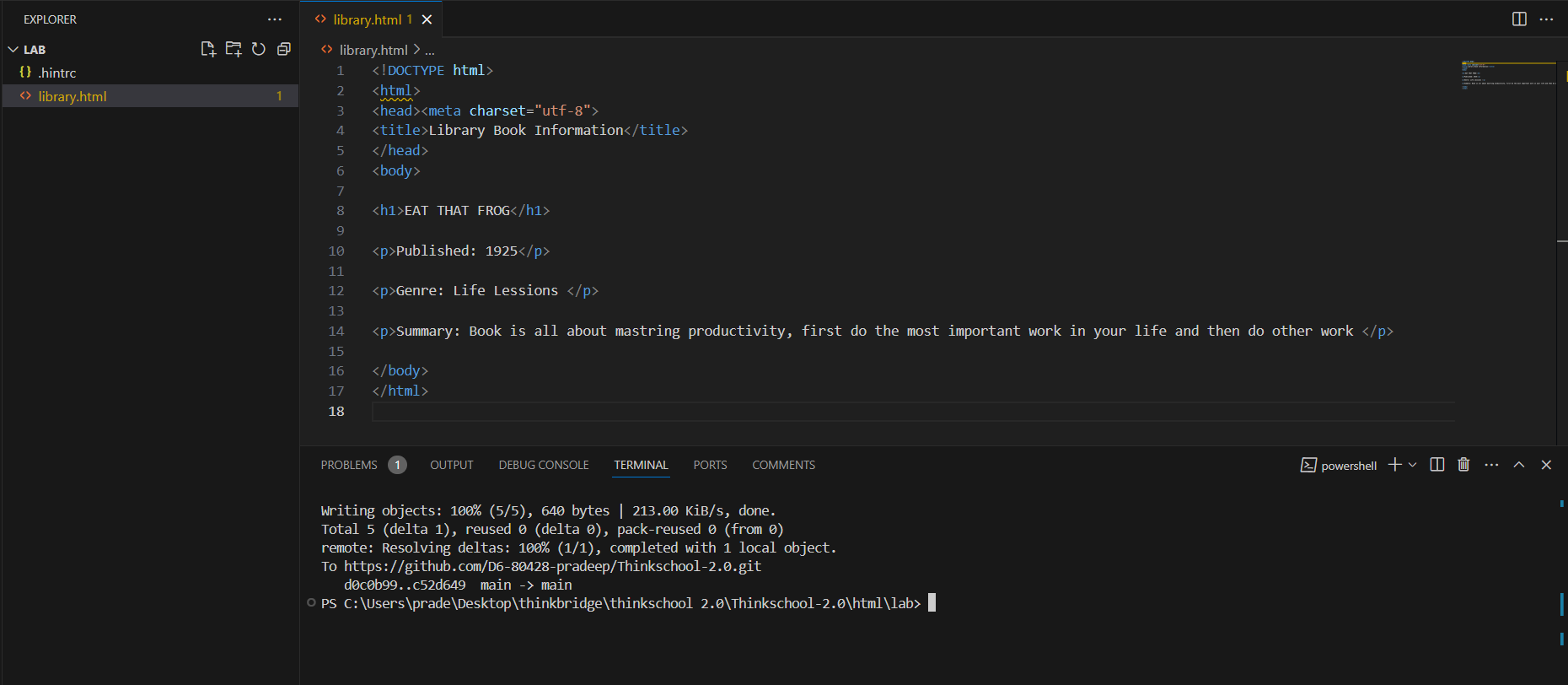
**Introduction And Basics: HTML**

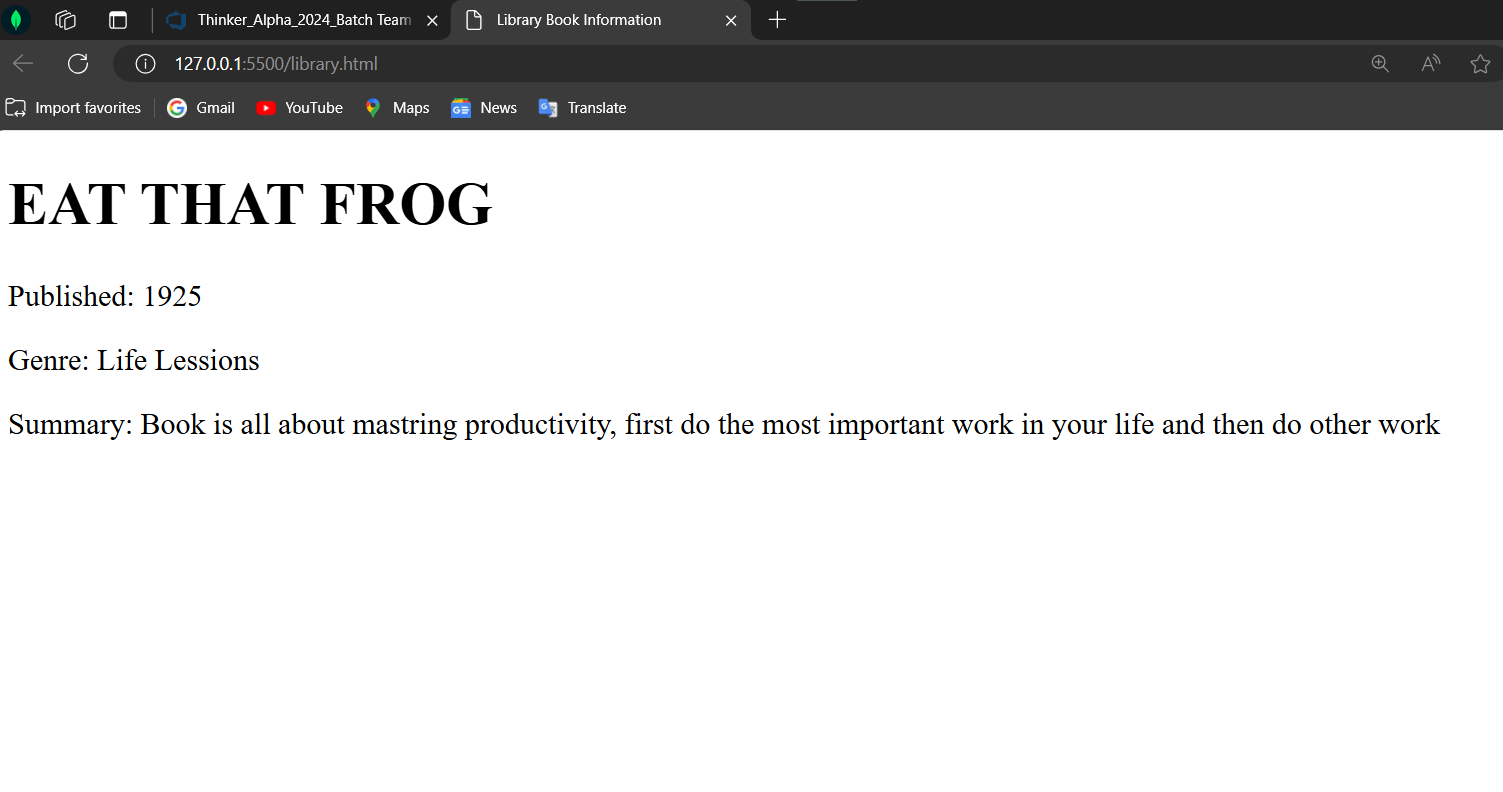
What is HTML?

HTML is like the skeleton of a webpage. It provides the structure, while CSS is like the skin and clothing that makes the webpage look visually appealing, and JavaScript is like the muscles that enable all the interactive elements.

Let’s see below a simple example based on library



Out put:



In above code:

<!DOCTYPE html>: This declaration helps with browser compatibility.

<html>: This is the root element of an HTML page.

<head>: This contains meta-information about the HTML document that isn’t displayed on the page.

<title>: This sets the title of the webpage, which appears in the browser tab.

<body>: This contains the content that is displayed to the user.

<h1>: This creates a top-level heading.

<p>: This creates a paragraph.

