



آموزش CSS

عنوان درس: مهندسی اینترنت

استاد: دکتر حقیقی

حل تمرین: عباس نادری

دانشجو: سید فرهنگ حسینی ۸۵۲۱۳۰۴۶

سید مهدی جزایری ۸۵۲۱۳۱۷۰

بهار ۹۰

فهرست

- ۳..... نحوه شروع کار با (CSS)
- ۳..... سلکتور کلاس
- ۴..... افزودن استایل به نوع مشخصی از عناصر
- ۵..... سلکتور id
- ۶..... وارد کردن CSS در تگهای HTML با استفاده از شناسه Style
- ۷..... تعریف کردن استایل در بخش head صفحه HTML
- ۸..... تنظیم عرض، ارتفاع، و overflow با CSS
- ۱۱..... ویژگیهای متن در CSS
- ۱۳..... ویژگیهای مربوط به متن در CSS

نحوه شروع کار با (CSS)

CSS یا همان *Cascading Style Sheet* روشی است برای قالب بندی و طراحی اجزای صفحه از قبیل متن، تصویر، زمینه صفحه، کادرها و تقریباً هر چیزی که در طراحی صفحه استفاده می شود بدون استفاده از کدهای HTML. از طریق این استایل شیتها می توانیم از دوباره نویسی کدهای HTML که باعث مشکل شدن کد نویسی و همچنین کند شدن سرعت بارگذاری صفحه در اینترنت می شود جلوگیری کنیم. برای مثال می توانیم یک بار نوع فونت مورد استفاده در یک صفحه را تعیین کنیم و دیگر مجبور نباشیم کد مربوط به فونت را در صفحه به کار ببریم. همچنین می توانیم از این ابزار قدرتمند در مواردی به جای جاوا اسکریپت نیز استفاده کنیم و چون مشکلات جاوا اسکریپت مثلاً غیر فعال بودن جاوا اسکریپت در مرورگر را ندارد با خیالی آسوده تر می توانیم از آن استفاده کنیم .

CSS هم مانند جاوا اسکریپت این امکان را به ما می دهد که از یک فایل استایل خارجی برای تعیین خواص بخشهای مختلف صفحه های یک سایت استفاده کنیم. در این صورت می توانیم ظاهر همه صفحه ها را با ایجاد یک تغییر کوچک در فایل استایل خارجی تغییر دهیم. مثلاً فرض کنید که رنگ متن همچنین نوع فونت متن صفحه هایمان را در یک فایل استایل تعیین کرده ایم و از آن در طراحی چندین صفحه استفاده کرده ایم، حالا اگر بخواهیم رنگ متن همه این صفحات را تغییر دهیم تنها کافی است تا رنگ متن را در همان فایل استایل تغییر دهیم و نیازی به تغییر دادن صفحات نداریم.

سلکتور کلاس

با استفاده از کلاس می توانیم برای یک تگ HTML استایلهای متفاوتی را تعریف کنیم. برای نمونه فرض کنیم می خواهیم در صفحه دو نوع پاراگراف داشته باشیم: یکی با متن مشکی و دیگری با متن قرمز. برای مشخص کردن این دو نوع پاراگراف می توانیم از سلکتور کلاس استفاده کنیم. ابتدا استایل را به صورت زیر می نویسیم:

```
p.black {color: black}
p.red {color:red}
```

در مرحله بعد باید برای هر پاراگرافی که می خواهیم به رنگ مشکی باشد از شناسه **class="black"** در تگ `<p>` استفاده کنیم و برای پاراگراف قرمز از شناسه **class="red"** البته نام کلاسها اختیاری است ولی بهتر است آنها را طوری انتخاب کنید که مفهوم داشته باشند تا وقتی که حجم استایل شما بیشتر شد به گنگ نباشد. در مثال زیر می توانید کد مربوط به نوشتن دو پاراگراف بالا را در متن HTML مشاهده کنید:

```
</p>این متنی است که به رنگ مشکی نمایش داده می شود<p class="black">
</p>این متن به رنگ قرمز نمایش داده می شود<p class="red">
```

