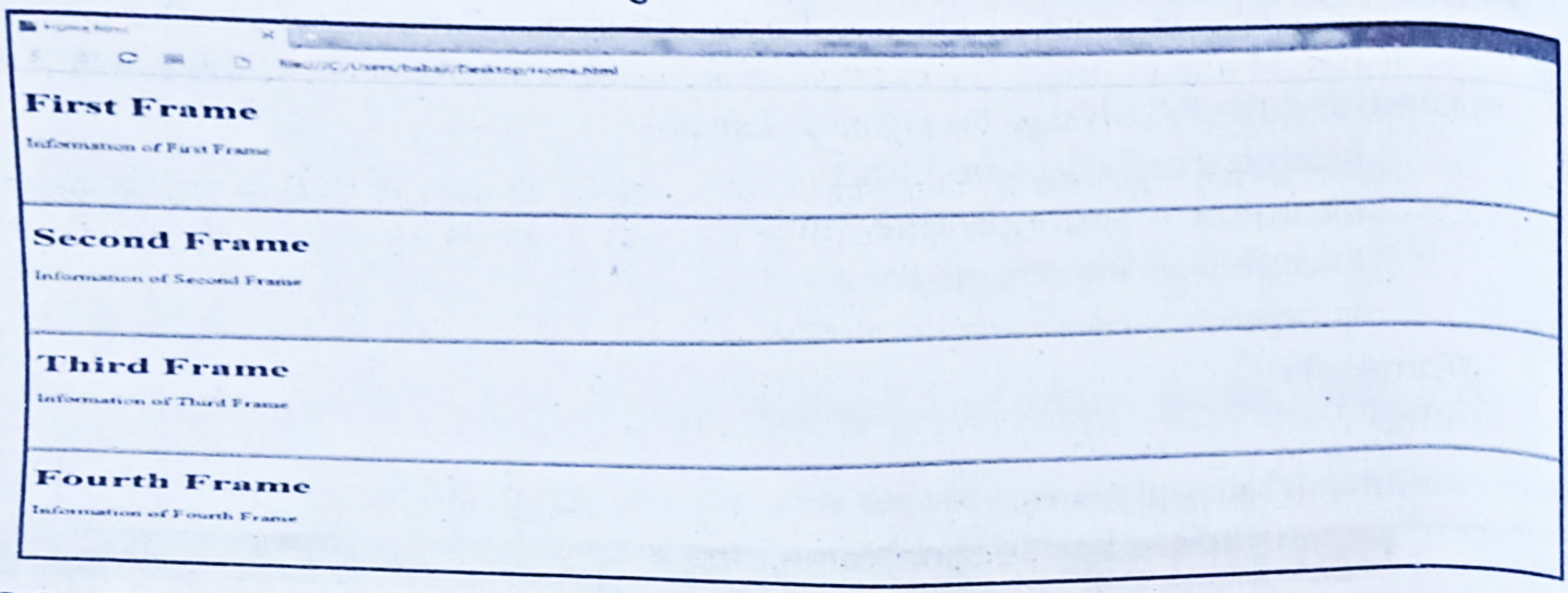
And here's how that HTML will render. -
और यहां बताया गया है कि एचटीएमएल कैसे प्रस्तुत करेगा



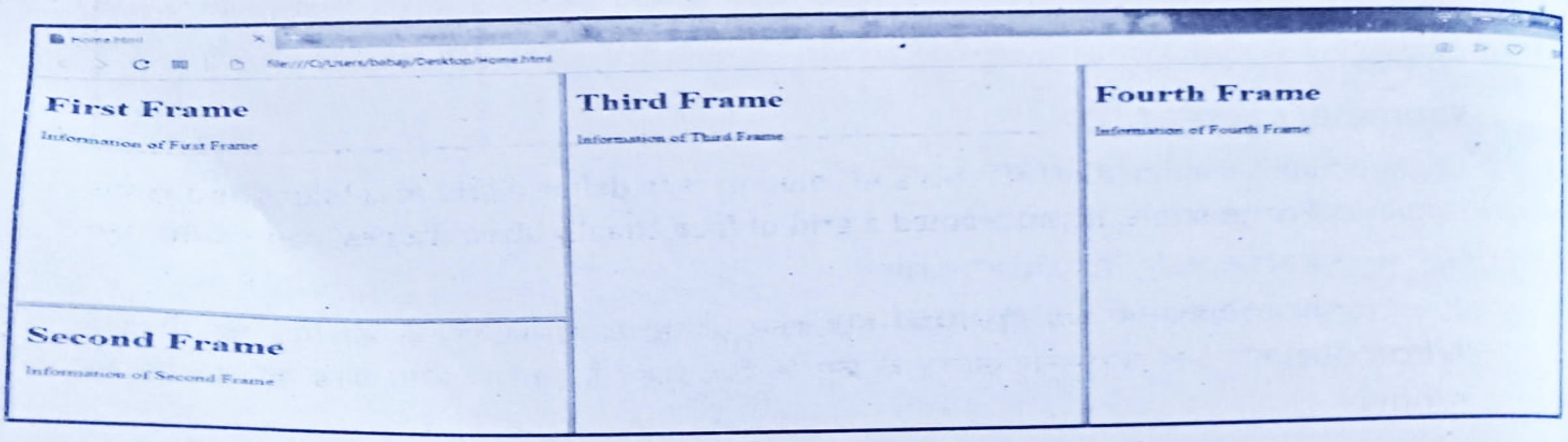
3.8.3 Mixing Columns and Rows - कॉलम और रो का मिश्रण

Columns and rows of frames can both appear on the same webpage by nesting one frameset inside of another. To do this, we first create a frameset and then nest a child frameset within the parent element. Here's an example of how we could nest two rows within a set of three columns.

