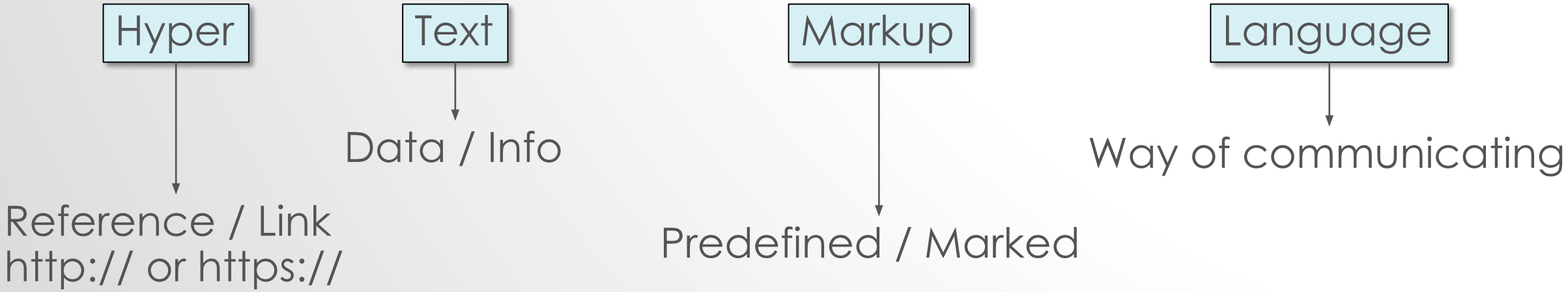
```
<html lang="en">
    <head>
        <script type="text/javascript" src="somefile.js"></script>
    </head>

    <!-- I'm a comment -->
    <body class="default-theme">
        <div id="menu">
            <span> Hello World! </span>
        </div>
    </body>
</html>
```

# HTML

The language of the internet
by: tjjenk2

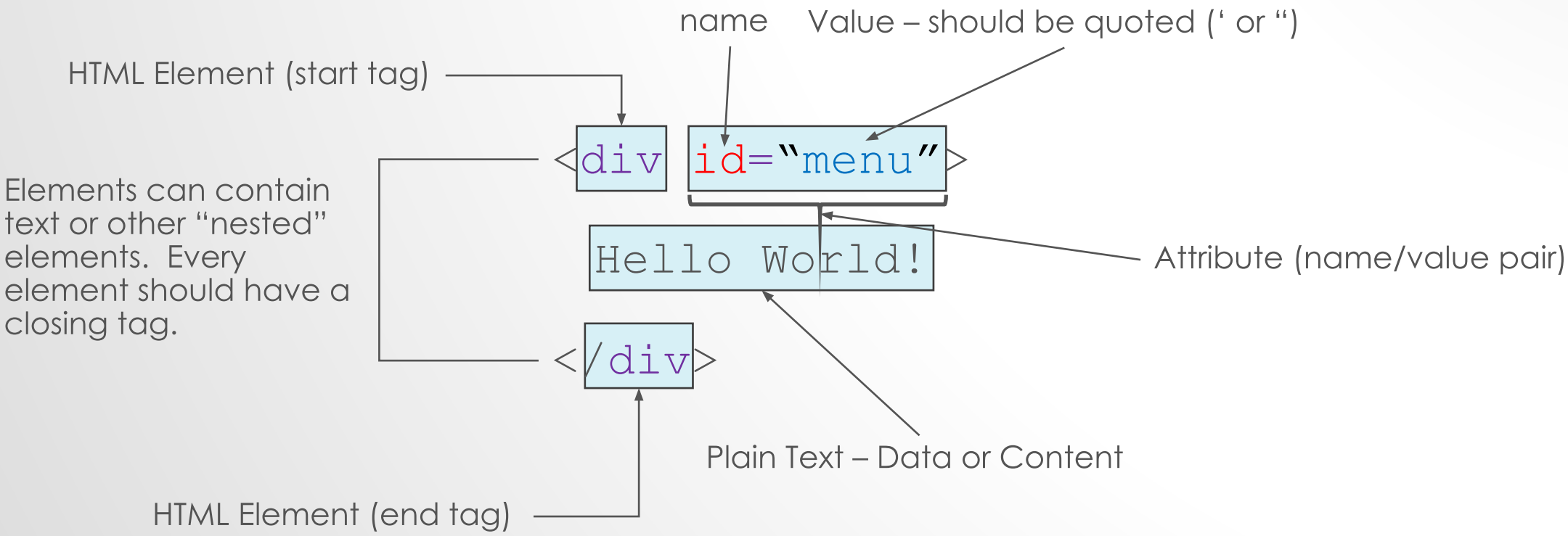
# WHAT IS HTML

- It is a Markup Language – there are other kinds (XML, SGML)
- It is not a programming language
- A Markup Language is a structured way of defining data
- It is the primary building block of a web page
- It stands for Hypertext Markup Language

