

به نام خدا

برنامه نویسی مبتنی بر وب

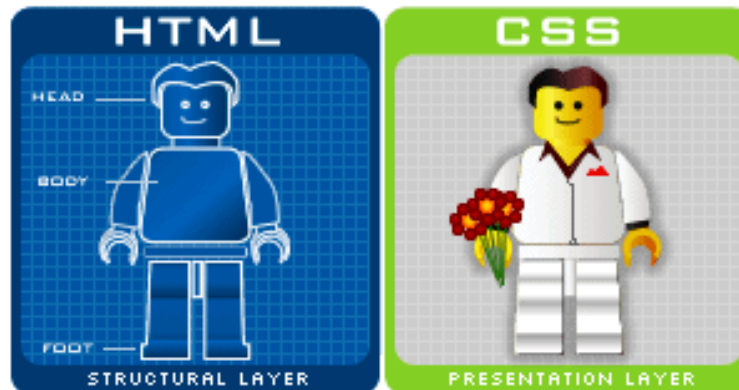
فصل دوم

CSS 3



CSS چیست؟

- واژه CSS از عبارت Cascading Style Sheets گرفته شده است که با استفاده از آن نوع متن، رنگ، نحوه چینش نوشته، تصاویر و ... را کنترل کرد. در واقع می توان گفت که با html می توان محتوای صفحه را ایجاد کرد و با CSS نحوه چگونگی نمایش این محتوا را تعیین کرد.



CSS 3

- CSS 3 آخرین نسخه از مستندات CSS است. قابلیت های CSS 3 عبارتند از پشتیبانی از گزینشگرهای (selectors) بیشتر، سایه انداختن زیر متن و عناصر، گوشه های گرد، تصاویر پس زمینه متعدد، انیمیشن، شفافیت و بسیاری از قابلیت های دیگر.

با CSS قادر خواهیم بود ...

- ظاهر صفحات را یکجا تعریف کنیم بدون اینکه نیاز به تکرار بخش های زیادی از سایت باشیم.
- می توان ظاهر صفحات را به طور یکجا تعریف کرد.
- می توانیم به آسانی ظاهر صفحات را پس از تغییرشان تغییر دهیم.
- قادر به تعیین اندازه های مختلف قلم خواهیم بود و دیگر به تعداد سائزهای کم html محدود نخواهیم بود.
- و ...

One HTML Page - Multiple Styles!

Welcome to My Homepage

Use the menu to select different Stylesheets

Stylesheet 1

Stylesheet 2

Stylesheet 3

Stylesheet 4

No Stylesheet

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links: [Stylesheet1](#), [Stylesheet2](#), [Stylesheet3](#), [Stylesheet4](#).

No Styles

Welcome to My Homepage

Use the menu to select different Stylesheets

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links: [Stylesheet1](#), [Stylesheet2](#), [Stylesheet3](#), [Stylesheet4](#).

No Styles

This page uses DIV elements to group different sections of the HTML page. Click here to see how the page looks like with no stylesheet: [No Stylesheet](#).

Stylesheet 1

Stylesheet 2

Stylesheet 3

Stylesheet 4

No Stylesheet

Welcome to My Homepage

Use the menu to select different Stylesheets

Stylesheet 1

Stylesheet 2

Stylesheet 3

Stylesheet 4

No Stylesheet

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links: [Stylesheet1](#), [Stylesheet2](#), [Stylesheet3](#), [Stylesheet4](#).

No Styles

Welcome to My Homepage

Use the menu to select different Stylesheets

- [Stylesheet 1](#)
- [Stylesheet 2](#)
- [Stylesheet 3](#)
- [Stylesheet 4](#)
- [No Stylesheet](#)

Side-Bar

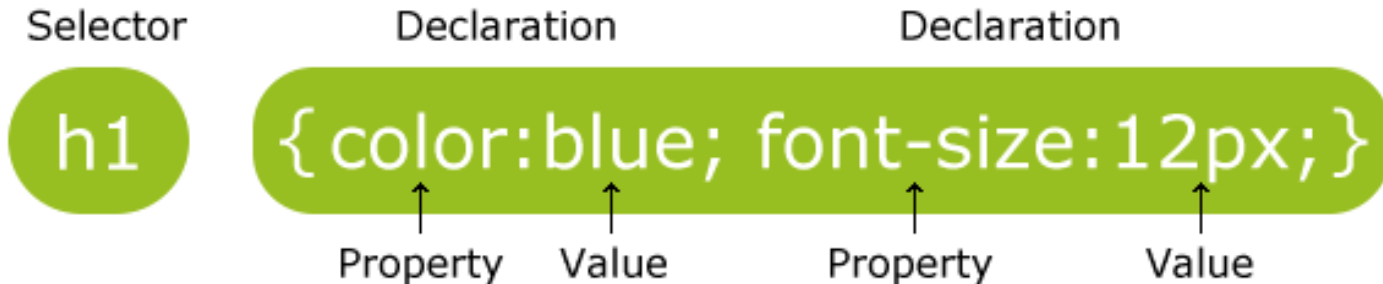
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links: [Stylesheet1](#), [Stylesheet2](#), [Stylesheet3](#), [Stylesheet4](#).

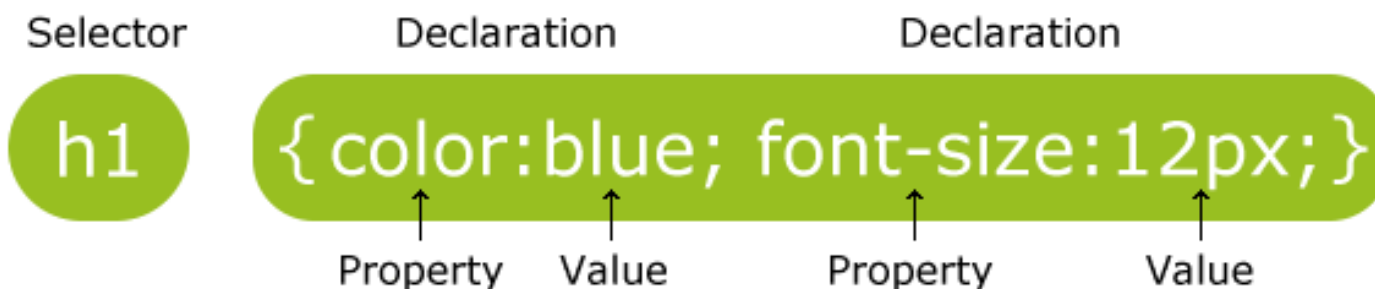
CSS Syntax

- یک قاعده CSS (rule set) شامل دو بخش انتخاب گر (Selector) و اعلان (Declaration) است.



انتخاب گر Selector

- برای انتخاب بخشی که می خواهیم CSS به آن اعمال شود از **Selector** ها استفاده می نمائیم.



انواع مختلف انتخابگرها

- Element Selector
- Id Selector
- Class Selector
- Grouping Selectors

انتخابگر بر اساس نام عنصر Element Selector

- در این روش المان های بر اساس نام انتخاب می شوند تا ویژگی های مورد نظر بر آنها اعمال شود.

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
    text-align: center;
    color: red;
}
</style>
</head>
<body>
```

Every paragraph will be affected by the style.

Me too!

And me!

```
<p>Every paragraph will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```



code 1.txt

انتخابگر بر اساس شناسه id Selector

- با استفاده از این نوع انتخابگر می توان یک نوع قالب بندی را بر روی فقط یکی از عناصر صفحه اعمال کرد. این نوع انتخابگر ها با علامت # (hash character) تعریف می شوند. Id حالت یکتا و منحصر به فرد دارد و باید به یک برچسب خاص اختصاص داده شود.

Note: An id name cannot start with a number!

انتخابگر بر اساس شناسه id Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>

</body>
</html>
```

Hello World!

This paragraph is not affected by the style.



code 2.txt

انتخابگر بر اساس کلاس Class Selector

- با استفاده از این نوع انتخابگر، یک کلاس را تعریف می کنیم و سپس در برچسب های مورد نظر، مشخصه کلاس را با نام این کلاسی که تعریف کرده ایم مقداردهی می کنیم. از نام کلاس می توان برای چندین تگ استفاده کرد. در واقع تگ هایی که نام کلاس مشترک می باشند، خصوصیت تعیین شده در CSS برا آنها اعمال می شود. جهت استفاده از نام کلاس در CSS باید از نقطه (.) استفاده کرد.

Note: A class name cannot start with a number!

انتخابگر بر اساس کلاس Class Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
.center {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<h1 class="center">Red and center-aligned heading</h1>
<p class="center">Red and center-aligned paragraph.</p>

</body>
</html>
```

Red and center-aligned heading

Red and center-aligned paragraph.



code 3.txt

نکته

- چنانچه بخواهیم کلاس را فقط به تگ خاصی منحصر کنیم، باید قبل از نام کلاس از سلکتور آن تگ نیز استفاده کنیم.

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-
aligned.</p>

</body>
</html>
```

This heading will not be affected

This paragraph will be red and center-aligned.



code 4.txt

نکته

- عناصر html می توانند بیش از یک کلاس داشته باشند.

Example

```
<p class="center large">This paragraph refers to two classes.</p>
```

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
    text-align: center;
    color: red;
}
```

```
p.large {
    font-size: 300%;
}
</style>
</head>
<body>
```

```
<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-aligned.</p>
<p class="center large">This paragraph will be red, center-aligned, and in a
large font-size.</p>

</body>
</html>
```

This heading will not be affected

This paragraph will be red and center-aligned.

This paragraph will be red,
center-aligned, and in a large
font-size.



code 5.txt

انتخابگرهای گروهی Grouping Selectors

- اگر عناصری با استایل یکسان به صورت زیر تعریف شده باشند می توان برای کمتر کردن کدها از انتخابگرهای گروهی استفاده کرد.

```
h1 {  
    text-align: center;  
    color: red;  
}
```

```
h2 {  
    text-align: center;  
    color: red;  
}
```

```
p {  
    text-align: center;  
    color: red;  
}
```

```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```



code 6.txt

توضیحات Comments

- برای گذاشتن توضیحات در CSS می توان از `/* comments */` استفاده کرد.

Example

```
p {  
    color: red;  
    /* This is a single-line comment */  
    text-align: center;  
}  
  
/* This is  
a multi-line  
comment */
```

روش های اعمال قالب بندی CSS به HTML

- تعبیه شده (Inline style)

✓ در این روش کدهای CSS درون تگ با مشخصه style تعریف می شود.

- داخلی (Internal style sheet)

✓ در این روش کدهای CSS در بخش تگ head قرار می گیرد.

- خارجی (External style sheet)

✓ در این روش کدهای CSS در یک فایل مجزا نوشته می شود و به صفحه html مورد

نظر معرفی می شود.

سبک دهی تعبیه شده Inline Styles

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

This is a heading

This is a paragraph.

```
<h1 style="color:blue;margin-left:30px;">This is a heading</h1>  
<p>This is a paragraph.</p>  
  
</body>  
</html>
```



code 7.txt

Tip: An inline style loses many of the advantages of a style sheet (by mixing content with presentation). Use this method sparingly.

سبک دهی داخلی Internal Style Sheet

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    background-color: linen;
}
h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
<body>

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

</body>
</html>
```



code &txt

سبک دهی خارجی External Style Sheet

- در این روش تنظیمات CSS درون یک فایل مجزا نگهداری می شود و سپس نشانی این فایل به صفحه ای که قرار است با استفاده از مشخصه های این فایل قالب بندی شود معرفی می گردد. مزیت این روش این است که فایل ایجاد شده می تواند برای همه صفحات وب سایت مورد استفاده قرار گیرد و به این ترتیب زمان طراحی کاهش می یابد. مزیت دیگر این روش، افزایش سرعت بارگذاری صفحات است به طوری که فایل CSS یک بار بر روی مرورگر دانلود می شود و بارها مورد استفاده قرار می گیرد.

سبک دهی خارجی External Style Sheet

- برای ارجاع دادن به فایل CSS نوشته شده باید کد زیر در تگ head نوشته شود.

Example

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

مثال

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

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

</body>
</html>
```

This is a heading

This is a paragraph.

