

# Web Dev: Node.js Stack

By Naimul Islam Mehedi

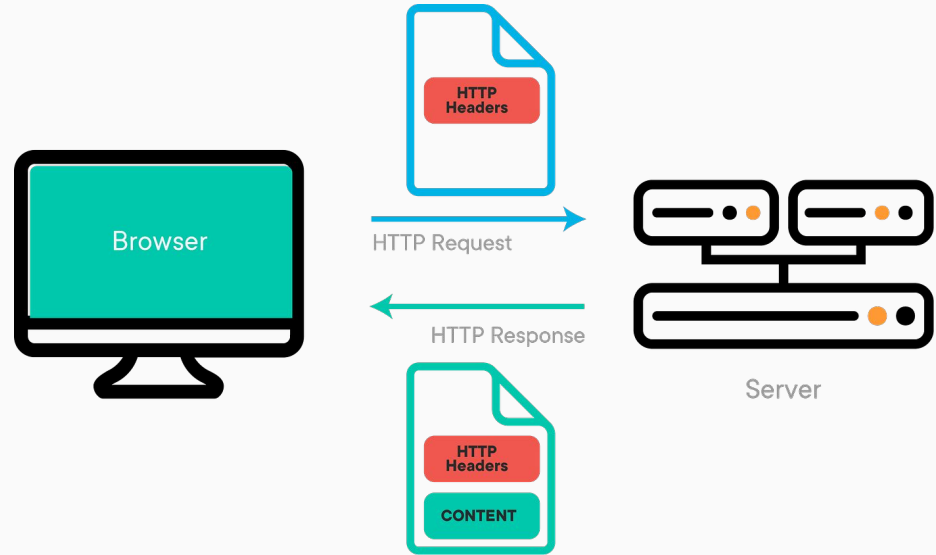


# **Module 1: Web Foundations - The Static Page**

# 1. How the Web Works

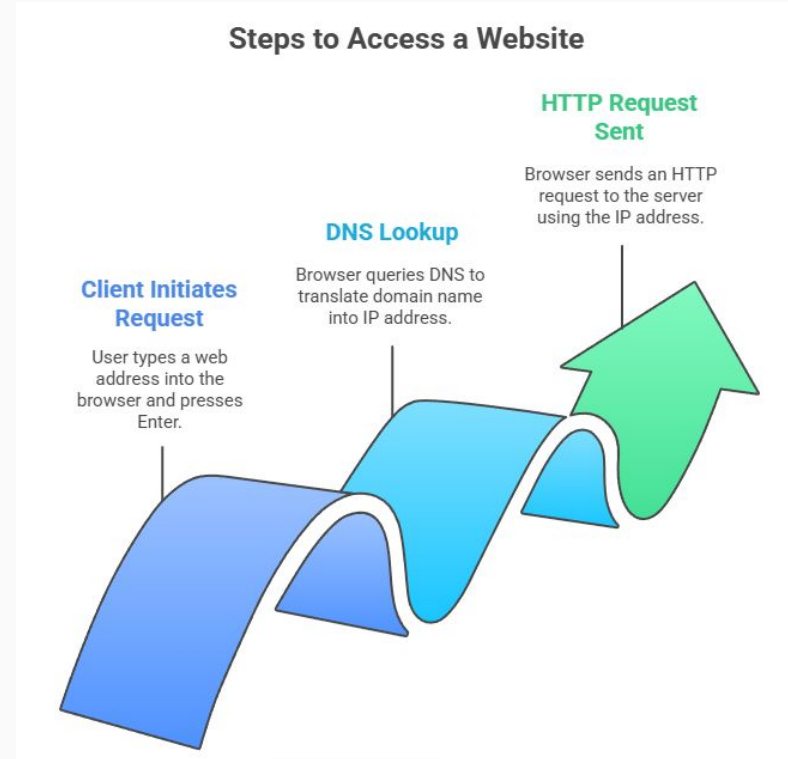
# How the Web Works?

Think of it like **ordering a pizza**: you (the client) make a request, and the restaurant (the server) prepares and sends your order back to you.