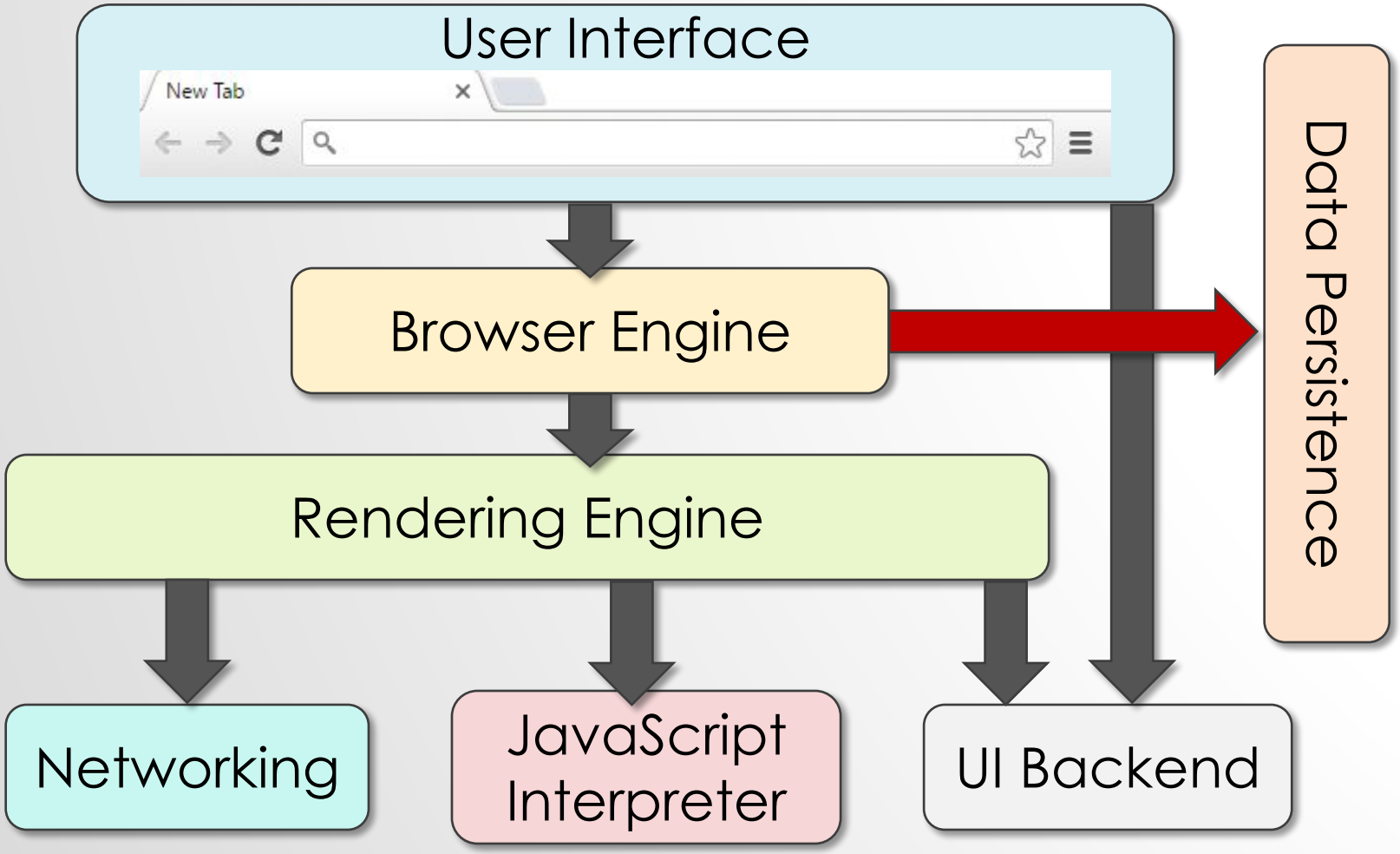


# ANATOMY OF A MARKUP TAG

- HTML documents contain **tags** and plain **text**
- Most tags are pre-defined and have special meaning (but not all)

