

সিএসএস

বাংলা

ইবুক

CSS



Written By

আবদুল্লাহ আল ফারুক

Published By : WordPress Group

<http://Ebooks.WpBangla.com>

সিএসএস বাংলা ই-বুক
CSS Bangla E-book
(Specially For Web Designers)

মোঃ আবদুল্লাহ আল-ফারুক

<https://www.facebook.com/faruk.ice09>

<http://www.WebTechnologyBlog.com>

<http://www.farukbd.info>

প্রথম প্রকাশ
২৯ জানুয়ারি ২০১৩

প্রকাশক
ওয়ার্ডপ্রেস গ্রুপ বাংলাদেশ
(<https://www.facebook.com/groups/Wordpress2Smashing>)

ওয়ার্ডপ্রেস গ্রুপ ওয়েবসাইট
(<http://www.wpbangla.com>)

লেখক
মোঃ আবদুল্লাহ আল-ফারুক
(<https://www.facebook.com/faruk.ice09>)
(<http://www.WebTechnologyBlog.com>)
(<http://www.farukbd.info>)

প্রচ্ছদ
জামিল হোসেন সিজান
(<https://www.facebook.com/zamil.hossainsezan32>)

কপিরাইট
আবদুল্লাহ আল-ফারুক ও ওয়ার্ডপ্রেস গ্রুপ বাংলাদেশ

সতর্কতা
এই বইটি বিক্রয়ের জন্য নয়
বইটি বিনামূল্যে বিতরণযোগ্য

CSS Bangla E-book is written by MD.Abdullah Al Faruk & Published by Wordpress Group Bangladesh. Copyright by MD.Abdullah Al Faruk & Wordpress Group Bangladesh. Caution: **This book is not for sell. It's free to distribute.**

উৎসর্গ

শ্রদ্ধেয় আব্বু ও আম্মুকে.....
যাদের অক্লান্ত পরিশ্রমে আমি এতদূর আসতে পেরেছি।

আমার সম্পর্কে-

আমি মোঃ আবদুল্লাহ আল-ফারুক(<https://www.facebook.com/faruk.ice09>) পড়ালেখা করছি কুষ্টিয়া ইসলামী বিশ্ববিদ্যালয়ের ইনফরমেশন এন্ড কমিউনিকেশন ইঞ্জিনিয়ারিং (ICE) বিভাগের তৃতীয় বর্ষে। আমার লেখা এইচটিএমএল বইটি প্রকাশ হবার পর অনেকেই আমাকে দ্রুত সিএসএস বইটা প্রকাশ করার জন্য অনুরোধ করেছিলেন। তাই সবার কথা মাথায় রেখে অনেক ব্যস্ততার মাঝেও সিএসএস বইটি লিখে শেষ করলাম। দ্রুত শেষ করার কারণে কিছু ভুলত্রুটি থাকতে পারে আশা করি ক্ষমা সুন্দর দৃষ্টিতে দেখবেন। আর এই বইটি যদি আপনাদের শেখার কাজে সামান্য উপকারে লাগে তবে নিজেকে ধন্য ও আমার পরিশ্রম সার্থক হয়েছে বলে মনে করবো। আমি বিশেষভাবে ধন্যবাদ জানাই জামিল হোসেন সিজান (<https://www.facebook.com/zamil.hossainsezan32>) ও ওয়ার্ডপ্রেস গ্রুপ বাংলাদেশ (<https://www.facebook.com/groups/Wordpress2Smashing/>) কে।

এই বইটির স্বব্বস্বত্ব আমার। অনুগ্রহ করে অনুমতি ছাড়া এই বইটির আংশিক বা সম্পূর্ণ কপি বা বিকৃত বা নিজের নামে চালিয়ে দেয়ার চেষ্টা করবেন না। আপনাদের নিজ নিজ ব্লগের মাধ্যমে বইটি সবার মাঝে ছড়িয়ে দিন। শেয়ার করুন সবার সাথে। জ্ঞান বিতরণে কমে না, তাই যে যা জানেন সবার সাথে শেয়ার করুন।

-আবদুল্লাহ আল-ফারুক

সূচিপত্র

অধ্যায়-একঃ সাধারণ আলোচনা

১. ভূমিকা
২. যা যা প্রয়োজন হবে
৩. সিএসএস দ্বারা যেসকল কাজ করা যায়
৪. সিএসএস সাধারণ গঠন
৫. স্টাইলশীট ব্যবহারের পদ্ধতি
 - ৫.১. ইন্টারনাল সিএসএস
 - ৫.২. এক্সটার্নাল সিএসএস
 - ৫.৩. ইনলাইন সিএসএস
৬. মন্তব্য যোগ করা

অধ্যায়-দুইঃ সিএসএস টেক্সট প্রপাটি

১. সিএসএস-এ ব্যবহৃত সকল টেক্সট প্রপাটি
 - ১.১. টেক্সট কালার প্রপাটি
 - ১.২. টেক্সট ডিরেকশান প্রপাটি
 - ১.৩. লেটার স্পেসিং প্রপাটি
 - ১.৪. লাইন হাইট প্রপাটি
 - ১.৫. টেক্সট এলাইন প্রপাটি
 - ১.৬. টেক্সট ডেকোরেশান প্রপাটি
 - ১.৭. টেক্সট ইনডেন্ট প্রপাটি
 - ১.৮. টেক্সট শ্যাডো প্রপাটি
 ১. ৯. টেক্সট ট্রান্সফরমেশান প্রপাটি

১.১০.ভাটিক্যাল এলাইন প্রপাটি

১.১১. হোয়াইট স্পেস প্রপাটি।

১.১২.ওয়র্ড স্পেসিং প্রপাটি

অধ্যায়-তিনঃ সিএসএস পরিমাপক প্রপাটি

১.সিএসএস হাইট ও উইথ প্রপাটি

২.ম্যাক্স হাইট প্রপাটি

৩.ম্যাক্স উইথ প্রপাটি

৪.মিন হাইট প্রপাটি

৫.মিন উইথ প্রপাটি

অধ্যায়-চারঃ ফন্ট প্রপাটি

১.ফন্ট প্রপাটি

২.ফন্ট ফ্যামেলি প্রপাটি

৩.ফন্ট সাইজ প্রপাটি

৪.ফন্ট স্টাইল প্রপাটি

৫.ফন্ট ভ্যারিয়েন্ট প্রপাটি

৬.ফন্ট ওয়েট প্রপাটি

অধ্যায়-পাঁচঃ লিস্ট প্রপাটি

১.লিস্টের জন্য ব্যবহৃত সকল প্রপাটি

১.১.লিস্ট স্টাইল প্রপাটি

১.২.লিস্ট স্টাইল ইমেজ প্রপাটি

১.৩.লিস্ট স্টাইল পজিশান প্রপাটি

১.৪.লিস্ট স্টাইল টাইপ প্রপাটি

অধ্যায়-ছয়ঃ লিংক প্রপাটি

১.লিঙ্কের চারটি অবস্থা নিয়ে বিস্তারিত আলোচনা।

অধ্যায়-সাতঃ ব্যাকগ্রাউন্ড প্রপাটি

- ১.সিএসএস-এ ব্যবহৃত সকল ব্যাকগ্রাউন্ড প্রপাটি
 - ১.১.ব্যাকগ্রাউন্ড প্রপাটি
 - ১.২.ব্যাকগ্রাউন্ড এটাচমেন্ট প্রপাটি
 - ১.৩.ব্যাকগ্রাউন্ড কালার প্রপাটি
 - ১.৪.ব্যাকগ্রাউন্ড ইমেজ প্রপাটি
 - ১.৫.ব্যাকগ্রাউন্ড পজিশান প্রপাটি
 - ১.৬. ব্যাকগ্রাউন্ড রিপট প্রপাটি

অধ্যায়-আটঃ পজিশান প্রপাটি

- ১.স্ট্যাটিক পজিশনিং
- ২.ফ্লক্সড পজিশনিং
৩. রিলেটিভ পজিশনিং
৪. অ্যাবসুলেট পজিশনিং
- ৫.ক্লিপ প্রপাটি (Clip Property)
- ৬.কর্সর প্রপাটি (cursor Property)
- ৭.ওভারফ্লো প্রপাটি (Overflow Property)
- ৮.জেড ইনডেক্স (Z-index Property)

অধ্যায়-নয়ঃ বক্স প্রপাটি

- ১.বক্স মডেলের গঠন
- ২.সিএসএস-এ ব্যবহৃত সকল মার্জিন প্রপাটি।
 - ২.১.মার্জিন শর্টহ্যান্ড প্রপাটি।
৩. বর্ডার প্রপাটি
 - ৩.১.সিএসএস-এ ব্যবহৃত সকল বর্ডার প্রপাটি
৪. সিএসএস প্যাডিং
 - ৪.১.সকল সিএসএস প্যাডিং প্রপাটি
- ৫.সিএসএস আউটলাইন প্রপাটি
 - ৫.১.সিএসএস-এ ব্যবহৃত সকল আউটলাইন প্রপাটি

অধ্যায়-দশঃ টেবিলে সিএসএস-এর ব্যবহার

১. টেবিল বর্ডার
২. কলাপ্স বর্ডার

অধ্যায়- এগারঃ সিএসএস ফ্লোট ও ক্লিয়ার প্রপাটি

- ১.সিএসএস ফ্লোট প্রপাটি
- ২.সিএসএস ক্লিয়ার প্রপাটি

অধ্যায়-বারঃ সিএসএস সিলেক্টর

- ১.ইউনিভার্সাল সিলেক্টর
- ২.আইডি সিলেক্টর
- ৩.সিএসএস ক্লাস সিলেক্টর
- ৪.টাইপ সিলেক্টর
- ৫.ডিসেনডেন্ট সিলেক্টর
- ৬.direct children সিলেক্টর
- ৭.গ্রুপিং সিলেক্টর

অধ্যায়-তেরঃ সিএসএস সিউডো ক্লাস ও সিউডো এলিমেন্ট

- ১.সিএসএস সিউডো ক্লাস
- ২.সিউডো ক্লাসের প্রকারভেদ

- ২.১.এংকর সিউডো ক্লাস (Anchor Pseudo class)
- ২.২.ফাস্ট চাইল্ড সিউডো ক্লাস (First child Pseudo-class)
- ২.৩.ল্যাঙ্গুয়েজ সিউডো ক্লাস (Lang Pseudo-class)
- ২.৪.ফোকাস সিউডো ক্লাস (focus Pseudo-class)

- ৩.সিএসএস সিউডো এলিমেন্ট

- ৪.সিএসএস সিউডো এলিমেন্টের সাধারণ গঠন

- ৫.সিউডো এলিমেন্টের প্রকারভেদ

- ৫.১.প্রথম অক্ষর সিউডো ইলিমেন্ট-(First letter Pseudoelement)
- ৫.২.প্রথম লাইন সিউডো ইলিমেন্ট-(First first line Pseudoelement)

৫.৩.বিফোর সিউডো ইলিমেন্ট (Before Pseudo element)

৫.৪.আফটার সিউডো ইলিমেন্ট (After Pseudo element)

৬.সিএসএস-এ ব্যবহৃত সকল সিউডো ক্লাস/ এলিমেন্ট

অধ্যায়-চৌদ্দঃ সিএসএস ডিসপ্লে ও ভিজিবিলাটি প্রপাটি

১.সিএসএস ডিসপ্লে প্রপাটি

২.ডিসপ্লে নান -(display:none)

৩.ডিসপ্লে ইন লাইন -(display:inline)

৪.ডিসপ্লে ব্লক -(display:block)

৫.ভিজিবিলাটি প্রপাটি

অধ্যায়-এক

ভূমিকাঃ

ওয়েবপেজে স্টাইলশীটের মাধ্যমে অতিসহজেই ওয়েব পেজকে মনোগ্রাহী রূপে উপস্থাপন করা যায়। এইচ টি এম এল এর মাধ্যমে একটি ওয়েব পেইজ বাহ্যিক রূপ পায় আর সি এস এস এর মাধ্যমে এর সৌন্দর্য পরিপূর্ণতা লাভ করে। মূলত সি এস এস ব্যবহারে পেইজ ডিজাইনিং এর কাজে প্রচুর সময় বাঁচে। ওয়েব ডিজাইনের দক্ষতা উপরের ধাপে উন্নিত করতে চাইলে **Cascading Style Sheets (CSS)** এর কোন বিকল্প নেই। সিএসএস একই ডকুমেন্টে একাধিকবার ব্যবহার হতে পারে এবং ডকুমেন্টের বিভিন্ন অংশে বিভিন্ন স্টাইল ব্যবহার করা যেতে পারে। স্টাইলশীটের ব্যবহারে ডকুমেন্টের স্টাইল বদলে যাবে কিন্তু স্ট্রাকচার অপরিবর্তিত থাকবে। একই স্টাইলশীট একাধিক পেজে কিংবা একাধিক স্টাইলশীট একই পেজে ব্যবহার করা যেতে পারে। আপনার স্টাইলশীট (যেখানে আপনি কোডগুলো লিখবেন) টি সম্পূর্ণভাবে এইচটিএমএল ডকুমেন্ট হতে পৃথক হবে যখন আপনি সিএসএস এবং এইচটিএমএল এ দক্ষতা অর্জন করতে পারবেন। আর একটি কথা না বললেই নয় সিএসএস শেখার আগে অবশ্যই এইচটিএমএল সম্বন্ধে ভাল জানতে হবে। এই বইটিতে আমরা ধারাবাহিকভাবে সিএসএস নিয়ে আলোচনা করব।

যা যা প্রয়োজন হবেঃ

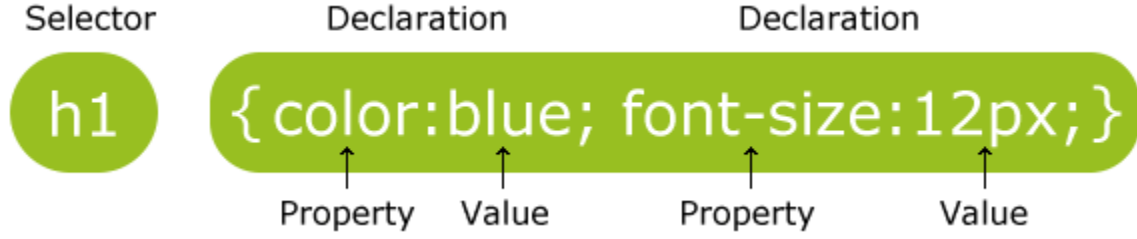
সিএসএস লেখার জন্য আলাদা কোন টেক্সট এডিটর সফটওয়্যার ব্যবহার না করলেও হবে, আপনার কম্পিউটারে **Notepad** নামে যে টেক্সট এডিটর আছে সেখানেই কোড লিখতে পারবেন। তবে নোটপ্যাডের উন্নত সংস্করণ **Notepad++** বা আরও উন্নত এডিটর যেখানে আপনি একই সাথে কোড লিখতে ও তার আউটপুট দেখতে পারবেন যেমন- **Adobe Dreamweaver, HTML Kit** ব্যবহার করতে পারেন। আপাতত আমি আপনাদের **Notepad++** ব্যবহার করতে বলব। এবার সিএসএস কোডকে রান করানোর জন্য যেকোন একটি ব্রাউজার হলেই হবে। যেমনঃ **Internet Explorer, Mozilla Firefox, Google Chrome** ইত্যাদি। তৈরিকৃত সিএসএস ফাইলটি **.css Extension** দিয়ে সেভ করুন।

সিএসএস দ্বারা যেসকল কাজ করা যায়ঃ

- ১ সিএসএস এর মাধ্যমে এইচটিএমএল এ তৈরিকৃত পেজটি আরও দৃষ্টিনন্দন করা যায়।
- ২ কিছু সিএসএস কোড পরিবর্তন করে সম্পূর্ণভাবে সাইটের **restyle** করা যায়।

সিএসএস সাধারণ গঠনঃ

সিএসএস এর গঠনকে দুটি অংশে ভাগ করা যায়। একটি হল সিলেক্টর এবং অন্যটি হল এক বা একাধিক ডিকলারেশন। নিচে সাধারণ গঠনের চিত্র দেওয়া হল-



সিলেক্টর হল সাধারণ HTML এলিমেন্ট (h1→h6, p etc)

ডিকলারেশনের আবার দুইটি অংশ থাকবে-

Declaration={Properties: Value}

একাধিক Declaration থাকতে পারে যা সেমিকোলন দিয়ে পৃথক থাকবে।

আরও সহজ ভাবে লেখা যায়-

"HTML tag" { "CSS Property" : "Value" ; }

স্টাইলশীট ব্যবহারের পদ্ধতিঃ

এইচটিএমএল ডকুমেন্টে স্টাইলশীট ব্যবহারের তিনটি পদ্ধতি রয়েছে -**internal**, **external** এবং **inline**। একটা এইচটিএমএল পেজে <head> ট্যাগের ভিতর <style> ট্যাগ দিয়ে সিএসএস কোড যোগ করে পেজ স্টাইলিং করা যায়-এটা হচ্ছে ইন্টারনাল সিএসএস। আর যদি সিএসএস কোড বেশি হয়ে যায় তখন সিএসএস কোড আলাদা ফাইলে লেখা হয় এবং <head> ট্যাগের ভিতর <link> ট্যাগ দিয়ে সংযুক্ত করা হয়-এই পদ্ধতি হচ্ছে এক্সটার্নাল সিএসএস। আমরা ওয়েব পেজ ডিজাইন করতে এক্সটার্নাল সিএসএস বেশি ব্যবহার করব। আর এইচটিএমএল-এর প্রতিটা এলিমেন্টের সাথে এককভাবে স্টাইল নির্ধারণ করা হল-ইনলাইন সিএসএস। যে কোন এইচটিএমএল ডকুমেন্টে আপনি স্টাইলশীট ব্যবহার করতে পারেন। এখন আমরা ধরাবাহিকভাবে এই তিনটি পদ্ধতি নিয়ে আলোচনা করবঃ

১.ইন্টারনাল সিএসএসঃ

ইন্টারনাল সিএসএস সাধারণত তখনই ব্যবহার করা হয় যখন একটি এইচটিএমএল ডকুমেন্টের জন্য একটি ডিজাইন দরকার। ইন্টারনাল সিএসএস-এ <head> ট্যাগের ভিতর <style> ট্যাগ দিয়ে সিএসএস কোড যোগ করা হয়। যেমন-

```
<html>
<head>
<style type="text/css">
h1{color:#ff00ff;}
```

```
p{word-spacing:10px;}
</style>
<body>
<h1>My first CSS Codeing</h1>
<p>
```

CSS stands for Cascading Style Sheets. Styles define how to display HTML elements. Styles were added to HTML 4.0 to solve a problem. External Style Sheets can save a lot of work. External Style Sheets are stored in CSS files

```
</p>
</body>
</html>
```

উপরের কোডগুলো [index.html](#) নামে সেভ করে রান করে আউটপুট দেখুন।

২. এক্সটার্নাল সিএসএসঃ

সবচেয়ে ভাল হয় যদি আপনারা সিএসএস কে এইচটিএমএল হতে পৃথক রাখেন, আর এটাই হল এক্সটার্নাল সিএসএস। যখন একটি সিএসএস কোড অনেক ডকুমেন্টের জন্য ব্যবহার করা হয় তখন এক্সটার্নাল সিএসএস উত্তম। এক্সটার্নাল সিএসএস -এ সিএসএস ফাইলকে **<head>** ট্যাগের ভিতর **<link>** ট্যাগ দিয়ে সংযুক্ত করা হয়। যেমন-

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css" />
</head>
```

এক্সটার্নাল সিএসএস ফাইলটি শুধুমাত্র সিএসএস code ধারণ করে এবং এই ফাইলটিকে ".css" ফাইল extension দিয়ে সেভ করা হয়। একটি উদাহরণ দেখুন-

```
H3 {color:#FF0000;}
p {
background:#00F;
text-align:center;
}
```

এবার উপরের কোডগুলো style.css নাম দিয়ে সেভ করুন। এরপর নিচের মত করে এইচটিএমএল ফাইলের সাথে সিএসএস ফাইলটিকে লিংক করে দিন।

```
<html>

<head>

<link rel="stylesheet" type="text/css" href="style.css">

</head>

<body>

<h3>A white Header</h3>

<p>This paragraph has a blue font.

