**Notes**

A **semantic element** tells you *what it is* — not just how it looks.

* **Accessibility**: Screen readers and assistive tech can navigate content better.
* **SEO**: Search engines understand your content structure more clearly.
* **Maintainability**: Easier to read, write, and update code.
* **Consistency**: Helps other developers (and future-you) understand what each part of the page does.

The <fieldset> element in HTML is used to group related elements within a form, typically form controls like input fields, radio buttons, checkboxes, and select menus. It provides a way to organize the form into sections that are semantically meaningful. This improves both the visual structure and accessibility of a form. Here's what it is used for:

**Main Uses of <fieldset>:**

1. **Grouping Form Elements**: The primary purpose is to group related form controls together, such as all the options for a particular part of the form (e.g., selecting a pizza size or choosing toppings).
2. **Improved Accessibility**: When you group related inputs inside a <fieldset>, it helps screen readers and other assistive technologies understand the structure of the form better. The <legend> tag, which is often used with <fieldset>, provides a clear label for the group.
3. **Styling**: A <fieldset> often has a visual effect on the form. It typically renders as a box around the grouped form elements with a border. This makes it easier for users to identify and differentiate sections of a form.

The main use of the <article> element in HTML is to represent **a self-contained, independent piece of content** that can stand alone and potentially be reused or syndicated (like in RSS feeds).

1. **Use <article> for:**

* **Blog posts**
* **News articles**
* **Forum posts**
* **User-generated content (e.g., reviews or comments)**
* **Product descriptions (if standalone)**

1. **Key Characteristics:**

* **Standalone**: The content should make sense on its own without needing surrounding context.
* **Reusable**: It could be distributed or displayed independently.
* **Has its own heading**: Usually contains a heading (<h1>–<h6>) and related content.

**<header> — *Intro or navigational section***

**Purpose**: Defines **introductory content** or a **set of navigational links** for a page or a section.

1. **Use it for:**

* Page titles and taglines
* Logos and branding
* Navigation menus
* Section-specific intros

1. **Can be used:**

* Once for the whole page (site-wide header)
* Inside <article>, <section>, or <aside> for local headers

**<main> — *Primary page content***

**Purpose**: Contains the **main content unique to the page** (excluding headers, footers, sidebars, etc.).

1. **Use it for:**

* The central content that reflects the page’s main topic
* Only one <main> per page!

1. **Don’t use it for:**

* Site-wide navigation
* Headers/footers/repeating sidebars.

The :root pseudo-class in CSS targets the highest-level parent element in the document — usually the <html> element. It’s used primarily to define **global CSS variables** (also known as custom properties).

**Key Purposes of :root:**

1. **Global Variable Declaration:**

* You use :root to declare variables that can be accessed anywhere in your stylesheet.

1. **Higher Specificity Than html:**

* :root has a higher specificity than html, which helps ensure that variable definitions aren’t easily overridden by accident.

1. **Theming & Maintainability:**

* It makes your CSS more maintainable — change a value in one place and it updates everywhere.
* Ideal for dark/light themes or branding.
* **Reuse Across Components:**
* Makes it easy to reuse colors, font sizes, spacing units, etc., without hardcoding them throughout the stylesheet.