HTML5 and CSS3 – The Future of the Web Programming

CSS

Sergio Luján Mora
Content

- Introduction
- Linking HTML and CSS
- Rules, selectors, and properties
- Text properties
- Background
- Links
- Box model
- Layout

Introduction

CSS (Cascading Style Sheets):
- A style sheet is a set of instructions each of which tells a browser how to draw a particular element on a page
- HTML documents are a collection of elements arranged in a hierarchy
University of Alicante

Department of Software and Computing Systems

The Department of Software and Computing Systems teaches the following courses: Programming, Object-Oriented Programming, Web Programming, and Databases and Software Engineering.

Teaching Staff
Introduction

• Hierarchy → inheritance:
  – If you apply style to an element (parent) which contains other elements (children) then this will be inherited by the elements inside.

• Rules have two parts: a selector and a declaration
  – The selector tells a browser which elements in a page will be affected by the rule. There are a number of different types of selector.
  – The declaration tells the browser which set of properties to apply. There are many different properties.
Introduction

<p>The Department of Software and Computing Systems teaches the following courses: <em>Programming, Object-Oriented Programming, Web Programming, and Databases and Software Engineering</em>.</p>

<h2>Teaching Staff</h2>
Introduction

```html
<style type="text/css">
h1 {color: blue;}

h2 {color: red;}

p {font-size: 1.4em;}
</style>
```
Exercise

• Give color green for the paragraph

Exercise - Solution

• One solution:
  p {font-size: 1.4em;}
  p {color: green;}

Also (better):
  p {font-size: 1.4em; color: green;}

Exercise

• Give color green for the paragraph
Introduction

• Some examples:
  
  body {background: white; color: black; font-size: 15px;}
  h1 {background: blue; font-size: 30px;}
  p {background: yellow;}
  strong {background: red;}
  em {background: red; color: yellow;}

• If the value has two or more words:
  p {font-family: "Times New Roman", serif}

• Comments (1 or more lines):
  
  /*
   * Paragraph */
  p {
    text-align: center; /* Center */
    color: black; /* Black */
    font-family: Arial; /* Font face */
  }
Introduction

• Different versions:
  – CSS1: CSS level 1. 1996.
    • http://www.w3.org/TR/REC-CSS1
    • http://www.w3.org/TR/REC-CSS2/
  – CSS3: CSS level 3. (Working draft)
• Problem: incompatibilities between browsers

• More information:
  – W3C: http://www.w3.org/
  – W3 Schools: http://www.w3schools.com/
Linking HTML and CSS

• Three ways of making a style sheet affect the appearance of an HTML document:
  – External (linking):
    
    ```html
    <head>
    <link rel="stylesheet" href="style.css" type="text/css" media="screen" />
    </head>
    ```
  – Internal (embedded):
    
    ```html
    <head>
    <style type="text/css" media="screen">
    p {text-indent: 10pt}
    </style>
    </head>
    ```
  – Inline:
    
    ```html
    <p style="text-indent: 10pt">indented paragraph</p>
    ```
Exercise

• Change the previous example: link the web page to an external style sheet

Linking HTML and CSS

• External:
  – Many pages can be linked to a single style sheet
  – Making changes to this style sheet can then change the appearance of all the web pages linked to it

• Internal:
  – When the style sheet is used for only one web page

• Inline:
  – When the style is used for only one HTML element
Linking HTML and CSS

- The most popular method is the external style sheet:
  
  ```html
  <link rel="stylesheet" href="style.css" type="text/css" media="screen" />
  ```

- `href`:
  - Tells the browser where to locate the style sheet, with either a relative or absolute URL

- `rel`:
  - Tells the browser what to expect
    - `stylesheet`
    - `alternate stylesheet`

If you define different alternative style sheets, you have to assign a title to each style sheet:

```html
<link rel="alternate stylesheet" href="style.css" type="text/css" media="screen" title="Style 1" />
```

- Example: webpage of Spanish Social System (Seguridad Social)
Linking HTML and CSS

```html
<link rel="stylesheet" href="/ireach/internet/css/estilos_B.css" media="screen" type="text/css">

<link rel="alternate stylesheet" href="/ireach/internet/css/estilosHC_B.css" media="screen" type="text/css" title="2.Alto Contraste">

<link rel="alternate stylesheet" href="/ireach/internet/css/estilosHT_B.css" media="screen" type="text/css" title="3.Texto Grande">

<link rel="stylesheet" href="/ireach/internet/css/estilosP_B.css" media="print" type="text/css">
```
Linking HTML and CSS

