# 4

# CSS

# सीएसएस

Web Designing and Publishing

## 4.1 Introduction to CSS - सीएसएस एक परिचय

A CSS (cascading style sheet) file allows you to separate your web sites(X) HTML content from its style. As always you use your (X) HTML file to arrange the content, but all of the presentation (fonts, colors, background, borders, text formatting, link effects & so on...) are accomplished within a CSS. At this point you have some choices of how to use the CSS, either internally or externally.

कैस्केडिंग स्टाइल शीट या संक्षिप्त में सीएसएस एक प्रकार का डीएचटीएमएल कंपोनेंट होता है जो किसी वेब पेज में कंटेंट्स जैसे टेक्स्ट, इमेजेज, ग्राफिक्स, टेबल्स इत्यादि को फॉर्मेट एवं अलाइन (व्यवस्थापन) करने के काम में आती है। सीएसएस के भी अपने कुछ क्लासेज एवं एलीमेंट्स होते हैं जिन्हें प्रयोग कर हम किसी वेब पेज को डायनामिक बनाते हैं। यहां हम सीएसएस को कैसे इंटर्नली या एक्सटर्नली यूज करेंगे इसके बारे में जानेंगे।

In other words we can say that it is the first technology you should start learning after HTML. It is used to format the layout's of web pages in the website. By using this we can define the text styles, table sizes and other aspects of Web pages that previously could only be defined in a HTML pages. While HTML is used to define the structure and semantics of your content. CSS describes how HTML elements are to be displayed on screen, paper or in other media.

दूसरे शब्दों मे हम यह कह सकते है कि यह ऐसी पहली टेक्नोलॉजी है जिसे आप HTML के बाद सीखना प्रारम्भ करते हैं। इसका प्रयोग वेबसाइट में वेब पेजों के लेआउट को फॉर्मेट करने के लिए किया जाता है। इसका प्रयोग करके हम टेक्स्ट स्टाईल, टेबल साइज और वेब पेजों के अन्य पहलुओं को परिभाषित कर सकते हैं जो पहले केवल एक HTML पेजों में परिभाषित किए जाते थे। जबकि HTML आपके कंटेट के स्ट्रक्चर आदि को डिफाईन करने के लिए प्रयोग किया जाता है। CSS यह बताता है कि HTML एलिमेन्टो को स्क्रीन, पेपर या अन्य मीडिया में कैसे प्रदर्शित किया जाना है।

Before starting CSS, we have to know the purpose. HTML was never intended to contain tags for formatting a web page. HTML was created to describe the content of a web page, like headings or paragraph etc.

CSS प्रारम्भ करने से पहले हमें इसके उद्देश्य को जान लेना चाहिए। HTML मे वेब पेजों के फ़ार्मेटिंग के लिए टैग्स को शामिल नहीं किया गया था। इसे केवल वेब पेज के कंटेंट को डिस्क्राइब करने के लिए इसे क्रिएट किया गया था जैसे हेडिंग या पैराग्राफ आदि।

```
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
```

Before CSS, If we want to assign a typographic characteristics to say, all H2 headings had to repeat HTML presentational markup for each occurrence of that heading type. This made documents more complex, larger, more error-prone and is difficult to maintain.

CSS से पहले, यदि हम यह कहने के लिए एक टाइपोग्राफिक विशेषताओं को निर्दिष्ट करना चाहते हैं, तो सभी H2 हेडिंग्स को उस शीर्षक प्रकार की प्रत्येक घटना के लिए HTML प्रेजेंटेशनल मार्कअप दोहराना था। इसने डॉक्यूमेंटों को अधिक जटिल, बड़ा, अधिक एरर–प्रोन बना दिया और इसे बनाए रखना मुश्किल है।

To solve this problem, the World Wide Web Consortium (W3C) created CSS. It removed the style formatting from the HTML page! It also Saves a Lot of Work and Time! The style definitions are normally saved in external .css files. With an external CSS file, you can change the look of an entire website by changing just one file at a time.

इस प्राब्लम को सॉल्व करने के लिए, वर्ल्ड वाइड वेब कॉन्सोर्शियम ने सीएसएस बनाया। यह एचटीएमएल पेजों से स्टाइल फार्मेटिंग को रिमूव कर दिया। जिससे हम अपने वर्क एंड टाइम को बचा सकते है। सभी स्टाइल डेफिनेसन को हम एक्सटर्नल डॉट सीएसएस फाइल मे सेव करते है। आप इस एक्सटर्नल डॉट सीएसएस फाइल से अपने वेबसाइट के लूक को आसानी से एक ही बार में एक ही फाइल से मैनेज कर सकते है।

CSS allows the separation of presentation from structure. CSS can define color, font, text alignment, size, borders, spacing, layout and many other typographic characteristics and can do so independently for on-screen and printed views. CSS also defines non-visual styles, such as reading speed.

