WEB PROGRAMMING SCV1223

Cascading Style Sheets (CSS)

Dr. Md Sah bin Hj Salam

En. Jumail bin Taliba

CC BY NC SA

Why CSS?

To separate presentation from content

Without CSS: content and presentation are combined

HTML

Content
+
Presentation

Example:

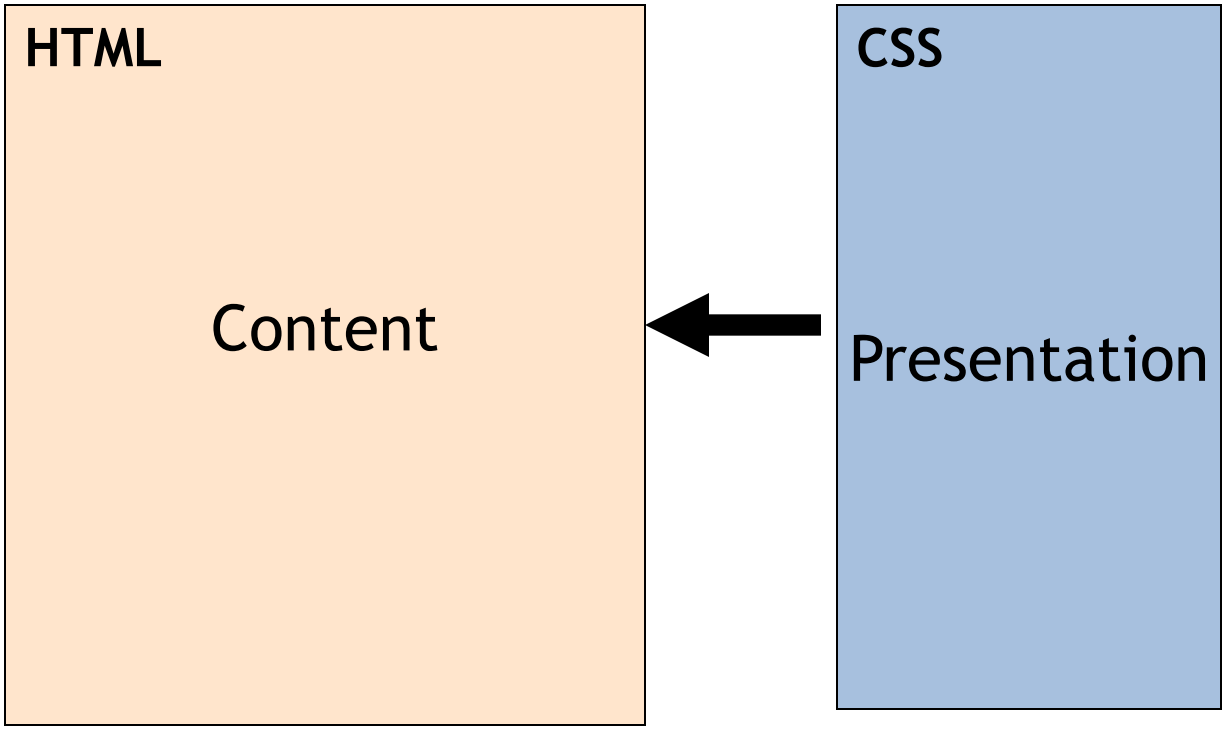
```
<font color="red">
    Welcome to the world
    of CSS
</font>
```