- **type:**
  - Tells the browser the type of document linked
  - Common values:
    - text/css
    - text/javascript

- **media:**
  - Tells the browser the type of device the style sheet is for:
    - screen: Computer display
    - print: Printer
    - projection: Projector
    - aural: Speech synthesizer
    - braille: Braille line
    - tty: Console (text) display
    - tv: Television
    - all: All devices (default value)
Rules, selectors, and properties

- **Rules**: the selector, followed by the set of properties, which are surrounded by curly braces (that is { and })

h1 {color: #FF0000; background: #00FF00}

- **Selector**:
  - Type: select every instance of the specified type of HTML element (tag)
  - Class: class name preceded by “.”
  - ID: identifier name preceded by “#”
  - Pseudo-classes: name of the pseudo-class
Rules, selectors, and properties

- Property and value: the **property name** followed by a **colon**, a **space** (optional) and then the **value** (if necessary, with its **unit**, though no **space** before the unit!)

- The last property/value pair doesn’t need the separator **semi-colon ( ; )**

---

Rules, selectors, and properties

- Advices:
  - Make sure you get the property name exactly right: it’s got to be color, not colour or collor
  - All CSS properties can only take a specified range of values: get used to consulting the specification
  - Don’t get the punctuation wrong (don’t forget the semi-colon between properties)
Exercise

• Write a new webpage
• Give a background color (light gray) and a color (dark blue) to the whole page
  – Selector: html or body
  – Property: background-color
    • Value: #999999
  – Property: color
    • Value: #0000AA

Exercise - Solution

```html
html {
  background-color: #999999;
  color: #0000AA;
}
```

• Tips:
  – The style sheet is easier to read if you put each property on a new line
  – It's very easy to forget semi-colons (;). Always put that extra one on the last property in a statement. You won't forget to do it when you add a new property later on
What are style sheets?

A style sheet is a set of instructions each of which tells a browser how to draw a particular element on a page. It's very important to grasp this idea of HTML elements when you are working with CSS. Well-formed HTML documents are a collection of elements arranged in a kind of containment hierarchy.

Rules, selectors, and properties

- **Class**: is an attribute we can add to HTML elements so that we can identify them from the style sheet, using a class selector, and give them their own style

```html
<p class="footer">
  color: blue;
</p>
```

- The form of a class selector is very simple, and it would select the specified element with the class attribute class-name

```css
p.footer {
  color: blue;
}
```
Rules, selectors, and properties

- If you don’t specify an element, the class name applies to all the elements:
  
  ```html
  .important {color: red;}
  
  <p class="important">
  Bla, bla, bla.
  </p>
  
  <ul class="important">
    <li>Bla, bla.</li>
    <li>Bla, bla.</li>
  </ul>
  ```

Exercise

- Write a new webpage with three paragraphs
- Define three classes for the three paragraphs:
  - `<p class="normal">` → Color dark blue
  - `<p class="important">` → Color red, background color yellow
  - `<p class="extra">` → Color green, background color gold
<html>
<head><title>Three paragraphs</title>
<style type="text/css">
 .normal {color: #0000AA;}
 .important {color: red; background-color: yellow;}
 .veryimportant {color: yellow; background-color: red;}
</style>
<body>
<p class="normal">This is the first paragraph. Bla, bla, bla, bla.</p> 
<p class="important">This is the second paragraph. Bla, bla, bla, bla.</p> 
<p class="veryimportant">This is the third paragraph. Bla, bla, bla, bla.</p> 
</body></html>
Rules, selectors, and properties

- **ID**: is an attribute we can add to HTML elements so that we can uniquely identify them from the style sheet, using an ID selector, and give them their own style
  ```html
  <p id="paragraph-1">
  ```
- The form of an ID selector is very simple, and it would select the specified element with the ID attribute value
  ```css
  p#paragraph-1 {
    color: blue;
  }
  ```

Exercise

- Write a new webpage with three titles and four paragraphs
Exercise

- The main title (h1) is color white on black
- The chapter titles (h2) are blue on yellow
- The default color of paragraphs is green
- The first paragraph of the first chapter is black
- An important paragraph is yellow on red
Exercise - Solution