The background color of this page is gray because we changed it with CSS!</p>

</body>

</html>
```

তারপর এইচটিএমএল ফাইল টিকে index.html হিসাবে একই ফোল্ডারে অর্থাৎ যেখানে সিএসএস ফাইল আছে সেখানে সেভ করে index.html ফাইলটি রান করে আউটপুট দেখুন।

৩. ইনলাইন সিএসএসঃ

ডকুমেন্টের প্রতিটি এইচটিএমএল ট্যাগের সাথে **Style** এট্রিবিউট ব্যবহার করে ডকুমেন্টে স্টাইল নির্ধারণ করাই হল ইনলাইন সিএসএস। ইনলাইন সিএসএস ক্ষেত্রে **Style** নির্দেশ দেয়ার সময় **Type Attribute** ডিকলারেশন প্রয়োজন হয় না এবং দ্বিতীয় বন্ধনী **{}** এর বদলে উদ্ধৃতি চিহ্ন ব্যবহার করা হয়। নিচের উদাহরণে দেখুন-

```
<p style="color:sienna;margin-left:20px">This is a paragraph.</p>
```

এক্সটার্নাল বা ইন্টারনাল সিএসএস হতে ইনলাইন সিএসএস এর প্রাধান্য বেশি। এর অর্থ হচ্ছে এক্সটার্নাল বা ইন্টারনাল সিএসএস দ্বারা স্টাইল নির্দেশিত থাকলেও আপনি ইনলাইন সিএসএস দ্বারা ঐ স্টাইল **override** করতে পারেন। যাহোক, ইনলাইন সিএসএস যা সিএসএস এর সঠিক উদ্দেশ্য হতে বিচ্যুতি ঘটায় তাই এটা আংশিকভাবে ব্যবহার করা উচিত।

নোটঃ যদি এক্সট্রানাল সিএসএস ফাইলের লিংক ইন্টারনাল সিএসএস-এর পরে স্থাপন করা হয় তবে এক্সট্রানাল সিএসএস, ইন্টারনাল সিএসএস-কে **override** করবে।

মন্তব্য যোগ করাঃ

মন্তব্য যোগ করা যে কোন ল্যাংগুয়েজের একটি সাধারণ বিষয়। সিএসএস-এ মন্তব্য **"/*"** দিয়ে শুরু হয় এবং **"/"** দিয়ে শেষ হয়। যেমন-

```
1  /*This is a comment*/
2  p
3  {
4  text-align:center;
5  /*This is another comment*/
6  color:black;
7  font-family:arial;
8  }
```

অধ্যায়-দুই

[সিএসএস টেক্সট প্রপার্টি]

সিএসএস-এর বেশ কিছু প্রপার্টি সরাসরি টেক্সটকে নিয়ন্ত্রণ করে। একটি ওয়েব পেজে হেডিং হিসেবে বা প্যারাগ্রাফ হিসেবে বা অন্য কিছু হিসেবে অনেক অনেক টেক্সট থেকে থাকে। আর ওয়েব পেজকে সাজাতে এই সকল টেক্সটের স্টাইল নির্ধারণ করা আবশ্যিক। সিএসএস টেক্সট প্রপার্টির মাধ্যমে আপনি টেক্সট এর **spacing, decoration, color, alignment** সহ আরও অনেক কিছুই নিয়ন্ত্রণ করতে পারেন। আমরা এই অধ্যায়ে সিএসএস টেক্সট প্রপার্টি নিয়ে বিস্তারিত আলোচনা করবো।

সিএসএস-এ ব্যবহৃত সকল টেক্সট প্রপার্টি

প্রপার্টি	বর্ণনা
1.color	টেক্সটের রঙ নির্ধারণ করে।
2.direction	টেক্সটের দিক নির্দেশ করে।
3.letter-spacing	এর মাধ্যমে একটি টেক্সটের characters গুলোর মধ্যে space এর পরিমাণ কমানো বা বাড়ানো যায়।
4.line-height	লাইনের উচ্চতা নির্দেশ করে।
5.text-align	টেক্সটের আনুভূমিক এলাইনমেন্ট নির্দেশ করে।
6.text-decoration	এই প্রপার্টির মাধ্যমে টেক্সটকে বিভিন্নভাবে প্রদর্শন করা যেতে পারে।
7.text-indent	text indent মানে লেখাটি বাম দিকে থেকে কতটুকু ছাড় দিয়ে শুরু হবে।
8.text-shadow	টেক্সটে শ্যাডো ইফেক্ট যোগ করে।
9.text-transform	টেক্সটের ক্যাপিটলাইজেশন নিয়ন্ত্রণ করে।

10.vertical-align	টেক্সটের উল্লম্ব এলাইনমেন্ট নির্দেশ করে।
11.white-space	Specifies how white-space inside an element is handled
12.word-spacing	এর মাধ্যমে পাশাপাশি শব্দগুলোর মধ্যে white space এর পরিমাণ কমানো বা বাড়ানো যায়।

১. টেক্সট কালার প্রপার্টি:

ওয়েব পেজের যে কোন টেক্সটের কালার নির্ধারণ করতে কালার প্রপার্টি ব্যবহার করা হয়। যেমন-

```
<html>
<head>
<style type="text/css">
h1 {color:#00ff00;}
p {color:rgb(0,0,255);}
</style>
</head>
<body>
<h1>This is heading 1</h1>
<p>This is an ordinary paragraph.</p>
</body>
</html>
```

আউটপুট:

This is heading 1

This is an ordinary paragraph.

বিঃ দ্রঃ কালার ভেলু আপনার এইচটিএমএল করার সময় যে ভাবে সেট করেছেন এখানে সেই একই ভাবে সেট করতে পারবেন।

২. টেক্সট ডিরেকশান প্রপার্টি:

টেক্সটের দিক নির্দেশ করতে এই প্রপার্টি ব্যবহার করা হয়। এর ব্যবহার অনেক কম। এটা না জানলেও হবে।

```
<html>
<head>
<style type="text/css">
p {direction:rtl;}
h5{direction:ltr}
```



```
</style>
</head>
<body>
<p>Some text. Default writing direction.</p>
<h5>Some text. Default writing direction.</h5>
</body>
</html>
```

এই প্রপাটি দুটি মান ব্যবহার করে-

<i>ltr</i>	টেক্সটের দিক বাম থেকে ডানে হবে। এটা ডিফল্ট মান।
<i>rtl</i>	টেক্সটের দিক ডান থেকে বামে হবে।

৩. লেটার স্পেসিং প্রপাটিঃ

সিএসএস **letter-spacing** প্রপাটির মাধ্যমে একটি টেক্সটের ক্যারেক্টারগুলোর মধ্যে স্পেস এর পরিমাণ কমানো বা বাড়ানো যায়। যেমন-

```
<html>
<head>
<style type="text/css">
h1 {letter-spacing:2px;}
h2 {letter-spacing:-3px;}
</style>
</head>
```

```
<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
</body>
</html>
```

আউটপুটঃ

This is heading 1

This is heading 2

লক্ষ্য করুন নেগেটিভ মান ব্যবহারের কারণে লেখাগুলো গায়ে গায়ে লেগে গিয়েছে।

৪.লাইন হাইট প্রপাটিঃ

দুটো লাইনের মাঝে কি পরিমাণ ফাঁকা স্থান থাকবে তা নির্দেশ করা হয় এই প্রপাটির মাধ্যমে।

[বিঃ দ্রঃ নিচের উদাহরণটিতে সিএসএস ক্লাস সিলেক্টর ব্যবহার করা হয়েছে। সিলেক্টর সম্বন্ধে জানতে আগে সিলেক্টর অধ্যায়টি পড়ে আসুন।]

```
<html>
<head>
<style type="text/css">
p.small {line-height:90%;}
p.big {line-height:200%;}
</style>
</head>
```

```
<body>
```

```
<p>
```

This is a paragraph with a standard line-height.

The default line height in most browsers is about 110% to 120%.

This is a paragraph with a standard line-height.

```
</p>
```

```
<p class="small">
```

This is a paragraph with a smaller line-height.

This is a paragraph with a smaller line-height.

This is a paragraph with a smaller line-height.

```
</p>
```

```
<p class="big">
```

This is a paragraph with a bigger line-height.

This is a paragraph with a bigger line-height.

This is a paragraph with a bigger line-height.

```
</p>
```

```
</body>
```

```
</html>
```

কোড বিশ্লেষণ: উপরের উদাহরণটিতে সিএসএস লেখার সময় আমরা `p.small` ও `p.big` ব্যবহার করেছি যা অনেকেই হয়তো বুঝতে পারেন নি। বডি অংশে লক্ষ্য করুন এখানে তিনটি প্যারাগ্রাফ আছে আমরা চাই তিনটি প্যারাগ্রাফে তিনটি

আলাদা স্টাইল হোক তাই প্রথমে তাদেরকে `<p class="small">` ও `<p class="big">` নামে চিহ্নিত করে রেখেছি পরে স্টাইল শীটে ক্লাস নাম ধরে কল করেছি, এটাকে বলে ক্লাস সিলেক্টর।

line-height প্রপাটিতে ব্যবহৃত মানের এককসমূহ-

১. সরাসরি নাম্বার ব্যবহার করা যায়। (এক্ষেত্রে লাইনের উচ্চতা হবে ফন্ট সাইজের মানের সাথে গুন করে যা হবে)

২. লাইনের উচ্চতা *px*, *pt*, *cm* এককের মাধ্যমে নির্ণয় করা।

৩. শতকরা (%) মান ব্যবহার করা হলে লাইনের উচ্চতা হবে ব্যবহৃত ফন্ট সাইজের সাপেক্ষে।

৫. টেক্সট এলাইন প্রপাটিঃ

ওয়েব সাইটে সাধারণত টেক্সটগুলোর লেফট এলাইনমেন্ট নিবাচিত থাকে। `text-align` প্রপাটির সাহায্যে টেক্সট এর বিভিন্ন এলাইনমেন্ট দেয়া যায়। এর মান হতে পারে-`left`, `right`, `center`, `justify`। একটি উদাহরণ দেখুন-

```
<html>
<head>
<style type="text/css">
h1 {text-align:center}
h2 {text-align:left}
h3 {text-align:right}
</style>
</head>
<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
```

```
</body>
```

```
</html>
```

৬. টেক্সট ডেকোরেশান প্রপাটিঃ

এই প্রপাটির মাধ্যমে টেক্সটকে বিভিন্নভাবে প্রদর্শন করা যেতে পারে। **text-decoration** প্রপাটি সবচেয়ে বেশি ব্যবহার করা হয় লিঙ্কের আন্ডারলাইন দূর করতে। মেনু তৈরি করার সময় এটা অনেক কাজে লাগবে।

একটি উদাহরণ দেখুন-

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
h1 {text-decoration:overline;}
```

```
h2 {text-decoration:line-through;}
```

```
h3 {text-decoration:underline;}
```

```
h4 {text-decoration:blink;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>This is heading 1</h1>
```

```
<h2>This is heading 2</h2>
```

```
<h3>This is heading 3</h3>
```

```
<h4>This is heading 4</h4>
```

</body>

</html>

আউটপুটঃ

This is heading 1

This is heading 2

This is heading 3

This is heading 4

Text-decoration যে মানগুলো ব্যবহার করতে পারে-

ভেলু	বর্ণনা
<i>none</i>	নরমাল টেক্সট নির্দেশ করে। এটা ডিফল্ট।
<i>underline</i>	টেক্সটে আন্ডারলাইন হবে
<i>overline</i>	টেক্সটে ওভারলাইন হবে
<i>line-through</i>	টেক্সটের মাঝ বরাবর দাগ হবে।
<i>blink</i>	টেক্সট জ্বলা নেভা করবে।

৭. টেক্সট ইনডেন্ট প্রপার্টিঃ

সিএসএস **text-indent** কে আমরা এইচটিএমএল (<pre>) ট্যাগ পরিবর্তে ব্যবহার করতে পারি। সরাসরি কোন মান বা শতকরা মান ব্যবহার করে আমরা **text-indent** নির্দিষ্ট করতে পারি। **text indent** মানে লেখাটি বাম দিকে থেকে কতটুকু ছাড় দিয়ে শুরু হবে। এর মান হিসেবে দৈর্ঘ্য(em,ex,pt) ও শতকরা হার(%) ব্যবহৃত হতে পারে। নেগেটিভ মান দেওয়া যেতে পারে।

একটি উদাহরণ দেখুন-

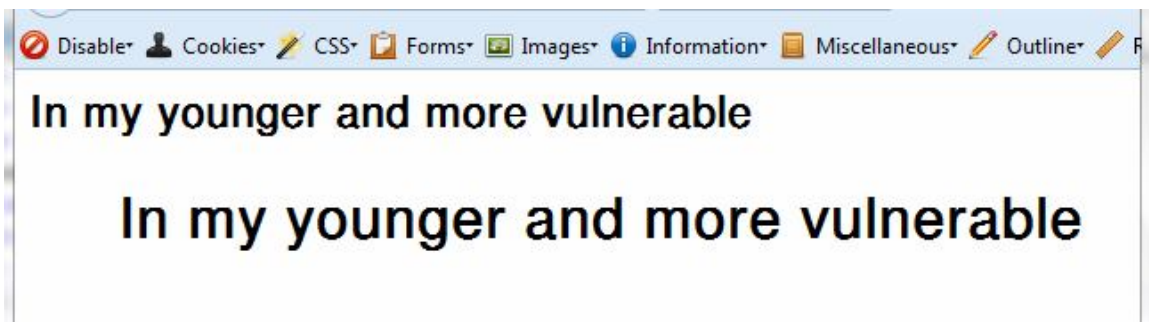
<html>

<head>

<style type="text/css">

```
h1 {text-indent:50px;}
</style>
</head>
<body>
<h2>In my younger and more vulnerable </h2>
<h1>In my younger and more vulnerable </h1>
</body>
</html>
```

আউটপুটঃ



কোড বিশ্লেষণঃ উপরের উদাহরণটিতে লক্ষ্য করুন <h1> এর text-indent:50px দেওয়ার কারণে লেখাটি বামদিক থেকে 50px দূরে সরে গিয়েছে।

৮. টেক্সট শ্যাডো প্রপার্টিঃ

কোন টেক্সটে শ্যাডো এফেক্ট প্রয়োগ করা হবে কি না তা নির্দেশ করা হয় এই প্রপার্টি দিয়ে।

একটি উদাহরণ দেখুন-

```
<html>
<head>
<style type="text/css">
h1 {
background: white;
color: red;
text-shadow: 5px 3px 5px red ;
```

```
}  
</style>  
</head>  
<body>  
<h1>Red Shadow Effect </h1>  
</body>  
</html>
```

আউটপুটঃ

Red Shadow Effect

ভেলুগুলো নির্দেশ করে কোনদিকে কত পিক্সেল শ্যাডো হবে আর রঙের নাম নির্দেশ করে কি রঙের শ্যাডো হবে।

৯. টেক্সট ট্রান্সফরমেশান প্রপার্টিঃ

টেক্সট Transform এর মাধ্যমে টেক্সট এর দ্রুত **capitalize**(ছোট থেকে বড় বা বড় থেকে ছোট হরফে পরিবর্তন) করা যায়। এর মাধ্যমে প্রতিটা শব্দের প্রথম অক্ষরকেও **capitalize** করা যেতে পারে। এর চারটি মান হতে পারে।যেমন-

ভেলু	বর্ণনা
none	Capitalization হবে না। এটা ডিফল্ট ।
capitalize	প্রতিটা ওয়ার্ড এর প্রথম অক্ষর uppercase -এ পরিবর্তন করবে।
uppercase	সকল ক্যারেক্টারকে uppercase -এ পরিবর্তন করবে।
lowercase	সকল ক্যারেক্টারকে lowercase -এ পরিবর্তন করবে।

একটি উদাহরণ দেখুন-

```
<html>  
  
<head>  
  
<style type="text/css">
```



```

p.uppercase {text-transform:uppercase;}
p.lowercase {text-transform:lowercase;}
p.capitalize {text-transform:capitalize;}
</style>
</head>
<body>
<p class="uppercase">This is some text.</p>
<p class="lowercase">This is some text.</p>
<p class="capitalize">This is some text.</p>
</body>
</html>

```

আউটপুটঃ

THIS IS SOME TEXT.

this is some text.

This Is Some Text.

১০. ভাটিক্যাল এলাইন প্রপাটিঃ

টেক্সটের উল্লম্ব এলাইনমেন্ট নির্দেশ করা হয় এই প্রপাটির মাধ্যমে।

Vertical-align property যে মানগুলো ব্যবহার করতে পারে-

ভেলু	বর্ণনা
<i>length</i>	নির্দিষ্ট ভেলুর সাহায্যে কোন এলিমেন্টের সাইজ বাড়ায় বা কমায়। নেগেটিভ মান গ্রহণযোগ্য।
<i>%</i>	<i>Raises or lower an element in a percent of the "line-height" property.</i> নেগেটিভ মান গ্রহণযোগ্য।

<i>baseline</i>	মূল উপাদানের বেজলাইনের সাথে বর্তমান উপাদানকে এলাইন করে। <i>This is default</i>
<i>sub</i>	বর্তমান উপাদানকে <i>subscript</i> হিসেবে দেখায়।
<i>super</i>	বর্তমান উপাদানকে <i>superscript</i> হিসেবে দেখায়।
<i>top</i>	বর্তমান উপাদানের <i>top</i> কে বর্তমান সারির সবচেয়ে উঁচু উপাদানের <i>top</i> এর সাথে এলাইন করে।
<i>text-top</i>	বর্তমান উপাদানকে মূল উপাদানের <i>top of the font</i> এর সাথে এলাইন করে।
<i>middle</i>	বর্তমান উপাদানকে মূল উপাদানের মাঝে এলাইন করে।
<i>bottom</i>	বর্তমান উপাদানের তলকে ওই লাইনের সবচেয়ে নিচের উপাদানটির তলের সাথে এলাইন করে।
<i>text-bottom</i>	বর্তমান উপাদানের তল(<i>bottom</i>) মূল উপাদানের তলের সাথে এলাইন করে।

একটি উদাহরণ দেখুন-

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
img.top {vertical-align:text-top;}
```

```
img.bottom {vertical-align:text-bottom;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>An  image  
with a default alignment.</p>
```

```
<p>An  image with a text-top alignment.</p>
```

```
<p>An  image with a text-bottom alignment.</p>
```

</body>

</html>

১১. হোয়াইট স্পেস প্রপার্টি:

হোয়াইট স্পেস প্রপার্টির মাধ্যমে একটি লেখা এক লাইনে চলতে থাকবে যতক্ষণ না আপনি **break
** ট্যাগ না দিবেন। **</br>** ট্যাগ দিলে তারপর থেকে নিচে লাইন যাবে।

white-space property যে মানগুলো ব্যবহার করতে পারে-

ভেলু	বর্ণনা
normal	পাশাপাশি অনেকগুলো <i>whitespace</i> একটি <i>whitespace</i> -এ পরিণত হবে। টেক্সট মুড়িয়ে(<i>wrap</i>) যাবে যখন প্রয়োজন। এটা ডিফল্ট।
nowrap	পাশাপাশি অনেকগুলো <i>whitespace</i> একটি <i>whitespace</i> -এ পরিণত হবে। টেক্সট মুড়িয়ে পরিবর্তী লাইনে যাবে না। কিন্তু এক লাইনে চলতে চলতে যদি
 ট্যাগ পায় তবে মুড়িয়ে যাবে।
Pre	কোডিং-এ যতগুলো <i>Whitespace</i> থাকবে তার সবই ব্রাউজারে দেখা যাবে। <i>Text will only wrap on line breaks Acts like the <pre> tag in HTML</i>
pre-line	পাশাপাশি অনেকগুলো <i>whitespace</i> একটি <i>whitespace</i> -এ পরিণত হবে। টেক্সট মুড়িয়ে(<i>wrap</i>) যাবে যখন প্রয়োজন হবে কিন্তু লাইন ব্রেক হবে না।
pre-wrap	কোডিং-এ যতগুলো <i>Whitespace</i> থাকবে তার সবই ব্রাউজারে দেখা যাবে। টেক্সট মুড়িয়ে(<i>wrap</i>) যাবে যখন প্রয়োজন হবে কিন্তু লাইন ব্রেক হবে না।

একটি উদাহরণ দেখুন-

<html>

<head>

<style type="text/css">

p

{

white-space:nowrap;

}

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>
```

This is some text. This is some text. This is some text.

This is some text. This is some text. This is some text.

This is some text. This is some text. This is some text.

