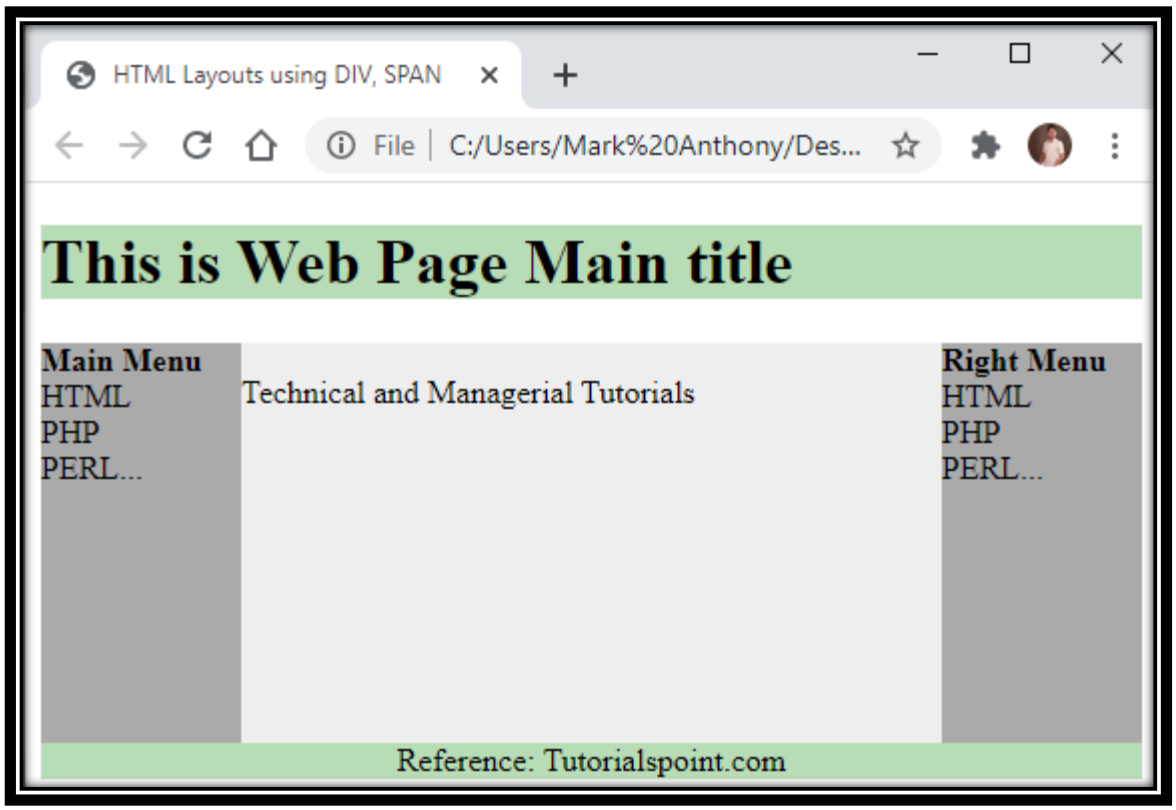
Note − This example makes use of Cascading Style Sheet (CSS), so before understanding this example you need to have a better understanding on how CSS works.

Example

Here we will try to achieve same result using tag along with CSS, whatever you have achieved using <table> tag in previous example.

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Layouts using DIV, SPAN</title>
   </head>

   <body>
      <div style = "width:100%">

         <div style = "background-color:#b5dcb3; width:100%">
            <h1>This is Web Page Main title</h1>
         </div>

         <div style = "background-color:#aaa; height:200px;
width:100px; float:left;">
            <div><b>Main Menu</b></div>
            HTML<br />
            PHP<br />
            PERL...
         </div>

         <div style = "background-color:#eee; height:200px;
width:350px; float:left;" >
            <p>Technical and Managerial Tutorials</p>
         </div>

