

Border

The CSS border properties allow you to specify the style, width, and color of an element's border.

- **1. Border-style**

The border-style property specifies what kind of border to display.

The following values are allowed:

- dotted - Defines a dotted border
- dashed - Defines a dashed border
- solid - Defines a solid border
- double - Defines a double border
- none - Defines no border
- hidden - Defines a hidden border

Example, `p {border-style: solid;}`

- **2. Border-width**

The border-width property specifies the width of the four borders.

The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.

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

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

- **3. Border-color**

- The border-color property is used to set the color of the four borders.

-

- `p.one {
border-style: solid;`

```
border-color: red;
}
```

-

- **For specific border color**

- ```
p.one {
border-style: solid;
border-color: red green blue yellow; /* red top, green right, blue bottom and yellow left
*/
}
```

- 

- **There are also properties for specifying each of the borders (top, right, bottom, and left)**

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

-

- **Note: Shortcut for above properties.**

- ```
p.one {
border: 1px solid red;
}
```

- 

- **4. Rounded Border**

- The border-radius property is used to add rounded borders to an element.

- 

- ```
p {
border: 2px solid red;
border-radius: 10px;
}
```

Backgrounds

The CSS background properties are used to add background effects for elements.

- The background-color property specifies the background color of an element.
- Example,
-

```
<body>
  <div class="main-div">
    <h1>This is heading</h1>
    <p>This is content section</p>
  </div>
</body>
```

```
<style>
  .main-div {
    background-color: aqua;
  }
</style>
```

Output:



- The background-image property specifies an image to use as the background of an element.
- By default, the background-image property repeats an image both horizontally and vertically.
- We can set background-repeat: no-repeat; to show image only once
- To repeat an image vertically, set background-repeat: repeat-y;

- To repeat an image horizontally, set background-repeat: repeat-x;
-
- Example,
-

```
<style>
  .main-div {
    background-image: url("image.jpg");
    background-repeat: no-repeat; /*repeat-x repeat-y*/
  }
</style>
```

-
-
- The background-position property is used to specify the position of the background image.
-

```
<style>
  .main-div {
    background-image: url("image.jpg");
    background-repeat: no-repeat; /*repeat-x repeat-y*/
    background-position: right top; /*left bottom/ 50px 100px/ 10% 50%*/
  }
</style>
```

- The background-attachment property specifies whether the background image should scroll or be fixed.
- Specify that the background image should scroll with the rest of the page. (default)
- Specify that the background image should be fixed.

-

```
<style>
  .main-div {
    background-image: url("image.jpg");
    background-repeat: no-repeat; /*repeat-x repeat-y*/
    background-position: right top; /*left bottom/ 50px 100px/ 10% 50%*/
    background-attachment: scroll; /* fixed; */
  }
</style>
```

Example,

```
<body>
  <div class="main-div">
    <h1>This is heading</h1>
    <p>This is content section Lorem ipsum dolor sit amet consectetur adipisicing elit. Soluta labore iusto quisquam enim tempora cumque reiciendis dignissimos! Nam incidunt, saepe voluptates vitae, dolore perspiciatis accusamus veniam sunt quas quam eligendi? Ipsum mollitia incidunt eum, culpa, excepturi voluptatibus nesciunt, voluptas unde iusto veritatis quos doloribus! Animi debitis perspiciatis excepturi similique, iusto quam fugiat quos sit consequuntur quae praesentium enim eaque reiciendis. Distinctio reiciendis facilis a, architecto odit impedit, recusandae magni magnam voluptatibus velit deleniti illo vitae, eum aliquid molestias. Quaerat aliquid quam illum obcaecati ad debitis modi doloremque vitae alias dolorum! laudantium sed soluta eius magni laborum animi! Distinctio vel eaque nihil atque, molestias totam aliquid possimus consequatur natus quae amet est tempora reiciendis numquam tenetur, itaque rerum perspiciatis reprehenderit. Expedita? Excepturi voluptate dolore, dolor iure fugiat totam quaerat aliquid provident quae rem ipsa error, quod et odit omnis dicta saepe a sunt ratione ad eius! Voluptatem dolorem sint minus praesentium. Voluptatum, autem voluptatibus reprehenderit atque est nesciunt doloremque aliquid minus ipsum obcaecati praesentium ab inventore maiores et quidem rerum saepe nemo, numquam at iusto corrupti quae eius mollitia modi? Dolorem! Veniam praesentium voluptas totam possimus quasi facere maxime quia molestiae suscipit ex vel delectus consectetur laboriosam dicta illo, beatae dignissimos exercitationem
```

```
.main-div {
  color: ■ bisque;
  width: auto;
  background-image: url("bg.jpg");
  background-attachment: fixed;
}
```

