





CSS
cascading style sheet

CSS, History

CSS를 배우기에 앞서... 탄생 비화를 통해 CSS의 발전 흐름을 알아봅니다.



History of CSS

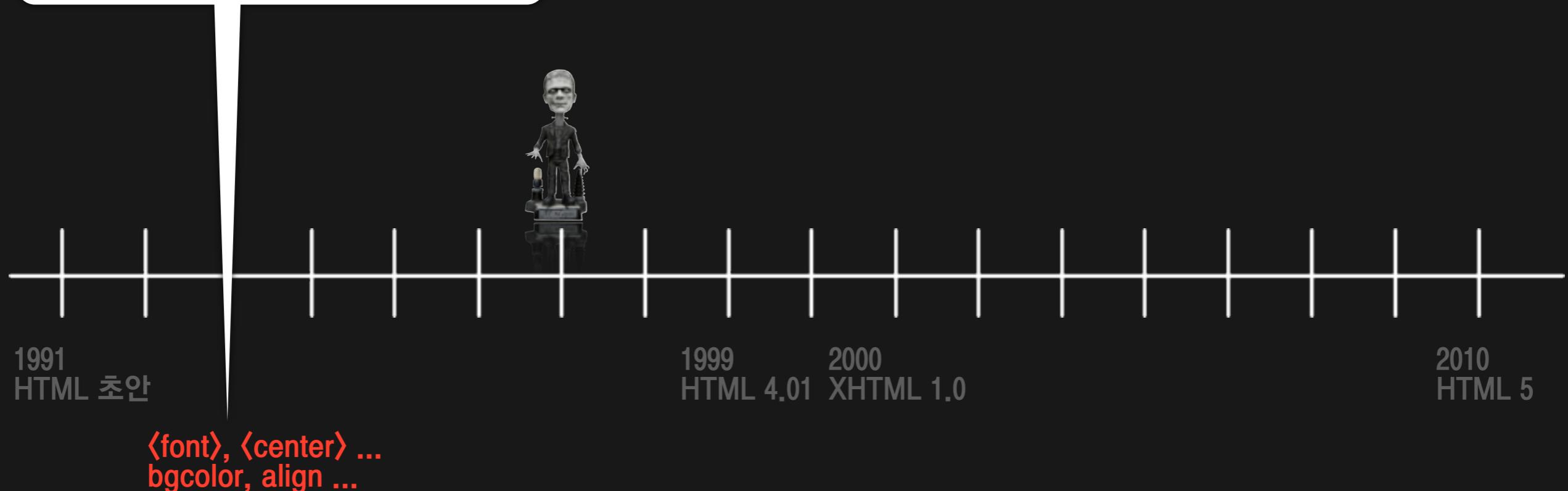
웹 스타일에 관한 짧은 역사, 1996 – 2011



History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011

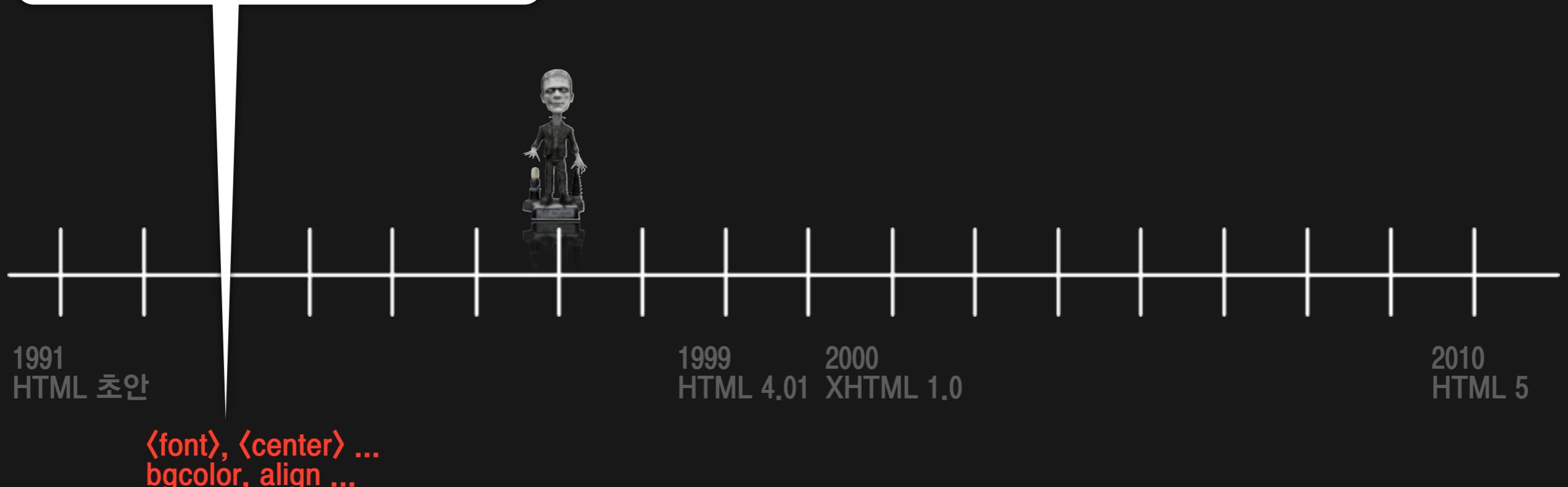
1990년대 중반 브라우저 제조사는
웹 디자인의 수요에 맞추어 개별적으로 HTML에
표현을 위한 태그 등을 추가



History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011

1990년대 중반 브라우저 제조사는
웹 디자인의 수요에 맞추어 개별적으로 HTML에
표현을 위한 태그 등을 추가



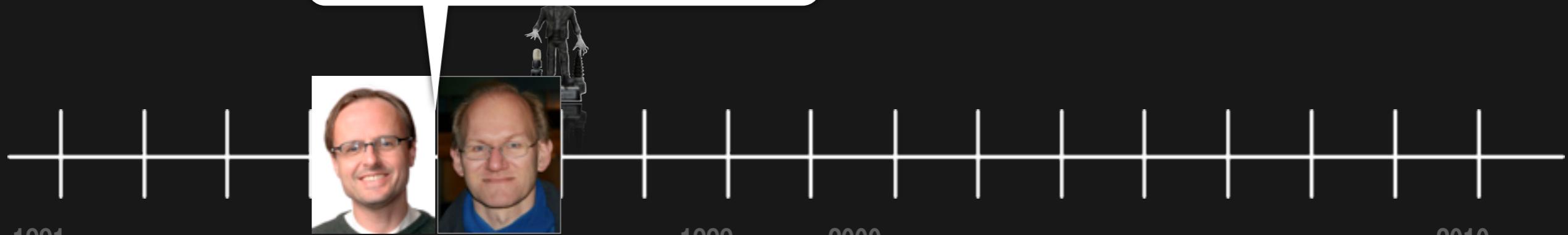
현대에 와서 심각한 문제 발생!!!

HTML문서에 디자인 속성이 추가되면서
검색엔진, 스크린리더기에서 내용 파악이
어려워짐. 유지/보수 또한 어려움.

History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011

1996년 W3C는 호콘 비움 리에와 베트 보스가
앞서 개발한 내용을 바탕으로 CSS 1 발표



`, <center> ...` 1996
`bgcolor, align ...` CSS 1.0

History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011



1991
HTML 초안

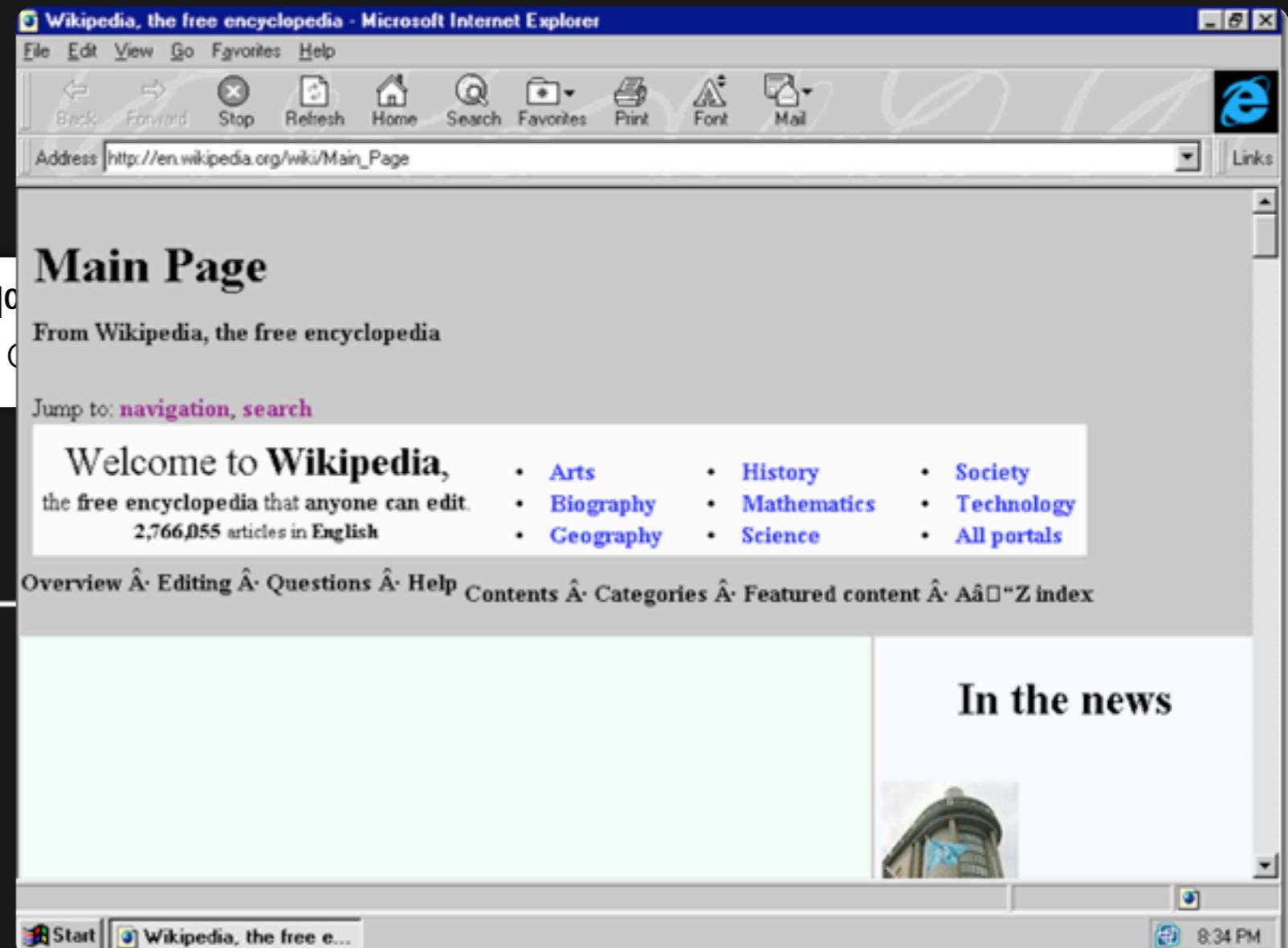
, <center> ...
bgcolor, align ...

1996
CSS 1.0



IE 3 CSS 1 시범적용

이후 IE 4, NN 4에서 좀 더 적극적 지원.
폰트, 컬러, 여백 등 조절할 수 있는
기본적 디자인 옵션 탑재.



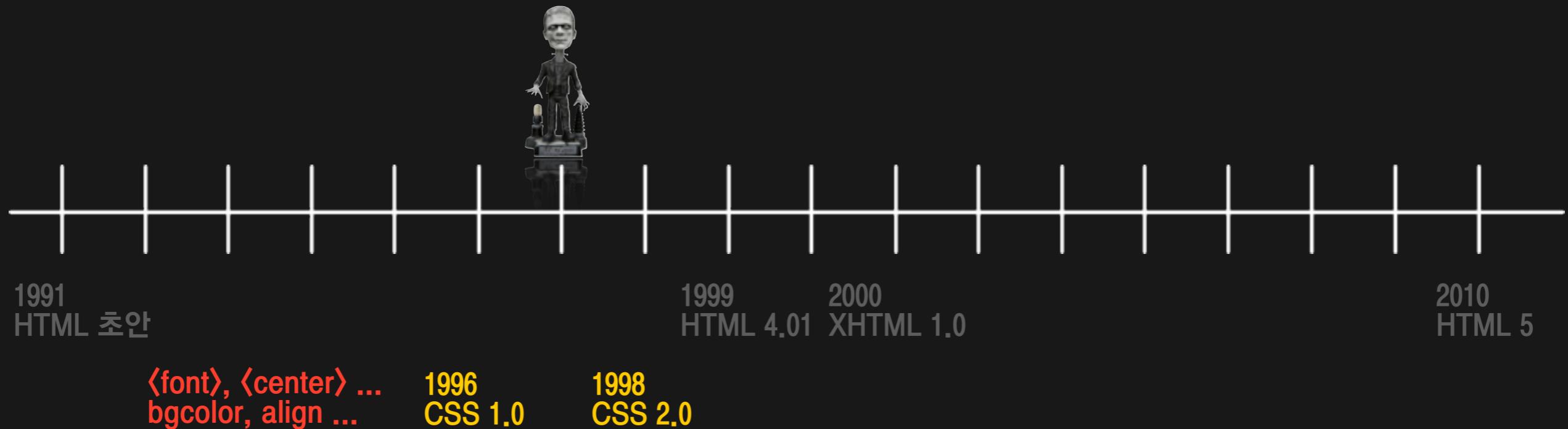
In the news



8:34 PM

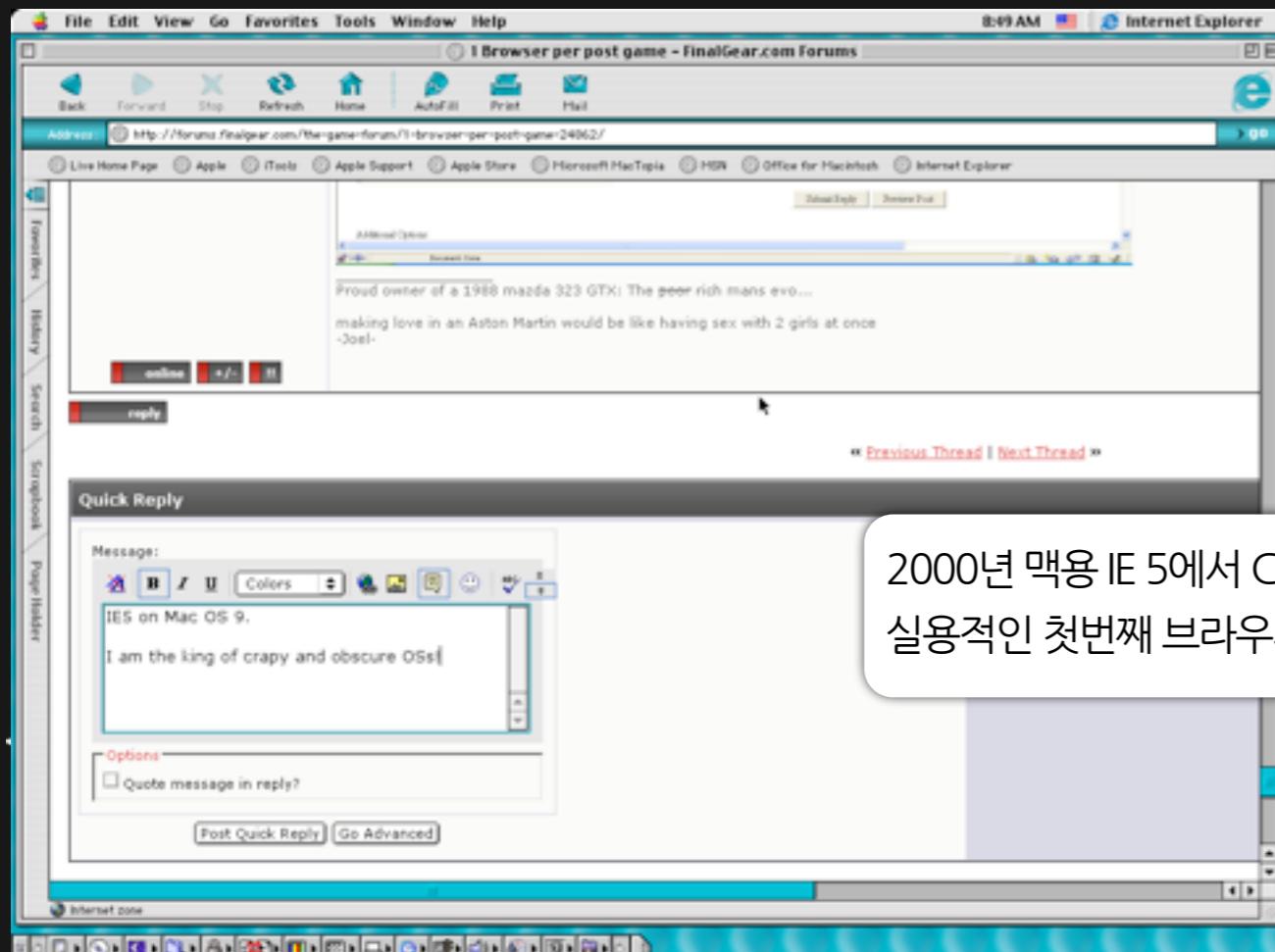
History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011



History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011



2000년 맥용 IE 5에서 CSS1-2를 지원하는
실용적인 첫번째 브라우저를 발표

2000
XHTML 1.0

, <center> ...
bgcolor, align ...

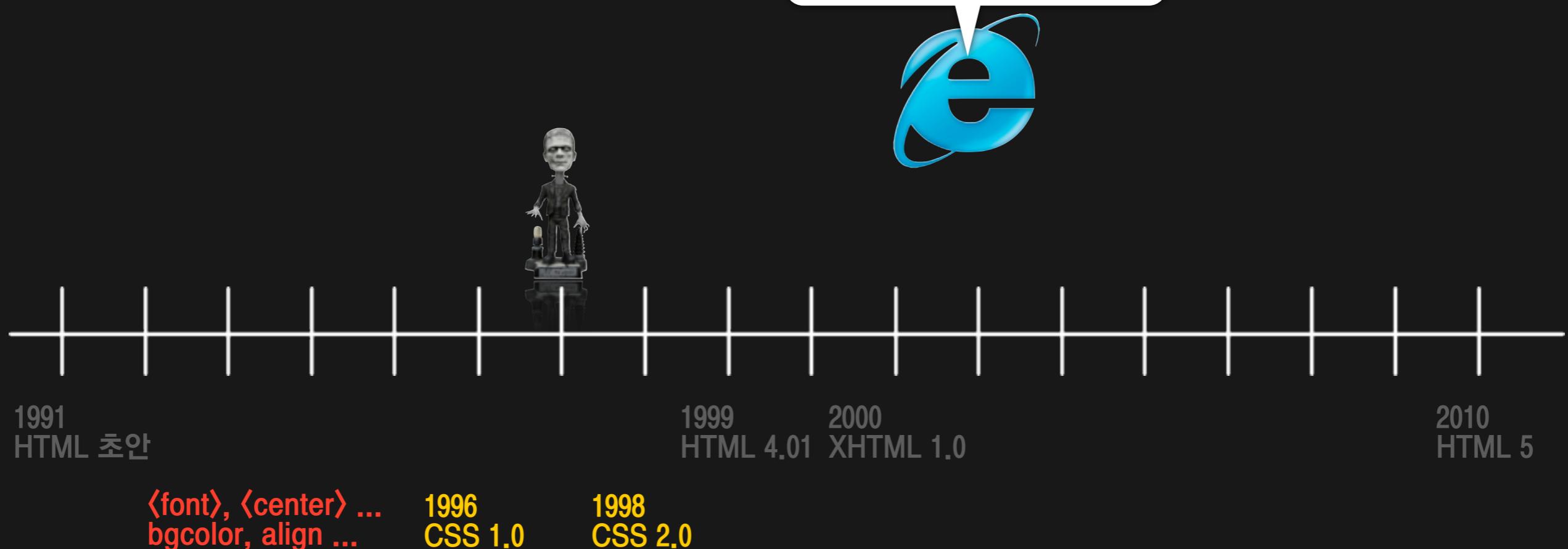
1996
CSS 1.0

1998
CSS 2.0

2010
HTML 5

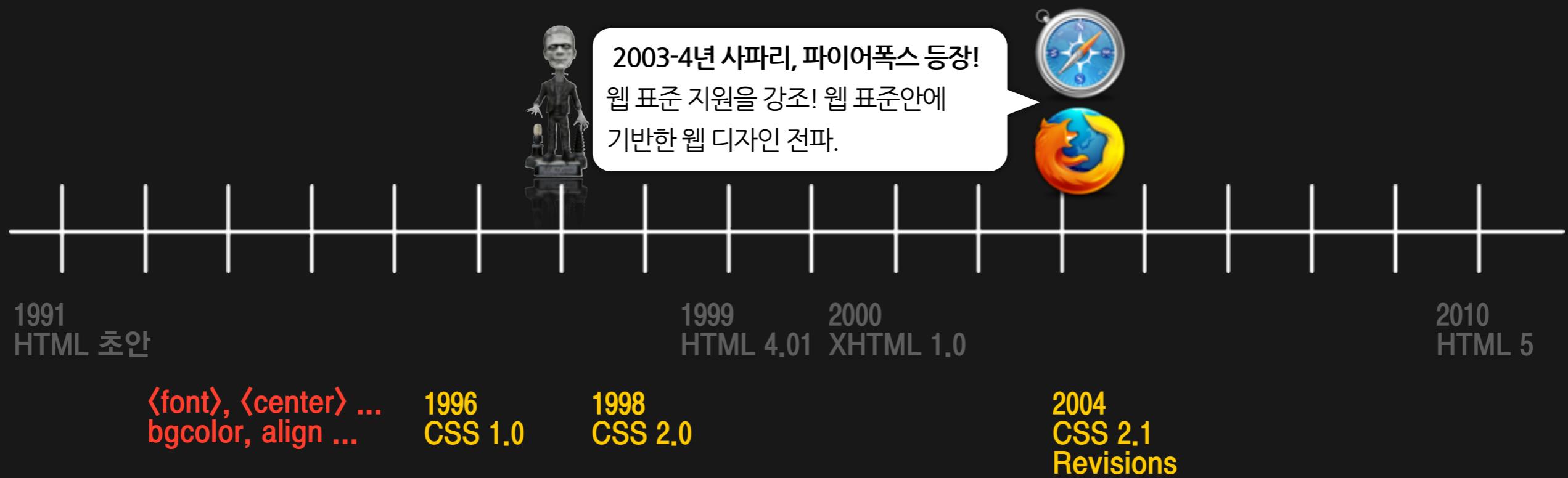
History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011



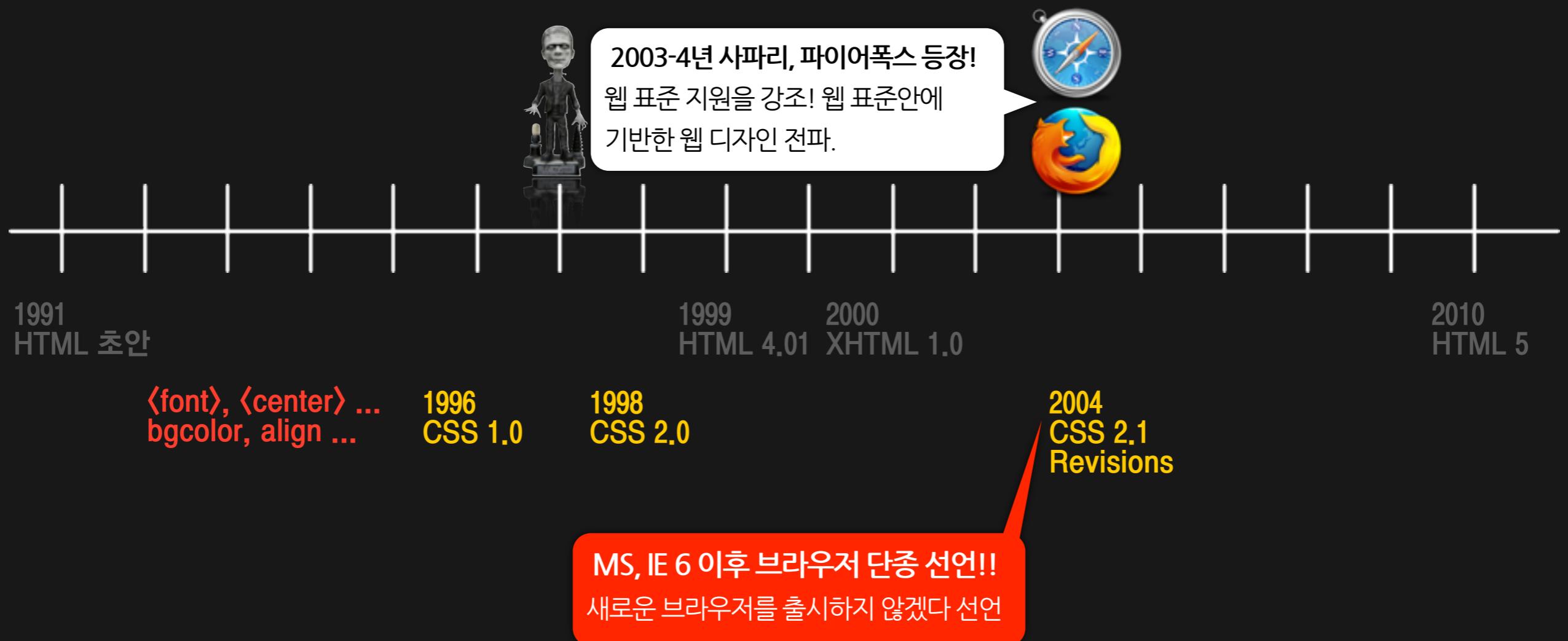
History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011



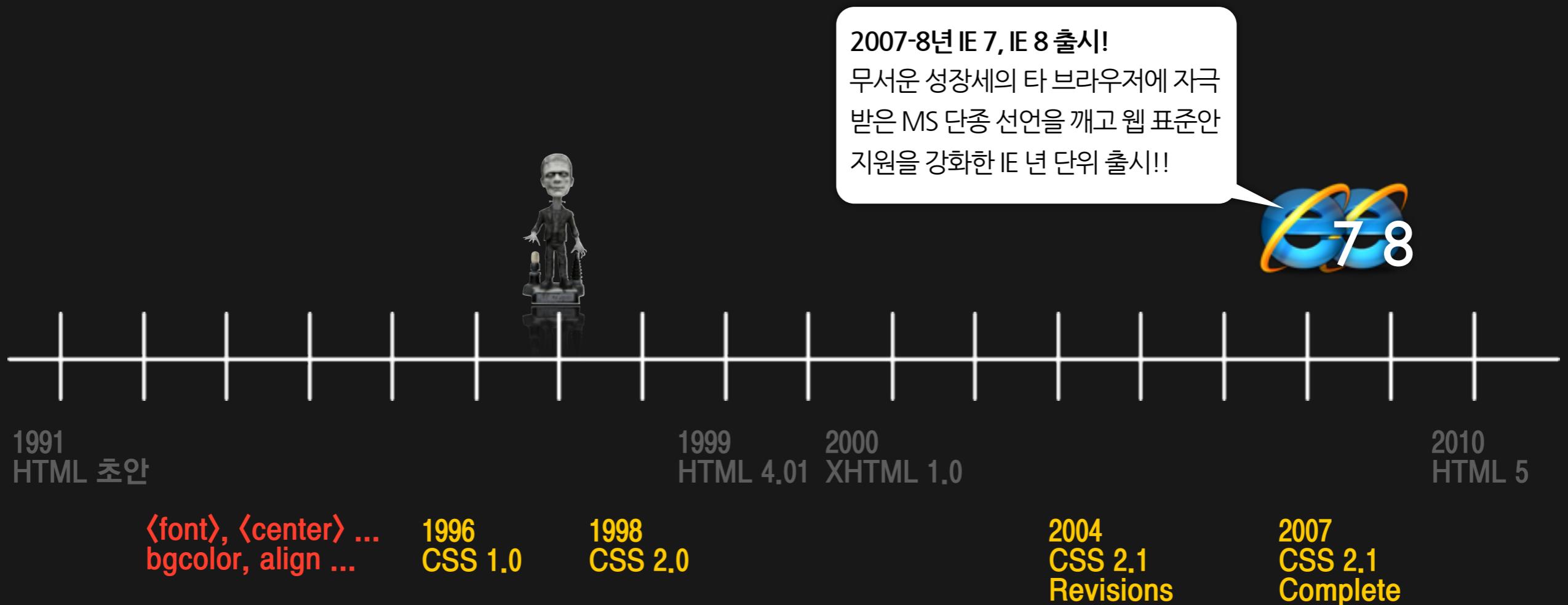
History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011



History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011



History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011

2007-11년 스마트 시대 개막!
아이폰 출시 이후 시장은 급변!!
웹 표준안 지원이 필수로 자리잡음.