**The HTML element**: is everything from the start tag to the end tag:

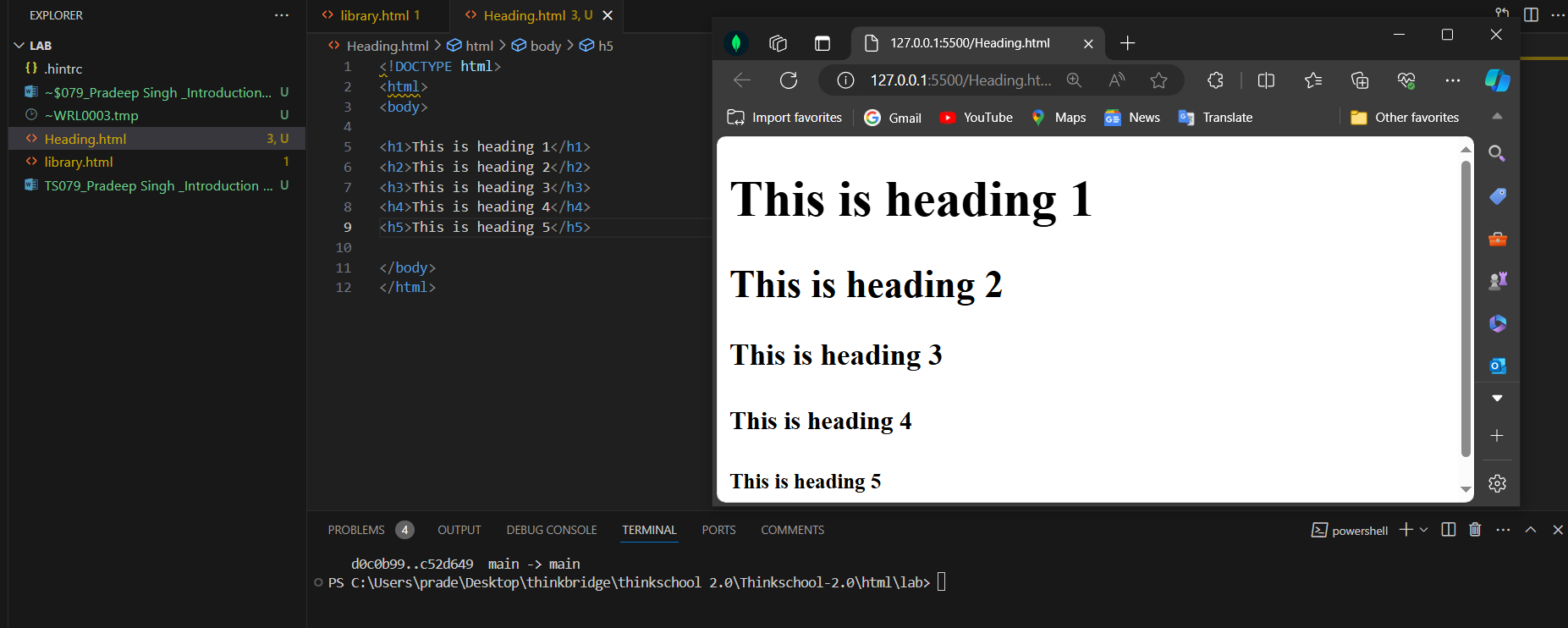
<tagname>: The start tag

Content goes here...: The content.

</tagname>: The end tag.

**Web Browsers:** A web browser is a software application that people use to access the internet and view web pages on their computer or mobile device. Examples of web browsers include Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge. Think of a web browser as a translator.

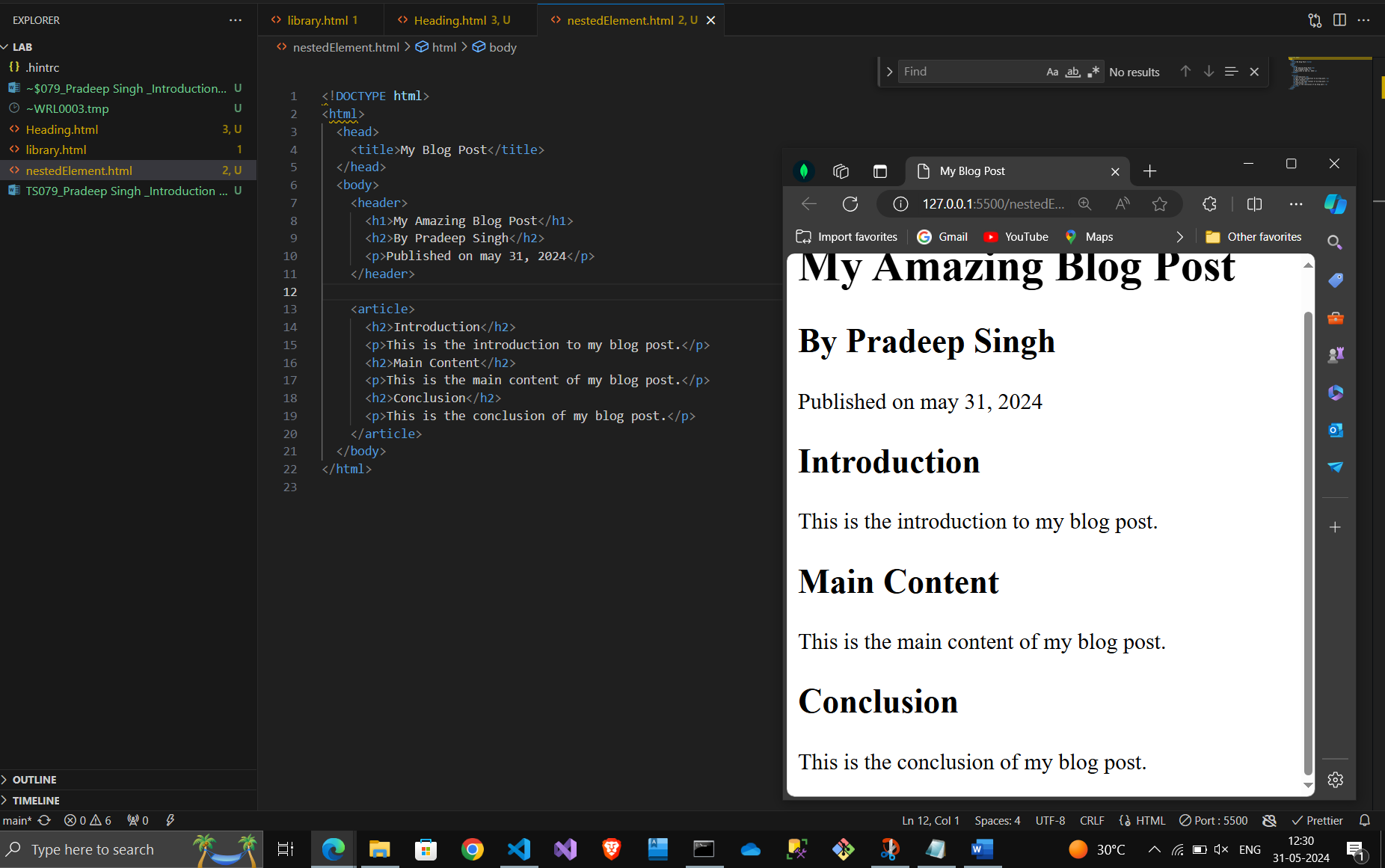
**HTML Headings:** HTML headings are like the chapter titles in a book. They give structure and meaning to the web content. Headings are defined with the <h1> to <h6> tags, with <h1> being the largest and <h6> the smallest.