<html>
<head><title>Three paragraphs</title>
<style type="text/css">
p {color: green;}
#title {color: white; background-color: black;}
#first-chapter-par {color: black;}
.chapter-title {color: blue; background-color: yellow;}
.important {color: yellow; background-color: red;}
</style>
<body>
<h1 id="title">The title of the book</h1>
<h2 class="chapter-title">The first chapter</h2>
<p id="first-chapter-par" >This is the first paragraph. Bla, bla, bla, bla.</p>
<p class="important">This is the second paragraph. Bla, bla, bla, bla.</p>
<h2 class="chapter-title">The second chapter</h2>
<p>This is the third paragraph. Bla, bla, bla, bla.</p>
<p class="important">This is the third paragraph. Bla, bla, bla, bla.</p>
</body>
</html>
Text properties

- `background-color`: background color of element
- `color`: color of text
Text properties

• Colors:
  – Name of color → red
  – rgb(x,x,x) → rgb(255,0,0)
  – rgb(y%, y%, y%) → rgb(100%,0%,0%)
  – #rrggb → #ff0000
  – #rgb = #rrggb → #f00 = #ff0000

Text properties

• font-family: specifies a list of one or more fonts using the family name of each
  – The font names are separated by commas
  – A browser then uses the first font in the list that is installed on its system
  – At the end of that list you should always put one of five generic font names:
    • serif (e.g. Times)
    • sans-serif (e.g. Helvetica)
    • cursive (e.g. Zapf-Chancery)
    • fantasy (e.g. Western)
    • monospace (e.g. Courier)
  – If you want to use a font with more than a single word name make sure you put it between quote marks, like this: "Times New Roman"
Text properties

- `font-size`: can take what is referred to in CSS as length values

Text properties

- **Units:**
  - %: percentage
  - in: inches
  - cm: centimeters
  - mm: millimeters
  - em: equal to the current size of text
  - ex: equal to letter “x” height
  - pt: point (1 pt = 1/72 inches)
  - pc: pica (1 pc = 12 points)
  - px: pixels
Text properties

- **Units:**
  - **Keywords:**
    - xx-small, x-small, small, medium, large, x-large, xx-large
    - smaller, larger
  - **Absolute:**
    - in, cm, mm, pt, pc, px
  - **Relative:**
    - %, em, ex

- **Best unit:** **em**
  - If you set the font-size using em units, they will always remain relative to the main text on the page, whatever that might be
  - For example, if you set `<h1>` to 2em, it will be twice as big as the rest of the text on the page
Text properties

• **font-style**: sets the style of the font
  - *normal*: default, normal font
  - *italic*: an italic font
  - *oblique*: an oblique font

• **font-variant**: displays text in normal or small-caps font
  - *normal*
  - *small-caps*

• **font-weight**: controls the boldness of text
  - *normal*
  - *bold*
  - *bolder*
  - *lighter*
  - *100, 200, ..., 900*
Text properties

- `text-align`: controls the justification of text
  - left
  - right
  - center
  - justify

Text properties

- `text-decoration`: for underlining and striking through text
  - none
  - underline
  - overline
  - line-through
  - blink
Text properties

- **text-transform**: controls the letters in an element
  - none
  - capitalize: each word in a text starts with a capital letter
  - uppercase
  - lowercase

- **letter-spacing**: controls the spacing between characters
- **word-spacing**: controls the spacing between words
- **line-height**: sets the distance between lines
Exercise

- Use the different text properties:
  - Add some special styles to make the headings stand out more from the main text
  - Make the text in the paragraphs look more clean and attractive

Background

- background-color: defines the background color
- background-image: puts an image in the background
  - background-image: url(logo.png);
- background-repeat: defines how the background image is going to be repeated
- background-position: defines the position of the background image
Background

- `background-repeat:`
  - `repeat:` the image will tile to fill the whole element
  - `repeat-x:` the image will only repeat horizontally across the element
  - `repeat-y:` the image will only repeat vertically down the element
  - `no-repeat:` only a single instance of the element will appear

Exercise

- Create a new web page
- Put an image as background image
- Try the different “repeat” values
Background

- `background-position: specify 2 values, the first for where you want the image to be horizontally, the second for where it will be vertically`
  - Length values
  - Keyword values: top, bottom, left, right, center

- Example:
  - `background-position: center center` - places the image right in the center of the element, both horizontally and vertically
  - `background-position: top left` - places the image in the top left corner
  - `background-position: right bottom` - places the image in the bottom right corner
Exercise

- Create a new web page
- Put an image as background image in the center of the web page

Exercise

- Create a new web page with three paragraphs
- Put an image as background image in each one of the paragraphs
Links

- Links can have four different states:
  - \textit{link}: this is the normal state
  - \textit{visited}: when a browser has visited that destination recently
  - \textit{hover}: while the cursor is over the link
  - \textit{active}: while the link is being clicked

- We can create rules that apply to links in any of these states

Links

- Pseudo-classes:
  - \texttt{a:link} {...}
  - \texttt{a:visited} {...}
  - \texttt{a:hover} {...}
  - \texttt{a:active} {...}
Exercise

- Give links in both their normal and visited state a background-color of #95b7cd and make their text color the same as the regular text on the page. A user can't tell whether a link is to a page they have recently viewed or not.