फ्रेम के कॉलम और रो दोनों एक ही देबपेज पर एक फ्रेमसेट को दूसरे के अंदर nesting से प्रकट कर सकते हैं। ऐसा करने के लिए, हम पहले एक फ्रेमसेट बनाते हैं और फिर पैरेंट (मूल) ऐलिमेंट के भीतर एक चाइल्ड फ्रेमसेट बनाते हैं। यहाँ एक उदाहरण है कि हम तीन कॉलम के सेट के भीतर दो रो को कैसे nest बना सकते हैं।

```
<!DOCTYPE html>
<html>
<frameset cols="*,*,*">
        <frameset rows="*,*">
                <frame src="FirstFrame.html">
                <frame src="SecondFrame.html">
        </frameset>
        <frame src="ThirdFrame.html">
        <frame src="FourthFrame.html">
</frameset>
</html>
```

And here's how that HTML will render. - और यहां बताया गया है कि एचटीएमएल कैसे प्रस्तुत करेगा



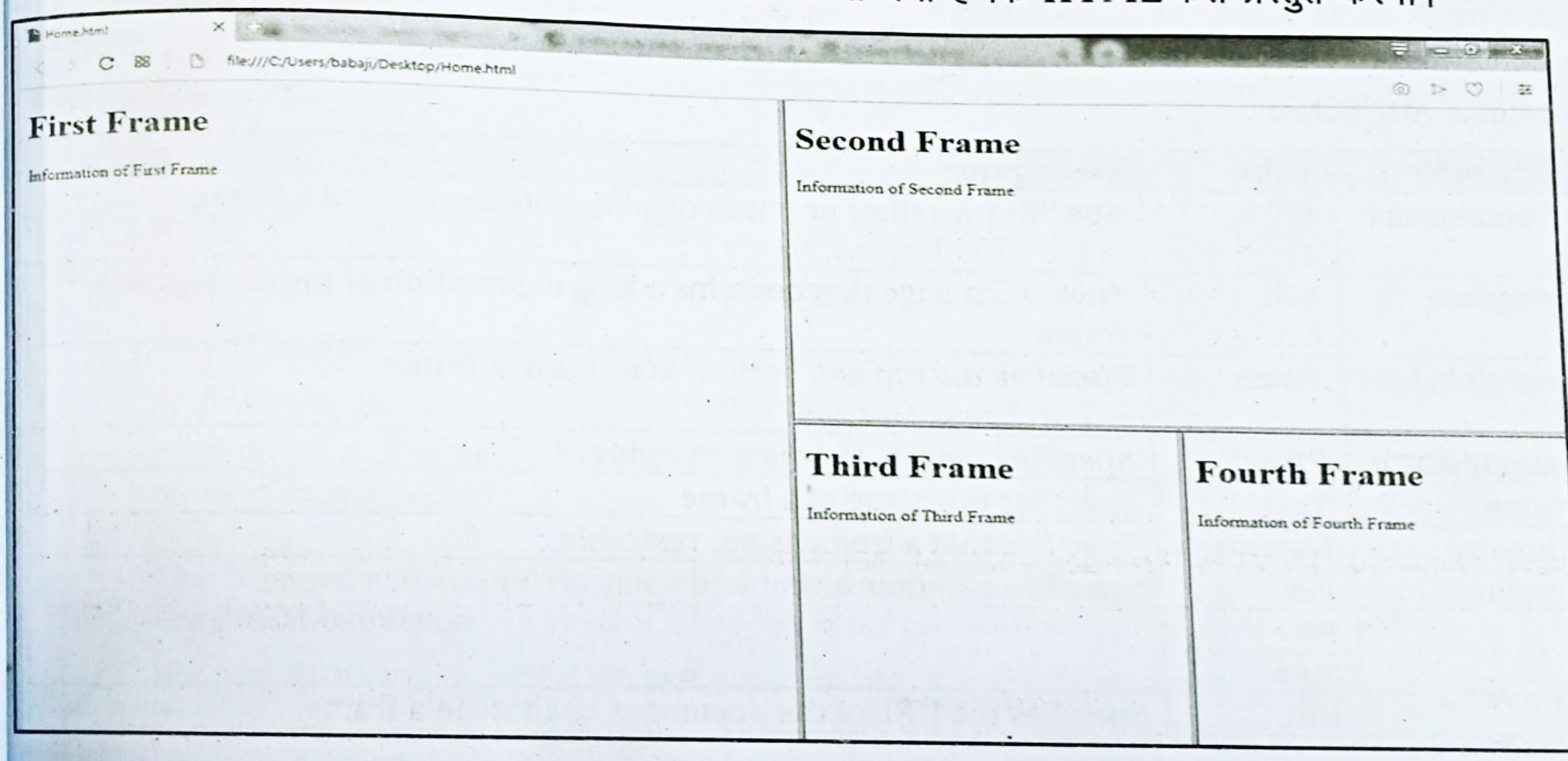
3.9 Nested Frames- नेस्टेड फ्रेम

The nested frameset takes the place of the first frame within the parent element. The nested element can be placed in any position. For example, if we want nested element to appear in the center position we would just re-arrange the elements like this.