Output:



The background-size Specifies the size of the background image(s). We have different parameter to specify background-size property.

- Auto: Default value. The background image is displayed in its original size
- Percentage: Sets the width and height of the background image in percent of the parent element. The first value sets the width, the second value sets the height.
- Cover: Resize the background image to cover the entire container, even if it has to stretch the image or cut a little bit off one of the edges
- Contain: Resize the background image to make sure the image is fully visible.

```
background-size: auto; /* contain/cover/50% 20% */
```

Height/Width

The CSS height and width properties are used to set the height and width of an element.

The CSS max-width property is used to set the maximum width of an element.

The height and width properties may have the following values:

- auto - This is default. The browser calculates the height and width
- length - Defines the height/width in px, cm etc.
- % - Defines the height/width in percent of the containing block
-

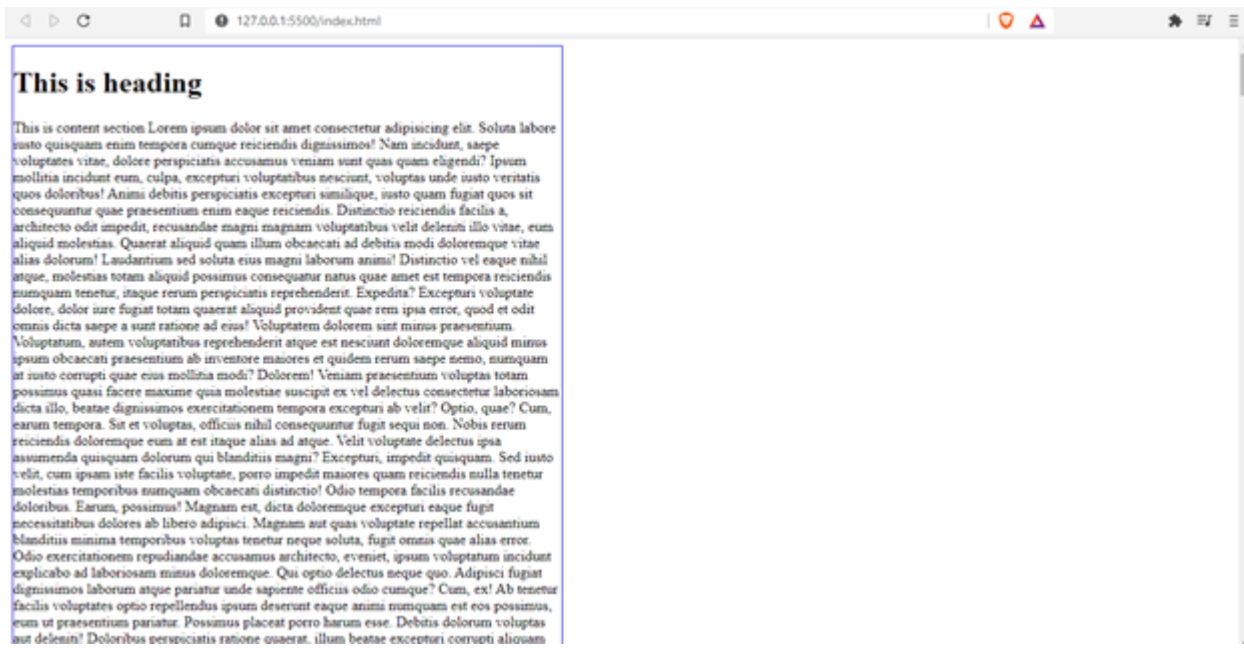
Note: Remember that the height and width properties do not include padding, borders, or margins! They set the height/width of the area inside the padding, border, and margin of the element!

