

# Putting on the style

intermediate web design and development

Week 1 – Getting started

Glenn Clarkson

# Learning outcomes for tonight

## Understand

- The Browser Wars
- What is a style sheet?
- Separating presentation and style from structure and content
- Style declarations
- Shorthand notation

# History of web browsers

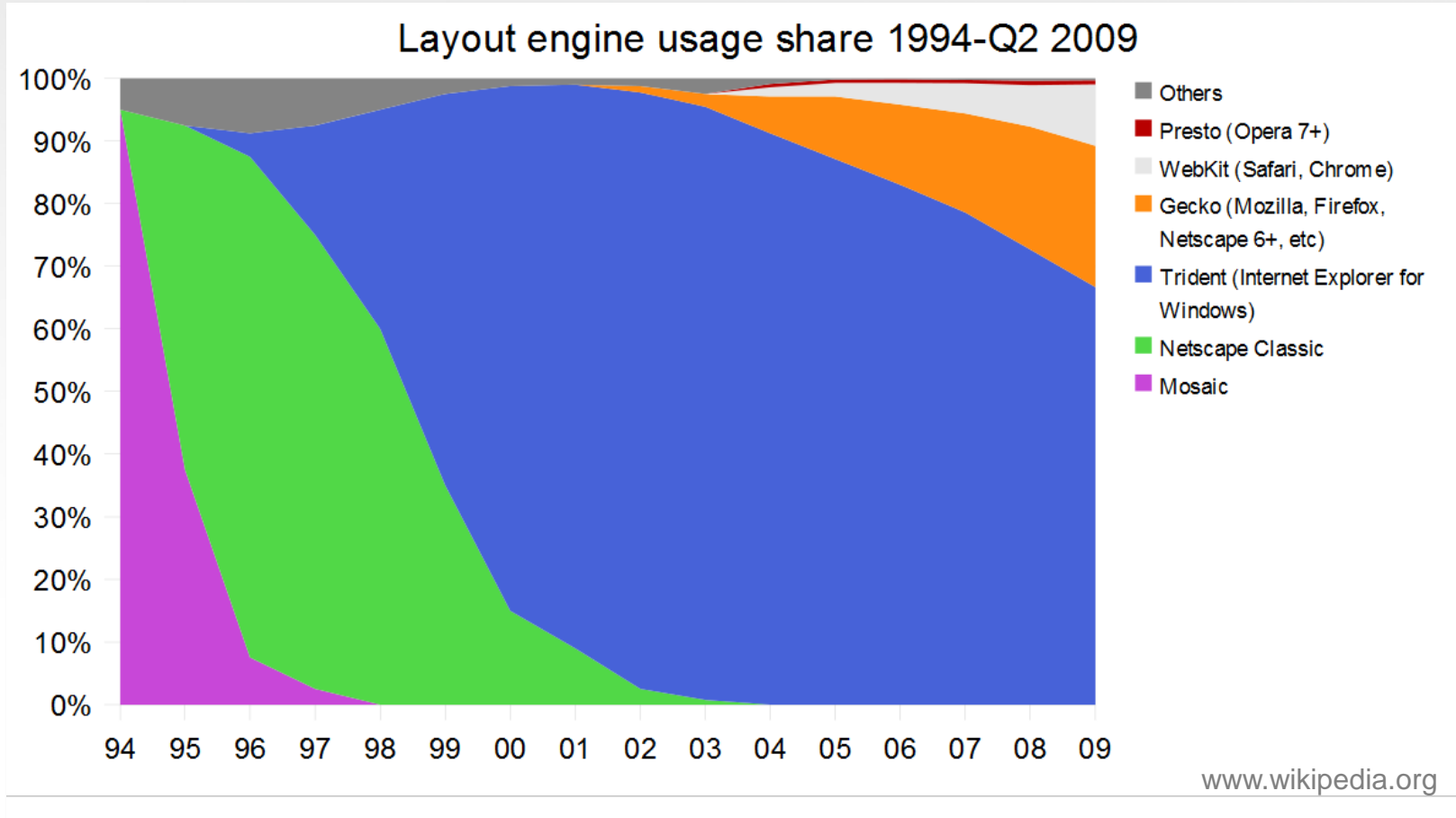
- NCSA\* Mosaic V1.0 released in September 1993
- Netscape Navigator – October 1994
- IBM WebExplorer – 1994
- Internet Explorer – August 16 1995
- UdiWWW – 1995 (first browser to support HTML, abandoned in 1996)

## 1995: “The Browser Wars”

- Netscape Navigator most widely used until Microsoft gave Explorer away for free, first as an update to Windows 95 and in November 1995 as a free download
- Development was rapid and new features routinely added