```
body {
    background-color: lightblue;
}

h1 {
    color: navy;
    margin-left: 20px;
}
```

کد CSS

Note: Do not add a space between the property value and the unit (such as `margin-left: 20 px;`). The correct way is: `margin-left: 20px;`

ترتیب اولویت سبک ها Cascading Order

- اولویت اول: سبک های تعبیه شده
- اولویت دوم: سبک های داخلی
- اولویت سوم: سبک های خارجی
- اولویت چهارم: تنظیمات مرورگر

ترتيب اولويت سبك ها Cascading Order

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
<style>
body {background-color: linen;}
</style>
</head>
<body style="background-color: lightcyan">
```



code 9.txt

```
<h1>Multiple Styles Will Cascade into One</h1>
<p>In this example, the background color is set inline, in an internal
stylesheet, and in an external stylesheet.</p>
<p>Try experimenting by removing styles to see how the cascading stylesheets
work. (try removing the inline first, then the internal, then the external)</p>

</body>
</html>
```

Multiple Styles Will Cascade into One

In this example, the background color is set inline, in an internal stylesheet, and in an external stylesheet.

Try experimenting by removing styles to see how the cascading stylesheets work. (try removing the inline first, then the internal, then the external)

رنگ ها در CSS

- Color Names
- Color Values
 - ✓ RGB Value
 - ✓ HEX Value
 - ✓ HSL Value
 - ✓ RGBA Value
 - ✓ HSLA Value

انتخاب رنگ بر اساس نام

- می توان برای تعیین رنگ از نام رنگ استفاده کرد. HTML از ۱۴۰ نام رنگ مختلف پشتیبانی می کند.



https://www.w3schools.com/colors/colors_names.asp

انتخاب رنگ بر اساس نام Color Names

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

<h1 style="background-color:Tomato;">Tomato</h1>
<h1 style="background-color:Orange;">Orange</h1>
<h1 style="background-color:DodgerBlue;">DodgerBlue</h1>
<h1 style="background-color:MediumSeaGreen;">MediumSeaGreen</h1>
<h1 style="background-color:Gray;">Gray</h1>
<h1 style="background-color:SlateBlue;">SlateBlue</h1>
<h1 style="background-color:Violet;">Violet</h1>
<h1 style="background-color:LightGray;">LightGray</h1>

</body>
</html>
```

Tomato

Orange

DodgerBlue

MediumSeaGreen

Gray

SlateBlue

Violet

LightGray

رنگ متن ، رنگ پس زمینه و کادر دور متن

- Background Color

Example

```
<h1 style="background-color:DodgerBlue;">Hello World</h1>  
<p style="background-color:Tomato;">Lorem ipsum...</p>
```

Hello World

ipiscing elit, sed diam nonum
d minim veniam, quis nostrud

- Text Color

Example

```
<h1 style="color:Tomato;">Hello World</h1>  
<p style="color:DodgerBlue;">Lorem ipsum...</p>  
<p style="color:MediumSeaGreen;">Ut wisi enim...</p>
```

Hello World

Lorem ipsum dolor sit amet, co
magna aliquam erat volutpat.

Ut wisi enim ad minim veniam,
consequat.

- Border Color

Example

```
<h1 style="border:2px solid Tomato;">Hello World</h1>  
<h1 style="border:2px solid DodgerBlue;">Hello World</h1>  
<h1 style="border:2px solid Violet;">Hello World</h1>
```

Hello World

Hello World

Hello World

انتخاب رنگ بر اساس مقادیر Color Values

- رنگ ها را می توان بر اساس مقادیر آنها بر اساس یکی از مدل های **RGB**, **HEX**, **HSL**, **HSLA** نیز می توان انتخاب کرد.

rgb(255, 99, 71)

#ff6347

hsl(9, 100%, 64%)

Example

```
<h1 style="background-color:rgb(255, 99, 71);">...</h1>  
<h1 style="background-color:#ff6347;">...</h1>  
<h1 style="background-color:hsl(9, 100%, 64%);">...</h1>
```

نمونه های RGB

`rgb(255, 0, 0)`

`rgb(0, 0, 255)`

`rgb(60, 179, 113)`

`rgb(238, 130, 238)`

`rgb(255, 165, 0)`

`rgb(106, 90, 205)`

`rgb(0, 0, 0)`

`rgb(60, 60, 60)`

`rgb(120, 120, 120)`

`rgb(180, 180, 180)`

`rgb(240, 240, 240)`

`rgb(255, 255, 255)`

CSS Backgrounds

• خاصیت **Backgrounds** برای تعریف پس زمینه عناصر استفاده می شود.

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

Background Color

- خاصیت `background-color` تعیین کننده رنگ پس زمینه عنصر می باشد.

Example

```
body {  
    background-color: lightblue;  
}
```

Example

```
h1 {  
    background-color: green;  
}  
  
div {  
    background-color: lightblue;  
}  
  
p {  
    background-color: yellow;  
}
```

CSS background-color example!

This is a text inside a div element.

This paragraph has its own background color.

We are still in the div element.



code 10.txt

Background Image

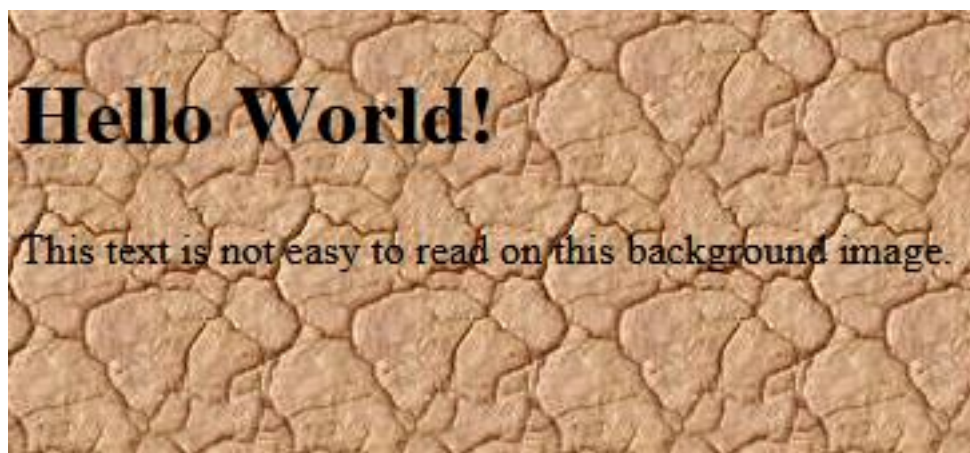
- خاصیت `background-image` تعیین کننده یک تصویر به عنوان پس زمینه عنصر می باشد. به صورت پیش فرض تصویر مورد نظر تکرار می شود تا کل عنصر را پوشش دهد.

Example

```
body {  
    background-image: url("paper.gif");  
}
```

توجه

- هنگامی که از عکس پس زمینه استفاده می کنید، تصویری را انتخاب نمایید که باعث ایجاد مزاحمت در خواندن متن نشود.



Example of a **bad** combination of text and background image

Background Image - Repeat Horizontally or Vertically

- به طور پیش فرض خاصیت background-image تصویر را به هر دو صورت افقی و عمودی تکرار می کند. بعضی تصاویر باید برای بهتر دیده شدن تنها به یکی از دو صورت افقی و عمودی تکرار شوند.

Example

```
body {  
    background-image: url("gradient_bg.png");  
}
```

Hello World!

Strange background image...

Example

```
body {  
    background-image: url("gradient_bg.png");  
    background-repeat: repeat-x;  
}
```

Hello World!

Here, a background image is repeated only horizontally!

توجه

```
<style>
body {
  background-image: url("gradient_bg.png");
  background-repeat: repeat-y;
}
</style>
```

- تکرار عمودی (vertically)

background-repeat: repeat-y; ✓

```
<style>
body {
  background-image: url("gradient_bg.png");
  background-repeat: repeat-x;
}
</style>
```

- تکرار افقی (Horizontally)

background-repeat: repeat-x; ✓

```
<style>
body {
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
}
</style>
```

- عدم تکرار تصویر

background-repeat: no-repeat; ✓

Background Image - no-repeat

- برای عدم تکرار عکس پس زمینه و تنها یک بار استفاده آن ویژگی **background-repeat** را باید تعیین کرد.

Example

```
body {  
    background-image: url("img_tree.png");  
    background-repeat: no-repeat;  
}
```



```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    background-image: url("img_tree.png");
    background-repeat: no-repeat;
}
</style>
</head>
<body>

<h1>Hello World!</h1>
<p>W3Schools background image example.</p>
<p>The background image is only showing once, but it is disturbing the
reader!</p>

</body>
</html>
```



code 11.txt



Hello World!

W3Schools background image example.

The background image is only showing once, but it is disturbing the reader!

Background Image - Set position

- در مثال قبل عکس پس زمینه در یک مکان مانند متن قرار می گیرد. برای تغییر موقعیت عکس پس زمینه از خاصیت **background-position** می توان استفاده کرد.

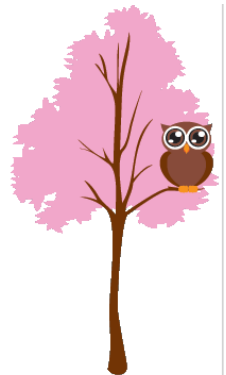
```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    background-image: url("img_tree.png");
    background-repeat: no-repeat;
    background-position: right top;
    margin-right: 200px;
}
</style>
</head>
```

Hello World!

W3Schools background no-repeat, set position example.

Now the background image is only shown once, and positioned away from the text.

In this example we have also added a margin on the right side, so the background image will never disturb the text.



code 12.txt

Background Image - Fixed position

- برای ثابت ماندن عکس پس زمینه و اسکرول نشدن می توان از خاصیت **background-attachment** استفاده کرد.

Example

```
body {  
    background-image: url("img_tree.png");  
    background-repeat: no-repeat;  
    background-position: right top;  
    background-attachment: fixed;  
}
```

Hello World!

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.

The background-image is fixed. Try to scroll down the page.



حالت خلاصه نویسی

The shorthand property for background is `background` :

Example