نکته : شما می توانید بیش از یک کلاس برای یک تگ HTML تعریف کنید اما برای این کار نباید دو بار از شناسه **class** استفاده کرد. بلکه باید در یک شناسه **class** نام دو کلاس مورد نظر را با یک فاصله بین آنها وارد کرد. برای مثال برای پاراگراف زیر از دو کلاس فرضی **red** و **center** استفاده شده است:

```
<p class="red center">این متن پاراگرافی است که تحت تأثیر دو کلاس
```

در نوشتن استایل می توان از نوشتن نام تگ در سلکتور کلاس چشم پوشی کرد. در این صورت ویژگیهای تعریف شده در آن کلاس بر هر تگ *HTML* که از کلاس مربوطه در شناسه *class* آن استفاده شده باشد اثر می گذارد. برای مثال در کلاس زیر ویژگی وسط چین را تعریف می کنیم و از نام تگ در سلکتور کلاس استفاده نمی کنیم:

```
.center {text-align: center}
```

اکنون برای هر تگی که احتیاج به وسط چین داشته باشد می توانیم از کلاس بالا استفاده کنیم. به چند نمونه در زیر توجه کنید:

```
<p class="center">متن این پاراگراف به صورت وسط چین نمایش داده می شود</p>
<div class="center">متن این قسمت به صورت وسط چین نمایش داده می شود</div>
<h2 class="center">متن این سرفصل به صورت وسط چین نمایش داده می شود</h2>
```

ما می توانیم از سلکتور کلاس برای تگی که حاوی شناسه *class* است استفاده کنیم که در قسمتهای قبلی در این مورد توضیح داده شد. علاوه بر این می توانیم از ویژگیهای تعریف شده در سلکتور کلاس برای تگهایی که در یک تگ محتوی شناسه *class* قرار دارند نیز استفاده کنیم. مثلاً برای لینکهایی که در یک *DIV* قرار دارند و برای *DIV* کلاس تعریف شده است. برای این کار در نوشتن استایل مربوطه باید پس از نقطه که مشخص کننده سلکتور کلاس است نام کلاس را وارد کنیم و پس از وارد کردن یک فضای خالی نام تگی که می خواهیم ویژگیهای بر روی آن اعمال شود، به مثال زیر توجه کنید:

در این مثال از استایل زیر در استفاده می کنیم:

```
.test { border: #F00 1px double }
.test a { color: green }
.test a:hover { border: yellow 2px solid }
```

کد *HTML* زیر برای این مثال مورد استفاده قرار می گیرد:

```
<div class="test">
<a href="#">لینک آزمایشی شماره ۱</a>
</div>
<a href="#">لینک آزمایشی شماره ۲</a>
```

افزودن استایل به نوع مشخصی از عناصر

همانطور که می دانید با وجود اینکه برخی از عناصر در صفحه با هم تفاوت دارند، از یک تگ *HTML* برای ایجاد آنها استفاده می شود. برای مثال برخی از این عناصر را که از تگ *<input>* در به وجود آوردن آنها استفاده می شود می توانید در زیر مشاهده کنید:

- **<"input type="text">**
کادرهای متنی که در فرمها استفاده می شود.

- `<"input type="checkbox">`
به وجود می آید `<input>` چک باکس هم با استفاده از تگ
- `<"input type="radio">`
در ساختن دکمه های رادیو هم از همان تگ استفاده می شود
- `<"input type="button">`
این تگ در ساختن هم استفاده می شود
- `<"input type="submit">`
استفاده می شود `<input>` برای ساختن دکمه ارسال فرم هم از تگ