**HTML Elements Nested:**

In HTML, elements can be nested, which means elements can contain other elements. This is similar to how boxes can be placed inside other boxes in real life.





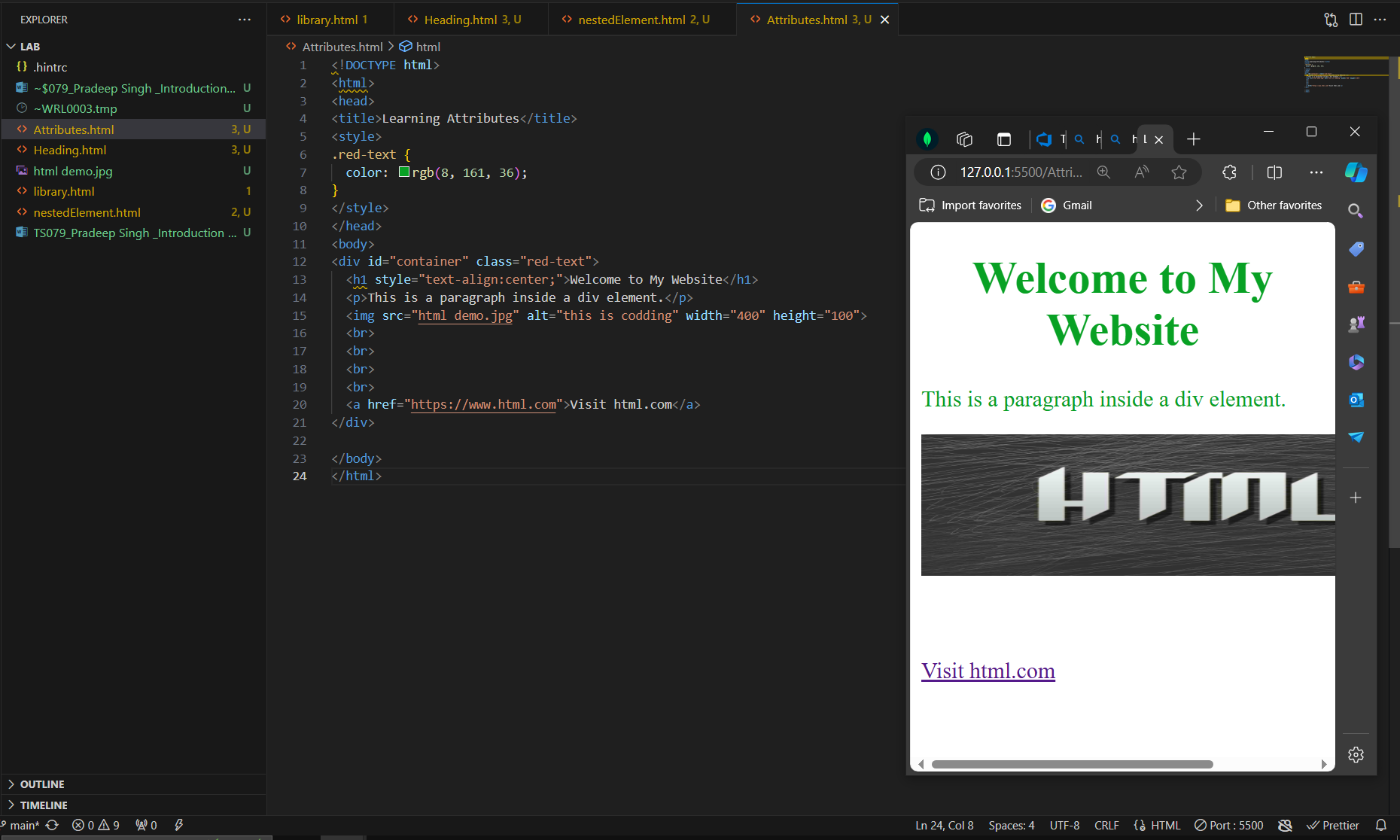
Above practical is showing nested element in html with output.

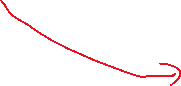
**Never Skip the End Tag:** In HTML, every opening tag should have a corresponding closing tag to ensure proper structure and avoid errors. For example, <p>This is a paragraph. </p>.

**Empty HTML Elements:** These are elements with no content and no closing tag, like <br> for a line break or <img> for an image.

**HTML is Not Case Sensitive:** HTML tags are not case sensitive, meaning <P> is the same as <p>. However, it’s a good practice to use lowercase for consistency and readability.

**HTML Attributes:** Attributes provide additional information about HTML elements. They are always specified in the start tag and usually come in name/value pairs like name="value". Here are some commonly used HTML attributes:





An HTML table is defined with the <table> tag. Each table row is defined with the <tr> tag. A table header is defined with the <th> tag, and a table data/cell is defined with the <td> tag.

Now, let’s discuss some common attributes used with HTML tables:

Border: This attribute specifies the width of the border around table cells. In HTML5, the use of this attribute is deprecated, and CSS should be used instead (e.g., style="border:1px solid black;").