# The Client and the Request

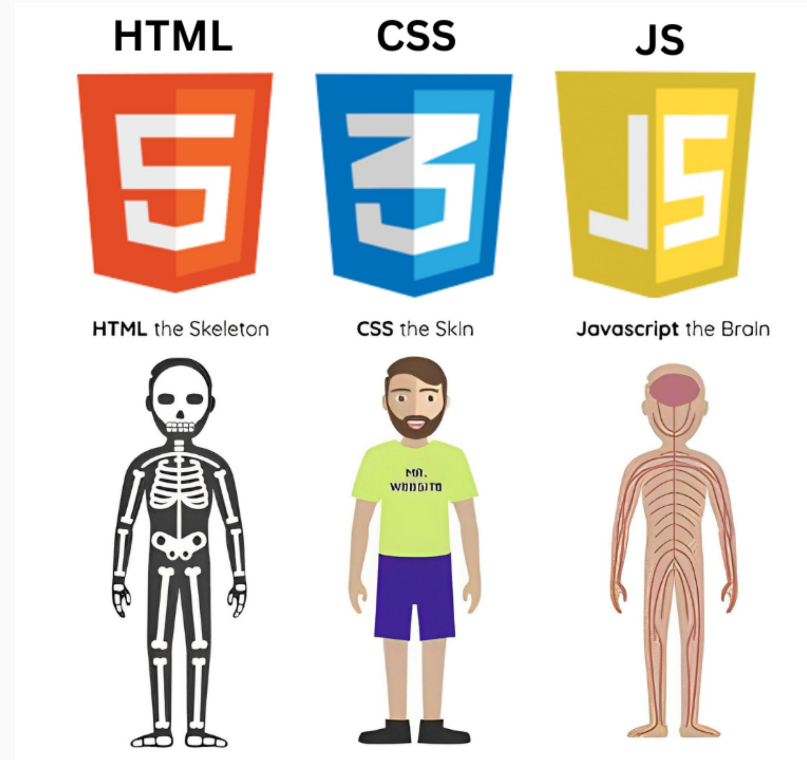
When you type a web address (like [www.google.com](http://www.google.com)) and hit Enter, you're initiating a request.



# The Browser Renders the Page

Your browser receives the bundle of files from the server and begins the final step: assembling and displaying the webpage. It interprets three main types of files:

- HTML (Hypertext Markup Language)
- CSS (Cascading Style Sheets)
- JavaScript (JS)



## 2. HTML - Core Concepts (Structure, Content Tags, Lists, Tables)

# HTML - Core Concepts (Structure, Content Tags, Lists, Tables)

HTML (HyperText Markup Language) is the standard language used to create and structure the content of a webpage. ***Think of it as the skeleton that gives a website its form.*** Every other technology, like CSS for styling and JavaScript for interactivity, is built on top of the HTML structure.





# Basic HTML Document Structure

- **<!DOCTYPE html>**: This is always the first line. It's a declaration that tells the browser the document is an HTML5 page.
- **<html>**: The root element. It wraps all the content on the entire page.
- **<head>**: This section contains meta-information about the page that isn't displayed in the main browser window. This includes the page <title>, links to stylesheets (CSS), and other metadata.
- **<body>**: This element contains all the content that is visible to the user, such as text, images, links, tables, and lists.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width,
6          initial-scale=1.0">
7      <title>Document</title>
8  </head>
9  <body>
10
11 </body>
12 </html>
```

# Essential Content Tags

- **Headings (<h1> to <h6>):** Used to define titles and subtitles. <h1> is the most important heading, and <h6> is the least. Search engines use these headings to understand the structure and importance of your content.
- **Paragraphs (<p>):** Used for blocks of text. Browsers automatically add some vertical space before and after each paragraph.
- **Emphasis (<strong>, <em>):** Use <strong> to indicate text with strong importance (typically displayed as bold). Use <em> to emphasize text (typically displayed as italic).