Code for Presentation

Content

Why CSS? *To separate presentation from content*

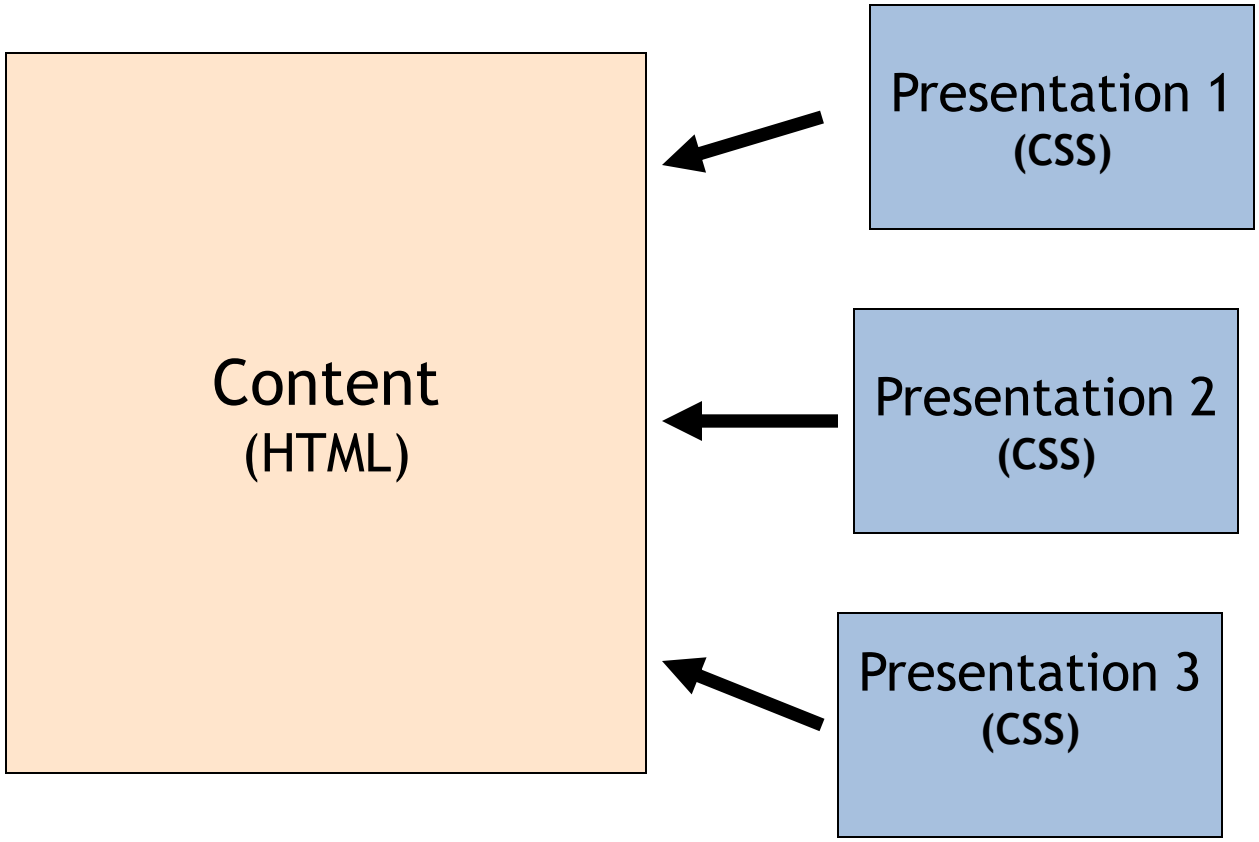
With CSS: content and presentation are separated



Why CSS?

To separate presentation from content

By using CSS, the same content can be presented in different appearances



File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://csszengarden.com/zengarden-sample.html

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.

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 majo 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-honor techniques in new and invigorating fashion. Become one with the web.

So What is This About?

