

19/05/2021

Web Technology & its Applications

(1) What is HTML? Explain the structure of HTML document

Ans:- HTML is defined as a markup language.

A markup language is simply a way of annotating a document in such a way to make the annotations distinct from the text being annotated.

Structure of HTML Document:-

① — <!DOCTYPE html>

— <html>

— <head lang="en">

— <meta charset="utf-8"> — ⑤

② — <title> Share Your Travels -- New York -- Central Park </title>

— <link rel="stylesheet" — ⑥

— href="css/main.css" — ⑦

— <script src="js/html5shiv.js"></script>

</head>

④ — <body>

— <h1> Main heading goes here </h1>

— . . .

</body>

</html>

① DOCTYPE :- tells the browser (or any other client software that is reading this HTML document) what type of document it is about to process.

② HTML, Head and Body :-

HTML5 does not require the use of the <html>, <head> and <body>. The <html> element is sometimes called the root element as it contains all the other HTML elements in the document.

③ Head and body : HTML pages are divided into two sections: the head and the body, which corresponds to the <head> and

and `<body>` elements.

The head contains descriptive elements about the document.

④ Inside the head

- `<head>` contains a variety of additional elements.
- The first of these is the `<meta>` element. Our example declares that the character encoding for the document is `UTF-8`.

⑤ Inside the head

No brains beet metas, ~~eg~~ styles and javascripts

⑥ It also references an external Javascript file.

Q2) Explain the following HTML elements with example. (i) image (ii) list

(i) Image:- Insert images using the `` tag.

- When the `` tag is used, it should also be mentioned which image needs to be displayed. This is done using `src` attribute.
- `Alt` attribute is also used, `alt` stands for alert.
- Some very old browsers would not be having capacity to display the images. In this case, whatever the message given to `alt` attribute, that would be displayed.
- Another use of `alt` → when image display option is disabled by user. The option is disabled when the size of the image is huge and takes time for downloading.

`<html>`

`<head>`
`<title> display image </title>`

`</head>`
`<body>`

```
<img src = "java.png" alt = "cannot display"/>  
</body>  
</html>
```

- (ii) List: HTML provides three types of lists.
- ⇒ Unordered List: Collections of items in no particular order.
These are by default rendered by browser as a bulleted list.
 - ⇒ Ordered List: Collection of items that have a set of order.
These are by default rendered by browser as a numbered list.
 - ⇒ Definition List: Collection of name and definition pairs.
These tend to be used infrequently. Perhaps the most common list example would be a FAQ list.

```
eg:- <ul>  
      <li><a href = "index.html"> Home</a></li>  
      <li>About Us</li>  
      <li>Products</li>  
      <li>Contact Us</li>  
    </ul>
```

- (3) While the division `<div>` based HTML semantics structure elements.
- Q3: The `<div>` tag is also a container element and is used to create a logical grouping of content.
- The `<div>` element has no intrinsic presentation.
 - It is frequently used in contemporary CSS-based layouts to mark out sections.

(84) Define CSS. Explain the location of styles.

CSS is a W3C standard for describing the presentation (or appearance) of HTML elements.

With CSS, we can assign:

- font properties
- colors
- sizes
- borders

• background images.

• even the position of elements.

CSS is a language in that it has its own syntax rules.
CSS can be added directly to any HTML element (via the style attribute), within the <head> element, or most commonly, in a separate text file that contains only CSS.

→ CSS style rules can be located in three different locations:

(1) Inline: An inline style only affects the element it is defined within and will override any other style definitions for the properties used in the inline style.

→ Generally discouraged since they increase bandwidth and decrease maintainability.

→ However be handy for quickly testing out a style change.

<h1> Share Your Travels </h1>

<h2> style = "font-size: 24pt; "Description </h2>

<h2> style = "font-size: 24pt; font-weight: bold;" > Reviews </h2>

(2) Embedded:

* Largely discouraged.