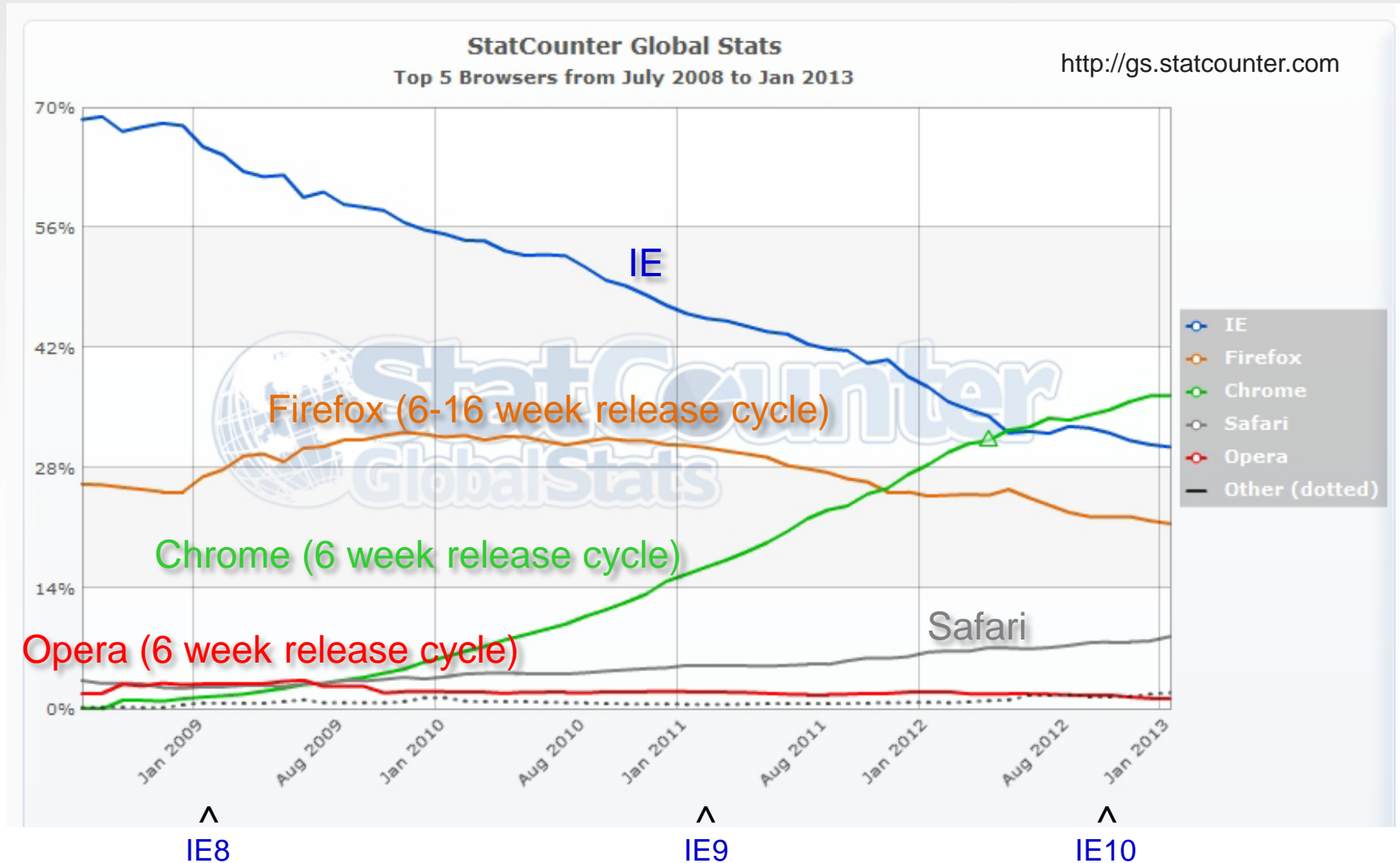
\*NCSA – National Centre for Supercomputing Applications, University of Illinois