CSS structure से प्रेजन्टेशन को अलग करने की अनुमति देता है। CSS रंग, फ़ॉन्ट, टेक्स्ट एलाइनमेंट, साइज, बार्डर, स्पेसिंग, लेआउट और कई अन्य टाइपोग्राफिक विशेषताओं को डिफाइन कर सकता है और ऑन–स्क्रीन और प्रिन्टेड व्यू के लिए स्वतंत्र रूप से ऐसा कर सकता है। CSS नॉन–विजुअल स्टाइल को भी डिफाइन करता है, जैसे पढ़ने की गति।

**Example1:** If you want to create a heading element with green text using HTML code would be written as:-

एचटीएमएल कोड द्वारा हेडिंग एलिमेंट के टेक्स्ट को ग्रीन कलर करना चाहते है तो यह इस प्रकार लिखा जाएगा:

```
<h2><font color="green"> Welcome to TBalaji Publication </font></h2>
```

**Example2:** If you want create the same heading element with green text and having css style then the CSS code is written as:

यदि आंप उसी हेडिंग एलिमेंट के टेक्स्ट को सीएसएस यूज कर ग्रीन कलर करना चाहते है तो नीचे लिखे सीएसएस कोड को लिखे:

```
<style>
h2 {
   color: green;
}
</style>
```

Now, All h2 elements in the document will then automatically become **green** without requiring any explicit code. But if you later wanted to make **h2 elements red instead**, then this could be done by changing the style element rather than with difficulty going through the web document and changing the color for each individual h2 element.

अब डाक्यूमेंट मे बिना किसी दूसरे कोड के सभी h2 एलिमेंट अपने आप ग्रीन कलर में हो जाएंगे। अगर आप बाद मे h2 एलिमेंट को ग्रीन कलर की जगह रेड करना चाहते है तो आप सिर्फ सीएसएस फाइल (स्टाइल एलिमेंट) मे जाकर h2 हेडिंग का कलर रेड सेट कर सकते है जिससे सभी h2 हेडिंग के कलर रेड हो जाएंगे जैसा की यहां दिखाया गया है:–

```
<style>
h2 {color: red; }
</style>
```

## 4.2 Types of CSS - सीएसएस के प्रकार

**There are three different ways you can use CSS in your web page. These are as follows:**
आपके वेब पेज में सीएसएस का उपयोग करने के तीन अलग–अलग तरीके हैं। ये इस प्रकार हैं:

### 1. Inline (Right next to the text it decorates):
### इनलाइन (इसे सजाने वाले टेक्स्ट के ठीक बगल में)

It helps you to apply unique style rules in specific HTML elements. The style attribute specifies properties and values. Inline styles are placed right where you need them, next to the text or graphic you wish to decorate. You can insert inline styles anywhere in the middle of your HTML code, giving you real freedom to specify each web page element. To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

यह HTML एलीमेंट को निर्दिष्ट करने में यूनिक स्टाईल नियम लागू करने में आपकी सहायता करता है। स्टाईल एट्रीब्यूट प्रॉपर्टीज और मूल्यों को निर्दिष्ट करती है। इनलाइन स्टाइलों को ठीक उसी स्थान पर रखा जाता है जहाँ आपको उनकी आवश्यकता होती है, जिस टेक्स्ट या ग्राफ़िक को आप सजाने की इच्छा रखते हैं। आप अपने HTML कोड के बीच में कहीं भी इनलाइन स्टाइलो को सम्मिलित कर सकते हैं, जिससे आपको प्रत्येक वेब पेज एलिमेंट को स्पेशिफाई करने की वास्तविक स्वतंत्रता मिलती है। इनलाइन स्टाईल का उपयोग करने के लिए, संबंधित एलिमेंट में स्टाईल ऐट्रीब्यूट को जोड़ें। स्टाईल एट्रीब्यूट में कोई भी सीएसएस गुण हो सकते हैं।