         <div style = "background-color:#aaa; height:200px;
width:100px; float:right;">
            <div><b>Right Menu</b></div>
            HTML<br />
            PHP<br />
            PERL...
         </div>

         <div style = "background-color:#b5dcb3; clear:both">
            <center>
               Reference: Tutorialspoint.com
            </center>
         </div>

      </div>
   </body>

</html>
```

This will produce the following result −



CHAPTER 8.3: 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.

Note − The <frame> tag deprecated in HTML5. Do not use this element.

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 = "top_frame.htm" />
      <frame name = "main" src = "main_frame.htm" />
      <frame name = "bottom" src = "bottom_frame.htm" />

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

   </frameset>

</html>
```

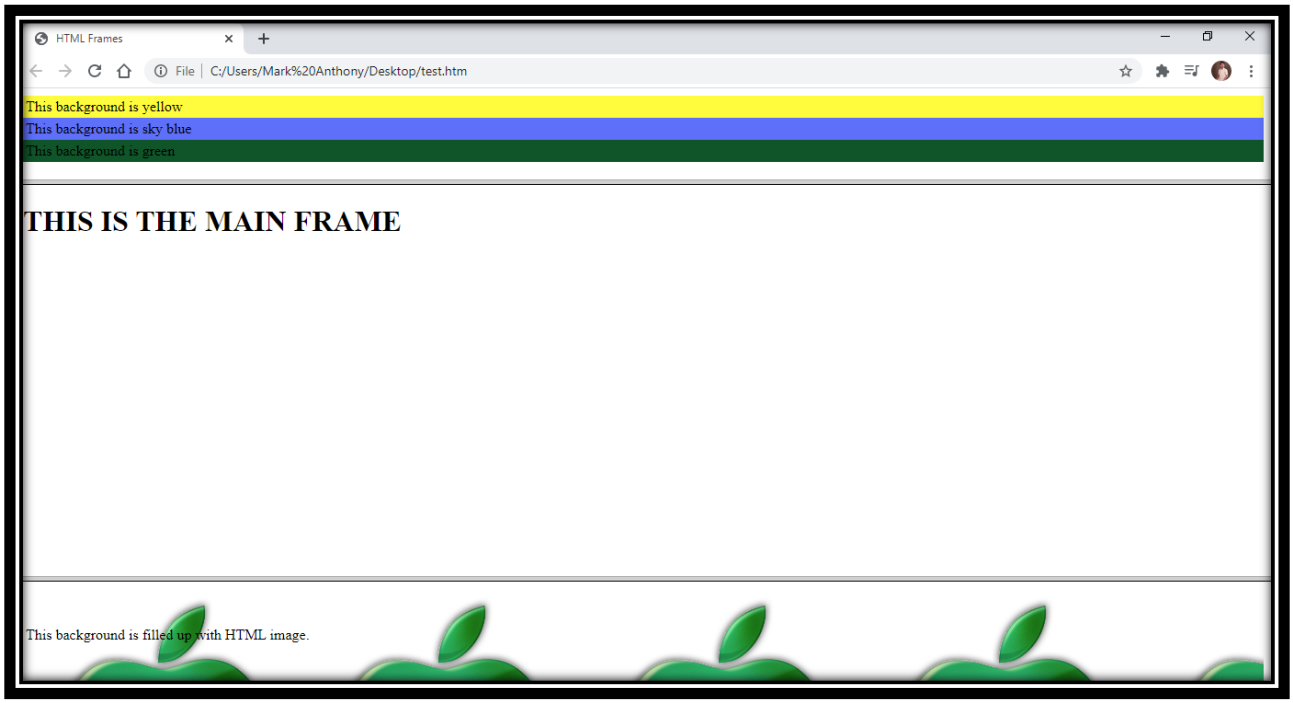
In this example, there are 3 other html file in the same directory, named as; top_frame.htm, main_frame.htm and bottom_frame.htm.

