What is css? what is the purpose of css?

CSS stands for cascading style sheets. css is used for styling the elements of a web page, we can change the color, size and other aspects in rending the elements of a web page using css.

Why do we need to use css, why not html provide tags for doing the same?

initially html has provided formatting tags like font, color, background-color etc. having these tags written at all the places of a web site in web pages is going to be complex and takes lot of time for developing the website.

For eg.. we wanted all the headings of our pages to be look alike "font: helvatica, font-size: 20px, color: red" if the same has to be applied for all the pages each heading should be sourrouned with formatting html tags that takes more time in developing web pages by the web developers.

To solve this problem w3c has introduced style sheets. cascading style sheets, once written will be cascade to all the elements based on matching / hierarchy. So by use of css we can reduce greatly the amount of code we have to write for applying styles to the web pages.

How to write cascading style sheet?

One cascading style is called style rule. rule looks as below

```
selector {declaration; declaration}
          prop: value; prop: value
```

selector = the elements on whom the style should be applied

all the declarations should be written in {} curly brackets

declaration = indicates a property and its value we want to apply on a element. The properties varies depends on the type of the element on whom we are applying.

prop = name of style property we want to modify

value = is a pre-defined value or a range that is allowed for that property

example:-

```
<style>
p {
  font-family: fantasy;
  font-size: 23px;
}
</style>
```

----

css selectors

Using css selector we can tell the style rule has to be applied to which elements on our web page. We can classify a style selector into below types

1. element selector
2. id selector
3. class selector
4. block selector
5. psuedo-selector

#1 element selector

We target the style to a element of a type in web page by writing tag name of the element as shown below.

```
<html>
  <head>
    <style>
      p {
        font-family: helvitica;
        font-size: 14px;
      }
    </style>
  </head>
  <body>
    <p>Paragraph 1</p>
    <p>Paragraph 2</p>
  </body>
</html>
```

#2 id selector

It is used for applying a style to a specific element of a web page rather than all the elements.

For every html element we can give an id attribute that helps in uniquely identifying and accessing the element.

```
<html>
  <head>
    <style>
      #para1 {
        font-family: helvitica;
        font-size: 14px;
      }
    </style>
  </head>
  <body>
    <p id="para1">Paragraph 1</p>
    <p>Paragraph 2</p>
  </body>
```

```
</html>
```

#3 class selector

we bind html elements to a class name and we apply styles to all the elements baring the class. In this approach we can select subset of elements on page to whom we want to apply the styles

```
<html>
  <head>
    <style>
      .mainContent {
        font-family: helvitica;
        font-size: 14px;
      }
    </style>
  </head>
  <body>
    <p class="mainContent">Paragraph 1</p>
    <p>Paragraph 2</p>
    <div class="mainContent">
      This a section of content
    </div>
  </body>
</html>
```

#4 block/group selector

to all the group of elements we want to apply a style

```
<html>
  <head>
    <style>
      p, a, div {
        font-family: helvitica;
        font-size: 14px;
      }

    </style>
  </head>
  <body>
    <p>Paragraph 1</p>
    <p>Paragraph 2</p>
    <ul>
      <li></li>
    </ul>
    <a href="">click me</a>
    <div>
      This is the content
    </div>
  </body>
</html>
```

$5 global selector
To every element on my page apply these styles

```
<html>
  <head>
    <style>
      * {
        font-family: helvitica;
        font-size: 14px;
      }

    </style>
  </head>
  <body>
    <p>Paragraph 1</p>
    <p>Paragraph 2</p>
    <ul>
      <li></li>
    </ul>
    <a href="">click me</a>
    <div>
      This is the content
    </div>
  </body>
</html>
```

#6 psuedo selector
based on condition/criteria we can select the elements for whom the style should be applied

```
<html>
  <head>
    <style>
      p:nth-child(odd) {
        font-family: helvitica;
        font-size: 14px;
      }

    </style>
  </head>
  <body>
    <p>Paragraph 1</p>
    <p>Paragraph 2</p>
    <ul>
      <li></li>
    </ul>
    <a href="">click me</a>
    <div>
      This is the content
    </div>
  </body>
</html>
---------------------------------------------------------------------------------------------------------------
```

How many ways we can write css in web pages?

There are three ways in writing the css for web pages.

1. external stylesheet = we write all the style rules in a external file and import into a web page
2. internal stylesheet = within the webpage we write style rules
3. inline styles = embedded within the element

1. external stylesheet
----

all the css rules for your website will be written in an external file with extension ".css". The file in which we write style rules is called "Stylesheet" | "css" file. We import the css file into the webpage to apply styles.