* Since each HTML document has its own <style> element, it is more difficult to consistently style multiple documents using this style.

```
<head lang = "en">
```

```
<meta charset = "utf-8">
```

```
<title> Share Your Travels -- New York -- Central Park </title>
```

```
<style> :
```

```
h1 { font-size : 24pt; }
```

```
h2 {
```

```
    font-size : 18pt;
```

```
    font-weight : bold;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1> Share Your Travels </h1>
```

```
<h2> New York - Central Park </h2>
```

(3) External :- Most common place to locate style rules because it provides the best maintainability.

⇒ When you make change to an external style sheet, all HTML documents that reference that style sheet will automatically use the updated version.

⇒ The browser is able to cache the external style sheet which can improve the performance of the site.

```
<head lang = "en">
```

```
<meta charset = "utf-8">
```

```
<title> Share Your Travels -- New York -- Central Park </title>
```

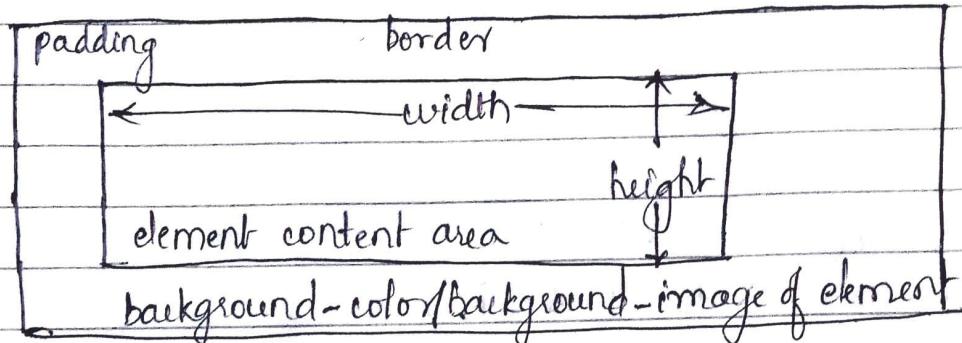
```
<link rel = "stylesheet" href = "styles.css" />
```

```
</head>
```

Q5) Illustrate the CSS Box model using to label each of the components of the box.

Soln:- In CSS, all HTML elements exist within an element box.

* It is absolutely essential that you familiarize yourself with the terminology and relationship of the CSS properties within the element box.



background-color/background-image of element's parent.

Background :- The background color or image element fills of an element out to its border; in contemporary web design it has become extremely common to use CSS to display purely presentational images rather than using `` elements.

Borders :- Borders provide a way to visually separate elements. You can put borders on all four sides of an element, or just one, two, or three of the sides.

Width and Height :- The width and height properties specify the size of the elements' content area.

- > Perhaps the only rival for collapsing margins in troubling our students, box dimensions have a number of potential issues
- > Only block-level elements and non-text inline elements such as

images have a width and height you can specify.

Q6) What are class selectors and id selectors?

Soln:- A class selector allows you to simultaneously target different HTML elements regardless of their position in the document tree.

If a series of HTML element have been labeled with the same class attribute value, then you can target them for styling by using a class selector, which takes the form : period(.) followed by the class name.

<head>

<title> Share Your Travels </title>

...

<h1 class = "first"> Reviews </h1>

<div>

<p class = "first"> By Ricardo on <time> September 15, 2012 </time> </p>

Id selector :- allows you to target a specific element by its id attribute regardless of its type or position.

→ If an HTML element has been labeled with an id attribute, then you can target it for styling by using an id selector, which takes the form : pound/hash (#) followed by the id name.

...

<h1> Reviews </h1>

<div> id = "latestComment">

<p> By Ricardo on <time> ...