```
8  <body>
9  <h1>Main Heading for the Page</h1>
10 <p>
11   This is an introductory paragraph. It explains the purpose of the page.
12   You can use <strong>strong emphasis</strong> for key terms that are
13   critical to understand.
14 </p>
15
16 <h2>A Secondary Heading</h2>
17 <p>
18   This paragraph provides more detail. Sometimes you just need to add a
19   little <em>emphasis</em> to make a point.
20 </p>
21 </body>
22 </html>
```

# Lists

Lists are essential for organizing information. HTML provides two primary types.

- **Unordered List (<ul>):** A bulleted list. Use this when the order of the items does not matter.
- **Ordered List (<ol>):** A numbered list. Use this for step-by-step instructions or when the sequence is important.
- **List Item (<li>):** Every item inside both <ul> and <ol> tags must be wrapped in an <li> tag.

```
21
22  <!-- Unordered List -->
23  <ul>
24    <li>Cyber Security Services</li>
25    <li>AI Solutions</li>
26    <li>Web Development</li>
27  </ul>
28
29  <!-- Ordered List -->
30  <ol>
31    <li>Analyze requirements</li>
32    <li>Develop the core logic</li>
33    <li>Deploy the application</li>
34  </ol>
35 </body>
36 </html>
37
```

# Tables

Tables are used to display data in a structured, grid-like format of rows and columns.

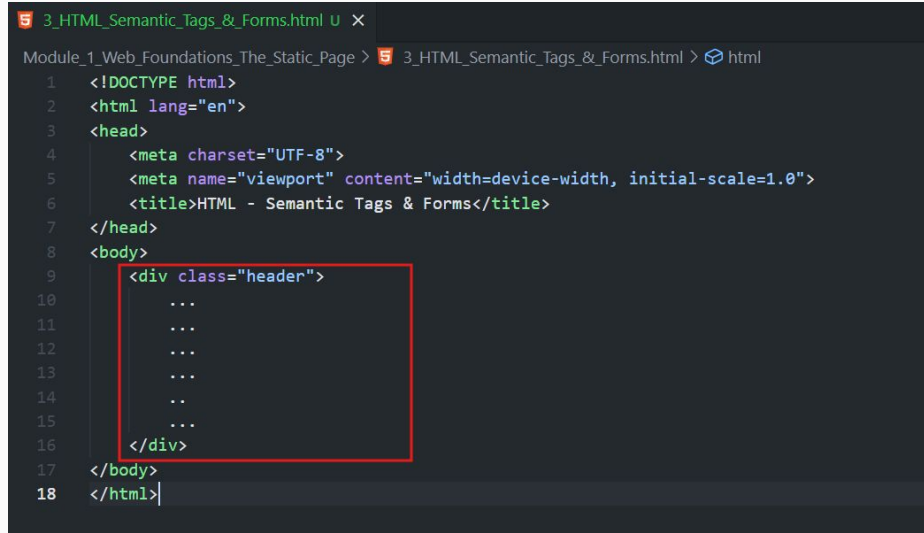
- **<table>**: The container for the entire table.
- **<tr>**: Defines a table row.
- **<th>**: Defines a table header cell. These are typically the titles for columns and are displayed as bold and centered.
- **<td>**: Defines a table data cell, which contains the main content.

```
36 <table>
37   <tr>
38     <th>Service Tier</th>
39     <th>Features</th>
40     <th>Price</th>
41   </tr>
42   <tr>
43     <td>Basic Threat Monitoring</td>
44     <td>Log analysis and alerts.</td>
45     <td>$500/month</td>
46   </tr>
47   <tr>
48     <td>Advanced Threat Hunting</td>
49     <td>AI-powered anomaly detection.</td>
50     <td>$1,500/month</td>
51   </tr>
52 </table>
53 </body>
54 </html>
55
```

# **3. HTML - Core Concepts (Structure, Content Tags, Lists, Tables)**

# HTML - Semantic Tags & Forms

In the early days of the web, developers often used generic `<div>` tags for almost everything, relying on class or id attributes to describe the content's purpose (e.g., `<div class="header">`). Semantic HTML introduces tags that have a clear, descriptive meaning for both the browser and the developer.



