Module No. 1

Style sheets: Introduction CSS, Applying CSS to HTML, Selectors, Properties and Values, CSS Colors and Backgrounds, CSS Box Model, CSS Margins, Padding, and Borders, CSS Text and Font Properties

How CSS is different from HTML?

- HTML is used to define a structure of a web page whereas CSS is used to style of the web pages by using different styling features.
- HTML consists of tags inside which text is enclosed and CSS consists of selectors and declaration blocks.
- CSS can be internal or external depending upon the requirement.
- We cannot use HTML inside a CSS sheet but we can use CSS inside an HTML document.
- CSS has comparatively higher backup and support than HTML.

CSS (Cascading Style Sheets)

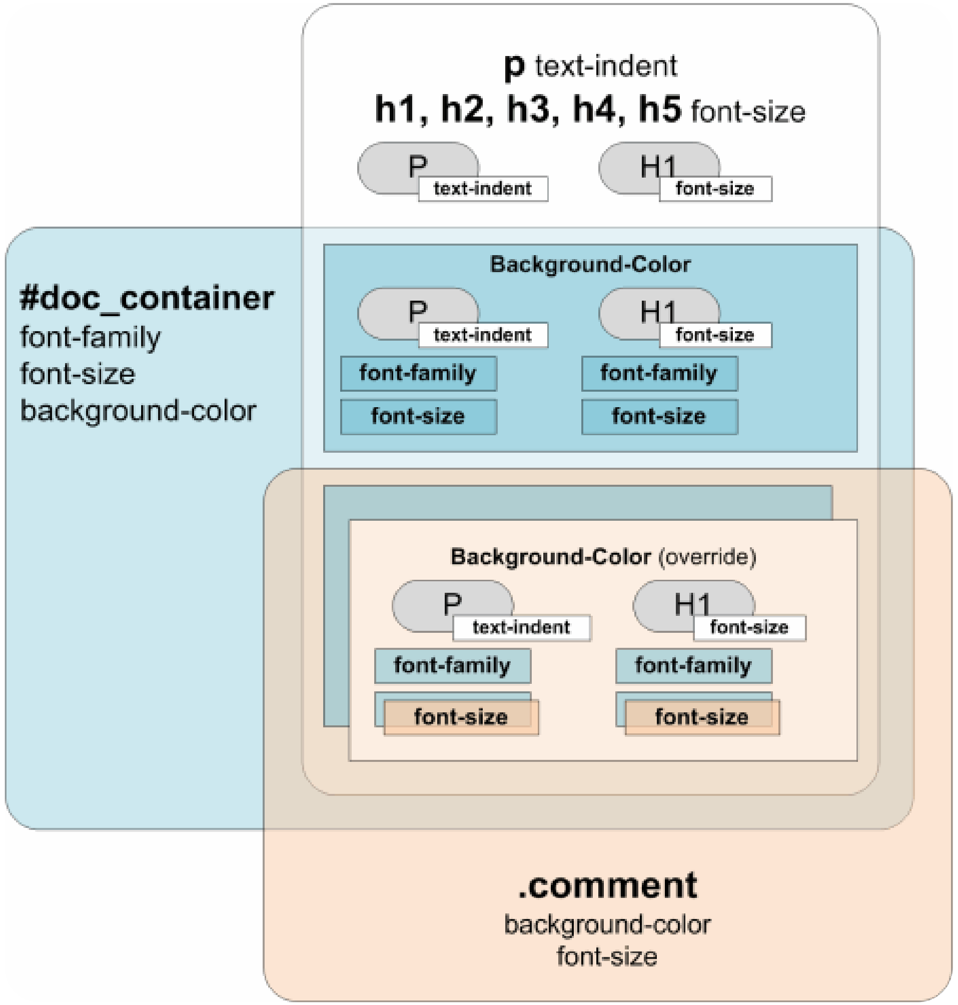
CSS (Cascading Style Sheets) is a style sheet language used to design a webpage to make it attractive. The reason for using this is to simplify the process of making web pages presentable. It allows you to apply styles on web pages.