**The example below shows how to change the color and the left margin of a <h1> element:**
**नीचे दिए गए उदाहरण से पता चलता है कि रंग और <h1> एलीमेंट के बाएं मार्जिन को कैसे बदलना है:**

Inline.html

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:green">This is a heading</h1>
<p style="color:blue">This is a paragraph.</p>

</body>
</html>
```

inline.html

file:///C:/Users/babaji/Desktop/css%20demos/inline.html

**This is a heading**

This is a paragraph.

### 2. Internal (At the top of a web page document)
### इंटर्नल (एक वेब पेज डॉक्यूमेंट के शीर्ष पर)

Internal styles are located at the top of each web page, before any of the content is listed. This is the next best thing to external, because they're easy to find, yet permit you to 'override' an external style sheet - for that special page that wants to be an individualist. Internal styles are defined within the **<style>** element, inside the **<head>** section of an HTML page like:

इंटर्नल स्टाइल प्रत्येक वेब पेज के शीर्ष पर स्थित होती है, इससे पहले कि कोई भी कन्टेंट सूचीबद्ध हो। यह एक्सटर्नल के लिए अगली सबसे अच्छी बात है, क्योंकि इन्हें खोजना आसान है, फिर भी आपको बाहरी स्टाईल शीट को 'ओवरराइड' करने की अनुमति है– उस विशेष पेज के लिए जो एक इंडिविजुअल बनना चाहता है। इंटर्नल स्टाईल को <style> एलिमेन्ट के भीतर डिफाईन किया गया है, जैसे HTML पेज के <head> सेक्शन के अंदर:

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
   background-color: light-green;
   font-size:20px;
}

h1 {
   color: maroon;
   margin-left: 10px;
}
</style>
</head>
<body>

<h1>Internal Style Sheet</h1>
<p>Internal styles are located at the top of each web page, before any of the content is listed. This is the next best thing to external, because they're easy to find, yet permit you to 'override' an external style sheet - for that special page that wants to be an individualist. </p>

</body>
</html>
```

**Run the program and it looks like in the browser:**

**प्रोग्राम को चलाएं और यह ब्राउजर में जैसा दिखता है:**

insidepage.html

file:///C:/Users/babaji/Desktop/css%20demos/insidepage.html

# Internal Style Sheet

Internal styles are located at the top of each web page, before any of the content is listed. This is the next best thing to external, because they're easy to find, yet permit you to 'override' an external style sheet - for that special page that wants to be an individualist.

## 3. External (In a separate file)

## एक्सटर्नल (एक अलग फ़ाइल में)

An external CSS file can be created with any text or HTML editor such as "Notepad", "Notepad++" or "Dreamweaver". A CSS file contains no (X) HTML, only CSS. You simply save it with the *.css* file extension. You can link to the file externally by placing one of the following links in the head section of every (X) HTML file you want to style with the CSS file.

एक एक्सटर्नल सीएसएस फाइल किसी भी टेक्स्ट या एचटीएमएल एडीटर जैसे "नोटपैड", "नोटपैड++" या "ड्रीमविवर" के साथ बनाई जा सकती है। एक सीएसएस फाइल में कोई (X) एचटीएमएल नहीं है, केवल सीएसएस है। आप बस .css फाइल एक्सटेंशन के साथ इसे सेव कर सकते हैं। आप प्रत्येक (X) एचटीएमएल फाइल के सीएसएस अनुभाग में निम्नलिखित लिंक में से एक को बाहरी फाइल में लिंक कर सकते हैं, जिसे आप सीएसएस फाइल के साथ स्टाईल करना चाहते हैं।

**For example:** An external style sheet can be written in any text editor. The file should not contain any html tags. The style sheet file must be saved with a ***.css** extension.

किसी भी टेक्स्ट एडिटर में एक बाहरी स्टाइल शीट लिखी जा सकती है। फ़ाइल में कोई HTML टैग नहीं होना चाहिए। स्टाइल शीट फ़ाइल को एक .css एक्सटेंशन के साथ save किया जाना चाहिए।

**Step 1:** Open **Notepad++** application and type the following lines.

Notepad++ एप्लिकेशन खोलें और नीचे लिखें लाईन को टाइप करें।

C:\Users\babaji\Desktop\css demos\myFirstStyleSheet.css - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

myFirstStyleSheet.css

```
1  body {
2      background-color: lightblue;
3  }
4  h1 {
5      color: navy;
6      margin-left: 20px;
7  }
```

length : 88 Ln : 7 Col : 2 Sel : 0 | 0 Windows (CR LF) UTF-8 INS

**Note:** Do not add a space between the property value and the unit (such as margin-left: **20 px**;).
**The correct way is:** margin-left: **20px**;
प्रॉपर्टी वैल्यू और यूनिट (जैसे मार्जिन–लेफ्टः 20 px;) के बीच स्पेस न जोड़ें। सही तरीका हैः मार्जिन–लेफ्टः 20px;

