**SEP 9TH Full stack development from udemy by angela yu**

<https://codepen.io/Devakari-Saiteja/pen/rNvZwoq> for best online full stack development practice

[https://web.archive.org/web/20030301000000\*/www.amazon.com](https://web.archive.org/web/20030301000000*/www.amazon.com) we can go back to years and can see the website in older versions

<https://unicode-table.com/en/> Here we can use the unique symbols

<https://colorhunt.co/palettes/popular> = for best colours to use

<https://www.flaticon.com/search?word=cooking&type=animated-icon> = is a website where we can find many images, gifs and vid based on our requirement.

<https://css3buttongenerator.com> = This web site helps us in generating the CSS buttons a=in different styles.

<em> with the help if emphasis tag we can make the inside data into italic form.

<strong> = stronger words

<i> = italic

<b> = bold

<br> = break

<h1> = head tags

<hr> = horizontal line , <hr size = “ 3 “ noshade/>

Noshade attribute is used to make the solid horizontal line to shaded line.

<ul> = un ordered list

<ol> = ordered list, <ol type = " I "> = for ROMAN numbers, <ol start = "6"> = to start with numbers

<li> = list of items

<img sir = "image.png" alt="generator" width="650" height="450"> = For image addition

<a href="http://www.amazon.com">This one</a> = for URL

<table> it’s the main tag for creating tables

<tr> = table row

<td> = table data

**HTMLFORMS =**

**<**forms>

<label> Name:</label>

<input type=”text”>

<input type=”submit”>

<input type=”file”>

<input type=”date”>

<input type=”range”>

</form>

GIT HUB

* In git hub while adding files to repository it’s must there should be one file with name index.html to open this file as a main file without any disturbances.

**CSS**

**<font>**

<center>

<h1 bgcolor=”#990000”>

<layout>

<body style="background-color:#94b8b8;"> = for background color

**<div> =** div is used to structure and divide up your code.

The <span> tag is an inline container used to mark up a part of a text, or a part of a document.

The <span> tag is easily styled by CSS or manipulated with JavaScript using the class or id attribute.

The <span> tag is much like the [<div>](https://www.w3schools.com/tags/tag_div.asp) element, but <div> is a block-level element and <span> is an inline element.

**P1:hover =** The :hover selector is used to select elements when you mouse over them.

**Tip:** The :hover selector can be used on all elements, not only on links.

**Note:** :hover MUST come after :link and :visited (if they are present) in the CSS definition, in order to be effective!

**Class =**

**Id =**

**Tags =**

**Some of common inline elements =** spans(**<span>**), images(**<img>**), Anchors( **<a>** )

**Display elements = Block, inline), inline-block, None**

( <https://www.w3schools.com/css/tryit.asp?filename=trycss_inline-block_span1> )

By using the inline-block we get data side by side

**Positions = absolute, fixed, sticky, relative etc.**

## position: static;

HTML elements are positioned static by default.

Static positioned elements are not affected by the top, bottom, left, and right properties.

An element with position: static; is not positioned in any special way; it is always positioned according to the normal flow of the page:

## position: relative;

An element with position: relative; is positioned relative to its normal position.

Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.

## position: fixed;

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 the page where it would normally have been located.

Notice the fixed element in the lower-right corner of the page. Here is the CSS that is used:

## position: absolute;

An element with position: absolute; is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.

**Note:** Absolute positioned elements are removed from the normal flow, and can overlap elements.

**Coordinates** = Top, Bottom, Left, Right.

1em, 2em these are used to size the text itself dynamically, then the scaling will be set based on user preferences.

For sizing text use rem instead of using em or percentages or pixel because that is the most adoptable and reliable one.

We know that 1em = 16px.

Q: if we need the font size from 90px to percentage

A: divide the 90 with 16 and multiply the answer with 100 so that’s the final out put = 562.5%

Q: if we need the font size from 40px to rem

A: divide the 40px with 16 = 2.5rem ( also written as 250% in percentage)

***text-align: left;*** is the command used to move the text to left apart from image and other.

The float property specifies whether an element should float to the left, right, or not at all.

**Note:** Absolutely positioned elements ignore the float property!

**Note:** Elements next to a floating element will flow around it. To avoid this, use the [clear](https://www.w3schools.com/csSref/pr_class_clear.php) property or the clear fix hack (see example at the bottom of this page).

The **clear property** is used to specify which side of floating elements are not allowed to float. It sets or returns the position of the element in relation to floating objects. If the element can fit horizontally in the space next to another element which is floated, it will.

CDN = cdn stands for content delivery network

In train journey (05/11/22)

<https://compass.talent.cognizant.com/psc/HCMPRD/EMPLOYEE/HRMS/c/NUI_FRAMEWORK.PT_LANDINGPAGE.GBL>?