2007 – 2011

iPhone &
Smart Device



1991
HTML 초안

, <center> ...
bgcolor, align ...

1996
CSS 1.0

1998
CSS 2.0

1999
HTML 4.01

2000
XHTML 1.0

2004
CSS 2.1
Revisions

2007
CSS 2.1
Complete

2010
HTML 5

History of CSS

웹 스타일에 관한 짧은 역사, 1996 – 2011



2007-11년 스마트 시대 개막!
아이폰 출시 이후 시장은 급변!!
웹 표준안 지원이 필수로 자리잡음.

2007 – 2011

iPhone &
Smart Device



2011년 IE 9 출시!
MS의 전통적인 관례를 깨고 웹 표준
지원을 강화한 IE 9 출시.

2010
HTML 5

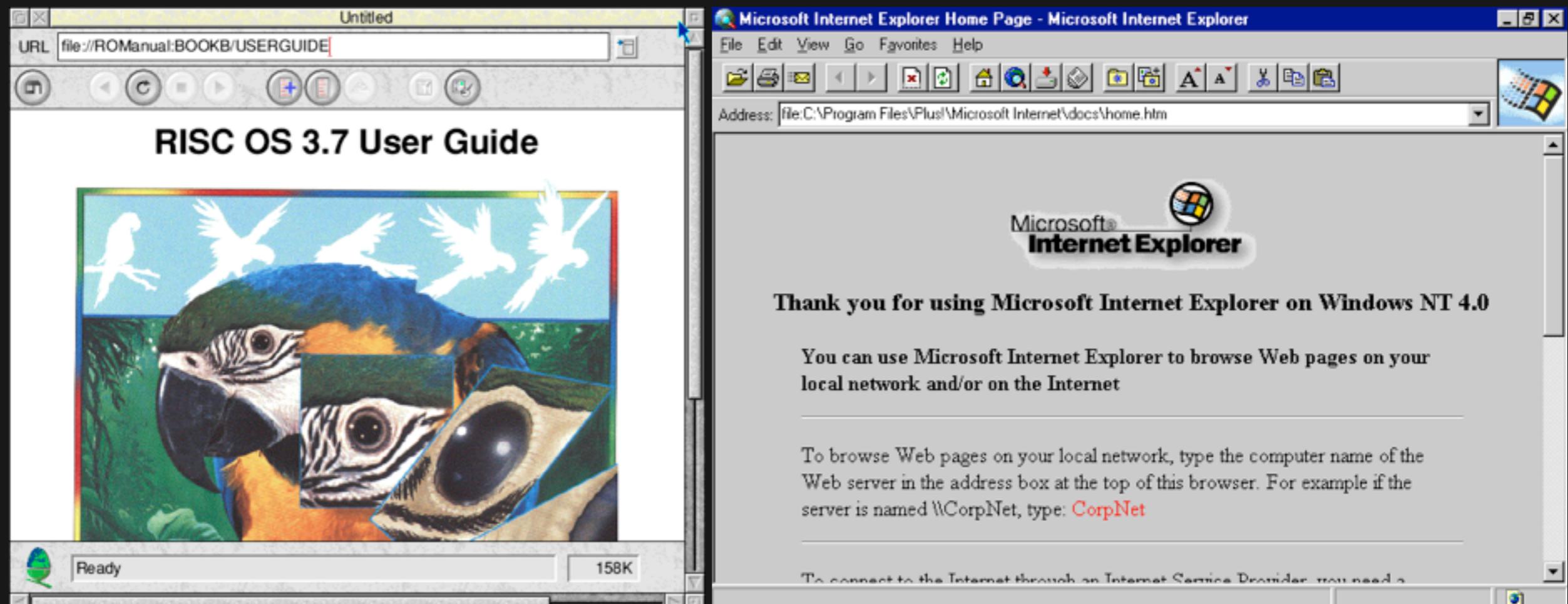
2011
CSS 3.0

2007
CSS 2.1
Complete

2004
CSS 2.1
Revisions

HTML Style Elements & Attributes for Designer

웹을 망가트린 암흑시대의 유물, 1994 –



HTML Style Elements & Attributes for Designer

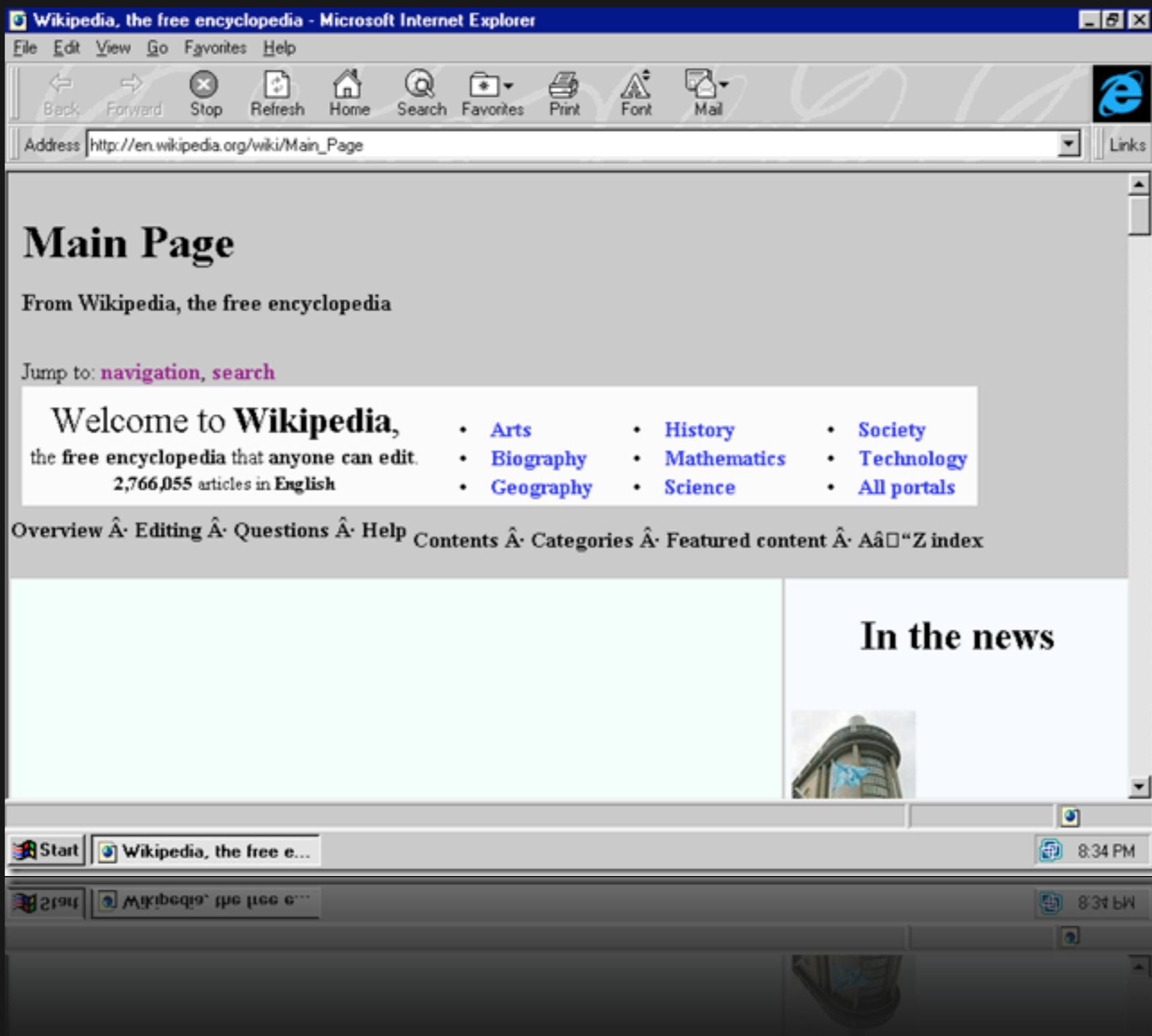
웹을 망가트린 암흑시대의 유물, 1994 –

The screenshot shows two windows side-by-side. The left window is a text editor with the title 'Untitled - Microsoft Internet Explorer' and the URL 'file:///ROManual:BOOKB/USERGUIDE'. It contains the source code for the 'RISC OS 3.7 User Guide'. The right window is Microsoft Internet Explorer 4.0 with the title 'Microsoft Internet Explorer Home Page - Microsoft Internet Explorer' and the address 'file:C:\Program Files\Plus\Microsoft Internet\docs\home.htm'. The page displays the same content as the source code, featuring a blue header with white text and several horizontal lines.

```
<body bgcolor="DeepSkyBlue" text="tan">
  <font size="2" face="Verdana" color="AliceBlue">
    웹을 망가트린 <b>암흑시대</b>의 <i>유물</i>
  </font>
  <hr size="1" width="100%" color="RoyalBlue">
  <br>
  <hr size="2" width="100%" color="MediumBlue">
  <br>
  <br>
  <br>
  <br>
  <hr align="left" width="30%" color="DarkSlateBlue">
  <br>
  <hr align="center" width="30%" color="MediumPurple">
  <br>
  <hr align="right" width="30%" color="Moccasin">
```

Internet Explorer 3, Windows

혁명의 서곡... CSS 1 시범 적용, 1996



Internet Explorer 4, Windows

혁명의 가능성 시사... CSS 1 확장 지원, 1997



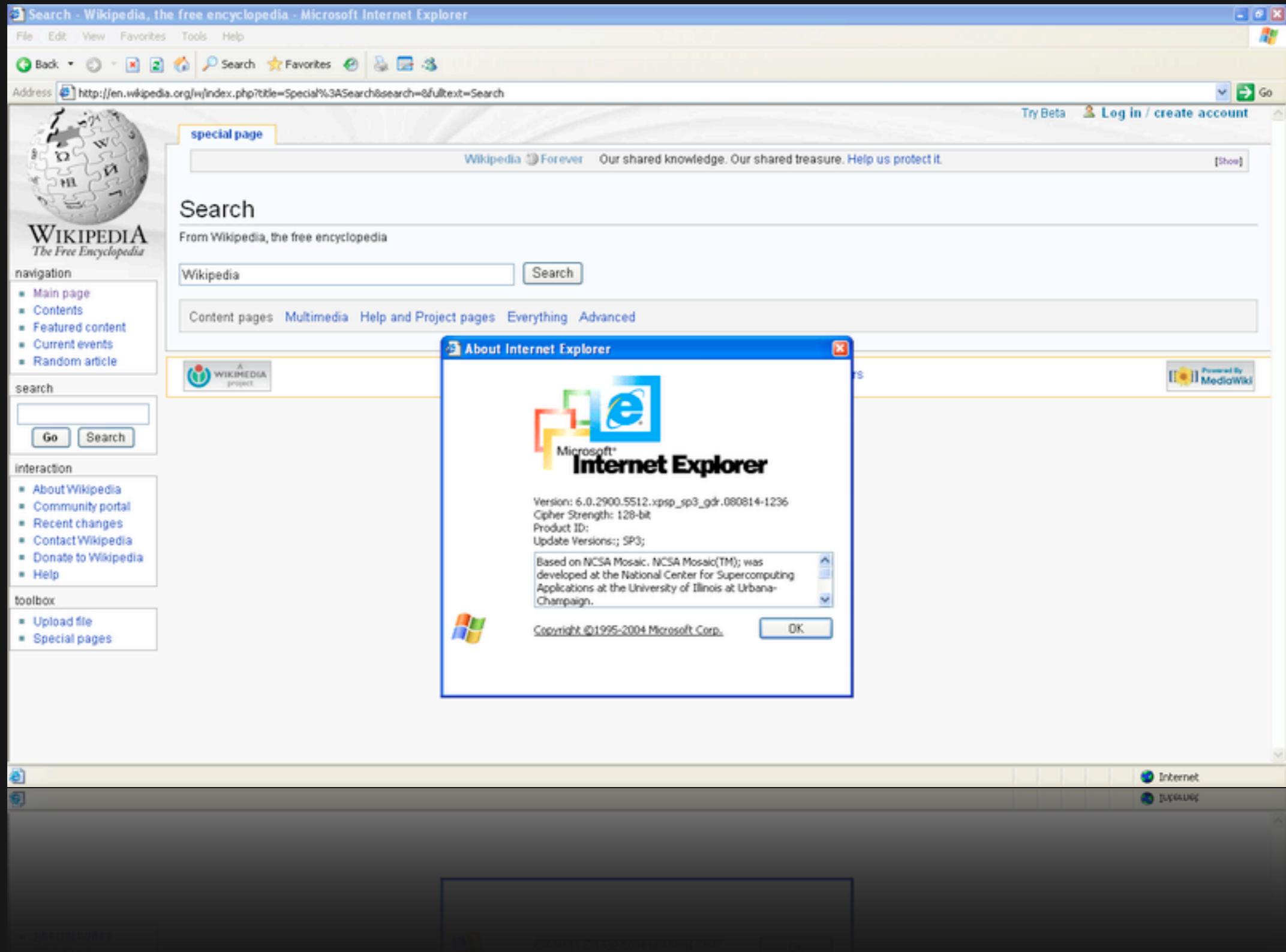
Internet Explorer 5, Mac

Tasman 렌더링 엔진의 등장... CSS 1, 2 지원, 2000



Internet Explorer 6, Windows

살아있는 전설... CSS 2의 상당 부분 지원, 2001



FireFox 1, Windows

비주류의 폭풍 역습... 웹 표준안 지원 강조, 2004

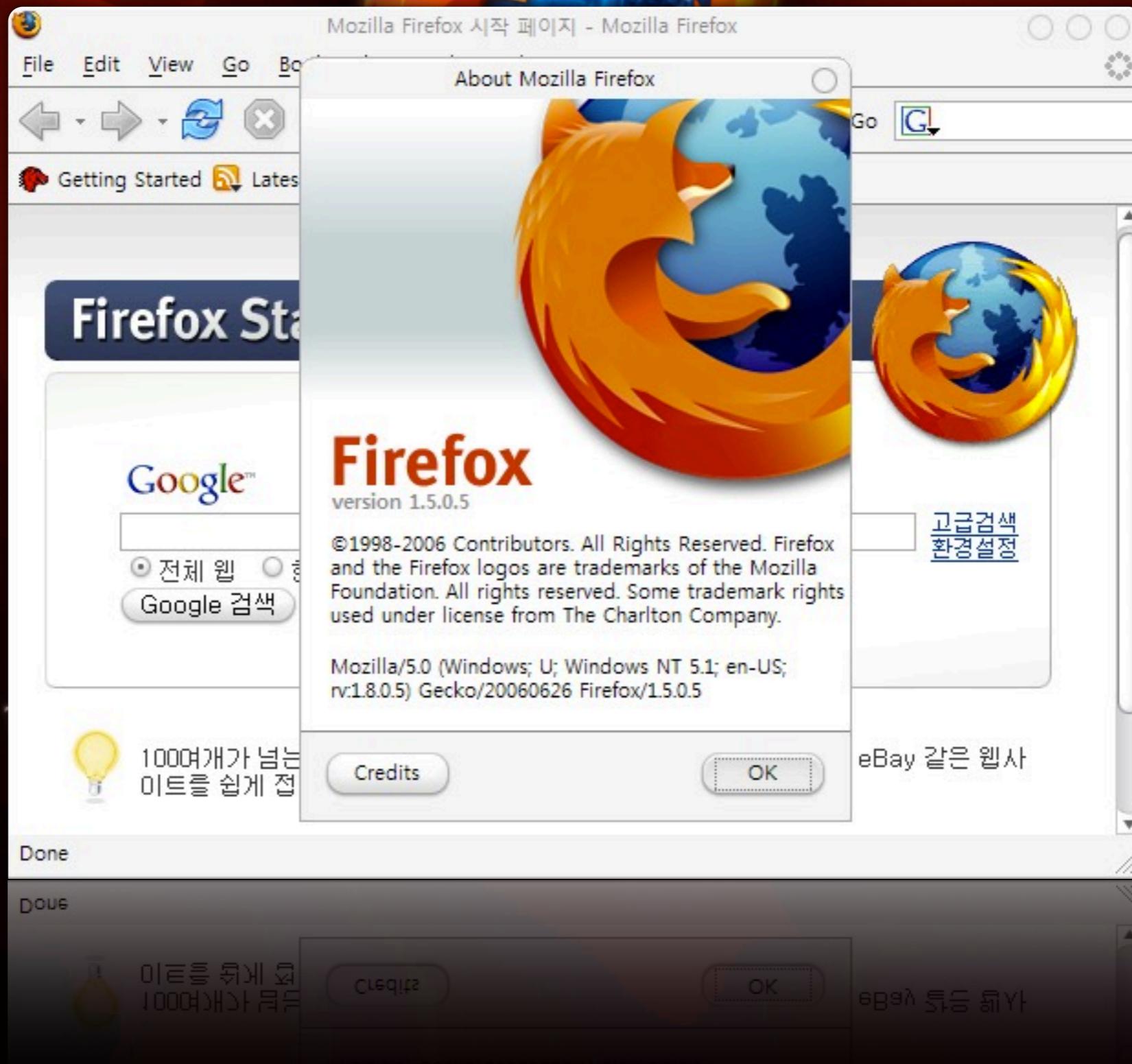


F I R E F O X

T A K E B A C K T H E W E B

FireFox 1, Windows

비주류의 폭풍 역습... 웹 표준안 지원 강조, 2004



Wrong Web Information

잘못된 정보가 판 치는 대한민국 웹, 2010

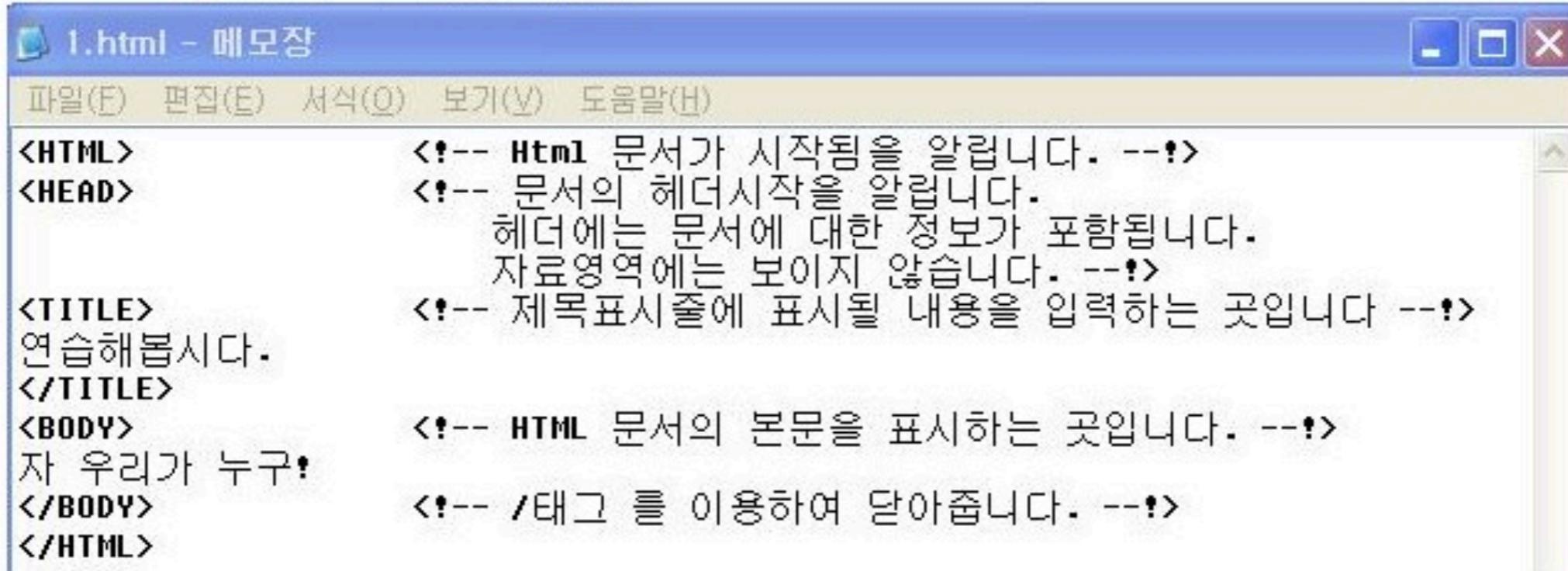
기본적인 HTML 태그를 연습하여봅시다.

조회 수 192 추천 수 0 | 2

[4] 스피지밥 <http://www.j...>

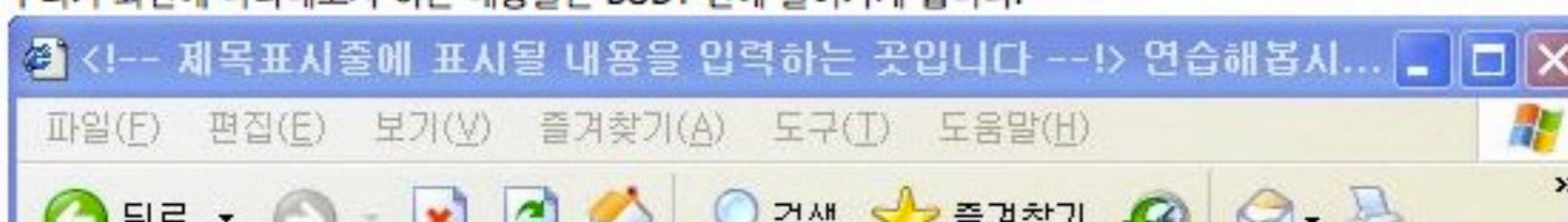
1. HTML 의 구조

일단 HTML 의 기본적인 구조부터 연습해보겠습니다.



```
<HTML>          <!-- Html 문서가 시작됨을 알립니다. --!>
<HEAD>          <!-- 문서의 헤더시작을 알립니다.
                  헤더에는 문서에 대한 정보가 포함됩니다.
                  자료영역에는 보이지 않습니다. --!>
<TITLE>          <!-- 제목표시줄에 표시될 내용을 입력하는 곳입니다 --!>
연습해봅시다.
</TITLE>
<BODY>          <!-- HTML 문서의 본문을 표시하는 곳입니다. --!>
자 우리가 누구!
</BODY>          <!-- /태그 를 이용하여 닫아줍니다. --!>
</HTML>
```

HTML 파일은 기본적으로 이렇게 구성됩니다.
HEAD 부분에는 TITLE 을 비롯한 각종 문서 정보가 포함되고
우리가 화면에 나타내고자 하는 내용들은 BODY 안에 들어가게 됩니다.