```
body {  
    background: #ffffff url("img_tree.png") no-repeat right top;  
}
```

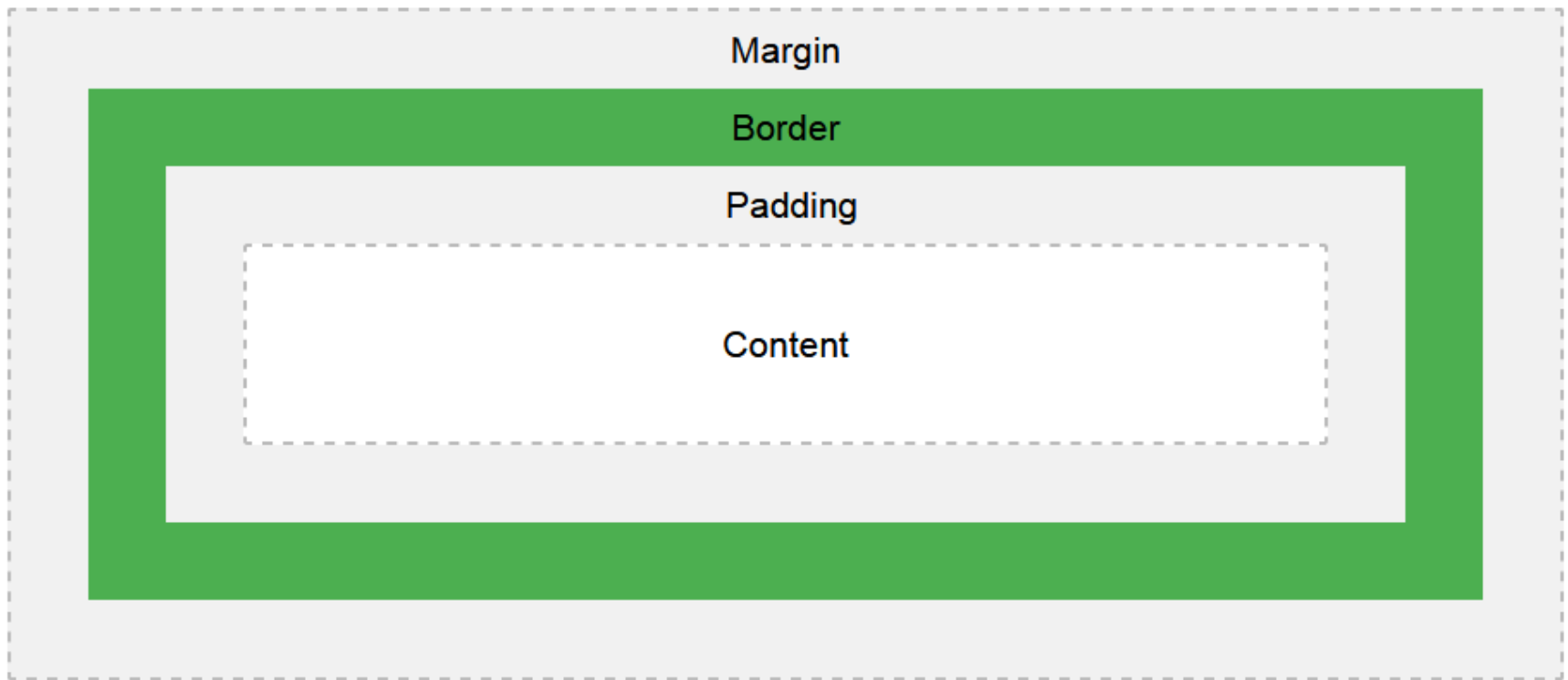
Hello World!

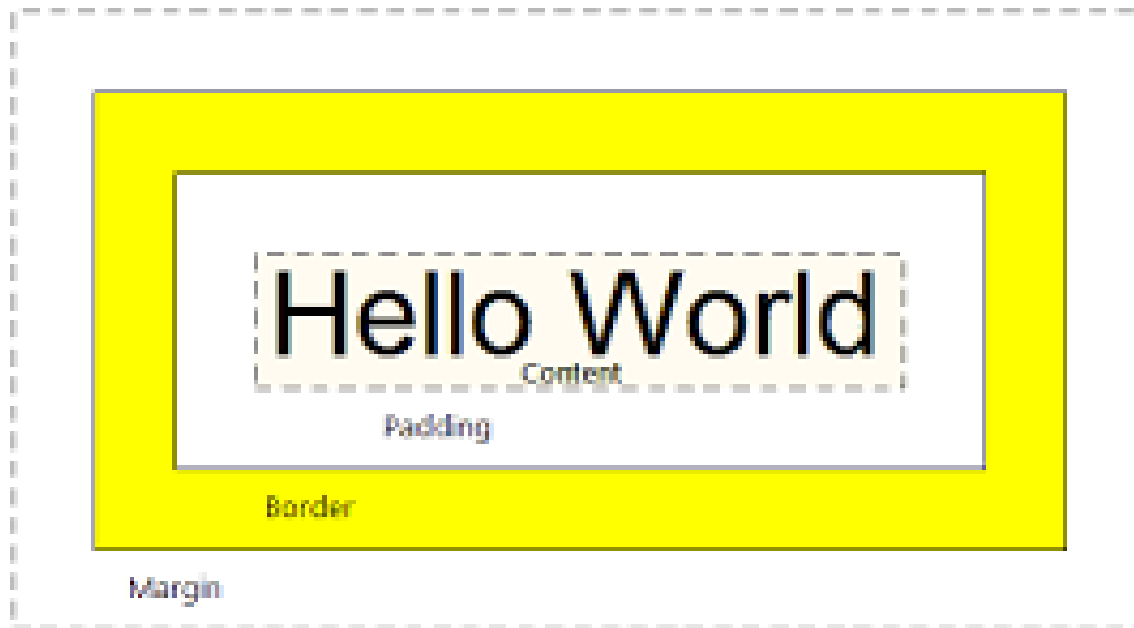
Now the background image is only shown once, and it is also positioned away from the text.

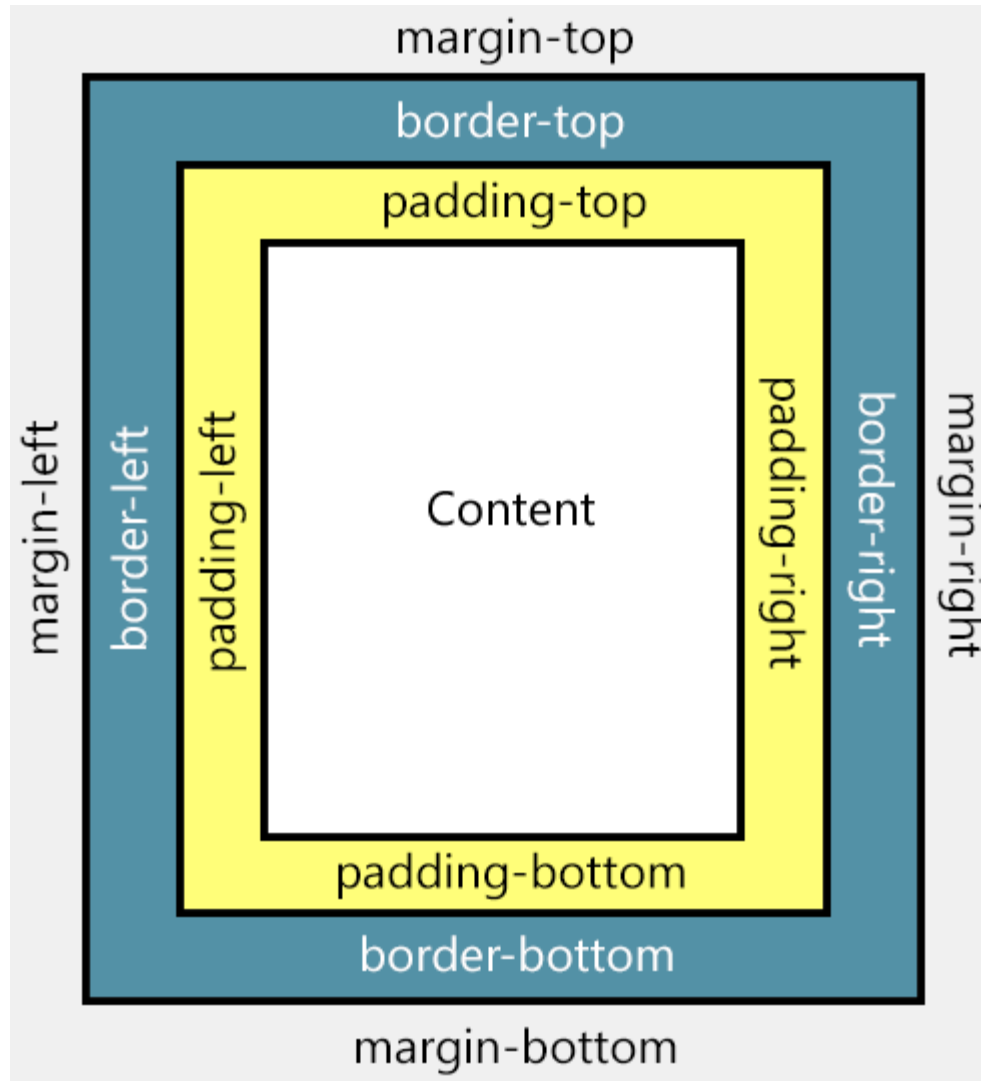
In this example we have also added a margin on the right side, so that the background image will not disturb the text.



Padding & Margin , Border







CSS Border Properties

- خاصیت border برای تعیین طرح، پهنا و رنگ کادر المان مربوطه بکار می رود.

I have borders on all sides.

I have a red bottom border.

I have rounded borders.

I have a blue left border.

CSS Border Properties

• خصوصیت **border-style** برای نوع نمایش کادر بکار می رود. که می تواند مقادر زیر را اختیار کند.

- **dotted** - Defines a dotted border
- **dashed** - Defines a dashed border
- **solid** - Defines a solid border
- **double** - Defines a double border
- **groove** - Defines a 3D grooved border. The effect depends on the border-color value
- **ridge** - Defines a 3D ridged border. The effect depends on the border-color value
- **inset** - Defines a 3D inset border. The effect depends on the border-color value
- **outset** - Defines a 3D outset border. The effect depends on the border-color value
- **none** - Defines no border
- **hidden** - Defines a hidden border

CSS Border Properties

• همچنین خاصیت **border-style** می تواند از یک تا چهار مقدار داشته باشد:

top border ✓

right border ✓

bottom border ✓

left border ✓

Example

```
p.dotted {border-style: dotted;}
p.dashed {border-style: dashed;}
p.solid {border-style: solid;}
p.double {border-style: double;}
p.groove {border-style: groove;}
p.ridge {border-style: ridge;}
p.inset {border-style: inset;}
p.outset {border-style: outset;}
p.none {border-style: none;}
p.hidden {border-style: hidden;}
p.mix {border-style: dotted dashed solid double;}
```



code 13.txt

Result:

A dotted border.

A dashed border.

A solid border.

A double border.

A groove border. The effect depends on the border-color value.

A ridge border. The effect depends on the border-color value.

An inset border. The effect depends on the border-color value.

An outset border. The effect depends on the border-color value.

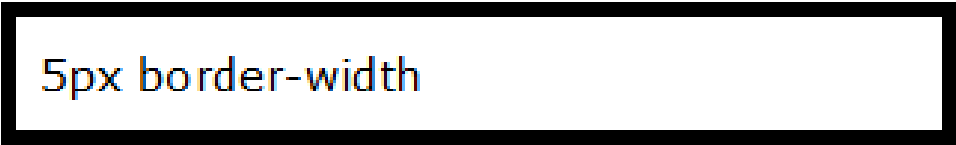
No border.

A hidden border.

A mixed border.

Border Width

- خصوصیت border-width تعیین کننده پهنای کادر دور المان است. این خصوصیت می تواند با واحد px, pt, cm, em, ... تنظیم شود و یا با مقادیر thin, medium, thick تنظیم شود.



5px border-width

مثال

```
<style>
p.one {
  border-style: solid;
  border-width: 5px;
}

p.two {
  border-style: solid;
  border-width: medium;
}

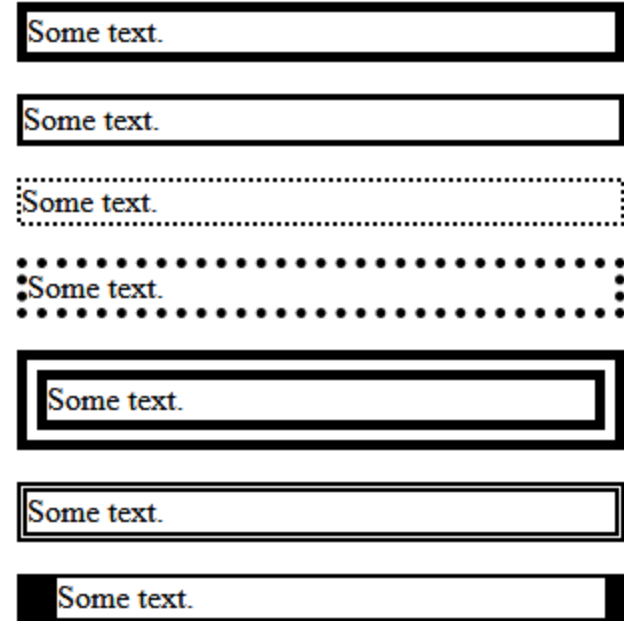
p.three {
  border-style: dotted;
  border-width: 2px;
}

p.four {
  border-style: dotted;
  border-width: thick;
}

p.five {
  border-style: double;
  border-width: 15px;
}

p.six {
  border-style: double;
  border-width: thick;
}

p.seven {
  border-style: solid;
  border-width: 2px 10px 4px 20px;
}
</style>
```



code 14.txt

Border Color

• خاصیت border-color برای تعیین رنگ کادر بکار می رود که می تواند مقادیر زیر را اختیار کند.

- name - specify a color name, like "red"
- Hex - specify a hex value, like "#ff0000"
- RGB - specify a RGB value, like "rgb(255,0,0)"
- transparent

مثال

```
<style>
p.one {
  border-style: solid;
  border-color: red;
}

p.two {
  border-style: solid;
  border-color: green;
}

p.three {
  border-style: solid;
  border-color: red green blue yellow;
}
</style>
```

A solid red border

A solid green border

A solid multicolor border



code 15.txt

Border - Individual Sides

Example

```
p {  
  border-top-style: dotted;  
  border-right-style: solid;  
  border-bottom-style: dotted;  
  border-left-style: solid;  
}
```

Different Border Styles

The example above gives the same result as this:

Example

```
p {  
  border-style: dotted solid;  
}
```



code 16.txt

If the `border-style` property has four values:

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

If the `border-style` property has three values:

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

If the `border-style` property has two values:

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

If the `border-style` property has one value:

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

Border - Shorthand Property

- می توان به منظور کوتاه نویسی کدها، همه خصوصیت های border را در یک خصوصیت نوشت.

Example

```
p {  
    border: 5px solid red;  
}
```

Result:



Some text

نکته

- همچنین می توان مشخصات تنها یک کادر از چهار کادر دور المان را تعیین کرد.

Left Border

```
p {  
  border-left: 6px solid red;  
  background-color: lightgrey;  
}
```

Result:

Some text

Bottom Border

```
p {  
  border-bottom: 6px solid red;  
  background-color: lightgrey;  
}
```

Result:

Some text

Rounded Borders

- با خاصیت `border-radius` می توان کادردورالمان را گرد کرد.

Normal border

Round border

Rounder border

Roudest border

Note: The `border-radius` property is not supported in IE8 and earlier versions.