# The Browser Wars



^ ^ ^ ^ ^ ^ ^  
IE2 IE3 IE4 IE5 IE6 IE7 IE8

# Market share today



# Browser defaults

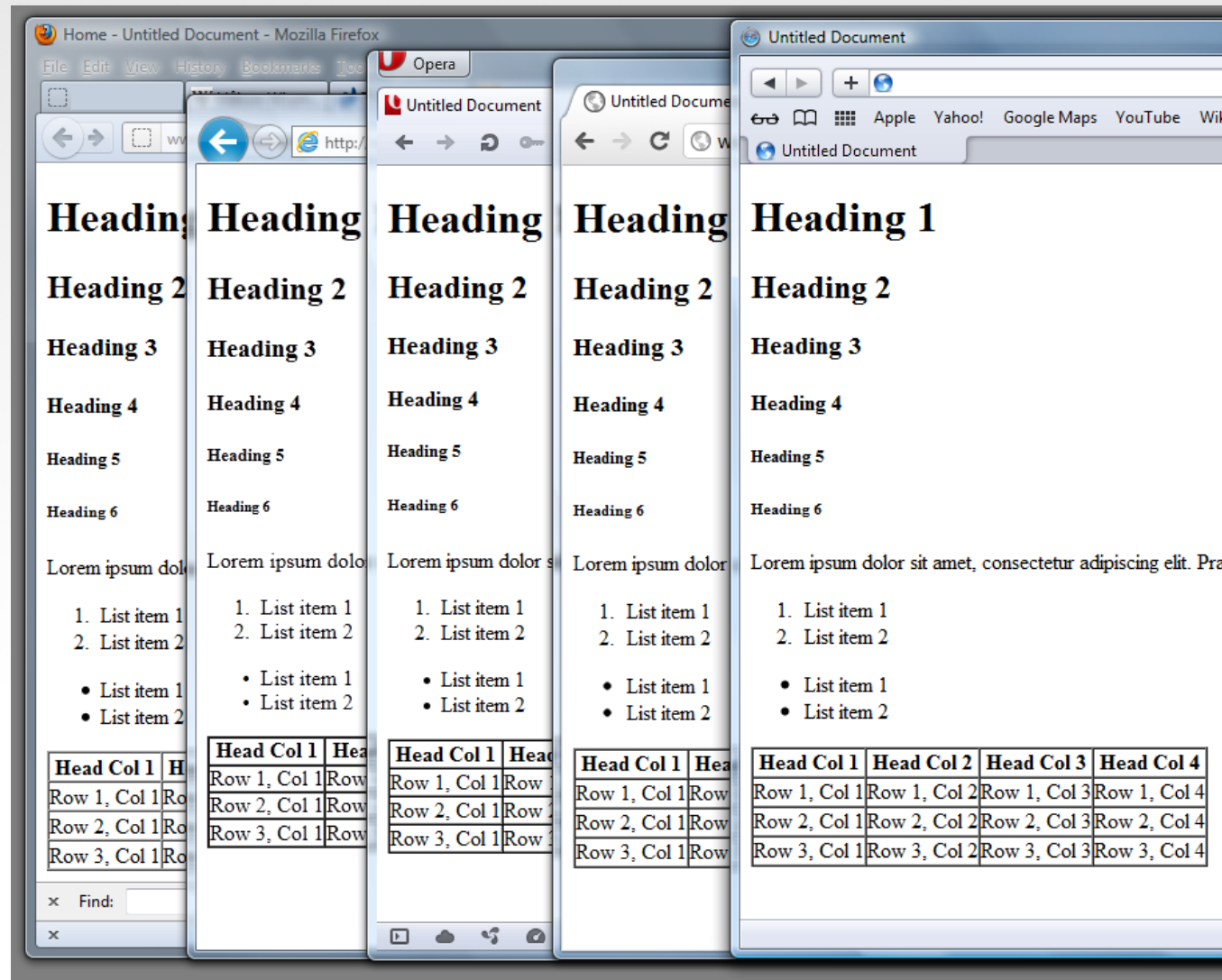
Firefox 8.0.1

Explorer 9

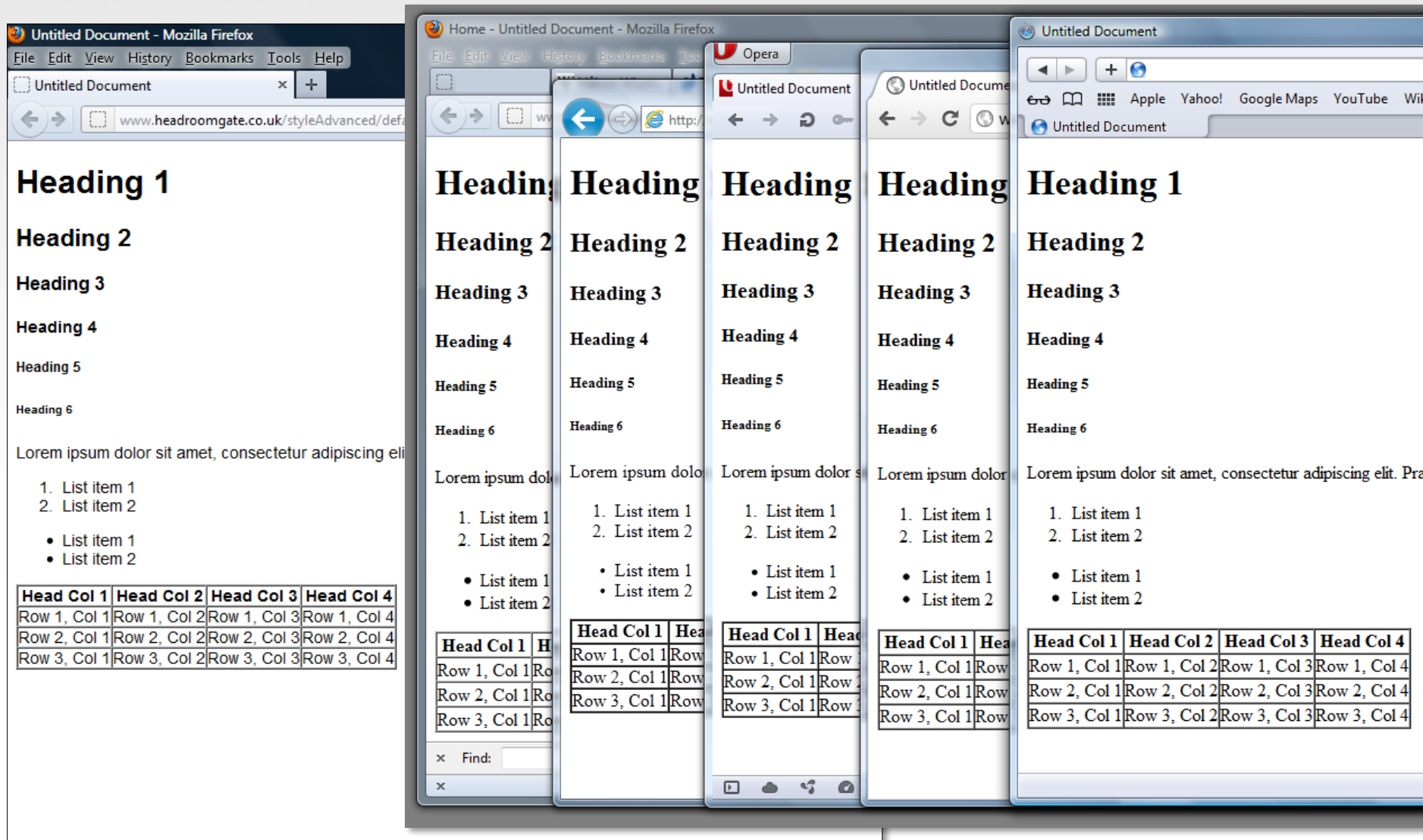
Opera 11.60

Chrome 16.0

Safari 5.1.1



# Browser defaults



# Style sheets

The first draft of *Cascading HTML Style Sheets* was published by Håkon Wium Lee (CERN) in October 1994

- The first commercial browser to support CSS was Microsoft's Internet Explorer 3 which was released in August 1996
- The next browser to announce support for CSS was Netscape's Navigator V4 in June 1997 (support was only partial)
- Opera was the third browser to support CSS in July 1998
- Both Apple's Safari (2003) and Google's Chrome (2008) have supported CSS from V1

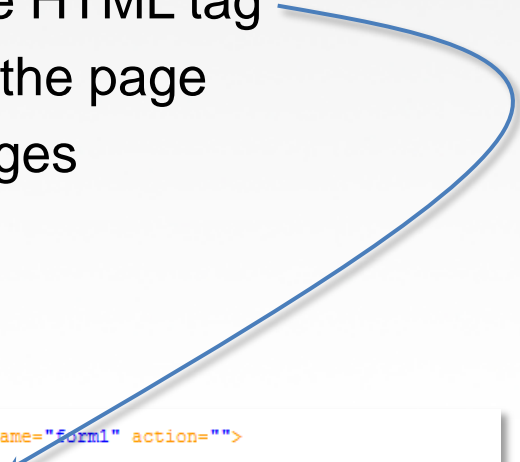


# Style sheets

- A style sheet is a set of instructions each of which tell a browser how to draw or render a particular element on a page
- The express intention of style sheets are to separate style and appearance from content and structure
- Before style sheets designers used HTML tables to control layout resulting in:
  - Longer page-load times
  - Use of placeholder images, further slowing performance
  - Nightmare maintenance – even small changes breaking the layout
  - Pages inaccessible to users not using a graphical browser

# Style sheets

- Style sheets are simple text files and instructions that can be created and edited in a simple text editor
- Three ways of imposing style:
  - inline style inline on the page with the HTML tag
  - embedded style embedded in the head of the page
  - external document linked to multiple pages



```
id= container >  
<form method="post" name="form1" action="">  
  <fieldset>  
    <legend style="font-size:16px;">Add a new page</legend>  
    <label for="pageTitle">Page Title:</label>  
    <input type="text" name="pageTitle" value="" placeholder="Page Title">  
    <label for="pageType">Page type:</label>  
    <select name="pageType">
```