We can easily change the look and feel of the controls of the website by simply changing the external css file. It is recommended to use external stylesheet for a web page. It helps us in reusing the style rules.

```
project
|-css
  |-default.css
  |-dark.css
|-our-services.html
```

```
css/default.css
----------------
body {
  font-family: fantasy;
  font-size: 14px;
}
.important {
  color: red;
}
```

```
css/dark.css
body {
  font-family: Tahoma;
  font-size: 15px;
}
.important {
  color: red;
  font-style: italic
}
```

```
our-services.html
------------------
<html>
  <head>
    <title>Our Services</title>
    <link rel="css/stylesheet" href="css/dark.css">
  </head>
  <body>
    <p class="important">
    </p>
  </body>
</html>
```

#2 Internal Stylesheet
If I have stylerules that has to applied to a specific web page only. in such a case define the style local to webpage using style tags

```
our-services.html
------------------
<html>
  <head>
    <title>Our Services</title>
    <style>
      .important {
        font-family: monospace;
      }
    </style>
  </head>
  <body>
    <p class="important">
    </p>
  </body>
</html>
```

#3 inline styles
Inline styles are written directly at the element level to which want to apply the style. it is highly dis-couraged to use inline styles. To write an inline style at the element level we need use an attribute for every element called style="declaration"

```
our-services.html
------------------
<html>
```

```
  <head>
    <title>Our Services</title>
  </head>
  <body>
    <p style="font-family: Tahoma; font-size: 18px;">
    </p>
  </body>
</html>
----------------------------------------------------------------------------------------------------------------
color css style = in web pages
What are the different ways we can use colors in a web page using css?

css background color
---------------------
<style>
  .subheading {
    background-color: cadetblue;
  }
</style>
<p class="subheading">We are using background color for this paragraph</p>

css text color
---------------
used for changing the color of text display
<style>
  p {
    color: red;
  }
</style>
<p>
  Text in color
</p>

css border color
---------------
border color will display the line around the element in color specified
note:- border color is applied only when set the border, border-style properties.

<style>
  p {
    border: 1px;
    border-style: solid;
    border-color: aqua;
  }
</style>
```

<p>
  Paragraph with colored border
</p>

we can apply border color to all sides also
border-left-color, border-right-color, border-top-color, border-bottom-color
the shorthend for writing it is border-color: leftcolor rightcolor topcolor bottomcolor;

css color choosing

We can directly specify the color names.
there are other ways we can customize the color we want to render using one of three ways
rgb();
HEX
HSL

<p style="color: rgb(255, 23, 200);"></p> = r=red, b=blue, g=green using you can create different colors
<p style="color: rgba(255, 23, 200, 0.5);"></p>

HEX = Hexa Decimal Numbers
<p style="color: #ad9333;"></p>

HSL = hue, saturation, lightness
<p style="color: hsl(0, 100%, 100%)"></p>

----
css background
There are several css properties available regarding managing the background for elements
1. background-color
2. background-image
3. background-repeat
4. background-attached

background-color = is to change the background color of an element.

<div style="background-color: rgb(10,03,40); opacity: 0.2">
</div>

background-image = used for displaying an image as a background for container if the image size is less than the container automatically the image will be repeated horizantal and vertical.

```

<div style="background-image: url(path)">
</div>

background-repeat = using this we can control the image repeatitions
note:- this property is applicable when we have background-image specified

<div style="background-image: url(CHERRYPICK.png); background-repeat: no-repeat"></div>

background-position:- position is used for placing the background image within the container
note:- if we use background-image with background-repeat : no-repeat then only background-position property is applied.
for eg.. top right | top left | bottom right | bottom left
----------------------------------------------------------------------------------------------------------------
css borders
border is a line we want to draw around an element.
1. border
2. border-color
3. border-style
  - solid
  - dotted
  - dashed
4. border-radius = is used for drawing rounded border around an element
----------------------------------------------------------------------------------------------------------------
css margins
css margin is used to create space around the element outside the border of the element

<style>
  p {
    border: 1px;
    border-style: solid;
    border-color: black;
    margin: 20px, 20px, 20px, 20px;
    margin-top: 20px;
    margin-bottom: 20px;
    margin-left: 20px;
    margin-right: 20px;
  }
</style>

<p>Top Tag</p>
```

```
<p>Bottom Tag</p>

-----------------------------------------------------------------------------------------------------------------
css padding
css padding creates a space around the contents of the element

<style>
  .subsection {
    border: 1px;
    border-style: solid;
    border-color: black;
    padding-top: 10px;
    padding-bottom: 10px;
    padding-left: 10px;
    padding-right: 10px;
    padding: 10px, 10px, 10px, 10px;
  }
</style>

<p class="subsection">Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.</p>
------------------------------------------------------------------------------------------------------------------
css height/width
css height and width is used for changing the heigh and with of an element in rendering.

for every element we can specific heigh and width property to render on the web page.
1. by default height and width of an element is auto (based on contents it will be rendered)
2. we can specify height and width in either cm, px or %

<style>
  div {
    height: 400px;
    width: 60%
  }
</style>

<div>
  Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.
```

```
</div>
```

