



TRAINING DRAGON WEB DESIGN

TRAINER: EMILIANO

BASIC TAGS

- comments
- headings
- header, main, footer
- paragraphs
- images
- anchors
- lists
- divs and spans
- flow content vs phrasing content elements

COMMENTS

- notes for designers / developers
- not meant to be shown
- describe code
- facilitate teamwork

```
<!-- this is a comment -->
```

- **DEMO:** comments, viewing source code

HEADINGS

- 6 headings
- h1 to h6
- block level / flow content elements
- used to describe pages, sections, subsections

```
<h1>this is an h1</h1>  
<h2>this is an h2</h2>  
<h3>this is an h3</h3>  
<h4>this is an h4</h4>  
<h5>this is an h5</h5>  
<h6>this is an h6</h6>
```

- **DEMO:** headings

HEADER, MAIN, FOOTER

- 3 parts the content is usually divided into
- **header**: may contain logo, navigation, heading(s)
- **main**: contains most of the content
- **footer**: often contains copyright info, useful links, breadcrumbs navigation

```
...  
<body>  
  <header>  
    <h1>a website</h1>  
    <hr>  
  </header>  
  <main>  
    <h2>main content</h2>  
  </main>  
  <footer>  
    <h4>a footer</h4>  
    <hr>  
  </footer>  
</body>  
...
```

- **DEMO:** dividing body into header, main, footer

P TAGS (PARAGRAPHS)

- block level / flow content elements
- can only contain inline elements (eg: images, links)
- define a block of text as a paragraph

```
<p>this is a short paragraph</p>  
<p>this is another paragraph</p>
```

- **DEMO:** inserting paragraphs inside main tag

IMAGES

- **img** tags
- inline level / phrasing content elements
- self-closing tags
- attributes: **src, width, height, alt, title**
- common formats: **jpeg / jpg, png, gif, svg**

```

```

- **DEMO:** display a picture in a paragraph

ANCHORS

- **a** tags
- used to create hyperlinks
- **phrasing content** elements
- attributes: **href**, **title**, **target**
- **absolute** or **relative** paths

```
<a href="http://url.com"  
  title="tooltip"  
  target="_blank">hyperlink</a>
```

- **DEMO:** add an external link to Training Dragon
- **DEMO:** add an internal link to a second page

LISTS

- ordered / unordered lists, list items
- ol, ul, li
- block elements
- used to group elements together
- commonly used to create navigations

```
<ul>
  <li>item</li>
  <li>item</li>
  <li>item</li>
</ul>
```

```
<ol>
  <li>item</li>
  <li>item</li>
  <li>item</li>
</ol>
```

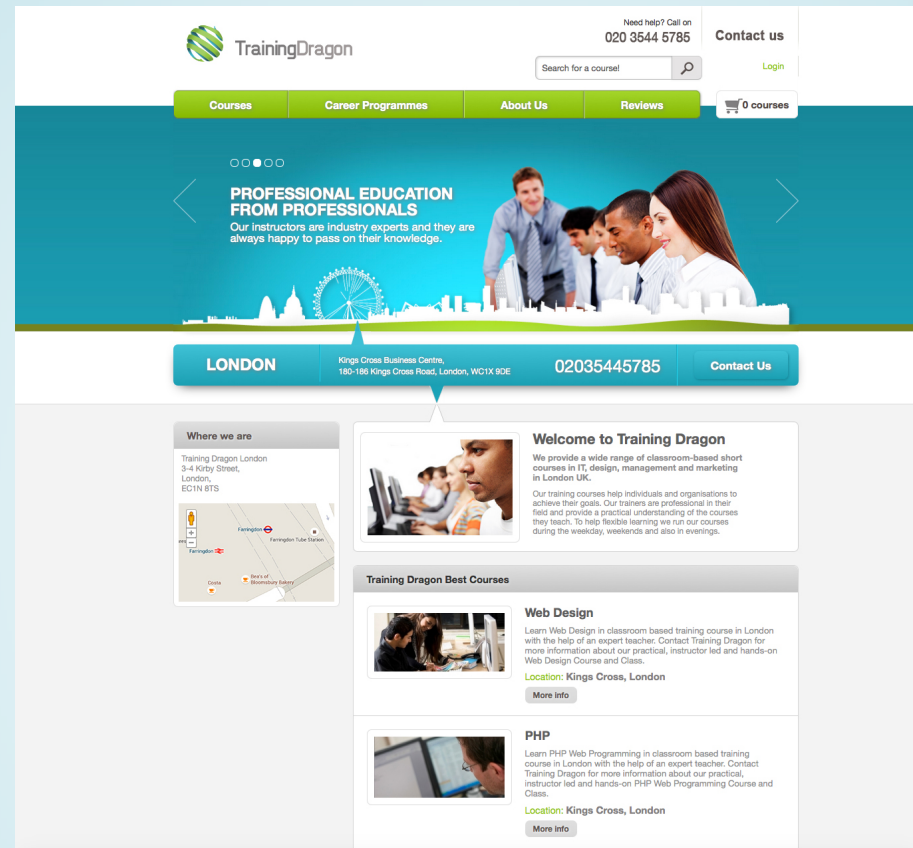
- **DEMO:** a bullet list and a numbered list