Width Vs Max-Width

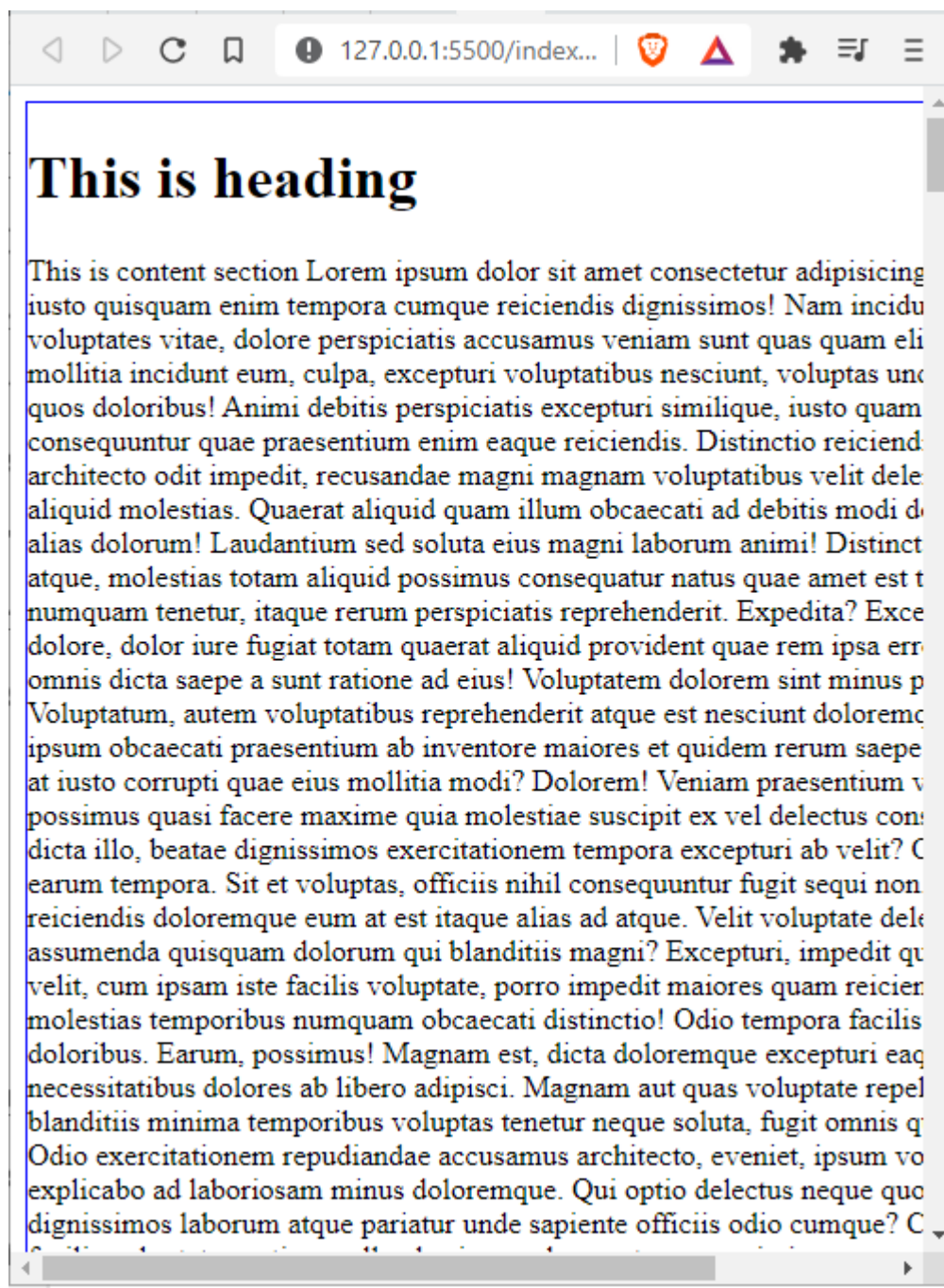
```
<div class="main-div">
  <h1>This is heading</h1>
  <p>This is content section Lorem ipsum dolor sit amet consectetur adipisicing elit.
  Soluta labore iusto quisquam enim tempora cumque reiciendis dignissimos! Nam incidunt,
  saepe voluptates vitae, dolore perspiciatis accusamus veniam sunt quas quam eligendi?
  Ipsum mollitia incidunt eum, culpa, excepturi voluptatibus nesciunt, voluptas unde
  iusto veritatis quos doloribus! Animi debitis perspiciatis excepturi similique, iusto
```

```
.main-div {
  color: ☐ black;
  width: 600px;
  border: 1px solid ☐ blue;
}
```