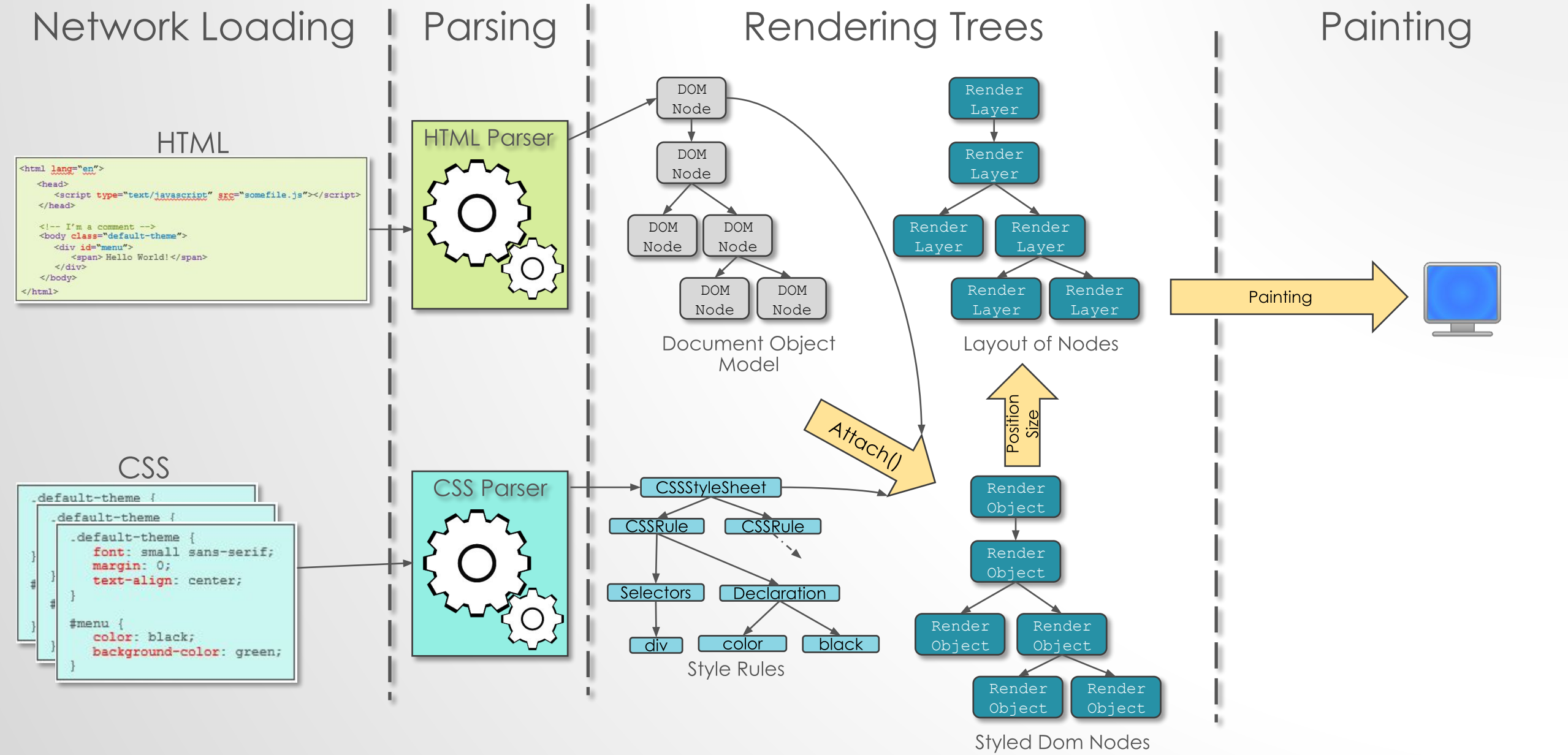


# ANATOMY OF A BROWSER

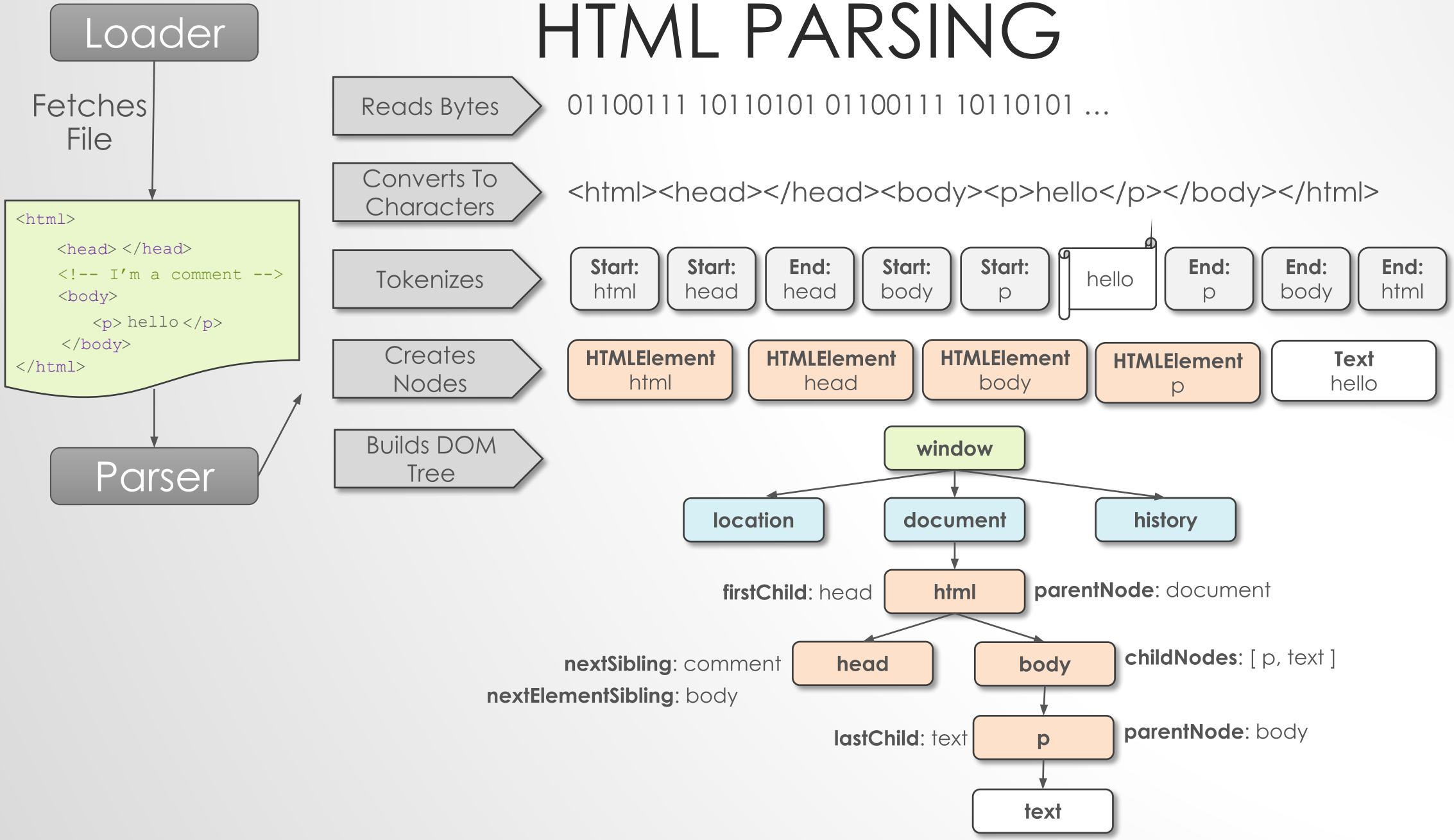



- Users Interact with UI
- UI actions are sent to BE which Queries and manipulates the rendering engine
- Displays the requested content on the screen by parsing HTML and CSS.
- Used for network calls, like HTTP requests; native to system.
- Parses and executes JavaScript
- Used for drawing basic widgets like combo boxes and windows. Exposes generic interface that is not platform specific. Uses OS for interface methods.
- Browsers store data, like cookies, to the hard drive