नेस्टेड फ्रेमसेट मूल एलिमेन्ट के भीतर पहले फ्रेम की जगह लेता है। नेस्टेड एलिमेंट को किसी भी स्थिति में रखा जा सकता है। उदाहरण के लिए, यदि हम चाहते थे कि नेस्टेड एलिमेंट सेन्टर की स्थिति में आए तो हम इस तरह से एलिमेंटो को फिर से व्यवस्थित करेंगे।

```
<!DOCTYPE html>
<html>
<frameset cols="*,*">
        <frame src="FirstFrame.html">
        <frameset rows="*,*">
                <frame src="SecondFrame.html">
                <frameset cols="*,*">
                        <frame src="ThirdFrame.html">
                        <frame src="FourthFrame.html">
                </frameset>
        </frameset>
</frameset>
</html>
```

And here's how that HTML will render. यहां बताया गया है कि **HTML** कैसे प्रस्तुत करेगा।

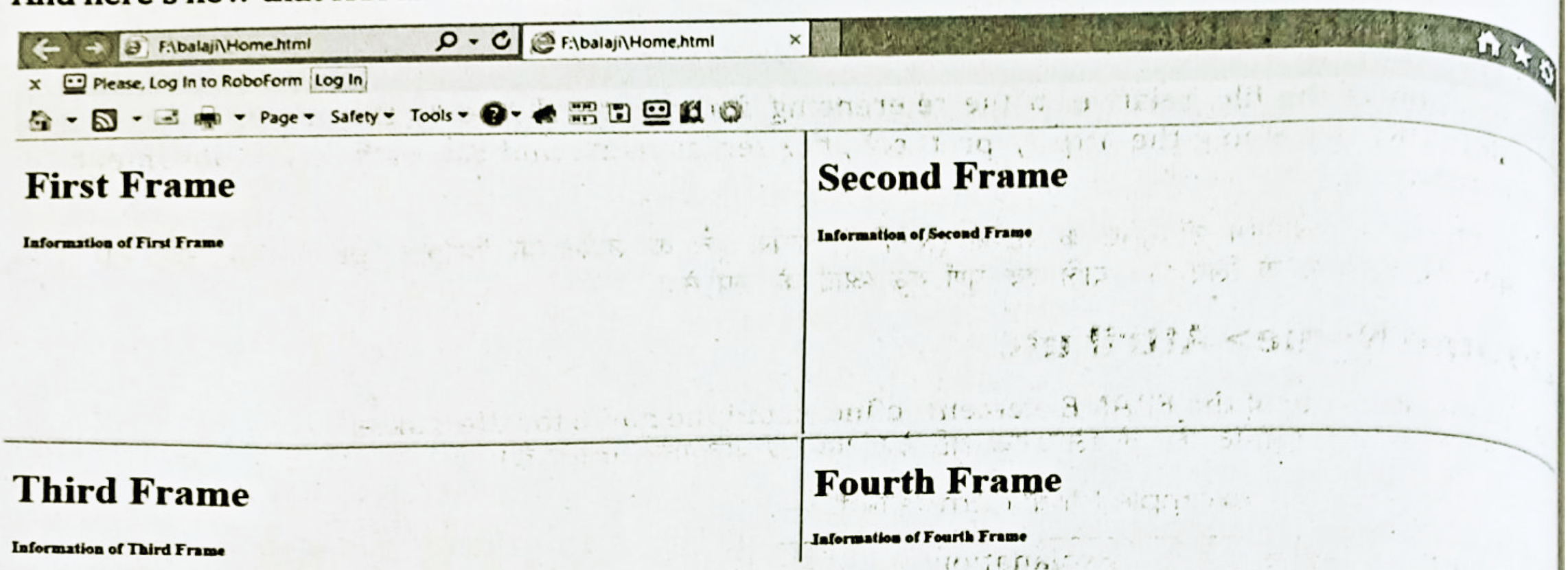


One more way to create a combination of rows and columns is to define a grid of columns and rows in a single frameset. For example, if you wanted a grid of four equally sized frames, you could use the following code.

रो और कॉलमो के संयोजन को बनाने का एक और तरीका एक एकल फ्रेमसेट में कॉलमो और रो की ग्रिड को डिफाईन करना है। उदाहरण के लिए, यदि आप चार समान आकार के फ्रेम का ग्रिड चाहते हैं, तो आप निम्न कोड का उपयोग कर सकते हैं।

```
<!DOCTYPE html>
<html>
<frameset rows="*,*" cols="*,*">
        <frame src=" FirstFrame.html">
        <frame src=" SecondFrame.html">
        <frame src=" ThirdFrame.html">
         <frame src=" FourthFrame.html">
         </frameset>
</html>
```

And here's how that HTML will render.