파일(E) 편집(E) 보기(V) 즐겨찾기(A) 도구(I) 도움말(H)

뒤로 뒤로 이동 뒤로 이동 검색 즐겨찾기

Wikipedia IE History

인터넷 익스플로러의 역사, 위키[피디아]

Internet Explorer

From Wikipedia, the free encyclopedia

Windows Internet Explorer (formerly **Microsoft Internet Explorer**, commonly abbreviated **IE** or **MSIE**) is a series of graphical web browsers developed by Microsoft and included as part of the Microsoft Windows line of operating systems starting in 1995. It was first released as part of the add-on package **Plus!** for Windows 95 that year. Later versions were available as free downloads, or in service packs, and included in the **OEM** service releases of Windows 95 and later versions of Windows.

Internet Explorer has been the most widely used web browser since 1999, attaining a peak of about 95% usage share during 2002 and 2003 with Internet Explorer 5 and Internet Explorer 6.^[citation needed] Since its peak of popularity its usage share has been declining in the face of renewed competition from other web browsers, and was 43.55% as of February 2011. Microsoft spent over US\$100 million per year on Internet Explorer in the late 1990s,^[1] with over 1000 people working on it by 1999.^[2]

Since its first release, Microsoft has added features and technologies such as basic table display (in version 1.5); XMLHttpRequest (in version 5), which aids creation of dynamic web pages; and Internationalized Domain Names (in version 7), which allow Web sites to have native-language addresses with non-Latin characters. The browser has also received scrutiny throughout its development for use of third-party technology (such as the source code of Spyglass Mosaic, used without royalty in early versions) and security and privacy vulnerabilities, and both the United States and the European Union have alleged that integration of Internet Explorer with Windows has been to the detriment of other browsers.

The latest stable release is Internet Explorer 9, which is available as a free update for Windows 7, Windows Vista SP2, Windows Server 2008 and Windows Server 2008 R2. Internet Explorer was to be omitted from Windows 7 and Windows Server



Internet Explorer 9 in Windows 7

| Details | |
|--|--|
| Type | Web browser |
| Included with | Windows 95 OSR 1 and later Windows NT 4 and later |
| Related components | |
| Windows Filtering Platform Windows Update WinSock Versions: 1 · 2 · 3 · 4 · 5 · 6 · 7 · 8 · 9 · 10 | |

Cascading Style Sheet History

CSS의 역사, CSS Neuse

CSS Neuse

CSS Information

The History of CSS

CSS Browser support

CSS3: What's the hype?

Limitations of CSS

The History of CSS

Cascading Style Sheets, or CSS, are a way to add formatting to HTML which has extremely limited means of presentation. For example, you can only format a paragraph with either horizontal or vertical alignments in plain HTML. The addition of style sheets allow for a wide range of looks which add to the appeal of web documents. There are many options available in CSS from font choices to alignment to indentation. One of the **major appeals of CSS** is the ability to provide a central place to control what your pages look like. Unlike HTML, where you have to change the settings on each page manually, with CSS you can change all your web pages from a central file stored on the server. This ability and design options make CSS the best choice for web design.

The first release of CSS was in 1996. This release called CSS1 was a product of the World Wide Web Consortium or W3C. The W3C is the source for all web design schema and the standards which are used across the globe. Since the original CSS was published there have been two other schema updates provided by the Consortium. CSS2 was released in 1998 and followed by CSS3 in 1999. The schema, or plans, for CSS are divided into modules and the current version has over 40 of these units. Since the first release in 1996 CSS has become a standard for web design.

Style sheets have become increasingly complicated since their first release. Each release added more capabilities to the system and broadened their appeal. Earlier versions provided the means to change fonts, set alignment, background colors and other simple formating. As the sheets system has developed you can now control almost all aspects of presentation within a web page. Many tags in HTML were deprecated and are no longer used. They were moved over to CSS to provide a central place for control of design and presentation.

The path for CSS was not always easy. In the beginning most browsers did not support the standard or only supported certain parts. Even now all browsers do not support style sheets the same way and this can cause bugs when designing web pages. Many programmers have developed work-arounds and hacks to provide for different browsers. Once browsers started accepting style sheets and presenting them correctly there were other barriers to implementation. Many designers and programmers were reluctant to use the formatting abilities. The slow progress made designers hesitant to use them since it was hard to know how, or if, they would be displayed. However, they are now considered the way to go with web design.



CSS, Syntax

CSS는 HTML 문서의 요소(대상)를 찾아 꾸밀 수 있는 스타일 문서로
HTML과는 다른 문법(문장 작성법)을 사용합니다.

Syntax of CSS

스타일 문서 기본 작성법

```
h1 {  
    color: deepSkyBlue;  
    font-size: 2em;  
}
```

Syntax of CSS

스타일 문서 기본 작성법

대상 선택자

```
h1 {  
    color: deepSkyBlue;  
    font-size: 2em;  
}
```

Syntax of CSS

스타일 문서 기본 작성법

```
대상 선택자 → h1 {  
    color: deepSkyBlue;  
    font-size: 2em;  
}
```

선언구간 시작

선언구간 종료

Syntax of CSS

스타일 문서 기본 작성법

```
대상 선택자 → h1 { 속성
                    선연구간 시작
                    선연구간 종료
                    color: deepSkyBlue;
                    font-size: 2em;
                }
```

Syntax of CSS

스타일 문서 기본 작성법



Syntax of CSS

스타일 문서 기본 작성법



Syntax of CSS

스타일 문서 기본 작성법



Embed CSS

HTML문서 head 내부에서 CSS 스타일링

```
<html>
  <head>
    <style type="text/css">
      h1 {
        color: deepSkyBlue;
        font-size: 2em;
      }
    </style>
  </head>
  <body>
    <h1>Embed Style Sheet</h1>
  </body>
</html>
```

Embed CSS

HTML문서 head 내부에서 CSS 스타일링

```
<html>          head 내부에서만 사용 가능
  <head>
    <style type="text/css">
      h1 {
        color: deepSkyBlue;
        font-size: 2em;
      }
    </style>
  </head>
  <body>
    <h1>Embed Style Sheet</h1>
  </body>
</html>
```

Embed CSS

HTML문서 head 내부에서 CSS 스타일링

```
<html>          head 내부에서만 사용 가능  
  <head>          MIME type 지정  
    <style type="text/css">  
      h1 {  
        color: deepSkyBlue;  
        font-size: 2em;  
      }  
    </style>  
  </head>  
  <body>  
    <h1>Embed Style Sheet</h1>  
  </body>  
</html>
```

inline CSS

요소에 직접 CSS 스타일링

```
<h1 style="color: deepSkyBlue; font-size:2em;">  
    Embed Style Sheet  
</h1>
```

inline CSS

요소에 직접 CSS 스타일링

요소 내부에 style 속성을 이용하여 CSS 코드 작성

```
<h1 style="color: deepSkyBlue; font-size:2em;">  
    Embed Style Sheet  
</h1>
```

External CSS

HTML문서 외부에서 CSS 스타일링

```
<html>
  <head>
    <style type="text/css">
      @import "css/style.css";
    </style>
  </head>
  <body>
    <h1>Embed Style Sheet</h1>
  </body>
</html>
```

External CSS

HTML문서 외부에서 CSS 스타일링

```
<html>
  <head>
    <style type="text/css">
      @import "css/style.css";
    </style>
  </head>
  <body>
    <h1>Embed Style Sheet</h1>
  </body>
</html>
```

@import 문을 이용하여 외부의 CSS 문서 호출

External CSS

HTML문서 외부에서 CSS 스타일링

```
<html>
  <head>
    <link
      href="css/style.css"
      type="text/css"
      rel="stylesheet"
    />
  </head>
  <body>
    <h1>Embed Style Sheet</h1>
  </body>
</html>
```

External CSS

HTML문서 외부에서 CSS 스타일링

```
<html>
  <head>
    <link
      href="css/style.css"
      type="text/css"
      rel="stylesheet"
    />
  </head>
  <body>
    <h1>Embed Style Sheet</h1>
  </body>
</html>
```

link 요소를 이용하여 외부의 CSS 문서 호출



CSS, Efficient

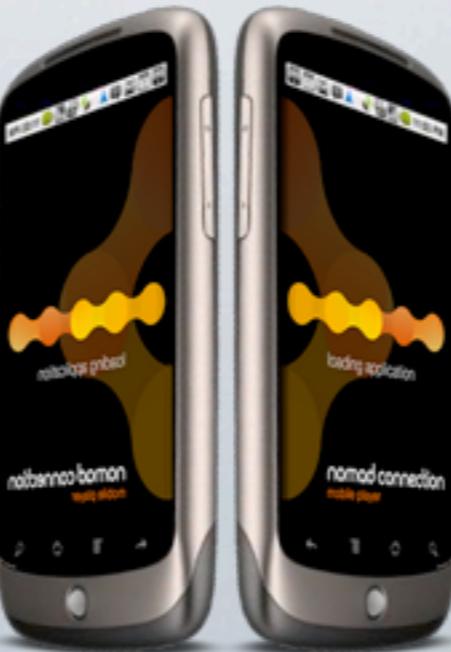
효과적인 CSS 작성법

수년에 걸친 노력으로 웹은 모바일 폰, 게임 장치, 텔레비전 등 어디서나 접속해도 볼 수 있게 되었습니다. 그리고 미래에는 더욱 더 많은 기기에서 웹은 활용될 수 있을 것입니다.





Windows Phone 7
LeSScro series
WVGA

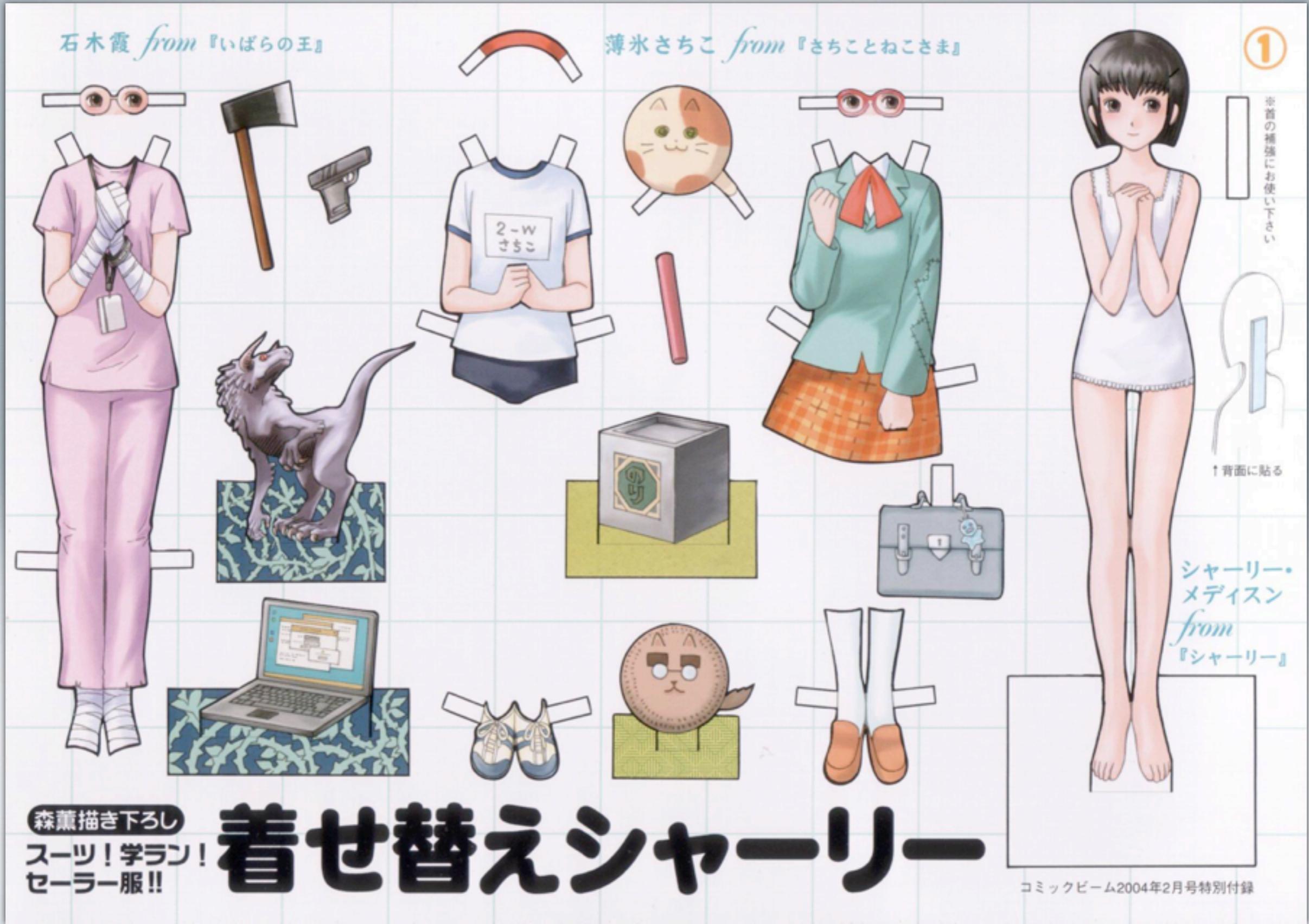


웹은 매우 빠르게 변하는 매체입니다,
오늘날까지 웹의 역사는 불과 20여년 남짓이지만 하루가 멀다하고 급변해왔습니다.
**웹 디자인을 하는 사람이라면 자신이 만든 디자인을 다양한 환경에서
접할 수 있다는 점을 고려하여 디자인하여야 합니다.**



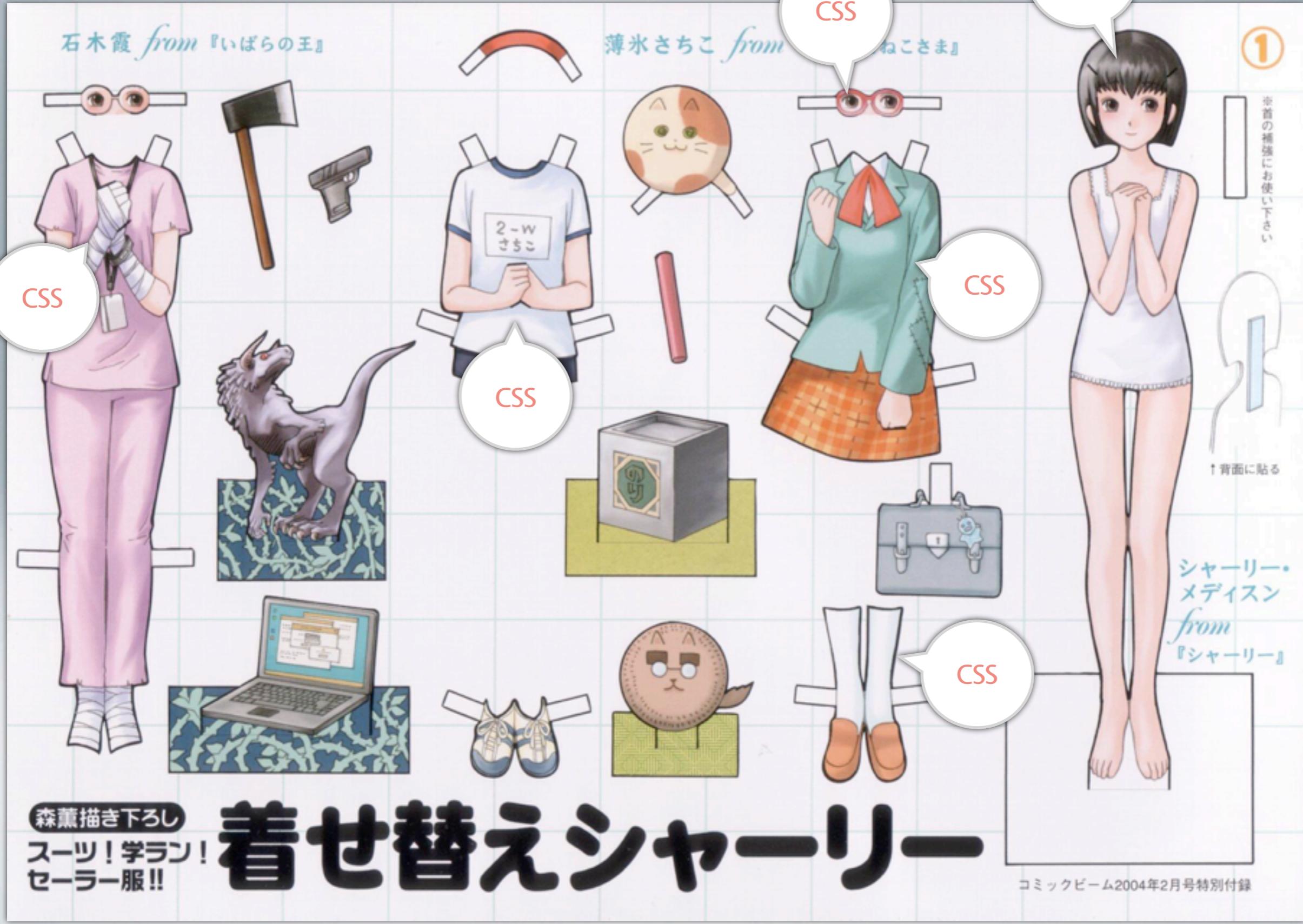
Efficient CSS is Style Division

효율적인 웹 디자인을 위한 CSS 활용법은 분리!



Efficient CSS is Style Division

효율적인 웹 디자인을 위한 CSS 활용법은 분리!



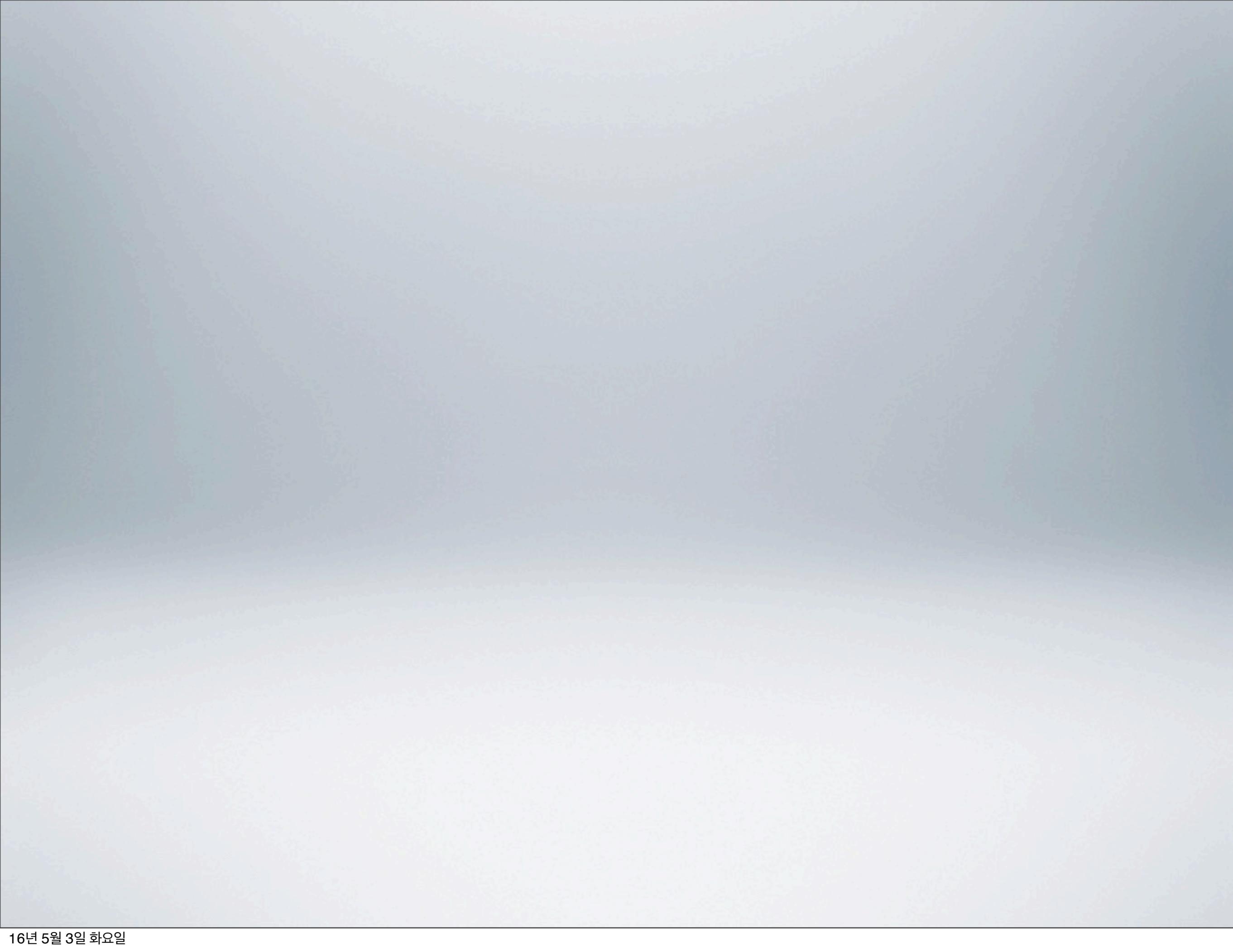
Efficient CSS is Style Division

효율적인 웹 디자인을 위한 CSS 활용법은 분리 !

Efficient CSS is Style Division

효율적인 웹 디자인을 위한 CSS 활용법은 분리 !

HTML과 CSS 분리가 필요한 이유입니다.



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 in bags. Incompatible DOCTYPEs, and broken CSS rules. Today, we must clear the mind of past practices. The Zen Garden invites you to relax. Important lessons of the masters. Begin to use the (yet to be) time-honored techniques. Learn to use the (yet to be) time-honored techniques. Become one with the web.

So What's This About?

There is clearly a need for CSS to be taken seriously. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the entries. Clicking on any one will load the style sheet. The code remains the same, the result is changed in the external css file. This makes CSS above complete and total control over the document. The only way this can be illustrated is by demonstrating what it means to place in the hands of those who know. To date, most examples of neat CSS demonstrated by structures and coders, does their mark. This needs to change.

Participation

Graphic artists only please. You are modifying styles are necessary, but commented well enough that even CSS novices can use them as starting points. Please see the CSS Resource Guide for advanced working with CSS.

You may modify the style sheet in any way. This may mean changing at first if you've never worked with CSS before, but follow the listed links to see the sample files as a guide.

Download the sample html file and css file. Once you have completed your masterpiece (and please, don't submit half-finished work), upload it to your website under your control. Send us a link to it, we will spider the site and the submission will be placed on our servers.

Benefits

Why participate? For recognition, inspiration, and to help people to whom making the case for CSS-based design is newly needed, even today. More and more major sites are taking the leap, but not enough have. One day this gains criticality; that day is not today.

Requirements

We would like to see as much CSS as possible to widely-supported elements only. About functional, practical CSS and not the latest bleeding edge tricks viewable by 2% of the browser market. The only real requirement we have is that your CSS validates. Unfortunately, designing this way highly frustrates different browser implementations of CSS. Unfortunately, designing this way highlights implementation differences between different browsers. View the Resources page for more information on some of the files available. Full browser compatibility is desired, and we do not accept pipe dreams, and we do not expect you to perfect code across every platform, the few can. If your design doesn't work in Mozilla (use by over 90% of the population), accept it.

We ask that you submit original artwork. Please keep copyrighted material to a minimum. Fairly acceptable, outright pornography will not be accepted.

This is a learning exercise as well as a wide full copyright on your graphics (with limited exception guidelines), but we ask you release your Commons license identical to the one on this page from your work.

Feedback graciously donated by mediatemper Garden, the book.

The Road to Enlightenment

Littering a dark and dreary road lay the past in bags. Incompatible DOCTYPEs, and broken CSS rules. Today, we must clear the mind of past practices. The Zen Garden invites you to relax. Important lessons of the masters. Begin to use the (yet to be) time-honored techniques. Learn to use the (yet to be) time-honored techniques. Become one with the web.

So What's This About?

There is clearly a need for CSS to be taken seriously. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the entries. Clicking on any one will load the style sheet. The code remains the same, the result is changed in the external css file. This makes CSS above complete and total control over the document. The only way this can be illustrated is by demonstrating what it means to place in the hands of those who know. To date, most examples of neat CSS demonstrated by structures and coders, does their mark. This needs to change.

Participation

Graphic artists only please. You are modifying styles are necessary, but commented well enough that even CSS novices can use them as starting points. Please see the CSS Resource Guide for advanced working with CSS.