Output in full screen:



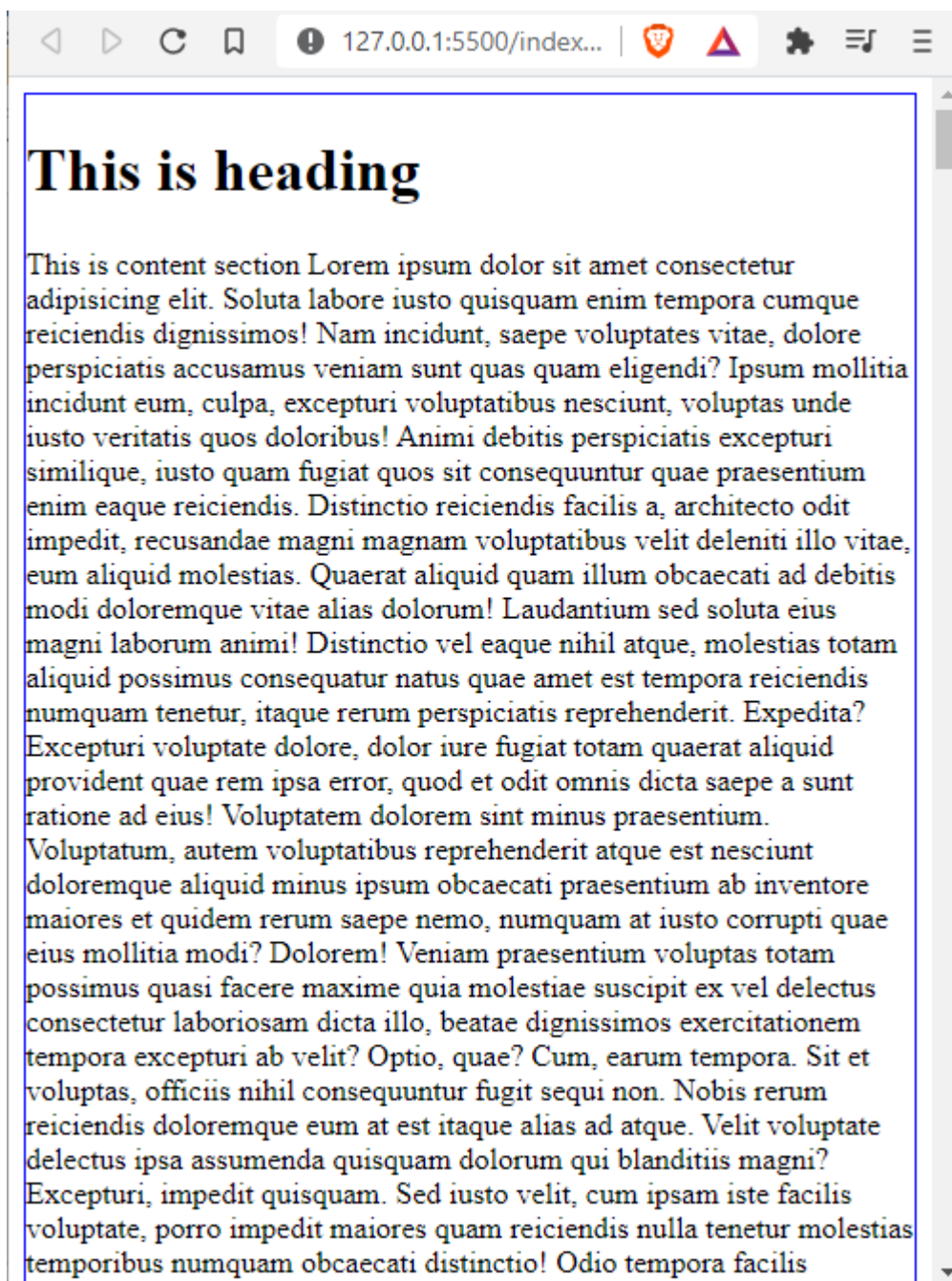
Output on screen width lesser than 600px