**Step 2:** Save this file with the name of ***"myFirstStyleSheet.css"***
इस फ़ाइल को "***myFirstStyleSheet.css***" के नाम से सेव करें

**Step 3:** Open Notepad++ application write HTML document. Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section.

Notepad++ एप्लिकेशन खोलें HTML डॉक्यूमेंट लिखे। प्रत्येक पृष्ठ में <link> एलिमेंट के अंदर बाहरी स्टाइल शीट फ़ाइल का रेफरेन्स शामिल होना चाहिए। <link> एलिमेंट <head> सेक्शन के अंदर होता है:

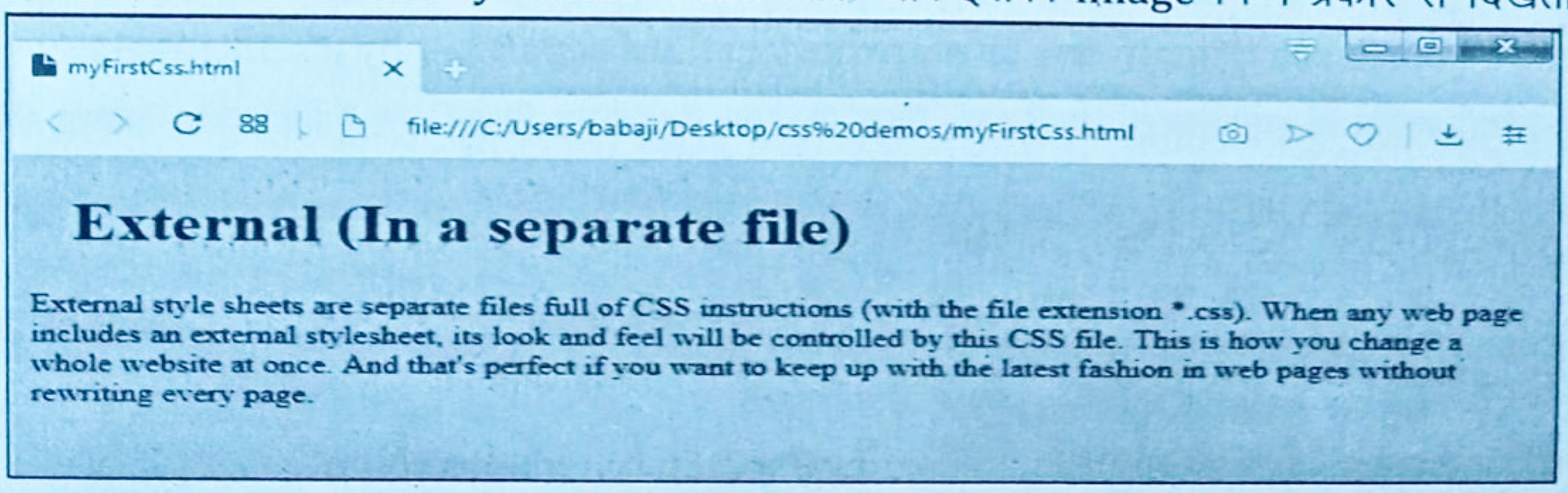
myFirstCss.html

```
<!DOCTYPE html>
<html>
<head>
    <link rel="stylesheet" type="text/css" href="myFirstStyleSheet.css">
</head>
<body>

<h1> External (In a separate file)</h1>
<p> External style sheets are separate files full of CSS instructions (with the file extension *.css).
When any web page includes an external stylesheet, its look and feel will be controlled by this CSS
file. This is how you change a whole website at once. And that's perfect if you want to keep up with the
latest fashion in web pages without rewriting every page.</p>
</body>
</html>
```

**Step 4:** Save the file with name of "myFirstCSS.html" where **myFirstStyleSheet.css** file is stored.
फ़ाइल को myFirstCSS.html के नाम से सेव करें जहाँ myFirstStyleSheet.css फ़ाइल स्टोर है।

**Step 5:** Open any browser and open **myFirstCss.html** and it looks like the following image...
किसी भी ब्राउज़र को खोलें और myFirstCss.html खोलें और इसकी image निम्न प्रकार से दिखती है...