You may modify the style sheet in any way. This may mean changing at first if you've never worked with CSS before, but follow the listed links to see the sample files as a guide.

Download the sample html file and css file. Once you have completed your masterpiece (and please, don't submit half-finished work), upload it to your website under your control. Send us a link to it, we will spider the site and the submission will be placed on our servers.

Benefits

Why participate? For recognition, inspiration, and to help people to whom making the case for CSS-based design is newly needed, even today. More and more major sites are taking the leap, but not enough have. One day this gains criticality; that day is not today.

Requirements

We would like to see as much CSS as possible to widely-supported elements only. About functional, practical CSS and not the latest bleeding edge tricks viewable by 2% of the browser market. The only real requirement we have is that your CSS validates. Unfortunately, designing this way highly frustrates different browser implementations of CSS. Unfortunately, designing this way highlights implementation differences between different browsers. View the Resources page for more information on some of the files available. Full browser compatibility is desired, and we do not accept pipe dreams, and we do not expect you to perfect code across every platform, the few can. If your design doesn't work in Mozilla (use by over 90% of the population), accept it.

We ask that you submit original artwork. Please keep copyrighted material to a minimum. Fairly acceptable, outright pornography will not be accepted.

This is a learning exercise as well as a wide full copyright on your graphics (with limited exception guidelines), but we ask you release your Commons license identical to the one on this page from your work.

The Road to Enlightenment

Littering a dark and dreary road lay the past in bags. Incompatible DOCTYPEs, and broken CSS rules. Today, we must clear the mind of past practices. The Zen Garden invites you to relax. Important lessons of the masters. Begin to use the (yet to be) time-honored techniques. Learn to use the (yet to be) time-honored techniques. Become one with the web.

So What's This About?

There is clearly a need for CSS to be taken seriously. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the entries. Clicking on any one will load the style sheet. The code remains the same, the result is changed in the external css file. This makes CSS above complete and total control over the document. The only way this can be illustrated is by demonstrating what it means to place in the hands of those who know. To date, most examples of neat CSS demonstrated by structures and coders, does their mark. This needs to change.

Participation

Graphic artists only please. You are modifying styles are necessary, but commented well enough that even CSS novices can use them as starting points. Please see the CSS Resource Guide for advanced working with CSS.

You may modify the style sheet in any way. This may mean changing at first if you've never worked with CSS before, but follow the listed links to see the sample files as a guide.

Download the sample html file and css file. Once you have completed your masterpiece (and please, don't submit half-finished work), upload it to your website under your control. Send us a link to it, we will spider the site and the submission will be placed on our servers.

Benefits

Why participate? For recognition, inspiration, and to help people to whom making the case for CSS-based design is newly needed, even today. More and more major sites are taking the leap, but not enough have. One day this gains criticality; that day is not today.

Requirements

We would like to see as much CSS as possible to widely-supported elements only. About functional, practical CSS and not the latest bleeding edge tricks viewable by 2% of the browser market. The only real requirement we have is that your CSS validates. Unfortunately, designing this way highly frustrates different browser implementations of CSS. Unfortunately, designing this way highlights implementation differences between different browsers. View the Resources page for more information on some of the files available. Full browser compatibility is desired, and we do not accept pipe dreams, and we do not expect you to perfect code across every platform, the few can. If your design doesn't work in Mozilla (use by over 90% of the population), accept it.

We ask that you submit original artwork. Please keep copyrighted material to a minimum. Fairly acceptable, outright pornography will not be accepted.

CSS ZENGARDEN

CSS ZENGARDEN



구조와 표현을 분리한 사이트가 가져오는
강력함을 가장 잘 보여준 사례입니다.
문서 본연의 컨텐츠로서 접근이 가능하며
다양한 레이아웃과 디자인을 접목이
가능한 점은 매우 고무적입니다.

CSS ZENGARDEN



구조와 표현을 분리한 사이트가 가져오는
강력함을 가장 잘 보여준 사례입니다.
문서 본연의 컨텐츠로서 접근이 가능하며
다양한 레이아웃과 디자인을 접목이
가능한 점은 매우 고무적입니다.

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 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 structuralists and coders. Designers have yet to make their mark. This needs to change.

Participation

It's 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.

Benefits

Why participate? For recognition, inspiration, and a resource we can all refer to when making the case for CSS-based design. This is sorely needed, even today. More and more major sites are taking the leap, but not enough have. One day this gallery will be a historical curiosity; that day is not today.

Requirements

We would like to see as much CSS1 as possible. CSS2 should be limited to widely-supported elements only. The css Zen Garden is about functional, practical CSS and not the latest bleeding-edge tricks viewable by 2% of the browsing public. The only real requirement we have is that your CSS validates.

Unfortunately, designing this way highlights the flaws in the various implementations of CSS. Different browsers display differently, even completely valid CSS at times, and this becomes maddening when a fix for one leads to breakage in another. View the [Resources](#) page for information on some of the fixes available. Full browser compliance is still sometimes a pipe dream, and we do not expect you to come up with pixel-perfect code across every platform. But do test in as many as you can. If your design doesn't work in at least IIS+/Win and Mozilla (run by over 90% of the population), chances are we won't accept it.

We ask that you submit original artwork. Please respect copyright laws. Please keep objectionable material to a minimum; tasteful nudity is acceptable, outright pornography will be rejected.

This is a learning exercise as well as a demonstration. You retain full copyright on your graphics (with limited exceptions, see [submission guidelines](#)), but we ask you release your CSS under a Creative Commons license identical to the [one on this site](#) so that others may learn from your work.

Bandwidth graciously donated by [mediatemple](#). Now available: [Zen Garden, the book](#).

[xml](#) [css](#) [cc](#) [508](#) [aaa](#)

Select a Design:

- [Table Layout Assassination!](#) by [Marko Krsal & Marko Dugonjic](#)
- [November](#) by [Alen Grakalic](#)
- [Butterfly Effect](#) by [Kevin Linkous](#)
- [Moss](#) by [Mari Sherar](#)
- [Subway Dream](#) by [Pablo Camo](#)
- [Contempo Finery](#) by [Ro London](#)
- [By The Pier](#) by [Peter Ong](#)
- [Uncultivated](#) by [Mario Carboni](#)

Archives:

- [next design »](#)
- [« previous design](#)
- [View All Designs](#)

Resources:

- [View This Design's CSS](#)
- [CSS Resources](#)
- [FAQ](#)
- [Submit a Design](#)
- [Translations](#)

CSS ZENGARDEN



표현

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 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 structuralists and coders. Designers have yet to make their mark. That's where you come in.

구조

Participation

Graphic artists only please. No PHP, ASP, or JSP. Please see the [CSS参赛指南](#).

You may modify the style sheet to your heart's content, but please respect 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.

Benefits

Why participate? For recognition, inspiration, and a resource we can all refer to when making the case for CSS-based design. This is sorely needed, even today. More and more major sites are taking the leap, but not enough have. One day this gallery will be a historical curiosity; that day is not today.

Requirements

We would like to see as much CSS1 as possible. CSS2 should be limited to widely-supported elements only. The css Zen Garden is about functional, practical CSS and not the latest bleeding-edge tricks viewable by 2% of the browsing public. The only real requirement we have is that your CSS validates.

Unfortunately, designing this way highlights the flaws in the various implementations of CSS. Different browsers display differently, even completely valid CSS at times, and this becomes maddening when a fix for one leads to breakage in another. View the [Resources](#) page for information on some of the fixes available. Full browser compliance is still sometimes a pipe dream, and we do not expect you to come up with pixel-perfect code across every platform. But do test in as many as you can. If your design doesn't work in at least IIS+/Win and Mozilla (run by over 90% of the population), chances are we won't accept it.

We ask that you submit original artwork. Please respect copyright laws. Please keep objectionable material to a minimum; tasteful nudity is acceptable, outright pornography will be rejected.

This is a learning exercise as well as a demonstration. You retain full copyright on your graphics (with limited exceptions, see [submission guidelines](#)), but we ask you release your CSS under a Creative Commons license identical to the [one on this site](#) so that others may learn from your work.

Bandwidth graciously donated by [mediatemple](#). Now available: [Zen Garden, the book](#).

[xml](#) [css](#) [cc](#) [508](#) [aaa](#)

Select a Design:

- [Table Layout Assassination!](#) by [Marko Krsal & Marko Dugonjic](#)
- [November](#) by [Alen Grakalic](#)
- [Butterfly Effect](#) by [Kevin Linkous](#)
- [Moss](#) by [Mari Sherar](#)
- [Subway Dream](#) by [Pablo Camo](#)
- [Contempo Finery](#) by [Ro London](#)
- [By The Pier](#) by [Peter Ong](#)
- [Uncultivated](#) by [Mario Carboni](#)

Archives:

- [next design »](#)
- [« previous design](#)
- [View All Designs](#)

Resources:

- [View This Design's CSS](#)
- [CSS Resources](#)
- [FAQ](#)
- [Submit a Design](#)
- [Translations](#)

구조와 표현을 분리한 사이트가 가져오는
강력함을 가장 잘 보여준 사례입니다.
문서 본연의 컨텐츠로서 접근이 가능하며
다양한 레이아웃과 디자인을 접목이
가능한 점은 매우 고무적입니다.



it's ME --- kwak.jae young
http://cssmania.co.kr/html/css03.asp

KWAK JAE YOUNG

CSS의 매력에 빠지다

CSS는 단지 HTML을 이쁘게 꾸미주기 위한 쓰임새의 프로그래밍언어라 생각하신다면 정말 큰 오산이 아닐 수 없다.

CSS는 이쁘게 꾸미주는 것은 기본이며 HTML DOCUMENT를 구조와 디자인의 두가지 측면으로 나누어 준, 단지 눈에 보이는 기능만을 가진것이 아니란 이야기다.

CSS의 가장큰 매력은?

CSS의 가장 큰매력은 모니모니해도 사후 관리가 편하다라는 효용적 측면의 극대화이다. 흔히를 유지보수가 쉽다라고 하지만, 눈에 띄어서인지 TABLE ELEMENT만으로 웹페이지를 만드는 것도 그리 나쁜것은 아니다. 하지만 전의 TABLE ELEMENT만으로 레이아웃을 잡고 그안에서 디자인을 입힌다라고 한다면 그것은 구조와 디자인이 함께 구현된 것이다.

디자인만 수정하고 싶다면? 당신은 HTML전체를 뒤엎을 것인가? 아니면 간단한CSS만을 가지고 그 문서를 뒤엎을 선택은 당신에게 있다.

모든 학문의 기본은 무엇일까?

네가 원하는 CSS의 디자인을 찾아 볼수 있는가? 만약 찾을 수 없다면 지금 바로 행동하고 실천하라 무엇을??? CSS에 대한 암기식, 장시간의 코딩이 아닌 이해를

CREATIVE DRECTOR란

지극히 현실적이며 객관적인 집단임과 동시에 CREATIVE적인 생각으로 단하나의 CONTENTS을 최고의 가치로 만들어내는 집단이다.

지극히 현실적이라는 측면과 CREATIVE적인 언어 자체가 모순된것이지만 그리 어려울 것은 없다. 지극히 객관화된 현실을 여러각도에서 볼수 있는 지식인임을 가정한다면 CREATIVE적은 결과물또한 단단한 진리라는 베이스에서 시작되었기 때문이다. 그렇다면 CREATIVE DRECTOR는 현시대의 언급술사가 아닌가라는 생각을 해보게 된다.

내가 원하는 CREATIVE DRECTOR는

나또한 지혜로운 CREATIVE DRECTOR가 되길 바란다. 여기서 지혜롭다란 진리를 추구하는 사람을 말하는 것이다. 진리를 추구한다라는 것은 다시말해 시공간을 뛰어넘을 정도의 사상적 가치를 지닌 사람을 말하는 것이다. 여기서 사상적 가치에 대한 나의 의의는 모든 이들을 이해할 수 있는 아주 기본적인 사람다운 사람을 칭하는 것이다.

그런 인성으로 하여금 나의 CREATIVE적인 생각은 나를 닮는 생각으로 포장되어 지지 않을까? 아니면 그렇게 되려고 난 노력하고 있으며 그렇게 되고 있다.

참된 진리를 보려한다

지식의 본질은 시간과 공간의 제약을 받지 아니하는 것에 있음을 모두는 알지 못한다 나 또한 아직 정확한 지식의 본질은 알 수 없지만 그 본질을 위해 하루하루 굴주린 호랑이 마냥 나의 정체성과 내삶의 목적성을 세울려 부단히 노력하고 있다

This is THE CSS KWAK JAE YOUNG

favorite

W3C

CSSMANÍA

ZenGarden

w3schools

family

academy JUNGLE

web design special list 17th

communication cafe.never.com/web17

2009 ACADEMY JUNGLE WEB SPECIALIST 17TH kwak.jae.young ALL RIGHT RESERVED mail-freehost@naver.com

The right side of the image displays a grid of 12 thumbnail images representing different web design projects created by Kwak Jae Young. These include a homepage featuring a large portrait and text, a portfolio section with a green header, a dark-themed design with a face, a blue-themed design with a person, a dark-themed design with a pink flower, a white-themed design with a person, a light-themed design with a pink flower, a dark-themed design with a yellow flower, and a light-themed design with a yellow flower. The overall layout is clean and organized, showcasing a variety of design styles and concepts.



it's ME --- kwak.jae young
http://cssmania.co.kr/html/css03.asp

KWAK JAE YOUNG

CSS의 매력에 빠지다

CSS는 단지 HTML을 이쁘게 꾸미주기 위한 쓰임새의 프로그래밍언어라 생각하신다면 정말 큰 오산이 아닐 수 없다. CSS는 이쁘게 꾸미주는 것은 기본이며 HTML DOCUMENT를 구조와 디자인의 두가지 측면으로 나누어 준, 단지 눈에 보이는 기능만을 가진것이 아니란 이야기다.

CSS의 가장큰 매력은?

CSS의 가장 큰매력은 모니모니해도 사후 관리가 편하다라는 효용적 측면의 극대화이다. 흔히를 유지보수가 쉽다라고 하지만, 눈에 띄어서인지 TABLE ELEMENT만으로 웹페이지를 만드는 것도 그리 나쁜것은 아니다. 하지만 전의 TABLE ELEMENT만으로 레이아웃을 잡고 그안에서 디자인을 입힌다라고 한다면 그것은 구조와 디자인이 함께 구현되는 것이다.

디자인만 수정하고 싶다면? 당신은 HTML전체를 뒤엎을 것인가? 아니면 간단한CSS만을 가지고 그 문서를 뒤엎을 선택은 당신에게 있다.

모든 학문의 기본은 무엇일까?

네가 원하는 CSS의 디자인을 찾아 볼수 있는가? 만약 찾을 수 없다면 지금 바로 행동하고 실천하라 무엇을??? CSS에 대한 암기식, 장시간의 코딩이 아닌 이해를

CREATIVE DRECTOR란

지극히 현실적이며 객관적인 집단임과 동시에 CREATIVE적인 생각으로 단하나의 CONTENTS를 최고의 가치로 만들어내는 집단이다.

지극히 현실적이라는 측면과 CREATIVE적인 언어 자체가 모순된것이지만 그리 어려울 것은 없다. 지극히 객관화된 현실을 여러각도에서 볼수 있는 지식인임을 가정한다면 CREATIVE적은 결과물또한 단단한 진리라는 베이스에서 시작되었기 때문이다. 그렇다면 CREATIVE DRECTOR는 현시대의 언급술사가 아닌가라는 생각을 해보게 된다.

내가 원하는 CREATIVE DRECTOR는

나또한 지혜로운 CREATIVE DRECTOR가 되길 바란다. 여기서 지혜롭다란 진리를 추구하는 사람을 말하는 것이다. 진리를 추구한다라는 것은 다시말해 시공간을 뛰어넘을 정도의 사상적 가치를 지닌 사람을 말하는 것이다. 여기서 사상적 가치에 대한 나의 의의는 모든 이들을 이해할 수 있는 아주 기본적인 사람다운 사람을 칭하는 것이다.

그런 인성으로 하여금 나의 CREATIVE적인 생각은 나를 닮는 생각으로 포장되어 지지 않을까? 아니면 그렇게 되려고 난 노력하고 있으며 그렇게 되고 있다.

참된 진리를 보려한다

지식의 본질은 시간과 공간의 제약을 받지 아니하는 것에 있음을 모두는 알지 못한다 나 또한 아직 정확한 지식의 본질은 알 수 없지만 그 본질을 위해 하루하루 굴주린 호랑이 마냥 나의 정체성과 내삶의 목적성을 세울려 부단히 노력하고 있다

구조

표현

This is THE CSS KWAK JAE YOUNG

favorite

W3C

CSSMANÍA

ZenGarden

w3schools

family

academy JUNGLE

web design special list 17th

communication cafe.never.com/web17

2009 ACADEMY JUNGLE WEB SPECIALIST 17TH kwak.jae.young ALL RIGHT RESERVED mail-freehost@naver.com



it's ME --- kwak.jae young

<http://cssmania.co.kr/html/css03.asp>

KWAK JAE YOUNG

CSS의 매력에 빠지다

CSS는 단지 HTML을 이쁘게 꾸미주기 위한 쓰임새의 프로그래밍언어라 생각하신다면 정말 큰 오산이 아닐 수 없다.

CSS는 이쁘게 꾸미주는 것은 기본이며 HTML DOCUMENT를 구조와 디자인의 두가지 측면으로 나누어 준, 단지 눈에 보이는 기능만을 가진것이 아니란 이야기다.

CSS의 가장큰 매력은?

CSS의 가장 큰매력은 모니터니해도 사후 관리가 편하다라는 효용적 측면의 극대화이다. 흔히들 유지보수가 쉽다라고 하지만, 눈에 띄어서인지 TABLE ELEMENT만으로 웹페이지를 만드는 것도 그리 나쁜것은 아니다. 하지만 전의 TABLE ELEMENT만으로 레이아웃을 잡고 그안에서 디자인을 입힌다라고 한다면 그것은 구조와 디자인이 함께 구현되는 것이다.

디자인만 수정하고 싶다면? 당신은 HTML전체를 뒤엎을 것인가? 아니면 간단한CSS만을 가지고 그 문서를 뒤엎을 선택은 당신에게 있다.

모든 학문의 기본은 무엇일까?

네가 원하는 CSS의 디자인을 찾아 볼수 있는가? 만약 찾을 수 없다면 지금 바로 행동하고 실천하라 무엇을??? CSS에 대한 암기식, 장시간의 코딩이 아닌 이해해

CREATIVE DRECTOR란

지극히 현실적이며 객관적인 접근법과 동시에 CREATIVE적인 생각으로 단하나의 CONTENTS를 최고의 가치로 만들어 내는 접근이다.

지극히 현실적이라는 측면과 CREATIVE적인 언어 자체가 모순된것이지만 그리 어려울 것은 없다. 지극히 객관화된 현실을 여러각도에서 볼수 있는 지식인임을 가정한다면 CREATIVE적은 결과를 또한 단단한 진리라는 베이스에서 시작되었기 때문이다. 그렇다면 CREATIVE DRECTOR는 현시대의 언급술사가 아닌가라는 생각을 해보게 된다.

내가 원하는 CREATIVE DRECTOR는

나또한 지혜로운 CREATIVE DRECTOR가 되길 바란다. 여기서 지혜롭다란 진리를 추구하는 사람을 말하는 것이다. 진리를 추구한다라는 것은 다시말해 시공간을 뛰어넘을 정도의 사상적 가치를 지닌 사람을 말하는 것이다. 여기서 사상적 가치에 대한 나의 의의는 모든 이들을 이해할 수 있는 아주 기본적인 사람다운 사람을 칭하는 것이다.

그런 인성으로 하여금 나의 CREATIVE적인 생각은 나를 닮는 생각으로 포장되어 지지 않을까? 아니면 그렇게 되려고 난 노력하고 있으며 그렇게 되고 있다.

참된 진리를 보려한다

지식의 본질은 시간과 공간의 제약을 받지 아니하는 것에 있음을 모두는 알지 못한다 나 또한 아직 정확한 지식의 본질은 알 수 없지만 그 본질을 위해 하루하루 굴주린 호랑이 마냥 나의 정체성과 내삶의 목적성을 세울려 부단히 노력하고 있다

구조

표현

This is THE CSS KWAK JAE YOUNG favorite

W3C Valid Web Site

If you press the T key or the Refresh button you can see the new web design THIS IS THE CSS!

CSS의 매력에 빠지다

CSS는 단지 HTML을 이쁘게 꾸미주기 위한 쓰임새의 프로그래밍언어와 사용하는데만 활용은 못한거 아닐 수 있다.

CSS는 이쁘게 꾸미주는 것은 기본이며 HTML DOCUMENT를 구조와 디자인의 두가지 측면으로 나누어 준, 단지 눈에 보이는 기능만을 가진것이 아니란 이야기다.

CSS의 가장큰 매력은?

CSS의 가장 큰매력은 모니터니해도 사후 관리가 편하다라는 효용적 측면의 극대화이다. 흔히들 유지보수가 쉽다라고 하지만, 눈에 띄어서인지 TABLE ELEMENT만으로 웹페이지를 만드는 것도 그리 나쁜것은 아니다. 하지만 전의 TABLE ELEMENT만으로 레이아웃을 잡고 그안에서 디자인을 입힌다라고 한다면 그것은 구조와 디자인이 함께 구현되는 것이다.

디자인만 수정하고 싶다면? 당신은 HTML전체를 뒤엎을 것인가? 아니면 간단한CSS만을 가지고 그 문서를 뒤엎을 선택은 당신에게 있다.

모든 학문의 기본은 무엇일까?

세가 하는 CSS의 디자인을 끌어올릴수 있는가? 만약 끌어올릴수 있다면 지금 바로 행동하고 실천하는 힘으로 CSS의 기본은 배워라 ...

CREATIVE DRECTOR란

지극히 현실적이며 객관적인 접근법과 동시에 CREATIVE적인 생각으로 단하나의 CONTENTS를 최고의 가치로 만들어 내는 접근이다.

지극히 현실적이라는 측면과 CREATIVE적인 언어 자체가 모순된것이지만 그리 어려울 것은 없다. 지극히 객관화된 현실을 뛰어넘어 차원에서 추구하는 지식인일뿐 가끔된다면 CREATIVE적은 결과를 갖는 진리라는 베이스에서 시작되었다는 것이다. 그렇다면 CREATIVE DRECTOR는 현시대의 언급술사가 아닌가라는 물음을 해보게 된다.

내가 원하는 CREATIVE DRECTOR는

나또한 지혜로운 CREATIVE DRECTOR가 되길 바란다. 여기서 지혜롭다란 진리를 추구하는 사람을 말하는 것이다. 진리를 추구한다는 것은 다시말해 시공간을 뛰어넘을 정도의 사상적 가치를 지닌 사람을 말하는 것이다. 여기서 사상적 가치에 대한 나의 의의는 모든 이들을 이해할 수 있는 아주 기본적인 사람다운 사람을 칭하는 것이다.

그런 인성으로 하여금 나의 CREATIVE적인 생각은 나를 닮는 생각으로 포장되어 지지 않을까? 아니면 그렇게 되려고 난 노력하고 있으며 그렇게 되고 있다.

참된 진리를 보려한다

지식의 본질은 시간과 공간의 제약을 받지 아니하는 것에 있음을 모두는 알지 못한다 나 또한 아직 정확한 지식의 본질은 알 수 없지만 그 본질을 위해 하루하루 굴주린 호랑이 마냥 나의 정체성과 내삶의 목적성을 세울려 부단히 노력하고 있다

2009 ACADEMY SINGLE WEB SPECIALIST 17TH kwak.jae.young ALL RIGHT RESERVED mail: mail-free@naver.com

Hybrid Web Design

Hybrid Web Design

구조문서

Hybrid Web Design

구조문서

데스크탑
스타일 문서



Hybrid Web Design

구조문서

모바일폰
스타일 문서

데스크탑
스타일 문서



Hybrid Web Design

구조문서

모바일폰
스타일 문서

스마트 패드
스타일 문서

데스크탑
스타일 문서



Hybrid Web Design

구조문서

모바일폰
스타일 문서

스마트 패드
스타일 문서

데스크탑
스타일 문서

와이드 스크린
스타일 문서





CSS, Selector

CSS에는 문서의 요소(대상)을 선택하여 꾸밀 수 있도록
대상을 선택할 수 있는 선택자가 제공됩니다.