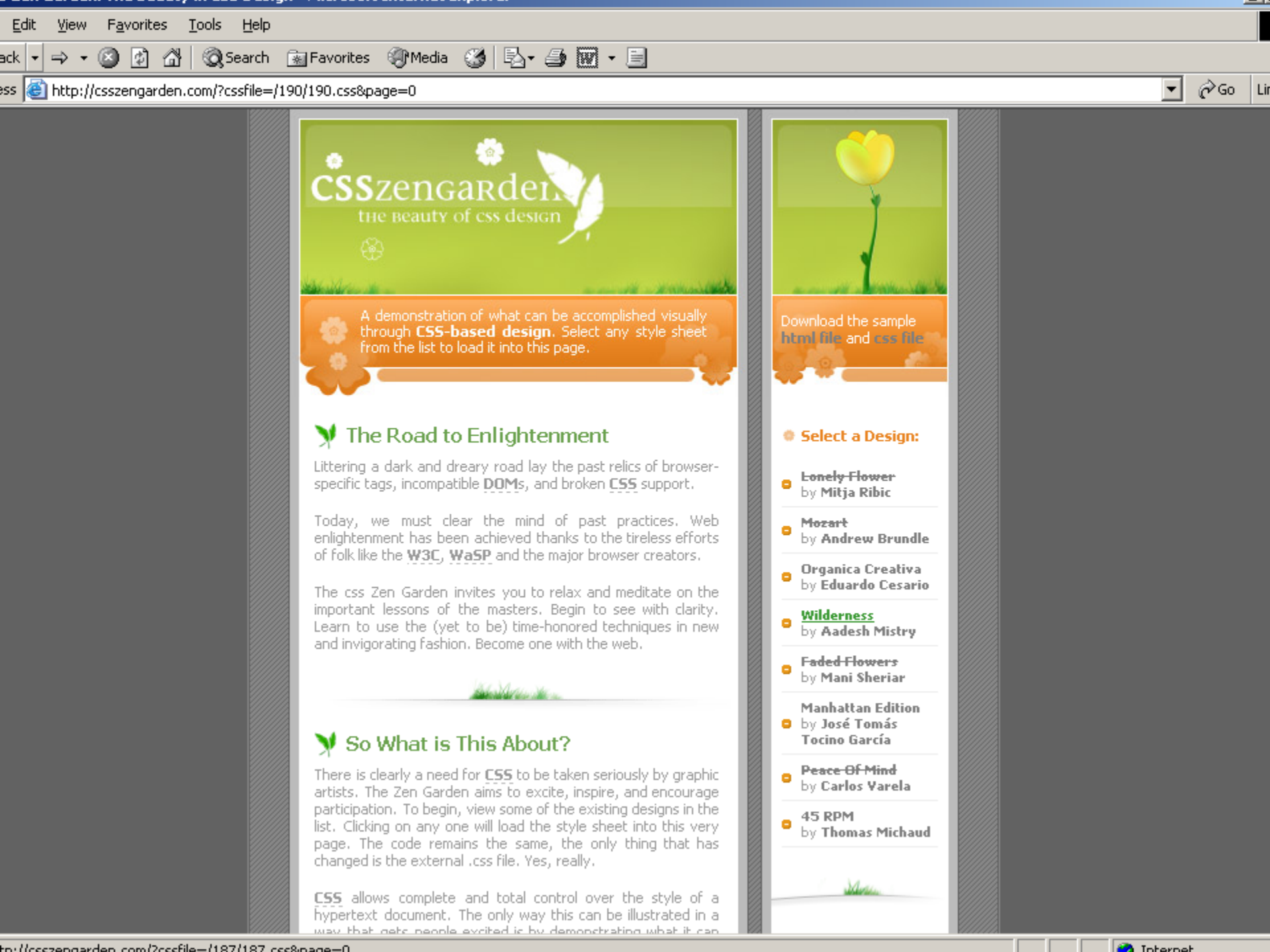
There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view so of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only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