```
<style>
p.normal {
    border: 2px solid red;
}

p.round1 {
    border: 2px solid red;
    border-radius: 5px;
}

p.round2 {
    border: 2px solid red;
    border-radius: 8px;
}

p.round3 {
    border: 2px solid red;
    border-radius: 12px;
}
</style>
```

The border-radius Property

This property is used to add rounded borders to an element:

Normal border

Round border

Rounder border

Roudest border

Note: The "border-radius" property is not supported in IE8 and earlier versions.



code 17.txt

حاشیه ها CSS Margins

- خاصیت **margin** برای ایجاد فضا اطراف عنصر بکار می رود. برای هر یک از چهار طرف عنصر خاصیت margin وجود دارد (top, right, bottom, left).

- `margin-top`
- `margin-right`
- `margin-bottom`
- `margin-left`

values:

- `auto` - the browser calculates the margin
- `length` - specifies a margin in px, pt, cm, etc.
- `%` - specifies a margin in % of the width of the containing element
- `inherit` - specifies that the margin should be inherited from the parent element

Example

```
p {  
    margin-top: 100px;  
    margin-bottom: 100px;  
    margin-right: 150px;  
    margin-left: 80px;  
}
```

Using individual margin properties

This div element has a top margin of 100px, a right margin of 150px, a bottom margin of 100px, and a left margin of 80px.

حالت خلاصه:

Example

```
p {  
    margin: 25px 50px 75px 100px;  
}
```



code 18.txt

حالات مختلف خلاصه نویسی

If the **margin** property has four values:

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

If the **margin** property has three values:

- **margin: 25px 50px 75px;**
 - top margin is 25px
 - right and left margins are 50px
 - bottom margin is 75px

If the **margin** property has two values:

- **margin: 25px 50px;**
 - top and bottom margins are 25px
 - right and left margins are 50px

If the **margin** property has one value:

- **margin: 25px;**
 - all four margins are 25px

مقدار auto

- می توان مقدار auto را به خاصیت margin نسبت داد تا عنصر را به طور افقی در مرکز المانی که در داخلش قرار گرفته، قرار دهد.

Example

```
div {  
  width: 300px;  
  margin: auto;  
  border: 1px solid red;  
}
```

Use of margin:auto

You can set the margin property to auto to horizontally center the element within its container. The element will then take up the specified width, and the remaining space will be split equally between the left and right margins:

This div will be horizontally centered because it has margin: auto;



code 19.txt

مقدار inherit

This example lets the left margin of the `<p class="ex1">` element be inherited from the parent element (`<div>`):

Example

```
div {  
  border: 1px solid red;  
  margin-left: 100px;  
}  
  
p.ex1 {  
  margin-left: inherit;  
}
```



code 20.txt

Margin Collapse

- گاهی مواقع حاشیه های بالا و پایین عناصر تبدیل به یک حاشیه که برابر بزرگترین مقدار دو است، می شوند. توجه شود که این اتفاق برای حاشیه های چپ و راست نمی افتد.

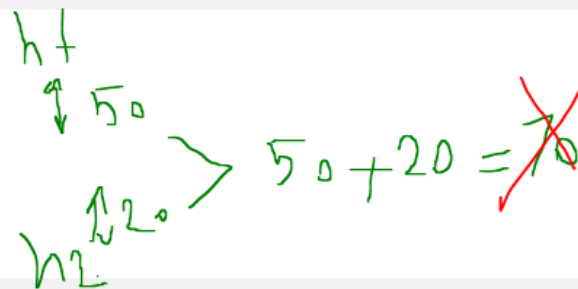
Look at the following example:

Example

```
h1 {  
  margin: 0 0 50px 0;  
}
```

```
h2 {  
  margin: 20px 0 0 0;  
}
```

Try it Yourself »



In the example above, the `<h1>` element has a bottom margin of 50px and the `<h2>` element has a top margin set to 20px.

Common sense would seem to suggest that the vertical margin between the `<h1>` and the `<h2>` would be a total of 70px (50px + 20px). But due to margin collapse, the actual margin ends up being 50px.

مثال

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {
    margin: 0 0 50px 0;
}

h2 {
    margin: 20px 0 0 0;
}
</style>
</head>
<body>
```

`<p>`In this example the h1 element has a bottom margin of 50px and the h2 element has a top margin of 20px. Then, the vertical margin between h1 and h2 should have been 70px (50px + 20px). However, due to margin collapse, the actual margin ends up being 50px.`</p>`

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>

</body>
</html>
```

In this example the h1 element has a bottom margin of 50px. Then, the vertical margin between h1 and h2, due to margin collapse, the actual margin is 50px.

Heading 1

Heading 2



code 21.txt

margin خاصیت خلاصہ

All CSS Margin Properties

Property	Description
<u>margin</u>	A shorthand property for setting the margin properties in one declaration
<u>margin-bottom</u>	Sets the bottom margin of an element
<u>margin-left</u>	Sets the left margin of an element
<u>margin-right</u>	Sets the right margin of an element
<u>margin-top</u>	Sets the top margin of an element

CSS Padding

• به فاصله محتوای عنصر تا لبه المان عنصر را Padding گویند.

- `padding-top`
- `padding-right`
- `padding-bottom`
- `padding-left`

All the padding properties can have the following values:

- *length* - specifies a padding in px, pt, cm, etc.
- % - specifies a padding in % of the width of the containing element
- inherit - specifies that the padding should be inherited from the parent element

Note: Negative values are not allowed.

مثال

```
<style>
div {
    border: 1px solid black;
    background-color: lightblue;
    padding-top: 50px;
    padding-right: 30px;
    padding-bottom: 50px;
    padding-left: 80px;
}
</style>
```



code 22.txt

This div element has a top padding of 50px, a right padding of 30px, a bottom padding of 50px, and a left padding of 80px.

```
<body>
```

```
<h2>Padding and element width</h2>
```

```
<div class="ex1">This div is 300px wide.</div>
```

```
<br>
```

```
<div class="ex2">The width of this div is 350px,  
300px in the CSS.</div>
```

```
</body>
```

This div is 300px wide.

The width of this div is 350px, even though it is defined as 300px in the CSS.

Example

```
div {  
  width: 300px;  
  padding: 25px;  
}
```

350px (300px + 25px of left padding + 25px of right padding):

To keep the width at 300px, no matter the amount of padding, you can use the `box-sizing` property.

```
<body>

<h2>Padding and element width</h2>

<div class="ex1">This div is 300px wide.</div>
<br>

<div class="ex2">The width of this div remains at 300px,
of total left and right padding, because of the box-sizing
property.
</div>

</body>
```

```
<style>
div.ex1 {
    width: 300px;
    background-color: yellow;
}

div.ex2 {
    width: 300px;
    padding: 25px;
    box-sizing: border-box;
    background-color: lightblue;
}
</style>
```

This div is 300px wide.

The width of this div remains at 300px, in spite of the 50px of total left and right padding, because of the `box-sizing: border-box` property.

width و height

- خاصیت width و height برای تنظیم ارتفاع و پهنای عنصر بکار می رود. این خاصیت ها می توانند به حالت خودکار تنظیم شوند که در این حالت مرورگر ابعاد عنصر را تعیین می کند (حالت پیش فرض) یا با مقادیری نظیر px, cm,... و یا به صورت درصدی مقدار دهی شوند.

Example

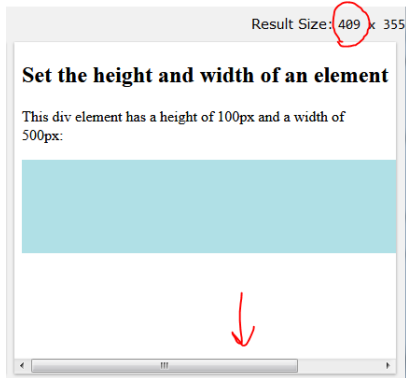
```
div {  
  height: 200px;  
  width: 50%;  
  background-color: powderblue;  
}
```

This element has a height of 200 pixels and a width of 50%

Note: The `height` and `width` properties do not include padding, borders, or margins; they set the height/width of the area inside the padding, border, and margin of the element!

خصوصیت max-width

- با استفاده از خصوصیت max-width می توان حداکثر عرض یک عنصر را مشخص کرد. تنظیم این خصوصیت باعث می شود مقدار ارتفاع از max-width بیشتر نشود.
- توجه: مقدار خصوصیت max-width مقدار خصوصیت width را لغو می کند.



The problem with the `<div>` above occurs when the browser window is smaller than the width of the element (500px). The browser then adds a horizontal scrollbar to the page.

Using `max-width` instead, in this situation, will improve the browser's handling of small windows.

مثال



code 23.txt

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
    height: 100px;
    background-color: powderblue;
}
</style>
</head>
<body>

<h2>Set the max-width of an element</h2>
<p>This div element has a height of 100px and a max-width of 500px:</p>

<div></div>

<p>Resize the browser window to see the effect.</p>

</body>
</html>
```

Set the max-width of an element

This div element has a height of 100px and a max-width of 500px:



Resize the browser window to see the effect.

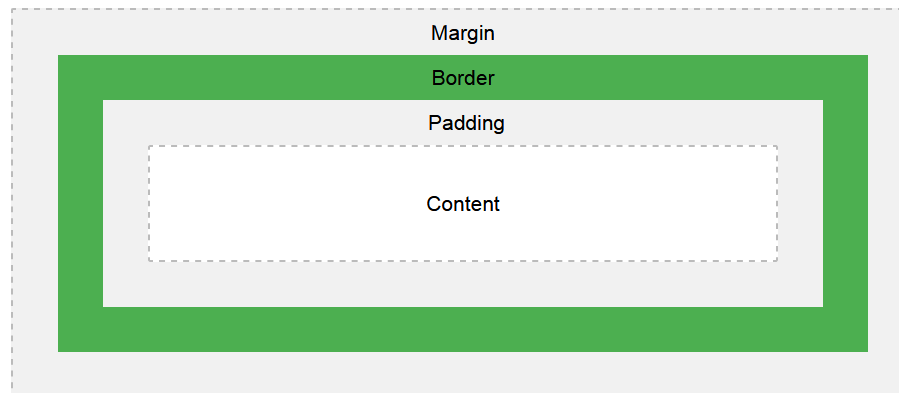
CSS Box Model

- تمامی عناصر HTML در CSS به صورت یک کادر (جعبه) چهار گوش تصور می شوند. مدل کادری، کادری است که در اطراف عناصر صفحه می تواند قرار بگیرد و شامل ویژگی های margin، border، padding و content میباشد. برای تنظیم عرض و ارتفاع حقیقی یک عنصر درک مفهوم مدل کادری بسیار مهم است.

CSS Box Model

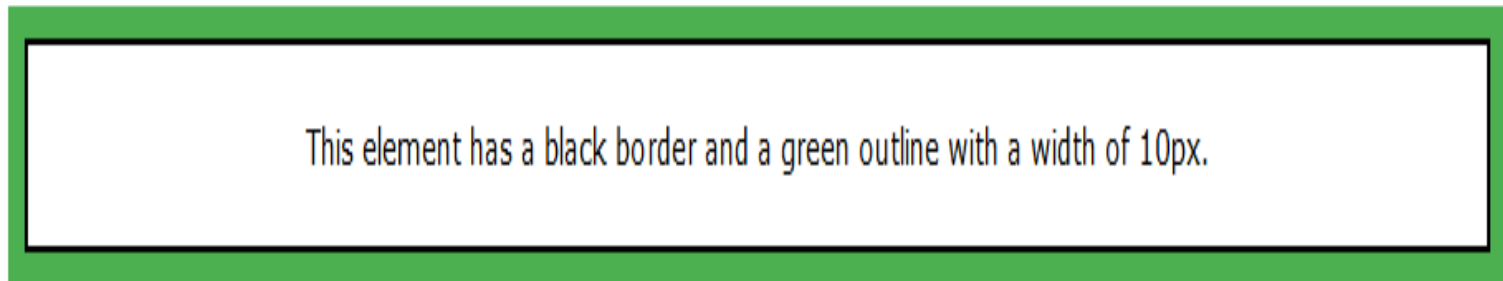
- **margin**: فضای اطراف border را پوشش می دهد . این قسمت نمی تواند رنگ پس زمینه داشته باشد و کاملاً شفاف است.
- **border**: این بخش فضای اطراف content و padding را پوشش می دهد و می تواند **رنگ** ، **سبک** و **ضخامت** خاص خود را داشته باشد .
- **padding**: فضای اطراف content را پوشش می دهد . این بخش می تواند تحت تاثیر رنگ پس زمینه عنصر قرار بگیرد .
- **content**: محتوای عنصر را در بر می گیرد . همان جایی که متن و عکس قرار می گیرد .