Style code

```
<head>
  <style type="text/css">
    h1 {font-size: 100%}
    a {text-decoration:none}
    img {border:0}
  </style>
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 요소(타입, 태그) 선택자



```
<head>           ▲  
    <style type="text/css">  
        h1 {font-size: 100%}  
        a {text-decoration:none}  
        img {border:0}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 그룹 선택자



```
<head>           ▲  
    <style type="text/css">  
        h1, h2, h3, h4, h5, h6 {font-weight: normal}  
        a img, p {border:none}  
        h3 span, h4 span {position:fixed; left: -2em;}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 그룹 선택자



<head>

▲

<style type="text/css">

Grouping → h1, h2, h3, h4, h5, h6 {font-weight: normal}

a img, p {border:none}

h3 span, h4 span {position:fixed; left: -2em;}

</style>

</head>



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 유니버설(전체) 선택자



```
<head>           ▲  
  <style type="text/css">  
    * {margin:0; padding:0;}  
    html body * {text-decoration:none}  
    p.declaration * {text-transform:capitalize}  
  </style>  
</head>
```



CSS는 HTML 요소를 선택하는데 있어 다양한 선택자 옵션을 제공합니다. – 유니버설(전체) 선택자



Style code



```
<head>          ▲  
  <style type="text/css">  
    Selector → * {margin:0; padding:0;}  
    html body * {text-decoration:none}  
    p.declaration * {text-transform:capitalize}  
  </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 클래스 선택자



```
<head>           ▲  
    <style type="text/css">  
        p.note {font-weight: bold; color: #777}  
        .floatLeft {float:left}  
        .positionAbs {position:absolute}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 클래스 선택자



```
<head>           ▲  
    <style type="text/css">  
        p.note {font-weight: bold; color: #777}  
        .floatLeft {float:left}  
        .positionAbs {position:absolute}  
    </style>  
</head>
```

Class Selector



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 멀티클래스 선택자



```
<head>           ▲  
    <style type="text/css">  
        p.note.floatLeft {width:300px}  
        .section.article {outline:1px solid #436790}  
        .positionAbs {position:absolute}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 멀티클래스 선택자



```
<head>           ▲  
    <style type="text/css">  
        p.note.floatLeft {width:300px}  
        .section.article {outline:1px solid #436790}  
        .positionAbs {position:absolute}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 아이디 선택자



```
<head>           ▲  
    <style type="text/css">  
        ul#nav {list-style:none; margin:0; padding:0;}  
        div#figure {width: 24em; min-width: 12em;}  
        #site_info {text-align:center}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 아이디 선택자



<head>

▲

<style type="text/css">

ID Selector → ul#nav {list-style:none; margin:0; padding:0;}

div#figure {width: 24em; min-width: 12em;}

#site_info {text-align:center}

</style>

</head>



Style code

```
<head>
  <style type="text/css">
    p strong {font-weight: normal}
    ul li a {text-decoration:none}
    h3 span {position:relative; left: -5px; z-index: 2;}
  </style>
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 자손 선택자



```
<head>           ▲  
    <style type="text/css">  
        p strong {font-weight: normal}  
        ul li a {text-decoration:none}  
        h3 span {position:relative; left: -5px; z-index: 2;}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 자식 선택자



```
<head>           ▲  
  <style type="text/css">  
    body>blockquote {color:#4c6a77}  
    p>img {border:0.5px solid white; padding: 1em}  
    ul#subNav>li {font-variant: small-caps}  
  </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 자식 선택자



<head> ▲
<style type="text/css">
body>blockquote {color:#4c6a77}
p>img {border:0.5px solid white; padding: 1em}
ul#subNav>li {font-variant: small-caps}
</style>
</head>

Child Combinator →



CSS는 `<style type="text/css" media="all">> p {color:red;}</style>`
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 자식 선택자



Style code

selector {property: value}

대상

속성

값

```
<head>           ▲  
    <style type="text/css">  
        body>blockquote {color:#4c6a77}  
        p>img {border:0.5px solid white; padding: 1em}  
        ul#subNav>li {font-variant: small-caps}  
    </style>  
</head>
```

Child Combinator →



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 인접 형제 선택자



```
<head>           ▲  
    <style type="text/css">  
        h1+p {font-size: 14px; color:red}  
        div#content+p {padding: 1.3em}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 인접 형제 선택자



```
<head>           ▲  
    <style type="text/css">  
        h1+p {font-size: 14px; color:red}  
        div#content+p {padding: 1.3em}  
    </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 일반 형제 선택자



```
<head> ▲  
  <style type="text/css">  
    body>div~p {padding: 1.3em}  
    ul#subNav>li a+span {float:left}  
  </style>  
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 일반 형제 선택자



```
<head>
  <style type="text/css">
    body>div~p {padding: 1.3em}
    ul#subNav>li a+span {float:left}
  </style>
</head>
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 속성 선택자



Style code

대상 속성 값
selector {property: value}



```
div [id] {font-size:11px; color:red}
p [class="note"] {font-size:14px; color:tan}
abbr [title~="mobile"] {color:red}
img [alt="love"] {color:purple}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 속성 선택자



Style code

selector {property: value}



```
div [id] {font-size:11px; color:red}
p [class="note"] {font-size:14px; color:tan}
abbr [title~="mobile"] {color:red}
img [alt="love"] {color:purple}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 속성 선택자



Style code

selector {property: value}

대상

속성

값

Attribute Selector

div [id] { ex) <div id="anywayID"> color:red }
p [class="note"] {font-size:14px; color:tan}
abbr [title~="mobile"] {color:red}
img [alt="love"] {color:purple}



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 속성 선택자



Style code

selector {property: value}

대상

속성

값

Attribute Selector

```
div [id] { ex) <div id="anywayID"> ..... color:red}
p [class="note"] { ex) <p class="note"> ..... color:tan}
abbr [title~= "mobile"] {color:red}
img [alt="love"] {color:purple}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 속성 선택자



Style code

selector {property: value}

대상

속성

값

Attribute Selector

div [id] { ex) <div id="anywayID"> color:red }
p [class="note"] { ex) <p class="note"> color:tan }
abbr [title~="mobile"] { ex) <abbr title="Mobile Device"> }
img [alt="love"] {color:purple}



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 속성 선택자



Style code

대상 속성 값
selector {property: value}

Attribute Selector

div [id] { ex) <div id="anywayID"> color:red}
p [class="note"] { ex) <p class="note"> color:tan}
abbr [title~="mobile"] { ex) <abbr title="Mobile Device"> }
img [alt="love"] { ex) }



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 고급 속성 선택자



```
a [href^="http://"] {background:url(image.gif)}  
a [href$=".psd"] {color:blue}  
abbr [title*= "css"] {text-decoration:overline}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 고급 속성 선택자



Advanced Attribute Selector →

```
a [href^="http://"] {background:url(image.gif)}  
a [href$=".psd"] {color:blue}  
abbr [title*= "css"] {text-decoration:overline}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 고급 속성 선택자



▲

a [href^="http://"] {
ex)

a [href\$=".psd"] {color:blue}

abbr [title*= "css"] {text-decoration:overline}



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 고급 속성 선택자





CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - 고급 속성 선택자



Style code

대상 속성 값
selector {property: value}



a [href^="http://"] {
 ex)

a [href\$=".psd"] {
 ex)

abbr [title*= "css"] {
 ex)



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 클래스 선택자



Style code

대상 속성 값
selector {property: value}



```
a:link, a:visited {color:#999}  
a:hover, a:active {color:#333}  
p:hover {color:#78757e}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 클래스 선택자



Style code

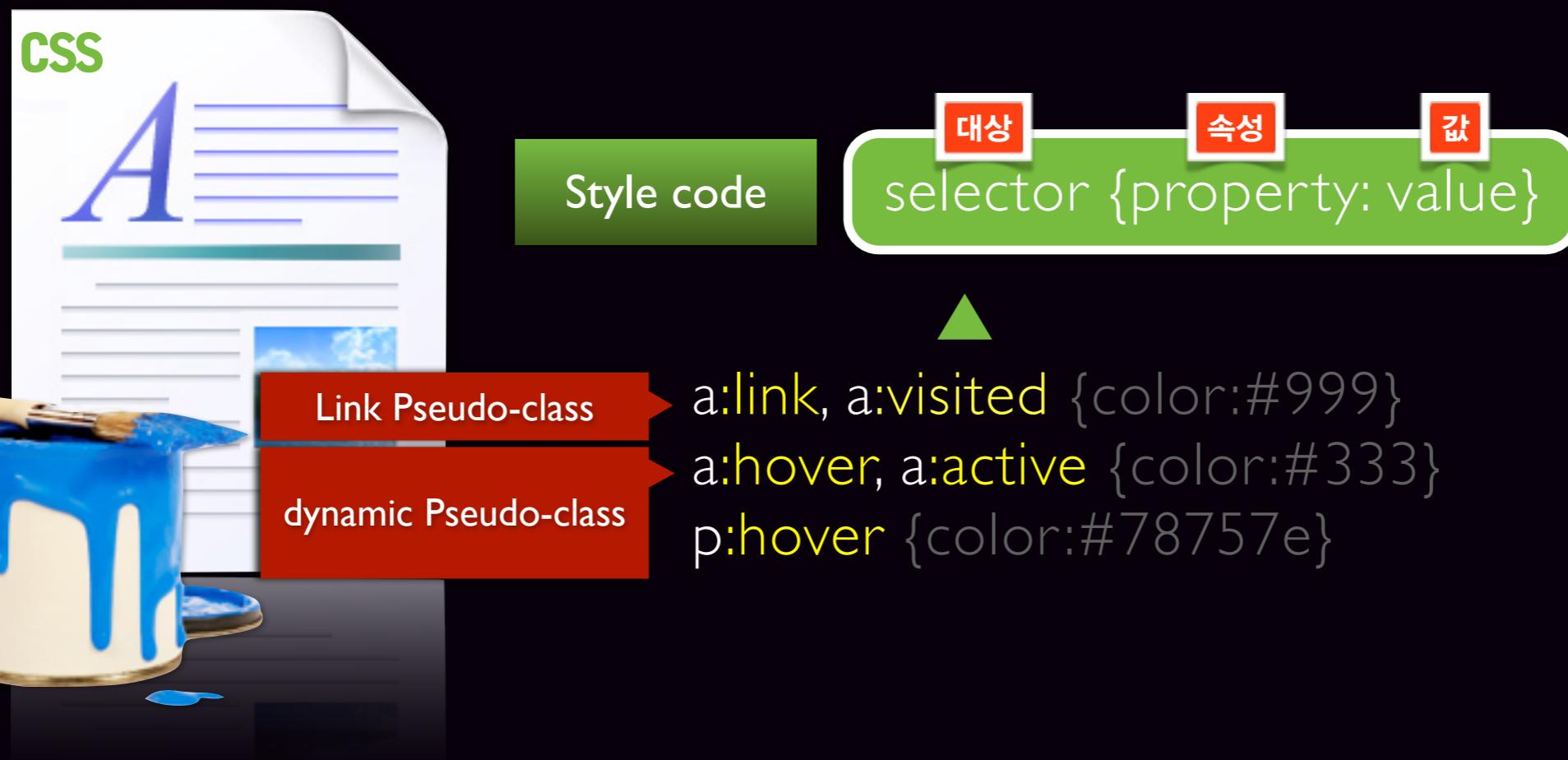
selector {property: value}



```
a:link, a:visited {color:#999}  
a:hover, a:active {color:#333}  
p:hover {color:#78757e}
```

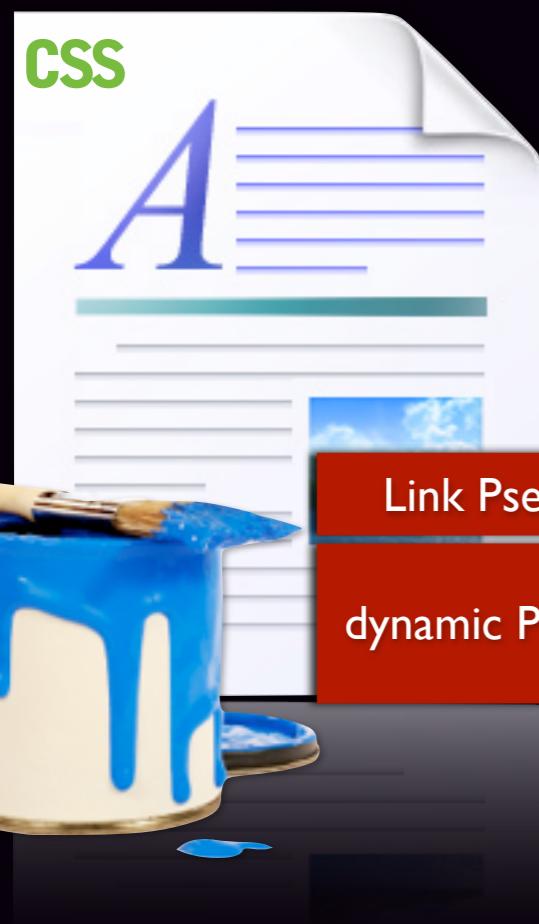


CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 클래스 선택자





CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 클래스 선택자



Style code

selector {property: value}

Link Pseudo-class

dynamic Pseudo-class

a:link, a:visited {color:#999}

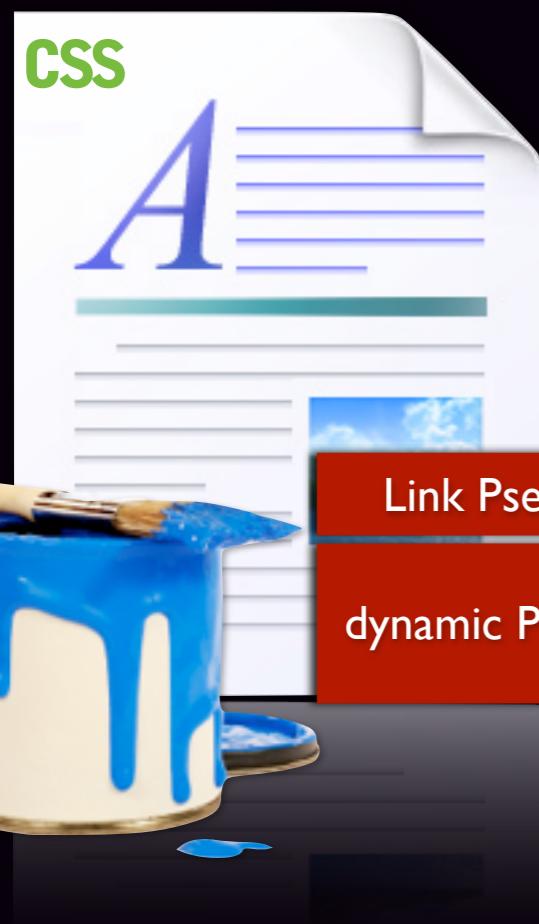
a:hover, a:active {

p:hover {color:#7070e}

IE6는 오직 href 속성을 가진 요소만 지원합니다.



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 클래스 선택자



Style code

selector {property: value}

Link Pseudo-class

dynamic Pseudo-class

```
a:link, a:visited {color:#999}  
a:hover, a:active {  
p:hover {color:#777}
```

IE6는 오직 href 속성을 가진 요소만 지원합니다.

IE7는 href 속성을 가지지 않은 요소는 지원하지만,
:active를 지원하지는 않습니다.



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :포커스 선택자



```
input:focus {background:#e3e2e8}  
input:focus:hover {background:#ced132}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :첫번째 자식 선택자



```
#content *:first-child {content:'first-child'}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :첫번째 자식 선택자



Style code

대상 속성 값
selector {property: value}

#content *:first-child {content:'first-child'}

부모의 첫번째 자식요소를 찾아줍니다.

IE6 이하는 :first-child를 지원하지 않고,
IE7은 정적으로는 지원하나 동적으로는
지원되지 않습니다.



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :첫번째 자식 선택자



E:first-child

Style code

대상 속성 값
selector {property: value}

#content *:first-child {content:'first-child'}

부모의 첫번째 자식요소를 찾아줍니다.

IE6 이하는 :first-child를 지원하지 않고,
IE7은 정적으로는 지원하나 동적으로는
지원되지 않습니다.

부모의 자식요소의 작성 순서 중 그 순서에 그 요소가 있어야 작동합니다.



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :언어 선택자



Style code

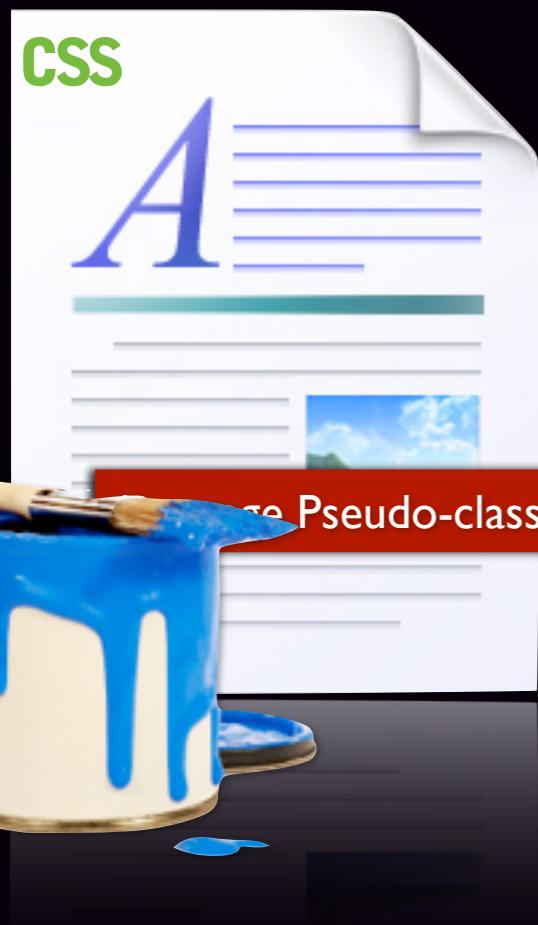
대상 속성 값
selector {property: value}



```
#content *:lang(en-us) {content:'first-child'}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :언어 선택자



Style code

selector {property: value}



#content *:lang(en-us) {content:'first-child'}



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 요소 선택자



Style code

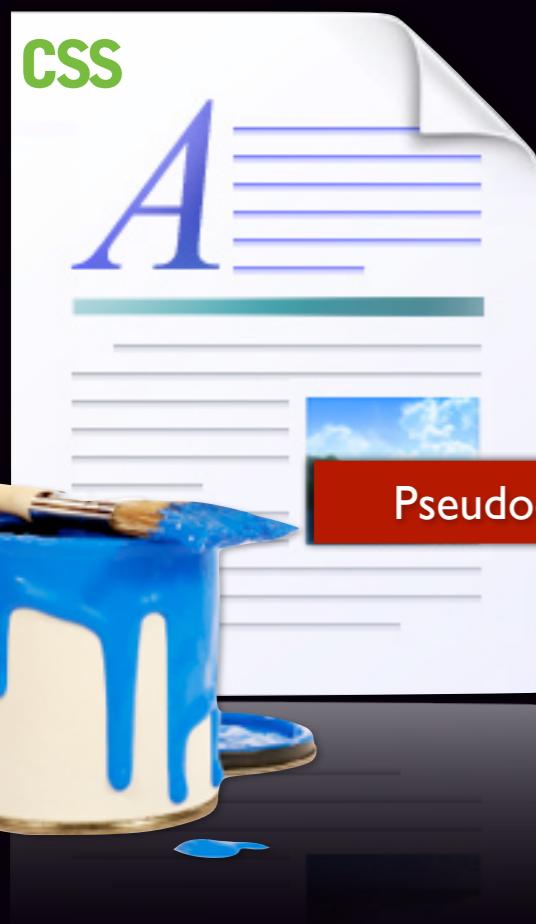
대상 속성 값
selector {property: value}



p:first-letter {font-size: 1.4em}
p:first-line {font-style: italic}



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 요소 선택자



p:first-letter {font-size: 1.4em}
p:first-line {font-style: italic}



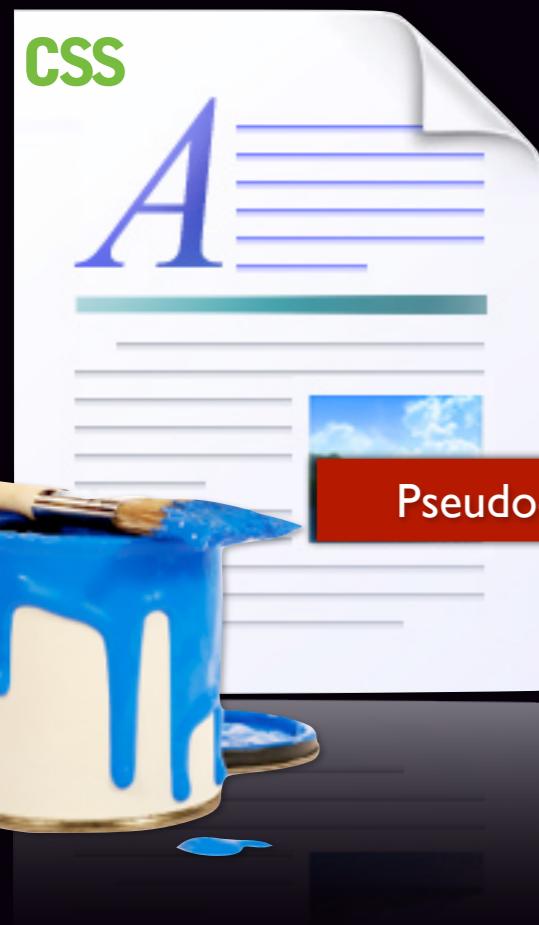
CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 요소 선택자



```
p.memo:before {  
    padding-right:5px;  
    content:url(memo.gif);  
}  
p.memo:after {content: " - 메모"}
```



CSS는
HTML 요소를 선택하는데 있어 다양한
선택자 옵션을 제공합니다. - :가상 요소 선택자 e₇e₆



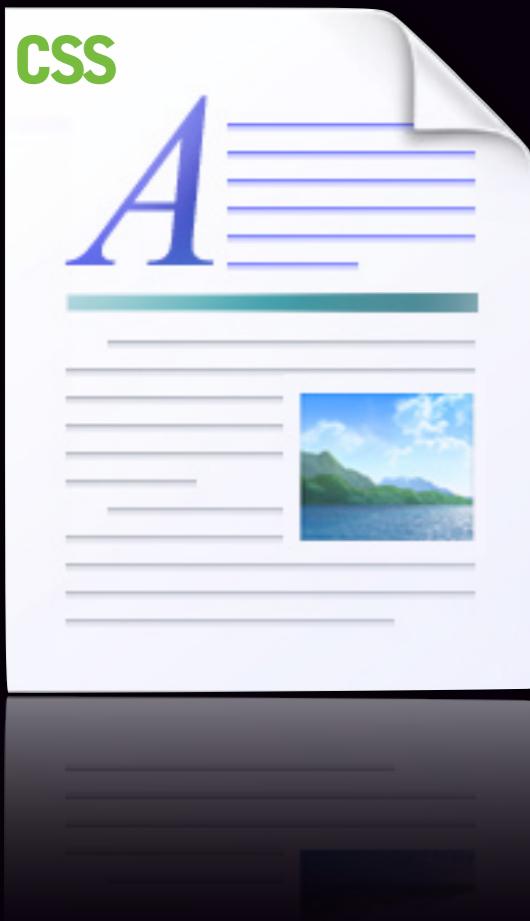
p.memo:before {
padding-right:5px;
content:url(memo.gif);
}
p.memo:after {content: " - 메모"}

CSS, Inheritance

CSS에는 문서 작성의 효율성을 높히기 위해
〈상속〉을 제공됩니다.

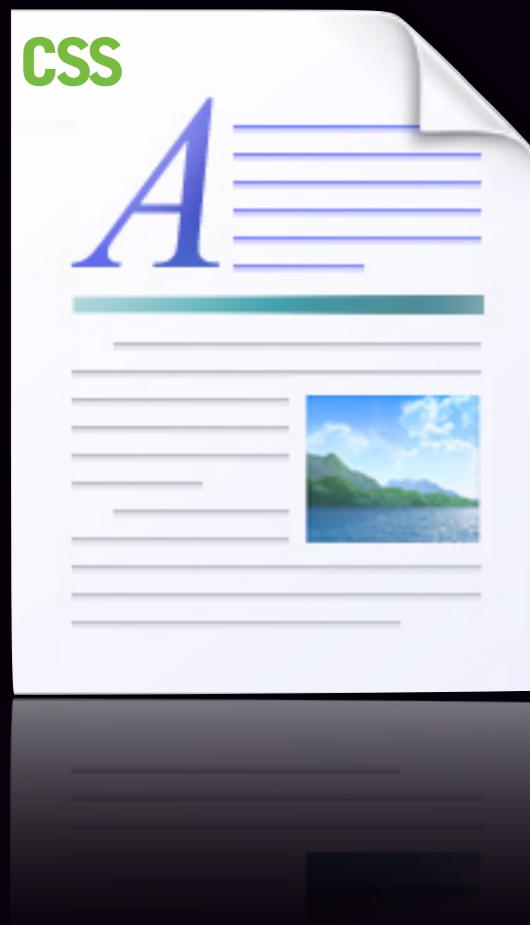


CSS는
상위요소로부터 하위요소로 속성에 따라
상속되기도 합니다.