Optional Attributes:

Attribute	Value	Description
frameborder	0 1	Specifies whether or not to display a border around a frame
longdesc	URL	Specifies a page that contains a long description of the content of a frame
marginheight	Pixels	Specifies the top and bottom margins of a frame
marginwidth	Pixels	Specifies the left and right margins of a frame
Name	Text	Specifies the name of a frame
noresize	Noresize	Specifies that a frame is not resizable
scrolling	yes no auto	Specifies whether or not to display scrollbars in a frame
Src	URL	Specifies the URL of the document to show in a frame

<Frame SRC> Attribute

The src attribute instructs the browser where on the server to look for the web page that should be presented to the user in the current frame. This may be a file in the same directory, a file somewhere else on the same server or a file held on another server. The example shows a file in the same directory as the frameset that is calling it, but if the file was in a directory that is one level higher than the referencing frameset, the syntax would be:

Src ऐट्रिब्यूट किसी भी ब्राउजर को यह बताता है, कि सर्वर पर किस वेब पेज पर अपेक्षित डेटा रखा गया है। यह एक फाइल के रूप में उसी डायरेक्टरी में, या कहीं और परन्तु उसी सर्वर पर या एक फील्ड की भांति किसी और सर्वर पर भी रखी हो सकती है। उदहारण, फ्रेमसेट में फाइल को किसी और स्थान पर रखा गया है, परन्तु यदि फाइल को किसी और डायरेक्टरी में रखा गया हो तो फ्रेमसेट में निम्नवत तरीके से बदलाव किया जा सकता है।

```
<frame src="../header.html" frameborder="1" name="nav"title="Meditation Window"/>
```

Equates to 'move up one directory in the hierarchy'. You can also reference a file relative to the web site's root (anything after the domain name):

```
<frame src="/header.html" frameborder="1" name="nav" title="Meditation window"/>
```

src attribute Value

The location of the file, relative to the referencing document, relative to the server root or as a complete URI containing the http:// protocol , the server name and the path to the document on that server.

फाइल का स्थान, रेफरेन्सिंग डॉक्यूमेट के सापेक्ष (रिलेटिव), सर्वर रूट के सापेक्ष या http:// protocol, सर्वर नाम और उस सर्वर पर डाक्यूमेंट के लिए पाथ वाले एक पूर्ण यूआरआई के रूप में।

<Frame Name> Attribute

The name attribute of the FRAME element defines a unique name for the frame.
फ्रेम एलीमेंट का नाम एलीमेंट फ्रेम के लिए एक अद्वितीय नाम को परिभाषित करता है।

```
<frame src="example.html" name="test">
```

Attribute	Value	Explanation
name=" "	frame name	arbitrary name

Frame name - फ्रेम नाम

- Frame names must start in the alphabet. -

  फ्रेम का नाम अल्फाबेट में शुरू होना चाहिए।

  name="a001"

- The following symbols can be included in a frame name. -

  निम्नलिखित सिम्बल को एक फ्रेम नाम में शामिल किया जा सकता है।
  Hyphen (-), underscore (_), colon (:), period (.)
  name="frame_name"

- The frame name is used for the target attribute of the A element and the target attribute of the FORM element. -

  फ्रेम नाम का उपयोग ए एलीमेंट की टारगेट एट्रीब्यूट और फार्म एलीमेंट के टारगेट एलीमेंट के लिए किया जाता है।

<Frame Marginheight> Attribute

The marginheight attribute specifies the height between the content and the top and bottom of the frame, in pixels.

मार्जिन हाइट ऐट्रीब्यूट पिक्सेल में content और फ्रेम के ऊपर और नीचे के बीच की ऊंचाई को स्पेशिफाई करती है।

Syntax:<frame marginheight="pixels">

Attribute Values

Value	Description
Pixels	Specifies the top and bottom margins of a frame

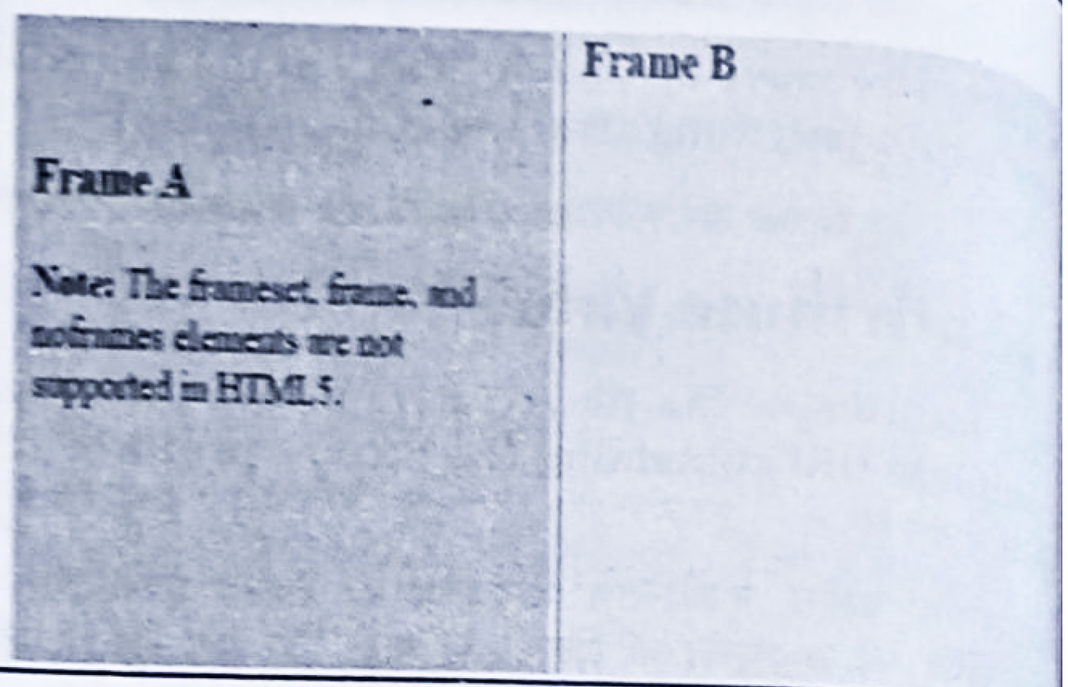
Edit This Code:

```
<!DOCTYPE html>
<html>

<frameset cols="50%,50%">
  <frame src="frame_a.htm" marginheight="50">
  <frame src="frame_b.htm" marginheight="0">
</frameset>

</html>
```