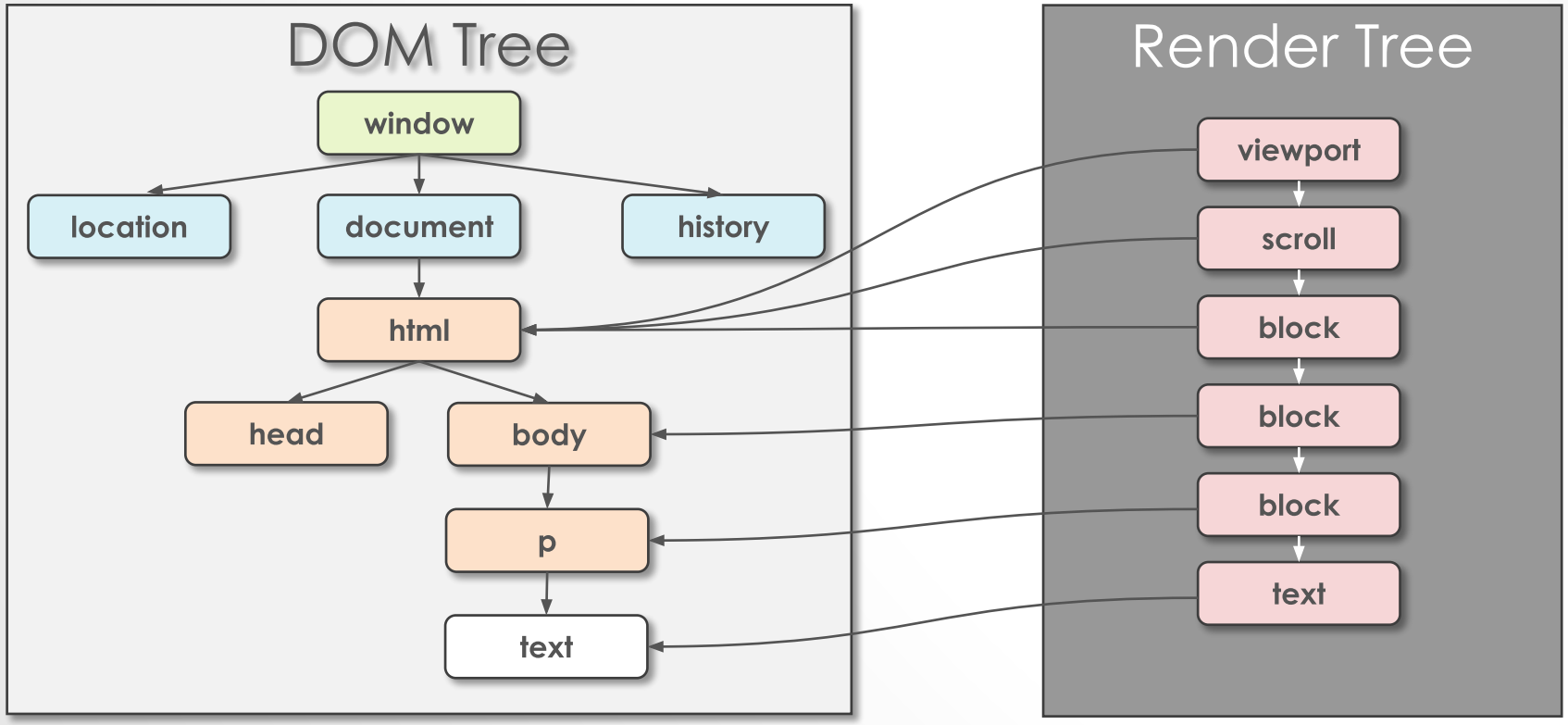
# BROWSER MAIN FLOW



# HTML PARSING

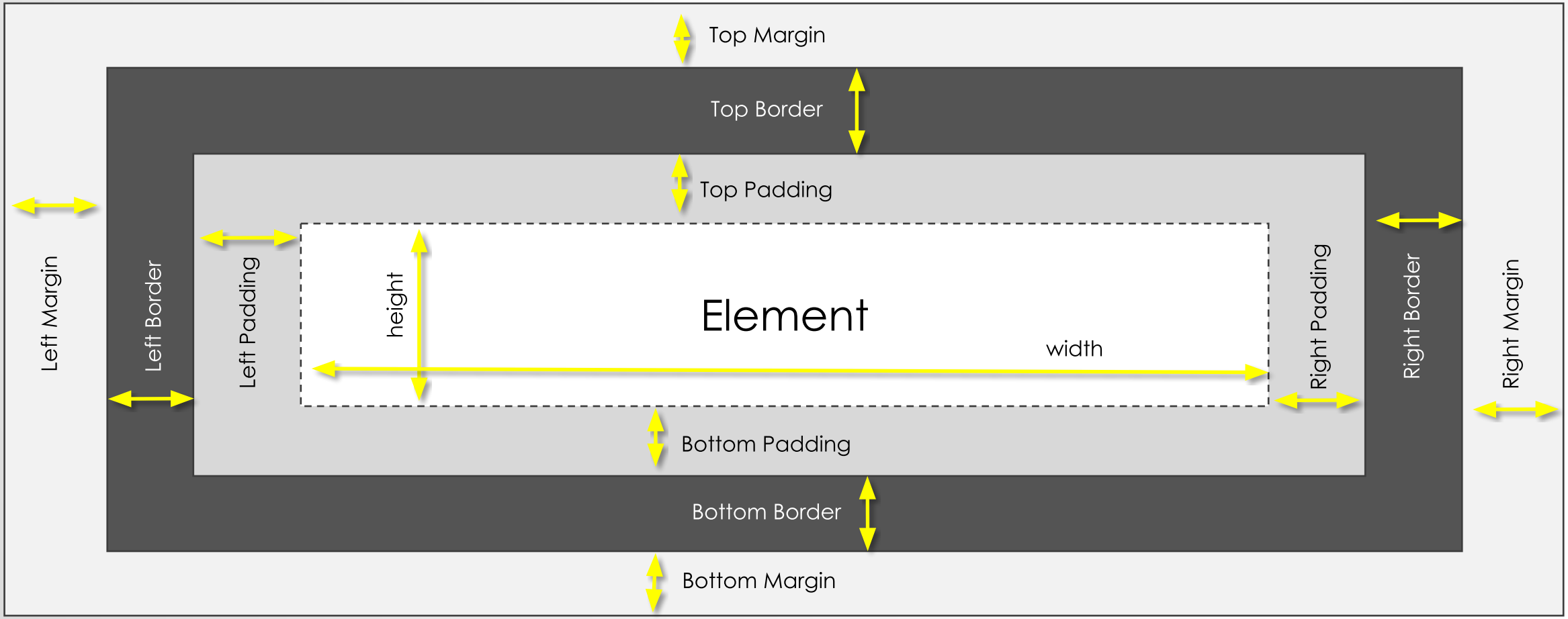


# THE RENDER TREE



# CSS BOX MODEL

- All HTML elements can be considered as boxes.
- In CSS, "box model" is used when talking about design and layout
- The CSS box model is a box that wraps around every HTML element