حالا فرض کنید ما در یک صفحه از چند نمونه از عناصر بالا استفاده کرده ایم ولی فقط می خواهیم ویژگیهای یکی از آنها را تغییر دهیم. دو روش برای این کار وجود دارد: یکی استفاده از کلاس است که در این روش اگر تعداد زیادی از عنصر مورد نظر در صفحه وجود داشته باشد باید به همان تعداد از شناسه `class` در تگهای آنها استفاده کنیم و ممکن است این کار باعث افزایش حجم صفحه یا موجب سختی در نوشتن کد شود. روش دوم تعریف کردن سلکتور `input` برای عناصر مورد نظر است که به صورت زیر انجام می گیرد:

فرض کنید می خواهیم فقط عرض کادرهای متنی را در صفحه تغییر دهیم. در صورتی که در استایل از سلکتور `input` به تنهایی استفاده کنیم عرض همه چک باکسها، دکمه ارسال، دکمه های رادیو و سایر عناصری که از تگ `<input>` استفاده کرده اند نیز تغییر می کند. برای انجام این کار می توانید سلکتور را به صورت زیر تعریف کنیم:

```
input[type="text"] {
width: 200px
}
```

کد استایلی که در بالا نوشتیم فقط بر روی کادر های متنی که در آنها از شناسه `"type="text"` استفاده شده است اثر می گذارد .

سلکتور id

روش دیگری که برای تعریف استایل وجود دارد استفاده از `id` است. شناسه `id` احتمالاً در آینده جایگزین شناسه `name` در تگهای `HTML` می شود بنا بر این در هر صفحه نباید بیش از یک بار از هر `ID` استفاده شود. اگر بیش از یک بار از یک `id` استفاده شود کارایی صفحه تغییری نمی کند ولی صفحه اعتبار خود را از دست می دهد. سلکتور `id` با علامت `#` مشخص می شود. با استایل زیر همه عناصر که شناسه `"id="border"` دارند با کادری سبز رنگ نمایش داده می شوند :

```
#border { border: green thin solid }
```

برای مثال ما از استایل بالا در این صفحه استفاده کرده ایم. حالا کد زیر را وارد صفحه می کنیم:

```
<h3 id="border">این یک سرفصل با اندازه ۳ است</h3>
<div id="border">این قسمت یک</div>
```

در قسمت زیر می توانید نتیجه کد بالا را مشاهده کنید:

این یک سرفصل با اندازه ۳ است

این قسمت یک *div* است

همانطور که کلاس را می توانستیم برای یک عنصر مشخص تعریف کنیم در مورد *id* نیز می توانیم این کار را انجام دهیم. به استایل زیر توجه کنید:

```
p#border { border: green thin solid }
```

استایل بالا فقط روی پاراگرافی که شناسه **"id=green"** در آن به کار رفته اثر می گذارد .

وارد کردن CSS در تگهای HTML با استفاده از شناسه Style

ساده ترین راه برای افزودن استایل به یک صفحه استفاده از شناسه **style** در تگهای *HTML* است. برای افزودن استایل به یک تگ باید عبارت **"=style"** را وارد تگ مربوطه کنیم و ویژگیهای مورد نیاز خود را بین دو دابل کوت (") وارد می کنیم. مثلاً فرض کنید می خواهیم به یک تگ *DIV* استایل اضافه کنیم. در این صورت تگ *DIV* به صورت زیر در می آید.

```
<div style="property:value">
```

برای نمونه وقتی بخواهیم متن داخل یک *DIV* به رنگ سبز نمایش داده شود می توانیم به صورت زیر تگ *DIV* را بنویسیم:

```
<div style="color:green"> متن داخل DIV </div>
```

توجه کنید که بین ویژگی *color* و رنگ سبز (*green*) از علامت دو نقطه (:) استفاده شده نه از علامت مساوی که در تگهای *HTML* استفاده می شود. همچنین لازم نیست مقادیر مورد نظر را بین علامت های نقل قول (") قرار دهیم. می توانید نتیجه به کارگیری کد بالا را در سطر زیر مشاهده کنید:

متن داخل DIV

به خاطر داشته باشید که برای اینکه خاصیت این استایل از بین برود باید تگی را که استایل در آن به کار رفته است ببندید. در اینجا تا زمانی که تگ *DIV* بسته نشده باشد متن ما به رنگ قرمز نمایش داده خواهد شد.