in addition to height and width the css has added 2 more properties

1. max-width
2. max-height

both these helps you in adjusting the element sizes based on screen resolution.

----

css outline

its an line that is drawn outside the border of the element to make a element highlighted.

Below are the outline properties with which we can drawn the outline of an element

1. outline-style:
2. outline-color:
3. outline-width:
4. outline

outline-style is mandatory to have other properties work

```
<style>
  div {
    border: 1px;
    border-style: solid;
    border-color: black;
    outline-style: solid;
    outline-width: 10px;
    outline-color: red;
  }
</style>

<div>
    Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.
</div>
```

----

css text

There are various css formatting options are available for text content we render in a element

1. text color
2. background-color
3. text alignment
4. text transformation

5. text spacing
6. text shadow

```
<style>
  body {
    color: aquamarine; /** applies to only text we render on the page **/
    text-align: justify;
    direction: rtl;
    vertical-align: top;
    text-decoration: line-through;
    text-transform: capitalize;
    letter-spacing: 5px;
    line-height: 5px;
    word-spacing: 10px;
    white-space: nowrap;
    text-indent: 50px;
    text-shadow: 2px 2px;
  }
  p {
    color: red;
  }
</style>

<body>
  How are you
  <p>
          Good Morning How are
    asdfafasfdafd
  </p>
  <p>
      G o o d       M o r n i n g

      H a v e     a      g r e a t     d a y
  </p>
</body>
------------------------------------------------------------------------------------------------------------
```

css fonts
using the font-family and font-size we can change the font of a web page in css.
all the browsers supports predefined set of fonts which are called "Font Web Safe"

"Font Web Safe" supported by all the browsers
  Arial (sans-serif)
  Verdana (sans-serif)
  Helvetica (sans-serif)
  Tahoma (sans-serif)
  Trebuchet MS (sans-serif)

Times New Roman (serif)
Georgia (serif)
Garamond (serif)
Courier New (monospace)
Brush Script MT (cursive)

but there is always a chance where due to different browsers and version few web safe fonts may not be supported by browser. due to this problem always use fallback fonts for your primary font as below.

font-family: Helvetica; sans-serif
The fallback fonts will be applied by the browser when the primary font is not available. and it is recommended to use fallback fonts as more generic fonts

Always one font is the king. If at all required use different fonts at hierarchy level
Heading H1- Use one font
h2 = another font
h3 = another font
subsection
important
paragraph
default font body level

----

css icons

```
<head>
  <script src="https://kit.fontawesome.com/6ea0d26f8e.js" crossorigin="anonymous"></script>
</head>
<body>
  <i class="fas fa-database">Database</i>
</body>
```

----

css links
we have various different styles we can apply for a anchor tag in a web page. by default anchor tag is displayed with underline.

default = blue
active=red
visited=purple

all of the above default behaviours can be modified using css links

```

<head>
  <style>
    a {
      text-decoration: none;
      color: brown;
    }
    a:active {
      color: chartreuse;
    }
    a:visited {
      color: aqua;
    }
    a:hover {
      color: aquamarine;
    }
  </style>
</head>
<body>
    <a href="#">Home</a>
</body>
--------------------------------------------------------------------------------------------------------------
```