Result:



The first frame with a top and bottom margin of 50 pixels, and the second frame with a top and bottom margin of 0 pixels:

पहला फ्रेम 50 पिक्सल के ऊपरी और निचले मार्जिन के साथ, और दूसरा फ्रेम 0 पिक्सल के ऊपर और नीचे के मार्जिन के साथ:

<Frame Marginwidth> Attribute

The marginwidth attribute specifies the height between the content and the left and right of the frame, in pixels. मार्जिन की विशेषता, सामग्री और फ्रेम के बाएँ और दाएँ के बीच की ऊँचाई को पिक्सेल में निर्दिष्ट करती है।

Syntax : <frame marginwidth="pixels">

Value	Description
Pixels	Specifies the left and right margins of a frame

<Frame Scrolling> Attribute

Frame A should always have scrollbars: - फ्रेम ए में हमेशा स्क्रॉलबार होना चाहिए

- The scrolling attribute specifies whether or not to display scrollbars in a <frame>. स्क्रॉलिंग विशेषता निर्दिष्ट करती है कि स्क्रॉलबार को <frame> में प्रदर्शित किया जाए या नहीं।
- By default, scrollbars appear in a <frame> if the content is larger than the <frame>. डिफॉल्ट रूप से, स्क्रॉलबार एक <frame> में दिखाई देता है यदि सामग्री <frame> से बड़ी हो।

Note: Internet Explorer 8 (and earlier) and Opera 12 (and earlier) supports the scrolling attribute. However, support was removed in IE9 and Opera 15.

इंटरनेट एक्सप्लोरर 8 (और पहले) और ओपेरा 12 (और पहले) स्क्रॉलिंग विशेषता का समर्थन करता है। हालांकि, IE9 और ओपेरा 15 में समर्थन हटा दिया गया था।

Syntax : <frame scrolling="auto|yes|no">

Attribute Values

Value	Description
Auto	Scrollbars appear if needed (this is default)
Yes	Scrollbars are always shown (even if they are not needed)
No	Scrollbars are never shown (even if they are needed)

Edit This Code:

```
<!DOCTYPE html>
<html>

<frameset cols="50%,50%">
  <frame src="frame_a.htm" scrolling="yes">
  <frame src="frame_b.htm">
</frameset>

</html>
```

Result:

Frame A

Note: The frameset, frame, and noframes elements are not supported in HTML5.

Frame B

<noresize> Attribute

The noresize attribute specifies that a <frame> element cannot be resized by the user.
Noresize एट्रीब्यूट निर्दिष्ट करती है कि उपयोगकर्ता द्वारा एक <frame> एलीमेंट का आकार परिवर्तन नहीं किया जा सकता है।

By default, each <frame> in a <frameset> can be resized by dragging the border between the frames. However, this attribute locks the size of a frame.

डिफॉल्ट रूप से, फ्रेम के बीच की सीमा को खींचकर <frame> में प्रत्येक <frameset> का आकार बदला जा सकता है। हालाँकि, यह विशेषता किसी फ्रेम का आकार लॉक करती है।

Syntax : <frame noresize="noresize">

Attribute Values

Value	Description
noresize	The frame cannot be resized by the user

<noframe> Tag

- The <noframes> tag is a fallback tag for browsers that do not support frames. It can contain all the HTML elements that you can find inside the <body> element of a normal HTML page.

  <Noframes> टैग फ्रेम का समर्थन नहीं करने वाले ब्राउजरों के लिए एक फॉलबैक टैग है। इसमें उन सभी HTML तत्वों को शामिल किया जा सकता है जिन्हें आप सामान्य HTML पेज के <body> तत्व के अंदर पा सकते हैं।
- The <noframes> element can be used to link to a non-frameset version of the web site or to display a message to users that frames are required.

  <Noframes> तत्व का उपयोग वेब साइट के नॉन–फ्रेमसेट संस्करण से लिंक करने या उपयोगकर्ताओं को एक संदेश प्रदर्शित करने के लिए किया जा सकता है, जिसके लिए फ्रेम की आवश्यकता होती है।
- The <noframes> element goes inside the <frameset> element.
  <Noframes> तत्व <frameset> तत्व के अंदर जाता है।

Note: If you want to validate a page containing frames, be sure the <!DOCTYPE> is set to either "HTML Frameset DTD" or "XHTML Frameset DTD".

नोट: यदि आप किसी पृष्ठ को फ्रेम से सत्यापित करना चाहते हैं, तो सुनिश्चित करें कि <! DOCTYPE> या तो "HTML Frameset DTD" या "XHTML Frameset DTD" पर सेट है।

3.9.1 Targeting Frames with Links - लिंक के साथ फ्रेम को लक्षित करना

One of the most common uses of frames is to build sticky navigation into a frame that is always visible regardless of the position of the contents of the other frames. When properly implemented, navigation links will cause new resources to load in one frame while the other frames remain static.

फ्रेम का सबसे आम उपयोग एक फ्रेम में sticky नेविगेशन का निर्माण करना है जो हमेशा अन्य फ्रेमों के कन्टेंट की स्थिति की परवाह किए बिना दिखाई देता है। जब ठीक से लागू किया जाता है, तो नेविगेशन लिंक एक फ्रेम में लोड करने के लिए नए रिसोर्सेज का कारण बनेंगे, जबकि अन्य फ्रेम स्थिर रहेंगे।

Anchors can be formatted to target specific frames by assigning a name attribute to a targeted frame element, and using the target attributed within an element to load the href in the targeted frame.