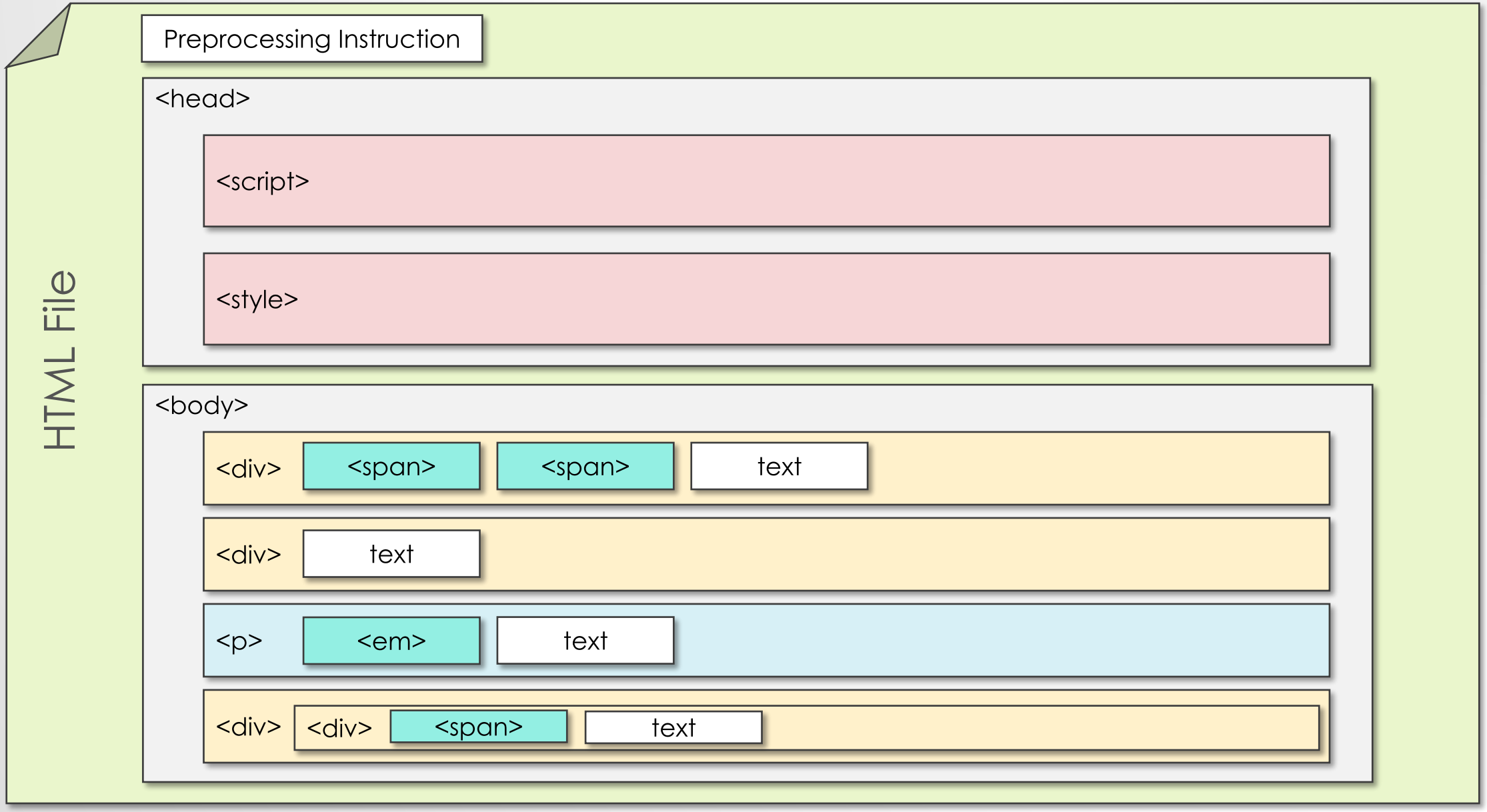
# HTML DEFAULT ELEMENT DISPLAY

- Every HTML element has a default display depending upon it's type
- The default display for most elements is block or inline (or a mixture of them)
- Block-Level Elements – always start on a new line and take up the full width available
  - 
  - <h1> - <h6>
  - <p>
  - <form>
  - Can be containers of other block or inline elements
  - Can adjust the width and height
- Inline Elements – does not start on a new line and only takes up as much width as necessary
  - 
  - <a>
  - <img>
  - inline elements may contain only data and other inline elements.
  - Treated as part of the text flow (left -> right, top -> bottom)
  - Cannot adjust the width or height

# HTML WEB PAGE

```
<!doctype html>
<html>
    <head>
        <script>
        ...
        </script>
        <style>
        ...
        </style>
    </head>

    <!-- I'm a comment -->
    <body>
        <div>
            <span>span-text-1</span><span>span-text-2</span>
            div-text-1
        </div>
        <span>span-text-3</span><em>em-text-1</em><span>span-text-4</span>
        <p><em>em-text-2</em></p>
        <div><div><span>span-text-5</span></div></div>
    </body>
</html>
```

# WEB PAGE STRUCTURE



# BLOCK-LEVEL ELEMENTS

- **Basic**: body, h1, h2, h3, h4, h5, h6, hr, p
- **Formatting**: blockquote, pre, progess
- **Forms and Input**: button, fieldset, form, output
- **Grouping**: div
- **Images**: canvas, figcaption, figure, map
- **Media**: video
- **Links**: nav
- **Lists**: dd, dl, dt, ul, ol, li
- **Tables**: caption, table, tbody, td, th, tr, thead, tfoot, col, colgroup
- **Styles and Semantics**: address, article, aside, div, header, footer, main, section
- **Programming**: embed, object, noscript