این امکان نیز وجود دارد که به یک تگ *HTML* بیش از یک ویژگی افزوده شود. تنها کافی است بین ویژگیهای مختلف از یک نقطه ویرگول (;) استفاده کنیم. برای مثال اگر بخواهیم متنی که در یک *DIV* قرار دارد به رنگ قرمز و به صورت *ایتالیک* باشد می توانیم به صورت زیر کد *DIV* را بنویسیم:

`<div style="color:red; font-style:italic">` این متن قرمز رنگ و ایتالیک است `</div>`

نتیجه به صورت زیر خواهد بود:

این متن قرمز رنگ و ایتالیک است.

در این روش می توانیم به هر تعداد که لازم باشد از ویژگیهای مختلف درون یک تگ *HTML* استفاده کنیم. به مثال زیر توجه کنید:

`<div style="color:blue; font-style:italic; font-weight:bold; font-size:12pt; text-align:center">` اینجا قرار می گیرد متن مورد نظر `</div>`

در این مثال ما متنی را با رنگ آبی، ایتالیک، ضخیم، اندازه فونت ۱۲ pt در وسط صفحه خواهیم داشت:

متن مورد نظر اینجا قرار می گیرد.

در قسمتهای بعدی لیستی را از ویژگیهای مختلفی که با استایل می توانیم آنها را کنترل کنیم خواهیم آورد. در اینجا فقط قصد آموزش نحوه قرارگیری استایل در تگهای *HTML* را داشتیم.

تعریف کردن استایل در بخش **head** صفحه **HTML**

استفاده از استایلهای می تواند به ما کمک کند تا از دوباره نویسی فرمانها جلوگیری کنیم. برای انجام این کار ما می توانیم یک *Style* را در بخش *HEAD* در متن *HTML* خود تعریف کنیم. یک استایل در *HEAD* با تگ `<STYLE>` آغاز می شود و با `</STYLE>` به پایان می رسد. در قسمت زیر می توانید یک نمونه از استایل را که مشخصاتی را برای تگ `` تعریف می کند مشاهده کنید که در بخش *HEAD* در متن قرار گرفته است.

```
<head>
<style>
<!--
span { color:red; font-style:italic }
-->
</style>
</head>
```

همانطور که می بینید استایل با تگ `<style>` آغاز می شود. سپس یک کامنت را شروع می کنیم تا مرورگرهایی که از *CSS* پشتیبانی نمی کنند استایل را نادیده بگیرند و در صفحه به نمایش در نیاورند. در قسمت بعد این خط را مشاهده می کنید:

`span { color:red; font-style:italic }`

این خط مشخص می کند که هر بار که در صفحه از تگهای `<STYLE> <STYLE/>` استفاده می شود. متن بین آنها به رنگ قرمز و به صورت ایتالیک نمایش داده می شود. در نوشتن استایل باید در نظر داشته باشید که نباید از علامتهای کوچکتر و بزرگتر (`<`) و (`>`) که در اطراف تگهای `HTML` استفاده می شود، استفاده کنید. همچنین به جای استفاده از علامتهای نقل قول ("`"`) که در متن `HTML` استفاده می شود از آکولاد (`{ }`) در اطراف خواص استایل استفاده می کنیم. خواص مختلف یک استایل هم به وسیله یک نقطه ویرگول (`;`) از هم جدا می شوند .

وقتی ما استایلی را در بخش `HEAD` متن خود برای یک تگ خاص تعریف می کنیم برای محتویات همه تگهایی که استایل برای آن تعریف شده در صفحه مورد استفاده قرار می گیرد. برای مثال وقتی که ما کد بالا را در `HEAD` متن خود قرار دادیم، هر تگ `` که در صفحه استفاده شود دارای متنی قرمز و ایتالیک است و نیاز ندارد هر بار دوباره مشخصات متن مورد نظر را تعریف شود.