CSS Introduction

- Cascading Style Sheets (CSS)
  - Used to describe the presentation of documents
  - Define sizes, spacing, fonts, colors, layout, etc.
  - Improve content accessibility
  - Improve flexibility
- Designed to separate presentation from content
- Due to CSS, all HTML presentation tags and attributes are deprecated, e.g. font, center, etc.
- CSS can be applied to any XML document
  - Not just to HTML / XHTML
- CSS can specify different styles for different media
  - On-screen
  - In print
  - Handheld, projection, etc.

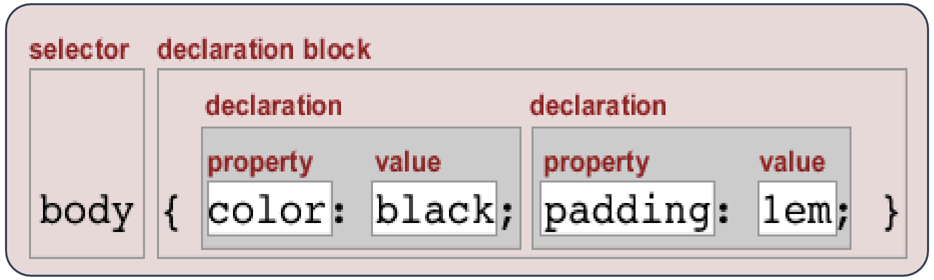
Why “Cascading”?

- Priority scheme determining which style rules apply to element
  - Cascade priorities or specificity (weight) are calculated and assigned to the rules
- Some CSS styles are inherited and some not
  - Text-related and list-related properties are inherited - color, font-size, font-family, line-height, text-align, list-style, etc
  - Box-related and positioning styles are not inherited - width, height, border, margin, padding, position, float, etc
  - <a> elements do not inherit color and text-decoration



Style Sheets Syntax

- Stylesheets consist of rules, selectors, declarations, properties and values



```
body { color: black; padding: 1em; }
```

http://css.maxdesign.com.au/

- Selectors are separated by commas
- Declarations are separated by semicolons
- Properties and values are separated by colons

```
h1,h2,h3 { color: green; font-weight: bold; }
```

Selectors

- Selectors determine which element the rule applies to:
  - All elements of specific type (tag)
  - Those that mach a specific attribute (id, class)
  - Elements may be matched depending on how they are nested in the document tree (HTML)
- Examples:

```
.header a { color: green }
```

```
#menu li { padding-top: 8px }
```

Selectors

- Three primary kinds of selectors:
  - By tag (type selector):

```
h1{ font-family: verdana,sans-serif; }
```

  - By element id:

```
#element_id { color: #ff0000; }
```

  - By element class name (only for HTML):

```
.myClass {border: 1px solid red}
```

- Selectors can be combined with commas:

```
h1, .link, #top-link {font-weight: bold}
```

This will match <h1> tags, elements with class link, and element with id top-link

Selectors

- Match relative to element placement:

```
p a {text-decoration: underline}
```

This will match all <a> tags that are inside of <p>

- * – universal selector (avoid or use with care!):

```
p * {color: black}
```

This will match all descendants of <p> element

- + selector – used to match “next sibling”:

```
img + .link {float:right}
```

This will match all siblings with class name link that appear immediately after <img> tag

Selectors

- `>` selector – matches direct child nodes:

  ```
  p > .error {font-size: 8px}
  ```

  This will match all elements with class `error`, direct children of `<p>` tag
- `[ ]` – matches tag attributes by regular expression:

  ```
  img[alt~=logo] {border: none}
  ```

  This will match all `<img>` tags with `alt` attribute containing the word `logo`
- `.class1.class2` (no space) - matches elements with both (all) classes applied at the same time

Values in the CSS Rules

- Colors are set in RGB format (decimal or hex):
  - Example: #a0a6aa = rgb(160, 166, 170)
  - Predefined color aliases exist: black, blue, etc.
- Numeric values are specified in:
  - Pixels, ems, e.g. 12px , 1.4em
  - Points, inches, centimeters, millimeters
    - E.g. 10pt , 1in, 1cm, 1mm
  - Percentages, e.g. 50%
    - Percentage of what?...
  - Zero can be used with no unit: border: 0;