css lists

Using css we can change the look and feel of a html list like
 we can change list icon, we can replace a list with image, we remove bullets and we change the position of the icons as well

```
<head>
  <style>
    ul.li {
      list-style-type: circle; = we can change icon
      list-style-image: url(''); = we can replace with image
      list-style-position: inside; = we can shift the position li in the list
      list-style-type: none; = we can hide the icon
    }
  </style>
</head>
<body>
  <ul>
    <li>Banana</li>
    <li>Apple</li>
    <li>Mango</li>
    <li>Oranges</li>
  </ul>
```

```
</body>
--------------------------------------------------------------------------------------------------------------
CSS Table
-----------
For a table we can apply
  1. borders
  2. size
  3. style

<head>
  <style>
    table td, th {
      border: 1px solid black;
      border-collapse: collapse;
    }
  </style>
</head>
<body>
  <table>
    <tr>
      <th>Job Title</th>
      <th>Description</th>
      <th>Organization</th>
      <th>Experience</th>
      <th>Location</th>
    </tr>
    <tr>
      <td></td>
      <td></td>
      <td></td>
      <td></td>
      <td></td>
    </tr>
  </table>
</body>
#1
we apply border to a table at different level
table {
  border: 1px solid black;
}
this will draw a border around the table but will not draw border for rows and columns

#2 we can apply border to only rows
table tr {
  border: 1px solid black;
}
this will leaves table with no borders but tr drawn with borders
```

Note:- when we apply borders at td/th level we dont need to write borders for tr

#3we can apply border for all the cells in the table

```
table, td, th {
  border: 1px solid black;
}
```

Now for table, td, th borders are applied and appears as double border around each cell, so we can collapse double border by writing border-collapse: collapse

2. heigh and width of a table

```
<style>
  table {
    height: 100px;
    width: 100%;
    margin: auto;
  }
</style>
```

3. text align of contents in table cells

we can align contents of the table cell by using text-align property at td/th level

```
td {
  text-align: center;
}
```

4. table styles

padding = can be given for each cell to create space around the content

```
table td, th {
  border-bottom: 1px solid black;
}

table tr:nth-child-element(even) {
  background-color: grey;
}
table tr:hover {
  backgroud-color: red;
}

th {
  background-color: red;
  color: white;
}
```

----

css display|visible

display: is an css property that can carry the below values.

1.none
2.block
3.inline

```
<style>
  #p1 {
    display: none;
  }
</style>
<p id="p1">Paragraph</p>
```

when we set display:none; property the element will be hidden in web page which can used to remove elements without removing them from page layout.

every element in html has a display attribute which is block-level or inline-level

block-level
- starts always in new line
- occupies to the max with of the page

inline-level
- relative page flow
- occupies only to the max of the content

we can change default display type of an element to block-inline and inline-block using display property

```
<style>
  p {
    display: inline;
  }
</style>

<p>p1</p>
<p>p2</p>
<p>p3</p>

<style>
  i {
    display: block;
  }
</style>
<i>i1</i><i>i2</i><i>i3</i>
```

Other css property is visibility which is used for controlling the display of the element whether it should be shown on the page or not. The possible values it can take is

1. visible
2. hidden = within the element an empty space will be left the page will not adjusted.

visibility: hidden = will hides the content but doesnt changes the page structure, so the when we hide the element any empty space will appear within its place

display: none = hides and removes the space pushing next set of elements to occupy the space

----

css position property

position property help us in positioning the html elements on a web page. it helps in managing the position method of html elements

we can move the elements or position the elements of a web page using left, right, top, bottom properties of an element, but these attributes will be applied based on position property of the element. Based on the position property the behaviour of left/right/top/bottom attributes will change

There are total 5 position methods are there allowed for position property

1. static
2. fixed
3. relative
4. absolute
5. sticky

for e.g.. below is the snippet shows you how to apply position property for an HTML element.

```
<div style="position: static">
</div>
```

1. static

HTML elements are positioned by default based on static position

if an elements position is "static" then left|right|top|bottom will not have any affect. The elements are positioned according to the page flow relative to the page.

```
<style>
  .static {
    position: static;
    border: 1px solid black;
    left: 30px;
  }
```

```
</style>

<div>
  Introduction to css
</div>
<div class="static">
  Paragraph with static position
</div>
```

3. relative

by default the elements with position: relative falls into the normal page flow. but left|top|right|bottom properties can be used for chaning the display position of the element.

when we use position: relative if any space left towards the leftside of the element will be blank and cannot be filled with any other element.