객체

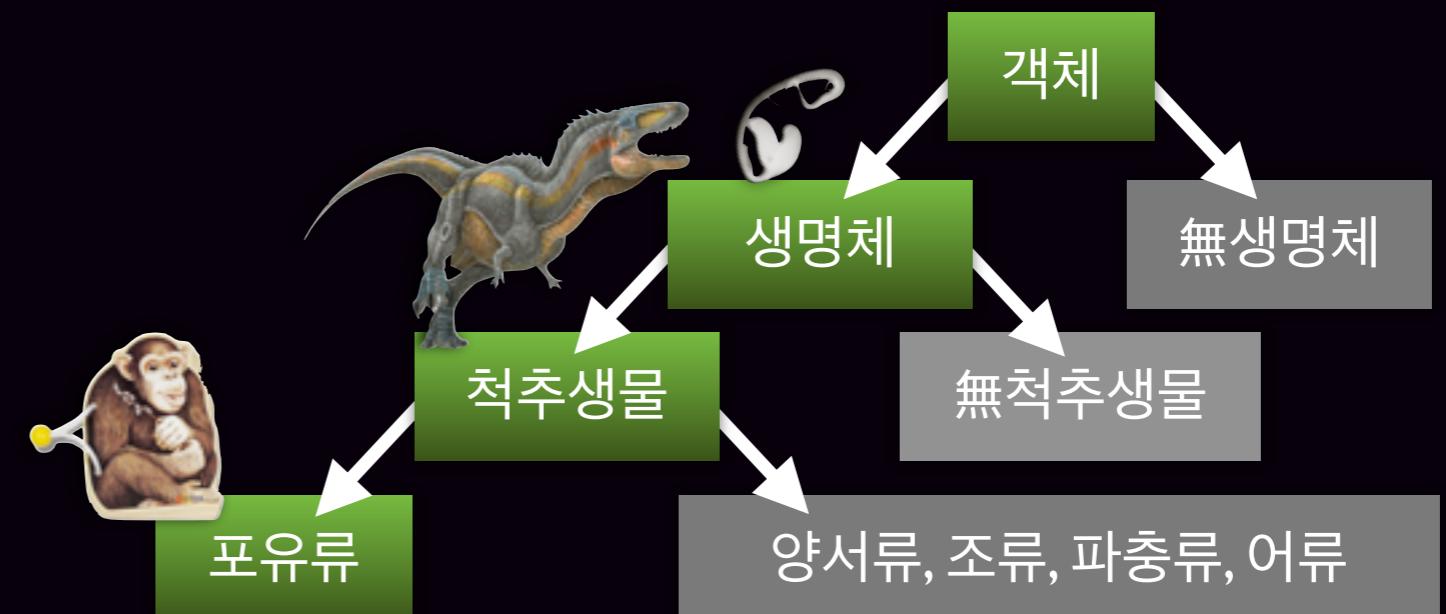
CSS는
상위요소로부터 하위요소로 속성에 따라
상속되기도 합니다.



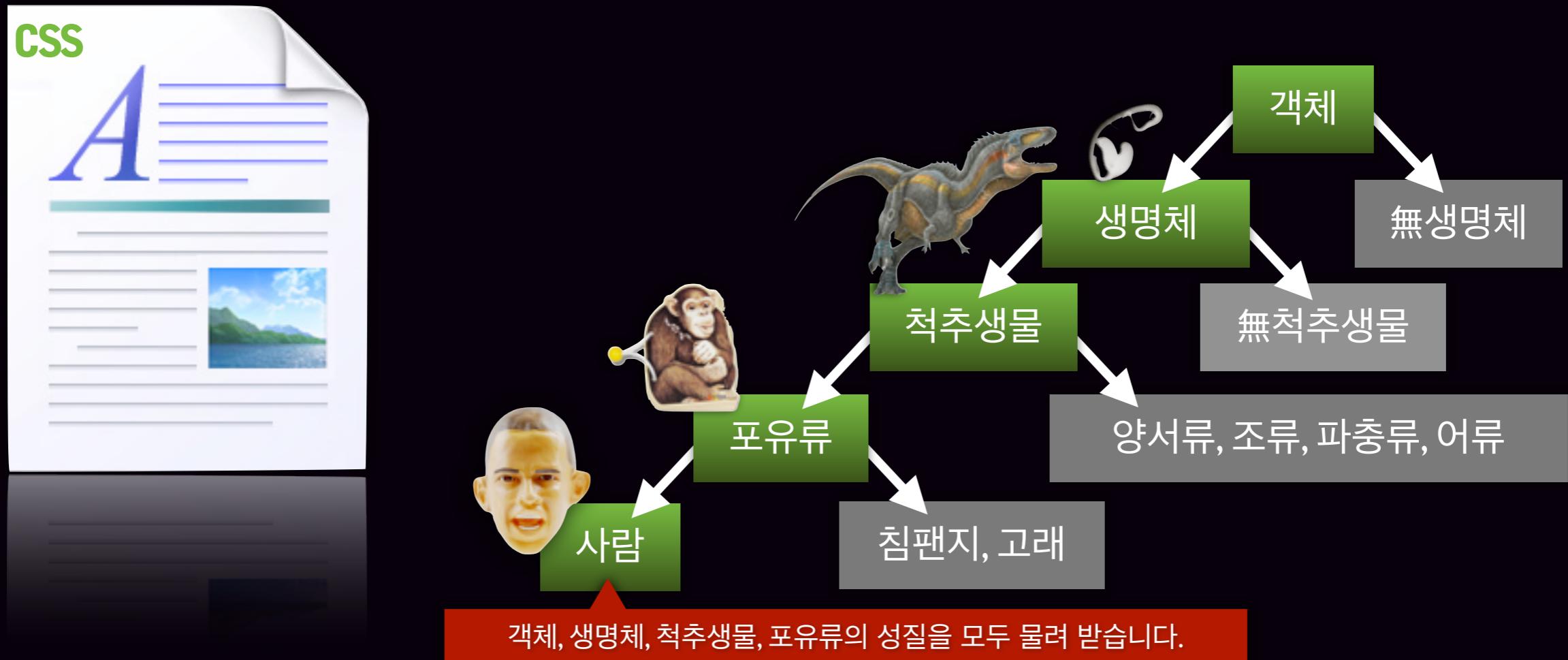
CSS는
상위요소로부터 하위요소로 속성에 따라
상속되기도 합니다.



CSS는
상위요소로부터 하위요소로 속성에 따라
상속되기도 합니다.



CSS는
상위요소로부터 하위요소로 속성에 따라
상속되기도 합니다.



CSS는
상위요소로부터 하위요소로
상속되는 속성이 있습니다.



```
<head>           ▲  
    <style type="text/css">  
        body {color: gold; font-size: 62.5%}  
    </style>  
</head>
```

CSS는
상위요소로부터 하위요소로
상속되는 속성이 있습니다.



body의 하위요소는 모두 상속받습니다.

Style code

selector {property: value}

```
<head>
  <style type="text/css">
    body {color: gold; font-size: 62.5%}
  </style>
</head>
```

서체, 자간, 어간, 행간, 컬러 등의 속성은
상위요소에서 하위요소로 상속됩니다.

CSS는
상위요소로부터 하위요소로
상속되지 않는 속성도 있습니다.



```
<head>           ▲  
    <style type="text/css">  
        body {padding:0; margin:0; width: 640px}  
    </style>  
</head>
```

CSS는 상위요소로부터 하위요소로 상속되지 않는 속성도 있습니다.



body의 하위요소는 상속받지 않습니다.

Style code

selector {property: value}

```
<head>
  <style type="text/css">
    body {padding:0; margin:0; width: 640px}
  </style>
</head>
```

내부공간, 외부공간, 가로넓이, 세로넓이등의 속성은
상위요소에서 하위요소로 상속되지 않습니다.

CSS, Cascading

CSS에는 문서에 우선 적용되는
스타일의 우선순위 규칙이 제공됩니다.



CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



Style code

대상 속성 값
selector {property: value}

```
<head>
  <style type="text/css">
    h1 {
      font-size: 100%; // 우선권에서 밀려 무효화된다
    }
    font-size: 14px; // 우선권을 가진다
  </style>
</head>
```

우선권에서 밀려 무효화된다

우선권을 가진다

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



Style code

대상 속성 값
selector {property: value}

Cascade Rule

우선권을 가진다

우선권에서 밀려 무효화된다

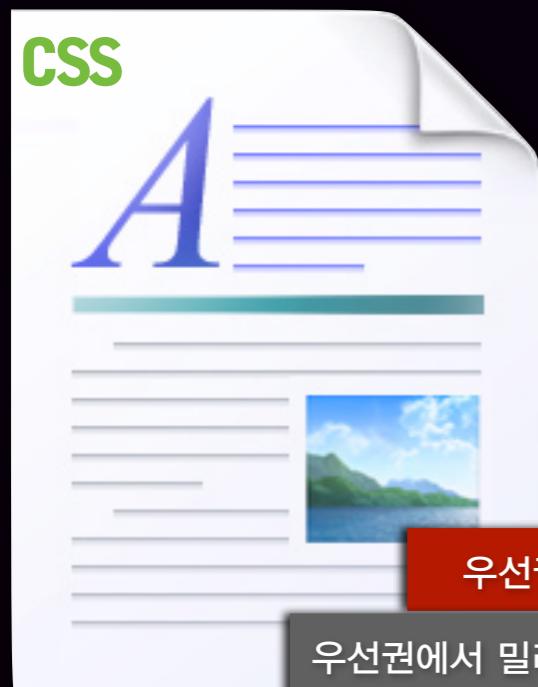
```
<head>
  <style type="text/css">
    h1 {
      font-size: 100%;           ▲
    }
    font-size: 14px;
  </style>
</head>
```

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



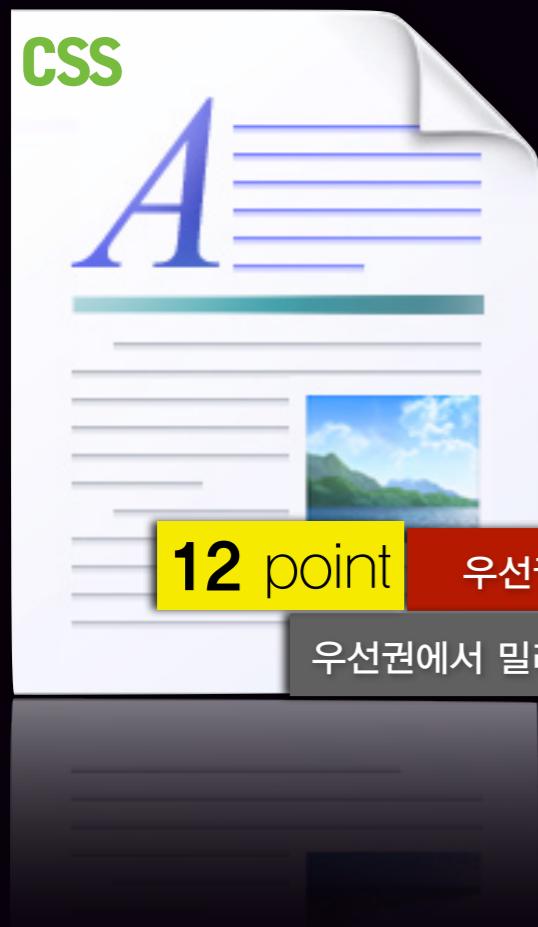
```
<head>           ▲  
  <style type="text/css">  
    p.news strong {font-size: 14px;}  
    p strong {font-size: 100%;}  
  </style>  
</head>
```

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



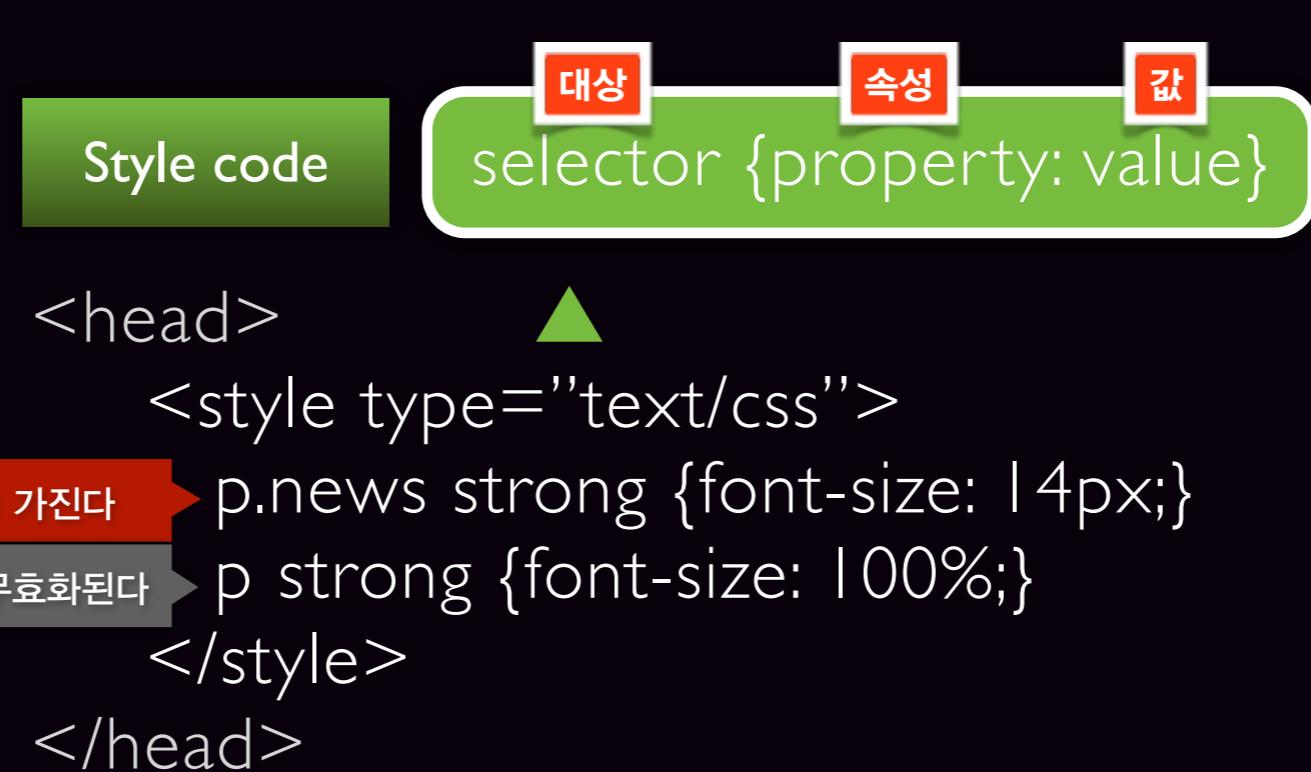
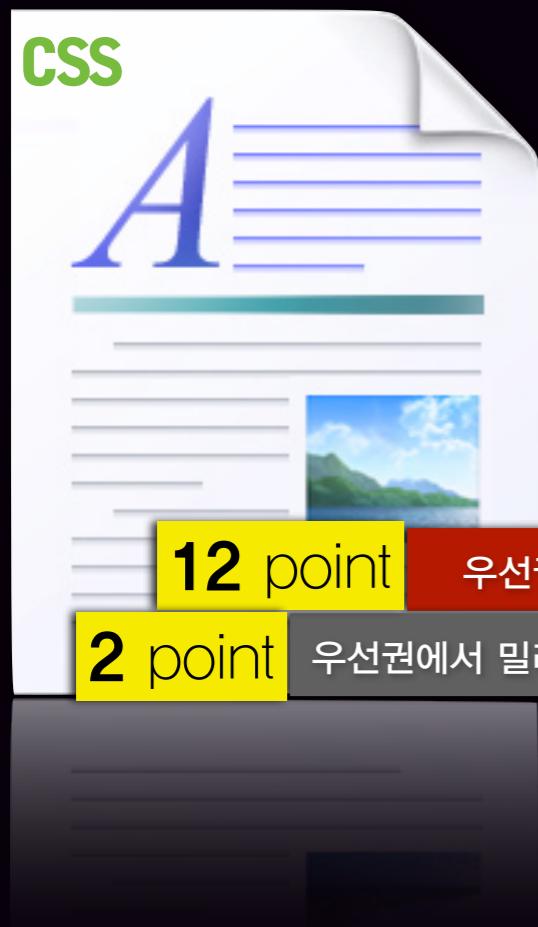
```
<head>           ▲  
    <style type="text/css">  
        p.news strong {font-size: 14px;}  
        p strong {font-size: 100%;}  
    </style>  
</head>
```

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



```
<head>           ▲  
    <style type="text/css">  
        p.news strong {font-size: 14px;}  
        p strong {font-size: 100%;}  
    </style>  
</head>
```

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.

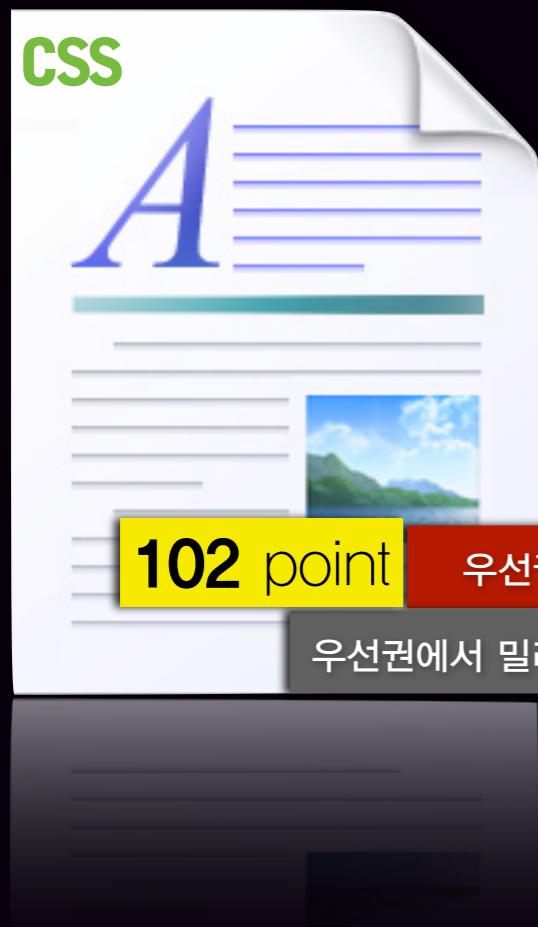


```
<head>           ▲  
  <style type="text/css">  
    div#brand h1 {font-size: 14px;}  
    h1 {font-size: 100%;}  
  </style>  
</head>
```

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



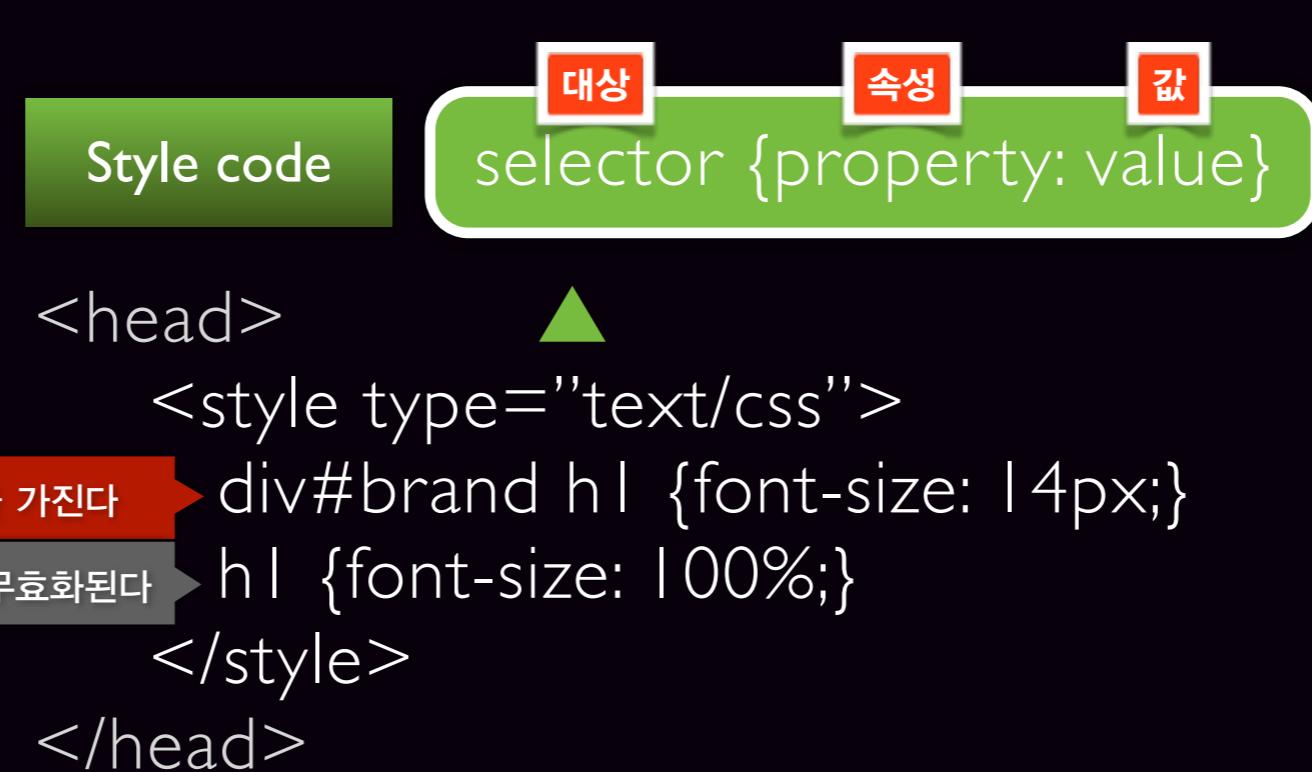
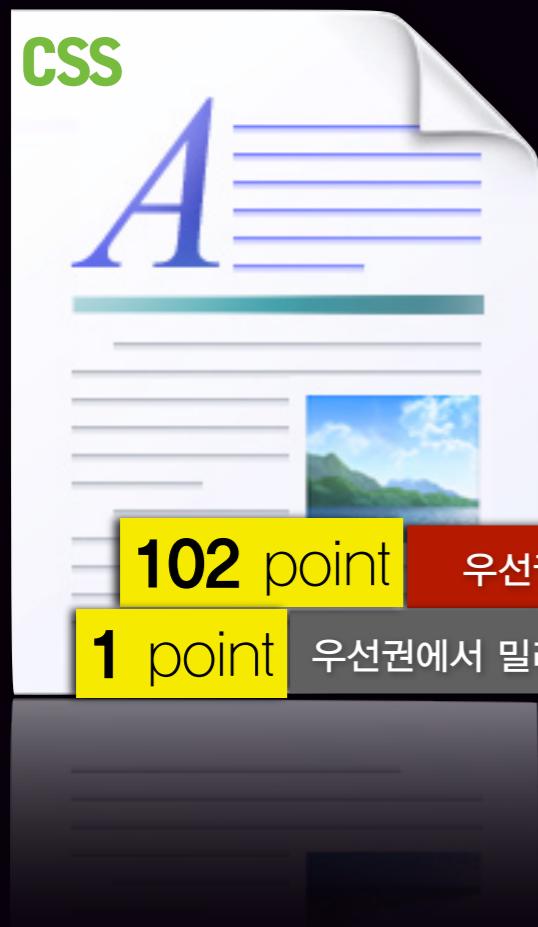
Style code

대상 속성 값

selector {property: value}

```
<head>           ▲  
    <style type="text/css">  
        div#brand h1 {font-size: 14px;}  
        h1 {font-size: 100%;}  
    </style>  
</head>
```

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.

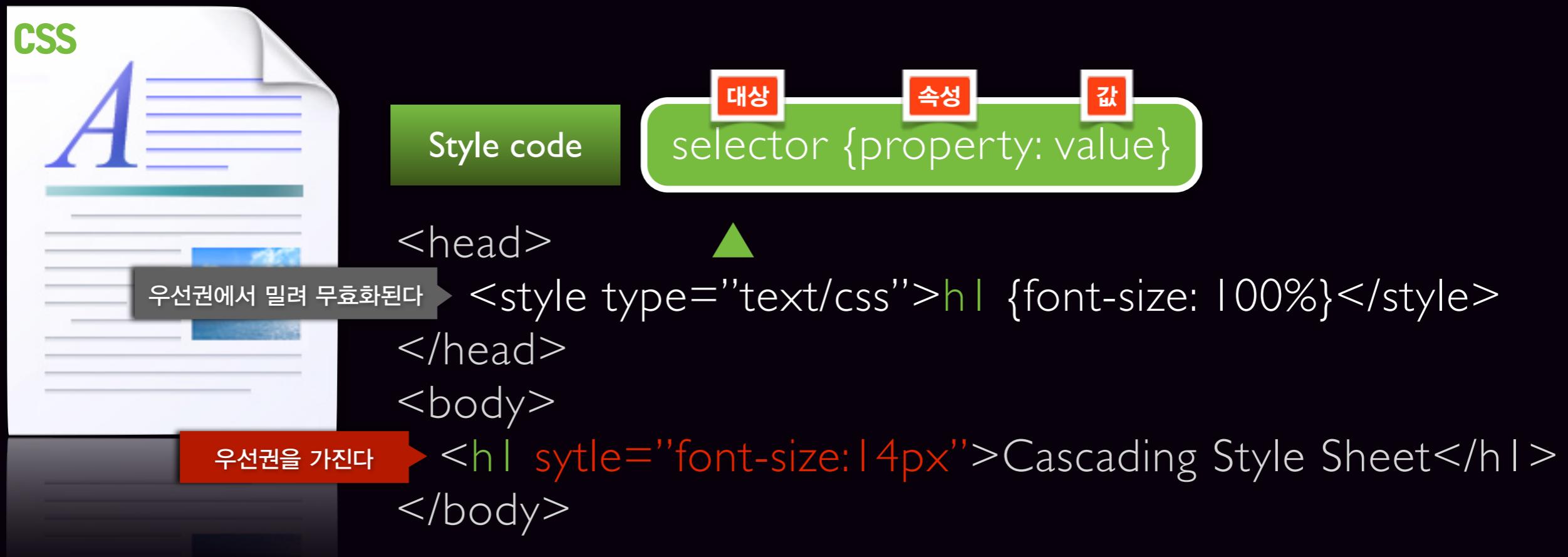


CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.