This is some text. This is some text. This is some text.

```
</p>
```

```
</body>
```

```
</html>
```

১২. ওয়ার্ড স্পেসিং প্রপার্টিঃ

Word Spacing property-এর মাধ্যমে পাশাপাশি শব্দগুলোর মধ্যে white space এর পরিমাণ কমানো বা বাড়ানো যায়। এর মান px, pt, cm, em এককে হতে পারে। নেগেটিভ মান ব্যবহার করা যেতে পারে।

একটি উদাহরণ দেখুন-

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
p
```

```
{
```

```
word-spacing:30px;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>
This is some text. This is some text.
</p>

</body>
</html>
```

আউটপুটঃ

This is some text. This is some text.

অধ্যায়-তিন

[সিএসএস পরিমাপক প্রপাটি]

এইচটিএমএল- এ আমরা **height** ও **width** এট্রিবিউটের মাধ্যমে বিভিন্ন এলিমেন্টের আকার নির্ধারণ করে দিয়েছি। কিন্তু আমরা যখন সিএসএস শিখব তখন এইচটিএমএল দিয়ে এই কাজগুলো আর করব না। সিএসএস-এ মোট ছয়টি প্রপাটির সাহায্যে কোন এলিমেন্টের আকার নির্ধারণ করা হয়। এই ছয়টি প্রপাটির মাঝে মূলত ১ ও ৬ নাম্বারটাই বেশি ব্যবহার করা হয়।

সিএসএস-এর সকল আকার নির্ধারক প্রপাটি

প্রপাটি	বর্ণনা	মান
1.height	এলিমেন্টের height নির্দেশ করে	<i>auto</i> <i>length</i> <i>%</i>
2.max-height	এলিমেন্টের সর্বচ্চ height নির্দেশ করে	<i>none</i> <i>length</i> <i>%</i>
3.max-width	এলিমেন্টের সর্বচ্চ width নির্দেশ করে	<i>none</i> <i>length</i> <i>%</i>
4.min-height	এলিমেন্টের সর্বনিম্ন height নির্দেশ করে	<i>length</i> <i>%</i>
5.min-width	এলিমেন্টের সর্বনিম্ন width নির্দেশ করে	<i>length</i> <i>%</i>

<i>6.width</i>	এলিমেন্টের <i>width</i> নির্দেশ করে	<i>auto</i> <i>length</i> %
----------------	-------------------------------------	-----------------------------------

উপরের এই ছয়টি প্রপার্টিতে ব্যবহৃত ভেলু গুলোর মান ও বর্ণনা নিম্নে দেওয়া হল-

None→ No maximum height, maximum width. এটা ডিফল্ট মান।

Auto→ ব্রাউজার নিজেই হাইট ক্যালকুলেট করে। এটা ডিফল্ট মান।

Length→ px, cm একক ব্যবহার করে।

%→ পুরো উইন্ডো কনটেন্টের জন্য সিলেক্ট হবে।

ধারাবাহিকভাবে নিম্নে এদের আলোচনা করা হলঃ

1+6.height and width Property:

কোন এলিমেন্টের **height** ও **width** নির্ণয় করা হয় এই প্রপার্টি দুয়ের দ্বারা। যেমন-

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
div{ height:250px; width:200px; background:#887733}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div><h3>This is a Block</h3></div>
```

```
</body>
```

```
</html>
```

আউটপুট:


```
</body>
```

```
</html>
```

3.CSS max-width Property:

এলিমেন্টের max-width নির্দেশ করে। যেমন-

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
p{max-width:100px; background-color:yellow;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>The maximum width of this paragraph is set to 100px.</p>
```

```
</body>
```

```
</html>
```

4.CSS min-height Property:

এলিমেন্টের min-height নির্দেশ করে। যেমন-

```
<html>
```

```
<head>
```

```
<style type="text/css">
```



```
p
{
min-height:100px;
background-color:yellow;
}
</style>
</head>

<body>
<p>The minimum height of this paragraph is set to 100px.</p>
</body>
</html>
```

5.CSS min-width Property:

এলিমেন্টের min-width নির্দেশ করে। যেমন-

```
<html>
<head>
<style type="text/css">
p{min-width:150px; background-color:yellow;}
```

```
</style>
</head>
<body>
<p>The minimum width of this paragraph is set to 150px.</p>
</body>
</html>
```

অধ্যায়-চার

[ফন্ট প্রপার্টি]

আমার লেখা এইচটিএমএল বইটিতে আমি বলেছিলাম ফন্টের পরিপূর্ণ নিয়ন্ত্রণ পাওয়া যাবে সিএসএস ব্যবহারের সময়। ফন্ট ওয়েব পেজের জন্য অত্যন্ত গুরুত্বপূর্ণ একটি বিষয়। ওয়েব পেজের অধিকাংশ জায়গা জুড়ে ফন্ট বিদ্যমান তাই একটি সুন্দর ও আকর্ষণীয় ওয়েব পেজ তৈরির জন্য ফন্টের গুরুত্ব অপরিসীম। এ অধ্যায়ে আমরা ফন্ট সম্পর্কিত যাবতীয় বিষয় আলোচনা করবো। সিএসএস- এ ফন্ট সম্পর্কিত ছয়টি প্রপার্টি সেট করা যেতে পারে। এগুলো হলঃ

প্রপার্টি	বর্ণনা
1. <i>font</i>	সকল <i>font properties</i> -এর জন্য একটি ডিকলারেশন।
2. <i>font-family</i>	টেক্সটের জন্য <i>font family</i> নির্ধারণ করে।
3. <i>font-size</i>	টেক্সটের জন্য <i>font size</i> নির্ধারণ করে।
4. <i>font-style</i>	টেক্সটের জন্য <i>font style</i> নির্ধারণ করে।
5. <i>font-variant</i>	ফন্ট <i>Variant</i> এর সাহায্যে আপনি ফন্ট কে <i>small caps</i> এ পরিবর্তিত করতে পারেন।
6. <i>font-weight</i>	একটি ফন্ট কী পরিমাণ মোটা দেখা যাবে তা বোঝানোর জন্য ব্যবহার করা হয় <i>font-weight</i> প্রপার্টি।

প্রতিটি প্রপার্টি সম্পর্কে নিচে আলোচনা করা হলঃ

১.ফন্ট প্রপার্টিঃ

ফন্টের বিভিন্ন প্রপার্টির মান ভিন্ন ভিন্ন ডিকলারেশনে না দিয়ে একই ডিকলারেশনের মাধ্যমে লেখার জন্য *font-property* ব্যবহার করা হয়। যেমনঃ একটি সাধারণ ডিকলারেশন-

```
p{
```

```
font-size:larger;  
  
font-style:italic;  
  
font:varient:small-caps;  
  
font-family:Arial;  
  
}
```

font-property ব্যবহার করে সহজভাবে নিচের মত লেখা যায়-

```
p{font: larger italic small-caps Arial }
```

প্রতিটি মানকে স্পেস দিয়ে আলাদা করতে হবে।

একটি উদাহরণ দেখুনঃ

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
p{font:15px arial,sans-serif;}
```

```
</style>
```

```
</head>
```

```
<body>
```

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

```
</body>
```

```
</html>
```

২. ফন্ট ফ্যামেলি প্রপাটিঃ

কোন ফন্ট ফ্যামেলি বা ফন্ট গ্রুপ ব্যবহার করতে চান তা নির্দেশ করতে পারেন এ প্রপাটির মাধ্যমে। [ফন্ট ফ্যামেলি বা ফন্ট গ্রুপ এবং জেনেরিক ফন্ট ফ্যামেলি সম্পর্কে বিস্তারিত জানতে আমার লেখা এইচটিএমএল বইটি পড়ুন]

একটি উদাহরণ দেখুনঃ

```
<html>
<head>
<style type="text/css">
p{font-family:"Times New Roman",Times,serif;}
</style>
</head>
<body>
<h1>CSS font-family</h1>
<p >This is a paragraph, shown in the Times New Roman font.</p>
</body>
</html>
```

ফন্টের নামের মাঝে স্পেস থাকলে তাকে অবশ্যই উদ্ধৃতি চিহ্নের মাঝে রাখতে হবে। অন্য ক্ষেত্রে এই নিয়ম না মানলেও চলবে। তবে উদ্ধৃতি চিহ্ন ব্যবহার করা উত্তম। **Single** অনেকগুলো ফন্ট একসাথে ব্যবহারের নিয়ম-

```
h1{font-family:"Arial Black",Arial }
```

এক্ষেত্রে আপনি জেনেরিক ফন্ট ফ্যামেলির নাম ও ব্যবহার করতে পারেন।

সিএসএস-এ দুই ধরনের ফন্ট ফ্যামেলি নাম রয়েছে-

1. family-name - ফন্টের নাম। যেমন-Arial, times new roman ইত্যাদি।

2. generic-family - জেনেরিক ফন্ট ফ্যামেলির নাম। যেমন-serif, sans-serif ইত্যাদি। গবেষণায় দেখা গিয়েছে serif fonts হতে sans-serif fonts যা computer monitor এ খুব সহজে পড়া যায়।

৩. ফন্ট সাইজ প্রপার্টিঃ

ফন্টের আকার নির্ধারণ করা হয় **font-size** প্রপার্টির মাধ্যমে। এই প্রপার্টির জন্য চার ধরনের মান ব্যবহার করা যেতে পারে- অ্যাবসুলেট, আপেক্ষিক, শতকরা ও দৈর্ঘ্য।

অ্যাবসুলেট সাইজঃ

কোন ব্রাউজারে টেক্সট সাইজ পরিবর্তনের সুযোগ দেয় না। এটা **accessibility** এর জন্য একটি বড় সমস্যা। যদি আউটপুট সাইজ আগেই জানা থাকে তবে অ্যাবসুলেট সাইজ ব্যবহার করা যেতে পারে।

আপেক্ষিক সাইজঃ টেক্সট সাইজ পরিবর্তনের সুযোগ দেয়।

নোটঃ আপনি যদি **font size specify** করে না দেন তবে ডিফল্ট ফন্টের যে সাইজ তাই দেখা যাবে। আর এই ডিফল্ট সাইজ হল- **16px (16px=1em)**।

font size property-যে সকল মান ব্যবহার করতে পারে-

ভেলু	বর্ণনা
	অ্যাবসুলেট মান
xx-small	ফন্ট সাইজ হবে xx-small সাইজ
x-small	ফন্ট সাইজ হবে extra small সাইজ
small	ফন্ট সাইজ হবে small সাইজ
medium	ফন্ট সাইজ হবে medium size . এটা ডিফল্ট।
large	ফন্ট সাইজ হবে large size
x-large	ফন্ট সাইজ হবে extra large size
xx-large	ফন্ট সাইজ হবে xx-large size
আপেক্ষিক মানঃ আগের উপাদানের সাপেক্ষে বর্তমান উপাদানের আকার ছোট বা বড় হয়ে থাকে।	
smaller	parent এলিমেন্টের চেয়ে ছোট আকারের ফন্ট পেতে এই মান ব্যবহার করতে হবে।
larger	parent এলিমেন্টের চেয়ে বড় আকারের ফন্ট পেতে এই মান ব্যবহার করতে হবে।
	দৈর্ঘ্য মান
length	নির্দিষ্ট সাইজের ফন্ট পেতে px, cm, etc একক ব্যবহার করা হয়।
	শতকরা মান
%	parent এলিমেন্টের সাপেক্ষে ফন্ট সাইজ নির্ধারণ করা হয়।

একটি উদাহরণ দেখুনঃ

<html>

```
<head>
<style type="text/css">
h1 {font-size:250%;}
h2 {font-size:20px;}
p {font-size:4em;}
</style>
</head>

<body>
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<p>This is a paragraph.</p>
</body>

</html>
```

আউটপুটঃ

This is heading 1

This is heading 2

This is a paragraph.

৪. ফন্ট স্টাইল প্রপার্টিঃ

ফন্টফেস অর্থাৎ ফন্টটি কেমন দেখা যাবে তার জন্য ব্যবহার করা হয় **font-style** প্রপার্টি। এর তিনটি ভ্যালু হতে আছে-

ভেলু	বর্ণনা
<i>normal</i>	স্বাভাবিক ফন্টফেস দেখাবে। এটা <i>default</i>
<i>italic</i>	ইটালিক দেখাবে।

oblique

ফন্টফেস *Oblique* ভাঁসনে দেখাবে।

একটি উদাহরণ দেখুনঃ

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
p.normal {font-style:normal;}
```

```
p.italic {font-style:italic;}
```

```
p.oblique {font-style:oblique;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p class="normal">This is a paragraph, normal.</p>
```

```
<p class="italic">This is a paragraph, italic.</p>
```

```
<p class="oblique">This is a paragraph, oblique.</p>
```

```
</body>
```

```
</html>
```

আউটপুটঃ

This is a paragraph, normal.

This is a paragraph, italic.

This is a paragraph, oblique.

৫. ফন্ট ভ্যারিয়েন্ট প্রপার্টিঃ

font-variant property নির্ধারণ করে আপনার ফন্ট **small-caps**-এ দেখাবে কি দেখাবে না। সিএসএস ফন্ট **Variant** এর সাহায্যে আপনি ফন্ট কে **small caps** এ পরিবর্তিত করতে পারেন। **small-caps** বলতে বুঝায় বড় হাতের অক্ষরগুলোর সাইজ হবে ছোট হাতের অক্ষরের সাইজের সমান। এর দুটি ভেলু হতে পারে-

ভেলু	বর্ণনা
normal	ব্রাউজার সাধারণ ফন্ট প্রকাশ করবে। এটা ডিফল্ট মান।
small-caps	এটা ব্যবহারে বড় হাতের অক্ষরগুলোর সাইজ হবে ছোট হাতের অক্ষরের সাইজের সমান।

একটি উদাহরণ দেখুনঃ

```
<html>
<head>
<style type="text/css">
p.normal {font-variant:normal;}
p.small {font-variant:small-caps;}
</style>
</head>
<body>
<p class="normal">My name is Hege Refsnes.</p>
<p class="small">My name is Hege Refsnes.</p>
</body>
</html>
```

আউটপুটঃ

My name is Hege Refsnes.

MY NAME IS HEGE REFSNES.

অনেক ফন্টের ক্ষেত্রে **small-caps** ভ্যারিয়ান্ট থাকে না। যদি কোন ফন্টের স্মলক্যাপ ভার্শন পাওয়া না যায় তবে তা বড় হাতের অক্ষর ব্যবহার করে।

৬. ফন্ট ওয়েট প্রপার্টিঃ

একটি ফন্ট কী পরিমাণ মোটা দেখা যাবে তা বোঝানোর জন্য ব্যবহার করা হয় **font-weight** প্রপার্টি। সাধারণত বিভিন্ন ফন্টের **bold, dark, heavy** ইত্যাদি ভার্শন থাকে। তাই এসব শব্দ দিয়ে **font-weight** প্রকাশ না করে সিএসএসে **font-weight** প্রপার্টির মান নির্দেশ করা হয় সংখ্যা দিয়ে। যেমন- ৪০০ অর্থ হল নরমাল ফন্ট, ৭০০হিল বোল্ড। এসব সংখ্যা ছাড়াও কয়েকটি কীওয়ার্ডের মাধ্যমে মান নির্দেশ করা যেতে পারে। এগুলো হল-**lighter, bolder, normal** এবং **bold, lighter** ও **bolder** ব্যবহার করা হলে ব্রাউজার **normal** ও বোল্ড ভার্শনের কাছাকাছি ফন্ট ভার্শন ব্যবহার করবে। যেমন-

font-weight property-যে সকল মান ব্যবহার করে-

ভেলু	বর্ণনা
normal	নরমাল <i>characters</i> ডিফাইন করে। এটা ডিফল্ট।
bold	বোল্ড <i>characters</i> ডিফাইন করে।
bolder	<i>Defines thicker characters</i>
lighter	<i>Defines lighter characters</i>

একটি উদাহরণ দেখুনঃ

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
p.normal {font-weight:normal;}
```

```
p.light {font-weight:lighter;}
```

```
p.thick {font-weight:bold;}
```

```
p.thicker {font-weight:900;}  
  
</style>  
  
</head>  
  
<body>  
  
<p class="normal">This is a paragraph.</p>  
<p class="light">This is a paragraph.</p>  
<p class="thick">This is a paragraph.</p>  
<p class="thicker">This is a paragraph.</p>  
  
</body>  
  
</html>
```

আউটপুটঃ

This is a paragraph.

This is a paragraph.

This is a paragraph.

This is a paragraph.

অধ্যায়-পাঁচ

[লিস্ট প্রপার্টি]

সিএসএস লিস্ট প্রপার্টি **unordered, ordered** এমনকি ইমেজকেও **List Marker** হিসেবে ব্যবহারের সুযোগ দেয়। সিএসএস দিয়ে লিস্ট কে এইচটিএমএল হতে অনেক বেশি কাস্টমাইজ করা যায়। এ অধ্যায়ে আমরা লিস্ট নিয়ে যাবতীয় কিছু আলোচনা করবো।

লিস্টের জন্য ব্যবহৃত সকল প্রপার্টিঃ

প্রপার্টি	বর্ণনা
1. <i>list-style</i>	<i>list-style property</i> একটি ডিকলারেশনেই সকল <i>list properties</i> কে তুলে ধরে
2. <i>list-style-image</i>	<i>list-style-image property</i> লিস্ট আইটেমকে ইমেজ দ্বারা রিপ্রেস করে।
3. <i>list-style-position</i>	<i>list-style-position property</i> নির্ধারণ করে লিস্টের আইটেমগুলো স্বাভাবিক অবস্থা থেকে <i>inside</i> না <i>Outside</i> -এ থাকবে।
4. <i>list-style-type</i>	<i>default number</i> -এর <i>ordered</i> বা <i>unordered</i> লিস্ট এর <i>bullets/discs</i> হতে ভিন্ন স্টাইল ব্যবহার করা যায়।

নিম্নে সকল প্রপার্টি নিয়ে বিস্তারিত আলোচনা করা হলঃ

১. লিস্ট স্টাইল প্রপার্টিঃ

list-style property একটি ডিকলারেশনেই সকল **list properties** কে তুলে ধরে। আর এই **list-style-type** এর মাধ্যমে এমন একটা ব্যবস্থা করা যায় যখন আপনার **list-style image** লোড নিবে না তখন অন্য **list-style** দেখাবে। **list-style shorthand property** ব্যবহারের ক্রম হল-

list-style-type > list-style-position > list-style-image

যদি কোন প্রপার্টির ভেলু বাদ যায় তবে কোন সমস্যা হবে না। এক্ষেত্রে ঐ প্রপার্টির ডিফল্ট ভেলু ধরে নেওয়া হবে।

একটি উদাহরণ দেখুনঃ

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
ul {list-style:square url("sqpurple.gif");}
</style>
</head>
<body>
<ul>
<li>Coffee</li>
<li>Tea</li>
<li>Coca Cola</li>
</ul>
</body>
</html>
```

আউটপুটঃ

- ☛ Coffee
- ☛ Tea
- ☛ Coca Cola

যদি ইমেজ লোড না হয় তবে নিচের মত আউটপুট দেখাবে।

- Coffee
- Tea
- Coca Cola

২.লিস্ট স্টাইল ইমেজ প্রপাটিঃ

ইমেজকে list item marker হিসেবে দেখাতে list-style-image property ব্যবহার করা হয়। list-style-image property লিস্ট আইটেমকে ইমেজ দ্বারা রিপ্লেস করে। .কোন কারণে ইমেজ লোড না হলে এই প্রপার্টি ব্যবহার করা হয়। সাধারণত image কে bullet points হিসাবে unordered লিস্ট এর জন্য ব্যবহার করা যায়।normal bullet এর পরিবর্তে image কে ব্যবহার করা যায়।সবচেয়ে ভাল হয় Unordered লিস্ট এর ক্ষেত্রে image কে ব্যবহার করা। নিচের উদাহরণ দেখুনঃ

```
<html>
<head>
<style type="text/css">
ul {list-style-image:url('sqpurple.gif');}
</style>
</head>
<body>
<ul>
<li>Coffee</li>
<li>Tea</li>
<li>Coca Cola</li>
</ul>
</body>
</html>
```

আউটপুটঃ

- Coffee
- Tea
- Coca Cola

যদি ইমেজ লোড না হয় তবে নিচের মত আউটপুট দেখাবে।

- Coffee
- Tea
- Coca Cola

৩.লিস্ট স্টাইল পজিশান প্রপার্টি:

`list-style-position` property নির্ধারণ করে লিস্টের আইটেমগুলো স্বাভাবিক অবস্থা থেকে `inside` না `Outside`-এ থাকবে। `list-style-position` এর ভেলু `inside` দিলে তখন লিস্টগুলো মার্জিন হতে ভিতরের দিকে থাকবে আর `Outside` দিলে `normal` লিস্ট Position এ থাকবে। `Outside` ভেলুটি একটি `default` ভ্যালু।

```
ul { list-style-position: inside; }  
ol { list-style-position: outside; }
```

একটি উদাহরণ দেখুনঃ

```
<html>  
  
<head>  
  
<style type="text/css">  
ul.a {list-style-position:inside;}  
ul.b {list-style-position:outside;}  
</style>  
</head>  
  
<body>  
  
<p>The following list has list-style-position: inside:</p>  
  
<ul class="a">
```

```
<li>Earl Grey Tea - A fine black tea</li>
<li>Jasmine Tea - A fabulous "all purpose" tea</li>
<li>Honeybush Tea - A super fruity delight tea</li>
</ul>
```

```
<p>The following list has list-style-position: outside:</p>
```

```
<ul class="b">
<li>Earl Grey Tea - A fine black tea</li>
<li>Jasmine Tea - A fabulous "all purpose" tea</li>
<li>Honeybush Tea - A super fruity delight tea</li>
</ul>
```

```
<p>"list-style-position: outside" is the default setting.</p>
```

```
</body>
```

```
</html>
```

আউটপুটঃ

The following list has list-style-position: inside:

- Earl Grey Tea - A fine black tea
- Jasmine Tea - A fabulous "all purpose" tea
- Honeybush Tea - A super fruity delight tea

The following list has list-style-position: outside:

- Earl Grey Tea - A fine black tea
- Jasmine Tea - A fabulous "all purpose" tea
- Honeybush Tea - A super fruity delight tea

"list-style-position: outside" is the default setting.

8. লিস্ট স্টাইল টাইপ প্রপার্টি:

আপনি যদি **default number** এর **ordered** লিস্ট বা **unordered** লিস্ট এর **bullets/discs** হতে ভিন্ন স্টাইল ব্যবহার করতে চান তাহলে আপনি লিস্ট এর জন্য বিভিন্ন ধরনের স্টাইল ব্যবহার করতে পারেন। এই কাজটি করা হয় নিচে **list-style-type property**-এর মাধ্যমে। যেমনঃ

```
ol { list-style-type: upper-roman; }  
ul { list-style-type: circle; }
```

একটি উদাহরণ দেখুনঃ

```
<html>  
  
<head>  
  
<style type="text/css">  
ul.a {list-style-type:circle;}  
ul.b {list-style-type:square;}  
ol.c {list-style-type:upper-roman;}  
ol.d {list-style-type:lower-alpha;}  
</style>  
</head>  
  