20/05/2021

Q1) What is role of `` and `` HTML tags with syntax and example

Sol:- The `` tag defines an unordered list.

```
<ul>
  <li><a href="index.html">Home </a></li>
  <li>About Us </li>
  <li>Products </li>
  <li>Contact Us </li>
```

~~~~

``

``

```
  <li>• Home
  <li>• About Us
  <li>• Products
  <li>• Contact Us
```

Use `` tag together with the `` tag to make unordered list

⇒ The `` tag defines ordered list. An ordered list can be numerical or alphabetical.

The `` tag is used to define each item (list item).

``

```
  <li> Coffee </li>
  <li> Tea </li>
  <li> Milk </li>
```

``

```
  1. Coffee
  2. Tea
  3. Milk
```

(e) Explain the need of 'cascade' in CSS. Illustrate three principles of cascade with suitable CSS snippet segments.

Soln:- The "cascade" in CSS refers to how conflicting rules are handled.

Cascade Principles:

(1) Inheritance: Many (but not all) CSS properties affect not only themselves but their descendants as well.

e) font, color, list and text properties are inheritable

e) layout, styling, borders, background and spacing properties are not.

(2) Specificity: is how the browser determines which style rule takes precedence when more than one style rule could be applied to the same element.

e) The more specific the selector, the more it takes precedence i.e., overrides the previous definition).

e) Browser assigns a weight to each style rule.

e) When several rules apply, the one with the greatest weight takes precedence.

(3) Location: When inheritance and specificity cannot determine style precedence, the principle of location will be used.

e) The principle of location is that when rules have the same specificity, then the latest are given more weight.

(Q3) Define class selectors and pseudo selector of CSS with relevant snippets.

Soln:- A pseudo-element is a way to select something that does not exist explicitly as an element in the HTML document tree but which is still a recognizable selectable object.

e) A pseudo-class selector does apply to an HTML element, but targets either a particular state or, in CSS, a variety of family relationships.

e) The most common use of this type of selector is targeting link states.

```
<style>
a:link {
    text-decoration:underline ;
    color:blue ;
}
```

```
<p> ... </p>
<ul><a href="#"> Canada </a> </li>
;
</ul>
</body>
```

Q4) Explain two types of URL referencing techniques with suitable scripts in HTML5.

Soln:- URL Absolute Referencing :- When referencing a page or resource on an external site, a full absolute reference is required that is,

- the protocol (typically, `http://`) ,
- the domain name
- any paths, and then finally
- the file name of the desired resource .

URL Relative Referencing :- We also need to be able to successfully reference files within our site

- ⇒ This requires learning the syntax for so-called relative referencing
- ⇒ When referencing a resource that is on the same server as your HTML document, then you can use brief relative referencing. If the URL does not include the "`http://`" then the browser will request the current server for the file .

(as) Explain the role of following semantic elements of HTML5 with syntax & script segments

(i) `<nav>`

(ii) `<section>`

(iii) `<aside>`

Soln:- (i) `<nav>` :- represents a section of page that contains links to other pages or to other parts within the same page.

`<header>`

`<nav role = "navigation">`

``

``

(ii) `<section>` : this element represents a section of a document, typically with a title or heading.

According to the W3C, `<section>` is a much ~~broad~~ broader element, while the `<article>` element is used for blocks of content that could potentially be read or consumed independently of the other content on the page.

(iii) `<aside>` : the `<aside>` element is similar to the `<figure>` element in that it is used for making up content that is separate from the main content on the page. The `<aside>` element could thus be used for side bars, pull quotes, groups of advertising images, or any other grouping of non-essential elements.

(b) Explain the following CSS properties with suitable examples :

(i) float

(ii) position

(iii) overflow.

Sol:

(i) **float** :- the float is property places an element on the left or right side of its container, allowing text and inline elements to wrap around it. The element is removed from the normal flow of the page, though still remaining a part of the flow.

float : none;
float : left;
float : right;

(ii) **position** :- An element with position : fixed; is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element. A fixed element does not leave a gap in a page, where it would normally have been located.

(iii) **Overflow** :- This property specifies whether to clip the content or to add scroll bars when the content of an element is too big to fit in the specified area. The overflow property has the following values now - The overflow is clipped, and a scrollbar is added to see the rest of the content

Ave / sep - 2020

- a) Q2) previously written
- b) With example explain HTML explain
- c) Discuss the structure of HTML documents .

- Q3) Explain any six HTML elements.

Sol:- An HTML element can contain text, other elements, or can be implied. It is identified in the HTML documents by tags.

⇒ HTML element can contain attributes. An attribute is a name = value pair that provides more information about the HTML element.

⇒ Insert images using the `` tag.

⇒ Create links with the `<a>` tag.

⇒ Create lists with the ``, `` and `` tags.

⇒ Create headings with `<H1>`, `<H2>`, ..., `<H6>`

⇒ Define metadata with `<meta>`.

Q4) What is CSS? Explain benefits of CSS.

Soln :- CSS is a language in that it has its own syntax rules.

Benefits of CSS :-

- The degree of formatting control in CSS is significantly better than that provided in HTML.
- Web sites become significantly more maintainable because all formatting can be centralized into one, or a small handful, of CSS files.
- CSS-driven sites are more accessible.
- A site built using a centralized set of CSS files for all presentations will also be quicker to download because each individual HTML file will contain less markup.
- CSS can be used to adapt a page for different output mediums.

Module-2

Q1. Explain the basic structure creating an HTML document for fig 0.1

One	Two
Three	Four

Q2) With the sample HTML form, explain how forms work.

Sohn: Forms provide the user with an alternative way to interact with a web server.

- Forms provide rich mechanisms like:

- ⇒ Text input

- ⇒ Password input

- ⇒ Option lists

- ⇒ Radio and check boxes