❖ برای تنظیم عرض و ارتفاع حقیقی یک عنصر درک مفهوم مدل کادری بسیار مهم است.



Outline

- outline خطی است که در اطراف عنصر بعد از border رسم می شود است که برای برجسته کردن عنصر بکار می رود.



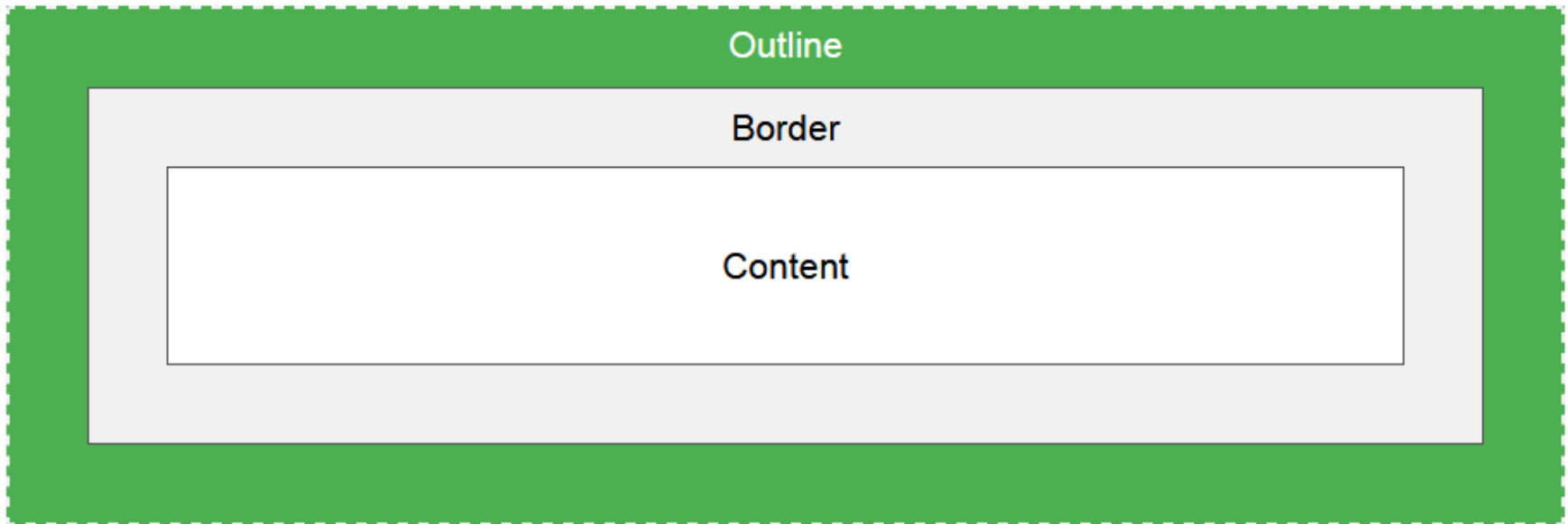
CSS has the following outline properties:

- `outline-style`
- `outline-color`
- `outline-width`
- `outline-offset`
- `outline`

Note: None of the other outline properties will have any effect, unless the `outline-style` property is set!

CSS Outline

An outline is a line that is drawn around elements, OUTSIDE the borders, to make the element "stand out".



مقادير خاصيت outline-style

- **dotted** - Defines a dotted outline
- **dashed** - Defines a dashed outline
- **solid** - Defines a solid outline
- **double** - Defines a double outline
- **groove** - Defines a 3D grooved outline
- **ridge** - Defines a 3D ridged outline
- **inset** - Defines a 3D inset outline
- **outset** - Defines a 3D outset outline
- **none** - Defines no outline
- **hidden** - Defines a hidden outline



code 24.txt

A dotted outline.

A dashed outline.

A solid outline.

A double outline.

A groove outline. The effect depends on the outline-color value.

A ridge outline. The effect depends on the outline-color value.

An inset outline. The effect depends on the outline-color value.

An outset outline. The effect depends on the outline-color value.

CSS Text

- Text Color •
- Text Alignment •
- Text Decoration •
- Text Transformation •
- Text Indentation •
- Line Height •
- Text Direction •
- Word Spacing •
- Text Shadow •

رنگ متن

- خاصیت **color** برای تعیین رنگ متن بکار می رود که می تواند مقادیر زیر را داشته باشد.

- a color name - like "red"
- a HEX value - like "#ff0000"
- an RGB value - like "rgb(255,0,0)"

Example

```
body {  
    color: blue;  
}  
  
h1 {  
    color: green;  
}
```

This is heading 1

This is an ordinary paragraph. Notice that this t
defined in the body selector.



code 25.txt

ترازبندی متن

- خاصیت **text-align** برای ترازبندی افقی متن بکار می رود. این خاصیت می تواند مقادیر **left**, **right**, **centered**, **justified** را به خود بگیرد.

Example

```
h1 {  
    text-align: center;  
}  
  
h2 {  
    text-align: left;  
}  
  
h3 {  
    text-align: right;  
}
```

Heading 1 (center)

Heading 2 (left)

Heading 3 (right)



code 26.txt

تعیین فرم های خاص برای نمایش متن

- خاصیت **text-decoration** برای از بین بردن decoration متن بکار می رود. مقدار **text-decoration: none;** اغلب برای خارج کردن خط زیر لینک ها بکار می رود.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
a {
```

```
    text-decoration: none;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<p>A link with no underline: <a
```

```
href="https://www.w3schools.com">W3Schools.com</a></p>
```

```
</body>
```

```
</html>
```

A link with no underline: [W3Schools.com](https://www.w3schools.com)



code 27.txt

Example

```
h1 {  
    text-decoration: overline;  
}  
  
h2 {  
    text-decoration: line-through;  
}  
  
h3 {  
    text-decoration: underline;  
}
```

This is heading 1

~~This is heading 2~~

This is heading 3



code 2&txt

Note: It is not recommended to underline text that is not a link, as this often confuses the reader.

دگرگونی متن

- خاصیت text-transform برای تعیین کوچکی یا بزرگی حروف بکار می رود.

Example

```
p.uppercase {  
    text-transform: uppercase;  
}  
  
p.lowercase {  
    text-transform: lowercase;  
}  
  
p.capitalize {  
    text-transform: capitalize;  
}
```

THIS IS SOME TEXT.

this is some text.

This Is Some Text.



code 29.txt

تورفتگی متن

- خاصیت text-indent برای تعیین تورفتگی در خط اول متن بکار می رود.

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
    text-indent: 50px;
}
</style>
</head>
<body>
```

In my younger and more vulnerable years my father gave me some advice that I've been turning over in my mind ever since. 'Whenever you feel like criticizing anyone,' he told me, 'just remember that all the people in this world haven't had the advantages that you've had.'

```
<p>In my younger and more vulnerable years my father gave me some advice that
I've been turning over in my mind ever since. 'Whenever you feel like criticizing
anyone,' he told me, 'just remember that all the people in this world haven't had
the advantages that you've had.'</p>
```

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



code 30.txt

فاصله بین حروف

- خاصیت **letter-spacing** برای تعیین فاصله بین کاراکترها در متن بکار می رود.

Example

```
h1 {  
    letter-spacing: 3px;  
}  
  
h2 {  
    letter-spacing: -3px;  
}
```

This is heading 1

Thisisheading2



code 31.txt

فاصله خطوط

- خاصیت **line-height** برای تعیین فاصله بین خطوط بکار می رود.

Example

```
p.small {  
    line-height: 0.8;  
}  
  
p.big {  
    line-height: 1.8;  
}
```

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 smaller line-height.
This is a paragraph with a smaller line-height.

This is a paragraph with a bigger line-height.

This is a paragraph with a bigger line-height.



code 32.txt

جهت نوشتاری متن

- خاصیت **direction** برای تغییر جهت نوشتاری عنصر مورد نظر بکار می رود.

Example

```
p {  
    direction: rtl;  
}
```

This is the default text direction.

.noitcerid txet tfel-ot-thgir si sihT



code 33.txt

فاصله کلمات

- خاصیت **word-spacing** برای تعیین فاصله بین کلمات بکار می رود.

Example

```
h1 {  
    word-spacing: 10px;  
}  
  
h2 {  
    word-spacing: -5px;  
}
```

This is heading 1

Thisisheading2



code 34.txt

سایه دادن به متن

- خاصیت **text-shadow** برای دادن سایه به متن بکار می رود.

The following example specifies the position of the horizontal shadow (3px), the position of the vertical shadow (2px) and the color of the shadow (red):

Example

```
h1 {  
    text-shadow: 3px 2px red;  
}
```



code 35.txt

Text-shadow effect

CSS Fonts

- صفات فونت در CSS، خانواده فونت، ضخامت، اندازه و استایل آن را تعریف می کنند.

Difference Between Serif and Sans-serif Fonts



Sans-serif



Serif



Serif
(red serifs)

Font Families

- در CSS دو نوع از نام های خانواده **فونت** ها موجود است :
 - ✓ خانواده ژنریک (generic family): گروهی از خانواده های فونت با ظاهری مشابه (مانند : Serif یا Monospace)
 - ✓ خانواده فونت (font family): یک خانواده فونت خاص (مانند Times New Roman یا Arial)

Generic family	Font family	Description
Serif	Times New Roman Georgia	Serif fonts have small lines at the ends on some characters
Sans-serif	Arial Verdana	"Sans" means without - these fonts do not have the lines at the ends of characters
Monospace	Courier New Lucida Console	All monospace characters have the same width

Note: On computer screens, sans-serif fonts are considered easier to read than serif fonts.

خاصیت font-family

- خاصیت **font-family** برای تعیین نوع فونت بکار می رود. این صفت چندین نام فونت را به عنوان سیستم پشتیبان در دسترس قرار می دهد. اگر مرورگر اولین فونت را نمایش ندهد ، فونت بعدی را امتحان می کند و اگر دومی را نتواند ، سومی را و ...

Note: If the name of a font family is more than one word, it must be in quotation marks, like: "Times New Roman".

More than one font family is specified in a comma-separated list:

Example

```
p {  
    font-family: "Times New Roman", Times, serif;  
}
```



code 36.txt

نکته

- در تخصیص نام فونت، ابتدا از فونتهایی که خودتان مد نظر دارید استفاده کنید و در انتها همیشه از فونت های مشابه استفاده کنید، در این صورت مرورگر، اگر فونت های مد نظر شما را در دسترس نداشت حتما از فونت های مشابه که در تخصیص ذکر شده است، استفاده می کند. مقادیر را با کاما از همدیگر جدا نمایید. اگر نام فونت حاوی خط فاصله بود، حتما باید در کوتیشن نوشته شود. تک کوتیشن هنگام استفاده از خاصیت `style` در تگ های `html` ضروری است.

Font Style

- خاصیت **font-style** مقادیر زیر را به خود میگیرد.
- normal - The text is shown normally
- italic - The text is shown in italics
- oblique - The text is "leaning" (oblique is very similar to italic, but less supported)

Example

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

This is a paragraph in normal style.

This is a paragraph in italic style.

This is a paragraph in oblique style.



code 37.txt

Font Size

- برای تعریف اندازه متن از صفت **font-size** استفاده می کنیم. صفت `font-size` می تواند یک مقدار **مطلق** یا **نسبی** داشته باشد.

❖ مقدار مطلق :

متن را در اندازه ای مشخص نمایش می دهد.
به کاربر امکان تغییر اندازه متن در همه مرورگرها را نمی دهد. (این ویژگی از نظر فاکتورهای دسترسی ، مناسب نیست)
مقدار مطلق زمانی قابل مفید است که اندازه فیزیکی خروجی ، مشخص است.