```
3_HTML_Semantic_Tags_&_Forms.html U x
Module_1_Web_Foundations_The_Static_Page > 3_HTML_Semantic_Tags_&_Forms.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>HTML - Semantic Tags & Forms</title>
7  </head>
8  <body>
9      <div class="header">
10         ...
11         ...
12         ...
13         ...
14         ..
15         ...
16     </div>
17 </body>
18 </html>
```

# Common Semantic Tags

- **<header>** : Defines the header for a document or a section. It often contains a logo, navigation (<nav>), and other introductory content.
- **<nav>** : Contains navigation links, such as the main menu of a site.

```
3_HTML_Semantic_Tags_&_Forms.html U X
Module_1_Web_Foundations_The_Static_Page > 3_HTML_Semantic_Tags_&_Forms.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>HTML - Semantic Tags & Forms</title>
7  </head>
8  <body>
9      <header>
10         <h1>My Tech Blog</h1>
11         <nav>
12             <ul>
13                 <li><a href="/">Home</a></li>
14                 <li><a href="/web">Web</a></li>
15                 <li><a href="/ai">AI</a></li>
16                 <li><a href="/cyber-security">Cyber Security</a></li>
17             </ul>
18         </nav>
19     </header>
20 </body>
21 </html>
```

# Common Semantic Tags

- **<main>**: Specifies the primary, unique content of the document. There should only be one <main> element per page.
- **<article>**: Represents a self-contained composition that is independently distributable or reusable (e.g., a forum post, a magazine or newspaper article, or a blog entry).
- **<section>**: Represents a standalone section of a document that doesn't have a more specific semantic element to represent it. It's a thematic grouping of content, typically with a heading.
- **<aside>**: Represents content that is tangentially related to the content around it. Often used for sidebars or callout boxes.

```
11     <nav>
12         <ul>
15             <li><a href="/ai">AI</a></li>
16             <li><a href="/cyber-security">Cyber Security</a></li>
17         </ul>
18     </nav>
19 </header>
20
21     <main>
22         <article>
23             <h2>The Rise of AI</h2>
24             <p>Artificial intelligence is transforming our world...</p>
25             <section>
26                 <h3>Machine Learning</h3>
27                 <p>A subset of AI, machine learning involves...</p>
28             </section>
29         </article>
30
31         <aside>
32             <h3>About the Author</h3>
33             <p>John Doe is a tech enthusiast with a passion for AI.</p>
34         </aside>
35     </main>
36
37 </body>
38 </html>
```



# Common Semantic Tags

- **<footer>**: Defines the footer for a document or section. It usually contains authorship information, copyright data, or links to related documents.
- **<figure> & <figcaption>**: Used to group media content (like an <img> or a code snippet) with a caption (<figcaption>).

```
31     <aside>
32       <h3>About the Author</h3>
33       <p>John Doe is a tech enthusiast with a passion for AI.</p>
34     </aside>
35   </main>
36
37   <footer>
38     <p>© 2025 My Tech Blog. All rights reserved.</p>
39   </footer>
40 </body>
41 </html>
```

# HTML Forms

Forms are used to collect data from users, such as for a contact page, a login screen, or a search bar.

## Key <form> Attributes

- **action:** The URL of the page or script on the server that will process the form data.
- **method:** The HTTP method to use when submitting the form.
- **GET:** Appends the form data to the URL. Good for non-sensitive data like search queries.
- **POST:** Sends the form data in the HTTP request body. Used for sensitive data (like passwords) or large amounts of data.

```
31     <aside>
32         <h3>About the Author</h3>
33         <p>John Doe is a tech enthusiast with
           a passion for AI.</p>
34     </aside>
35 </main>
36
37 <!-- Form -->
38 <form action="/submit-form" method="POST">
39     ...
40 </form>
41
42
43 <footer>
44     <p>© 2025 My Tech Blog. All rights
       reserved.</p>
45 </footer>
46 </body>
47 </html>
```

# HTML Forms

- **<label>**: Defines a label for an `<input>` element. It improves accessibility by linking the text label to the input field.
- **<input>**: The most-used form element. The type attribute specifies how it appears and behaves.
- **<textarea>**: Defines a multi-line text input control.
- **<select>**: Creates a dropdown list. Each item is defined by an `<option>` tag.
- **<button>**: Defines a clickable button that can be used to submit the form or trigger a script.