# Style sheets

- Style sheets are simple text files and instructions that can be created and edited in a simple text editor
- Three ways of imposing style:
  - inline style inline on the page with the HTML tag
  - embedded style embedded in the head of the page
  - external document linked to multiple pages

```
<!DOCTYPE HTML>
<html>
<head>
<meta charset="utf-8">
<style>
body {
width:800px;
font-family:Arial, Helvetica, sans-serif;
}
h1 {
color:#001c54;
font-size:20px;
}
h2 {
color:#001c54;
font-size:16px;
}
</style>
<title>Add a new page</title>
</head>
<body>
<div id="container">
<form method="post" name="form1" action="">
<fieldset>
<legend>Add a new page</legend>
<label for="pageTitle">Page Title:</label>
<input type="text" name="pageTitle" value="" placeholder="add new page title"><br><br>
</fieldset>
</div>
</body>
</html>
```

```
<div id="container">
<form method="post" name="form1" action="">
<fieldset>
<legend style="font-size:16px;">Add a new page</legend>
<label for="pageTitle">Page Title:</label>
<input type="text" name="pageTitle" value="" placeholder="add new page title">
<label for="pageType">Page type:</label>
<select name="pageType">
</select>
</fieldset>
</div>
```

# Style sheets

- Style sheets are simple text files and instructions that can be created and edited in a simple text editor
- Three ways of imposing style:
  - inline style inline on the page with the HTML tag
  - embedded style embedded in the head of the page
  - external document linked to multiple pages

```
/* CSS Document */
body {
  width:800px;
  font-family:Arial, Helvetica, sans-serif;
}
h1 {
  color:#001c54;
  font-size:20px;
}
h2 {
  color:#001c54;
  font-size:16px;
}
#headline {
  float: left;
  width:195px;
  background-color:#001c54;
  padding-left:25px;
}
#dashboard {
  float: left;
  width:106px;
  padding-left:14px;
  padding-top:4px;
}
#content {
  float: left;
  width:646px;
  padding-right:15px;
}
#colspan {
  color:#FFFFFF;
}
#submit {
  color:#FFFFFF;
  font-size:8px;
}
```

```
<!DOCTYPE HTML>
<html>
<head>
  <meta charset="utf-8">
  <title>Add a new page</title>
</head>
<body>
  <div id="container">
    <form method="post" name="form1" action="">
      <fieldset>
        <legend>Add a new page</legend>
        <label for="pageTitle">Page Title:</label>
        <input type="text" name="pageTitle" value="" placeholder="add new page title">
        <label for="pageType">Page type:</label>
        <select name="pageType">
          <option value="Text">Text</option>
          <option value="Gallery">Gallery</option>
          <option value="Contact">Contact</option>
          <option value="Product">Product</option>
        </select>
        <br>
        <label for="ArtistID">Artist:</label>
        <select name="ArtistID">
          <option value=""></option>
        </select>
        <br>
        <label for="mainimg">Product image:</label>
        <select name="mainimg">
          <option value=""></option>
        </select>
        <br>
        <label for="pageText">Page text:</label>
        <textArea name="pageText" placeholder="add text">
        <label for="submit">Submit</label>
        <input name="newPage" type="submit" value="Add new page">
      </fieldset>
    </form>
    <p><a href="dbIndex.php">Return to the dashboard</a></p>
  </div>
</body>
</html>
```

```
<!DOCTYPE HTML>
<html>
<head>
  <meta charset="utf-8">
  <title>Add a new page</title>
</head>
<body>
  <div id="container">
    <form method="post" name="form1" action="">
      <fieldset>
        <legend>Add a new page</legend>
        <label for="pageTitle">Page Title:</label>
        <input type="text" name="pageTitle" value="" placeholder="add new page title">
        <label for="pageType">Page type:</label>
        <select name="pageType">
          <option value="Text">Text</option>
          <option value="Gallery">Gallery</option>
          <option value="Contact">Contact</option>
          <option value="Product">Product</option>
        </select>
        <br>
        <label for="ArtistID">Artist:</label>
        <select name="ArtistID">
          <option value=""></option>
        </select>
        <br>
        <label for="mainimg">Product image:</label>
        <select name="mainimg">
          <option value=""></option>
        </select>
        <br>
        <label for="pageText">Page text:</label>
        <textArea name="pageText" placeholder="add text">
        <label for="submit">Submit</label>
        <input name="newPage" type="submit" value="Add new page">
      </fieldset>
    </form>
    <p><a href="dbIndex.php">Return to the dashboard</a></p>
  </div>
</body>
</html>
```

```
<div id="container">
  <form method="post" name="form1" action="">
    <fieldset>
      <legend style="font-size:16px;">Add a new page</legend>
      <label for="pageTitle">Page Title:</label>
      <input type="text" name="pageTitle" value="" placeholder="add new page title">
      <label for="pageType">Page type:</label>
      <select name="pageType">
        <option value="Text">Text</option>
        <option value="Gallery">Gallery</option>
        <option value="Contact">Contact</option>
        <option value="Product">Product</option>
      </select>
      <br>
      <label for="ArtistID">Artist:</label>
      <select name="ArtistID">
        <option value=""></option>
      </select>
      <br>
      <label for="mainimg">Product image:</label>
      <select name="mainimg">
        <option value=""></option>
      </select>
      <br>
      <label for="pageText">Page text:</label>
      <textArea name="pageText" placeholder="add text">
      <label for="submit">Submit</label>
      <input name="newPage" type="submit" value="Add new page">
    </fieldset>
  </form>
  <p><a href="dbIndex.php">Return to the dashboard</a></p>
</div>
```

