Lecture no 8

Web Intruders

Drop down menus:

Dropdown menus in HTML are typically created using **<select>** and **<option>** elements. The **<select>** element creates a dropdown list, and each **<option>** element within it represents an item in the dropdown menu.

Their appearance can be customized using CSS properties such as **background-color**, **color**, **border**, **padding**, etc. The options within the dropdown menu are styled using CSS as well.

Navigation Bars:

Navigation bars in HTML are elements used to create menus or navigational links on a webpage. They typically consist of a list of links to different sections or pages of a website.

One common way to create a navigation bar in HTML is by using an unordered list (**<ul>**) combined with list items (**<li>**) for each navigation item.

In HTML, navigation bars can be created using ordered lists (**<ol>**) or unordered lists (**<ul>**). Both types of lists are used to structure and organize content, but they have different visual representations and default styling.

1. **Ordered List (<ol>):**
   * An ordered list is a list where each item is sequentially numbered by default.
   * To create an ordered list for a navigation bar, you would typically use the **<ol>** tag.
   * Each item in the list is marked up with the **<li>** (list item) tag.

2. **Unordered List (<ul>):**

* An unordered list is a list where each item is bulleted or marked with some other symbol, typically a bullet point.
* To create an unordered list for a navigation bar, you would use the **<ul>** tag.
* Like ordered lists, each item in the list is marked up with the **<li>** tag.
* Top of Form

Both ordered and unordered lists can be styled using CSS to achieve the desired appearance, such as changing the bullet points for unordered lists or the numbering style for ordered lists. They offer flexibility and are commonly used for navigation menus due to their semantic meaning and ease of styling.

Top of Form

Transformation:

In HTML and CSS, transformation refers to the process of changing the appearance of an element by modifying its size, position, or shape. CSS provides several properties for applying transformations to elements, allowing for effects such as scaling, rotating, skewing, and translating (moving). Here are some common transformation properties and their types:

**Transform Property (transform):**

* The **transform** property is a shorthand property that allows you to apply various transformations to an element.
* It can accept multiple transformation functions separated by spaces.

**Translation (translate):**

* Translation moves an element from its current position along the horizontal and vertical axes.
* Syntax: **translate(<tx>, <ty>)**, where **<tx>** is the horizontal translation and **<ty>** is the vertical translation.

**Rotation (rotate):**

* Rotation rotates an element clockwise or counterclockwise around a specified point.
* Syntax: **rotate(<angle>)**, where **<angle>** is the rotation angle in degrees.

**Scaling (scale):**

* Scaling changes the size of an element along the horizontal and vertical axes.
* Syntax: **scale(<sx>, <sy>)**, where **<sx>** is the horizontal scale factor and **<sy>** is the vertical scale factor.

Skewing(skew):

* Skewing tilts or slants an element along the horizontal and vertical axes.
* Syntax: **skew(<ax>, <ay>)**, where **<ax>** is the horizontal skew angle and **<ay>** is the vertical skew angle.

Origin (transformation origin):

* The **transform-origin** property defines the point around which transformations are applied.
* By default, transformations are applied around the center of the element.

Animation:

Animations in HTML and CSS allow you to add movement and interactivity to your web pages without relying on external plugins or scripts. CSS animations are created using keyframes and animation properties. CSS animations are relatively easy to implement and can be used to create various effects like transitions, keyframe animations, and more

**Key Frame animation:**

* Keyframes animation allows you to define a sequence of keyframes at specific points in time to create complex animations.
* You define the animation using **@keyframes** rule and then apply it to an element using the **animation** property.

complete