<body>  
  
<p>Example of unordered lists:</p>  
  
<ul class="a">  
  <li>Coffee</li>  
  <li>Tea</li>  
  <li>Coca Cola</li>
```



```
</ul>
```

```
<ul class="b">
```

```
<li>Coffee</li>
```

```
<li>Tea</li>
```

```
<li>Coca Cola</li>
```

```
</ul>
```

```
<p>Example of ordered lists:</p>
```

```
<ol class="c">
```

```
<li>Coffee</li>
```

```
<li>Tea</li>
```

```
<li>Coca Cola</li>
```

```
</ol>
```

```
<ol class="d">
```

```
<li>Coffee</li>
```

```
<li>Tea</li>
```

```
<li>Coca Cola</li>
```

```
</ol>
```

</body>

</html>

আউটপুট:

Example of unordered lists:

- Coffee
- Tea
- Coca Cola

- Coffee
- Tea
- Coca Cola

Example of ordered lists:

- I. Coffee
- II. Tea
- III. Coca Cola

- a. Coffee
- b. Tea
- c. Coca Cola

list-style-type property যে সকল ভেলু ব্যবহার করতে পারে তাদের নাম ও বর্ণনা নিম্নে দেওয়া হলঃ

ভেলু	বর্ণনা
<i>armenian</i>	<i>Armenian numbering</i> চিহ্ন প্রকাশ করে।
<i>circle</i>	<i>Circle</i> চিহ্ন প্রকাশ করে।
<i>cjk-ideographic</i>	<i>plain ideographic numbers</i> চিহ্ন প্রকাশ করে।
<i>decimal</i>	স্বাভাবিক নম্বার প্রকাশ করে। এটা <code></code> এর জন্য ডিফল্ট।
<i>decimal-leading-zero</i>	নম্বারের শুরুতে ০ থাকবে (<i>01, 02, 03, ইত্যাদি.</i>)
<i>disc</i>	ভরাট বৃত্ত প্রকাশ করে। এটা <code></code> এর জন্য ডিফল্ট।
<i>georgian</i>	<i>Georgian numbering</i> চিহ্ন প্রকাশ করে।
<i>hebrew</i>	<i>Hebrew numbering</i> চিহ্ন প্রকাশ করে।
<i>hiragana</i>	<i>Hiragana numbering</i> চিহ্ন প্রকাশ করে।
<i>hiragana-iroha</i>	<i>Hiragana iroha numbering</i> চিহ্ন প্রকাশ করে।
<i>katakana</i>	<i>Katakana numbering</i> চিহ্ন প্রকাশ করে।
<i>katakana-iroha</i>	<i>Katakana iroha numbering</i> চিহ্ন প্রকাশ করে।

lower-alpha	ছোট হাতের অক্ষর প্রকাশ করে (a, b, c, d, e ইত্যাদি)
lower-greek	ছোট হাতের গ্রীক অক্ষর প্রকাশ করে
lower-latin	ছোট হাতের ল্যাটিন অক্ষর প্রকাশ করে (a, b, c, d, e, ইত্যাদি)
lower-roman	ছোট হাতের রোমান অক্ষর প্রকাশ করে (i, ii, iii, iv, v, ইত্যাদি)
none	কোন চিহ্ন দেখাবে না।
square	Square চিহ্ন প্রকাশ করে।
upper-alpha	বড় হাতের অক্ষর প্রকাশ করে (A, B, C, D, E, ইত্যাদি)
upper-latin	বড় হাতের ল্যাটিন অক্ষর প্রকাশ করে (A, B, C, D, E, ইত্যাদি)
upper-roman	বড় হাতের রোমান অক্ষর প্রকাশ করে (I, II, III, IV, V, ইত্যাদি)

অধ্যায়-ছয়

[লিংক প্রপার্টি]

সিএসএস এর মাধ্যমে এনকর ট্যাগ বা লিংকে আমরা ইফেক্ট যোগ করতে পারি। লিংকে বিভিন্ন CSS property (color, font-family, background, etc.) ব্যবহার করে বিভিন্ন স্টাইল নির্ধারণ করা যায়। কিন্তু লিঙ্কের স্পেশাল স্টাইল নির্ধারণ করা হয় লিঙ্কের অবস্থার উপর ভিত্তি করে।

লিঙ্কের চারটি অবস্থা হল-

• a:link → একটি সাধারণ লিংক যা ভিজিট করা হয় নি।
• a:visited → একটা লিংক যা পূর্বে ভিজিট করা হয়েছে
• a:hover → এটি একটি লিংক যার উপর মাউস রাখার সাথে সাথে ভিজুয়াল ইফেক্ট দেখাবে।
• a:active → এটি একটি লিংক যেটা বর্তমানে সক্রিয় আছে। অন্যভাবে বলা যায় ইউজার যখন লিংকে ক্লিক করে রেসপন্সের জন্য অপেক্ষা করে সে সময়ের ইফেক্ট।

নোটঃ লিঙ্কের জন্য বিভিন্ন স্টাইল দিতে চাইলে নিচের নিয়ম গুলো মেনে চলতে হবে-

- **a:hover**-কে **a:link** এবং **a:visited** এর পরে স্থাপন করতে হবে।
- **a:active** -কে **a:hover**-এর পরে স্থাপন করতে হবে।

একটি উদাহরণ দেখুন-

```
<html>
```

```
<head>
```

```
<style type="text/css">
a:link {color:#FF0000;} /* unvisited link */
a:visited {color:#00FF00;} /* visited link */
a:hover {color:#FF00FF;} /* mouse over link */
a:active {color:#0000FF;} /* selected link */
</style>
</head>

<body>
<p><b><a href="default.asp" target="_blank">This is a link</a></b></p>
to be effective.</p>
</body>
</html>
```

উপরের কোডগুলো index.html নামে সেভ করে আউটপুট দেখুন।

বিভিন্ন স্টাইলের লিঙ্কের উদাহরণ দেখুনঃ

```
<html>
<head>
<style type="text/css">
a.one:link {color:#ff0000;}
a.one:visited {color:#0000ff;}
```

```
a.one:hover {color:#ffcc00;}
```

```
a.two:link {color:#ff0000;}
```

```
a.two:visited {color:#0000ff;}
```

```
a.two:hover {font-size:150%;}
```

```
a.three:link {color:#ff0000;}
```

```
a.three:visited {color:#0000ff;}
```

```
a.three:hover {background:#66ff66;}
```

```
a.four:link {color:#ff0000;}
```

```
a.four:visited {color:#0000ff;}
```

```
a.four:hover {font-family:monospace;}
```

```
a.five:link {color:#ff0000;text-decoration:none;}
```

```
a.five:visited {color:#0000ff;text-decoration:none;}
```

```
a.five:hover {text-decoration:underline;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>Mouse over the links to see them change layout.</p>
```

```
<p><b><a class="one" href="default.asp" target="_blank">This link changes color</a></b></p>
```

```
<p><b><a class="two" href="default.asp" target="_blank">This link changes font-size</a></b></p>
```

```
<p><b><a class="three" href="default.asp" target="_blank">This link changes background-color</a></b></p>
```

```
<p><b><a class="four" href="default.asp" target="_blank">This link changes font-family</a></b></p>
```

```
<p><b><a class="five" href="default.asp" target="_blank">This link changes text-decoration</a></b></p>
```

```
</body>
```

```
</html>
```

উপরের কোডগুলো index.html নামে সেভ করে আউটপুট দেখুন।

অধ্যায়-সাত

[ব্যাকগ্রাউন্ড প্রপার্টি]

আপনার সাইটের ব্যাকগ্রাউন্ড অনেক বেশি গুরুত্বপূর্ণ। HTML-এ শুধু bgcolor ব্যবহার করে পুরো Background -এর রঙ বদলানো যায়। কিন্তু সিএসএস এর মাধ্যমে আপনি যেকোন HTML এলিমেন্টের (Heading, Paragraph, Link, Table, Span etc. এর background color or image সেট করতে পারেন। তাছাড়া background image কিভাবে প্রদর্শিত হবে সে বিষয়টি আপনি নিয়ন্ত্রন করতে পারবেন। আপনি পছন্দমত এটিকে horizontally, vertically repeat করতে পারেন। তাছাড়া আপনি background কে fixed position এ রাখতে পারেন বা scroll করতে পারেন। আমরা এ অধ্যায়ে ব্যাকগ্রাউন্ড নিয়ে বিস্তারিত আলোচনা করবো।

সিএসএস-এ ব্যবহৃত সকল ব্যাকগ্রাউন্ড প্রপার্টি

প্রপার্টি	বর্ণনা
1.background	সকল background properties-কে একটি ডিকলারেশনের মাধ্যমে তুলে ধরা যায়।

2.background-attachment	এই প্রপার্টি নির্ধারণ করে পেজ স্ক্রলিং-এর সাথে সাথে background ইমেজটি fixed থাকবে না পেজের সাথে সাথে স্ক্রল করবে।
3.background-color	background-color property কোন এলিমেন্টের background color নির্ধারণ করে।
4.background-image	background-image property ব্যবহার করে কোন এলিমেন্টের ব্যাকগ্রাউন্ডে ইমেজ সেট করা যায়।
5.background-position	ইমেজকে ঠিক কোন স্থানে স্থাপন করবেন তা বলে দেওয়া যেতে পারে background-position Property ব্যবহার করে।
6.background-repeat	কোন ব্যাকগ্রাউন্ড ইমেজ repeat হবে কি না, হলে কোন দিকে কি পরিমাণ হবে তা নির্ধারণ করা হয় CSS-এর background-repeat প্রপার্টির মাধ্যমে।

১.ব্যাকগ্রাউন্ড প্রপার্টিঃ

ব্যাকগ্রাউন্ড প্রপার্টির মাধ্যমে সকল background properties-কে একটি ডিকলারেশনের মাধ্যমে তুলে ধরা যায়। background প্রপার্টির অর্ডার হল- background-color>background-image>background-repeat>background-attachment>background-position. এদের কোনটার ভেলু বাদ পড়লে সমস্যা নেই।

একটি উদাহরণ দেখুন-

```
<head>
```

```
<style type="text/css">
```

```
body
```

```
{
```

```
background: #00ff00 url('smiley.gif') no-repeat fixed center;
```

```
}
```

```
</style>
```

```
</head>
```

২.ব্যাকগ্রাউন্ড এটাচমেন্ট প্রপার্টিঃ

background-attachment property নির্ধারণ করে পেজ স্ক্রলিং-এর সাথে সাথে **background** ইমেজটি **fixed** থাকবে না পেজের সাথে সাথে স্ক্রল করবে। আপনি ব্রাউজার উইন্ডোর স্ক্রলবার স্ক্রল করুন। ব্যাকগ্রাউন্ড ইমেজও পেজ স্ক্রলিংয়ের সাথে সাথে স্ক্রল হচ্ছে। এখন আপনি চাচ্ছেন পেজ স্ক্রল করবেন কিন্তু ব্যাকগ্রাউন্ড ইমেজ যেন স্থির থাকে। এটি করা সম্ভব **background-attachment** প্রপার্টির মাধ্যমে। পেজের টেক্সটের সাপেক্ষে স্থির এ ব্যাকগ্রাউন্ড ইমেজকে বলা হয় **Watermark**। ইমেজকে **Watermark** করার জন্য নিচের মত কোড লিখুন-

একটি উদাহরণ দেখুন-

```
<html>

<head>

<style type="text/css">

body

{

background-image:url('22.png');

background-repeat:repeat;

background-attachment:fixed;

}

div{height:1200px; width:960px; background:#112200; margin:0 auto;}

</style>

</head>

<body>

<div>

<p>This is a division</p>

</div>
```


</body>

</html>

উপরের উদাহরণটি ভালভাবে প্রাকটিচ করে দেখুন।

background-attachment property নিচের দুটি মান ব্যবহার করে-

ভেলু	বর্ণনা
scroll	ব্যাকগ্রাউন্ড ইমেজও পেজ স্ক্রলিংয়ের সাথে সাথে স্ক্রল হবে। এটা ডিফল্ট। .
fixed	ব্যাকগ্রাউন্ড ইমেজ ফিক্সড থাকবে।

৩. ব্যাকগ্রাউন্ড কালার প্রপার্টি:

background-color property কোন এলিমেন্টের **background color** নির্ধারণ করে। কোন এলিমেন্টের ব্যাকগ্রাউন্ড হল ঐ এলিমেন্টের মোট সাইজের সমান, যাতে **padding** এবং **border** অন্তর্ভুক্ত থাকবে কিন্তু **margin** থাকবে না। সাধারণত **background color** এবং **text color** একসাথে **background-color** প্রপার্টির সাথে ব্যবহার করা যায় যাতে টেক্সট ভাল পড়া যায়।

```
body
{
background-color:yellow;
}
h1
{
background-color:#00ff00;
}
p
{
background-color:rgb(255,0,255);
}
```

৪. ব্যাকগ্রাউন্ড ইমেজ প্রপার্টি:

background-image property ব্যবহার করে কোন এলিমেন্টের ব্যাকগ্রাউন্ডে ইমেজ সেট করা যায়।

```
Body { background-image: url(largePic.jpg); }
p { background-image: url(smallPic.jpg); background-color:#00ff00 }
```

এর ফলে background এ এক ছবি এবং Paragraph এ আরেক ছবি দেখা যাবে।

<code>url('URL')</code>	ইমেজের 'URL'।
<code>none</code>	কোন ব্যাকগ্রাউন্ড ইমেজ থাকবে না। এটা ডিফল্ট।

৫.ব্যাকগ্রাউন্ড পজিশান প্রপার্টি:

কোন ব্যাকগ্রাউন্ড ইমেজকে ঠিক কোন স্থানে স্থাপন করবেন তা বলে দেওয়া যেতে পারে **background-position Property** ব্যবহার করে। এর মান হিসেবে ব্যবহার করা যেতে পারে- দৈর্ঘ্য , শতকরা হার ও কিওয়াড। শতকরা হার ব্যবহার করা হয় দৈর্ঘ্য/প্রস্থ অবস্থান নির্দেশের জন্য। যেমন- ০%০% ব্যবহার করলে ঐ ইমেজ ব্রাউজার উইন্ডোর একদম উপরের বাম কোণায় অবস্থান করবে। আবার ৫০%৫০% ব্যবহার করলে তার অবস্থান হবে মধ্যবিন্দুতে। দৈর্ঘ্যের পরিমাপসমূহ (যেমন 4em 50%) হল **Vertical** অবস্থান নির্দেশক। কী ওয়াড ব্যবহারের বেলায় আনুভূমিক। লম্বিক মানের পরিবর্তন ঘটানো যেতে পারে, সেক্ষেত্রে কোন অসুবিধা হবে না। কারণ কীওয়াডগুলোর তিনটি (**Top, Middle, Bottom**) উল্লম্ব (**vertical**) অবস্থান নির্দেশক, আর অন্য তিনটি (**Left, Right, Center**) আনুভূমিক অবস্থান নির্দেশক। যেমন-

```
Body{background-position: top left}
```

এ ডিকলারেশনকে অন্যভাবেও লেখা যায়।

```
Body{background-position: left top}
```

বা, `Body{background-position: ০% ০%}`

background-position Property-তে ব্যবহৃত মানসমূহ-

ভেলু	বর্ণনা
	কিওয়াড
<i>left top</i> <i>left center</i> <i>left bottom</i> <i>right top</i> <i>right center</i> <i>right bottom</i> <i>center top</i> <i>center center</i>	যদি আপনি শুধু একটি keyword ব্যবহার করেন অন্য ভেলু হবে " center "

<i>center bottom</i>	
শতকরা হার	
<i>x% y%</i>	প্রথম মানটা <i>horizontal</i> অবস্থান এবং দ্বিতীয় ভেলুটা <i>the vertical</i> অবস্থান নির্দেশ করে। আপনি যদি একটি মান ঠিক করে দেন তবে অপর মানটি ৫০% হবে। ডিফল্ট মান হল- ০% ০%।
দৈর্ঘ্য	
<i>xpos ypos</i>	প্রথম মানটা <i>horizontal</i> অবস্থান এবং দ্বিতীয় ভেলুটা <i>the vertical</i> অবস্থান নির্দেশ করে। এই মান পিক্সেল বা অন্য কোন এককে হতে পারে। আপনি যদি একটি মান ঠিক করে দেন তবে অপর মানটি ৫০% হবে। আপনি % এবং <i>positions</i> এড্রু করতে পারেন।

একটি উদাহরণ দেখুন-

```
<html>
<head>
<style type="text/css">
body
{
background-image:url('smiley.gif');
background-repeat:no-repeat;
background-attachment:fixed;
background-position:center;
}
</style>
</head>
<body>
```

<p>Note: For this to work in Firefox and Opera, the background-attachment property must be set to "fixed".</p>

</body>

</html>

৬. ব্যাকগ্রাউন্ড রিপিট প্রপার্টি:

কোন ব্যাকগ্রাউন্ড ইমেজ রিপিট হবে কি না, হলে কোন দিকে কি পরিমাণ হবে তা নির্ধারণ করা হয় CSS-এর background-repeat প্রপার্টির মাধ্যমে। background-repeat প্রপার্টির মানগুলো হতে পারে-

ভেলু	বর্ণনা
repeat	ব্যাকগ্রাউন্ড ইমেজ <i>vertically</i> এবং <i>horizontally</i> রিপিট হবে। এটা ডিফল্ট।
repeat-x	ব্যাকগ্রাউন্ড ইমেজ শুধু <i>horizontally</i> রিপিট হবে।
repeat-y	ব্যাকগ্রাউন্ড ইমেজ শুধু <i>vertically</i> রিপিট হবে।
no-repeat	ব্যাকগ্রাউন্ড ইমেজ রিপিট হবে না।

```
p {
background-image: url (smallPic.jpg);
background-repeat: repeat; }
h4 {
background-image: url (smallPic.jpg);
background-repeat: repeat-y;}
ol {
background-image: url (smallPic.jpg);
background-repeat: repeat-x;}
ul {
background-image: url (smallPic.jpg);
background-repeat: no-repeat;}
```

অধ্যায়-আট

[পজিশান প্রপার্টি]

সিএসএস পজিশন এর মাধ্যমে আমরা এইচটিএমএল এলিমেন্টস এর সঠিক (**exact**) পজিশন নির্দেশ করতে পারি। এর মাধ্যমে আমরা একটি এলিমেন্টের পিছনে আরেকটি এলিমেন্ট স্থাপন করতে পারি এবং আরও নির্ধারণ করতে পারি যে যদি এলিমেন্ট কনটেন্ট বেশি বড় হয় তবে কি ঘটবে। **top, bottom, left,** এবং **right** প্রপার্টি ব্যবহার করে এলিমেন্টের অবস্থান ঠিক করা যায়। কিন্তু যতক্ষণ না **position property** সেট করা হবে ততক্ষণ এইসব প্রপার্টি কাজ করবে না। তারা **positioning method**-এর উপর ভিত্তি করে বিভিন্নভাবে কাজ করে থাকে। চার ধরনের **positioning method** আছে যা নিম্নরূপ-

1. Static Positioning

2. Fixed Positioning

3. Relative Positioning

4. Absolute Positioning

উদাহরণসহ নিম্নে আলোচনা করা হলঃ

1. Static Positioning:

HTML এলিমেন্টগুলো সাধারণত স্ট্যাটিক **positioned** হয়, অর্থাৎ কোন একটি HTML উপাদানের পজিশন প্রপার্টি যদি স্ট্যাটিক করে দেওয়া হয়; তাহলে তা HTML উপাদান সমূহের সাধারণ এবং স্বাভাবিক পজিশন নির্দেশ করবে। এক্ষেত্রে **top, bottom, left,** এবং **right** প্রপার্টি সমূহ ডিক্লারেশন করলেও এ অনুযায়ী কোন কাজ হবে না; অর্থাৎ **top, bottom, left, and right** প্রপার্টি সমূহ নিষ্ক্রিয় হয়ে যাবে। তাই বলা যায় **static positioned** প্রপার্টিগুলো **top, bottom, left,** এবং **right** প্রপার্টি দ্বারা প্রভাবিত হয় না।

একটি উদাহরণ দেখুন-

```
<html>
```

```
<head>
```

```
<style>
```

```
.static{
```

```
position:static;
```

```
left:100px;
```

```
top:70px;}
```

```
</style>
```

```
</head>
```

```
<body >
```

```
<p class="static">The CSS positioning properties allow you to position an  
element. </p>
```

```
<p>The CSS positioning properties allow you to position an element. </p>
```

```
</body>
```

```
</html>
```

আউটপুটঃ

The CSS positioning properties allow you to position an element.

The CSS positioning properties allow you to position an element.

কোড বিশ্লেষণঃ উপরের উদাহরণটিতে লক্ষ্য করুন আমরা দুটি প্যারাগ্রাফ নিয়েছি এবং একটার পজিসান স্টাটিক করে দিয়েছি এবং অন্যটা স্বাভাবিক অবস্থায় আছে কিন্তু তারপরও দুটি প্যারাগ্রাফের আউটপুটে কোন পরিবর্তন নাই , এটা থেকে বুঝা যায় যে, তাই বলা যায় **static positioned** প্রপাটিগুলো **top, bottom, left,** এবং **right** প্রপাটি দ্বারা প্রভাবিত হয় না।

2.Fixed Positioning:

কোন একটি HTML উপাদানকে মনিটর স্ক্রিনের সাপেক্ষে কোন নির্দিষ্ট স্থানে স্থিরভাবে স্থাপন করার জন্য পজিশন ফিক্সড ব্যবহার করা হয়। একটা এলিমেন্টের পজিশন যদি ফিক্সড করে দেন তাহলে সেটা ব্রাউজারের উইন্ডো সাপেক্ষে অবস্থান করবে এবং ব্রাউজার যদি স্ক্রল করেন তবুও সেটা স্থির (**fixed**) থাকবে। এজন্য প্রথমে **position:fixed;** ডিক্লারেশন করে, এর পর **top, bottom, left, and right** প্রোপাটি সমূহ ডিক্লারেশন করতে হয়। **Fixed positioned elements** গুলো অন্যান্য এলিমেন্টকে **overlap** করতে পারে।

একটি উদাহরণ দেখুনঃ

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
p.pos_fixed
```

```
{
```

```
position:fixed;
```

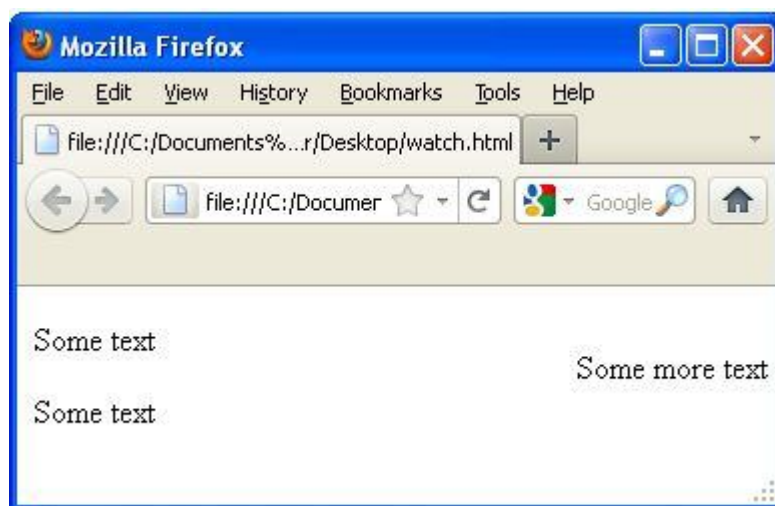
```
top:30px;
```

```
right:5px;
```

```

}
</style>
</head>
<body>
<p class="pos_fixed">Some more text</p>
<p><b>Note:</b> IE7 and IE8 supports the fixed value only if a
!DOCTYPE is specified.</p>
<p>Some text</p><p>Some text</p><p>Some text</p><p>Some text</p><p>Some
text</p><p>Some text</p><p>Some text</p><p>Some text</p><p>Some
text</p><p>Some text</p><p>Some text</p><p>Some text</p><p>Some
text</p><p>Some text</p><p>Some text</p><p>Some text</p>
</body>
</html>

```



উইন্ডোটা স্ক্রল করলেও ডানপাশের **some more text** লেখাটি স্ক্রল করবে না।

3. Relative Positioning:

এইচটিএমএল উপাদান সমূহের স্বাভাবিক অবস্থানের সাপেক্ষে **top, bottom, left, and right** প্রোপার্টি ব্যবহারের মাধ্যমে কোন বস্তুর অবস্থান নির্ধারণের জন্য রিলেটিভ পজিশন প্রোপার্টি ব্যবহার করা হয়। পজিশন রিলেটিভ প্রোপার্টির ক্ষেত্রে সকল হিসাব শুরু হয় ঐ উপাদানটির স্বাভাবিক অবস্থান থেকে। অর্থাৎ পজিশনিং না করা হলে উপাদানটি যে স্থানে অবস্থান করতো সেখান থেকে। **relatively positioned** এলিমেন্টের উপাদানগুলো অন্যান্য এলিমেন্টের উপাদানকে মুভ বা ওভারল্যাপ করতে পারে।

একটি উদাহরণ দেখুন-

```
<html>
<head>
<style type="text/css">
h2.pos_left
{
position:relative;
left:-20px;
}

h2.pos_right
{
position:relative;
left:20px;
}
</style>
</head>

<body>
<h2>This is a heading with no position</h2>
<h2 class="pos_left">This heading is moved left according to its normal
position</h2>
<h2 class="pos_right">This heading is moved right according to its normal
position</h2>
<p>Relative positioning moves an element RELATIVE to its original
position.</p>
</body>
```

</html>

রিলেটিভ পজিশন এর ক্ষেত্রে যে সকল প্রপাটি ব্যবহার করা হয় তাদের কিছু শর্ত দেখুনঃ

Move Left - এক্ষেত্রে *left* এর জন্য **negative value** ব্যবহার করতে হবে।

Move Right - এক্ষেত্রে *left* এর জন্য **positive value** ব্যবহার করতে হবে।

Move Up - এক্ষেত্রে *top* এর জন্য **negative value** ব্যবহার করতে হবে। .