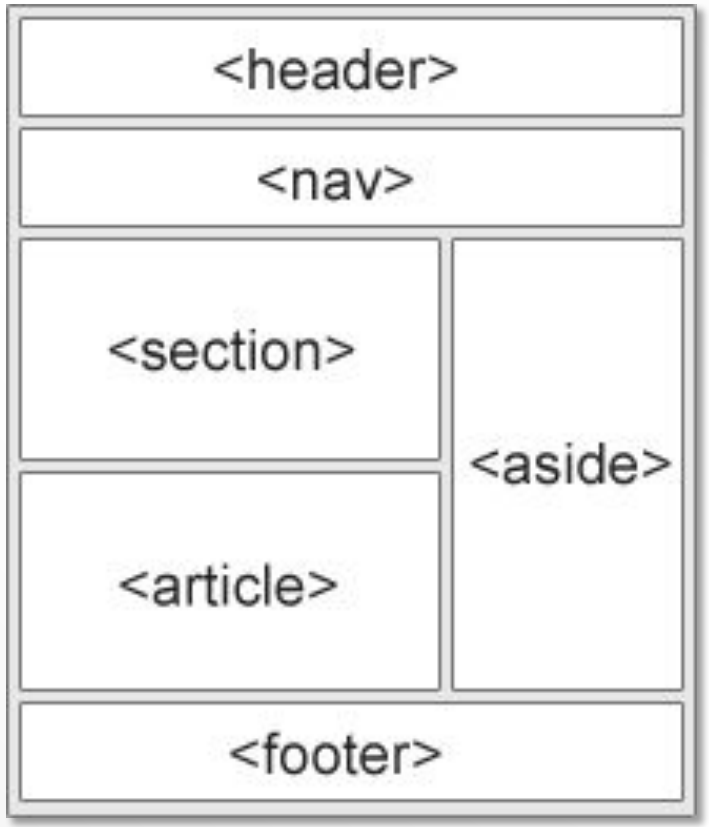
# INLINE ELEMENTS

- **Formatting**: abbr, address, b, cite, code, del, dfn, em, I, ins, kbd, mark, meter, q, samp, small, strong, sub, sup, time, u, var
- **Forms and Input**: datalist, input, label, legend, optgroup, option, select, textarea
- **Grouping**: span
- **Frames**: iframe
- **Images**: area, img
- **Media**: source
- **Links**: a, link
- **Styles and Semantics**: details, summary

# HTML LAYOUT

- Past: used to separate sections with div tags (e.g. )
- Present: HTML5 introduced semantic elements

- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>




http://www.w3schools.com/html/html_layout.asp

# CLIENT-SIDE WEB TECHNOLOGIES

## Basics

- HTML
  - Describes and define the **content** of a web page in a structured manner
- CSS
  - Describes the **appearance** of web content

## Scripting

- JavaScript
  - Provides **interactive** web site functionality
  - The programming language of the browser

## Graphics

- SVG (Scalable Vector Graphics) https://fortawesome.github.io/Font-Awesome/ http://www.w3schools.com/svg/svg_examples.asp
  - Are HTML elements that all you to dynamic draw images on a web page
  - Can change size and appearance dynamically
- WebGL (similar to OpenGL)
  - Allows for 2d or 3d graphics to be applied to the <canvas> element
  - JavaScript is the programming API

# TUTORIAL SITES

- **W3 Schools** (skip CSS stuff)
  - Tutorial: http://www.w3schools.com/html/default.asp
  - All Tags: http://www.w3schools.com/tags/tag_nav.asp
- **Code Academy**
  - HTML up to unit 4: https://www.codecademy.com
- **Tutorials Point**
  - Ref & Tut: http://www.tutorialspoint.com/html/html_tags_reference.htm
- **Practice Sites**:
  - Code Pen: https://codepen.io/pen/
  - JSFiddle: https://jsfiddle.net/
  - Plunker: https://plnkr.co/edit/?p=catalogue
- **Reference** (also see W3 schools above)
  - Mozilla Dev Network: https://developer.mozilla.org/en-US/docs/Web/HTML
- **Homework**
  - Create a web page that demonstrates every tag listed in W3 Schools

# CHROME TUTORIAL