❖ مقدار نسبی :

اندازه متن را نسبت به عناصر اطراف متن تنظیم می کند.
به کاربر امکان تغییر اندازه متن در مرورگرها را می دهد.

➤ **نکته :** اگر اندازه فونت را انتخاب نکنید ، اندازه پیش فرض برای متن های عادی مانند پاراگراف 16px ($16\text{px}=1\text{em}$) است.

Set Font Size With Pixels

Example

```
h1 {  
    font-size: 40px;  
}  
  
h2 {  
    font-size: 30px;  
}  
  
p {  
    font-size: 14px;  
}
```

Set Font Size With Em

The em size unit is recommended by the W3C.

Example

```
h1 {  
    font-size: 2.5em; /* 40px/16=2.5em */  
}  
  
h2 {  
    font-size: 1.875em; /* 30px/16=1.875em */  
}  
  
p {  
    font-size: 0.875em; /* 14px/16=0.875em */  
}
```



code 38.txt

خلاصه خاصیت های فونت در CSS

Property	Description
<u>font</u>	Sets all the font properties in one declaration
<u>font-family</u>	Specifies the font family for text
<u>font-size</u>	Specifies the font size of text
<u>font-style</u>	Specifies the font style for text
<u>font-variant</u>	Specifies whether or not a text should be displayed in a small-caps font
<u>font-weight</u>	Specifies the weight of a font

فرمت دهی لینک ها در CSS

- با استفاده از CSS به لینک ها می توان به روش های مختلفی استایل داد.

Text Link

Text Link

Link Button

Link Button

فرمت دهی لینک ها در CSS

- برای استایل دهی به لینک ها می توان از هر صفت CSS استفاده کرد. (مانند : color، background و ...)

Example

```
a {  
    color: hotpink;  
}
```

This is a link



code 39.txt

فرمت دهی لینک ها در CSS

- به علاوه ، لینک ها را می توان حتی متفاوت از وضعیتشان استایل دهی کرد.
چهار وضعیت لینک ها عبارتند از :

- `a:link` - a normal, unvisited link
- `a:visited` - a link the user has visited
- `a:hover` - a link when the user mouses over it
- `a:active` - a link the moment it is clicked

✓ `a:link` یک لینک مشاهده نشده.

✓ `a:visited` لینکی که قبلا توسط کاربر مشاهده شده.

✓ `a:hover` حالتی از لینک زمانی که ماوس روی آن باشد.

✓ `a:active` حالتی از لینک در لحظه کلیک شدن.

Example

```
/* unvisited link */
a:link {
    color: red;
}

/* visited link */
a:visited {
    color: green;
}

/* mouse over link */
a:hover {
    color: hotpink;
}

/* selected link */
a:active {
    color: blue;
}
```



code 40.txt

نکته

- زمانی که برای حالت های مختلفی از لینک استایل تعیین می کنید ، باید ترتیب های زیر را رعایت کنید :

❖ `a:hover` : باید بعد از `a:link` و `a:visited` قرار گیرد.

❖ `a:active` : باید بعد از `a:hover` قرار گیرد.

صفت text-decoration

- صفت **text-decoration** اکثراً برای حذف زیرخط از لینک ها استفاده می شود.

Example

```
a:link {  
    text-decoration: none;  
}  
  
a:visited {  
    text-decoration: none;  
}  
  
a:hover {  
    text-decoration: underline;  
}  
  
a:active {  
    text-decoration: underline;  
}
```



code 41.txt

رنگ پس زمینه

- از صفت `background-color` برای تعیین رنگ پس زمینه لینک ها نیز استفاده می شود.

Example

```
a:link {  
    background-color: yellow;  
}  
  
a:visited {  
    background-color: cyan;  
}  
  
a:hover {  
    background-color: lightgreen;  
}  
  
a:active {  
    background-color: hotpink;  
}
```

This is a link



code 42.txt

لینک های دکمه ای - پیشرفته

- مثال زیر ، نمونه ای حرفه ای از ترکیب برخی از صفات CSS برای نمایش لینک ها به شکل جعبه یا دکمه است.

Example

```
a:link, a:visited {  
    background-color: #f44336;  
    color: white;  
    padding: 14px 25px;  
    text-align: center;  
    text-decoration: none;  
    display: inline-block;  
}  
  
a:hover, a:active {  
    background-color: red;  
}
```

This is a link



code 43.txt

مثال دوم

```
<style>
a:link, a:visited {
    background-color: white;
    color: black;
    border: 2px solid green;
    padding: 10px 20px;
    text-align: center;
    text-decoration: none;
    display: inline-block;
}

a:hover, a:active {
    background-color: green;
    color: white;
}
</style>
```

This is a link



code 44.txt

لیست ها در CSS

- در HTML دو نوع لیست وجود دارد:

- ❖ لیست های **نامرتب** (unordered list) : آیتم های این نوع لیست با هرچیزی غیر از اعداد و حروف علامتگذاری می شوند.

- ❖ لیست های **مرتب** (order list) آیتم های لیست با اعداد و حروف، علامتگذاری می شوند.

- با CSS بیشتر می توان لیست ها را دستکاری کرد، مثلاً آیتم های لیست را با یک عکس علامتگذاری نمود.

تعیین علامت گذاری لیست List Item Markers

- با خاصیت `list-style-type` می توان نوع علامت گذاری آیتم های لیست را مشخص کرد.

Example

```
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;  
}
```

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



code 45.txt

تعیین عکس به عنوان نشانگر لیست

- با خاصیت `list-style-image` می توان برای آیتم های لیست از عکس دلخواه استفاده کرد.

Example

```
ul {  
    list-style-image: url('sqpurple.gif');  
}
```

- Coffee
- Tea
- Coca Cola



code 46.txt

خاصیت display در CSS

- خصوصیت display چگونگی نمایش یک عنصر را مشخص می کند. این خاصیت یکی از مهمترین خاصیت های در CSS برای طرح بندی صفحه (layout) است.

خاصیت display در CSS

• هر یک از عناصر html به طور پیش فرض نوع نمایشی به یکی از دو صورت **block** یا **inline** دارند.

❖ **block**: یک عنصر block عنصری است که تمام طول یک سطر را اشغال می کند و قبل و بعد از آن یک سطر وجود دارد.

❖ **inline**: یک عنصر inline تنها به اندازه ای که نیاز دارد، عرض یک سطر را اشغال می کند، و هیچ اجباری در اشغال کل یک سطر وجود ندارد.

The <div> element is a block-level element.

Examples of block-level elements:

- <div>
- <h1> - <h6>
- <p>
- <form>
- <header>
- <footer>
- <section>

This is an inline element inside a paragraph.

Examples of inline elements:

-
- <a>
-

Display: none

- یکی دیگر از مقادیر رایج برای `display` مقدار `none` است. برخی عنصرهای خاص مانند javascript از این مقدار به طور پیش فرض استفاده می کنند. در جاوااسکریپت از این ویژگی برای پنهان نمودن پویای برخی از قسمت های صفحه، بدون حذف نمودن آنها استفاده می شود.

Override The Default Display Value

- می توان مقدار پیش فرض نمایش عناصر را با بهره گیری از این خاصیت **override** کرد.

A common example is making inline `` elements for horizontal menus:

Example

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

Display a list of links as a horizontal menu:

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

```
<ul>  
  <li><a href="/html/default.asp" target="_blank">HTML</a></li>  
  <li><a href="/css/default.asp" target="_blank">CSS</a></li>  
  <li><a href="/js/default.asp" target="_blank">JavaScript</a></li>  
</ul>
```



code 47.txt

The following example displays `` elements as block elements:

Example

```
span {  
    display: block;  
}
```



code 4&txt

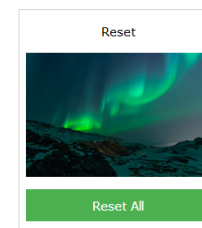
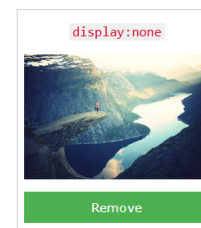
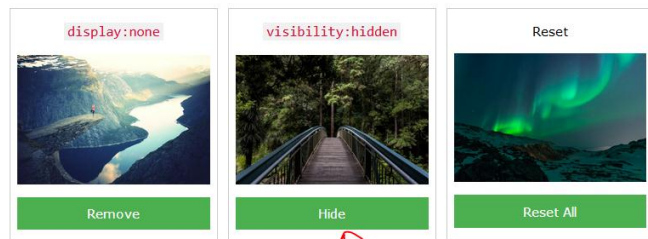
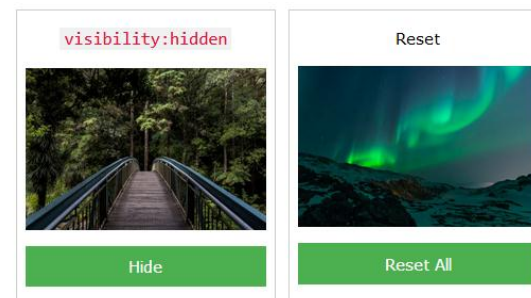
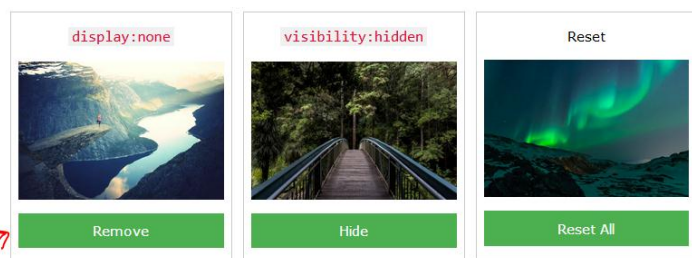
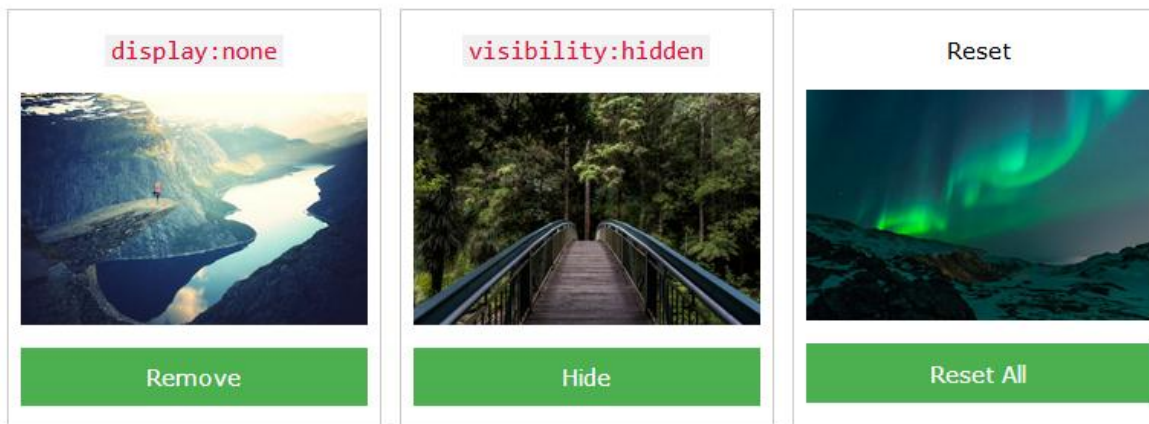
``A display property with a value of "block" results in`` ``a line break between the two elements.``

A display property with a value of "block" results in a line break between the two elements.

پنهان کردن یک عنصر

- برای پنهان کردن یک عنصر دو روش وجود دارد:
 - ✓ خصوصیت `display` را با مقدار `none` تنظیم کنیم.
 - ✓ خصوصیت `visibility` را با مقدار `hidden` تنظیم کنیم.
- به هر حال باید توجه داشته باشید که این دو روش، نتایج مختلفی را تولید خواهند کرد.
 - ✓ در روش اول (`display:none`) عنصر حذف می شود و اثری از آن در صفحه باقی نخواهد ماند.
 - ✓ در روش دوم (`visibility:hidden`) عنصر پنهان خواهد شد ولی اثر آن یا به عبارتی فضایی که اشغال کرده است، در صفحه باقی خواهد ماند.