- Give links in their hover state a background-color of #aaddee

- Give links in their active state a background-color of #3cc7f0

Exercise - Solution

```html
a:link {
  color: #666666;
  background-color: #95b7cd;
}

a:visited {
  color: #666666;
  background-color: #95b7cd;
}

a:hover {
  background-color: #aaddee;
}

a:active {
  background-color: #3cc7f0;
}
```

Roll over and click the links to see the effect of the hover and active states.
Exercise

- Most browsers have a default setting for links, which is to underline them. How can we change this?
- How can we draw a line through the text of visited links?

Exercise - Solution

```css
a:link {
    ...
    text-decoration: none;
}

a:visited {
    ...
    text-decoration: line-through;
}
```
Box model

- Boxes: each element of the web page is represented by a “box”

- **margin** is the distance between the edge of an element and its adjacent elements
- **padding** is the distance between the edge of an element and its content
Box model

- Padding, border, and margin are divided into four edges: top, bottom, left, and right.
- Therefore, we have: `border-left`, `border-right`, `border-top` and `border-bottom` (and the same for margin and padding).
- Padding, border, and margin apply the values to all four edges.

Box model

- Borders can be applied either to all edges of an element, or each edge individually.
Box model

- There are three characteristics of a border you can control:
  - Its style, using values like
    - solid
    - dotted
    - dashed
    - double
  - Its width, using all the usual length values
  - Its color, using the color values

Exercise

- Create a new web page
- Write four paragraphs
- Apply a different border style and background to each paragraph
A style sheet is a set of instructions each of which tells a browser how to draw a particular element on a page. It's very important to grasp the idea of HTML elements when you are working with CSS. Well-formed HTML documents are a collection of elements arranged in a kind of containment hierarchy.

Exercise - Solution

.p1 {
    background-color: #999999;
    border: solid 5px green;
}
.p2 {
    background-color: #aa6666;
    border: dotted 5px green;
}
.p3 {
    background-color: #66aa66;
    border: dashed 5px green;
}
.p4 {
    background-color: #6666aa;
    border: double 5px green;
}
Exercise 1

- Read exercise document: Curriculum vitae

Layout

- `<span>` and `<div>` are the main building blocks used in CSS page layouts
- They are simply generic HTML block element
  - `span`: inline
  - `div`: block
- You can wrap it around the different blocks of content you want to position on your page
Layout

- `<span>` and `<div>` need unique `id` attributes so that we can identify them and give them positioning properties in the style sheet.
- `ids` must be unique in any single HTML document, otherwise HTML document is not valid.

Example:

- HTML:
  ```html```
  ```
  <div id="header">
    ...
  </div>
  ```
  ```html```
- CSS:
  ```css```
  ```
  #header {background-color: gray; color: red}
  ```
  ```css```
Layout

- Basic three column layout with a navigation bar (navbar) on the left and a sidebar on the right
  - It uses a combination of static, relative and absolute positioning

```html
<div id="header"></div>
<div id="content">
  <div id="navbar"></div>
  <div id="main-text"></div>
  <div id="sidebar"></div>
</div>
```
Layout

- Both #header and #main-text are going to be positioned statically
  - They simply flow down the page, one after the other, in the same order as they occur in the XHTML

```html
body {
  margin: 0;
  background-color: #aaaaaa;
  text-align: center;
}

#header {
  background-color: #0000ff;
  color: #ffffff;
  text-align: center;
  font-size: 2em;
}

#content {
  position: relative;
}
```
#navbar {
    position: absolute;
    top: 0;
    left: 0;
    width: 198px;
    background-color: #ff0000;
}

#main-text {
    margin-left: 198px;
    margin-right: 198px;
    background-color: #ffffff;
}

#sidebar {
    position: absolute;
    top: 0;
    right: 0;
    width: 198px;
    background-color: #00ff00;
}
Exercise 2

- Read exercise document: 2-columns layout for curriculum vitae