Default Browser Styles

- Browsers have default CSS styles
  - Used when there is no CSS information or any other style information in the document
- Caution: default styles differ in browsers
  - E.g. margins, paddings and font sizes differ most often and usually developers reset them

```
* { margin: 0; padding: 0; }
```

```
body, h1, p, ul, li { margin: 0; padding: 0; }
```

Linking HTML and CSS

- HTML (content) and CSS (presentation) can be linked in three ways:
  - Inline: the CSS rules in the style attribute
    - No selectors are needed
  - Embedded: in the <head> in a <style> tag
  - External: CSS rules in separate file (best)
    - Usually, a file with .css extension
    - Linked via <link rel="stylesheet" href=…> tag or @import directive in embedded CSS block
- Using external files is highly recommended
  - Simplifies the HTML document
  - Improves page load speed as the CSS file is cached

Inline Styles: Example

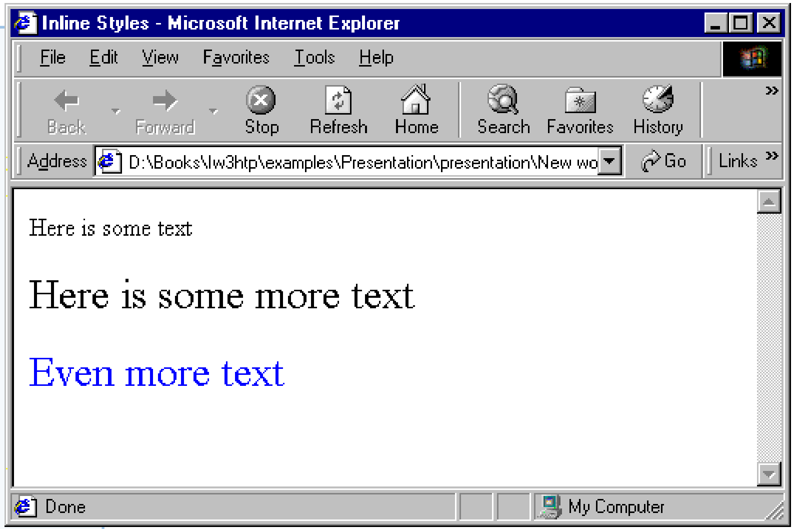
inline-styles.html

```
<!DOCTYPE html>
<html>
<head>
 <title>Inline Styles</title>
</head>
<body>
 <p>Here is some text</p>
<!--Separate multiple styles with a semicolon-->
 <p style="font-size: 20pt">Here is some
   more text</p>
 <p style="font-size: 20pt;color:
   #0000FF" >Even more text</p>
</body>
</html>
```

