**Training Day 1: Introduction to Frontend, Semantic HTML, Forms & Tables**

### 💡 Part 1: Introduction to Frontend Development

#### ✅ What is Frontend?

* The **frontend** is the **client-side** of a website or application—what users **see and interact with**.
* Everything visual: layout, colors, fonts, buttons, forms, animations—comes under frontend.

#### ✅ Frontend vs Backend

| Frontend | Backend |
| --- | --- |
| Runs in the browser | Runs on the server |
| Built with HTML, CSS, JS | Built with languages like Python, Node.js, Java, etc. |
| Focuses on user interface (UI) & experience (UX) | Focuses on business logic, data storage, and APIs |
| Interacts with backend using HTTP (AJAX/Fetch) | Sends data to frontend using APIs |

#### ✅ Roles of a Frontend Developer

* Building responsive UI with HTML, CSS, and JS
* Making sites accessible and SEO-friendly
* Consuming APIs and displaying dynamic data
* Optimizing performance for fast load times
* Testing UI across browsers and devices

#### ✅ Tools Used

| Tool | Purpose |
| --- | --- |
| **VS Code** | Code editor |
| **Browser DevTools** | Inspect and debug HTML/CSS/JS |
| **Git & GitHub** | Version control and collaboration |
| **Package Managers** | Manage libraries (e.g., npm) |
| **Build Tools** | Bundle/minify assets (Webpack, Vite, etc.) |

#### ✅ Technologies Involved

1. **HTML** – Structure
2. **CSS** – Style
3. **JavaScript** – Interactivity
4. **Frameworks**: React, Angular, Vue
5. **Responsive Design**: Media Queries, Flexbox, Grid
6. **Accessibility & SEO** best practices

### 🧠 Part 2: Semantic HTML – Deep Dive

#### ✅ What is Semantic HTML?

* Semantic HTML uses **meaningful tags** to define the **purpose** of content.
* Helps browsers, developers, **screen readers**, and **search engines** understand the structure and flow.

#### ✅ Why Use Semantic HTML?

1. **Accessibility** – Screen readers can interpret and navigate your content better.
2. **SEO** – Search engines give priority to well-structured semantic documents.
3. **Maintainability** – Easier to read, debug, and collaborate.
4. **Standardization** – Follows W3C guidelines for clean markup.

#### 📁 Semantic Tags (and Their Meaning)

| Tag | Meaning |
| --- | --- |
| <header> | Intro or navigational info (logo, nav, etc.) |
| <nav> | Navigation links |
| <main> | Main content (only one per page) |
| <section> | Thematic group of content |
| <article> | Self-contained piece (blog, news post, etc.) |
| <aside> | Sidebar, ads, related info |
| <footer> | Copyright, links, contact info |
| <figure> & <figcaption> | Image with caption |

#### ❌ Non-Semantic Tags

* <div> and <span> are **generic containers** with no semantic meaning.
* Use them only when no semantic tag fits the need.

#### 🔮 Example with Explanation

<body>  
 <header>  
 <h1>Welcome to My Blog</h1>  
 <nav>  
 <a href="/">Home</a>  
 <a href="/about">About</a>  
 </nav>  
 </header>  
  
 <main>  
 <article>  
 <h2>Why I Love Coding</h2>  
 <p>Coding allows me to build amazing things...</p>  
 </article>  
  
 <article>  
 <h2>Learning Semantic HTML</h2>  
 <p>It makes your website cleaner, more accessible and search engine friendly.</p>  
 </article>  
 </main>  
  
 <footer>  
 <p>© 2025 Kishore’s Blog</p>  
 </footer>  
</body>

**Explanation:**

* header → Represents the top of the page (branding + navigation)
* nav → Contains the main navigation links
* main → Central content of the page
* article → Individual blog entries
* footer → Closing section with copyright

### 🔧 DevTools Activity

Open any website, right-click → “Inspect” → Elements tab. Try to find:

* How many <div>s are used?
* Are they using <main>, <header>, <section>?
* Which tags could be replaced with semantic ones?

### 🔢 Part 3: HTML Forms

#### ✅ What is a Form?

* HTML forms are used to **collect input** from the user.
* Form elements include text fields, checkboxes, radio buttons, submit buttons, etc.

#### 🔮 Basic Form Structure

<form action="/submit" method="post">  
 <label for="name">Name:</label>  
 <input type="text" id="name" name="username" required>  
  
 <label for="email">Email:</label>  
 <input type="email" id="email" name="useremail">  
  
 <input type="submit" value="Submit">  
</form>

**Attributes**:

* action: Where the form data is sent
* method: HTTP method (usually GET or POST)
* required: Makes a field mandatory

#### 🔹 Common Input Types

| Type | Use |
| --- | --- |
| text | Single-line text input |
| password | Password input |
| email | Validates email input |
| number | Numeric input |
| checkbox | Multiple selection |
| radio | Single selection from group |
| file | Upload a file |
| submit | Submit form |
| button | Custom button |

#### 💡 Grouping Fields with <fieldset> and <legend>

<form>  
 <fieldset>  
 <legend>Personal Info</legend>  
 <label>Age: <input type="number"></label>  
 </fieldset>  
</form>

#### 🔧 Hands-On Task

* Create a form for job application with name, email, gender (radio), skills (checkbox), file upload for resume.

### 📊 Part 4: HTML Tables

#### ✅ What is a Table?

* Tables are used to **display tabular data** like reports, schedules, etc.

#### ⚡ Basic Structure

<table border="1">  
 <thead>  
 <tr>  
 <th>Name</th>  
 <th>Age</th>  
 </tr>  
 </thead>  
 <tbody>  
 <tr>  
 <td>John</td>  
 <td>30</td>  
 </tr>  
 </tbody>  
</table>

#### 🔹 Tags Overview

| Tag | Purpose |
| --- | --- |
| <table> | Container for table data |
| <tr> | Table row |
| <th> | Table header cell |
| <td> | Table data cell |
| <thead> | Header section (optional) |
| <tbody> | Body section (optional) |
| <tfoot> | Footer section (optional) |

#### 📙 Features

* colspan and rowspan for merging cells
* scope="col" or scope="row" for accessibility

#### 🔧 Hands-On Task

* Create a table showing student scores in subjects (Name, Math, Science, English)

Let me know if you’d like this in PDF format or with additional slide visuals!