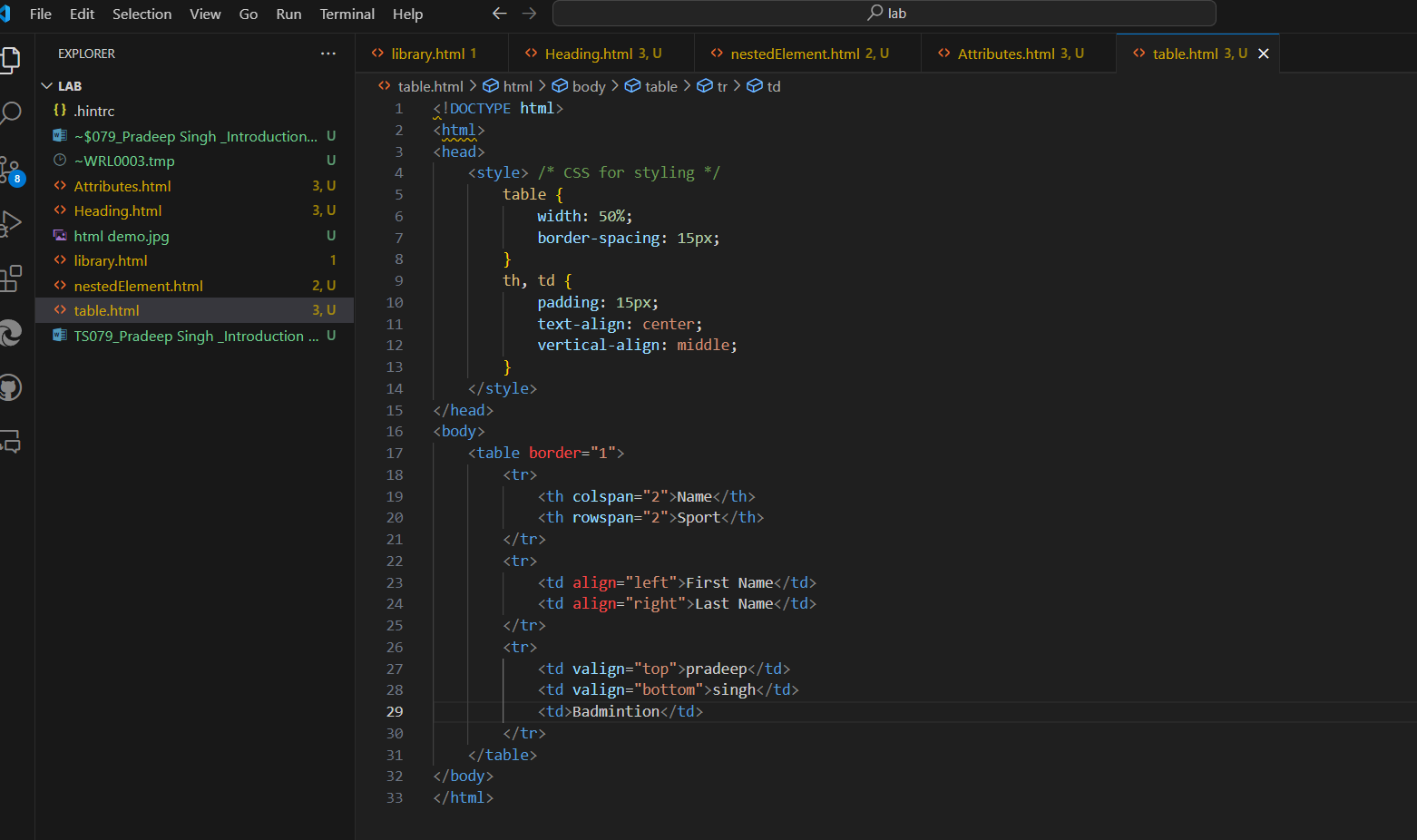
cellpadding: This attribute specifies the space between the cell wall and the cell content. In HTML5, the use of this attribute is deprecated, and CSS should be used instead (e.g., style="padding:15px;").

cellspacing: This attribute specifies the space between cells. In HTML5, the use of this attribute is deprecated, and CSS should be used instead (e.g., style="border-spacing:15px;").