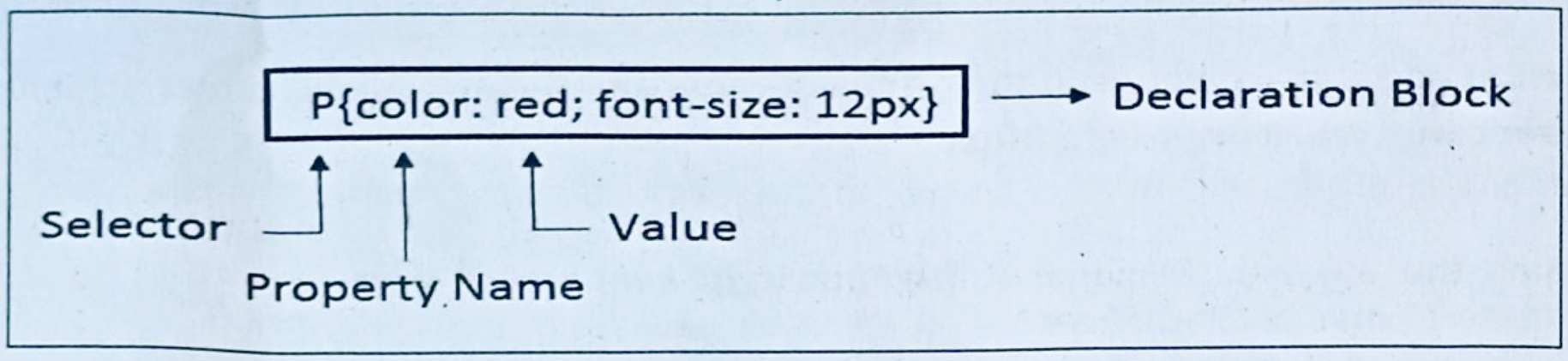
## 4.3 CSS Selectors - सीएसएस सेलेक्टर्स

The CSS selector is used to select the content that you want to style. It select the HTML element according to its id, class, type, attribute etc. The selectors are the part of css rule set which contains the selector name and the declaration block.

सीएसएस सेलेक्टर उस कंटैंट को सलेक्ट करने का काम करता है जिसे आप स्टाइल करना चाहते है। यह एचटीएमएल एलीमेंट को इसके आईडी, क्लास, टाइप, एवं एट्रीब्यूट द्वारा इसे सलेक्ट करता है। सलेक्टर्स सीएसएस रूल सेट का एक पार्ट है जिसमे सलेक्टर नेम और डिक्लेरेशन ब्लॉक होता है।

A CSS rule set contains a **selector** and a **declaration** block which is shown below:

एक CSS नियम सेट में एक सेलेक्टर और एक डिक्लेरेसन (घोषणा) ब्लॉक होता है।




The selector points to the HTML element you want to style. The declaration block contains one or more declarations separated by semicolons (;). Each declaration includes a CSS property name and a value, separated by a colon (:). A CSS declaration always ends with a semicolon and declaration blocks are surrounded by curly braces ({}).

सेलेक्टर उस HTML एलिमेंट को इंडिकेट करता है जिसे आप style करना चाहते हैं। डिक्लेरेसन ब्लॉक में सेमीकोलन (;) द्वारा अलग किए गए एक या एक से अधिक डिक्लेरेसन शामिल है। प्रत्येक डिक्लेरेसन (घोषणा) में एक सीएसएस प्रॉपर्टी का नाम और एक वैल्यू शामिल होती है, जिसे एक colon (:) द्वारा अलग किया जाता है। एक सीएसएस डिक्लेरेसन (घोषणा) हमेशा एक सेमीकोलन के साथ समाप्त होता है, और डिक्लेरेसन (घोषणा) ब्लॉक curly braces ({}) से घिरे होते हैं।

### 4.3.1 CSS Syntax - सीएसएस सिंटैक्स

**The CSS syntax consists of a selector and a declaration block which is given below:**

**CSS सिंटैक्स में एक सेलेक्टर और एक डिक्लेशन ब्लॉक होता है जो नीचे दिया गया है:**

**Selector:** It indicates the HTML element you want to style. It could be any tag like <h1>, <title> etc.
सेलेक्टर उस एचटीएमएल एलिमेंट को इंडिकेट करता हैं जिसको आप स्टाइल करना चाहते है यह कोई टैग हो सकता है जैसे <h1>, <title>.