برای مثال کد زیر را وارد متن `HTML` خود می کنیم:

است و مشخصات `` تگ اما این متن خارج `` این متن قرمز و ایتالیک است، `` است و مشخصات `
` آنرا ندارد `` این متن نیز قرمز و ایتالیک است ``

نتیجه را به صورت زیر مشاهده خواهید کرد:

این متن قرمز و ایتالیک است، اما این متن خارج تگ `` است و مشخصات آنرا ندارد. این متن نیز قرمز و ایتالیک است.

با این روش ما می توانیم تقریباً خواص همه تگهای `HTML` را کنترل کنیم، اما این کافی نیست. در قسمتهای بعدی روشهایی را معرفی می کنیم که یک استایل مشخص به وسیله تگهای مختلفی مورد استفاده قرار بگیرد. این کار را با استفاده از `CLASS` و `ID` انجام می دهیم که به آن خواهیم پرداخت.

تنظیم عرض، ارتفاع، و `overflow` با `CSS`

عرض و ارتفاع

`width`

این مشخصه به ما امکان می دهد تا عرض یک بخش از صفحه را تعیین کنیم. برای مثال می توانیم با آن عرض یک `DIV` را تعیین کنیم. در مثال زیر برای `DIV` عرضی معادل ۲۰۰ پیکسل تعیین می کنیم:

`<div style="width:200px">۲۰۰` پس از اینکه عرض آن از ۲۰۰ بعدی سطرهای پیکسل بیشتر شد در این جمله
`</div>` پیدا می کند ادامه

نتیجه به صورت زیر خواهد بود:

این جمله پس از اینکه عرض آن از ۲۰۰ پیکسل بیشتر شد در سطرهای بعدی ادامه پیدا می کند.

مقادیری که برای پارامتر *width* قرار می دهید می تواند بر اساس پیکسل باشد که در این صورت باید پس از مقدار مربوطه حروف *px* که نشانه پیکسل است را بنویسید. مقادیر ممکن است بر اساس درصد باشند که در آن صورت باید علامت درصد (%) را پس از مقدار قرار دهید.

height

این پارامتر به ما اجازه می دهد تا ارتفاع بخش مشخصی را تعیین کنیم. برای مثال وقتی که ما این پارامتر را برای یک *DIV* استفاده می کنیم و مقدار آن را ۱۰۰ پیکسل قرار می دهیم تا وقتی که متن موجود در *DIV* از ۱۰۰ پیکسل کمتر ارتفاع دارد ارتفاع *DIV 100* پیکسل خواهد بود. ولی وقتی که متن داخل *DIV* زیاد باشد و از ۱۰۰ پیکسل بیشتر شود در نتیجه ارتفاع *DIV* هم از ۱۰۰ تجاوز خواهد کرد تا همه متن موجود قابل خواندن باشد. به مثال زیر توجه کنید:

۱. ابتدا هنگامی که ارتفاع متن از ارتفاع *DIV* کمتر است:

`<div style="height:100px">` ۱۰۰ پیکسل کمتر `</div>` است ارتفاع این متن از

ارتفاع این متن از ۱۰۰ پیکسل کمتر است.

۲. حالا هنگامی که ارتفاع متن از ارتفاع *DIV* بیشتر است:

پیکسل `
100
` متن از `
` این `
` ارتفاع `<div style="height:100px">`
`
` کمتر `</div>` است.

ارتفاع

این

متن از

100

پیکسل

کمتر

است.

۳. scroll

این مقدار باعث خواهد شد تا در صورتی که محتویات اندازه ای بیش از طول و عرض لازم دارند با استفاده از اسکرول بار قابل مشاهده باشند .

به یک مثال توجه کنید:

```
<div style="width:200px; overflow:scroll; border:dashed 1px #3399FF; white-space:nowrap">
```

دیده نمی *DIV* قسمت‌های بیشتر از عرض *overflow* است ولی با استفاده از *DIV* طول این سطر بیشتر از عرض شوند.

```
</div>
```

این هم نتیجه:

طول این سطر بیشتر از عرض *DIV* است ولی با استفاده از *overflow* قسمت‌های بیشتر از عرض *DIV* دیده نمی شوند.