एंकरों को एक टार्गेट फ्रेम एलिमेंट में नाम ऐट्रीब्यूट बताकर स्पेशिफिक फ्रेमों को टारगेट करने के लिए फार्मेट किया जा सकता है, और एलिमेंट के भीतर टारगेट ऐट्रीब्यूट का उपयोग टारगेट फ्रेम मे एचआरईफ को लोड करने के लिए किया जाता है।

If all of that is a little confusing, let's take it step-by-step.
यदि यह सब थोड़ा कन्फ्यूसिंग है, तो इसे स्टेप–बाई–स्टेप देखें।

The first step in making this happen is to assign a name to the frame where we want links to open. Using the layout we created just a minute ago, we would probably want to use the left-hand column for our navigation and the center column as our targeted frame.

ऐसा करने के लिए पहला चरण उस फ्रेम का नाम स्पेशिफाई करना है जहां हम लिंक ओपेन करना चाहते हैं। हमने अभी एक मिनट पहले बनाए गए लेआउट का उपयोग करते हुए, हम संभवतः अपने नेविगेशन के लिए बाएं नेविगेशन कॉलम और सेंटर कॉलम को हम टारगेट फ्रेम के रूप में उपयोग करना चाहते हैं।

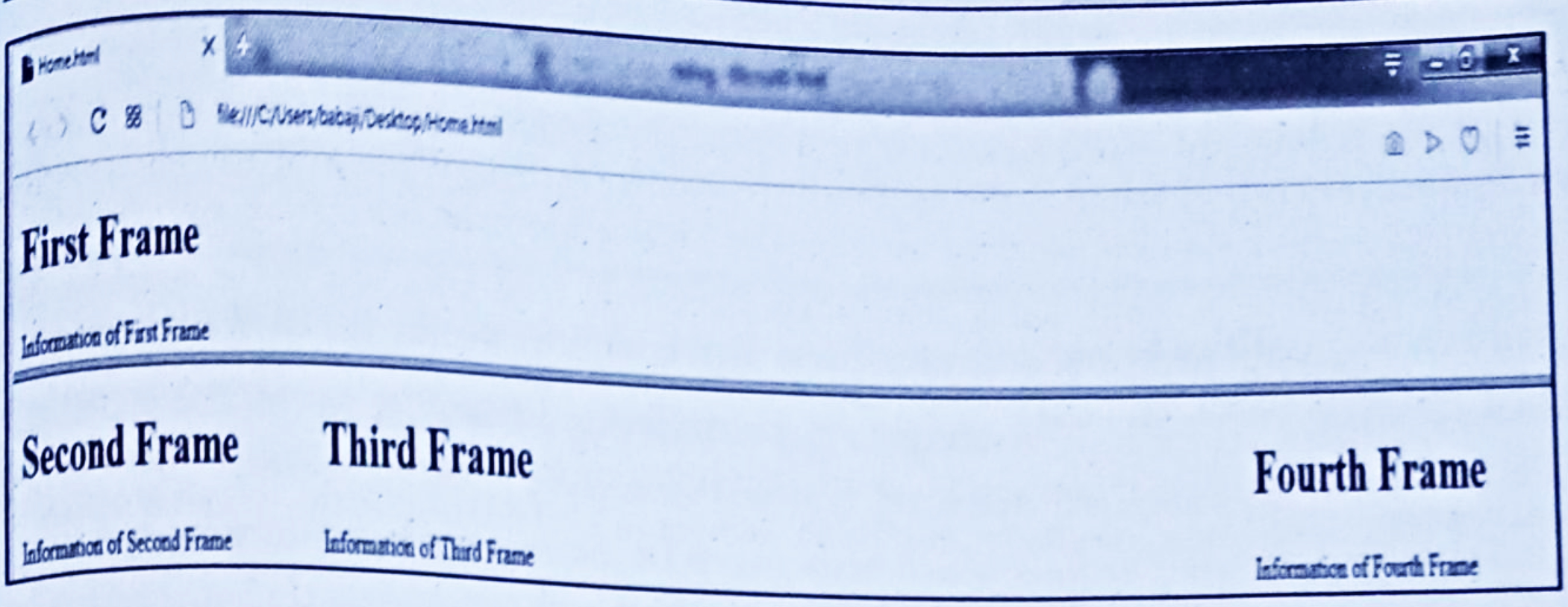
In order to do this, we need to assign a name to our target.
ऐसा करने के लिए, हमें अपने टारगेट को एक नाम स्पेशिफाई करना होगा।

Step 1: For this I have created **home.html** like this ...

```
<!DOCTYPE html>
<html>
<frameset rows="100px,*">
        <frame noresize src="FirstFrame.html" marginheight="15" scrolling="no">

        <frameset cols="20%,*,20%"><frame src="SecondFrame.html"
frameborder="0">
                <frame src="ThirdFrame.html" name="middle" frameborder="0">
                <frame src="FourthFrame.html" frameborder="0">
        </frameset>
</frameset>
</html>
```

Step 2: Save the file **home.html**, this web page looking like this way...