```

37 <!-- Form -->
38 <form action="/submit-form" method="POST">
39   <h3>Contact us;</h3>
40   <div>
41     <label for="userName">Name:</label>
42     <input type="text" id="userName" name="user_name" placeholder="Enter your full name" required>
43   </div>
44
45   <div>
46     <label for="userEmail">Email:</label>
47     <input type="email" id="userEmail" name="user_email" required>
48   </div>
49
50   <div>
51     <p>Inquiry Type:</p>
52     <input type="radio" id="inquiryGeneral" name="inquiry_type" value="general">
53     <label for="inquiryGeneral">General</label><br>
54     <input type="radio" id="inquirySupport" name="inquiry_type" value="support">
55     <label for="inquirySupport">Support</label>
56   </div>
57

```

```
50 <div>
51   <p>Inquiry Type:</p>
52   <input type="radio" id="inquiryGeneral" name="inquiry_type" value="general">
53   <label for="inquiryGeneral">General</label><br>
54   <input type="radio" id="inquirySupport" name="inquiry_type" value="support">
55   <label for="inquirySupport">Support</label>
56 </div>
```

```
57
58 <div>
59   <label for="message">Your Message:</label>
60   <textarea id="message" name="user_message" rows="5"></textarea>
61 </div>
62
63 <div>
64   <input type="checkbox" id="newsletter" name="newsletter_signup" value="true">
65   <label for="newsletter">Sign up for our newsletter</label>
66 </div>
67
68 <button type="submit">Submit</button>
69
70
```

## **4. CSS - Core Concepts (Selectors, Box Model, Colors, Typography)**

# CSS Selectors

Selectors are the foundation of CSS. The most common selectors target elements by their tag name, their class, or their unique ID.

- **Tag Selector:** Selects all elements with a specific HTML tag (e.g., p, h1, div).
- **Class Selector:** Selects all elements that have a given class attribute. You define it with a dot (.) followed by the class name (e.g., .highlight). An element can have multiple classes.
- **ID Selector:** Selects a single, unique element that has a given id attribute. You define it with a hash (#) followed by the ID name (e.g., #header). An ID must be unique on a page.

## HTML:

```
<div id="intro">
  <h1>Main Title</h1>
  <p class="highlight">This paragraph is important.</p>
  <p>This is a regular paragraph.</p>
</div>
```

## CSS:

```
/* Tag selector: styles all <p> elements */
p {
  color: #555;
}
```

```
/* Class selector: styles only elements with
class="highlight" */
.highlight {
  background-color: yellow;
  font-style: italic;
}
```

```
/* ID selector: styles the single element with id="intro" */
#intro {
  border: 1px solid #ccc;
}
```

# The Box Model

Selectors are the foundation of CSS. The most common selectors target elements by their tag name, their class, or their unique ID.

- **Tag Selector:** Selects all elements with a specific HTML tag (e.g., p, h1, div).
- **Class Selector:** Selects all elements that have a given class attribute. You define it with a dot (.) followed by the class name (e.g., .highlight). An element can have multiple classes.
- **ID Selector:** Selects a single, unique element that has a given id attribute. You define it with a hash (#) followed by the ID name (e.g., #header). An ID must be unique on a page.

## HTML:

```
<div id="intro">
  <h1>Main Title</h1>
  <p class="highlight">This paragraph is important.</p>
  <p>This is a regular paragraph.</p>
</div>
```

## CSS:

**/\* Tag selector: styles all <p> elements \*/**

```
p {
  color: #555;
}
```

**/\* Class selector: styles only elements with class="highlight" \*/**

```
.highlight {
  background-color: yellow;
  font-style: italic;
}
```

**/\* ID selector: styles the single element with id="intro" \*/**

```
#intro {
  border: 1px solid #ccc;
}
```



# Thanks!

Contact us:

Your Company  
123 Your Street  
Your City, ST 12345

[no\\_reply@example.com](mailto:no_reply@example.com)

[www.example.com](http://www.example.com)