Move Down - এক্ষেত্রে *left* এর জন্য **positive value** ব্যবহার করতে হবে।

4. Absolute Positioning:

HTML উপাদান সমূহের প্রকৃত অবস্থান নির্ধারণের জন্য পজিশন এবসলিউট প্রোপাটি ব্যবহার করা হয়। কোন একটি HTML উপাদানের পজিশন প্রোপাটি যদি এবসলিউট করে দেওয়া হয় তাহলে তার অবস্থান স্বভাবিক ক্রমে নির্দেশিত হয় না, এক্ষেত্রে **top, bottom, left, and right** প্রোপাটি সমূহের মাধ্যমে ঐ উপাদানটির অবস্থান নির্ধারিত হয়। এক্ষেত্রে অন্য একটি উপাদানের উপরেও ওভারল্যাপ করে ঐ উপাদানটির অবস্থান নির্ধারণ করা সম্ভব হয়।

উদাহরন:

```
<html>
<head>
<style type="text/css">
h2
{
position:absolute;
left:100px;
top:150px;
}
</style>
</head>

<body>
<h2>This is a heading with an absolute position</h2>
```

<p>With absolute positioning, an element can be placed anywhere on a page. The heading below is placed 100px from the left of the page and 150px from the top of the page.</p>

</body>

</html>

সিএসএস-এ ব্যবহৃত আরও কিছু পজিশান প্রপার্টি

ক্লিপ প্রপার্টি (Clip Property):

কোন উপাদানের কোন অংশ দৃশ্যমান হবে তা নির্ধারণ করা হয় **Clip** প্রপার্টির মাধ্যমে। যেমন- কোন ইমেজের সাইজ অনেক বড় হলে তার কতটুকু দেখা যাবে তা এই প্রপার্টির মাধ্যমে নির্ধারণ করা যায়। এর সিনট্যাক্স হল-

Clip: sClip

এখানে **sClip** এর মান হতে পারে দুটি-

shape	একটি এলিমেন্ট ক্লিপ করবে যার মান হতে পারে- <i>rect (top, right, bottom, left)</i>
auto	কোন ক্লিপিং হবে না। এটা ডিফল্ট।

কোন সাইডের মান দেয়া না হলে তা **auto** হিসেবে ধরে নেয়, অর্থাৎ সে দিকটা **clip** করে না। ক্লিপ প্রপার্টি কাজ করে না যদি "**overflow:visible**" লিখা হয়। কোন ইমেজকে ক্লিপ করানোর জন্য নিচের মত ইমেজ ডিকলারেশন ব্যবহার করা যেতে পারে-

<html>

<head>

<style type="text/css">

img

{

position:absolute;

```
clip:rect(0px,60px,200px,0px);  
}  
</style>  
</head>  
  
<body>  
  
  
  
</body>  
  
</html>
```

কর্সর প্রপার্টি (cursor Property):

আমরা উইন্ডোজ, লিনাক্স ব্যবহার সময় বিভিন্ন মাউস কর্সর আইকন দেখেছি। বিভিন্ন পরিস্থিতি সাপেক্ষে মাউস কর্সর এর আইকন বিভিন্ন হয়। জনসাধারণ যখন আপনার **site visit** করবে সেখানেও বিভিন্ন পরিস্থিতি সাপেক্ষে মাউস এর আইকন বিভিন্ন করে তৈরী করা যায়। এটি বেশি ব্যবহৃত হয় লিঙ্কের জন্য। যা সিএসএস দ্বারা করা সম্ভব। যেমন-

```
p { cursor: wait }  
h4 { cursor: help }  
h5 { cursor: crosshair }
```

যে কর্সর ব্যবহার করবেন তার **URL** -ও লেখা যেতে পারে-

```
P{cursor: url("sajkdg.cur") url("sdjg.cur"),text}
```

কর্সর প্রপার্টি যে সকল মান ব্যবহার করতে পারে-

```
cursor: auto  
cursor: inherit  
cursor: crosshair  
cursor: default  
cursor: help  
cursor: move  
cursor: pointer  
cursor: progress
```

```
cursor: text
cursor: wait
cursor: e-resize
cursor: ne-resize
cursor: nw-resize
cursor: n-resize
cursor: se-resize
cursor: sw-resize
cursor: s-resize
cursor: w-resize
```

একটি উদাহরণ দেখুন-

```
<html>
<body>
<p>Mouse over the words to change the cursor.</p>
<span style="cursor:auto">auto</span><br />
<span style="cursor:crosshair">crosshair</span><br />
<span style="cursor:default">default</span><br />
<span style="cursor:e-resize">e-resize</span><br />
<span style="cursor:help">help</span><br />
<span style="cursor:move">move</span><br />
<span style="cursor:n-resize">n-resize</span><br />
<span style="cursor:ne-resize">ne-resize</span><br />
<span style="cursor:nw-resize">nw-resize</span><br />
<span style="cursor:pointer">pointer</span><br />
<span style="cursor:progress">progress</span><br />
<span style="cursor:s-resize">s-resize</span><br />
<span style="cursor:se-resize">se-resize</span><br />
<span style="cursor:sw-resize">sw-resize</span><br />
<span style="cursor:text">text</span><br />
<span style="cursor:w-resize">w-resize</span><br />
<span style="cursor:wait">wait</span><br />
</body>
</html>
```

ওভারফ্লো প্রপার্টি(Overflow Property):

যদি এলিমেন্ট কনটেন্ট বক্সের মাঝে এলিমেন্ট কনটেন্ট না ধরে তবে কি ঘটে তা নির্ধারণ করা হয় **overflow** প্রপার্টির মাধ্যমে। অন্যভাবে বলা যায়- **top left, height** ও **width** প্রপার্টির মাধ্যমে নির্ধারণকৃত বক্সের মাঝে কনটেন্ট না ধরলে তার কি হবে তা নির্ধারণ করা হয় **overflow** প্রপার্টির মাধ্যমে। এই প্রপার্টির চারটি মান হতে পারে-

visible	পুরো কনটেন্ট আঁটানোর জন্য উপাদানের আকার পরিবর্তন করবে।এটি ডিফল্ট মান।
hidden	এটি ব্যবহার করা হলে সাইজের বেশি টেক্সট দেখা যাবে না।
scroll	কনটেন্ট উপাদানের নির্ধারিত আকারের চেয়ে বেশি বা কম যাই হোক না কেন সবসময় স্ক্রলবার প্রদর্শন করবে।
auto	কনটেন্ট নির্ধারিত উপাদানের আকারের চেয়ে বেশি হলেই কেবল স্ক্রলবার দেখাবে।

```

<html>
<head>
<style type="text/css">
div.scroll
{
background-color:#00FFFF;
width:100px;
height:100px;
overflow:scroll;
}

div.hidden
{
background-color:#00FF00;
width:100px;
height:100px;
overflow:hidden;
}
</style>
</head>

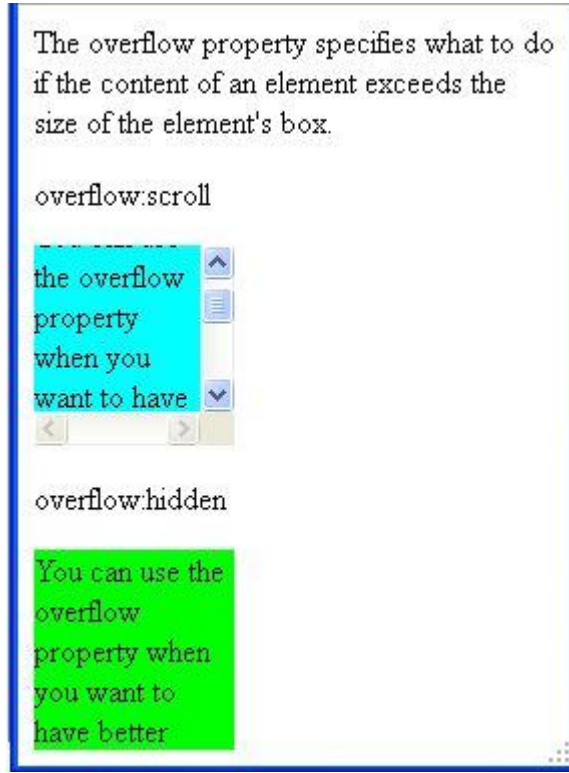
<body>
<p>The overflow property specifies what to do if the content of an element exceeds
the size of the element's box.</p>

<p>overflow:scroll</p>
<div class="scroll">You can use the overflow property when you want to have better
control of the layout. The default value is visible.</div>

```

```
<p>overflow:hidden</p>  
<div class="hidden">You can use the overflow property when you want to have better  
control of the layout. The default value is visible.</div>  
</body>  
</html>
```

আউটপুটঃ



জেড ইনডেক্স (Z-index Property):

পজিশন এবসলিউট (position:absolute) করে এবং top, bottom, left, and right ইত্যাদি প্রোপার্টি ব্যবহার করে একাধিক HTML উপাদানকে পরস্পরের উপর ওভারল্যাপ করা যায়। একাধিক HTML উপাদানকে পরস্পরের উপর ওভারল্যাপ করা কোন উপাদানটি কার উপরে অবস্থান করবে তা জেড ইনডেক্স (Z-index) প্রোপার্টি ব্যবহার করে নির্ধারণ করা হয়। (Z-index) প্রোপার্টির মান হিসেবে আমরা auto এবং কোন সংখ্যা যেমন 1,2,3,-1 প্রভৃতি ব্যবহার করা হয়।

আমরা দেখেছি `position: absolute` এর সাথে `top`, `bottom`, `left`, `and right` ইত্যাদি প্রপাটি মিশিয়ে বিভিন্ন উপাদানকে ওভারল্যাপ করা যেতে পারে। বিভিন্ন উপাদান ওভারল্যাপ করানো হলে কোনটি কার উপরে অবস্থান করবে তা জেড ইনডেক্স (Z - i n d e x) প্রোপাটি ব্যবহার করে নির্ধারণ করা হয়।

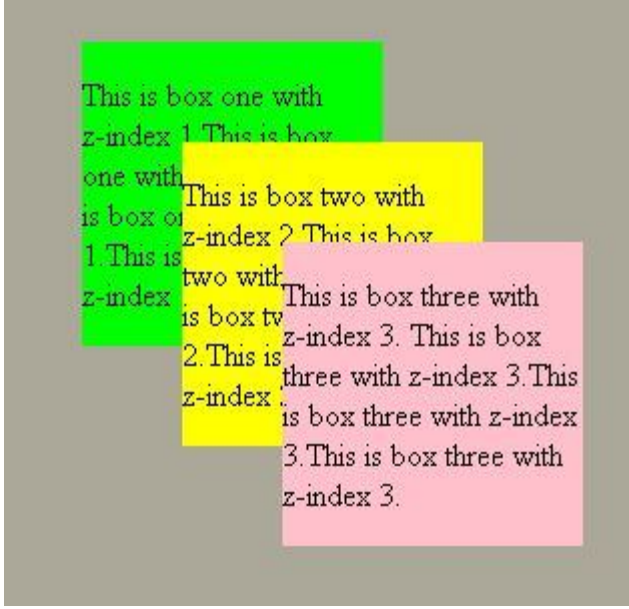
z-index প্রপাটি দুই ধরনের মান ব্যবহার করতে পারে-

<code>auto</code>	Sets the stack order equal to its parents. This is default
<code>number</code>	Sets the stack order of the element. Negative numbers are allowed

একটি উদাহরণ দেখুন-

```
<html>
<head>
<title>Z-index Example</title>
</head>
<body>
<div ID="one" style="position:absolute; left:50; top:50; width:150; background-color:lime; z-index:1">
<p>This is box one with z-index 1.This is box one with z-index 1.This is box one with z-index 1.
This is box one with z-index 1.</p></div>
<div ID="two" style="position:absolute; left:100; top:100; width:150; background-color:yellow; z-index:2">
<p>This is box two with z-index 2.This is box two with z-index 2.This is box two with z-index 2.
This is box two with z-index 2. </p>
</div>
<div ID="three" style="position:absolute; left:150; top:150; width:150; background-color:pink; z-index:3">
<p>This is box three with z-index 3. This is box three with z-index 3.This is box three with z-index 3
.This is box three with z-index 3.</p>
</div>
</body>
</html>
```

প্রদর্শনঃ



অধ্যায়-নয়

[বক্স প্রপাটি]

সিএসএস এর ক্ষেত্রে প্রতিটি ব্লক লেভেল উপাদানকে একটি বক্স হিসেবে ধরে নেয়া হয়। এই বক্সের আছে কিছু প্রপাটি। সিএসএস-এ এসব প্রপাটি ও তার ব্যবহারকে বক্স মডেল বলে অভিহিত করা হয়। এই বক্সের বিভিন্ন প্রপাটি পরিবর্তন করে বিশেষ করে কোন উপাদানের চারপাশে বর্ডার বা ইনডেন্ট তৈরির জন্য বক্স মডেল বোঝা খুব দরকার। ডিজাইন ও লেআউটের জন্য "box model" খুবই গুরুত্বপূর্ণ। "box model" আসলে একটি বক্স যা HTML উপাদানকে ঘিরে থাকে। বক্স মডেলের উপাদানগুল হল-

1.margins 2.borders 3.padding 4.contents

বক্স মডেলের গঠন নমুনা দেখানো হলঃ



Margin - Margin বলতে এইচটিএমএল এলিমেন্ট এর border এর চারপাশের অংশকে বা এলিমেন্টের চারপাশের অংশকে বোঝায়

Border - padding এবং content-এর চারপাশের অংশই হল বর্ডার। বর্ডার ব্যাকগ্রাউন্ড কালার দ্বারা প্রভাবিত হয়।

Padding - বর্ডার এবং কনটেন্টের মাঝের ফাঁকা স্থানই হল padding । padding বর্ডার ব্যাকগ্রাউন্ড কালার দ্বারা প্রভাবিত হয়।

Content - যেখানে এইচটিএমএল এলিমেন্ট যেমন-টেক্সট , ইমেজ ইত্যাদি থাকে।

কোন কনটেন্টের জন্য ফিক্সড কোন এরিয়া নির্ধারণ করতে আমরা সিএসএস- এ সাধারণত width and height প্রপার্টি ব্যবহার করে থাকি। কিন্তু পুরো এরিয়া নির্ধারণ করতে padding, border and margin ব্যবহার করতে হবে।

৩০০ পিক্সেল কনটেন্ট এরিয়ার জন্য ডিকলারেশন দেখুনঃ

```
width:250px;
padding:10px;
border:5px solid gray;
margin:10px;
```

হিসাব করে দেখা যাকঃ

```
250px (width)
+ 20px (left and right padding)
+ 10px (left and right border)
```

+ 20px (left and right margin)
= 300px

নিচের সূত্র প্রয়োগ করে হিসাব করুনঃ

→ Total element width = width + left padding + right padding + left border + right border + left margin + right margin

→ Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

একটি উদাহরণ দেখুনঃ

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
div.ex
```

```
{
```

```
width:220px;
```

```
padding:10px;
```

```
border:5px solid gray;
```

```
margin:0px;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<br /><br />
```

```
<div class="ex">The line above is 250px wide.<br />
```

```
The total width of this element is also 250px.</div>
```

```
</body>
```

```
</html>
```

আউটপুটঃ

The line above is 250px wide.
The total width of this element
is also 250px.

সিএসএস-এ ব্যবহৃত সকল *Margin Properties:*

প্রপার্টি	বর্ণনা
<i>margin</i>	সকল মার্জিন প্রপার্টিকে একটি প্রপার্টি-তে নিয়ে আসা যায় <i>margin shorthand property ("margin")</i> ব্যবহার করে।
<i>margin-bottom</i>	এলিমেন্টের <i>bottom margin</i> -এর জন্য ব্যবহার করা হয়।
<i>margin-left</i>	এলিমেন্টের <i>left margin</i> -এর জন্য ব্যবহার করা হয়।
<i>margin-right</i>	এলিমেন্টের <i>right margin</i> -এর জন্য ব্যবহার করা হয়।
<i>margin-top</i>	এলিমেন্টের <i>top margin</i> -এর জন্য ব্যবহার করা হয়।

ধারাবাহিকভাবে প্রপার্টিগুলো নিম্নে আলোচনা করা হলঃ

1.Margin - Shorthand property

সকল margin property কে একটি property-তে নিয়ে আসা যায় margin shorthand property ("margin")ব্যবহার করে। margin প্রপার্টি top>right>bottom>left এই ক্রম মেনে চলে।

margin property চারটি ভেলু নিয়ে গঠিত।

- **margin:25px 50px 75px 100px;**

বিস্তারিত কোড হবে নিচের মতঃ

- top margin is 25px
- right margin is 50px
- bottom margin is 75px
- left margin is 100px

ডিকলারেশনঃ

```
<head>
<style type="text/css">
p.ex1 {margin:2cm 4cm 3cm 4cm}
</style>
</head>
```

কোন margin property এর ভ্যালু নির্ধারণ করে না দিলে তার মান হবে বিপরীত দিকের মানের সমান। যেমনঃ

`Img{margin:1cm 2cm}` হয় তবে,

`Img{margin:1cm 2cm 1cm 2cm }` হবে।

- **margin:25px 50px 75px;**
 - top margin is 25px
 - right and left margins are 50px [বিপরীত দিকের মান ব্যবহার করেছে]
 - bottom margin is 75px
- **margin:25px 50px;**
 - top and bottom margins are 25px
 - right and left margins are 50px
- **margin:25px;**
 - all four margins are 25px

একটি উদাহরণ দেখুনঃ

`<html>`

```
<head>

<style type="text/css">

p
{
background-color:yellow;
}

p.margin
{
margin-top:100px;
margin-bottom:100px;
margin-right:50px;
margin-left:50px;
}