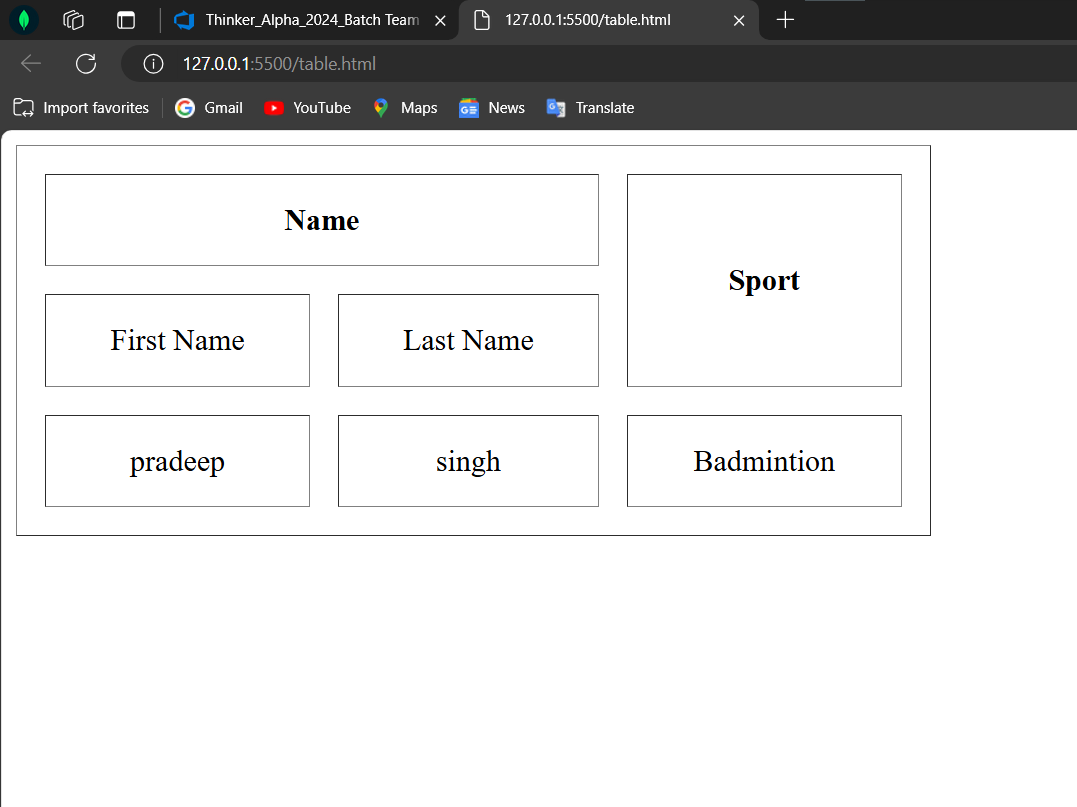
width and height: These attributes specify the width and height of the table. In HTML5, the use of these attributes is deprecated, and CSS should be used instead (e.g., style="width:50%; height:200px;").

colspan and rowspan: These attributes specify how many columns or rows a cell should span. They are used within <td> or <th> tags.

align and valign: These attributes specify the horizontal and vertical alignment of cell data. In HTML5, the use of these attributes is deprecated, and CSS should be used instead (e.g., style="text-align:center; vertical-align:middle;").