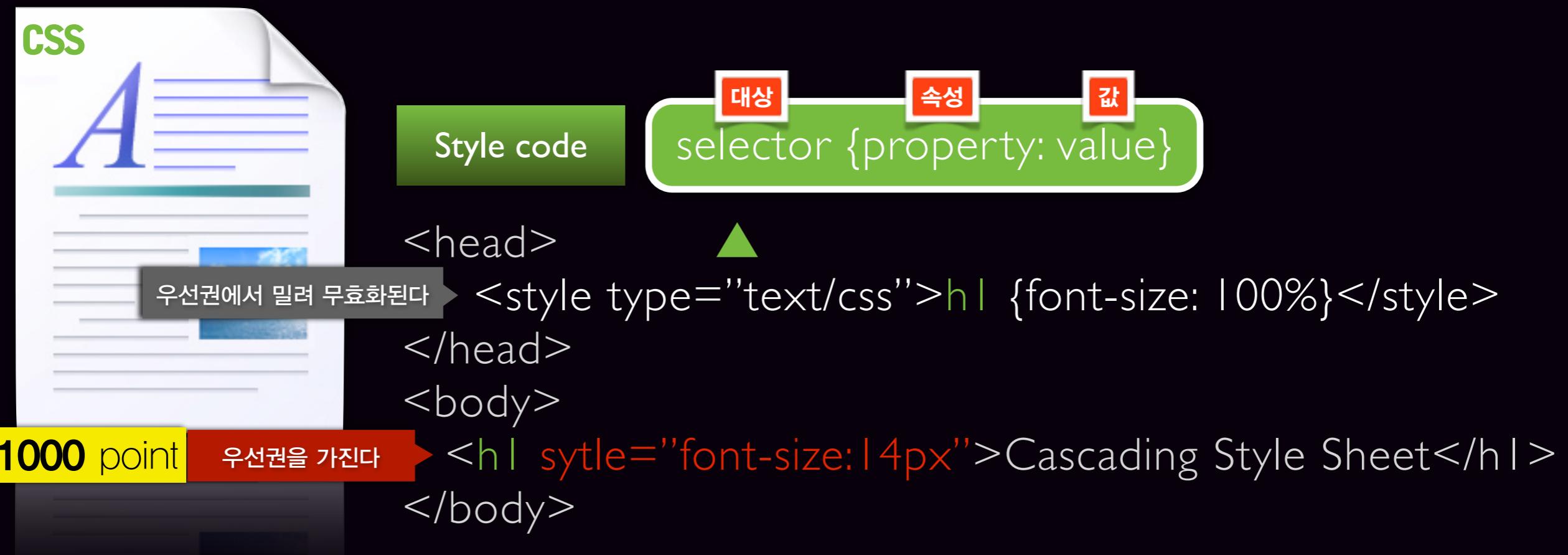


```
<head>           ▲  
    <style type="text/css">h1 {font-size: 100%}</style>  
</head>  
<body>  
    <h1 style="font-size: 14px">Cascading Style Sheet</h1>  
</body>
```

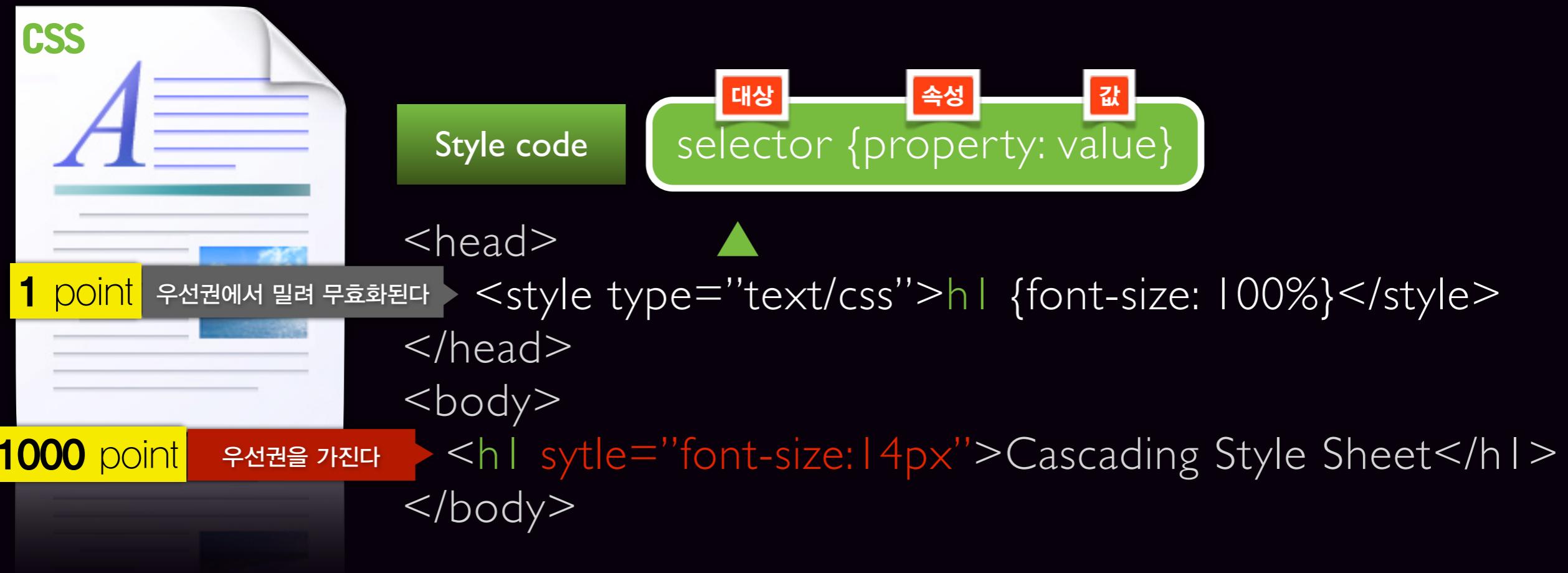
CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.

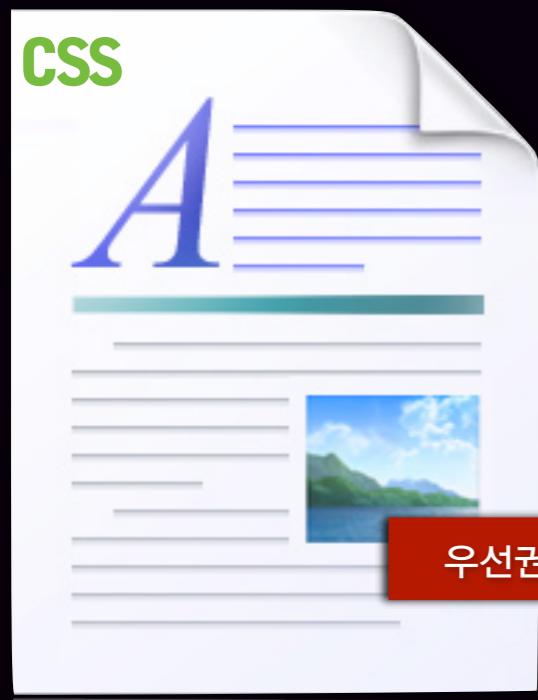


CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



```
<head>           ▲  
  <style type="text/css">  
    h1 {font-size: 100% !important}  
  </style>  
</head>  
<body>  
  <h1 style="font-size: 14px">Cascading Style Sheet</h1>  
</body>
```

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



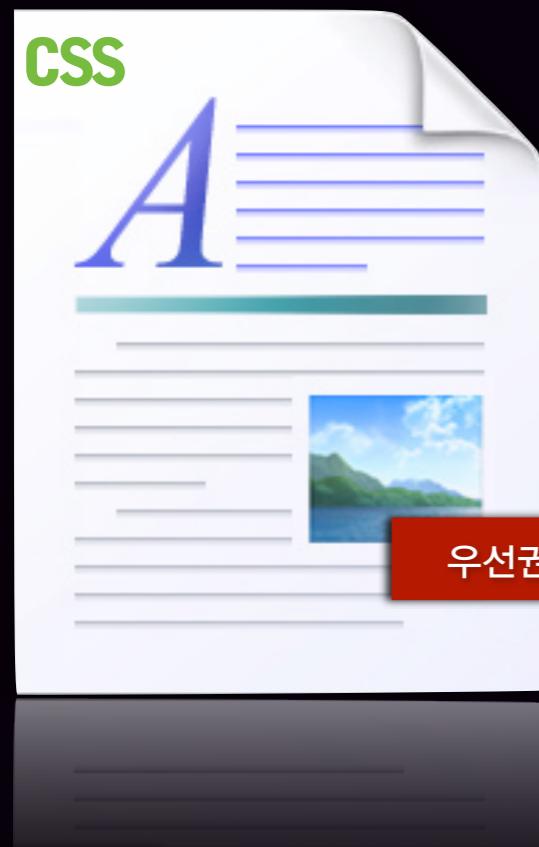
Style code

대상 속성 값
selector {property: value}

```
<head>           ▲  
  <style type="text/css">  
    h1 {font-size: 100% !important}  
  </style>  
</head>  
<body>  
  <h1 style="font-size: 14px">Cascading Style Sheet</h1>  
</body>
```

A diagram showing the structure of a CSS rule. It consists of three boxes: "대상" (target) with "selector", "속성" (property) with "{property: value}", and "값" (value). Below this, a code snippet shows how such a rule is implemented in HTML. A red arrow points from the "우선권을 가진다" text in the document icon to the "!important" declaration in the code. Another red arrow points from the "우선권에서 밀려 무효화된다" text in the document icon to the inline style in the body tag.

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



Style code

대상 속성 값
selector {property: value}

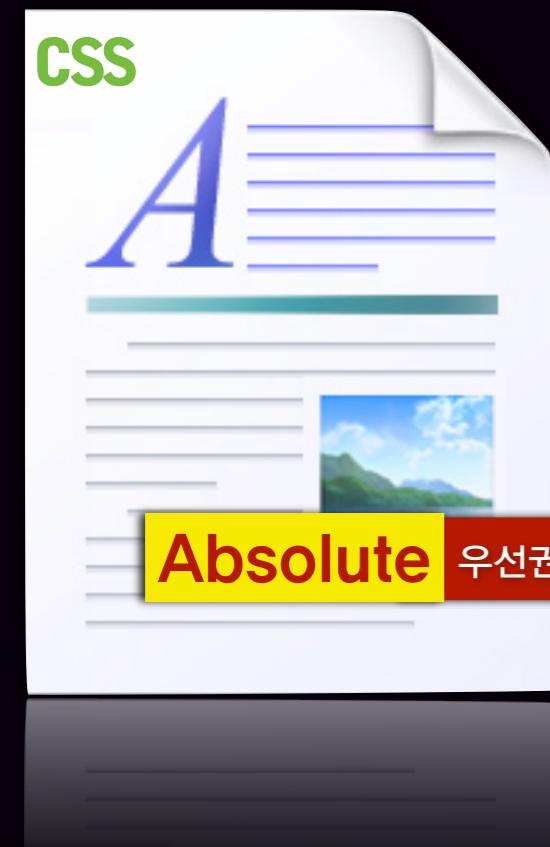
```
<head>           ▲  
  <style type="text/css">  
    h1 {font-size: 100% !important}  
  </style>  
</head>  
<body>  
  <h1 style="font-size: 14px">Cascading Style Sheet</h1>  
</body>
```

우선권을 가진다

우선권에서 밀려 무효화된다

1000 point

CSS는
Cascade 원리가 적용되어 각 규칙마다
우선순위를 부여하여 방식으로 동작합니다.



Style code

대상 속성 값
selector {property: value}

```
<head>           ▲  
    <style type="text/css">  
        h1 {font-size: 100% !important}  
    </style>  
</head>  
<body>  
    <h1 style="font-size: 14px">Cascading Style Sheet</h1>  
</body>
```

Absolute 우선권을 가진다

1000 point 우선권에서 밀려 무효화된다

Cascading Rule

!important > inline style > ID > class > type

Cascading Rule



Absolute

1000

100

10

1

!important > inline style > ID > class > type



External Styling

현대의 웹 디자인에서 가장 중요한 것은
구조에서 표현(스타일)을 분리하는 것입니다.



html code

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

```
<head>           ▲  
<style type="text/css">  
<!--  
    CSS Code  
-->  
</style>  
</head>
```

CSS는 가급적이면 완벽히 분리된
형태로 외부에서 작성하는 것이 좋습니다.



html code

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

```
<head>           ▲  
<style type="text/css">  
<!--  
    CSS Code  
-->  
</style>  
</head>
```

CSS는 가급적이면 완벽히 분리된 형태로 외부에서 작성하는 것이 좋습니다.

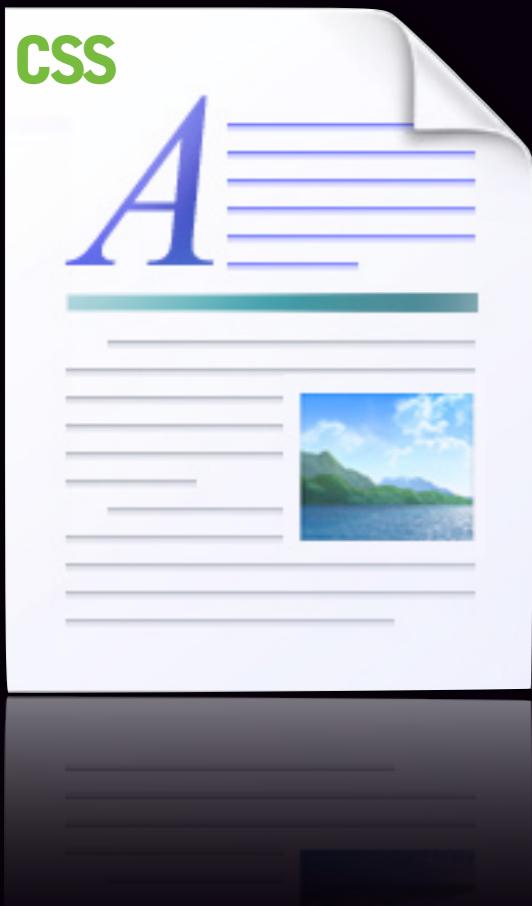


html code

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

```
<head>           ▲  
  <style type="text/css">  
    <!--  
      CSS Code  
    -->  
  </style>  
</head>
```

CSS 코드가 인식되지 않을 경우를 고려하여, HTML 주석 사용



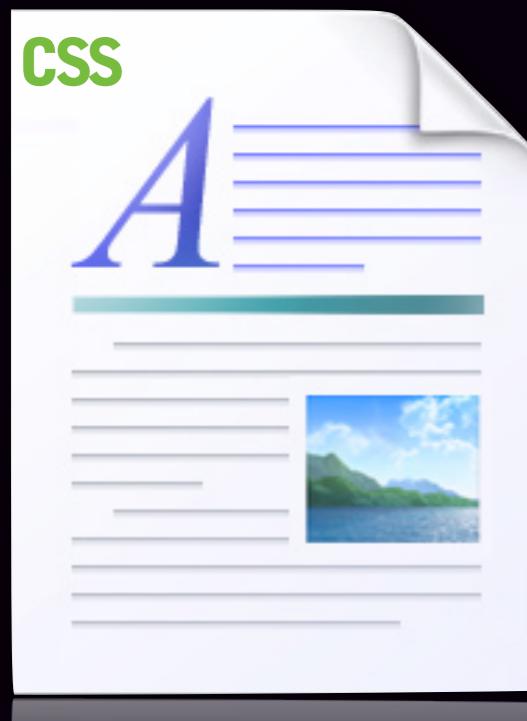
html code

<link href="" />

```
<head>
  <link href='style.css' type='text/css' rel='stylesheet' />
</head>
```



CSS는 가급적이면 완벽히 분리된
형태로 외부에서 작성하는 것이 좋습니다.



html code

<link href="" />

css 코드를 외부로 내보내는 방법!

```
<head>           ▲  
    <link href='style.css' type='text/css' rel='stylesheet' />  
</head>
```



Style code

@import “value”;

```
<head>           ▲  
    <style type="text/css">  
        @import “css/layout.css”;  
        @import url(css/font.css);  
    </style>  
</head>
```

CSS는 가급적이면 완벽히 분리된
형태로 외부에서 작성하는 것이 좋습니다.



Style code

@import "value";

css 코드를 외부로 내보내는 방법2

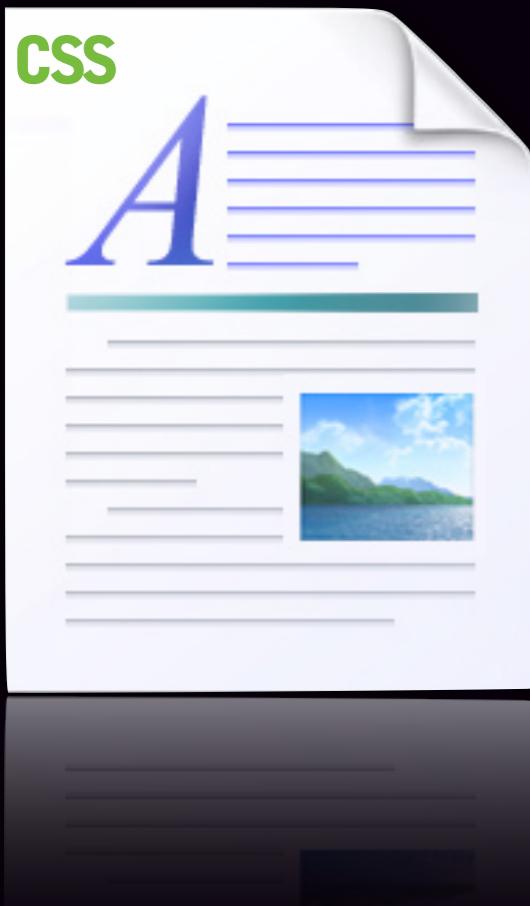
```
<head>           ▲  
    <style type="text/css">  
        @import "css/layout.css";  
        @import url(css/font.css);  
    </style>  
</head>
```



<link href=" " />

VS

@import " ";



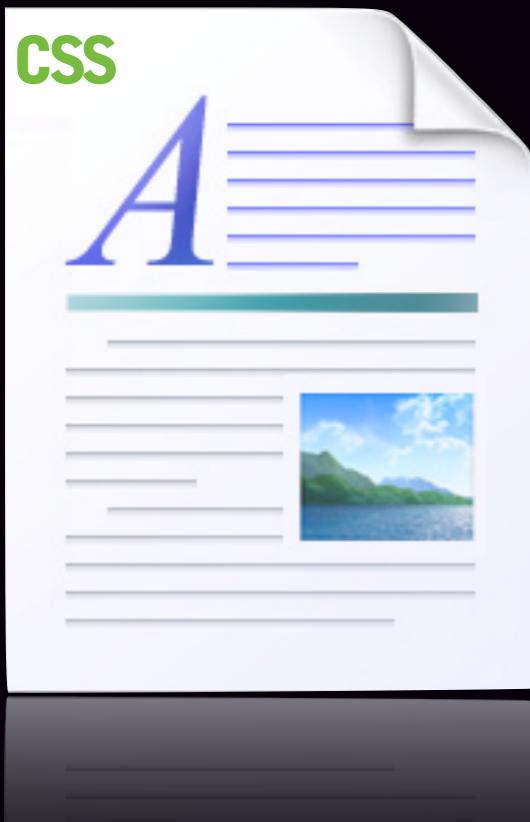
<link href=" " />

VS

@import " ";

HTML 내부에서
외부 CSS를 불러 올 경우,
권장합니다.

CSS 내부에서
외부 CSS를 불러 올 경우,
권장합니다.



<link href=" " />

VS

@import " ";

HTML 내부에서
외부 CSS를 불러 올 경우,
권장합니다.

CSS 내부에서
외부 CSS를 불러 올 경우,
권장합니다.

```
<head>
    <meta content="text/html; charset=utf-8" http-equiv="Content-Type" />
    <title>캠퍼스라이프</title>
    <link rel="stylesheet" type="text/css" href="css/style.css" media="scr
</head>
```

CSS는 가급적이면 완벽히 분리된 형태로 외부에서 작성하는 것이 좋습니다.



<link href=" " />

VS

@import " ";

HTML 내부에서
외부 CSS를 불러 올 경우,
권장합니다.

CSS 내부에서
외부 CSS를 불러 올 경우,
권장합니다.

```
<head>
    <meta content="text/html; charset=utf-8" http-equiv="Content-Type">
    <title>캠퍼스라이프</title>
    <link rel="stylesheet" type="text/css" href="css/style.css" >
</head>
```

```
style.css x default.css x color_font.css x layout.css x
1 @charset "utf-8";
2
3 /* 외부 제작자 스타일 문서 호출 */
4 @import "default.css";
5 @import "color_font.css";
6 @import "layout.css";
```



Style code

code comment

/* property: value */

```
<head>
  <style type="text/css">
    /* 문서 전체에 폰트 컬러, 사이즈 상속 */
    body { color: #535353; font-size: 75% }
    /* 제목 스타일 초기화 */
    h1, h2, h3, h4, h5, h6 { /* font-size: 100% */ }
    a img{border:none}
  </style>
</head>
```

CSS는 코드분량이 많아지면,
관리가 힘들어집니다. 코드주석을 달아
알아보기 쉽게 만드는 것이 좋습니다.



Style code

code comment

/* property: value */

```
<head>
  <style type="text/css">
    /* 문서 전체에 폰트 컬러, 사이즈 상속 */
    body { color: #535353; font-size: 75% }
    /* 제목 스타일 초기화 */
    h1, h2, h3, h4, h5, h6 { /* font-size: 100% */ }
    a img{border:none}
  </style>
</head>
```

CSS는 코드분량이 많아지면,
관리가 힘들어집니다. 코드주석을 달아
알아보기 쉽게 만드는 것이 좋습니다.

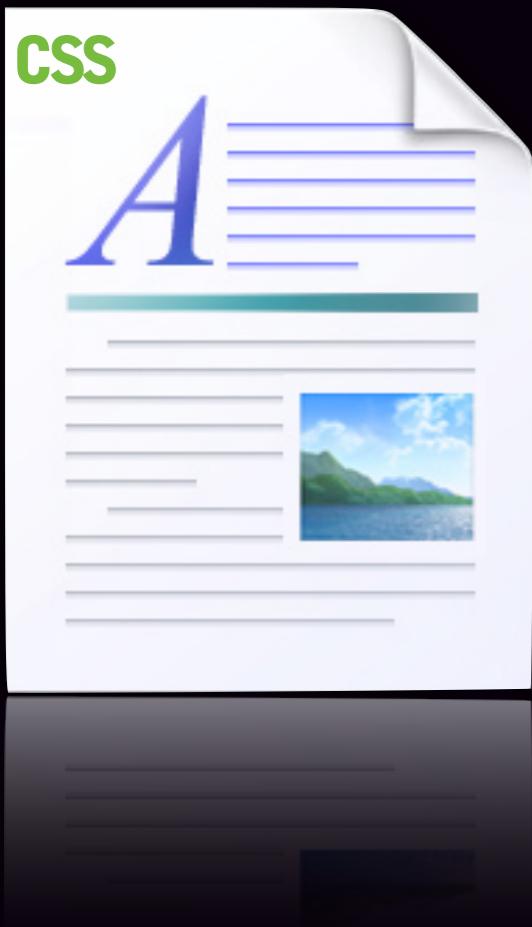


Style code

code comment

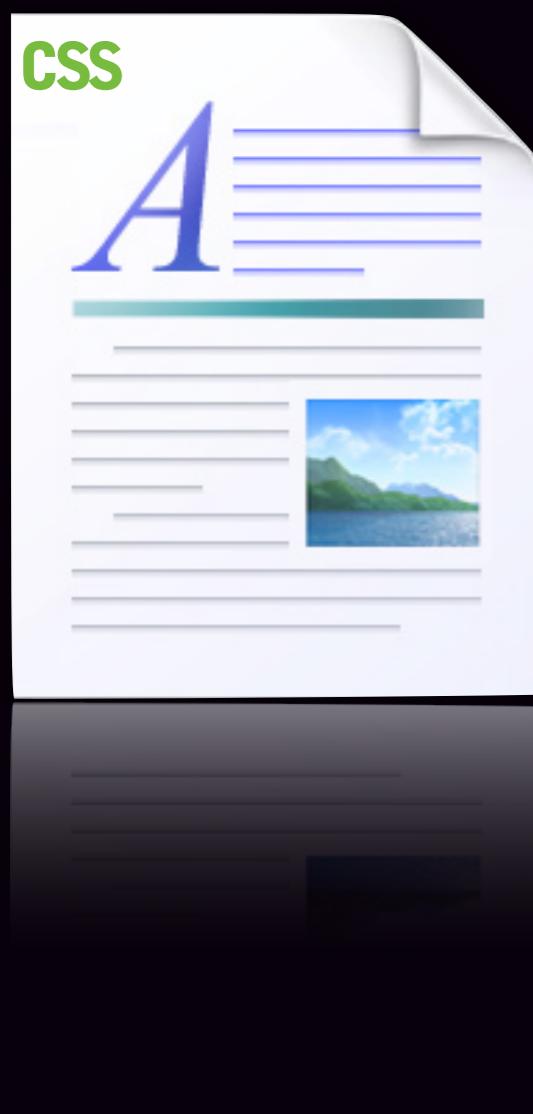
```
<head>
  <style type="text/css">
    /* 문서 전체에 폰트 컬러, 사이즈 설정 */
    body { color: #535353; font-size: 1em; }
    /* 제목 스타일 초기화 */
    h1, h2, h3, h4, h5, h6 { /* font-size: 1em; */
      a img{border:none}
    }
  </style>
</head>
```

```
36  textarea { overflow:auto; }
37
38  /* For Screen Reader */
39  legend
40  {
41    position:absolute;
42    top:0;
43    left:0;
44    width:0;
45    height:0;
46    overflow:hidden;
47    visibility:hidden;
48    font-size:0;
49    line-height:0;
50  }
51
52  /* Layout Selector */
53  #wrap { }
54
55  #wrapb { }
```



Style code

CSS는 작성 시,
스타일 가이드라인 작성 규칙을
정하는 것이 좋습니다.