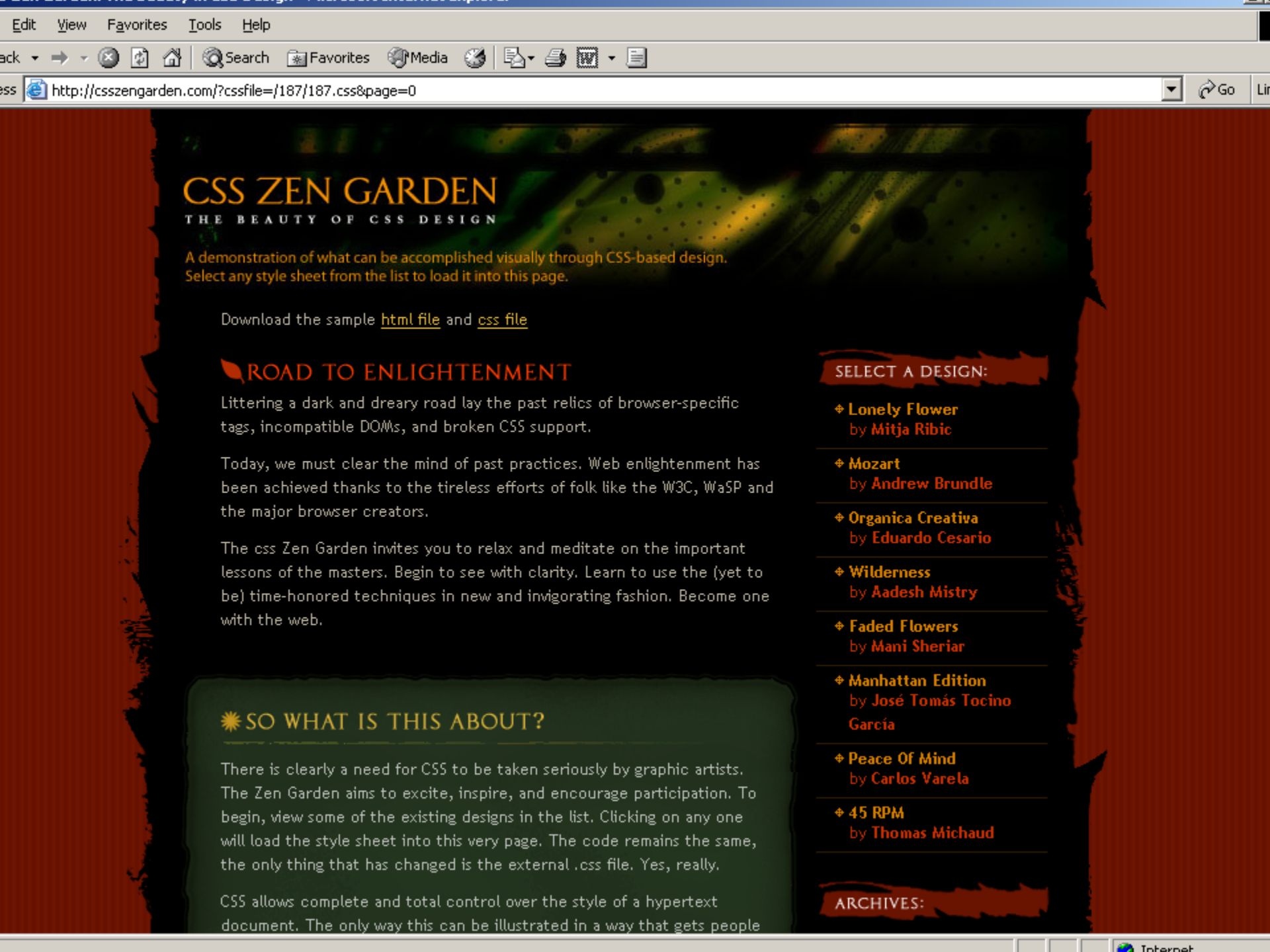
CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks a 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

css Zen Garden: The Beauty in CSS Design
Internet


Edit View Favorites Tools Help
Search Favorites Media
http://csszengarden.com/?cssfile=/190/190.css&page=0
Go
CSSzengarden
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
Select a Design:
Lonely Flower by Mitja Ribic
Mozart by Andrew Brundle
Organica Creativa by Eduardo Cesario
Wilderness by Aadesh Mistry
Faded Flowers by Mani Sheriar
Manhattan Edition by José Tomás Tocino García
Peace Of Mind by Carlos Varela
45 RPM by Thomas Michaud


Edit View Favorites Tools Help
Search Favorites Media
http://csszengarden.com/?cssfile=/187/187.css&page=0
Go
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.
Download the sample html file and css file
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
SELECT A DESIGN:
Lonely Flower by Mitja Ribic
Mozart by Andrew Brundle
Organica Creativa by Eduardo Cesario
Wilderness by Aadesh Mistry
Faded Flowers by Mani Sheriar
Manhattan Edition by José Tomás Tocino García
Peace Of Mind by Carlos Varela
45 RPM by Thomas Michaud
ARCHIVES:

Edit View Favorites Tools Help

Search Favorites Media

http://csszengarden.com/?cssfile=/182/182.css&page=0

Go

classic 45's 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!