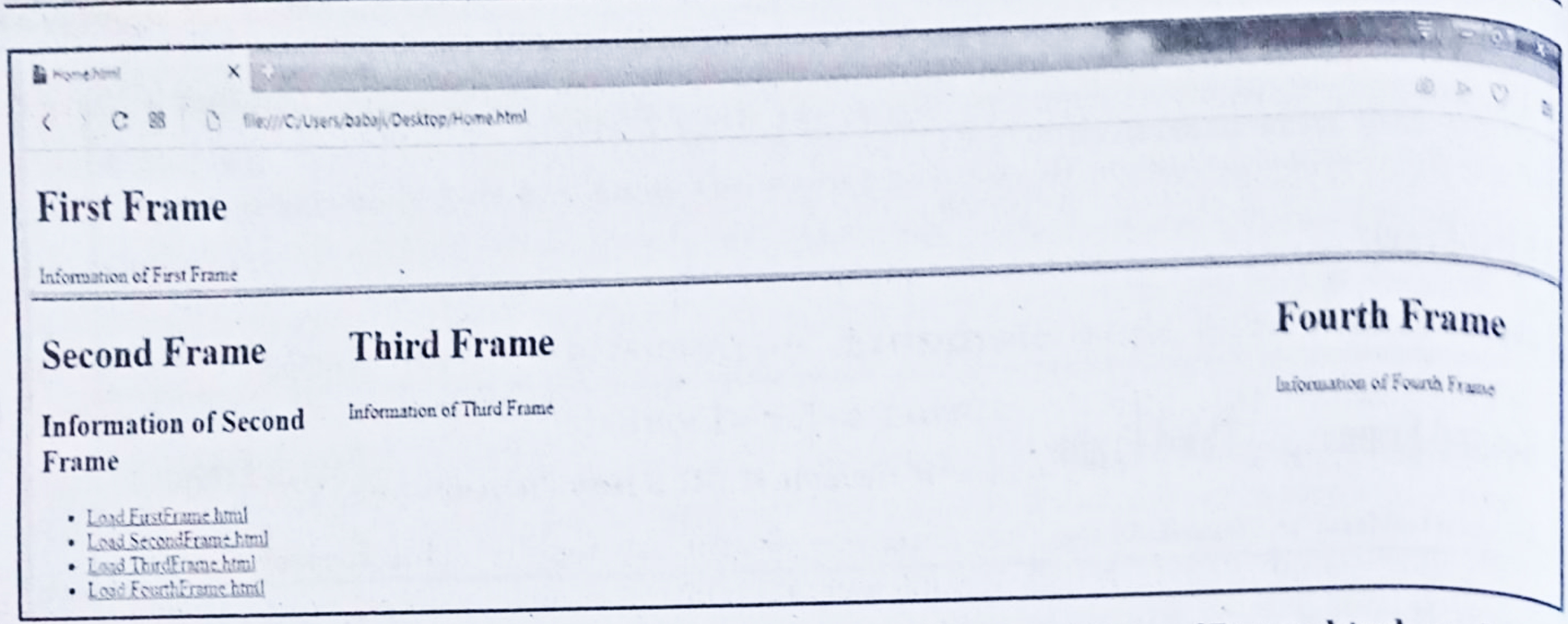
Step 3: Now that we've named the center column **name="middle"** we can create a couple of links in our left-hand column source webpage **SecondFrame.html** and target the center column.

```
<!DOCTYPE html>
<html>
<body>
        <h1>Second Frame</h1>
        <h2>Information of Second Frame</h2>
        <ul>
            <li><a href="FirstFrame.html" target="middle">Load
            FirstFrame.html</a></li>
            <li><a href="SecondFrmae.html" target="middle">Load
            SecondFrame.html</a></li>
            <li><a href="ThirdFrame.html" target="middle">Load
            ThirdFrame.html</a></li>
            <li><a href="FourthFrame.html" target="middle">Load
            FourthFrame.html</a></li>
        </ul>
</body>
</html>
```

Now when we load our webpage we have four navigation links in the left sidebar, and when we click a link the contents of that file are loaded in the middle column frame with the attribute **name="middle"**.

अब जब हम अपने वेबपेज को लोड करते हैं तो हमारे बाएं साइडबार में चार नेविगेशन लिंक होते हैं, और जब हम एक लिंक पर क्लिक करते हैं तो उस फाइल की सामग्री को विशेषता **name="middle"** के साथ मध्य कॉलम फ्रेम में लोड किया जाता है।

Step 4: When we load our page here's what we initially see.
जब हम अपने पेज को यहां लोड करते हैं तो हम शुरू में देखते हैं।



Step 5: Click on the navigation links **Load FirstFrame.html** or **Load SecondFrame.html** or **Load ThirdFrame.html** or **Load FourthFrame.html** you will get the answer how the websites are building in real time scenario.

नेविगेशन लिंक पर क्लिक करें लोड FirstFrame.html या लोड SecondFrame.html या लोड ThirdFrame.html या लोड FourthFrame.html। आपको उत्तर मिलेगा कि वेबसाइट वास्तविक समय परिदृश्य में कैसे निर्माण कर रही हैं।

3.10 Introduction of HTML 5 - एचटीएमएल 5 एक परिचय

HTML5 is the latest version of the standard that defines HTML. The term represents two different concepts. It is a new version of the language HTML, with new elements, attributes, and behaviors, and a larger set of technologies that allows the building of more diverse and powerful Web sites and applications. This set is sometimes called HTML5 & friends and often shortened to just HTML5.