# Style sheets

Style sheets specify how a document is to be presented on different media

- **All**: suitable for all devices
- **Braille**: for Braille tactile feedback devices
- **Embossed**: for paged Braille printers
- **Handheld**: for handheld devices (small screen, limited bandwidth)
- **Print**: for documents viewed on screen in print preview mode
- **Projection**: intended for projected presentations
- **Screen**: intended primarily for colour computer screens
- **Speech**: intended for speech synthesizers
- **tty**: for media using a fixed-pitch character grid, such as teletypes
- **tv**: television-type devices (low resolution, colour, limited-scrollability screens, sound available).

# Style sheets

Style sheets specify how a document is to be presented on different media

- **All**: suitable for all devices
- **Braille**: for Braille tactile feedback devices
- **Embossed**: for paged Braille printers
- **Handheld**: for handheld devices (small screen, limited bandwidth)
- **Print**: for documents viewed on screen in print preview mode
- **Projection**: intended for projected presentations
- **Screen**: intended primarily for colour computer screens
- **Speech**: intended for speech synthesizers
- **tty**: for media using a fixed-pitch character grid, such as teletypes
- **tv**: television-type devices (low resolution, colour, limited-scrollability screens, sound available).

# Style sheets

- Style sheets should be completely independent of the document's structure
- Apply initial document coding with only the structure and content in mind
- Structure code consistently throughout the site
- Produces lean well-formed mark-up that is easier to style
- Keep embedded/inline styling to the absolute minimum (ideally none at all)



# Zen Garden

The Beauty of CSS Design



*A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.*

Download the sample [html file](#) and [css file](#)

## The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

## So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks have been demonstrated by structurists and coders. Designers have yet to make their mark. This needs to change.

## Participation

Graphic artists only please. You are modifying this page, so strong CSS skills are necessary, but the example files are commented well enough that even CSS novices can use them as starting points. Please see the [CSS Resource Guide](#) for advanced tutorials and tips on working with CSS.

You may modify the style sheet in any way you wish, but not the HTML. This may seem daunting at first if you've never worked this way before, but follow the listed links to learn more, and use the sample files as a guide.

Download the sample [html file](#) and [css file](#) to work on a copy locally. Once you have completed your masterpiece (and please, don't submit half-finished work) upload your .css file to a web server under your control. [Send us a link](#) to the file and if we choose to use it, we will spider the associated images. Final submissions will be placed on our server.

select a design:

[Under the Sea! by Eric Stoltz](#)

[Make 'em Proud by Michael McAgthon and Scotty Reifsnyder](#)

[Orchid Beauty by Kevin Addison](#)

[Oceanscape by Justin Gray](#)

[CSS Co., Ltd. by Benjamin Klemm](#)

[Sakura by Tatsuya Uchida](#)

[Kyoto Forest by John Politowski](#)

[A Walk in the Garden by Simon Van Hauwermeiren](#)

archives:

[next designs »](#)

[View All Designs](#)

resources:

[View This Design's CSS](#)

[CSS Resources](#)

[FAQ](#)

[Submit a Design](#)

[Translations](#)



# Zen Garden

The Beauty of CSS Design



*A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.*

Download the sample [html file](#) and [css file](#)

## The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

## So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks have been demonstrated by structurists and coders. Designers have yet to make their mark. This needs to change.

## Participation

Graphic artists only please. You are modifying this page, so strong CSS skills are necessary, but the example files are commented well enough that even CSS novices can use them as starting points. Please see the [CSS Resource Guide](#) for advanced tutorials and tips on working with CSS.

You may modify the style sheet in any way you wish, but not the HTML. This may seem daunting at first if you've never worked this way before, but follow the listed links to learn more, and use the sample files as a guide.

Download the sample [html file](#) and [css file](#) to work on a copy locally. Once you have completed your masterpiece (and please, don't submit half-finished work) upload your .css file to a web server under your control. [Send us a link](#) to the file and if we choose to use it, we will spider the associated images. Final submissions will be placed on our server.