Use of max-width instead of width

```
.main-div {  
  color:  black;  
  max-width: 600px;  
  border: 1px solid  blue;  
}
```

Output on smaller screen (less than 600px)



Margins

Margins are used to create space around elements, outside of any defined borders.

CSS has properties for specifying the margin for each side of an element:

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

All the margin properties can have the following 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

Note: Negative values are allowed.

Note: You can set the margin property to auto to horizontally center the element within its container.

Margin Collapse

Top and bottom margins of elements are sometimes collapsed into a single margin that is equal to the largest of the two margins.

This does not happen on left and right margins! Only top and bottom margins!

Look at the following example:

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

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

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.

Padding

The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

CSS has properties for specifying the padding for each side of an element:

- 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.

Box Sizing

Suppose, the `<div>` element is given a width of 300px. However, the actual width of the `<div>` element will be 350px (300px + 25px of left padding + 25px of right padding).

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

To keep the width at 300px, no matter the amount of padding, you can use the box-sizing property. This causes the element to maintain its width; if you increase the padding, the available content space will decrease.

```
div {
```

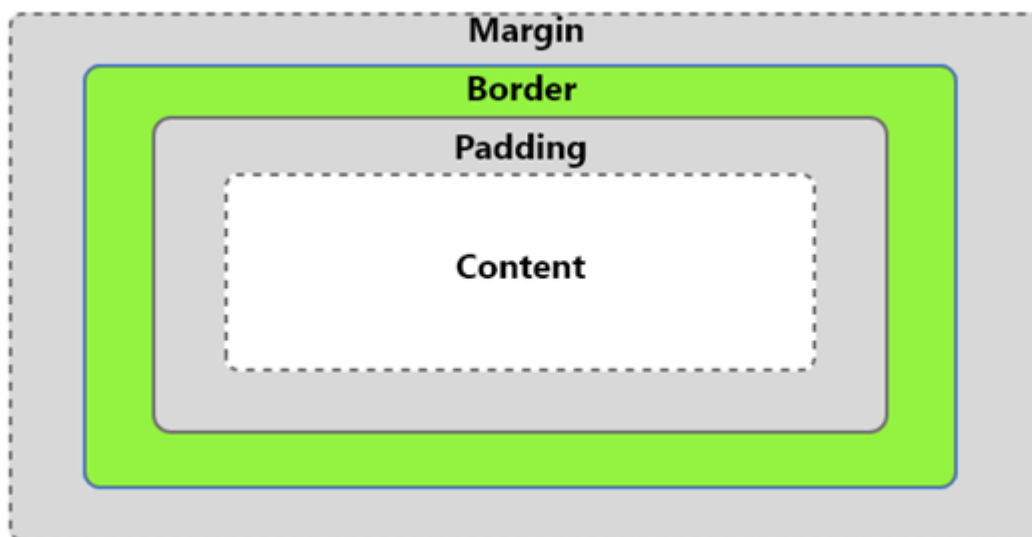
```
width: 300px;  
padding: 25px;  
box-sizing: border-box;  
}
```

Note: Box-sizing has the default value of **content-box**.

Box-Model

In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content. The image below illustrates the box model.