Inline Styles: Example

inline-styles.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Inline Styles</title>
</head>
<body>
  <p>Here is some text</p>
<!--Separate multiple styles with a semicolon-->
  <p style="font-size: 20pt">Here is some
    more text</p>
  <p style="font-size: 20pt;color:
    #0000FF" >Even more text</p>
</body>
</html>
```



D:\Books\Iw3htp\examples\Presentation\presentation\New wo

Here is some text

Here is some more text

Even more text

CSS Cascade (Precedence)

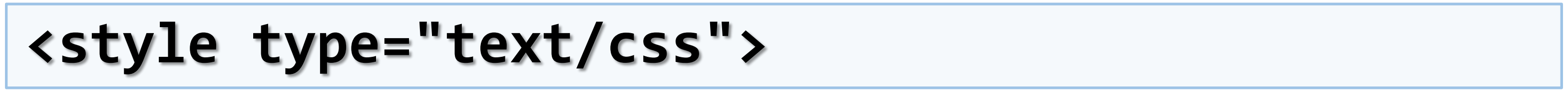
- There are browser, user and author stylesheets with "normal" and "important" declarations
  - Browser styles (least priority)
  - Normal user styles
  - Normal author styles (external, in head, inline)
  - Important author styles
  - Important user styles (max priority)

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

http://www.slideshare.net/maxdesign/css-cascade-1658158

Embedded Styles

- Embedded in the HTML in the <style> tag:



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

  - The <style> tag is placed in the <head> section of the document
  - type attribute specifies the MIME type
    - MIME describes the format of the content
    - Other MIME types include text/html, image/gif, text/javascript …
- Used for document-specific styles

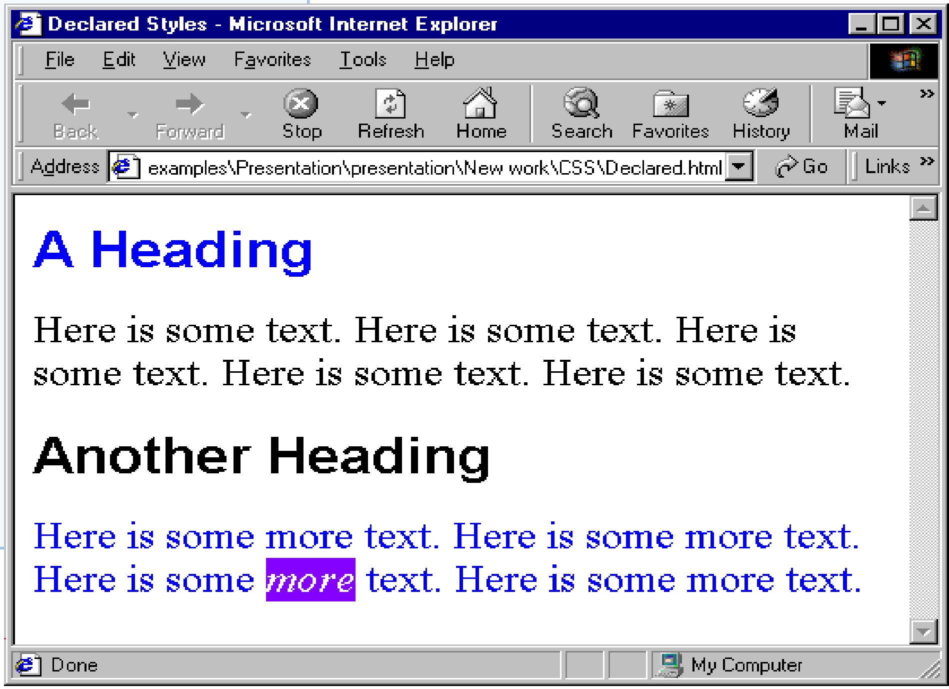
Embedded Styles: Example

embedded-stylesheets.html

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Style Sheets</title>
  <style type="text/css">
    em {background-color:#8000FF; color:white}
    h1 {font-family:Arial, sans-serif}
    p  {font-size:18pt}
    .blue {color:blue}
  </style>
<head>
```

Embedded Styles: Example

```
…
<body>
  <h1 class="blue">A Heading</h1>
  <p>Here is some text. Here is some text. Here
  is some text. Here is some text. Here is some
  text.</p>
  <h1>Another Heading</h1>
  <p class="blue">Here is some more text.
  Here is some more text.</p>
  <p class="blue">Here is some <em>more</em>
  text. Here is some more text.</p>
</body>
</html>
```



A Heading

Here is some text. Here is some text. Here is some text. Here is some text. Here is some text.

Another Heading

Here is some more text. Here is some more text.

Here is some *more* text. Here is some more text.

