HYPER TEXT  MARKUP  LANGUAGE

Separation of Concerns

html = STRUCTURE
css = STYLE
javascript = BEHAVIOR
HTML

at the root of every web page there is html

html is used to define your contents meaning

html uses elements to describe the structure of pages

the tags/the elements/the markup describes the meaning of the content, this is called semantics
<!DOCTYPE html>, <!doctype html>,
<!DOCTYPE HTML> isn’t case sensitive, but it is more common to use the first example
HTML syntax

Start tag
<tag name>

Contents

End tag
</tag name>
HTML

syntax

start tag

option

<tag name attribute name=""value">
<head>
    <title>each html page must have a title element</title>
</head>

a title should be short, descriptive and unique to each page

titles are used by search engines to get a sense of your pages content and is typically the text that appears in their search results
a title element is required

it can not contain any formatting, html, images or links
HTML

<body>anything written inside the <body> tag will be displayed in the browser window. </body>
headings - establish a hierarchy of information

<h1>largest heading</h1> where h1 is top level heading

<h2>........................</h2>
<h3>........................</h3>
<h4>........................</h4>
<h5>........................</h5>
<h6>smallest heading</h6>
while browsers progressively show headings smaller moving from h1 to h6, you should choose headings solely on the hierarchy appropriate for your content

this makes your page stronger semantically

improves seo and accessibility
search engines weigh headings heavily, particularly h1

screen readers navigate headings via keyboards

think of headings as outlines
HTML

sometimes a heading has multiple consecutive headings.

grouping these in a hgroup will indicate that these are related

<hgroup>
  <h1>Animal Crackers</h1>
  <h2>Whole Wheat, Ginger bread, Etc...</h2>
</hgroup>
sectioning content elements such as:

article, aside, nav, section, footer, header

demarcate distinct sections within a document and define the scope of h1-h6 elements within them
<header>
the header tag represents a group of introductory or navigational aids</header>

<header>
is also appropriate for marking up a group of introductory or navigational content deeper within a page</header>

<header>
is NOT interchangeable with a heading! Each has its own semantic purpose.
</header>
HTML

<header>don’t nest a header within a footer, nor a header within a header, nor a header within an address element</header>

<header>doesn't always have to contain a nav element</header>
<nav>the nav tag represents a section of a page that links to other pages or parts within the page</nav>

<nav>use the nav element only for your documents most important groups of links, not all of them</nav>

<nav>it is best practice to structure your links in an ordered list <ol> or unordered list <ul> element</nav>
<article>the article tag represents a self-contained composition in a document, page, application, or site and that is, in principle, independently distributable or reusable, e.g. in syndication.</article>

<article>youd can nest article tags inside of other articles tags so long as the inner article tag is related to the whole</article>
<section>the section tag represents a generic section of a document.</section>

<section>It is a thematic group of content typically with a heading</section>

<section>it is not a generic container element</section>
HTML

<aside>
the aside tag represents a section of a page that consists of content that is tangentially related to the content around the aside element, and which could be considered separate from that content.
</aside>

<aside>
the aside element doesn’t affect the layout of the page</aside>
<footer>the footer represents a footer for its nearest ancestor sectioning content or sectioning root element.</footer>

<footer>typically includes information about its section, such as links to related documents, copyright information, etc.</footer>

<footer>you can’t nest a header in a footer</footer>