مثال



This is a visible heading

Notice that the h1 element with display: none; does not take up any space.

```
<!DOCTYPE html>
<html>
<head>
<style>
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 h1 element with display: none; does not take up any space.</p>

</body>
</html>
```



code 49.txt

تنظیم حداکثر عرض یک عنصر

- با استفاده از خصوصیت `max-width` می توان حداکثر عرض یک عنصر را مشخص کرد. تنظیم این خصوصیت باعث می شود مقدار ارتفاع از `max-width` بیشتر نشود. مقدار خصوصیت `max-width` مقدار خصوصیت `width` را لغو می کند.

Example

```
div.ex1 {  
  width: 500px;  
  margin: auto;  
  border: 3px solid #73AD21;  
}  
  
div.ex2 {  
  max-width: 500px;  
  margin: auto;  
  border: 3px solid #73AD21;  
}
```



code 50.txt

This div element has width: 500px;

This div element has max-width: 500px;

This div element has width: 500px;

This div element has max-width: 500px;

خاصیت position

- خاصیت position نحوه قرارگیری عنصر را در صفحه مشخص می کند. مقدار این خاصیت می تواند یکی از مقادیر زیر باشد:

static ✓

relative ✓

fixed ✓

absolute ✓

sticky ✓

نکته

- موقعیت عناصر در صفحه با خصوصیات `top` و `bottom` و `left` و یا `right` تنظیم می شود، اما این خصوصیات کار نخواهند کرد مگر اینکه ابتدا خصوصیت `position` تنظیم شود و همچنین خصوصیات ذکر شده با توجه به روش موقعیت دهی، به صورت متفاوت عمل خواهند کرد.

position: static;

- موقعیت عناصر HTML در صفحه، به صورت پیش فرض **static** است. اگر موقعیت یک عنصر با مقدار **static** تنظیم شود، همیشه مطابق جریان معمول نمایش داده می شود یا به عبارتی در همان جایی که آن قرار دارد نمایش داده می شود.
- **توجه:** با تنظیم خصوصیت **position** با مقدار **static**، خصوصیات **top** و **bottom** و **right** و **left** اثری نخواهند داشت.

Example

```
div.static {  
    position: static;  
    border: 3px solid #73AD21;  
}
```

position: static;

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

This div element has position: static;



code 51.txt

position: relative;

- یک عنصر نسبت به مکان عادی خودش، موقعیت دهی می شود.

Example

```
div.relative {  
    position: relative;  
    left: 30px;  
    border: 3px solid #73AD21;  
}
```



code 52.txt

position: relative;

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

This div element has position: relative;

position: fixed;

- یک عنصر با موقعیت ثابت (fixed) در واقع نسبت به پنجره مرورگر، موقعیت دهی می شود. موقعیت عنصر حتی در صورتی که به بالا و پایین صفحه برویم (scrolled) یا اندازه پنجره مرورگر را تغییر دهیم، در همان موقعیت قبلی باقی می ماند.

Example

```
div.fixed {  
    position: fixed;  
    bottom: 0;  
    right: 0;  
    width: 300px;  
    border: 3px solid #73AD21;  
}
```



code 53.txt

position: absolute;

- یک عنصر نسبت به مکان والد خود موقعیت دهی می شود و اگر والدی وجود نداشت، نسبت به بلوک `<html>` موقعیت دهی خواهد شد.

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

Example

```
div.relative {  
  position: relative;  
  width: 400px;  
  height: 200px;  
  border: 3px solid #73AD21;  
}  
  
div.absolute {  
  position: absolute;  
  top: 80px;  
  right: 0;  
  width: 200px;  
  height: 100px;  
  border: 3px solid #73AD21;  
}
```

This <div> element has position: relative;

This <div> element has
position: absolute;



code 54.txt

روی هم قرار دادن عناصر Overlapping Elements

- زمانی که موقعیت دهی عناصر، خارج از حالت عادی باشد، می توانند روی دیگر عناصر قرار بگیرند. خصوصیت **z-index** ترتیب عناصر را مشخص می کند (کدام عنصر باید جلو یا پشت دیگر عناصر باشد). مقدار این خصوصیت می تواند **مثبت** یا **منفی** باشد.

Example

```
img {  
  position: absolute;  
  left: 0px;  
  top: 0px;  
  z-index: -1;  
}
```



code 55.txt

خاصیت float در CSS

- با این خصوصیت می توان یک عنصر را در صفحه به سمت چپ یا راست فشار داد و اجازه داد تا دیگر عناصر اطراف آنرا بپوشانند.

The CSS `float` property specifies how an element should float.

The CSS `clear` property specifies what elements can float beside the cleared element and on which side.

Float Left

Float Right

خاصیت float در CSS

• خاصیت float مقادیر زیر را می تواند به خود بگیرد:

Left ✓

right ✓

None ✓

inherit ✓

- left - The element floats to the left of its container
- right- The element floats to the right of its container
- none - The element does not float (will be displayed just where it occurs in the text). This is default
- inherit - The element inherits the float value of its parent

Example - float: right;

The following example specifies that an image should float to the **right** in a text:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac...



Example

```
img {  
    float: right;  
}
```



code 56.txt

Example - float: left;

The following example specifies that an image should float to the **left** in a text:



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent

convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac...

Example - No float

In the following example the image will be displayed just where it occurs in the text (float: none;):



Lorem ipsum dolor sit amet, consectetur

adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac...

Example

```
img {
  float: left;
}
```

Example

```
img {
  float: none;
}
```


خاصیت clear

- با استفاده از خاصیت **clear** می توان کناره هایی از عنصر را که می خواهیم متن یا عکس در آن سمت قرار نگیرد را مشخص می کنیم، در واقع از این خاصیت برای خالی کردن فضای اطراف یک عنصر استفاده می شود. که حالت های مختلفی دارد.

✓ **None**: در دو طرف عنصر مجاز است که عکس یا متن قرار بگیرد که حالت پیش فرض می باشد.

✓ **Right**: در سمت راست هیچ عکس و متنی نمی تواند قرار گیرد.

✓ **Left**: در سمت چپ هیچ عکس و متنی نمی تواند قرار گیرد.

✓ **Both**: دو طرف عنصر هیچ عکس و متنی نمی تواند قرار گیرد.

✓ **inherit**: مقدار عنصر والد رو برای عنصر فرزند لحاظ می کنه . مثلا اگر عنصر والد دارای مقدار **left** باشه عنصر فرزند دارای مقدار **inherit** باشه . مقدار **left** رو به عنصر فرزند اختصاص میده

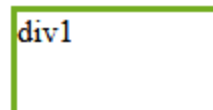
```
.div1 {
  float: left;
  width: 100px;
  height: 50px;
  margin: 10px;
  border: 3px solid #73AD21;
}
```

```
.div2 {
  border: 1px solid red;
}
```

```
.div3 {
  float: left;
  width: 100px;
  height: 50px;
  margin: 10px;
  border: 3px solid #73AD21;
}
```

```
.div4 {
  border: 1px solid red;
  clear: left;
}
```

Without clear

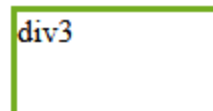


div1

div2 - Notice that div2 is after div1 in the HTML code.

However, since div1 floats to the left, the text in div2 flows around div1.

With clear



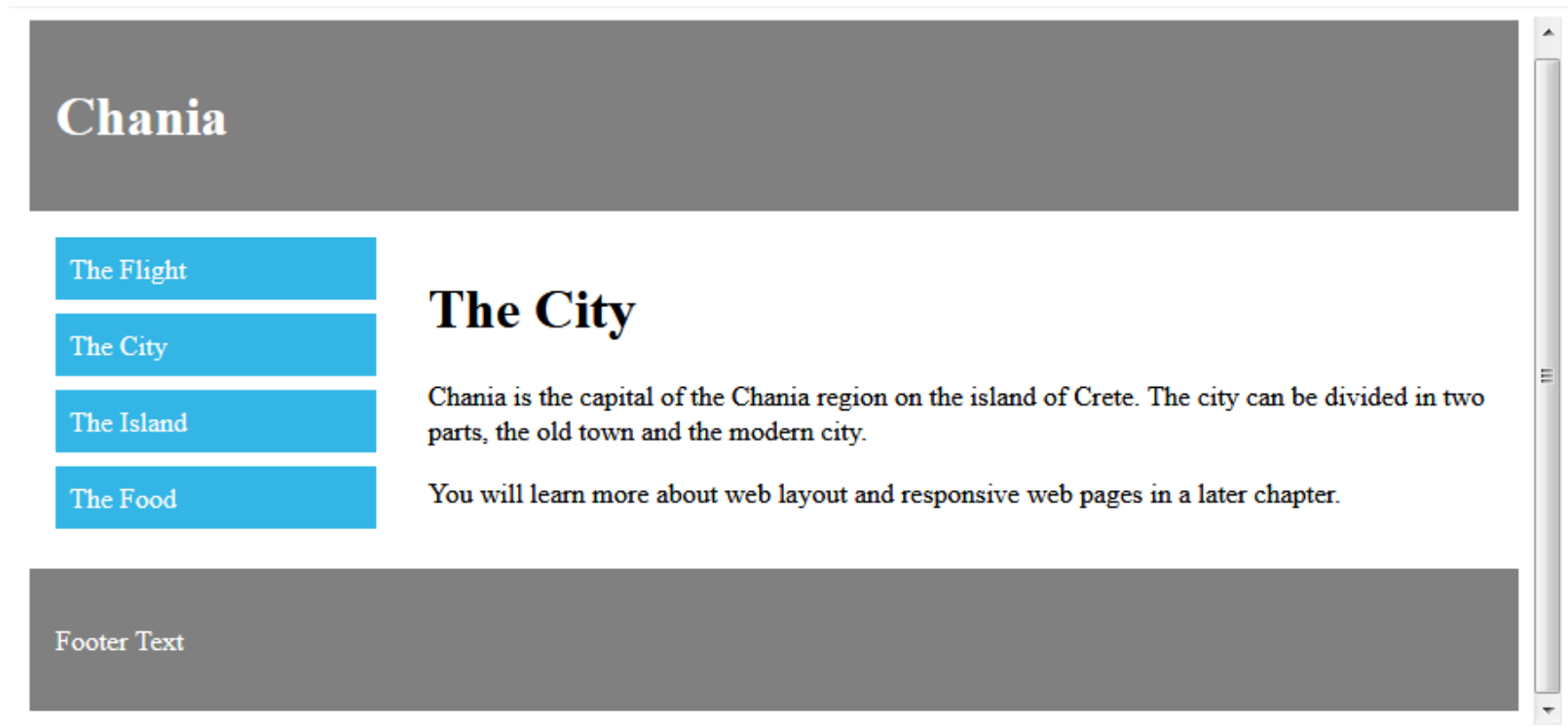
div3

div4 - Here, clear: left; moves div4 down below the floating div3. The value "left" clears elements floated to the left. You can also clear "right" and "both".



code 57.txt

Web Layout Example



```

.header, .footer {
    background-color: grey;
    color: white;
    padding: 15px;
}

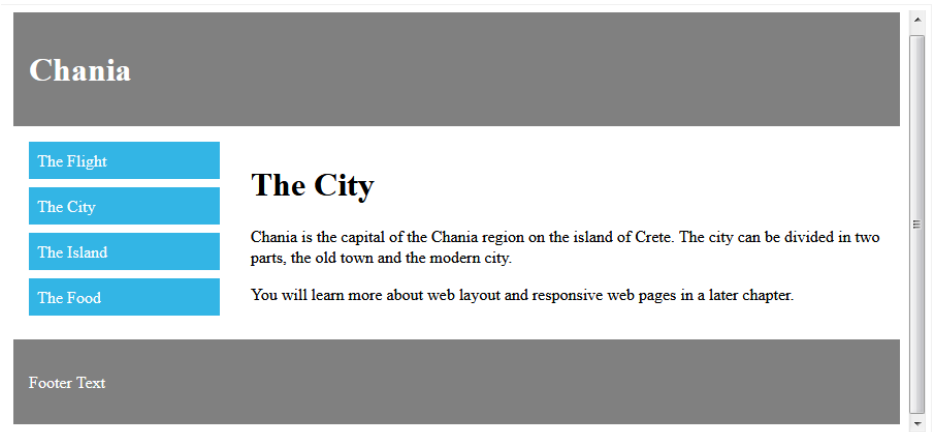
.column {
    float: left;
    padding: 15px;
}

.clearfix::after {
    content: "";
    clear: both;
    display: table;
}

.menu {
    width: 25%;
}

.content {
    width: 75%;
}

```



code 58.txt

::after pseudo-element

- شبه عناصر before و after همونطور که هم که از اسمشون پیداست میتونند در اول و آخر یک عنصر، عنصر جدیدی ایجاد کنند تا به وسیله ی اون بتونیم کنترل بیشتری برای استایل دادن به عناصر HTML داشته باشیم.