External CSS Styles

- External linking
  - Separate pages can all use a shared style sheet
  - Only modify a single file to change the styles across your entire Web site (see http://www.csszengarden.com/)
- link tag (with a rel attribute)
  - Specifies a relationship between current document and another document

  - link elements should be in the <head>

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

External CSS Styles

@import

- Another way to link external CSS files
- Example:

```
<style type="text/css">
  @import url("styles.css");
  /* same as */
  @import "styles.css";
</style>
```

- Ancient browsers do not recognize @import
- Use @import in an external CSS file to workaround the IE 32 CSS file limit

External Styles: Example

styles.css

```
/* CSS Document */

a           { text-decoration: none }

a:hover { text-decoration: underline;
       color: red;
       background-color: #CCFFCC }

li em   { color: red;
       font-weight: bold }

ul          { margin-left: 2cm }

ul ul       { text-decoration: underline;
       margin-left: .5cm }
```

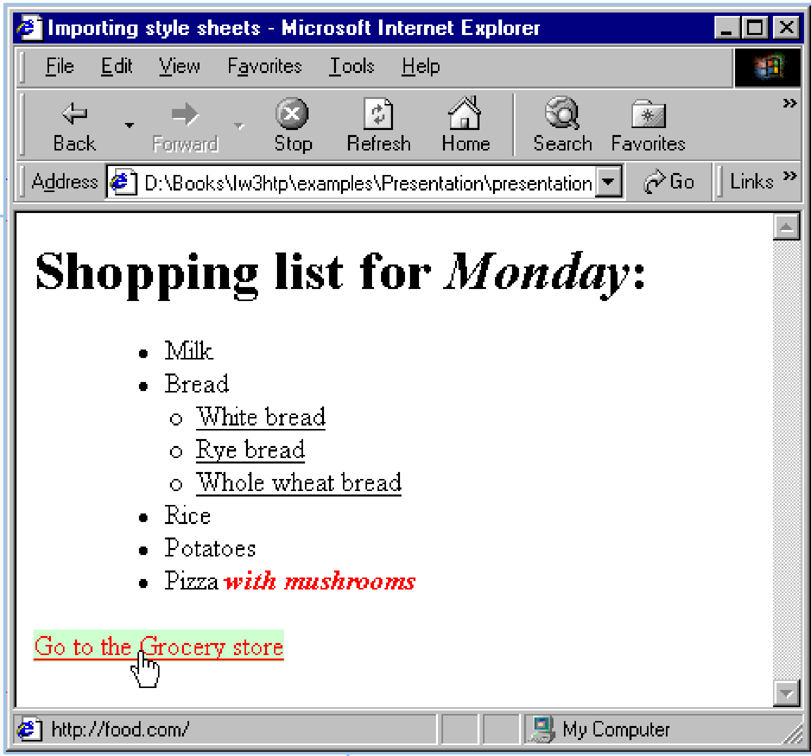
External Styles: Example

external-styles.html

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Importing style sheets</title>
  <link type="text/css" rel="stylesheet"
    href="styles.css"  />
</head>
<body>
  <h1>Shopping list for <em>Monday</em>:</h1>
  <li>Milk</li>
  …
```

External Styles: Example

```
  …
  <li>Bread
    <ul>
      <li>White bread</li>
      <li>Rye bread</li>
      <li>Whole wheat bread</li>
    </ul>
  </li>
  <li>Rice</li>
  <li>Potatoes</li>
  <li>Pizza <em>with mushrooms</em></li>
</ul>
<a href="http://food.com" title="grocery
  store">Go to the Grocery store</a>
</body>
</html>
```



Shopping list for *Monday*:

- Milk
- Bread
  - White bread
  - Rye bread
  - Whole wheat bread
- Rice
- Potatoes
- Pizza *with mushrooms*

Go to the Grocery store

Shorthand Font Property

- `font`
  - Shorthand rule for setting multiple font properties at the same time is equal to writing this:

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

```
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, no-repeat
- background-attachment
    - fixed / scroll
- 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 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

