



# CSS: Properties

---

```
171 #content .article img.left.border {  
172     padding: 0 9px 9px 0;  
173     border-right: 1px dotted #999;  
174     border-bottom: 1px dotted #999; }  
175 #content .article blockquote {  
176     margin-left: 10px;  
177     padding-left: 10px;  
178     border-left: 3px solid #252525; }  
179 #content .article ul {  
180     padding-left: 1em;  
181     list-style-type: circle; }
```

Course Code: CIS121

Module Leader: Md. Minhaj Hosen

Lecturer (CIS),

Daffodil International University (DIU)

# Text-related CSS Properties

- ◆ **color** – specifies the color of the text
- ◆ **font-size** – size of font: **xx-small**, **x-small**, **small**, **medium**, **large**, **x-large**, **xx-large**, **smaller**, **larger** or numeric value
- ◆ **font-family** – comma separated font names
  - ◆ Example: **verdana**, **sans-serif**, etc.
  - ◆ The browser loads the first one that is available
  - ◆ There should always be at least one generic font
- ◆ **font-weight** can be **normal**, **bold**, **bolder**, **lighter** or a number in range [100 .. 900]

# CSS Rules for Fonts (2)

- ◆ **font-style** – styles the font
  - ◆ **Values:** normal, italic, oblique
- ◆ **text-decoration** – decorates the text
  - ◆ **Values:** none, underline, line-through, overline, blink
- ◆ **text-align** – defines the alignment of text or other content
  - ◆ **Values:** left, right, center, justify

# Shorthand Font Property

## ◆ font

- ◆ Shorthand rule for setting multiple font properties at the same time

```
font:italic normal bold 12px/16px verdana
```

is equal to writing this:

```
font-style: italic;  
font-variant: normal;  
font-weight: bold;  
font-size: 12px;  
line-height: 16px;  
font-family: verdana;
```

# Backgrounds

- ◆ background-image

- ◆ URL of image to be used as background, e.g.:

```
background-image:url("back.gif");
```

- ◆ background-color

- ◆ Using color and image and the same time

- ◆ background-repeat

- ◆ repeat-x, repeat-y, repeat, no-repeat

- ◆ background-attachment

- ◆ fixed / scroll

## Backgrounds (2)

- ♦ **background-position:** specifies vertical and horizontal position of the background image
  - ♦ Vertical position: top, center, bottom
  - ♦ Horizontal position: left, center, right
  - ♦ Both can be specified in percentage or other numerical values
- ♦ Examples:

```
background-position: top left;
```

```
background-position: -5px 50%;
```

# Background Shorthand Property

- ♦ **background:** shorthand rule for setting background properties at the same time:

```
background: #FFF0C0 url("back.gif") no-repeat  
fixed top;
```

is equal to writing:

```
background-color: #FFF0C0;  
background-image: url("back.gif");  
background-repeat: no-repeat;  
background-attachment: fixed;  
background-position: top;
```

- ♦ Some browsers will not apply BOTH color and image for background if using shorthand rule

# Background-image or <img>?

- ◆ Background images allow you to save many image tags from the HTML
  - ◆ Leads to less code
  - ◆ More content-oriented approach
- ◆ All images that are not part of the page content (and are used only for "beautification") should be moved to the CSS

# Borders

- ◆ `border-width: thin, medium, thick` or numerical value (e.g. `10px`)
- ◆ `border-color: color` alias or RGB value
- ◆ `border-style: none, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset`
- ◆ Each property can be defined separately for left, top, bottom and right
  - ◆ `border-top-style, border-left-color, ...`

# Border Shorthand Property

- ◆ **border:** shorthand rule for setting border properties at once:

```
border: 1px solid red
```

is equal to writing:

```
border-width:1px;  
border-color:red;  
border-style:solid;
```

- ◆ Specify different borders for the sides via shorthand rules: **border-top**, **border-left**, **border-right**, **border-bottom**
- ◆ When to avoid **border:0**

# Width and Height

- ◆ `width` – defines numerical value for the width of element, e.g. `200px`
- ◆ `height` – defines numerical value for the height of element, e.g. `100px`
  - ◆ By default the height of an element is defined by its content
  - ◆ Inline elements do not apply height, unless you change their `display` style.