</style>

</head>

<body>

<p>This is a paragraph with no specified margins.</p>

<p class="margin">This is a paragraph with specified margins.</p>

</body>

</html>
```

margin property যে সকল মান ব্যবহার করতে পারেঃ

<i>auto</i>	ব্রাউজার নিজেই মার্জিন সেট করে নিবে। এটা ব্রাউজারের উপর নির্ভর করে।
<i>length</i>	<i>px, pt, em</i> এককে <i>fixed margin</i> ব্যবহার করা যায়।
<i>%</i>	এলিমেন্টের সাপেক্ষে শতকরায় মার্জিন প্রকাশ করা হয়।

2. Border Property:

এইচটিএমএল এলিমেন্টস এর চারপাশে যে বর্ডার দেখা যায় তা সিএসএস বর্ডার এর মাধ্যমে সম্পূর্ণভাবে *customize* (style, color)করা যায়।

সিএসএস-এ ব্যবহৃত সকল বর্ডার প্রপার্টি:

প্রপার্টি	বর্ণনা
1. border	<i>border shorthand property ("border":)</i> সকল <i>border properties</i> -কে একটি ডিকলারেশনের মাধ্যমে প্রকাশ করে।
2. border-bottom	<i>border-bottom shorthand property(border-bottom:)</i> সকল <i>bottom border properties</i> -কে একটি ডিকলারেশনের মাধ্যমে প্রকাশ করে।
3. border-bottom-color	<i>border-bottom-color property</i> কোন এলিমেন্টের <i>bottom border</i> -এর কালার নির্ধারণ করে।
4. border-bottom-style	<i>Sets the style of the bottom border</i>
5. border-bottom-width	<i>Sets the width of the bottom border</i>
6. border-color	<i>Sets the color of the four borders</i>
7. border-left	<i>border-left shorthand property</i> সকল <i>left border properties</i> কে একটি ডিকলারেশনের মাধ্যমে প্রকাশ করে।
8. border-left-color	<i>Sets the color of the left border</i>
9. border-left-style	<i>Sets the style of the left border</i>
10. border-left-width	<i>Sets the width of the left border</i>
11. border-right	<i>Sets all the right border properties in one declaration</i>
12. border-right-color	<i>Sets the color of the right border</i>
13. border-right-style	<i>Sets the style of the right border</i>

14.border-right-width	<i>Sets the width of the right border</i>
15.border-style	<i>Sets the style of the four borders</i>
16.border-top	<i>Sets all the top border properties in one declaration</i>
17.border-top-color	<i>Sets the color of the top border</i>
18.border-top-style	<i>Sets the style of the top border</i>
19.border-top-width	<i>Sets the width of the top border</i>
20.border-width	<i>Sets the width of the four borders</i>

1.border property:

border shorthand property ("border:") সকল border properties-কে একটি ডিকলারেশনের মাধ্যমে প্রকাশ করে। ডিকলারেশনের ক্রম হল→ border-width>border-style>border-color। কোন ভ্যালু বাদ পড়লে সমস্যা নেই, এক্ষেত্রে ঐ প্রপাটির ডিফল্ট মান বসবে।

একটি উদাহরণ দেখুন-

```

1  <html>
2  <head>
3  <style type="text/css">
4  p
5  {
6  border:5px solid red;
7  }
8  </style>
9  </head>
10
11 <body>
12 <p>This is some text in a paragraph.</p>
13 </body>
14 </html>

```

2.border-bottom property:

border-bottom shorthand property(border-bottom:) সকল bottom border properties-কে একটি ডিকলারেশনের মাধ্যমে প্রকাশ করে। ডিকলারেশনের ক্রম হল→ border-bottom-width>border-bottom-style> border-bottom-color.কোন ভ্যালু বাদ পড়লে সমস্যা নেই।


```

1  <html>
2  <head>
3  <style type="text/css">
4  p
5  {
6  border-style:solid;
7  border-bottom:thick dotted #ff0000;
8  }
9  </style>
10 </head>
11 <body>
12 <p>This is some text in a paragraph.</p>
13 </body>
14 </html>

```

3.border-bottom-color:

border-bottom-color property কোন এলিমেন্টের bottom border-এর কালার নির্ধারণ করে।

```

<style type="text/css">
p
{
border-style:solid;
border-bottom-color:#ff0000;
}
</style>

```

4.border-bottom-style property:

border-bottom-style property কোন এলিমেন্টের বর্ডারের bottom অংশের স্টাইল নির্ধারণ করে।

```

3 <style type="text/css">
4   p {border-style:solid;}
5   p.none {border-bottom-style:none;}
6   p.dotted {border-bottom-style:dotted;}
7   p.dashed {border-bottom-style:dashed;}
8   p.solid {border-bottom-style:solid;}
9   p.double {border-bottom-style:double;}
10  p.groove {border-bottom-style:groove;}
11  p.ridge {border-bottom-style:ridge;}
12  p.inset {border-bottom-style:inset;}
13  p.outset {border-bottom-style:outset;}
14 </style>

```

border-bottom-style property যে সকল মান ব্যবহার করতে পারেঃ

none→hidden→dotted→dashed→solid→double→groove→ridge→inset→outset→inherit

5.border-bottom-width property:

border-bottom-width property কোন এলিমেন্টের বর্ডারের bottom অংশের প্রশস্ততা নির্ধারণ করে।

```

3 <style type="text/css">
4   p
5   {
6     border-style:solid;
7     border-bottom-width:15px;
8   }
9 </style>

```

বিঃ দ্রঃ border-bottom-width প্রপার্টি ডিকলার করার আগে border-style প্রপার্টি ডিকলার করতে হবে।

border-bottom-width প্রপার্টিতে যে সকল মান ব্যবহার করা যায়- thin→medium→thick অথবা সরাসরি কোন সংখ্যা ব্যবহার করতে পারেন।

6.border-color property:

border-color property ব্যবহার করে বর্ডারের রঙ নির্ধারণ করা হয়। এর মান HTML এ ব্যবহৃত তিনটি রঙের নাম ব্যবহারের পদ্ধতি অনুসারে হবে। "border-color" property এককভাবে কাজ করতে পারে না, এর সাথে "border-style" property ব্যবহার করতে হয়, যা প্রথমে ডিকলার করতে হয়। You can also

set the border color to "transparent" Specifies that the border color should be transparent. This is default.

কোন এলিমেন্টের border-width চারদিকেই হতে পারে। যেমনঃ

- **border-color:red green blue pink;**
 - top border is red
 - right border is green
 - bottom border is blue
 - left border is pink

- **border-color:red green blue;**
 - top border is red
 - right and left borders are green
 - bottom border is blue

- **border-color:dotted red green;**
 - top and bottom borders are red
 - right and left borders are green

- **border-color:red;**
 - all four borders are red

```

<style type="text/css">
p.one
{
border-style:solid;
border-color:#0000ff;
}
p.two
{
border-style:solid;
border-color:#ff0000 #0000ff;
}
p.three
{
border-style:solid;
border-color:#ff0000 #00ff00 #0000ff;
}
p.four
{
border-style:solid;
border-color:#ff0000 #00ff00 #0000ff rgb(250,0,255);
}
</style>

```

7.border-left property:

border-left shorthand property সকল left border properties কে একটি ডিকলারেশনের মাধ্যমে প্রকাশ করে। ডিকলারেশনের ক্রম হল→ border-left-width→border-left-style→border-left-color.

```

<style type="text/css">
p
{
border-style:solid;
border-left:thick double #ff0000;
}
</style>

```

8.border-left-color property:

border-left-color property বামদিকের বর্ডারের রঙ নির্ধারণ করে।

```

3 <style type="text/css">
4   p
5   {
6     border-style:solid;
7     border-left-color:#ff0000;
8   }
9 -</style>

```

9.border-left-style property:

border-left-style property বামদিকের বর্ডারের স্টাইল নির্ধারণ করে।

```

3 <style type="text/css">
4   p
5   {
6     border-style:solid;
7   }
8   p.none {border-left-style:none;}
9   p.dotted {border-left-style:dotted;}
10  p.dashed {border-left-style:dashed;}
11  p.solid {border-left-style:solid;}
12  p.double {border-left-style:double;}
13  p.groove {border-left-style:groove;}
14  p.ridge {border-left-style:ridge;}
15  p.inset {border-left-style:inset;}
16  p.outset {border-left-style:outset;}
17 -</style>

```

border-bottom-style property যে সকল মান ব্যবহার করতে পারে **border-left-style property**-ও সেই সকল মান ব্যবহার করতে পারবে।

10.border-left-width property:

এর সকল কাজ border-bottom-width property এর মতোই।

```

3 <style type="text/css">
4   p
5   {
6     border-style:solid;
7     border-left-width:15px;
8   }
9 -</style>

```

11.border-right property:

এর সকল কাজ [border-left property](#)-এর মতোই।

12.border-right-color property:

এর সকল কাজ [border-left-color property](#)-এর মতোই।

13.border-right-style property:

এর সকল কাজ [border-bottom-style property](#) ও [border-left-style property](#) এর মত।

14.border-right-width property:

এর সকল কাজ [border-bottom-width property](#) ও [border-left-width property](#) এর মতোই।

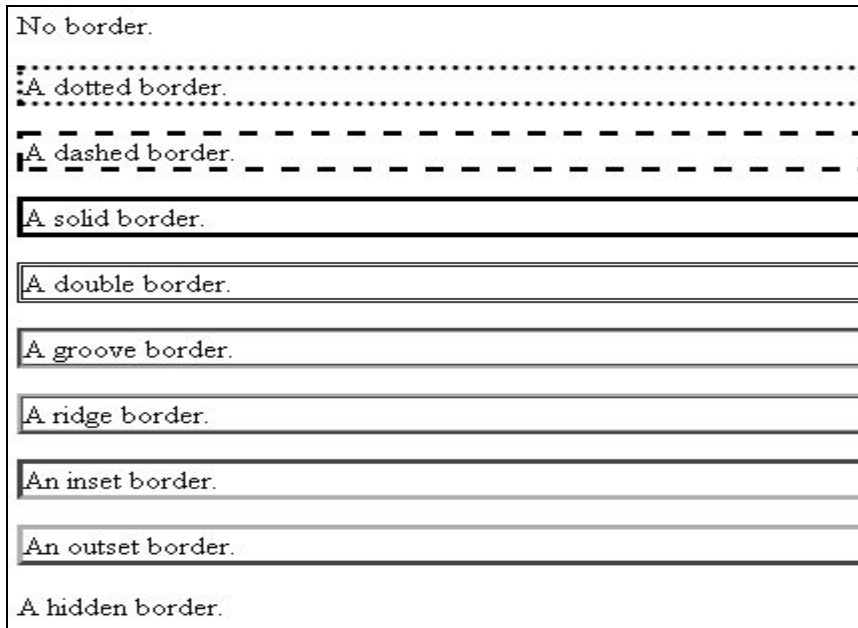
15.border-style Properties:

বর্ডারের স্টাইল কেমন হবে তা [border-style](#):-এর মাধ্যমে প্রকাশ করা হয়। যে ভেলু গুলো ব্যবহার করা যায় তা নিম্নরূপঃ

ভেলু	বর্ণনা
none	কোন বর্ডার দেখা যাবে না।
hidden	The same as "none", except in border conflict resolution for table elements
dotted	একটি dotted border দেখা যাবে।
dashed	একটি dashed border দেখা যাবে।
solid	একটি solid border দেখা যাবে।
double	একটি double border দেখা যাবে।
groove	3D grooved border দেখা যাবে। যার ইফেক্ট border-color-এর উপর নির্ভর করে।
ridge	3D ridged border দেখা যাবে। যার ইফেক্ট border-color-এর উপর নির্ভর করে।
inset	3D inset border দেখা যাবে। যার ইফেক্ট border-color-এর উপর নির্ভর করে।
outset	3D outset border দেখা যাবে। যার ইফেক্ট border-color-এর উপর নির্ভর করে।

নিম্নে উদাহরণসহ দেখানো হলঃ

```
1 <html>
2 <head>
3 <style type="text/css">
4   p.none {border-style:none;}
5   p.dotted {border-style:dotted;}
6   p.dashed {border-style:dashed;}
7   p.solid {border-style:solid;}
8   p.double {border-style:double;}
9   p.groove {border-style:groove;}
10  p.ridge {border-style:ridge;}
11  p.inset {border-style:inset;}
12  p.outset {border-style:outset;}
13  p.hidden {border-style:hidden;}
14 </style>
15 </head>
16
17 <body>
18 <p class="none">No border.</p>
19 <p class="dotted">A dotted border.</p>
20 <p class="dashed">A dashed border.</p>
21 <p class="solid">A solid border.</p>
22 <p class="double">A double border.</p>
23 <p class="groove">A groove border.</p>
24 <p class="ridge">A ridge border.</p>
25 <p class="inset">An inset border.</p>
26 <p class="outset">An outset border.</p>
27 <p class="hidden">A hidden border.</p>
28 </body>
29
30 </html>
```



চারদিকের বর্ডারের জন্য চারটি মান সেট করা যেতে পারেঃ

- **border-style:dotted solid double dashed;**
 - top border is dotted
 - right border is solid
 - bottom border is double
 - left border is dashed

- **border-style:dotted solid double;**
 - top border is dotted
 - right and left borders are solid
 - bottom border is double

- **border-style:dotted solid;**
 - top and bottom borders are dotted
 - right and left borders are solid

- **border-style:dotted;**
 - all four borders are dotted

নিচের উদাহরণ দেখুনঃ

```

1 <html>
2 <head>
3 <style type="text/css">
4   p.one {border-style:dotted solid dashed double;}
5   p.two {border-style:dotted solid dashed;}
6   p.three {border-style:dotted solid;}
7   p.four {border-style:dotted;}
8 </style>
9 </head>
10
11 <body>
12   <p class="one">This is some text in a paragraph.</p>
13   <p class="two">This is some text in a paragraph.</p>
14   <p class="three">This is some text in a paragraph.</p>
15   <p class="four">This is some text in a paragraph.</p>
16 </body>
17 </html>

```

Output:

The output shows four paragraphs of text, each enclosed in a rectangular border with a different style:

- Paragraph 1: Dotted border (p.one)
- Paragraph 2: Dashed border (p.two)
- Paragraph 3: Solid border (p.three)
- Paragraph 4: Dotted border (p.four)

16.border-top property:

The border-top shorthand property sets all the top border properties in one declaration.ক্রম হলঃ border-top-width→ border-top-style→ border-top-color.

```

3 <style type="text/css">
4   p
5   {
6     border-style:solid;
7     border-top:thick double #ff0000;
8   }
9 </style>

```

17.border-top-color property:

এর কাজ [border-left-color](#) property: এর মতোই।

18.border-top-style property:

এর কাজ border-bottom-style property ও border-left-style property-এর মত।

19.border-top-width property:

এর সকল কাজ [border-bottom-width](#) property এর মত।

20.border-width property:

বর্ডারের প্রশস্ততা নির্ধারণ করতে border-width property ব্যবহার করা হয়। "border-width" property এককভাবে কাজ করতে পারে না, এর সাথে "border-style" property ব্যবহার করতে হয়, যা প্রথমে ডিক্লার করতে হয়। এর মান দুইভাবে ডিফাইন করা যায়-১.পিক্সেল ২. pre-defined values । pre-defined value গুলো হলঃ

ভ্যালু	বর্ণনা
thin	thin border নির্দেশ করে।
medium	medium border নির্দেশ করে। এটা default
thick	thick border নির্দেশ করে।
length	ইচ্ছামত মান দেওয়া যেতে পারে।

কোন এলিমেন্টের border-width চারদিকেই হতে পারে। যেমনঃ

- **border-width: thin medium thick 10px;**
 - top border is thin
 - right border is medium
 - bottom border is thick
 - left border is 10px

- **border-width: thin medium thick;**
 - top border is thin
 - right and left borders are medium
 - bottom border is thick

- **border-width: thin medium;**
 - top and bottom borders are thin
 - right and left borders are medium

- **border-width: thin;**
 - all four borders are thin

```

<style type="text/css">
p.one
{
border-style:solid;
border-width:5px;
}
p.two
{
border-style:solid;
border-width:medium;
}
p.three
{
border-style:solid;
border-width:1px;
}
</style>

```

৩. সিএসএস প্যাডিং প্রপার্টিঃ

সিএসএস প্যাডিং বলতে বোঝায় এলিমেন্ট এর **border** এবং তার **content** এর মাঝের অংশকে। সিএসএস প্যাডিং ব্যবহার করে আপনি এইচটিএমএল এলিমেন্ট (paragraphs, tables ইত্যাদি) এর **default** প্যাডিং পরিবর্তন করতে পারেন। **Padding** এর উপর **background color** এর প্রভাব রয়েছে।

সকল CSS Padding Properties

প্রপার্টি	বর্ণনা
1. padding	padding shorthand property(padding:) সকল padding properties কে একটি ডিক্লারেশনের মাধ্যমে তুলে ধরে।
2. padding-bottom	কোন এলিমেন্টের বটমের প্যাডিং নির্ধারণ করে।
3. padding-left	কোন এলিমেন্টের বাম পাশের প্যাডিং নির্ধারণ করে।
4. padding-right	কোন এলিমেন্টের ডান পাশের প্যাডিং নির্ধারণ করে।
5. padding-top	কোন এলিমেন্টের টপের প্যাডিং নির্ধারণ করে।

সকল **CSS Padding Properties** গুলো যে ভাষা ব্যবহার করতে পারে-

<i>length</i>	<i>px, pt, em</i> এককে <i>fixed padding</i> ব্যবহার করা যায়।
<i>%</i>	এলিমেন্টের সাপেক্ষে শতকরায় প্যাডিং প্রকাশ করা হয়।

ধারাবাহিকভাবে **CSS Padding Properties** গুলো নিম্নে বর্ণনা করা হলঃ

1. padding shorthand property:

padding shorthand property(padding:) সকল padding properties কে একটি ডিক্লারেশনের মাধ্যমে তুলে ধরে। padding shorthand property চারটি ভেলু ব্যবহার করে। যেমনঃ

- **padding:10px 5px 15px 20px;** [সঠিক ক্রমে সাজানো]
 - top padding is 10px
 - right padding is 5px
 - bottom padding is 15px
 - left padding is 20px

- **padding:10px 5px 15px;**
 - top padding is 10px
 - right and left padding are 5px
 - bottom padding is 15px
- **padding:10px 5px;**
 - top and bottom padding are 10px
 - right and left padding are 5px
- **padding:10px;**
 - all four paddings are 10px

```

3 <style type="text/css">
4   p.ex1 {padding:2cm;}
5   p.ex2 {padding:0.5cm 3cm;}
6 </style>

```

2.CSS padding-bottom Property:

The padding-bottom property sets the bottom padding (space) of an element.

```

3 <style type="text/css">
4   p.padding {padding-bottom:2cm;}
5   p.padding2 {padding-bottom:50%;}
6 </style>

```

3.CSS padding-left Property:

The padding-left property sets the left padding (space) of an element.

```

3 <style type="text/css">
4   p.padding {padding-left:2cm;}
5   p.padding2 {padding-left:50%;}
6 </style>

```

4.CSS padding-right Property

The padding-right property sets the right padding (space) of an element.

```

3 <style type="text/css">
4   p.padding {padding-right:2cm;}
5   p.padding2 {padding-right:50%;}
6 </style>

```

5.CSS padding-top Property

The padding-top property sets the top padding (space) of an element.

```

3 <style type="text/css">
4   p.padding {padding-top:2cm;}
5   p.padding2 {padding-top:50%;}
6 </style>

```

8. সিএসএস আউটলাইন প্রপার্টিঃ

কোন এলিমেন্টের বর্ডারের চারপাশে আরেকটি বর্ডারই হল আউটলাইন।

সিএসএস-এ ব্যবহৃত সকল Outline Properties:

প্রপার্টি	বর্ণনা	ভেলু
1.outline	সকল আউটলাইন প্রপার্টিকে একটি ডিকলারেশনের মাধ্যমে তুলে ধরে।	<i>outline-color outline-style outline-width</i>
2.outline-color	আউটলাইনের কালার সেট করে।	<i>color_name</i>

		<i>hex_number</i> <i>rgb_number</i> <i>invert</i>
3.outline-style	আউটলাইনের স্টাইল নির্ধারণ করে।	<i>none</i> <i>dotted</i> <i>dashed</i> <i>solid</i> <i>double</i> <i>groove</i> <i>ridge</i> <i>inset</i> <i>outset</i>
4.outline-width	আউটলাইনের প্রসঙ্গতা নির্ধারণ করে।	<i>Thin/medium</i> <i>thick/length</i>

ধারাবাহিকভাবে নিম্নে এদের বর্ণনা দেওয়া হলঃ

১.আউটলাইন স্টাইল প্রপার্টিঃ

সকল আউটলাইন প্রপার্টিকে একটি ডিকলারেশনের মাধ্যমে তুলে ধরে। এদের ক্রম হল **outline-color**→**outline-style**→ **outline-width**.

```

<head>
<style type="text/css">
  p
  {
    border:1px solid red;
    outline:green dotted thick;
  }
</style>
</head>

```

২.আউটলাইন কালার প্রপার্টিঃ

আউটলাইনের কালার সেট করে। **outline-style** property কে **outline-color** property -এর আগে ডিকলার করতে হবে।

color	আউটলাইনের কালার নির্ধারণ করে।
invert	বিপরীত কালার প্রদর্শন করে। এটা নিশ্চিত করে ব্যাকগ্রাউন্ড কালার ছড়াই আউটলাইন দেখা যাবে। এটা ডিফল্ট।

```

4 <style type="text/css">
5   p
6   {
7     border:1px solid red;
8     outline-style:dotted;
9     outline-color:#00ff00;
10  }
11 </style>

```

৩. আউটলাইন স্টাইল প্রপার্টিঃ

outline-style property আউটলাইনের স্টাইল নির্ধারণ করে।

outline-style property -তে ব্যবহৃত মানসমূহঃ

মান	বর্ণনা
none	কোন আউটলাইন হবে না
dotted	ডটেড আউটলাইন হবে
dashed	ড্যাশড আউটলাইন হবে
solid	সলিড আউটলাইন হবে
double	ডাবল আউটলাইন হবে
groove	3D grooved আউটলাইন হবে
ridge	3D ridged আউটলাইন হবে
inset	3D inset আউটলাইন হবে
outset	3D outset আউটলাইন হবে

একটি উদাহরণ দেখুন-

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

```

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
p {border:1px solid red;}
```



```
p.dotted {outline-style:dotted;}
```

```
p.dashed {outline-style:dashed;}
```

```
p.solid {outline-style:solid;}
```

```
p.double {outline-style:double;}
```

```
p.groove {outline-style:groove;}
```

```
p.ridge {outline-style:ridge;}
```

```
p.inset {outline-style:inset;}
```

```
p.outset {outline-style:outset;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p class="dotted">A dotted outline</p>
```

```
<p class="dashed">A dashed outline</p>
```

```
<p class="solid">A solid outline</p>
```

```
<p class="double">A double outline</p>
```

```
<p class="groove">A groove outline</p>
```

```
<p class="ridge">A ridge outline</p>
```

```
<p class="inset">An inset outline</p>
```

```
<p class="outset">An outset outline</p>
```

specified.

</body>

</html>

আউটপুটঃ



Note: IE8 supports the outline properties only if a !DOCTYPE is specified.

8. সিএসএস আউটলাইন উইথ প্রপার্টিঃ

আউটলাইনের প্রসঙ্গতা নির্ধারণ করে। `outline-style property` কে `outline-width property` -এর আগে ডিকলার করতে হবে।

`outline-width property`-তে ব্যবহৃত মান সমূহ-

মান	বর্ণনা
thin	Specifies a thin outline
medium	Specifies a medium outline. This is default
thick	Specifies a thick outline
<i>length</i>	Allows you to define the thickness of the outline

This example demonstrates how to set the width of an outline.