HTML5 नवीनतम संस्करण है जो HTML को परिभाषित करता है। यह शब्द दो भिन्न कान्सेप्ट को रिप्रेसेन्ट करता है। यह भाषा HTML का एक नया वर्जन है, जिसमें नए एलिमेंट, ऐट्रीब्यूट और बिहैवियर, और टेक्नॉलिजीस का एक बड़ा सेट है जो अधिक विविध और शक्तिशाली वेब साइटों और एप्लिकेशन के निर्माण की अनुमति देता है। इस सेट को कभी–कभी HTML5 & friends कहा जाता है और अक्सर इसे शार्ट मे HTML5 कहते है।

HTML5 technologies, classified into several groups based on their function, they are:
HTML5 प्रौद्योगिकियो को उनके कार्य के आधार पर अनेक समूहों में वर्गीकृत किया गया हैं, वे हैं:

- **Semantics:** allowing you to describe more precisely what your content is. - आपको और अधिक सटीक वर्णन करने की अनुमति देता है कि आपकी सामग्री क्या है।
- **Connectivity:** allowing you to communicate with the server in new and innovative ways. - आपको नए और नए तरीकों से सर्वर से संवाद करने की अनुमति देता है
- **Offline and storage:** allowing webpages to store data on the client-side locally and operate offline more efficiently. - वेबपेजों को क्लाइंट–साइड पर डेटा को स्थानीय रूप से संग्रहीत करने और ऑफलाइन अधिक कुशलता से संचालित करने की अनुमति देता है।
- **Multimedia:** making video and audio first-class citizens in the Open Web. - ओपेन वेब में वीडियो और ऑडियो प्रथम श्रेणी के नागरिक बनाना।
- **2D/3D graphics and effects:** allowing a much more diverse range of presentation options. - प्रस्तुति विकल्पों की एक बहुत अधिक विविध रेंज की अनुमति।

- **Performance and integration:** providing greater speed optimization and better usage of computer hardware. - अधिक गति अनुकूलन और कंप्यूटर हार्डवेयर का वेहतर उपयोग प्रदान करना।
- **Device access:** allowing for the usage of various input and output devices. - विभिन्न इनपुट और आउटपुट डिवाइस के उपयोग की अनुमति।
- **Styling:** letting authors write more sophisticated themes. - लेखकों को अधिक परिष्कृत विषय लिखने दें।

3.10.1 HTML5 New Elements: एचटीएमएल 5 एक नया एलिमेंट

HTML5 added the following attributes for <input>:

(These attributes we are going discuss in detail in HTML 5 New Elements topic.)

- **Autocomplete:** A string that describes what if any type of autocomplete functionality the input should provide. A typical implementation of autocomplete simply recalls previous values entered in the same input field, but more complex forms of autocomplete can exist. For instance, a browser could integrate with a device's contacts list to autocomplete email addresses in an email input field.

  एक स्ट्रिंग जो बताती है कि यदि किसी भी प्रकार की autocomplete फंक्शनलिटी इनपुट प्रोवाइड करना चाहिए। autocomplete का एक विशिष्ट इम्प्लिमेंटेसन है समान इनपुट फील्ड मे इन्टर करते समय पिछली वैल्यू अपने आप रिकॉल होती है, लेकिन autocomplete के अधिक जटिल रूप मौजूद हो सकते हैं। उदाहरण के लिए, एक ब्राउजर एक ईमेल इनपुट क्षेत्र में autocomplete ईमेल पते के लिए डिवाइस की संपर्क सूची के साथ एकीकृत कर सकता है।

- **Autofocus:** A Boolean which, if present, makes the input take focus when the form is presented. एक Boolean, जो यदि मौजूद है, तो फॉर्म को प्रस्तुत करने पर इनपुट को ध्यान में रखता है
- **Form:** The id of the <form> of which the input is a member; if absent, the input is a member of the nearest containing form, or is not a member of a form at all.

  <Form> की आईडी जिसमें इनपुट एक मेम्बर है; यदि अनुपस्थित है, तो इनपुट निकटतम सम्मिलित फॉर्म का एक मेम्बर है, या किसी फॉर्म का मेम्बर नहीं है।

- **List:** The id of a <datalist> element that provides a list of suggested values for the input.

  <datalist> element की id जो इनपुट के लिए सुझाए गए मूल्यों की एक सूची प्रदान करती है।

- **Name:** The input's name, to identify the input in the data submitted with the form's data.

  इनपुट का नाम, फॉर्म के डेटा के साथ सबमिट किए गए डेटा में इनपुट की पहचान करने के लिए।

- **Required:** A Boolean which, if true, indicates that the input must have a value before the form can be submitted.

  एक Boolean, जो true है, इंडिकेट करता है कि फॉर्म सबमिट करने से पहले एक इनपुट वैल्यू होना चाहिए।

- **Tabindex:** A numeric value providing guidance to the user agent as to the order in which controls receive focus when the user presses the Tab key. एक न्यूमेरिक वैल्यू यूजर एजेन्ट को गाईड करता है एक क्रम में जिसमें कन्ट्रोल फोकस रिसीव करता है जब यूजर टैब की प्रेस करें।

HTML5 offers new elements for better document structure:
HTML5 बेहतर दस्तावेज संरचना के लिए नए एलिमेंट प्रदान करता है

1. **Section tag:** The HTML <section> element represents a standalone section — which doesn't have a more specific semantic element to represent it — contained within an HTML document. The <section> tag defines sections in a document, such as chapters, headers, footers, or any other sections of the document.Typically, but not always, sections have a heading.