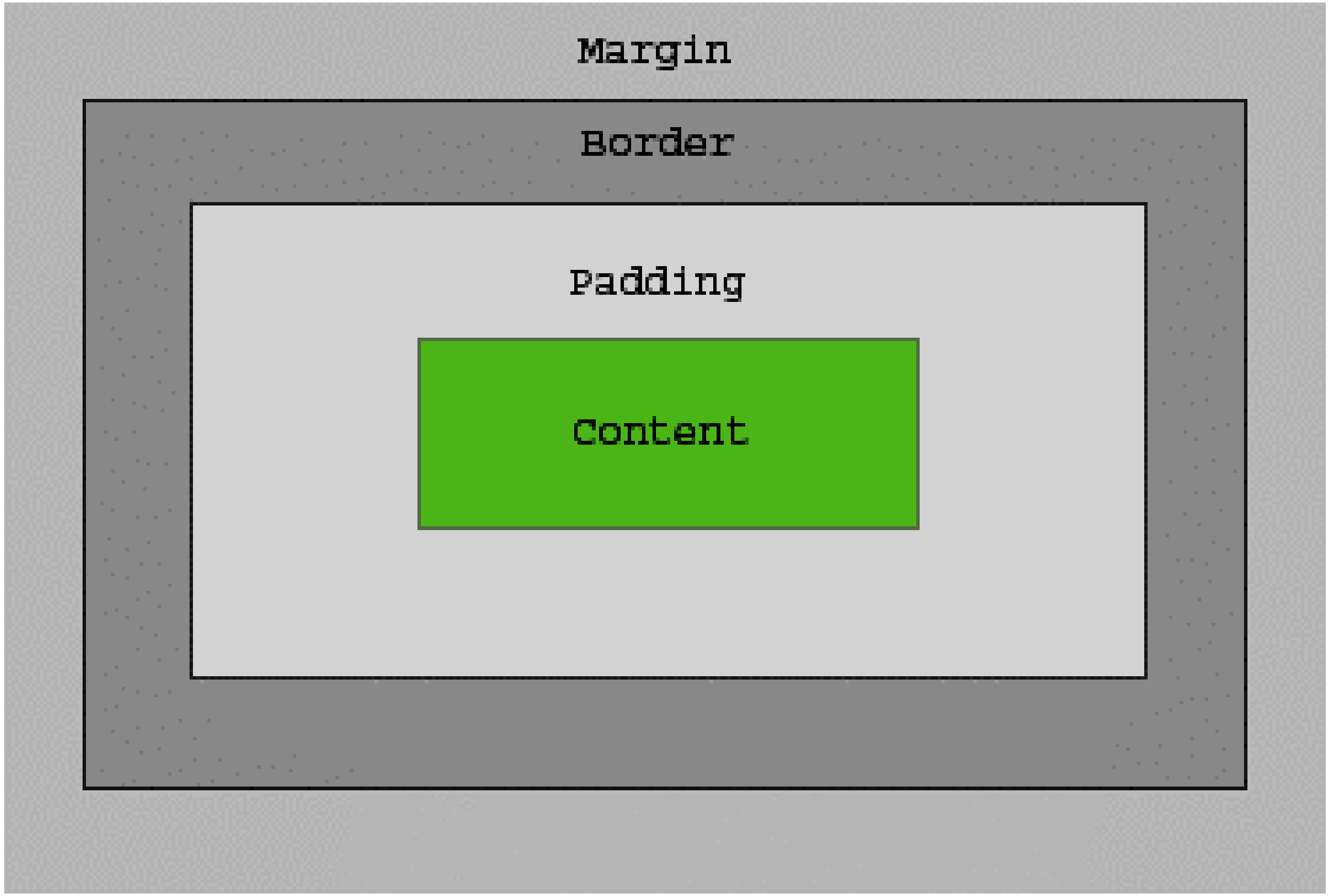
Width and Height

- 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.

Margin and Padding

- margin: 5px;
  Sets all four sides to have margin of 5 px;
- margin: 10px 20px;
  top and bottom to 10px, left and right to 20px;
- margin: 5px 3px 8px;
  top 5px, left/right 3px, bottom 8px
- margin: 1px 3px 5px 7px;
  top, right, bottom, left (clockwise from top)
- Same for padding

The Box Model



Benefits of using CSS

- More powerful formatting than using presentation tags
- Your pages load faster, because browsers cache the .css files
- Increased accessibility
- Pages are easier to maintain and update