```
<style type="text/css">
p.one
{
border:1px solid red;
outline-style:solid;
outline-width:thin;
}
p.two
{
border:1px solid red;
outline-style:dotted;
outline-width:3px;
}
</style>
```

অধ্যায়-দশ

[টেবিলে সিএসএস-এর ব্যবহার]

সিএসএস-এর মাধ্যমে টেবিলকে কীভাবে আকর্ষণীয় করা যায় আমরা এখানে সেটাই আলোচনা করব-

টেবিল বঁড়ার:

টেবিলের বঁড়ারের স্টাইল নির্ণয় করতে *border property* ব্যবহার করা হয়। যেমন-

```
1 <html>
2 <head>
3 <style type="text/css">
4   table,th,td
5   {
6     border:1px solid black;
7   }
8 </style>
9 </head>
10
11 <body>
12 <table>
```

```
13 <tr>
14   <th>Firstname</th>
15   <th>Lastname</th>
16 </tr>
17 <tr>
18   <td>Peter</td>
19   <td>Griffin</td>
20 </tr>
21 <tr>
22   <td>Lois</td>
23   <td>Griffin</td>
24 </tr>
25 </table>
26 </body>
27 </html>
```

Firstname	Lastname
Peter	Griffin
Lois	Griffin

টেবিলে লক্ষ্য করুন এখানে ডাবল বর্ডার আছে, কারণ `table`, `th`, এবং `td` এলিমেন্টের `separate border` আছে। একক বর্ডার দেখাতে `border-collapse` property ব্যবহার করুন।

কলাপ্স বর্ডার:

```
1 <html>
2 <head>
3 <style type="text/css">
4   table
5   {
6     border-collapse: collapse;
7   }
8   table, td, th
9   {
10    border: 1px solid black;
11  }
12 </style>
13 </head>
14
15 <body>
16 <table>
```

```
14 |
15 | <body>
16 | <table>
17 | <tr>
18 |   <th>Firstname</th>
19 |   <th>Lastname</th>
20 | </tr>
21 | <tr>
22 |   <td>Peter</td>
23 |   <td>Griffin</td>
24 | </tr>
25 | <tr>
26 |   <td>Lois</td>
27 |   <td>Griffin</td>
28 | </tr>
29 | </table>
30 |
31 | </body>
32 | </html>
```

Firstname	Lastname
Peter	Griffin
Lois	Griffin

অধ্যায়- এগার

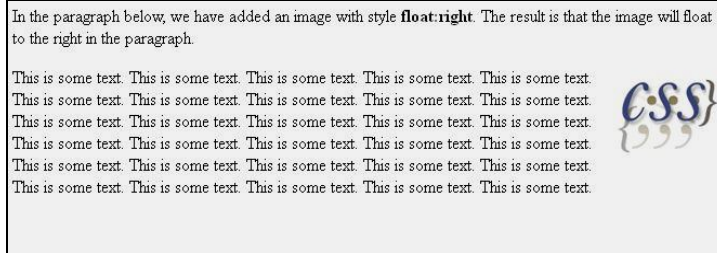
[সিএসএস ফ্লোট ও ক্লিয়ার প্রপার্টি]

সিএসএস ফ্লোট প্রপার্টি:

HTML-এ **align attribute**- এর মতোই কাজ করে ফ্লোট প্রপার্টি। ফ্লোট দ্বারা ছবি কে ওয়েব পেজের এক পাশে বা অন্য পাশে অবস্থান করানো যায় এবং এর সাহায্যে খুব সহজে টেক্সট কে ছবির চারপাশে **Wrapping** করা যায়। যেমনটি দেখা যায় **magazines** বা সংবাদপত্রে বিভিন্ন আকারের ছবির চারপাশে টেক্সট এর অবস্থান। ফ্লোট প্রপার্টির চারটি মান ব্যবহার করা যায়। যথা- **left, right, none, inherit**. এগুলোর মাধ্যমে উপাদানের অবস্থান নির্দেশ করা হয়।

```
1  <html>
2  <head>
3  <style type="text/css">
4    img
5    {
6    float:right;
7    }
8  </style>
9  </head>
10
11 <body>
12 <p>In the paragraph below, we have added an image with style <b>float:right</b>.
13 The result is that the image will float to the right in the paragraph.</p>
14 <p>
15   
16   This is some text. This is some text. This is some text.
17   This is some text. This is some text. This is some text.
18   This is some text. This is some text. This is some text.
19   This is some text. This is some text. This is some text.
20   This is some text. This is some text. This is some text.
21   This is some text. This is some text. This is some text.
22   This is some text. This is some text. This is some text.
23   This is some text. This is some text. This is some text.
24   This is some text. This is some text. This is some text.
25   This is some text. This is some text. This is some text.
26 </p>
27 </body>
28
29 </html>
```

Output:



দেখুন ইমেজটি টেক্সটের ডানপাশে অবস্থান করছে।

সিএসএস ক্লিয়ার প্রপার্টি:

Clear প্রপার্টির মাধ্যমে নির্দেশ করা যেতে পারে সেই উপাদানের পাশে অন্য কোন উপাদান **float** করতে পারবে কি না। এটি কেবল ব্লক লেভেল উপাদানের ক্ষেত্রেই ব্যবহার করা যাবে। এর মান হিসেবে **left, right, none** ও **both** ব্যবহৃত হতে পারে। যেমন-

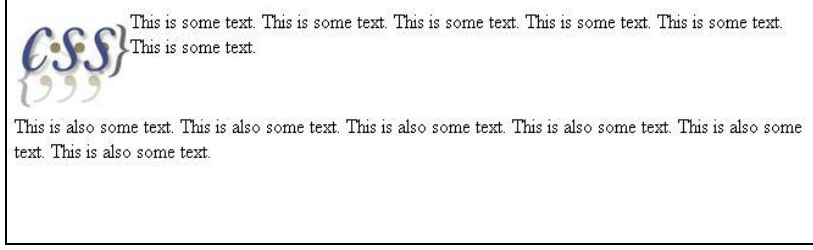
P{clear: both}→ প্যারাগ্রাফের কোন পাশেই ফ্লোট হতে পারবে না।

P{clear: left}→ কেবল বা পাশে ফ্লোট হতে পারবে না।

P{clear: right}→ কেবল ডানপাশে ফ্লোট হতে পারবে না।

P{clear: none}→ Default. Allows floating elements on both sides

```
1  <html>
2  <head>
3  <style type="text/css">
4      img
5      {
6          float:left;
7      }
8      p.clear
9      {
10         clear:both;
11     }
12 </style>
13 </head>
14 <body>
15 
16 <p>This is some text. This is some text.
17 This is some text.</p>
18 <p class="clear">This is also some text.
19 This is also some text.</p>
20 </body>
21 </html>
22
```

অধ্যায়-বার

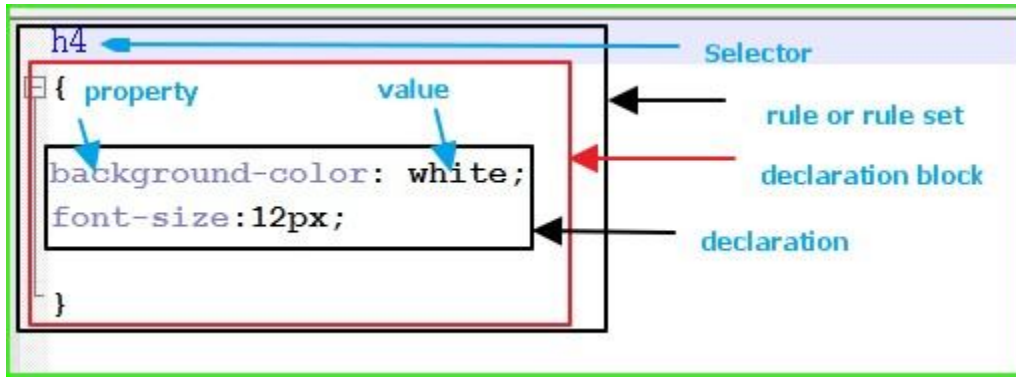
[সিএসএস সিলেক্টর]

সিএসএস সিলেক্টর কে সিএসএস এর হৃদপিণ্ড বলা হয়। বিভিন্ন ধরনের সিলেক্টর আছে, এসব দিয়ে এইচটিএমএল পেজের এলিমেন্ট সিলেক্ট করা যায় এবং সিলেক্ট করার পর নিজের ইচ্ছেমত স্টাইলিং করা যায়। সিএসএস সিলেক্টর এর ব্যাপারে খুব গভীর এবং স্বচ্ছ ধারণা থাকতে হবে। স্টাইলশীটে **<BODY>** অংশে ব্যবহৃত উৎসসমূহের সাথে নতুন সিলেক্টর যোগ করা যেতে পারে **Class** ও **ID** ব্যবহার করে। সিলেক্টর নিয়ে কাজ করার আগে কিছু টার্ম জানতে হবে যেমনঃ

১. এইচটিএমএল এলিমেন্ট ২. এট্রিবিউট → এই দুটি সম্পর্কে আমরা জানি।

৩. রুল সেট বা রুলঃ

রুল সেটের মধ্য প্রপাটি ও তার মান দেয়া হয়। এটি থাকে কোন ডিকলারেশন ব্লকের মধ্য। ডিকলারেশন ব্লক শুরু হয় সেকেন্ড ব্যাকেট { দিয়ে ও শেষ হয় } দিয়ে। এর মাঝে একাধিক ডিকলারেশন থাকতে পারে যা সেমিকোলন দিয়ে পৃথক থাকতে পারে। ডিকলারেশন ব্লকের আগে অর্থাৎ { এর আগে যা কিছু থাকে তাই সিলেক্টর। আপনি যে **html** উপাদানের প্রপাটি পরিবর্তন করতে চান সেটিকে সিলেক্টর হিসেবে ব্যবহার করতে হবে। এরপর ডিকলারেশন ব্লকের মাঝে যে নিয়ম দেবেন তা প্রয়োগ হবে ঐ সিলেক্টরের ক্ষেত্রে। যেমন নিচের উদাহরণে **h1** হল সিলেক্টর , আর { } এর মাঝে রুল সেট। পুরো { } ডিকলারেশন ব্লক।



```
h4{background-color:white;font-size:12px;}
p{color:red;}
```

এখানে `h4`, `p` এসব হচ্ছে সিলেক্টর, `h4` এর দ্বারা এইচটিএমএল পেজের `<h4></h4>` এর ভিতরে যা আছে তা সিলেক্ট করবে বা বলা যায় `h4` এলিমেন্ট সিলেক্ট করবে। `h4` এর পর সেকেন্ড ব্রাকেট যেখান থেকে শুরু হয়েছে এবং `p` এর আগে যেখানে সেকেন্ড ব্রাকেট শেষ হয়েছে এইটুকু ডিক্লেয়ারেশন ব্লক। সেকেন্ড ব্রাকেটের ভিতরে সবটুকু হচ্ছে ডিক্লেয়ারেশন। `background-color` হচ্ছে প্রোপার্টি এবং `white` হচ্ছে এর ভ্যালু। আর পুরোটা অর্থাৎ `h4` থেকে শুরু করে এর সেকেন্ড ব্রাকেট যেখানে শেষ হয়েছে এইটুকু রুল বা রুল সেট। উপরের কোডে দুটি রুল আছে `h4` এবং `p`। নিচে বিভিন্ন সিলেক্টর এর নাম এবং এরা কিভাবে কাজ করে তা দেয়া হল:

১. ইউনিভার্সাল সিলেক্টরঃ

ইউনিভার্সাল সিলেক্টর এইচটিএমএল পেজের প্রতিটি এলিমেন্ট কে সিলেক্ট করে। যেমন নিচের কোডটি সব এইচটিএমএল এলিমেন্ট এর রং লাল করে দেবে।

```
*{color:red;}
```

২. আইডি সিলেক্টরঃ

HTML element-এর জন্য একটি স্টাইল নির্ধারণ করতে CSS আপনাকে আপনার নিজস্ব **selector** ব্যবহারের সুযোগ দিয়েছে যা "id" এবং "class" নামে পরিচিত। ID সিলেক্টর একটা এলিমেন্টের স্টাইল দিতে ব্যবহৃত হয়। ID সিলেক্টর HTML এলিমেন্টের id attribute ব্যবহার করে এবং এটা একটা "#" -এর মাধ্যমে ডিফাইন করা হয়। একটি ID নাম নাম্বার দিয়ে শুরু করা যাবে না এটা Firefox.-এ কাজ করবে না।

```

7 <style type="text/css">
8   p#example1 { background-color: orange; }
9   p#example2 { text-transform: uppercase; }
10 </style>
11 </head>
12 <body>
13 <p id="ExampleID1">This paragraph has an ID name of
14   "example1" and has a orange CSS defined background</p>
15 <p id="ExampleID2">This paragraph has an ID name of
16   "example2" and has had its text transformed to uppercase letters. </p>
17 </body>

```

৩. সিএসএস ক্লাস সিলেক্টরঃ

সিএসএস এর মাধ্যমে এইচটিএমএল এলিমেন্ট এর অসংখ্য স্টাইল দেয়া যায়। মূলত a group of elements-এর style দিতে সিএসএস class selector ব্যবহার করা হয়। class selector ব্যবহার করে HTML elements-এর জন্য কতোগুলো particular style নির্ধারণ করা যায় same class-এর মাধ্যমে। class selector ব্যবহার করে HTML-এর class attribute এবং এটা একটি ডট(".")-এর মাধ্যমে ডিফাইন করা হয়। Class ব্যবহার করা খুব সাধারণ। সাধারণ সিএসএস কোড এর অংশকে বর্ধিত (extension) করতে হবে। আর এই বর্ধিত অংশকে অবশ্যই এইচটিএমএল ট্যাগ এ Class এর মান হিসাবে বসাতে হবে। Class Selector এর সাধারণ গঠনঃ

Selector/HTML Element . Class Name {Declaration}

```

6 <style type="text/css">
7   p.first{ color: blue; }
8   p.second{ color: red; }
9   span.highlighted {color: red;}
10 </style>
11 <html>
12 <body>
13 <p>This is a normal paragraph.</p>
14 <p class="first">This is a paragraph that uses the p.first CSS code!</p>
15 <p class="second">This is a paragraph that uses the p.second CSS code!</p>
16 <span class=" highlighted"> Span Text</span>
17 </body>
18 </html>

```

নোটঃ class নাম সংখ্যা দিয়ে শুরু করবেন না তাহলে এটা ইন্টারনেট এক্সপ্লোরার এ কাজ করবেনা।

৪.টাইপ সিলেক্টর.

```
1  p{
2  background-color:orange;
3  }
4  a{
5  color:maroon;
6  }
```

p, a হচ্ছে এখানে টাইপ সিলেক্ট। p সিলেক্টর এইচটিএমএল পেজের সব <p> এলিমেন্ট কে সিলেক্ট করবে এবং এদের ব্যাকগ্রাউন্ড রং কমলা করে দেবে আর a সিলেক্টর এইচটিএমএল সব লিংক কে সিলেক্ট করে রং খয়েরি করে দেবে।

৫.ডিসেনডেন্ট সিলেক্টরঃ

এক সিলেক্টরের মাঝে আরেক সিলেক্টর থাকলে দ্বিতীয় সিলেক্টরের কোন প্রপাটি পরিবর্তন করতে চাইলে ডিসেনডেন্ট সিলেক্টর ব্যবহার করতে হয়। যেমনঃ

```
li a {
text-decoration: none;
color:red;
}
```

এইচটিএমএল ফাইল

```
1  <div id="menu">
2  <ul>
3  <li><a href="#" title="">html</a></li>
4  <li><a href="#" title="">css</a>
5  <ul>
6  <li> Child </li>
7  </ul>
8  </li>
9  <li><a href="#" title="">javascript</a></li>
10 <li><a href="#" title="">php</a></li>
11 <li><a href="#" title="">MySql</a></li>
12 </ul>
13 </div>
```

ধরুন আপনি `li` এলিমেন্ট এর অধীনস্থ `a` এলিমেন্ট কে টার্গেট করতে চাচ্ছেন তখন সিএসএস রুল লিখতে হবে উপরের মত। এই কোডের প্রভাব আপনার এইচটিএমএল পেজের শুধু `li` এর অধীনস্থ `a` তে গিয়ে পরবে। এ রকম ডিকলারেশনের ক্ষেত্রে সিলেক্টর দুটোর মাঝে শুধু একটি স্পেস থাকবে, কমা , সেমিকোলন বা ফুলস্টপ থাকবে না।

৬. direct children সিলেক্টরঃ

```
div#menu> ul { border: 1px solid black; }
```

এই রুলটি যদি উপরের এইচটিএমএল পেজে প্রয়োগ করেন তাহলে `ul` এলিমেন্ট এর উপর একটা বঁডার হবে তবে `Children` লেখাটির উপর বঁডার আসবেনা যদিও এটা আরেকটা `ul` এর মধ্যে আছে। কারন `Children` লেখাটি যে `ul` এর মধ্যে আছে সেটা `menu` আইডির `direct children` নয়। আরেকটি উদাহরণঃ `div ul>li>p`

এর অর্থ হল `div` এর অধীনে `ul` এবং তার চাইল্ড `li`। আবার `li` এর ডিসেন্ডেন্ট হবে `p`।

৭. গ্রুপিং সিলেক্টরঃ

স্টাইলশীটে ব্যবহৃত কোডের পরিমাণ কমাতে স্টাইলের সিলেক্টর এবং তৎসংক্রান্ত নির্দেশসমূহ একটা গ্রুপরূপে উল্লেখ করা যায়। যদি একাধিক এলিমেন্টের একই স্টাইল করতে চান তখন গ্রুপিং সিলেক্টর ব্যবহার করবেন। ধরুন আপনি চাচ্ছেন আপনার পেজের সব হেডিং এবং লিংক এর রং একই হবে তখন নিচের মত করে লিখতে পারেন।

```
h1,h2,h3,h4,a{color:red;}
```

উপরোক্ত কোডকে আমরা সাধারনভাবে লিখতাম-

```
<style type="text/css">
```

```
H1{color:red;}
```

```
H2{color:red;}
```

```
H3{color:red;}
```

```
H4{color:red;}
```

```
a{color:red;}
```

```
</style>
```

ডিকলারেশনসমূহ গ্রুপ করাঃ

```
<style type="text/css">
```

```
H1{color : red}
```

```
H1{font-family : Arial, Arial black}
```

H1{font-size : 16pt}

</style>

এখনে h1 সিলেক্টরের জন্য তিনটি ডিকলারেশন দেয়া হয়েছে। এ তিন ডিকলারেশনকে গ্রুপ হিসেবে প্রকাশ করা যেতে পারে এভাবে-

<style type="text/css">

H1{color : red; font-family : Arial, Arial black; font-size : 16pt }

</style>

বিঃদ্রঃ সিলেক্টরের গ্রুপ গঠনের সময় সিলেক্টরসমূহকে পৃথক করা হয় কমা(,) দিয়ে এবং ডিকলারেশন নিয়ে গ্রুপ গঠনের সময় পৃথক করা হয় সেমিকোলন (;) দিয়ে।

অধ্যায়-তের

সিএসএস সিউডো ক্লাস ও সিউডো এলিমেন্ট

সিএসএস সিউডো ক্লাসঃ

সিএসএস সিউডো ক্লাস ব্যবহার করে বিভিন্ন উপাদানের অবস্থান, অবস্থা ইত্যাদি সিলেক্টর দ্বারা সিলেক্ট করে স্পেশাল ইফেক্ট দেওয়া হয়। ক্লাস সিলেক্টরের মত সিউডো ক্লাস ব্যবহার করে ডকুমেন্টের বিভিন্ন এলিমেন্ট সিলেক্ট করা যায়। নেভিগেশন বারের বিভিন্ন মেনু বিভিন্ন অবস্থায় কেমন দেখাবে তা সিউডো ক্লাস দ্বারা নির্ধারণ করা হয়। এছাড়া বিভিন্ন চাইল্ড সিলেক্টরের ফাণ্ট চাইল্ড সিলেক্ট করা সহ আরো অনেক ধরনের কাজ সিউডো ক্লাস দ্বারা করা যায়।

সিএসএস সিউডো ক্লাসের সাধারণ গঠন-

Selector: pseudo class {property: value;}

সিএসএস ক্লাসকে সিউডো ক্লাসের সাথে ব্যবহার করা যায়। যেমন-

Selector. Class: pseudo class {property: value;}

a.red:visited{color:#FF0000;}

CSS Syntax

সিউডো ক্লাসের প্রকারভেদঃ

- ১.এংকর সিউডো ক্লাস (Anchor Pseudo class)
- ২.ফাস্ট চাইল্ড সিউডো ক্লাস (First child Pseudo-class)
- ৩.ল্যাঙ্গুয়েজ সিউডো ক্লাস (lang Pseudo-class)
- ৪.ফোকাস সিউডো ক্লাস(focus Pseudo-class)

বিভিন্ন সিউডো ক্লাসের বর্ণনা নিম্নে দেওয়া হল-

১.এংকর সিউডো ক্লাসঃ লিংক তৈরির জন্য আমরা সাধারণত এংকর ট্যাগ(<a>) ব্যবহার করে থাকি এবং সিএসএস ব্যবহার করে স্টাইল নির্ধারণ করা হয়। কিন্তু আমরা যদি চাই বিভিন্ন অবস্থার উপর নির্ভর করে বিভিন্ন ইফেক্ট (যেমনঃ লিংকের উপর মাউস নিয়ে গেলে একধরনের ইফেক্ট, লিংক ভিজিট করলে আরেক ধরনের ইফেক্ট ইত্যাদি) দেখাতে চাই তবে সিএসএস এংকর সিউডো ক্লাস ব্যবহার করতে হবে। সিএসএস-এ মোট চারটি এংকর সিউডো ক্লাস রয়েছে-

• a:link → একটি সাধারণ লিংক যা ভিজিট করা হয় নি।
• a:visited → একটি লিংক যা পূর্বে ভিজিট করা হয়েছে
• a:hover → এটি একটি লিংক যার উপর মাউস রাখার সাথে সাথে ভিজুয়াল ইফেক্ট দেখাবে।
• a:active → এটি একটি লিংক যেটা বর্তমানে সক্রিয় আছে। অন্যভাবে বলা যায় ইউজার যখন লিংকে ক্লিক করে রেসপন্সের জন্য অপেক্ষা করে সে সময়ের ইফেক্ট।

একটি উদাহরণ দেখুন-

<html>

<head>

<style type="text/css">

a:link {color:#0000FF;} /* unvisited link */

a:visited {color:#FF0000;} /* visited link */