This is the bottom_frame.htm

top_frame.htm look like this

And this is the main_frame.htm

This will produce the following result –



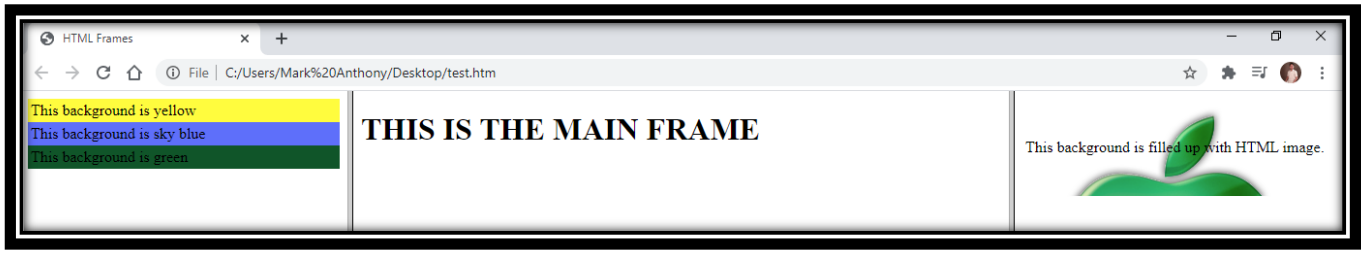
Example

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

With the same top frame, main frame and bottom frame as above

```
<!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 the following result –



The <frameset> Tag Attributes

Following are important attributes of the <frameset> tag −

Number	Attribute & Description
1	**cols** 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 the four ways − Absolute values in pixels. For example, to create three vertical frames, use *cols = "100, 500, 100"*. A percentage of the browser window. For example, to create three vertical frames, use *cols = "10%, 80%, 10%"*. Using a wildcard symbol. For example, to create three vertical frames, use *cols = "10%, *, 10%"*. In this case wildcard takes remainder of the window. As relative widths of the browser window. For example, to create three vertical

	frames, use *cols = "3*, 2*, 1*"*. This is an alternative to percentages. You can use relative widths of the browser window. 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.
2	**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 *rows = "10%, 90%"*. You can specify the height of each row in the same way as explained above for columns.
3	**border** This attribute specifies the width of the border of each frame in pixels. For example, border = "5". A value of zero means no border.
4	**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 frameborder = "0" specifies no border.
5	**framespacing** This attribute specifies the amount of space between frames in a frameset. This can take any integer value. For example framespacing = "10" means there should be 10 pixels spacing between each frames.

The <frame> Tag Attributes

Following are the important attributes of <frame> tag −

Sr.No	Attribute & Description
1	**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" will load an HTML file available in html directory.

2	**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.
3	**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).
4	**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".
5	**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".
6	**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".
7	**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.
8	**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" />

      <noframes>
         <body>Your browser does not support frames.</body>
      </noframes>
   </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 menu bar 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 menu bar, we have mentioned target frame as **main_page**, so whenever you click any of the links in menu bar, 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 *targetattribute* can also take one of the following values −

Number	Option & Description
1	**_self** Loads the page into the current frame.
2	**_blank** Loads a page into a new browser window. Opening a new window.
3	**_parent** Loads the page into the parent window, which in the case of a single frameset is the main browser window.
4	**_top** Loads the page into the browser window, replacing any current frames.
5	**targetframe** Loads the page into a named targetframe.

VIDEO TUTORIAL LINK

HTML – Background

https://www.youtube.com/watch?v=QQj-ULba4_0

HTML - Frame

https://www.youtube.com/watch?v=HYQJyTdph5E&list=PLWPirh4EWFpFI-w62QiV62FOVHcXnynon&index=13

HTML - Layout

https://www.youtube.com/watch?v=gPJN_RPVJgw

REFERENCES:

- https://www.tutorialspoint.com/html/html_backgrounds.htm
- https://www.tutorialspoint.com/html/html_frames.htm
- https://www.tutorialspoint.com/html/html_layouts.htm