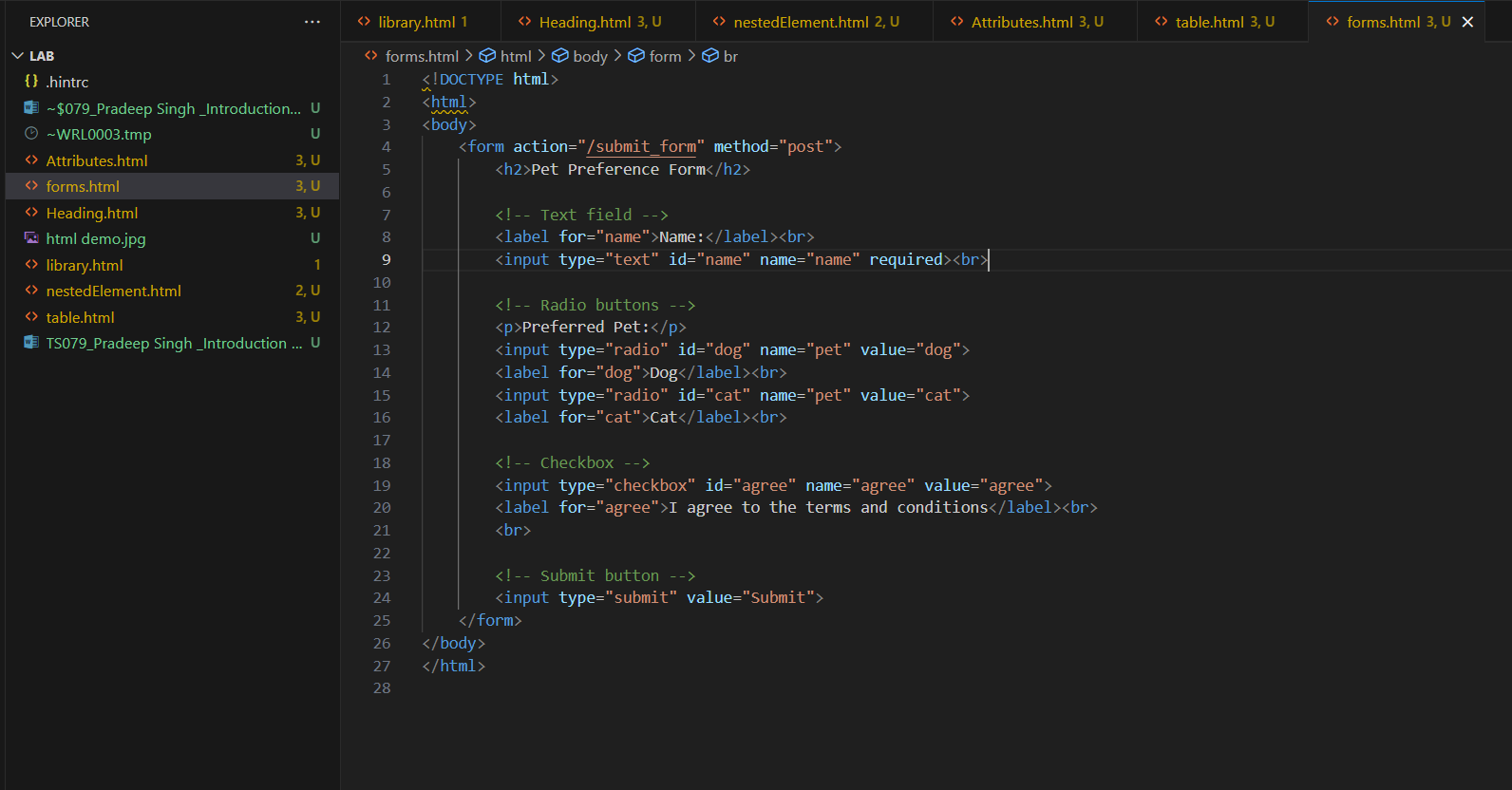


In above code I have created a Table which will look like as shown below.