DIVS

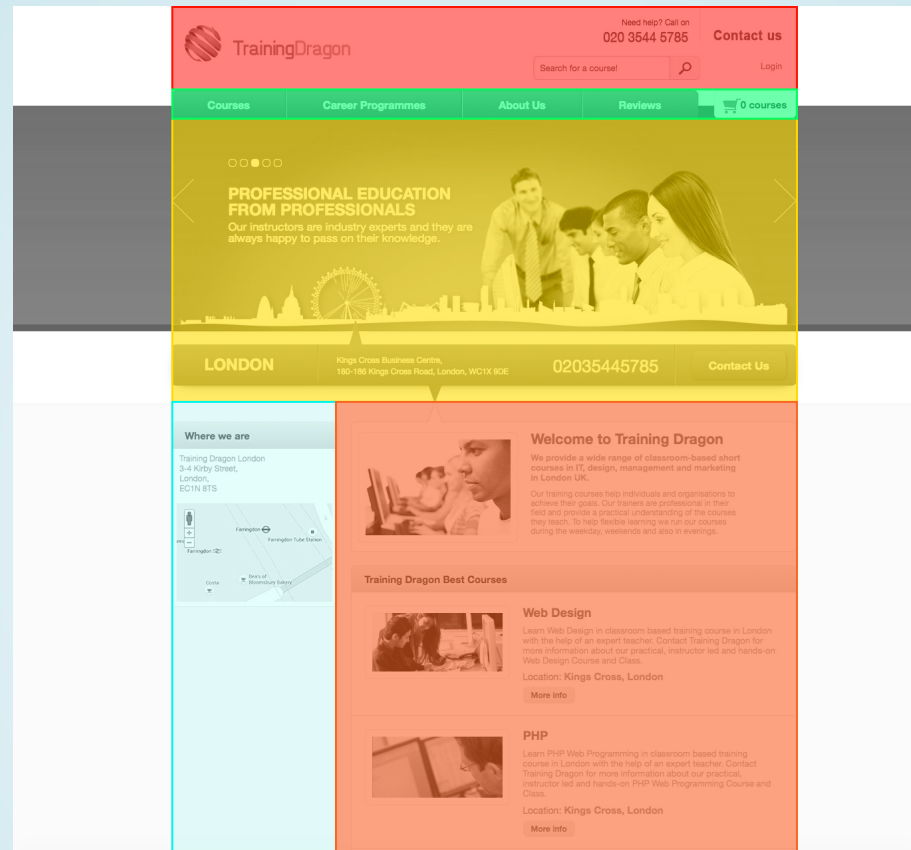
- division elements
- flow content (block) elements
- generic containers
- used to organise content and arrange elements for layouts

```
<div>  
  <h3>this is a div with a paragraph</h3>  
  <p>this is a paragraph</p>  
</div>
```

A LAYOUT



DIVS



DIVS - SOURCE CODE

about our practical, instructor led and hands-on Web Design Course and Class.

Location: Kings Cross, London

[More info](#)

PHP

Learn PHP Web Programming in classroom based training course in London with the help of an expert teacher. Contact Training Dragon for more information about our practical, instructor led and hands-on PHP Web Programming Course and Class.

Location: Kings Cross, London

[More info](#)

Graphic Design

Learn Graphic Design in classroom based training course in London with the help of an expert teacher. Contact Training Dragon for more information about our practical, instructor led and hands-on Graphic Design Course and Class.

Location: Kings Cross, London

[More info](#)

ElementsNetworkSourcesTimelineProfilesResourcesAuditsConsole

S...C...S... (index) (index):formatted x

```
32 if (search_val == "Search for a course!" || search_val == "")
33     return false;
34 if (search_val != "") {
35     //search_val = search_val.replace(/-/g, "");
36     //search_val = search_val.replace("&", "_");
37     search_val = encodeURIComponent(search_val);
38     window.location.href = "/course/search/" + search_val;
39 }
40 }
41 </script>
42 <div id="header" class="">
43   <div id="top-bar" class="clearfix container">
44     <div class="topBarWrap">
45       <div class="branding clearfix">
46         <a href="/" class="logo">
47           <div class="visible-desktop">
48             <div id="headLeft" style="display: inline-block;">
49               <div class="logo-rotate hiddenFace horLogo" id="hl-left"></div>
50               <div class="logo-rotate hiddenFace horLogo" id="hl-right"></div>
51               <div class="logo-rotate horLogo" id="hl-front"></div>
52               <div class="logo-rotate hiddenFace horLogo" id="hl-back"></div>
53             </div>
54             
58           <div class="branding-right pull-right visible-desktop">
59             <div class="branding-phone">
60               Need help? Call on
61               <span class="sprite-header-phone"></span>
62             </div>
63           </div>
```

Aa .* div 171 matches Cancel

Line 1, Column 1

Scope Watch

BLOCK LEVEL VS INLINE LEVEL ELEMENTS - I

- **Block** level elements begin on new lines and occupy the whole space of their parent tag
- **Inline** level elements can start anywhere in the line

BLOCK LEVEL VS INLINE LEVEL ELEMENTS - II

- **Block** level elements can contain both inline and block elements
- **Inline** level elements can only contain other inline elements

```
<div>
  
  
</div>
<div>
  <p>a paragraph</p>
  <p>another paragraph</p>
</div>
```