```
<form method="get" action="process.php">
```

```
</fieldset>
```

```
<legend> Details </legend>
```

```
<p>
```

```
<label> title : </label>
```

```
<input type="text" name="title"/>
```

```
</p>
```

```
<p>
```

```
<label> country : </label>
```

```
<select name="where">
```

```
<option> choose a country </option>
```

```
<option> Canada </option>
```

```
<option> Finland </option>
```

```
<option> United States </option>
```

```
<select>
```

```
</p>
```

```
<input type="submit" />
```

```
</fieldset>
```

```
</form>
```

What are the various form related HTML elements

- Soln: The two essential features of any form, namely the action & method attributes.
- The action attribute specifies the URL of the server-side resource that will process the form data.
 - ⇒ The method attribute specifies how the query string data will be transmitted from the browser to the server.
- GET
 - POST

```
<form method="get" action="process.php">  
<form method="post" action="process.php">
```

POST :- Data can contain binary data. Data is hidden from user.
Submitted data is stored in cache, history or bookmarks.

Type	Description
<button>	Defines a clickable button
<datalist>	An HTML5 element from defines lists to be used with other form elements.
<fieldset>	Groups related elements in a form together
<form>	Defines the form container
<input>	Defines an input field. HTML5 defines over 20 different types of input.
<label>	Defines a label for a form input element
<legend>	Defines the label for a fieldset group
<option>	Defines an option in a multi-item list.

Qn) Explain positioning elements in CSS

Soln :- The position property is used to specify the type of positioning and the possible values are:

<u>Value</u>	<u>Description</u>
Absolute	The element is removed from normal flow and positioned in relation to its nearest position ancestor.
