

Programme: B.Sc.

Subject: Computer Science

Semester: I

Paper Code: CSG110

Paper Title: GE (Client Side Web Development)

Unit I: Introduction

Module Name: Nesting Elements,

Block versus Inline Elements, Empty

Elements, Emphasis and importance

Module No: 03

Name of the Presenter: Ms. Dipti Nene

Nesting

Nesting elements

You can put elements inside other elements too — this is called **nesting**. If we wanted to state that our cat is **very** grumpy, we could wrap the word "very" in a `` element, which means that the word is to be strongly emphasized:

```
<p>My cat is <strong>very</strong> grumpy.</p>
```

You do however need to make sure that your elements are properly nested: in the example above, we opened the `p` element first, then the `strong` element, therefore we have to close the `strong` element first, then the `p`. The following is incorrect:

```
<p>My cat is <strong>very grumpy.</p></strong>
```

The elements have to open and close correctly, so they are clearly inside or outside one another. If they overlap like above, then your web browser will try to make a best guess at what you were trying to say, and you may well get unexpected results. So don't do it!!

Children

An element that is an immediate descendent of another element or nested within another element is called a child. These become useful when using CSS child selectors and psuedo-elements.

There are two important categories of elements in HTML which you should know about. They are block-level elements and inline elements.

- **Block-level elements** form a visible block on a page — **they will appear on a new line** from whatever content went before it, and any content that goes after it will also appear on a new line. Block-level elements tend to be structural elements on the page that represent, for example, paragraphs, lists, navigation menus, footers, etc. A block-level element wouldn't be nested inside an inline element, but it might be nested inside another block-level element.

```
<p>fourth</p><p>fifth</p><p>sixth</p>
```

o/p

fourth

fifth

sixth

- **Inline elements** are those that are contained within block-level elements and surround only small parts of the document's content, not entire paragraphs and groupings of content. An inline element will **not cause anew line to appear** in the document; they would normally appear inside a paragraph of text,
- for example an <a> element (hyperlink) or emphasis elements such as or .

Take the following example:

Inline----- `firstsecondthird`

o/p

first **second** third

Emphasis and Strong importance

- To emphasize important words in spoken language ,we tend stress certain words, altering the meaning of what we are saying. Similarly, in written language we tend to stress words by highlighting them in italics/bold etc. .
- This liquid is `highly toxic`
- `<i>` is used to convey a meaning traditionally conveyed by italic: foreign words, taxonomic designation, technical terms, a thought...
- `` is used to convey a meaning traditionally conveyed by bold: key words, product names, lead sentence...
- `<u>` is used to convey a meaning traditionally conveyed by underline: proper name, misspelling...

empty elements

Not all elements follow the above pattern of opening tag, content, closing tag. Some elements consist only of a single tag, which is usually used to insert/embed something in the document at the place it is included.

For example, the `` element embeds an image file onto a page in the position This would output the following on your page:

```
<img  
src=https://raw.githubusercontent.com/mdn/beginner-html-site/gh-pages/  
images/firefox-icon.png>/>
```

Note: Empty elements are also sometimes called *void elements*.