ویژگیهای متن در CSS

در این قسمت برخی از ویژگیهای متن را که در *CSS* استفاده می شوند به شما معرفی می کنیم. با کمک این ویژگیها می توانیم نحوه قرار گرفتن متن در صفحه، فاصله بین حروف و بسیاری از خواص یک متن را تغییر دهیم.

letter-spacing

این ویژگی به ما اجازه می دهد تا فاصله میان حروف یک کلمه را در یک متن تعیین کنیم. این ویژگی می تواند مقادیری را در مقیاس پیکسل به خود بگیرد یا از مقدار *normal* استفاده کند. گزینه پیش فرض برای این ویژگی همان *normal* است. در اینجا به ذکر یک مثال در مورد این ویژگی می پردازیم:

فرض کنید می خواهیم از این ویژگی برای متن موجود در یک تگ *DIV* استفاده کنیم:

```
<div style="letter-spacing:5px">
```

حروف این متن با فاصله ۵ پیکسل از یکدیگر نمایش داده خواهند شد

```
</div>
```

نتیجه را به صورت زیر مشاهده خواهید کرد:

حروف این متن با فاصله ۵ پیکسل از یکدیگر نمایش داده خواهند شد .

text-align

این ویژگی به ما اجازه می دهد تا نحوه قرارگیری قسمتی از متن را از لحاظ راست چین، چپ چین یا وسط چین بودن تنظیم کنیم.

این ویژگی می تواند مقادیر زیر را به مورد استفاده قرار دهد:

- **left :**

با استفاده از این مقدار می توانیم محتویات یک تگ را در سمت چپ صفحه نمایش دهیم .

- **right :**

این مقدار معین می کند که محتویات یک تگ باید در سمت راست صفحه نمایش داده شوند .

- **center :**

این مقدار محتویات یک تگ را به صورت وسط چین تعریف میکند.

برای مثال اگر می خواهید متنی که در یک *DIV* قرار دارد به صورت چپ چین نمایش داده شود می توانیم از این ویژگی به مانند زیر استفاده کنیم :

در سمت چپ *DIV* محتویات این `<div style="text-align:left; border:thin dashed #00CCFF">` در سمت چپ *DIV* محتویات این `</div>` صفحه نمایش داده می شوند

نتیجه به صورت زیر خواهد بود:

محتویات این *DIV* در سمت چپ صفحه نمایش داده می شوند.

text-decoration

این ویژگی به ما اجازه می دهد تا برای قسمت خاصی از متن خاصیت‌های از قبیل زیرخط تعریف کنیم. مثلاً فرض کنید می خواهیم یک متن را با خطی بالای آن به وجود بیاوریم. در اینجا می توانیم از این ویژگی به صورت زیر استفاده متین:

`<div style="text-decoration: overline">!!به این متن توجه کنید</div>`

این ویژگی چندین مقدار دارد که در اینجا به آنها می پردازیم:

- **none**

این گزینه به صورت پیش فرض بر روی متن اعمال می شود (البته برای لینکها به صورت پیش فرض *underline* اعمال می شود) و در نتیجه آن متنها به صورت ساده و معمولی نمایش داده می شوند. در

صورتی که بخواهیم لینک‌هایی در متن ما بدون زیر خط نشان داده شوند می‌توانیم از این گزینه برای تگ لینک استفاده کنیم .

- **underline**

از این گزینه می‌توان برای زیر خط دار کردن متنها استفاده کرد .

- **overline**

این گزینه برای نمایش خط بالای متن استفاده می‌شود .

- **line-through**

این گزینه یک خط را در میانه متن ایجاد می‌کند و آنرا به صورت خط خورده نشان می‌دهد .