**Declaration Block: डिक्लेरेसन ब्लॉक :**

The declaration block can contain one or more declarations separated by a semicolons.

डिक्लेरेसन (घोषणा) ब्लॉक में सेमीकोलन द्वारा अलग किए गए एक या अधिक डिक्लेरेसन (घोषणा) हो सकता हैं।

The example has two declarations:

उदाहरण में दो डिक्लेरेसन (घोषणाए) हैं:

```
color: red;
font-size: 12px;
```

Each declaration contains a property **name** and **value**, separated by a **colon**.

प्रत्येक घोषणा में एक Property का नाम और वैल्यू होता है, जो एक **colon** द्वारा अलग होता है।

**Property Name: प्रॉपटी नेम**

A Property is a type of attribute of HTML element. It could be color, border etc.
एक प्रॉपर्टी HTML एलिमेंट के ऐट्रीब्यूट का एक प्रकार है। यह कलर, बार्डर आदि हो सकता है।

**Value:** Values are assigned to CSS properties. In the above example, value "red" is assigned to color property.
**वैल्यू–** वैल्यू सीएसएस प्रापर्टीज को असाइन किया गया है। उपरोक्त उदाहरण में, वैल्यू "red" को कलर प्रॉपर्टी से असाइन किया गया है।

```
Selector {Property1: value1; Property2: value2; .....;}
```

CSS selectors are used to select the content you want to style. Selectors are the part of CSS rule set. CSS selectors select HTML elements according to its id, class, type, attribute etc.

CSS सेलेक्टर का उपयोग उस कन्टेंट का चयन करने के लिए किया जाता है जिसे आप style करना चाहते हैं। सेलेक्टर CSS नियम सेट का हिस्सा हैं। CSS सेलेक्टर अपनी id, क्लास, टाईप, ऐट्रीब्यूट आदि के अनुसार HTML एलिमेंटो को सेलेक्ट करते हैं।

## Types of CSS selectors - सीएसएस सेलेक्टर्स के प्रकार

**There are several different types of selectors in CSS.**
**CSS में कई अलग–अलग प्रकार के सेलेक्टर हैं।**

1. Universal Selector - यूनिवर्स सेलेक्टर
2. ID Selector - आईडी सेलेक्टर
3. Tag Selector - टैग सेलेक्टर
4. Class Selector - क्लास सेलेक्टर
5. Sub Selector - सब सेलेक्टर
6. Attribute Selector - एट्रीब्यूट सेलेक्टर
7. Group Selector - ग्रुप सेलेक्टर

## 4.4 Universal Selector - यूनिवर्सल सेलेक्टर

The universal selector, written as * i.e. **asterisk** or **star symbol**, matches every single element on the page. The universal selector may be omitted if other conditions exist on the target element. This selector is often used to remove the default margins and paddings from the elements for quick testing purpose.

यूनिवर्सल सेलेक्टर, जिसे * यानी एसट्रिक चिह्न या स्टार सिंबल के रूप में लिखा जाता है, पेज पर मौजूद प्रत्येक एक एलिमेंट से मेल खाता है। अगर टारगेट एलिमेंट पर अन्य कंडीशन मौजूद हैं, तो यूनिवर्सल सेलेक्टर को छोड़ा जा सकता है। इस सेलेक्टर का उपयोग अक्सर त्वरित परीक्षण उद्देश्य के लिए एलिमेंटो से डिफॉल्ट मार्जिन और पैडिंग को हटाने के लिए किया जाता है।

**Syntax:**

```
* {
    style properties
}
```

**Example:**

```
* {
  color: blue;
}
```

**The asterisk may be used in combination with namespaces:**

**नेमस्पेस के साथ कॉम्बिनेशन में तारांकन का उपयोग किया जा सकता है:**

**ns**|* - matches all elements in namespace ns
*|* - matches all elements
|* - matches all elements without any declared namespace

The style rules inside the * selector will be applied to every element in a document.

*सेलेक्टर के अंदर की style के नियम एक डॉक्यूमेंट में हर एलिमेंट पर लागू होंगे।

**Note:** It is not recommended to use the universal selector * in a production environment, since this selector matches every element on a page that puts too much of unnecessary pressure on the browsers.