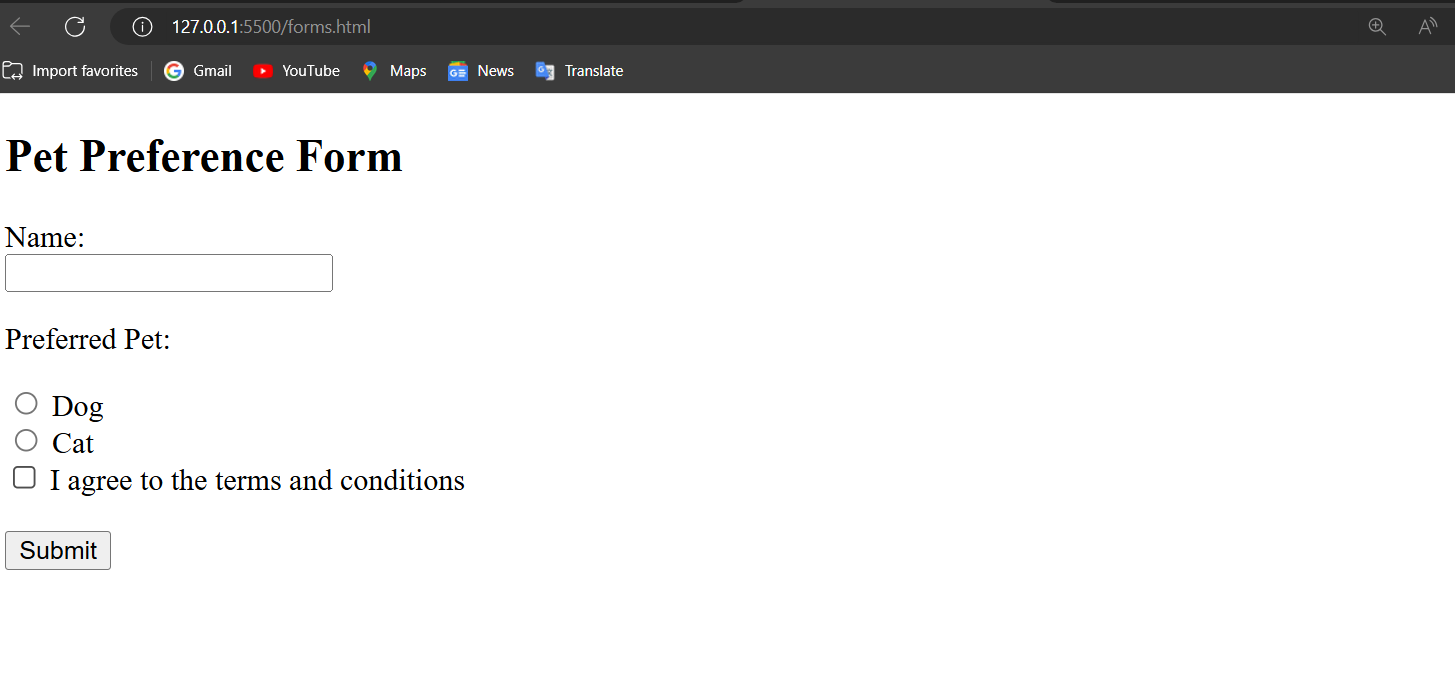


**HTML forms**: are used to collect user input. The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

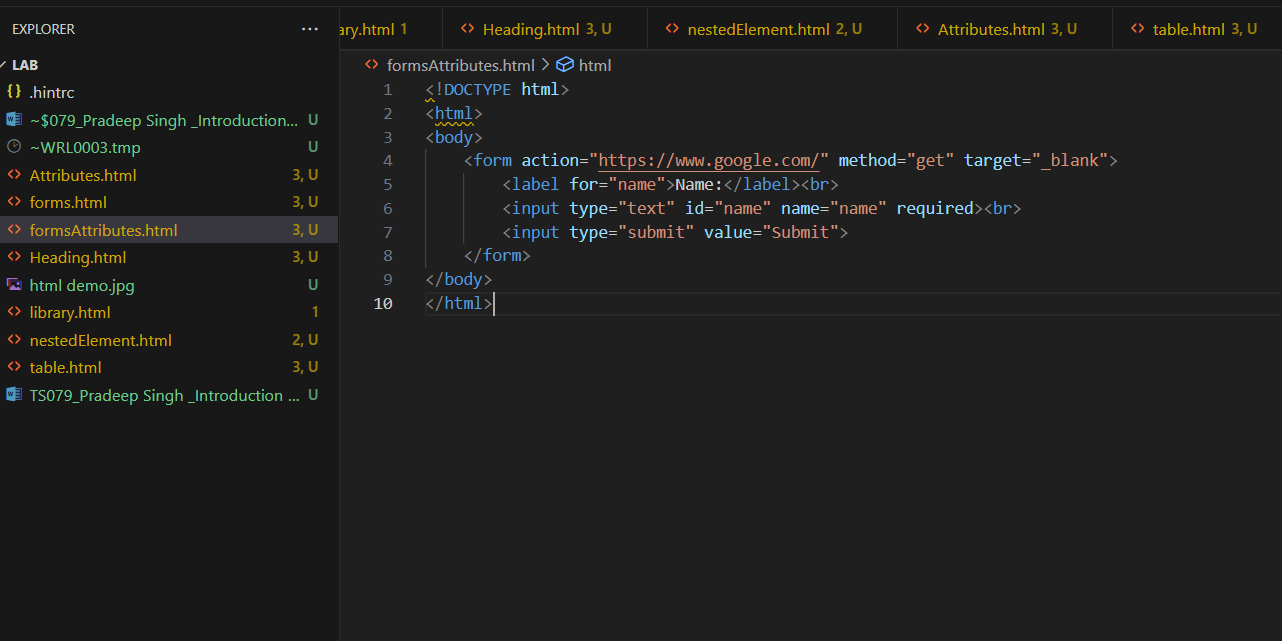
To understand html form, I have created a basic pet preference form in which I have included text fields, checkboxes, radio buttons, submit buttons



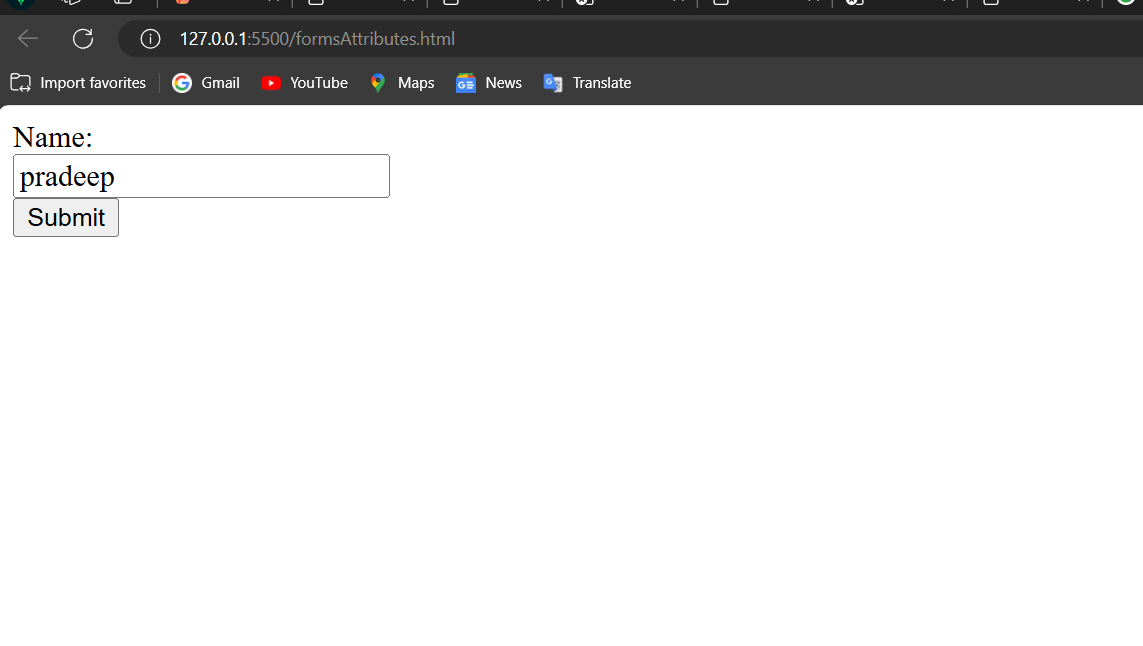
For above code out put will look like as below:



In below practical I have used Action Attribute and method attribute for better understanding:







**The Novalidate Attribute:** The novalidate attribute is a Boolean attribute. When present, it specifies that the form-data (input) should not be validated when submitted. If this attribute is not set, the form-data will be validated.

List of All <form> Attributes: Here are the most commonly used attributes in an HTML <form>:

action: Specifies where to send the form-data when a form is submitted.

method: Specifies the HTTP method to use when sending form-data.

target: Specifies where to display the response that is received after submitting the form.

enctype: Specifies how the form-data should be encoded when submitting it to the server (only for method= “post”).

accept-charset: Specifies the character encodings that are to be used for the form submission.

autocomplete: Specifies whether a form should have autocompleted on or off.

novalidate: Specifies that the form should not be validated when submitted.\