- **blink**

از این گزینه می‌توانیم برای چشمک زن کردن متن استفاده کنیم. البته این گزینه توسط مرورگر اینترنت اکسپلورر حمایت نمی‌شود و فقط در مرورگرهای Netscape نمایش داده می‌شود .

در اینجا یک لینک را می‌بینید که با استفاده از ویژگی بالا بدون زیر خط نمایش داده می‌شود:

`متن مورد استفاده در لینک`

ویژگیهای مربوط به متن در CSS

در این قسمت به ادامه ویژگی‌هایی که بر روی متن‌ها اثر می‌گذارند می‌پردازیم.

text-transform

این ویژگی مشخص می‌کند که متن با استفاده از حروف بزرگ نوشته شود یا حروف کوچک و یا به صورت معمولی البته بدون آنکه در حروفی که تایپ شده است تغییری به وجود آید. با این روش اگر خواستید قسمتی از متن مثلاً برای تأکید بیشتر با حروف بزرگ مشاهده شود نیازی نیست که دوباره آن قسمت را از اول تایپ کنید. البته این ویژگی در زبان فارسی چندان استفاده‌ای ندارد ولی در زبان‌هایی مثل انگلیسی می‌توانید آنرا به کار ببرید.

به یک مثال در مورد این ویژگی توجه کنید:

```
<div style="text-transform: uppercase">This is an example of uppercase in text-transform property.</div>
```

این هم نتیجه:

THIS IS AN EXAMPLE OF UPPERCASE IN TEXT-TRANSFORM PROPERTY.

همانطور که مشاهده می‌کنید بدون استفاده از حروف بزرگ در نوشتن کد، متن ما با حروف بزرگ نمایش داده می‌شود.

این ویژگی چند مقدار دارد که در اینجا مشاهده می‌کنید:

- **none**
با انتخاب این گزینه متن به همان صورتی که تایپ شده است نمایش داده می شود .
- **lowercase**
با انتخاب این گزینه متن با حروف کوچک نمایش داده می شود .
- **uppercase**
این گزینه متن را با حروف بزرگ نمایش می دهد .
- **capitalize**
این گزینه حروف اول کلماتی را که با حروف کوچک نوشته شده اند با حروف بزرگ و بقیه حروف را با حروف کوچک نمایش می دهد. به مثال زیر توجه کنید:

`<div style="text-transform: capitalize">this is an example of capitalize value for text-transform propety</div>`

متن به صورت زیر نمایش داده می شود:

this is an example of capitalize value for text-transform propety

line-height

این ویژگی به ما امکان می دهد تا ارتفاع سطرها را مشخص کنیم. این ارتفاع می تواند به صورت درصد یا در مقیاس پیکسل باشد و فاصله بین سطرها را مشخص می کند.

برای مثال اگر بخواهیم سطرها از هم ۵۰ پیکسل فاصله داشته باشند می توانیم به صورت زیر عمل کنیم:

`<div style="line-height: 50px">پیکسل فاصله وجود

20 بین این سطر و سطر بعدی</div>` دارد.

نتیجه به صورت زیر خواهد بود:

بین این سطر و سطر بعدی
20 پیکسل فاصله وجود دارد.

text-indent

این ویژگی مقدار تو رفتگی متن را در سطر اول هر قسمت (مثلاً سطر اول هر پاراگراف) مشخص می کند. این مقدار می تواند بر حسب پیکسل یا درصد باشد.

در مثال زیر این مقدار تو رفتگی را ۳۰ پیکسل در نظر می گیریم:

```
<div style="text-indent: 30px">  
<br>سطر اول این قسمت با ۳۰ پیکسل تو رفتگی نمایش داده می شود  
<br>در حالی که بقیه سطرها به صورت عادی و  
بدون تو رفتگی نمایش داده می شوند  
</div>
```

این هم نتیجه:

سطر اول این قسمت با ۳۰ پیکسل تو رفتگی نمایش داده می شود.
در حالی که بقیه سطرها به صورت عادی و
بدون تو رفتگی نمایش داده می شوند.