Fixed	The element is fixed in a specific position in the window even when the document is scrolled.
Relative	The element is moved relative to where it would be in the normal flow.
Static	The element is positioned according to the normal flow. This is default.

Q5) What is responsive design? Explain the four key components that make responsive design work.

Soln: In a responsive design, the page "responds" to changes in the browser size that go beyond the width. scaling of a liquid layout. There are 4 important key components to make responsive design work.

1. Liquid layouts
2. Scaling images to viewport size.
3. Setting viewports via the `<meta>` tag.
4. Customizing the CSS for different viewports using media queries

DEC/JAN-2020

Q1) Compare radio and checkbox controls of HTML with example.

Radio Buttons:

Radio Buttons are useful when you want the user to select a single item from a small list of choices and you want all the choices to be visible.

- ⇒ Radio buttons are added via the `<input type = "radio">` element

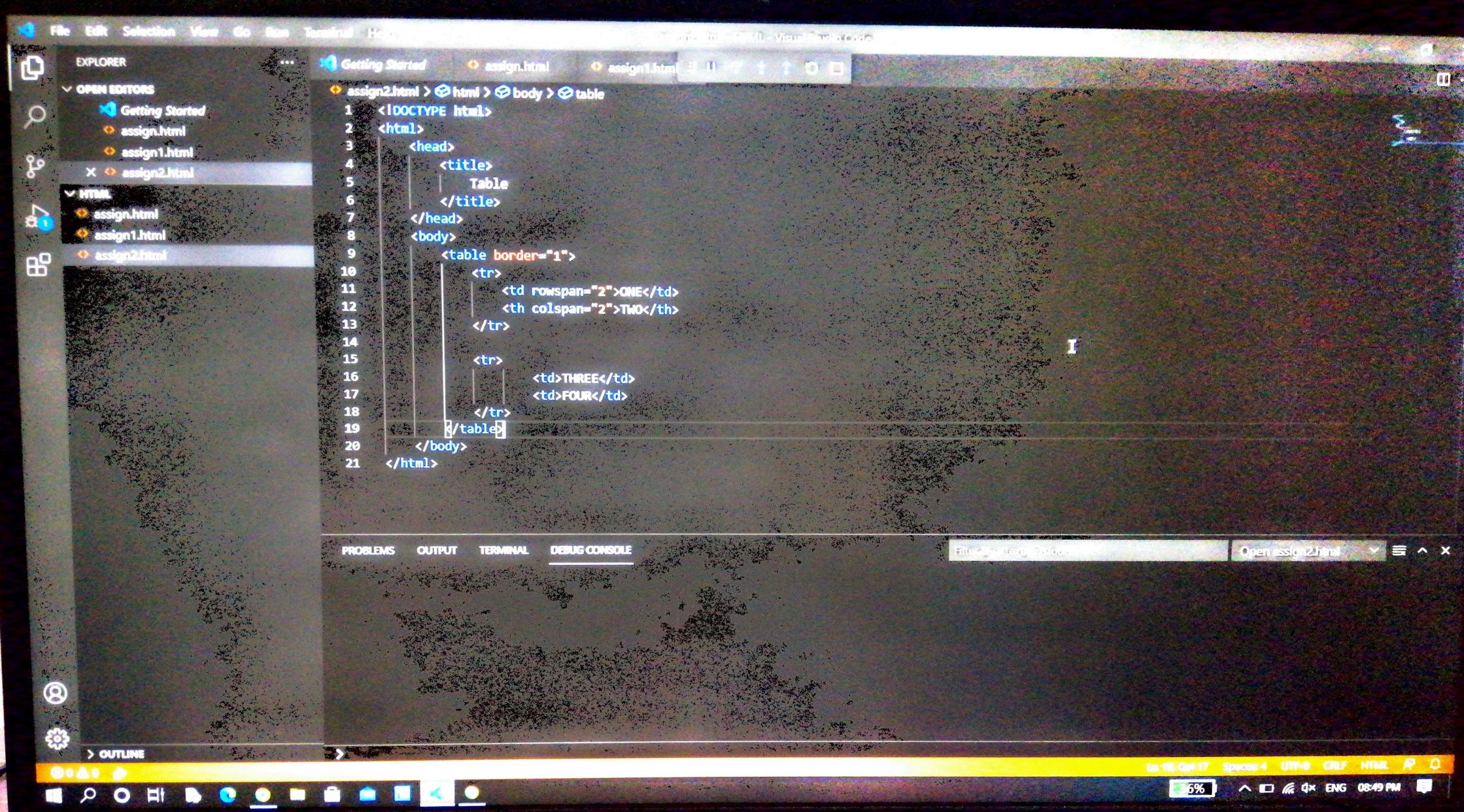
continent:

- North America `<input type = "radio" name = "where" value = "1">`
- South Africa `<input type = "radio" name = "where" value = "2">`

checkboxes:

checkboxes are used for getting yes/no or on/off responses from the user.

- ⇒ checkboxes are added via the `<input type = "checkbox">` element.
- ⇒ checkboxes are also group checkboxes together by having them share the same name attribute.





Table



① File | C:/Users/lbtesam%20Zarrine/HTML/assign2.html

ONE	TWO
THREE	FOUR



EXPLORER

OPEN EDITORS

Getting Started

assign.html

assign1.html

assign2.html

HTML

assign.html

assign1.html

assign2.html

Getting Started

assign.html

assign1.html

> html > form > label > label