Explanation of the different parts:

- Content - The content of the box, where text and images appear
- Padding - Clears an area around the content. The padding is transparent
- Border - A border that goes around the padding and content
- Margin - Clears an area outside the border. The margin is transparent

The total width of an element should be calculated like this:

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

The total height of an element should be calculated like this:

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

Text

CSS has a lot of properties for formatting text.


```
<body>
  <div class="main-div">
    <h1>This is heading</h1>
    <p id="para1">This is content section Lorem ipsum dolor sit amet consectetur
adipiscing elit. Soluta labore iusto quisquam enim tempora cumque reiciendis
dignissimos! Nam incidunt, saepe voluptates vitae, dolore perspiciatis accusamus
veniam sunt quas quam eligendi?
```

```
#para1 {
  color: ■ cadetblue;
  text-align: right; /* left center justify*/
  text-decoration: underline; /* overline line-through none*/
  text-transform: capitalize; /* uppercase lowercase */
  line-height: 30px;
  text-indent: 50px;
  letter-spacing: 10px;
  word-spacing: 10px;
}
```

Links

With CSS, links can be styled in many different ways.





Links can be styled with any CSS property (e.g. color, font-family, background, etc.).

```
a {  
  color: orange;  
  text-decoration: none;  
}
```

In addition, links can be styled differently depending on what state they are in.

The four links states are:

- 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

```
/* unvisited link */  
a:link {  
  color: orange;  
}  
  
/* visited link */  
a:visited {  
  color: green;  
}  
  
/* mouse over link */  
a:hover {  
  color: black;  
}  
  
/* selected link */  
a:active {  
  color: red;  
}
```


When setting the style for several link states, there are some order rules:

- a:hover MUST come after a:link and a:visited
 - a:active MUST come after a:hover

Fonts

In CSS, we use the font-family property to specify the font of a text.

The font-family property should hold several font names as a "fallback" system, to ensure maximum compatibility between browsers/operating systems. Start with the font you want, and end with a generic family (to let the browser pick a similar font in the generic family, if no other fonts are available). The font names should be separated with comma.

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

Font Style

The font-style property is mostly used to specify italic text.

This property has three values:

- 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)

Font Weight

The font-weight property specifies the weight of a font. This property has different values such as normal (default), bold, bolder and values from 100 to 900.

Font Variant

The font-variant property specifies whether or not a text should be displayed in a small-caps font.

In a small-caps font, all lowercase letters are converted to uppercase letters. However, the converted uppercase letters appears in a smaller font size than the original uppercase letters in the text.

Font Size

The font-size property sets the size of the text.

```
body {  
  font-family: "Times New Roman", Arial, Helvetica, sans-serif;  
  font-style: italic;  
  /* normal oblique */  
  font-weight: bold;  
  /* normal bolder 100 200 ... 900 */  
  font-variant: small-caps;  
  /* normal */  
  font-size: 16px;  
}
```

Lists

In HTML, there are two main types of lists:

- unordered lists () - the list items are marked with bullets
- ordered lists () - the list items are marked with numbers or letters


The CSS list properties allow you to:

- Set different list item markers for ordered lists
- Set different list item markers for unordered lists
- Set an image as the list item marker
- Add background colors to lists and list items


```
ul {  
  list-style-type: none;  
  /* disc circle square */  
}  
  
ol {  
  list-style-type: lower-alpha;  
  /* upper-alpha lower-roman upper-roman */  
}
```

Tables

The look of an HTML table can be greatly improved with CSS.

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

```
table {
  border-collapse: collapse;
  /* separate */
  width: 100%;
  border-spacing: 15px;
  empty-cells: hide;
  /* show */
}
```

```
caption {
  border: 1px solid  red;
  caption-side: bottom;
  /* top */
}
```

```
th,
td {
  padding: 15px;
  text-align: left;
  vertical-align: middle;
  /* top bottom */
}

tr:hover {
  background-color: #f5f5f5;
}

/* table-striped */
tr:nth-child(even) {
  background-color: #f2f2f2;
}
```