```
a:hover {color:#00FF00;} /* mouse over link */
a:active {color:#000000;} /* selected link */
</style>
</head>
<body>
<a href="http://www.webtechnologyblog.com"
target="_blank">WebTechnologyBlog</a>
</body>
</html>
```

নোটঃ **a:hover**-কে অবশ্যই **a:link**-এর পরে লিখতে হবে, এছাড়া **a:visited** কার্যকর হবে না। এবং **a:active** -কে অবশ্যই **a:hover**-এর পরে লিখতে হবে এছাড়া **a:active** কার্যকর হবে না।

২. ফাস্ট চাইল্ড সিউডো ক্লাসঃ কোন এলিমেন্টের প্রথম উপাদান (প্রথম চাইল্ড) এর প্রপার্টি পরিবর্তন করতে এই ক্লাস ব্যবহার করা হয়। যেমন-

```
<body>
<p>This is first child paragraph</p>
<h2>This is heading</h2>
<p> This is second child paragraph </p>
</body>
```

এখানে **body** এর মধ্যে দুইটি প্যারাগ্রাফ এলিমেন্ট চাইল্ড হিসেবে আছে। এখন যদি শুধুমাত্র প্রথম প্যারাগ্রাফ এলিমেন্টের জন্য স্টাইল তৈরি করতে চাওয়া হয় তাহলে **p** এর ফাস্ট চাইল্ড সিউডো ক্লাস সিলেক্টর ব্যবহার করতে হবে। যেমন-

```
<html>
<head>
```



```
<style type="text/css">
p:first-child
{
color:blue;
}
</style>
</head>

<body>
<p>This is first child paragraph</p>

<h2>This is heading</h2>

<p> This is second child paragraph </p>

</body>
</html>
```

এর ফলে প্রথম প্যারাগ্রাফের টেক্সটের রঙ নীল হবে।

Note: For `:first-child` to work in IE8 and earlier, a [<!DOCTYPE>](#) must be declared.

ফাস্ট চাইল্ড সিউডো ক্লাসের কয়েকটি উদাহরণ দেখুন-

In the following example, the selector matches the first `<i>` element in all `<p>` elements:

```
<html>
<head>
<style type="text/css">
p > i:first-child
{
font-weight:bold;
}
</style>
</head>

<body>
```

```
<p>I am a <i>strong</i> man. I am a <i>strong</i> man.</p>
<p>I am a <i>strong</i> man. I am a <i>strong</i> man.</p>
</body>
</html>
```

In the following example, the selector matches all `<i>` elements in `<p>` elements that are the first child of another element:

```
<html>
<head>
<style type="text/css">
p:first-child i
{
color:blue;
}
</style>
</head>

<body>
<p>I am a <i>strong</i> man. I am a <i>strong</i> man.</p>
<p>I am a <i>strong</i> man. I am a <i>strong</i> man.</p>
</body>
</html>
```

৩. ল্যাঙ্গুয়েজ সিউডো ক্লাসঃ বিভিন্ন ভাষার জন্য বিশেষ কিছু নিয়ম নির্ধারণ করতে ল্যাঙ্গুয়েজ সিউডো ক্লাস ব্যবহার করা হয়। নিচের উদাহরণে ল্যাঙ্গুয়েজ সিউডো ক্লাস ব্যবহার করে `q element`-এ কোটেশান চিহ্ন দেওয়া হয়েছে `lang="no"`-এর সাহায্যে।

উপরের তিনটি সিউডো ক্লাস ছাড়াও আরও একটি

```

4   q:lang(no)
5   {
6   quotes: "~" "~";
7   }
8   </style>
9   </head>
10
11  <body>
12  <p>Some text <q lang="no">A quote in a paragraph</q> Some text.</p>
13  </body>

```

Some text ~A quote in a paragraph~ Some text.

Note: IE8 supports the :lang pseudo-class only if a [!DOCTYPE](#) is specified.

৪.ফোকাস সিউডো ক্লাস: কোন এলিমেন্টকে ফোকাস করতে *:focus* সিউডো ক্লাস ব্যবহার করা হয়। এ সকল এলিমেন্টের ক্ষেত্রে ফোকাস সিউডো ক্লাস ব্যবহার করা হয় যারা ইউজার ইনপুটের উপর নির্ভরশীল যেমন- টেক্সট ফিল্ড, পাসওয়ার্ড ফিল্ড ইত্যাদি। একটি উদাহরণ দেখুন-

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

```

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
input:focus
```

```
{
```

```
background-color:yellow;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
<form action="form_action.asp" method="get">
First name: <input type="text" name="fname" /><br />
Last name: <input type="text" name="lname" /><br />
<input type="submit" value="Submit" />
</form>
</body>
</html>
```

আউটপুটঃ

First name:

Last name:

আমরা যদি অংশের ফিল্ডটিতে ক্লিক করি তবে ব্যাকগ্রাউন্ড হলুদ হয়ে যাবে অর্থাৎ ফোকাস করবে।

সিএসএস সিউডো এলিমেন্টঃ সিএসএস সিউডো ক্লাসের মতোই সিএসএস সিউডো এলিমেন্ট ব্যবহার করে বিভিন্ন উপাদানের অবস্থান, অবস্থা ইত্যাদি সিলেক্টর দ্বারা সিলেক্ট করে স্পেশাল ইফেক্ট দেওয়া হয়। ক্লাস সিলেক্টরের মত সিউডো এলিমেন্ট ব্যবহার করে ডকুমেন্টের বিভিন্ন এলিমেন্ট সিলেক্ট করা যায়।

সিএসএস সিউডো এলিমেন্টের সাধারণ গঠন হল-

Selector: pseudo-element {property: value;}

সিএসএস class-কে pseudo-elements-এর সাথে ব্যবহার করা যায়। যেমন-

Selector.class: pseudo-element {property: value;}

কোন প্যারাগ্রাফের প্রথম অক্ষর বা প্রথম লাইন কেমন দেখাবে তা সিউডো ইলিমেন্ট দ্বারা নির্ধারণ করা হয়। এছাড়া কোন প্যারাগ্রাফ বা হেডিং এর শুরুতে কোন ইমেজ, বা বিশেষ কোন টেক্সট যুক্ত করা সহ আরো অনেক ধরনের কাজ সিউডো ইলিমেন্ট দ্বারা করা যায়।

সিউডো এলিমেন্টের প্রকারভেদঃ

১. প্রথম অক্ষর সিউডো ইলিমেন্ট-(First letter Pseudoelement)
২. প্রথম লাইন সিউডো ইলিমেন্ট-(First first line Pseudoelement)
৩. বিফোর সিউডো ইলিমেন্ট (Before Pseudo element)
৪. আফটার সিউডো ইলিমেন্ট (After Pseudo element)

নিম্নে সিএসএস সিউডো এলিমেন্টগুলো নিয়ে আলোচনা করা হল-

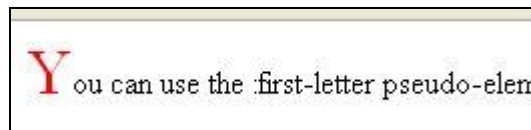
১. প্রথম অক্ষর সিউডো ইলিমেন্টঃ প্রথম অক্ষর সিউডো ইলিমেন্টের মাধ্যমে কোন প্যারাগ্রাফের বা হেডিং এর প্রথম অক্ষরকে নির্ধারিত স্টাইলে দেখানো যায়। যেমন-

P:first-letter{font-weight:bold}

এর ফলে প্রতিটি প্যারাগ্রাফের প্রথম অক্ষর বোল্ট দেখাবে।

```
3 <style type="text/css">
4 p:first-letter
5 {
6 color:#ff0000;
7 font-size:xx-large;
8 }
9 </style>
10 </head>
11 <body>
12 <p>You can use the :first-letter p
```

আউটপুটঃ



Y ou can use the :first-letter pseudo-elem

first-letter pseudo-element শুধুমাত্র এলিমেন্টের ক্ষেত্রে ব্যবহার করা যায়। নিচের প্রপার্টিগুলো **first-letter pseudo-element**-এ ব্যবহার করা যায়-

- font properties

- color properties
- background properties
- margin properties
- padding properties
- border properties
- text-decoration
- vertical-align (only if "float" is "none")
- text-transform
- line-height
- float
- clear

২.প্রথম লাইন সিউডো ইলিমেন্ট: প্রথম লাইন সিউডো ইলিমেন্টের মাধ্যমে কোন প্যারাগ্রাফের প্রথম লাইনকে নির্ধারিত স্টাইলে দেখানো যায়।যেমন-

P:first-line{font-style:bold}

এর ফলে প্রতিটি প্যারাগ্রাফের প্রথম লাইন বোল্ড দেখাবে।

```

3  <style type="text/css">
4  p:first-line
5  {
6  color:#ff0000;
7  font-variant:small-caps;
8  }
9  </style>
10 </head>
11 <body>
12 <p>You can use the :first-line pseudo-el

```

এর ফলে প্রতিটি প্যারাগ্রাফের প্রথম লাইনের কালার লাল দেখাবে।

first-line pseudo-element শুধুমাত্র **block-level elements**-এর ক্ষেত্রে ব্যবহার করা যায়।

নিচের প্রপাটিগুলো **first-line pseudo-element**-এ ব্যবহার করা যায়-

- font properties
- color properties
- background properties
- word-spacing
- letter-spacing
- text-decoration

- vertical-align
- text-transform
- line-height
- clear

৩.বিফোর সিউডো ইলিমেন্টঃ বিফোর সিউডো ইলিমেন্ট ব্যবহার করে কোন কনটেন্টের সামনে আরেকটি কনটেন্ট ইনসার্ট করা যায়। এটা আমাদের অনেক কাজে লাগবে। নিচের উদাহরণে আমরা দেখিয়েছি কীভাবে প্রতিটি হেডিং-এর সামনে একটি ইমেজ ইনসার্ট করা যায়।

ইমেজ ইনসার্টের জন্য লিখতে হবেঃ

Selector:before{content:url(image_name.image_format);}

```

3  <style type="text/css">
4    h1:before {content:url(1h.gif);}
5  </style>
6  </head>
7
8  <body>
9    <h1>This is a heading</h1>
10   <p>The :before pseudo-element inserts content.</p>
11   <h1>This is a heading</h1>
12 </body>
13 </html>

```



অন্যান্য ক্ষেত্রে লিখতে হবে-

Selector:before{content:" যে কনটেন্টকে অন্য কনটেন্টের সামনে দেখাতে চান তা এই কোটেশানের মাঝে লিখুন";}

একটি উদাহরণ দেখুনঃ

```
p:before
{
content:"Read this -";
background-color:yellow;
color:red;
font-weight:bold;
}
```

৪.আফটার সিউডো ইলিমেন্টঃ আফটার সিউডো ইলিমেন্ট ব্যবহার করে কোন কনটেন্টের পরে আরেকটি কনটেন্ট ইনসার্ট করা যায়। এটা আমাদের অনেক কাজে লাগবে। নিচের উদাহরণে আমরা দেখিয়েছি কীভাবে প্রতিটি হেডিং-এর পরে একটি ইমেজ ইনসার্ট করা যায়।

```
3 <style type="text/css">
4   h1:after {content:url(1h.gif) ;}
5 </style>
6 </head>
7 <body>
8   <h1>This is a heading</h1>
9   <p>The :after pseudo-element inserts content after an element.</p>
10  <h1>This is a heading</h1>
11 </body>
```



অন্যান্য ক্ষেত্রে লিখতে হবে-

Selector: after {content:" যে কনটেন্টকে অন্য কনটেন্টের পরে দেখাতে চান তা এই কোটেশানের মাঝে লিখুন।";}

একটি উদাহরণ দেখুনঃ


```

5 <style type="text/css">
6   p:after
7   {
8     content:"- Remember this";
9     background-color:yellow;
10    color:red;
11    font-weight:bold;
12   }
13 </style>
14 </head>
15 <body>
16 <p>My name is Donald</p>
17 <p>I live in Ducksburg</p>
18

```

Output:

My name is Donald- **Remember this**

I live in Ducksburg- **Remember this**

সিএসএস-এ ব্যবহৃত সকল সিউডো ক্লাস/ এলিমেন্ট

সিলেক্টর	উদাহরণ	উদাহরণের বর্ণনা
<code>:link</code>	<code>a:link</code>	ভিজিট করা হয়নি এমন লিংক।
<code>:visited</code>	<code>a:visited</code>	ভিজিট করা হয়েছে এমন লিংক।
<code>:active</code>	<code>a:active</code>	সক্রিয় লিংক।
<code>:hover</code>	<code>a:hover</code>	মাউস ওভারে লিংকের ইফেক্ট।
<code>:focus</code>	<code>input:focus</code>	ইনপুটের কোন এলিমেন্টকে ফোকাস করে।
<code>:first-letter</code>	<code>p:first-letter</code>	প্রতিটা প্যারাগ্রাফের প্রথম বর্ণকে সিলেক্ট করে।
<code>:first-line</code>	<code>p:first-line</code>	প্রতিটা প্যারাগ্রাফের প্রথম লাইনকে সিলেক্ট করে।
<code>:first-child</code>	<code>p:first-child</code>	প্রতিটা প্যারাগ্রাফের প্রথম চাইল্ডকে সিলেক্ট করে।
<code>:before</code>	<code>p:before</code>	প্যারাগ্রাফের সামনে নুতন কোন এলিমেন্ট ইনসার্ট করে।
<code>:after</code>	<code>p:after</code>	প্যারাগ্রাফের পিছনে নুতন কোন এলিমেন্ট ইনসার্ট করে।
<code>:lang(language)</code>	<code>p:lang(it)</code>	Selects every <p> element with a lang attribute value starting with "it"

অধ্যায়-চৌদ্দ

[সিএসএস ডিসপ্লে ও ভিজিবিলাটি প্রপাটি]

সিএসএস ডিসপ্লে প্রপাটিঃ

কোন উপাদান কীভাবে ডকুমেন্টে প্রদর্শিত হবে তা নির্দেশ করা হয় **display** প্রপাটির মাধ্যমে। এই প্রপাটি সব উপাদান ও মিডিয়াম ক্ষেত্রে ব্যবহৃত হতে পারে। ডিসপ্লে প্রপাটির বেশ কিছু **value** হতে পারে, এদের মধ্যে সবচেয়ে বেশি ব্যবহৃত হয়,

- ডিসপ্লে নান -(**display:none**)
- ডিসপ্লে ইন লাইন -(**display:inline**)
- ডিসপ্লে ব্লক - (**display:block**)

ডিসপ্লে নান -(**display:none**)→ কোন বিশেষ উপাদানকে প্রদর্শন না করার জন্য ডিসপ্লে নান ব্যবহার করা হয়, এজন্য **Declaration** করতে হয় **display:none**;। এর মান দেওয়া হলে কোন বক্স তৈরি হবে না এবং স্ক্রীন-এ কিছু দেখা যাবে না। ওই উপাদানের চাইল্ড যারা তাদের জন্যও কোন বক্স তৈরি হবে না। যেমন-

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
h1.hidden {display:none;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>This is a visible heading</h1>
```

```
<h1 class="hidden">This is a hidden heading</h1>
```

```
<p>Notice that the hidden heading does not take up space.</p>
```

```
</body>
```

```
</html>
```

কোড বিশ্লেষণঃ `display:none` ব্যবহারের ফলে **This is a hidden heading** লেখাটি ব্রউজারে দেখা যাবে না।

ডিসপ্লে ইন লাইন `-(display:inline)→` সি এস এস এর মাধ্যমে এইচটিএমএল উপাদান সমূহকে একই লাইনে প্রদর্শনের জন্য ডিসপ্লে ইন লাইন ব্যবহার করা হয়। এর ফলে উপাদানটি ইনলাইন উপাদান হিসেবে প্রদর্শিত হবে অর্থাৎ এর আগে বা পিছে লাইন ব্রেক তৈরি হবে না। এটা মূলত সবচেয়ে বেশি ব্যবহৃত হয় নেভিগেশন বারে। এইচটিএমএল উপাদান সমূহকে একই লাইনে প্রদর্শনের জন্য **Declaration** করতে হয় `display:inline`।

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

```
li{display:inline;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>Display this link list as a horizontal menu:</p>
```

```
<ul>
```

```
<li><a href="../html/index.php" target="_blank">HTML</a></li>
```

```
<li><a href=" index.php " target="_blank">CSS</a></li>
<li><a href=" ../js/ index.php " target="_blank">JavaScript</a></li>
<li><a href=" ../xml/ index.php " target="_blank">XML</a></li>
</ul>
```

```
</body>
```

```
</html>
```

আউটপুটঃ

Display this link list as a horizontal menu:

[HTML](#) [CSS](#) [JavaScript](#) [XML](#)

কোড বিশ্লেষণঃ **display:inline** ব্যবহারের ফলে লিস্ট আইটেমগুলো এক লাইনে দেখাচ্ছে।

ডিসপ্লে ব্লক - (**display:block**)→ সি এস এস এর মাধ্যমে এইচটিএমএল উপাদান সমূহকে একই কলামে প্রদর্শনের জন্য ডিসপ্লে ব্লক ব্যবহার করা হয়। এর মান দেয়া হলে ঐ উপাদান একটি ব্লক হিসেবে কাজ করবে, অর্থাৎ সেটি ব্লক লেভেল উপাদানের মতোই আগে বা পিছে লাইন ব্রেক তৈরি করবে। এটা সবচেয়ে বেশি ব্যবহৃত হয় ভাটিক্যাল নেভিগেশন বারে।

নিচের উদাহরণে **display** প্রপার্টির ব্যবহার দেখানো হল-

```
<html>
```

```
<head>
```

```
<style>
```

```
ul
```

```
{
```

```
list-style-type:none;
```

```
margin:0;
```

```
padding:0;
}
a
{
display:block;
width:60px;
}
</style>
</head>

<body>
<ul>
<li><a href="#home">Home</a></li>
<li><a href="#news">News</a></li>
<li><a href="#contact">Contact</a></li>
<li><a href="#about">About</a></li>
</ul>

</body>
</html>
```

আউটপুটঃ

[Home](#)
[News](#)
[Contact](#)
[About](#)

কোড বিশ্লেষণঃ **display: block** ব্যবহারের ফলে লিস্ট আইটেমগুলো লম্বালম্বি (ভাটিক্যাল) দেখাচ্ছে এবং এরা একটি ব্লক তৈরি করেছে।

ভিজিবিলাটি প্রপাটিঃ

কোন উপাদান ব্রাউজারে দেখা যাবে (**visible**), না কি দেখা যাবে না (**hidden**) তা নির্ধারণ করা হয় **visibility** প্রপাটির মাধ্যমে। এর দুটি মান হতে পারে-**visible** ব্যবহার করা হলে ঐ উপাদান দৃশ্যমান হবে, আর **hidden** ব্যবহার করা হলে তা দেখা যাবে না। কোন উপাদানকে প্রদর্শন করার জন্য **Declaration** করতে হয় **visibility:visible;**। **visibility:hidden** ব্যবহার করা হলে ঐ উপাদান দেখা না গেলেও তার জন্য যে পরিমাণ জায়গা বরাদ্দ আছে তা ফাঁকা দেখা যাবে অর্থাৎ লেআউটের উপর প্রভাব থেকে যাবে। **display :none** ব্যবহার করা হলে সেই উপাদান দেখা যায় না এবং সে স্থান দখল করে নেয় অন্য উপাদান। **display** প্রপাটির সাথে **visibility** প্রপাটির পার্থক্য এখানেই। নিচের উদাহরণে এটি দেখানো হল-

```
<html>
<head>
<style type="text/css">
h1.hidden {visibility:hidden;}
</style>
</head>
<body>
<h1>This is a visible heading</h1>
```

```
<h1 class="hidden">This is a hidden heading</h1>
```

```
<p>Notice that the hidden heading still takes up space.</p>
```

```
</body>
```

```
</html>
```

আউটপুটঃ

This is a visible heading

Notice that the hidden heading still takes up space.

== समाप्त ==