The left, right, top and bottom attributes will be taken in to account relative to the current page flow.

```
<style>
  div.relative {
    position: relative;
    left: 30px;
    border: 1px solid black;
  }
</style>

<div>
  CSS Position Property
</div>
<div class="relative">
  Position can be used for changing the position of page elements
</div>
```

4. fixed

fixed position elements are positioned relative to the viewport. these always stays in the fixed position even we scroll the page as well and will not leave any gap like relative elements

```
<style>
  body {
      margin-bottom: 2000px;
  }
  div.fixedposition {
      position: fixed;
      border: 1px solid black;
      bottom: 0;
      width: 55%;
  }
</style>
<div class="fixedposition">
  One day, a thief who knew the old miser’s routine, waited for the old man to go back into his house
</div>

2. absolute
is positioned relative to the nearest parent and if there is no parent element then it will be placed relative to the viewport of the page and behaves as fixed but scrolls along with the page.


<div style="position: relative">
    One day, a thief who knew the old miser’s routine, waited for the old man to go back into his house
    <div style="position: absolute; top: 30px">
      He continued this routine every day, but not once did he spend the gold he saved.
    </div>
</div>

5. sticky
sticky is a position used for sticking the page elements on user scroll position.
<style>
  body {
    margin-bottom: 2000px;
  }
  div.stickyposition {
    position: sticky;
    top: 0;
    border: 1px solid black;
    width: 100%;
    height: 40px;
  }
</style>

<body>
  <div>
      One day, a thief who knew the old miser’s routine, waited for the old man to go back into his house
  </div>
  <div class="stickyposition">
    He continued this routine every day, but not once did he spend the gold he saved.
  </div>
  <div>
```

```
    is positioned relative to the nearest parent and if there is no parent element then it will be placed relative to the viewport of the page and behaves as fixed but scrolls along with the page.
  </div>
</body>
------------------------------------------------------------------------------------------------------------
position = we can change the position of the page elements using css position property
  - static = default, places the elements according to the page flow. left|right|top|bottom are not applied
  - relative = places the elements according to the default page flow and left|right|top|bottom will changes the element position in the page. but any space created to the left of the element cannot be fixed.
  - fixed = always fixed to the current viewport, if scrolled also there is no affect
  - absolute = used for nested elements, where we want to place a child element relative to the parent we usually use nested. if not a parent element it is positioned to the viewport, but scrolls along with the page.
  - sticky = element sticks to user scroll position. we tell at which place in the page the element should stick.
-----------------------------------------------------------------------------------------------------------
css overflow
overflow css property is used for managing the contents of a block level element when the content is more than the size of the element.
note:-
overflow is applied only to block level elements only
for element to apply overflow property we should have fixed height

overflow property can have the below values
1. visible
2. hidden
3. scroll
4. auto

<style>
  .para {
    border: 1px solid black;
    height: 200px;
    width: 200px;
    overflow: visible;
  }
</style>

```

```html
<p class="para">
  Once upon a time, there lived a shepherd boy who was bored watching his flock of sheep on the hill. To amuse himself, he shouted, “Wolf! Wolf! The sheep are being chased by the wolf!” The villagers came running to help the boy and save the sheep. They found nothing and the boy just laughed looking at their angry faces
</p>
```

by default the content overflow of the element area will be visible.
hidden = will hides the content and displays only the content that fits to the area
scroll = will generates scroll bar if the contents of the element is more than the size of the element
auto = based on the contents it generates either veritical or horizantal scroll bar
overflow-x: can be used for enabling horizantal scrollbar
overflow-y: can be used for enabling vertical scrollbar.

----

css float
to float the images around the content we use float property. we can float an image to the left of the text or right of the text using css float property. used for positioning and formatting.

```html
<p>
  <img src="../images/p1.jpg">
  Once upon a time, there lived a shepherd boy who was bored watching his flock of sheep on the hill. To amuse himself, he shouted, “Wolf! Wolf! The sheep are being chased by the wolf!” The villagers came running to help the boy and save the sheep. They found nothing and the boy just laughed looking at their angry faces. “Don’t cry ‘wolf’ when there’s no wolf boy!”, they said angrily and left. The boy just laughed at them.
</p>
```

Horizantal Navigation bar

```html
<body>
  <ul>
    <li>Our Services</li>
    <li>Careers</li>
    <li>Business Integrations</li>
    <li>About Us</li>
    <li>Contact Us</li>
  </ul>

  <h2>Bluedart</h2>
  <p>
    Paragraph
  </p>
</body>
-------------------------------------------------------------------------------------------------------------
```

How to style a select control in html?

```
<select name="city">
    <option value=""></option>
</select>
```

Intead of using the pre-defined select control of HTML we can create our own dropdowns using css.

```
<div>

</div>

-------------------------------------------------------------------------------------------------------------
```