CSS Syntax

```
::after {  
    css declarations;  
}
```

Example

Insert some text after the content of each <p> element:

```
p::after {  
    content: " - Remember this";  
}
```

Horizontal & Vertical Align

- برای ترازبندی در مرکز یک عنصر بلاک همانند div می توان از `margin: auto;` استفاده کرد.

Example

```
.center {  
  margin: auto;  
  width: 50%;  
  border: 3px solid green;  
  padding: 10px;  
}
```

This div element is centered.



code 59.txt

Center Align Text

To just center the text inside an element, use `text-align: center;`

Example

```
.center {  
  text-align: center;  
  border: 3px solid green;  
}
```

This text is centered.

ترکیب انتخابگرها CSS Combinators

- یکی از امکاناتی که CSS در اختیار ما قرار می دهد، ترکیب انتخابگرها بوسیله ترکیب کننده ها می باشد. در CSS ترکیب کننده هایی از قبیل `+` , `~` , `<` وجود دارند که به ما این امکان را می دهند تا در استفاده از انتخابگرها قدرت انتخاب بالاتری داشته باشیم.

There are four different combinators in CSS:

- descendant selector (space)
- child selector (`>`)
- adjacent sibling selector (`+`)
- general sibling selector (`~`)

انتخابگر نسل Descendant

- به منظور استفاده از انتخابگر نسل فقط کافی است بین دو انتخاب گر یک فاصله خالی قرار دهیم. به این ترتیب می توان یک رابطه والد - اولاد را بین عناصر شبیه سازی کرد. به عنوان مثال اگر بخواهیم لینک های موجود در فوتر را به رنگ خاکستری در بیاوریم می توانیم از قطعه کد زیر استفاده کنیم.

```
footer a {  
  color: gray;  
}
```

انتخابگر نسل Descendant

- توجه داشته باشید که لزومی ندارد لینک ها حتما مستقیما داخل تگ footer قرار داشته باشند و کدها می توانند به صورت زیر نیز باشند.

```
<footer>
  <a href="">link 1</a>
  <p>
    <a href="">link 2</a>
  </p>
  <a href="">link 3</a>
  <div>
    <span>
      <a href="">link 4</a>
    </span>
  </div>
</footer>
```

انتخابگر نسل Descendant

- این امکان وجود دارد که بیشتر از یک سطح هم به داخل حرکت کنیم مثلاً فرض کنید می خواهیم تمام لینک هایی که در عناصر menu یی که در عناصر nav قرار دارند را انتخاب کنیم. می توانیم بصورت زیر عمل کنیم:

```
nav menu a {  
  
}
```

مثال

```
<!DOCTYPE html>
<html>
<head>
<style>
div p {
    background-color: yellow;
}
</style>
</head>
<body>

<div>
    <p>Paragraph 1 in the div.</p>
    <p>Paragraph 2 in the div.</p>
    <span><p>Paragraph 3 in the div.</p></span>
</div>

<p>Paragraph 4. Not in a div.</p>
<p>Paragraph 5. Not in a div.</p>

</body>
</html>
```

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3 in the div.

Paragraph 4. Not in a div.

Paragraph 5. Not in a div.



code 60.txt

انتخابگر فرزند

- برخی مواقع نیاز داریم تا فقط فرزندان مستقیم یک عنصر را انتخاب کنیم. برای این منظور می توانیم از علامت `>` استفاده کنیم. به عنوان مثال قطعه کدی که در مورد `footer` مثال زدیم را در نظر بگیرید.
- فرض کنید این بار می خواهیم فقط رنگ لینک هایی که مستقیماً داخل تگ `footer` قرار دارند به به رنگ خاکستری تغییر دهیم، برای این منظور می توانیم به صورت زیر عمل کنیم:

```
<footer>
  <a href="">link 1</a>
  <p>
    <a href="">link 2</a>
  </p>
  <a href="">link 3</a>
</div>
  <span>
    <a href="">link 4</a>
  </span>
</div>
</footer>
```

```
footer > a {
  color:gray;
}
```

Child Selector

The child selector selects all elements that are the immediate children of a specified element.

مثال

```
<!DOCTYPE html>
<html>
<head>
<style>
div > p {
    background-color: yellow;
}
</style>
</head>
<body>

<div>
    <p>Paragraph 1 in the div.</p>
    <p>Paragraph 2 in the div.</p>
    <span><p>Paragraph 3 in the div.</p></span>
</div>

<p>Paragraph 4. Not in a div.</p>
<p>Paragraph 5. Not in a div.</p>

</body>
</html>
```

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3 in the div.

Paragraph 4. Not in a div.

Paragraph 5. Not in a div.



code 61.txt

انتخابگر برادر - خواهر Adjacent Sibling

فرض کنید عناصری بصورت زیر داشته باشیم:

```
<p class="first-paragraph"></p>
<p>عنصری که می خواهیم انتخاب کنیم</p>
<p></p>
<p></p>
```

حال فرض کنید می خواهیم رنگ پاراگراف دوم را قرمز کنیم، برای این منظور می توانیم با استفاده از علامت `+` عنصر کنار پاراگرافی که کلاس `first-paragraph` دارد را هدف قرار دهیم.

```
.first-text + p {
  color:red;
}
```

```
<!DOCTYPE html>
<html>
<head>
<style>
div + p {
    background-color: yellow;
}
</style>
</head>
<body>

<div>
    <p>Paragraph 1 in the div.</p>
    <p>Paragraph 2 in the div.</p>
</div>

<p>Paragraph 3. Not in a div.</p>
<p>Paragraph 4. Not in a div.</p>

</body>
</html>
```

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3. Not in a div.

Paragraph 4. Not in a div.



code 62.txt

انتخابگر برادر - خواهرهای بعدی General Sibling

انتخابگر قبلی فقط امکان انتخاب عنصری که دقیقاً بعد از عنصر مورد نظر قرار داشت را به ما می داد. اما با استفاده از علامت ~ در انتخابگر **General Sibling** می توان تمام عنصرهای بعدی را هدف قرار داد. فرض کنید در همان مثال بالا می خواهیم تمام پاراگراف هایی که بعد از کلاس **first-paragraph** آمده اند را قرمز رنگ کنیم. برای این منظور می توانیم از انتخابگر **General Sibling** به صورت زیر استفاده کنیم.

```
.first-paragraph ~ p {  
  color:red;  
}
```

```
<!DOCTYPE html>
<html>
<head>
<style>
div ~ p {
    background-color: yellow;
}
</style>
</head>
<body>

<p>Paragraph 1.</p>

<div>
    <code>Some code.</code>
    <p>Paragraph 2.</p>
</div>

<p>Paragraph 3.</p>
<code>Some code.</code>
<p>Paragraph 4.</p>

</body>
</html>
```

Paragraph 1.

Some code.

Paragraph 2.

Paragraph 3.

Some code.

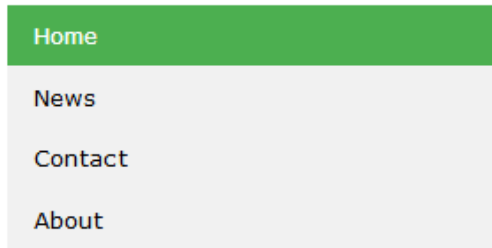
Paragraph 4.



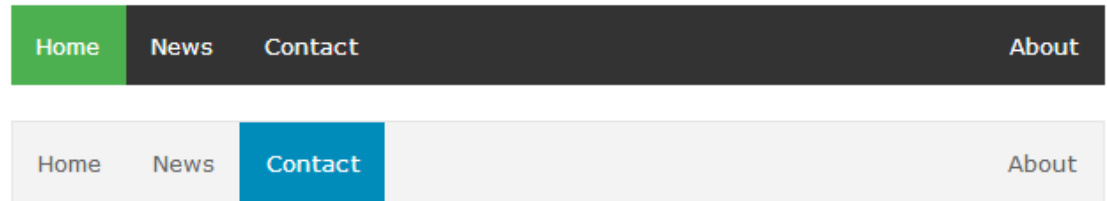
code 63.txt

CSS Navigation Bar

Vertical



Horizontal

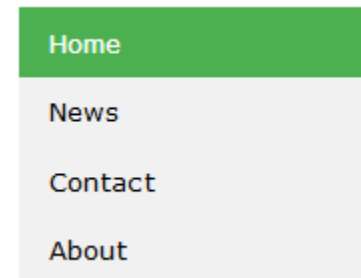


مثالی از نوار منوی عمودی

Example

```
ul {  
  list-style-type: none;  
  margin: 0;  
  padding: 0;  
  width: 200px;  
  background-color: #f1f1f1;  
}  
  
li a {  
  display: block;  
  color: #000;  
  padding: 8px 16px;  
  text-decoration: none;  
}  
  
/* Change the link color on hover */  
li a:hover {  
  background-color: #555;  
  color: white;  
}
```

```
.active {  
  background-color: #4CAF50;  
  color: white;  
}
```



code 64.txt

مثالی از نوار منوی افقی

Example

```
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333;
}

li {
  float: left;
}

li a {
  display: block;
  color: white;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}

/* Change the link color to #111 (black) on hover */
li a:hover {
  background-color: #111;
}
```

Home

News

Contact

About

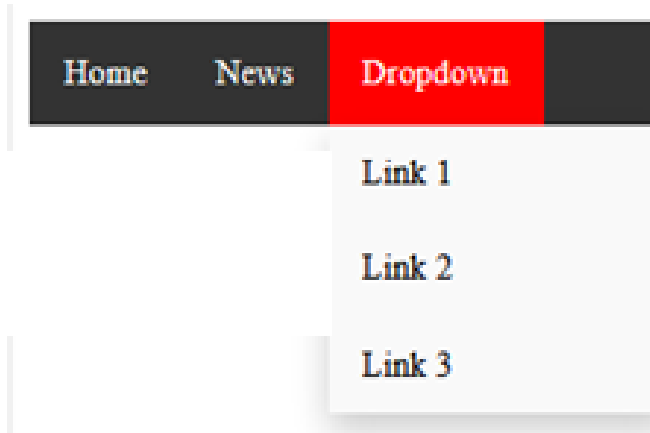
Example

```
.active {
  background-color: #4CAF50;
}
```



code 65.txt

Dropdown Navbar



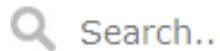
code 66.txt

CSS Forms

https://www.w3schools.com/css/css_form.asp

Input with icon/image

If you want an icon inside the input, use the `background-image` property and position it with the `background-position` property. Also notice that we add a large left padding to reserve the space of the icon:



Example

```
input[type=text] {  
    background-color: white;  
    background-image: url('searchicon.png');  
    background-position: 10px 10px;  
    background-repeat: no-repeat;  
    padding-left: 40px;  
}
```



code 67.txt

CSS Transitions

How to Use CSS Transitions?

To create a transition effect, you must specify two things:

- the CSS property you want to add an effect to
- the duration of the effect

Note: If the duration part is not specified, the transition will have no effect, because the default value is 0.

مثال

```
<style>
div {
    width: 100px;
    height: 100px;
    background: red;
    -webkit-transition: width 2s; /* For Safari 3.1 to 6.0 */
    transition: width 2s;
}

div:hover {
    width: 300px;
}
</style>
```



code 68.txt

CSS Transitions

`transition` یک ویژگی از نوع ویژگی های مختصر نویسی می باشد که با استفاده از آن می توان ویژگی های زیر را به صورت یکجا و مختصر در CSS تعریف کرد:

`transition-property`

`transition-duration`

`transition-timing-function`

`transition-delay`

ترتیب نوشتن این ویژگی ها با استفاده از ویژگی `transition` مهم است و باید به ترتیب اشاره شده در بالا نوشته شوند. اولین عددی که نشان دهنده زمان باشد به عنوان مقدار ویژگی `transition-duration` و دومین زمان به عنوان مقدار ویژگی `transition-delay` ثبت می شوند.

```
div {  
  transition: background-color .3s linear 1s;  
}
```

Change Several Property Values

Example

```
div {  
    -webkit-transition: width 2s, height 4s; /* Safari */  
    transition: width 2s, height 4s;  
}
```

CSS Animations

w3schools.com