यह एक प्रोडक्शन इनवॉयरमेंट में यूनिवर्सल सेलेक्टर का उपयोग करने के लिए रिकमेन्डेड नहीं है, क्योंकि यह सेलेक्टर एक पेज पर प्रत्येक एलिमेंट से मेल खाता है जो ब्राउज़रों पर बहुत अधिक अनावश्यक दबाव डालता है।

**Example:**

h1 h2 h3 ¶p b i u span link img ul •li

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <style>
5  * {
6    color: green;
7    font-size: 20px;
8  }
9  </style>
10 </head>
11 <body>
12 <h2>Universal Selector</h2>
13 <p>This style will be applied on every paragraph.</p>
14 <p id="para1">Second Paragraph</p>
15 <p>and Third Paragraph</p>
16 </body>
17 </html>
```

PREVIEW

Universal Selector

This style will be applied on every paragraph.

Second Paragraph

and Third Paragraph

## 4.5 ID selector - आईडी सेलेक्टर

This selector is used to select the id attribute of an HTML element for selecting the specific element. An id is always unique within the page so it is chosen to select a single, unique element. This selector is always written with the (#) character followed by id of the element. Thus, the ID style is used to identify the sidebar section of a page could look like this.

आईडी सेलेक्टर एक विशिष्ट एलिमेंट को सेलेक्ट करने के लिए एक एचटीएमएल एलिमेंट की आईडी ऐट्रीब्यूट को सेलेक्ट करता है। एक आईडी हमेशा पेज के भीतर यूनिक होती है इसलिए इसे सिंगल, यूनिक एलिमेंट को सेलेक्ट करने के लिए चुना जाता है। इस hash character (#) के बाद आईडी एलिमेन्ट को लिखा गया है। इस प्रकार एक पेज के साइडबार सेक्शन की पहचान करने के लिए उपयोग की जाने वाली आईडी स्टाईल निम्न प्रकार से दिख सकती है।

**Note:** Don't include any space between the # and the style name.

**नोट:** # और स्टाईल नेम के बीच कोई स्पेस शामिल नहीं है।

Let's take an example with the id "id1".

आइए आईडी "id1" के साथ एक उदाहरण लें।

```
h1 h2 h3 p b i u span link img ul li
<!DOCTYPE html>                                                    </h1>
<html>
<head>
<style>
#id1{
   color: red;
   text-align: center;