[Under the Sea! by Eric Stoltz](#)

[Hate Anonymous by Michael McAgthon and Scotty Reifsnyder](#)

[Orchid Beauty by Kevin Addison](#)

[Oceanscape by Justin Gray](#)

[CSS Co., Ltd. by Benjamin Klemm](#)

[Sakura by Tatsuya Uchida](#)

[Kyoto Forest by John Politowski](#)

[A Walk in the Garden by Simon Van Hauwermeiren](#)

archives:

[next designs »](#)

[View All Designs](#)

resources:

[View This Design's CSS](#)

[CSS Resources](#)

[FAQ](#)

[Submit a Design](#)

[Translations](#)





A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.

DOWNLOAD THE SAMPLE HTML FILE AND CSS FILE



#### THE ROAD TO ENLIGHTENMENT

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

The Beauty of Web Design



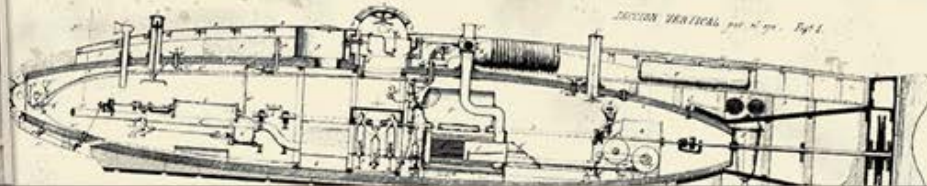
Fig. 11



#### SELECT A DESIGN

- Under the Sea!  
by Eric Stale
- Make 'em Proud  
by Michael McAgghon and  
Scotty Reijnsnyder
- Orchid Beauty  
by Kevin Addison
- Oceanscape  
by Justin Gray
- CSS Co., Ltd.

ICTINES-MONTURIO



SECTION TRANSVERSAL par le m. - Fig. 1.







*A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.*

### GET STARTED

Download the sample **html file** and **css file**

### OTHER SCOUTS

**Under the Sea!**  
by Eric Stoltz



### THE PATH TO ACHIEVEMENT

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

### SO WHAT IS THIS ABOUT?



# Css Zen Garden

the beauty of css design.

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.

[\[icon\] DOWNLOAD THE SAMPLE HTML FILE AND CSS FILE](#)

## The Road To Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

## So What is This About

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To

### SELECT A DESIGN:

[\[icon\] Under the Sea!](#)  
by Eric Stoltz

[\[icon\] Make 'em Proud](#)  
by Michael McAdon and Scotty Reiffnyder

[\[icon\] Orchid Beauty](#)  
by Kevin Addison

[\[icon\] Oceanscape](#)  
by Justin Gray

[\[icon\] CSS Co., Ltd.](#)  
by Benjamin Klemm

[\[icon\] Sakura](#)  
by Tatsuya Uchida

[\[icon\] Kyoto Forest](#)  
by John Polkowski

[\[icon\] A Walk in the Garden](#)  
by ...

# Statements

- A CSS style sheet is composed from a list of statements. A statement is either a **rule set** or an **at-rule**
- This example is a **rule set** delimited by the closing curly brace

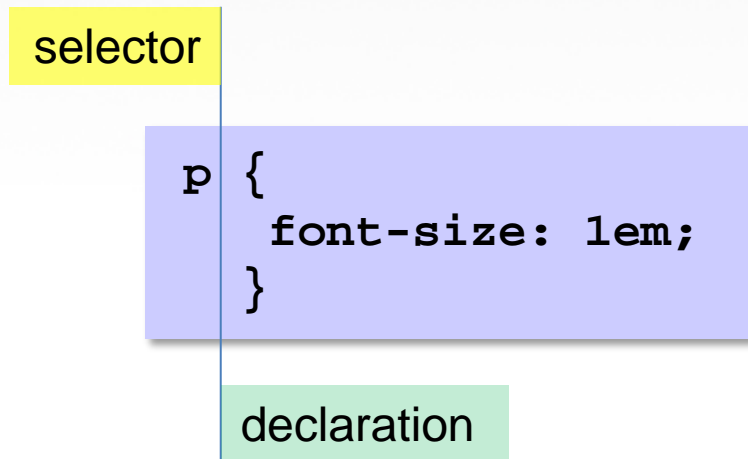
```
h2 {  
    color: #666;  
    font-size: 1.1em;  
}
```

- This example is an **at-rule** delimited by the semicolon at the end of the first line

```
@import url(basis.css);
```

# Rule sets

- A *rule set* (also called a rule) comprises a **selector** followed by a **declaration** block
- The rule set applies the declarations listed in the declaration block to all **elements** matched by the selector