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 *DOM*s, 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's 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

Current Hits

Lonely Flower by Mitja Ribic

Mozart by Andrew Brundle

Organica Creativa by Eduardo Cesario

Wilderness by Aadesh Mistry

Faded Flowers by Mani Sheriar

Manhattan Edition by José Tomás Tocino García

Peace Of Mind by Carlos Varela

45 RPM by Thomas Michaud

Golden Oldies

next designs »

View All Designs

Resources

Internet

Why CSS? To standardize the coding style

In HTML, different objects have different styles of coding or different property for the same presentation settings.

Example:

*Setting the **color** of table borders and text are different*

```
<table border="1" bordercolor="red">
……….
</table>
```

```
<font color="red">
 ……..
</font>
```

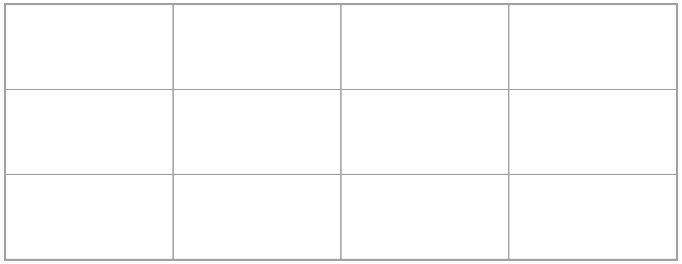
Why CSS?

To extend the attributes and features of HTML Tags

An HTML tag has limited attributes

Example:

What HTML CAN do



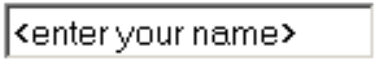
What HTML CAN'T do

Setting borders individually

Why CSS?

To extend the attributes and features of HTML Tags

What HTML CAN do



What HTML CAN'T do

Putting icon on a button

Setting colors, font, etc of form elements

Introduction

What are Cascading Style Sheets?

A set of formatting instructions

Most of the html elements have these attributes (specified for CSS):

- style
- id
- class

Using CSS involves two steps:

- Define styles
- Apply styles

Define Styles

Three ways to define styles:

- Inline
- Embedded
- Linked

Define Inline Styles CSS

Inline Styles

- Using the STYLE attribute
- Put directly into the start tags

Example

```
<p style="color:red">Red Text </p>
```

Define Embedded Styles CSS

Embedded Styles

- Style definitions are embedded in a document's head using the <STYLE> tag

Example

```
<head>
<style type="text/css">
  p {color:red;}
  em {font-size: 16pt; color: green;}
</style>
</head>

<body>
  <p> This is a red text </p>
</body>
```

Define Linked / External Styles CSS

Linked Styles

- Style definitions are put in an external file *(normally we use file extension .css for this file)*
- The external file is linked to the document using the <LINK> tag and put in head.

Example

mystyle.css

```
p {color:re
em {font-size: 16pt; color: green;}
```

document.html

```
<head>
d;} <link rel="stylesheet" href="mystyle.css">
</head>
```

Define Styles: Precedence

Styles Precedence:

- Inline styles have the highest precedence because they are the closest to the tag.

Example

```
<head>
<style type="text/css">
  p {color:red;}
</style>
</head>

<body>
  <p style="color:blue">The blue color will take
  precedence</p>
</body>
```

CSS Selector Format

Format of a CSS Style

```
    selector {property:value}
or
    selector {property_1:value_1;
              property_2:value_2;
              property_n:value_n;}
```

CSS Selectors:

- Tag/Element
- Identifaction (id)
- Class
- Tag-spesific
- Grouping
- Contextual
- Pseudo-class

CSS Selector Types

CSS Selectors:

- Tag/Element
- Identifaction (id)
- Class
- Tag-spesific
- Grouping
- Contextual
- Pseudo-class

CSS Selectors : Tag / Element

Tag/Element Selector

- Using name of element/tag
- To redefine the default value of tags

Example

Define

```
p {color:red;}
/*makes all p tags red*/
h1 {font-size: 20px;}
/*makes all text set by h1 tag to be 20 pixel in size*/
```