}
</style>
</head>
<body>
   <p id="id1">ID Selector</p>
   <p>This paragraph is not Affected.</p>
   <p id="id1">This paragraph is Affected.</p>
</body>
</html>
```

PREVIEW

ID Selector

This paragraph is not Affected.

This paragraph is Affected.

What's special about ID styles is that they should be used only once per page. ID selectors are great for creating page layouts where you want to define each section of a page only once; they are ideal for creating a CSS layout.

आईडी स्टाईल के बारे में विशेष बात यह है कि उन्हें प्रत्येक पेज पर केवल एक बार उपयोग किया जाना चाहिए। आईडी सेलेक्टर पेज लेआउट बनाने के लिए सबसे अच्छा हैं जहां आप किसी पेज के प्रत्येक सेक्शन को केवल एक बार परिभाषित करना चाहते हैं; वे CSS लेआउट बनाने के लिए आदर्श हैं।

## 4.6 Tag Selector- टैग सेलेक्टर

The **tag selector** is used to redefine existing HTML tags. Select this option if you want to change the formatting options for an HTML tag, such as the <h1> (heading 1) tag or the <ul> (unordered list) tag. In many cases, redefining existing HTML tags with CSS has advantages over creating new styles. For example, content formatted with the H1 tag is well recognized on the Web as the most important text on a page. For that reason, many search engines give priority to text formatted with the H1 tag. Similarly, the hierarchical structure of the <h1> to <h6> tags helps ensure that even if a visitor to your site changes the text size in his or her Web browser, text formatted with the H1 tag is still larger relative to text formatted with an H2 tag, which is larger than text formatted with the H3 tag.

टैग सेलेक्टर का उपयोग मौजूदा HTML टैग को पुनः परिभाषित करने के लिए किया जाता है। यदि आप HTML टैग के लिए फार्मेटिंग विकल्पों को बदलना चाहते हैं, तो यह विकल्प चुनें, जैसे कि <h1> (heading 1) टैग या <ul> (unordered list) टैग। कई दशाओं में, CSS के साथ मौजूदा HTML टैग को पुनः परिभाषित करना नई style को बनाने में लाभदायक हैं। उदाहरण के लिए, h1 टैग के साथ format कन्टेंट वेब पर एक पेज पर सबसे महत्वपूर्ण टेक्स्ट के रूप में अच्छी तरह से पहचानी जाती है। इस कारण, कई सर्च इंजन h1 टैग के साथ टेक्स्ट फारमेट को प्राथमिकता देते हैं। इसी तरह, <h1> से <h6> टैग की पदानुक्रमित संरचना यह सुनिश्चित करने में मदद करती है कि भले ही आपकी साइट पर कोई विज़िटर अपने वेब ब्राउज़र में टेक्स्ट का आकार बदलता है, एच 1 टैग के साथ फार्मेट टेक्स्ट अभी भी H2 टैग के साथ फार्मेटेड टेक्स्ट के सापेक्ष बड़ा है, ज़ो H3 टैग के साथ टेक्स्ट फार्मेट से बड़ा है।

The ability to change the appearance of headings and other tags with CSS makes it possible to retain these advantages while still being able to use the font, size, color, spacing and other formatting options that you prefer in your Web design. When you use the **tag selector**, the style definition is applied automatically to any text or other element that's been formatted with the corresponding tag. Thus, if you've formatted a headline with the <h1> tag and then create a new <h1> style, the formatting you used to define the style will apply automatically to the headline as soon as the style is created.

सीएसएस के साथ हेडिंग और अन्य टैग की उपस्थिति को बदलने की क्षमता, इन advantages को बनाए रखना संभ बनाती है जबकि अभी भी फ़ॉन्ट, आकार, रंग, स्पेसिंग और अन्य फार्मेटिंग विकल्पों का उपयोग करने में सक्षम है जो आ अपने वेब डिज़ाइन में पसंद करते हैं। जब आप टैग सेलेक्टर का उपयोग करते हैं, तो style definition किसी भी टेक्स या अन्य एलिमेन्ट के लिए स्वतः रूप से लागू होती है जिसे संबंधित टैग के साथ format किया गया है। इस प्रकार, यदि आपने <h1> टैग के साथ एक शीर्षक (हेडलाईन) तैयार किया है और फिर एक नई <h1> स्टाइल बनाते हैं, तो स्टाइल को डिफाईन करने के लिए आपने जिस format का उपयोग किया है, वह स्टाइल बनाते ही शीर्षक (हेडलाईन) पर स्वत लागू हो जाएगा।

## 4.7 Class Selector - क्लास सेलेक्टर

It is used to define any HTML element which has the class attribute's. The element having that class will be formatted/style according to the given rule. It can be defined by using "." (full stop) symbol along with the class name. For example: All HTML element with class name="abc" will be blue and left-aligned.

```
.abc {
        color: blue;
        text-align: left;
        }
```

क्लास सेलेक्टर का उपयोग किसी भी HTML एलिमेन्ट को डिफाइन करने के लिए किया जा सकता है जिसमें एक क्लास ऐट्रीब्यूट होता है। डिफाईन्ड नियम के अनुसार उस क्लास वाले सभी एलिमेन्टो को format किया जाएगा। क्लास सेलेक्टर को फुल स्टाप सिम्बल (.) के साथ परिभाषित किया गया है। उदाहरण के लिए, सभी एचटीएमएल एलिमेंट जिसमें क्लास नेम abc होगा वो ब्लू कलर एवं लेफ्ट एलाइंगमेंट मे होगा।

**Note:** A class name should not be started with a number.

एक क्लास का नाम एक संख्या के साथ शुरू नहीं किया जाना चाहिए।

**Syntax:**

```
.class_name
{
    style properties
}
```

**Example 1:**

```
<!DOCTYPE html>
<html>
<head>
<title>Class Selector</title>
<style type="text/css">
.abc{
   color:red;
   text-align: center;
}
</style>
</head>
<body>
   <h1 class="abc">This is Heading </h1>
   <p class="abc">First Paragraph</p>
   <p class="abc">Second Paragraph</p>
</body>
</html>
```

PREVIEW

This is Heading

First Paragraph

Second Paragraph

If you want to specify that only one specific HTML element should be affected then you should use the element name with class selector.

यदि आप यह स्पेशिफाई करना चाहते हैं कि केवल एक विशिष्ट HTML एलिमेन्ट प्रभावित होना चाहिए तो आपको क्लास सेलेक्टर के साथ एलिमेन्ट नाम का उपयोग करना चाहिए।