# Selectors

- The **selector** is a **pattern**, and the **declarations** within the block that follows the selector are **applied** to all the **elements** that match this pattern
- Can contain one or more 'simple' selectors separated by combinators e.g. commas

multiple selectors

```
h1, h2 {  
  color: # 666;  
}
```

declaration

# Declaration blocks

- Declaration blocks begin with a left curly brace, {, and end with a right curly brace, }.
- They contain one or more **declarations** separated by **semicolons**:

```
h2 {  
    color: #666;  
    font-size: 1.1em;  
}
```



# Properties

- Within a declaration block there are property and value pairs



- There is a large collection of property names that can be used in styling, but they must use the correct syntax
- Each property has its own syntactic and semantic requirements and restrictions

# Property values

- Property values can be expressed as:
  - Keywords `bold; solid; auto;`
  - Strings `"Trebuchet MS", "Book Antiqua";`
  - Colours `#001c54; rgba(51,51,0,1);`
  - Numbers `1; 1.5;`
  - Lengths `20px; 15pt; 3in;`
  - Percentages `10%; 25%; 100%;`
  - URIs `url(images/bg.gif);`

# Shorthand notation

- Some properties can be combined into *shorthand notation* a notation to specify values for related properties in a single declaration, e.g.:

```
margin: 1em 2em 3em 4em;
```

is the shorthand for:

```
margin-top: 1em;
```

```
margin-right: 2em;
```

```
margin-bottom: 3em;
```

```
margin-left: 4em;
```

# Shorthand notation

- Some properties can be combined into *shorthand notation* a notation to specify values for related properties in a single declaration, e.g.:

```
margin: 1em 2em 3em 4em;
```

is the shorthand for:

```
margin-top: 1em;
```

```
margin-right: 2em;
```

```
margin-bottom: 3em;
```

```
margin-left: 4em;
```

Note the order in which these properties occur in the shorthand notation. They appear in a clockwise order, starting at the top: Top, Right, Bottom, Left. Remembering this order keeps you out of **TRouBLE**.

# Shorthand notation

- If only two or three values are specified, the “missing” side is assigned the same value as the one opposite it:

```
margin: 1em 2em 3em;
```

That's equivalent to these:

```
margin-top: 1em;
```

```
margin-right: 2em;
```

```
margin-bottom: 3em;
```

```
margin-left: 2em; /* same as margin-right */
```

# Shorthand notation

- If only two or three values are specified, the “missing” side is assigned the same value as the one opposite it:

```
margin: 1em 2em 3em;
```

That's equivalent to these:

```
margin-top: 1em;
```

```
margin-right: 2em;
```

```
margin-bottom: 3em;
```

```
margin-left: 2em; /* same as margin-right */
```

Note this is the syntax  
for a comment in CSS

# Shorthand notation

- In the same vein, consider this declaration:

```
margin: 1em 2em;
```

That's equivalent to these:

```
margin-top: 1em;
```

```
margin-right: 2em;
```

```
margin-bottom: 1em; /* same as margin-top */
```

```
margin-left: 2em; /* same as margin-right */
```

- Different forms of shorthand notation are available in CSS and we shall use them later in the course

# Shorthand notation

- Colour values also have shorthand notation:



`#FF0000` can be written as `#F00`;

also as `rgb(255, 0, 0);`

---

`rgba(255, 0, 0, 0.5);` is  50% transparent



# Shorthand notation

- Colour values also have shorthand notation:



`#FF0000` can be written as `#F00`;

also as `rgb(255, 0, 0);`

`rgba(255, 0, 0, 0.5);` is  50% transparent

The value of the Alpha  
(transparency) channel  
minimum 0, maximum 1

# At-rules

- At-rules are instructions or directives to the CSS parser
- It starts with an **at-keyword** followed by an **identifier** e.g.
  - `@charset`
  - `@import`
  - `@media`
- An at-rule can be a block delimited by curly braces, or text delimited by a semi-colon:

```
@media print {  
    body {  
        color: #000;  
    }  
}
```

```
@import url(basis.css);
```

# At-rules

At-Rules	Explanation
<b>@charset</b>	specifies the character set of an external style sheet
<b>@import</b>	imports an external style sheet
<b>@media</b>	specifies the media to which the style should be applied
<b>@page</b>	defines a page box
<b>@font-face</b>	describes the details of a font

[http://www.tagindex.net/css/basic/at\\_rules.html](http://www.tagindex.net/css/basic/at_rules.html)

# Cross browser testing

- Although your web pages should look very similar in different browsers you will want to check what the differences are

Cross Browser Testing is available at:

- **`http://browsershots.org`**  
has free and commercial services

# Homework

- Browse the Internet and take a note of one site (for each category) that is
  - really pleasing to the eye in its presentation
  - very well structured
  - very easy to understand and use
- Also take note of a site for each category that is the exact opposite, i.e. not at all easy to use
- Be ready to make the URLs available to the class