Apply

```
<h1> <p> This is a red text with 20 pixel in
  size</p></h1>
```

CSS Selectors : ID

Id Selector

- To define styles, the selector names use prefix # character
- To apply the styles, set the ID attribute with the name

Example

```
Define
#red {color:red;}

Apply
<b id="red"> This is a red bold text </b>
```

CSS Selectors : class

Class Selector

- To define styles, the selector names use prefix . (dot) character
- To apply the styles, set the CLASS attribute with the name

Example

```
Define
.green {color:#00FF00; font-weight:bold;}

Apply
<font class="green"> I like green color</font>
```

CSS Selectors : Tag-Specific

Tag-Specific Selector

- Style can only be applied to elements which defined it.
- Selector name:

```
tag_name.style_name or tag_name#style_name
```

Example

```
Define
font.redbold {color:red; font-weight:bold;}

Apply
<font class="redbold"> This is a red bold text</font>
<b class="redbold"> redbold style has no effect here</b>
```

CSS Selectors: Grouping

Grouping Selector

- Define the same styles to a group of tags.

Example

```
Define
h1,h2,h3 {background-color: orange;}
/*sets the background color of all h1,h2,and h3 elements to orange */
```

CSS Selectors : Contextual

Contextual Selector

- Applies to descendent tags

Example

```
Define
p b{color:red; text-decoration: underline;}

Apply
<b>CSS has no effect here</b>
<p>CSS has no effect here</p>
<p>
   <b>CSS has effect here. This is a red underlined text
   </b>
</p>
```

CSS Selectors: Pseudo-class

Pseudo-class Selector

- **a:link** specifies the unvisited links
- **a:hover** specifies the link as mouse cursor is hovering on it
- **a:active** specifies the link as it is being clicked
- **a:visited** specifies the link after being clicked

Example

Define

```
a:link {font-weight: bold;}
/* makes unvisited links bold */
a:hover {text-transform: uppercase;}
/* makes links uppercase as mouse cursor is hovering on*/
a:active {color: red;}
/* makes links red as they are clicked */
a:visited {text-decoration: underline;}
/* makes visited links underlined*/
```

CSS Measurements

Unit	Description	Example
%	Defines a measurement as a percentage relative to another value, typically an enclosing element.	`p {font-size: 14pt; line-height: 150%;}`
cm	Defines a measurement in centimeters.	`div {margin-bottom: 1cm;}`
em	A relative measurement for the height of a font in em spaces. Because an em unit is equivalent to the size of a given font, if you assign a font to 12pt, each "em" unit would be 12pt; thus, 2em would be 24pt.	`p {letter-spacing: 5em;}`
ex	This value defines a measurement relative to a font's x-height. The x-height is determined by the height of the font's lowercase letter x.	`p {font-size: 14pt; line-height: 2ex;}`

CSS Measurements

Unit	Description	Example
in	Defines a measurement in inches.	`p {word-spacing: .25in;}`
mm	Defines a measurement in millimeters.	`p {word-spacing: 12mm;}`
pc	Defines a measurement in picas. A pica is equivalent to 12 points; thus, there are 6 picas per inch.	`p {font-size: 10pc;}`
pt	Defines a measurement in points. A point is defined as 1/72nd of an inch.	`body {font-size: 14pt;}`
px	Defines a measurement in screen pixels.	`p {padding: 15px;}`

CSS Properties

Categories

- **Font and Type Settings**
- **Background**
- **Block**
- **Box**
- **Border**
- **List**
- **Positioning**

Font and Type Properties

```
font-family
font-size
font-size-adjust
font-size-stretch
font-style
font-variant
font-weight

text-align
text-decoration
text-indent
text-shadow
text-transform
```

Background Properties

```
background-attachment
background-color
background-image
background-position
background-repeat
```

Border Properties

```
border-bottom
border-bottom-color
border-bottom-style
border-bottom-width
border-collapse
border-color
border-left
border-left-color
border-left-style
border-left-width
border-right
border-right-color
border-right-style
border-right-width
border-spacing
border-style
border-top
border-top-color
border-top-style
border-top-width
border-width
```