```
1  <!DOCTYPE html>
2  <html>
3      <title>
4          Assignment 1
5      </title>
6      <form> <label for="name">Name</label><br>
7          <input type="text" placeholder="Ibtessam Zarrine" id="name" required>
8          <br><br>
9          <label>USN</label>
10         <input type="text" placeholder="3gn18cs034" required>
11         <br><br>
12         <label for="idpass">Password</label><br>
13         <input type="password" id="idpass" placeholder="Enter the password" required>
14         <br><br>
15         <label>Date of birth</label><br>
16         <input type="date">
17         <br><br>
18         <label>Gender</label>
19         <input type="radio" name="gender" value="1" checked>Female
20         <input type="radio" name="gender" value="2">Male
21         <input type="radio" name="gender" value="3">Other
22         <br><br>
23         <label>Stream</label><br>
24             <select size="2" >
25                 <option value="1">CSE</option>
```

PROBLEMS

OUTPUT

TERMINAL

DEBUG CONSOLE

Filter (e.g. text, !exclude)

Open assign2.html



> OUTLINE



Name

Ibtesam Zarrine

USN 3gn18cs034

Password

ente the password

Date of birth

dd - mm - yyyy

Gender Female Male Other

Stream

CSE
ISE

Description

write your hobbies and skills

Reset

Submit

The screenshot shows a Visual Studio Code interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, Terminal, Help
- Title Bar:** Getting Started, assign.html X, assign1.html, assign2.html
- Explorer:** Shows "OPEN EDITORS" with "Getting Started" and "assign.html" selected. It also lists "assign1.html" and "assign2.html".
- Search:** A magnifying glass icon.
- Search Results:** A list of "HTML" files: assign.html, assign1.html, assign2.html.
- Code Editor:** The "assign.html" file is open, displaying the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     Assignment
5   </head>
6   <body>
7     <table border="1">
8       <thead style = "color: green">
9         <tr>
10        <th>Sl.no</th>
11        <th>Name</th>
12        <th>Usn</th>
13        <th>Dept</th>
14      </tr>
15    </thead>
16    <tbody>
17      <tr>
18        <td>1</td>
19        <td></td>
20        <td></td>
21        <td></td>
22      </tr>
23      <tr>
24        <td>2</td>
25        <td></td>
26        <td></td>
27        <td></td>
28      </tr>
29    </tbody>
30    <tfoot>
31      <tr>
32        <td>total no of rows</td>
33        <td>2</td>
34        <td></td>
35        <td></td>
36      </tr>
```

```
File Edit Selection View Go Run Terminal Help
Getting Started assign.html assign1.html assign2.html
EXPLORER
OPEN EDITORS
Getting Started
assign.html
assign1.html
assign2.html
HTML
assign.html
assign1.html
assign2.html
assign.html > html > body > table
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
<tfoot>
<tr>
<td>total no of rows</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tfoot>
</table>
</body>
</html>
```

Assignment

Sl.no	Name	Usn	Dept
1			
2			
total no of rows	2		