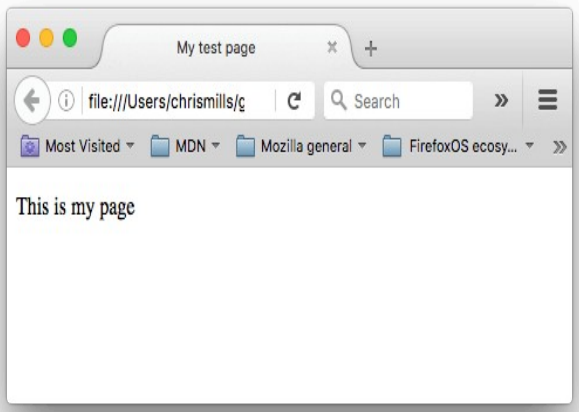
Programme: B.Sc.
Subject: Computer Science
Semester: I
Paper Code: CSG110
Paper Title: GE (Client Side Web Development)

Unit III: Introduction
Module Name: Anatomy of a HTML Document : Head,Title,Body, Metadata
Module No: 05
Name of the Presenter: Ms. Dipti Nene

Anatomy of a HTML document

That wraps up the basics of individual HTML elements, but they aren't very useful on their own. Now we'll look at how individual elements are combined to form an entire HTML page:

```
<html>
<head>
   <meta charset="utf-8">
   <title>My test page   </title>
 </head>
 <body>
            <p>This is my page</
p>
 </body>
 </html>
```



Here we have:

1. **<!DOCTYPE html>:** The doctype. In the mists of time, when HTML was young (about 1991/2), doctypes were meant to act as links to a set of rules that the HTML page had to follow to be considered good HTML, which could mean automatic error checking and other useful things. They used to look something like this:

   ```
   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
                         "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
   ```

   However, these days no one really cares about them, and they are really just a historical artifact that needs to be included for everything to work right. <!DOCTYPE html> is the shortest string of characters that counts as a valid doctype; that's all you really need to know.
2. **<html> </html>:** The <html> element. This element wraps all the content on the entire page, and is sometimes known as the root element.
3. **<head> </head>:** The <head> element. This element acts as a container for all the stuff you want to include on the HTML page that *isn't* the content you are showing to your page's viewers. This includes things like keywords and a page description that you want to appear in search results, CSS to style our content, character set declarations, and more. You'll learn more about this in the next article in the series.
4. **<meta charset="utf-8">:** This element sets the character set your document should use to UTF-8, which includes most characters from the vast majority of human written languages. Essentially it can now handle any textual content you might put on it. There is no reason not to set this, and it can help avoid some problems later.
5. **<title></title>:** The <title> element. This sets the title of your page, which is the title that appears in the browser tab the page is loaded in, and is used to describe the page when you bookmark/favorite it.

   Adding a title

   We've already seen the <title> element in action — this can be used to add a title to the document. This however can get confused with the <h1> element, which is used to add a top level heading to your body content — this is also sometimes referred to as the page title. But they are different things!

   The <h1> element appears on the page when loaded in the browser — generally this should be used once per page, to mark up the title of your page content (the story title, or news headline, or whatever is appropriate to your usage.)

The <title> element is metadata that represents the title of the overall HTML document (not the document's content.)

6. **<body> </body>:** The <body> element. This contains *all* the content that you want to show to web users when they visit your page, whether that's text, images, videos, games, playable audio tracks, or whatever else.

Example of HTML code:

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8">
  <title>My test page</title>
 </head>
 <body>
   <p><b>This is my page<b></p>
   <p><i>This is my page</i></p>
 </body>
</html>
```

Entity references: Including special characters in HTML

In HTML, the characters <, >,",' and & are special characters. They are parts of the HTML syntax itself, so how do you include one of these characters in your text, for example if you really want to use an ampersand or less than sign, and not have it interpreted as code as some browsers may do?
We have to use character references — special codes that represent characters, and can be used in these exact circumstances. Each character reference is started with an ampersand (&), and ended by a semicolon (;).

Literal character	Character reference equivalent
<	<
>	>

"	"
'	'
&	&

Example code 1 & 2:

```
<p>In HTML, you define a paragraph using the <p> element.</p>
<p>In HTML, you define a paragraph using the &lt;p&gt; element.</p>
```

Note: (element appeared on new line because of <p> second <p> in 1st example)

Output:

In HTML, you define a paragraph using the
element.
In HTML, you define a paragraph using the <p> element.

HTML comments

In HTML, as with most programming languages, there is a mechanism available to write comments in the code — comments are ignored by the browser and invisible to the user, and their purpose is to allow you to include comments in the code to say how your code works, what the different parts of the code do, etc. This can be very useful if you return to a code base that you've not worked on for six months, and can't remember what you did — or if you hand your code over to someone else to work on.
To turn a section of content inside your HTML file into a comment, you need to wrap it in the special markers <!-- and -->, for example:

```
<p>I'm not inside a comment</p> <!-- <p> I am </p> -->
```

live output

I'm not inside a comment