Style code

새 문서 작성시 CSS 스타일 선언 순서

1. 브라우저 스타일 초기화 [Type Selector]
2. 전역 레이아웃(Global Section(id)) 스타일 선언 [Layout Selector]
3. 전역 클래스(Global Class) 스타일 선언 [Class Selector]
4. 기타 : 전역에서 지역으로 스타일 선언 (아래 순서로)

Header

Footer

Navigation

Aside

Content



Style code

media type

/* property: value */

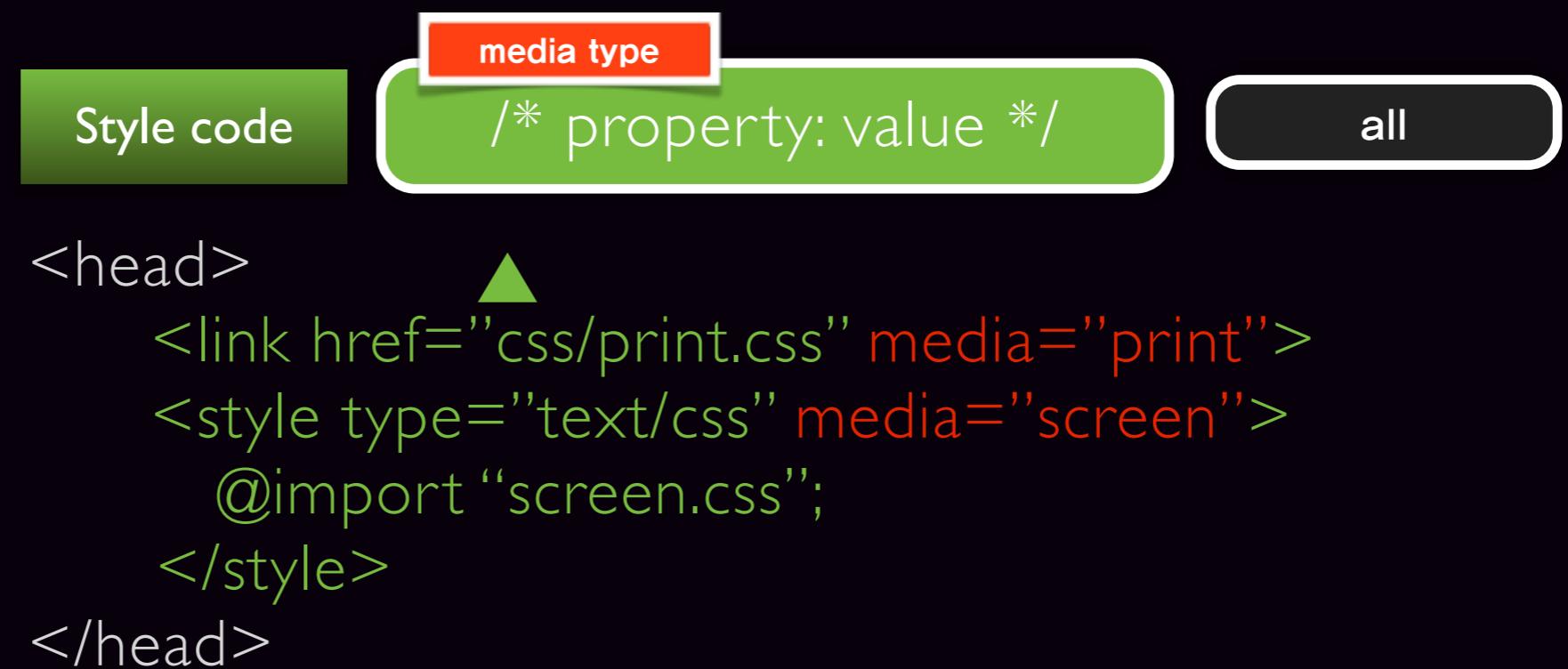
```
<head>
    <link href="css/print.css" media="print">
    <style type="text/css" media="screen">
        @import "screen.css";
    </style>
</head>
```

CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.

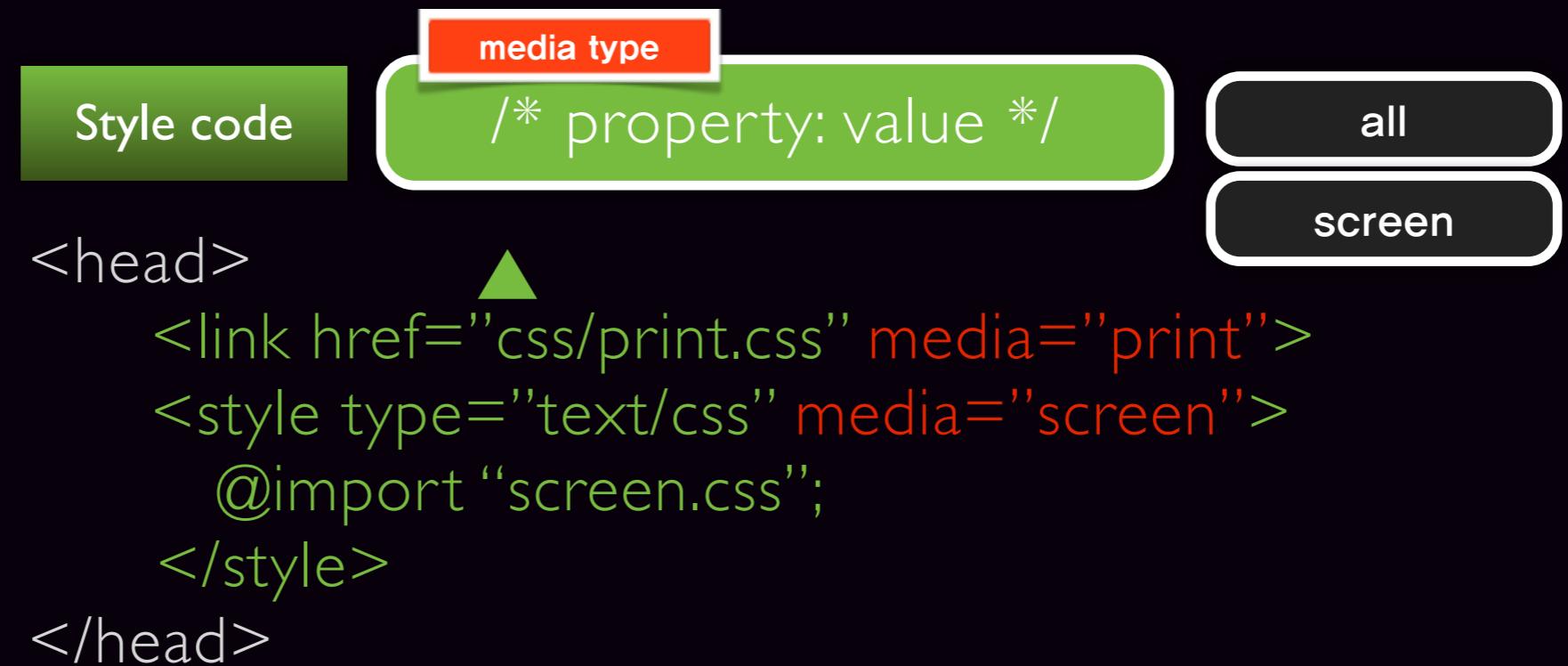


```
<head>
  <link href="css/print.css" media="print">
  <style type="text/css" media="screen">
    @import "screen.css";
  </style>
</head>
```

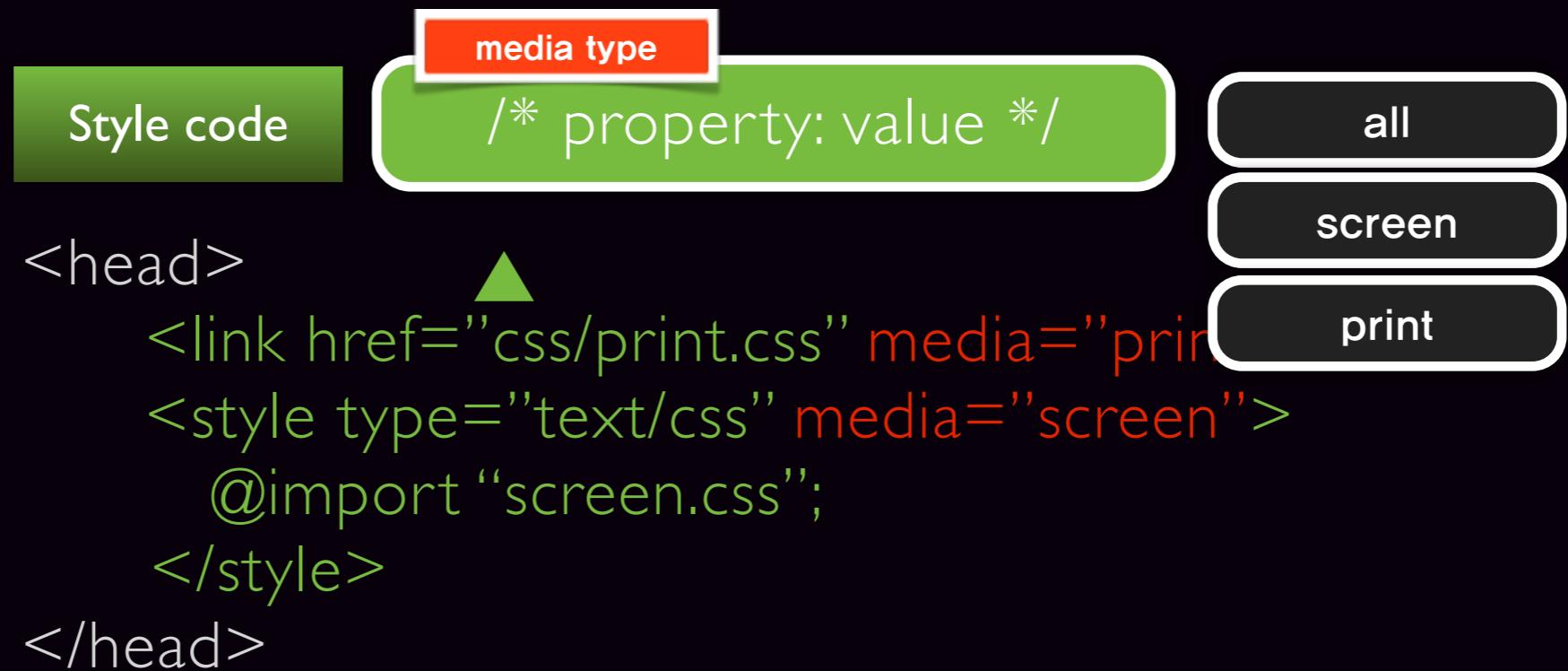
CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.



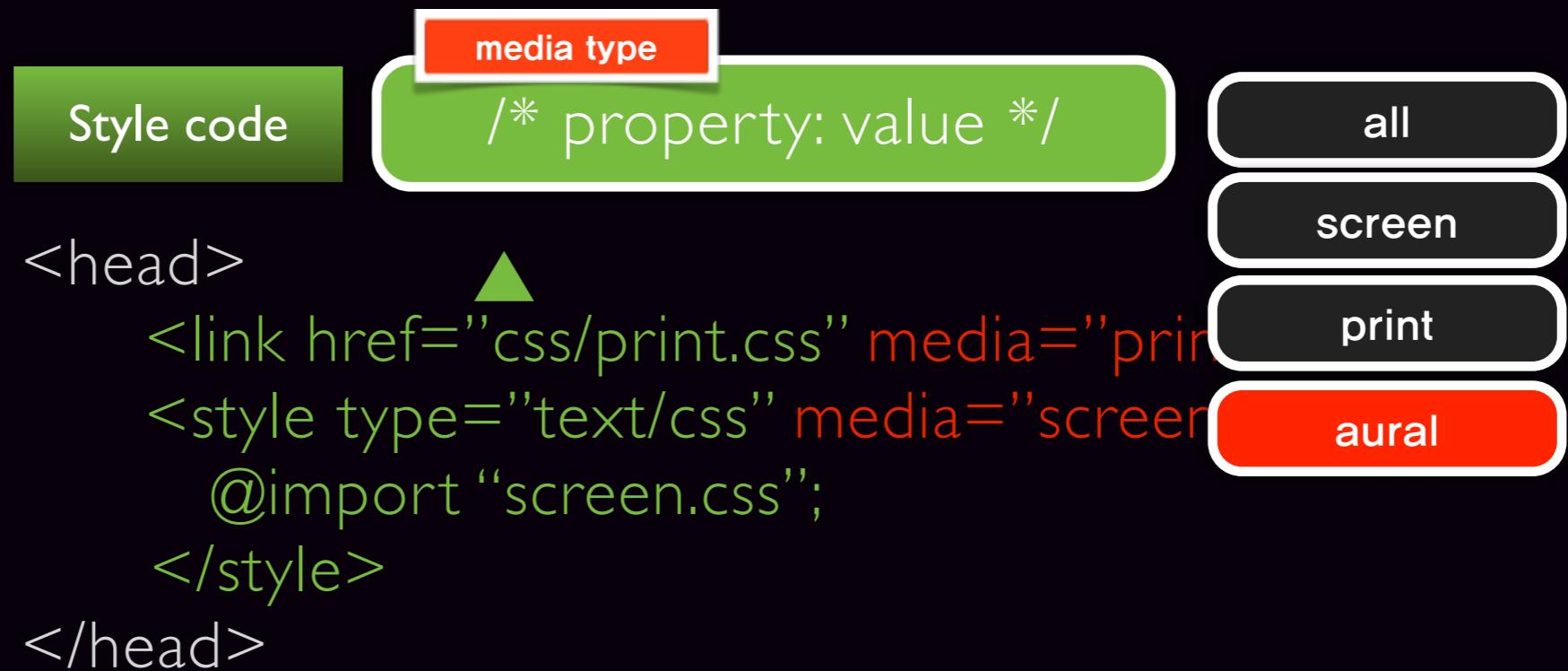
CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.



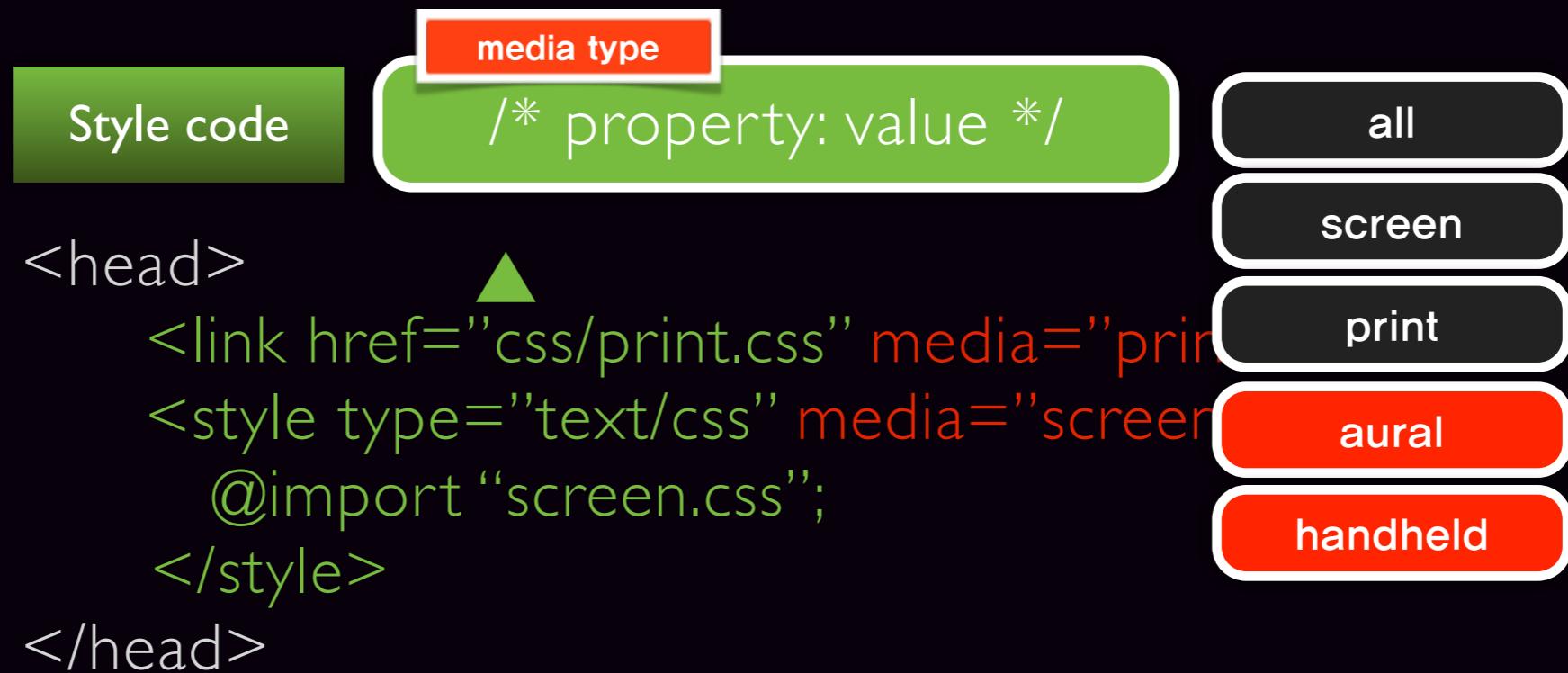
CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.



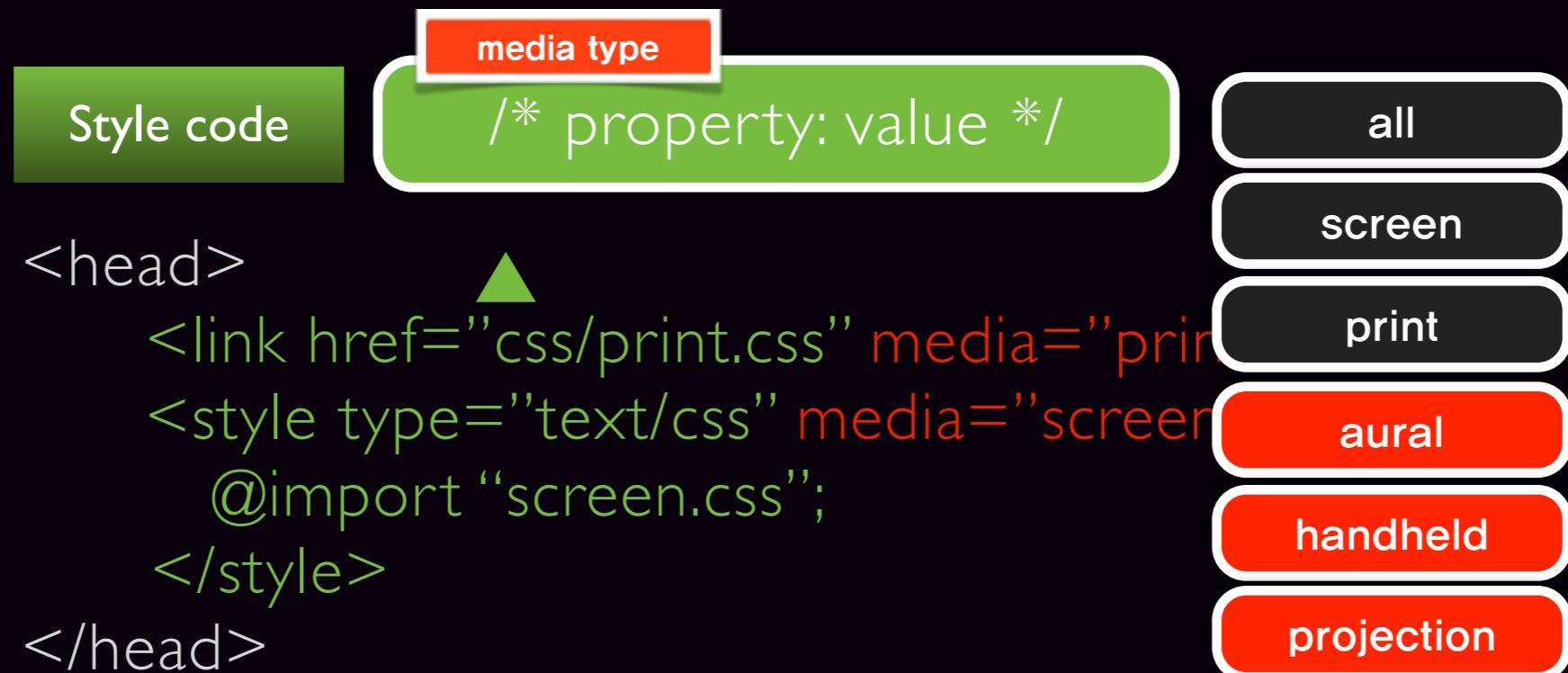
CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.



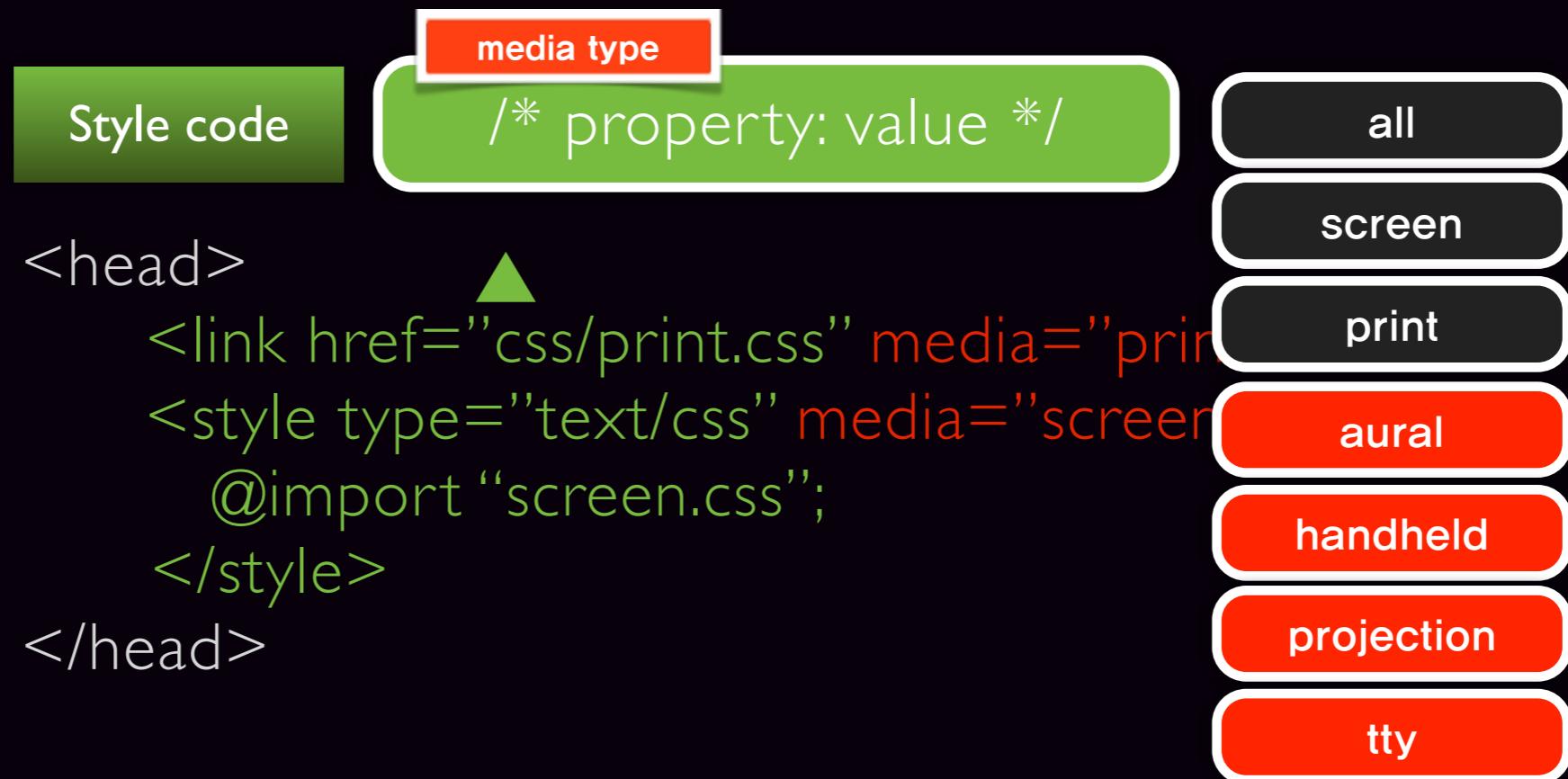
CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.



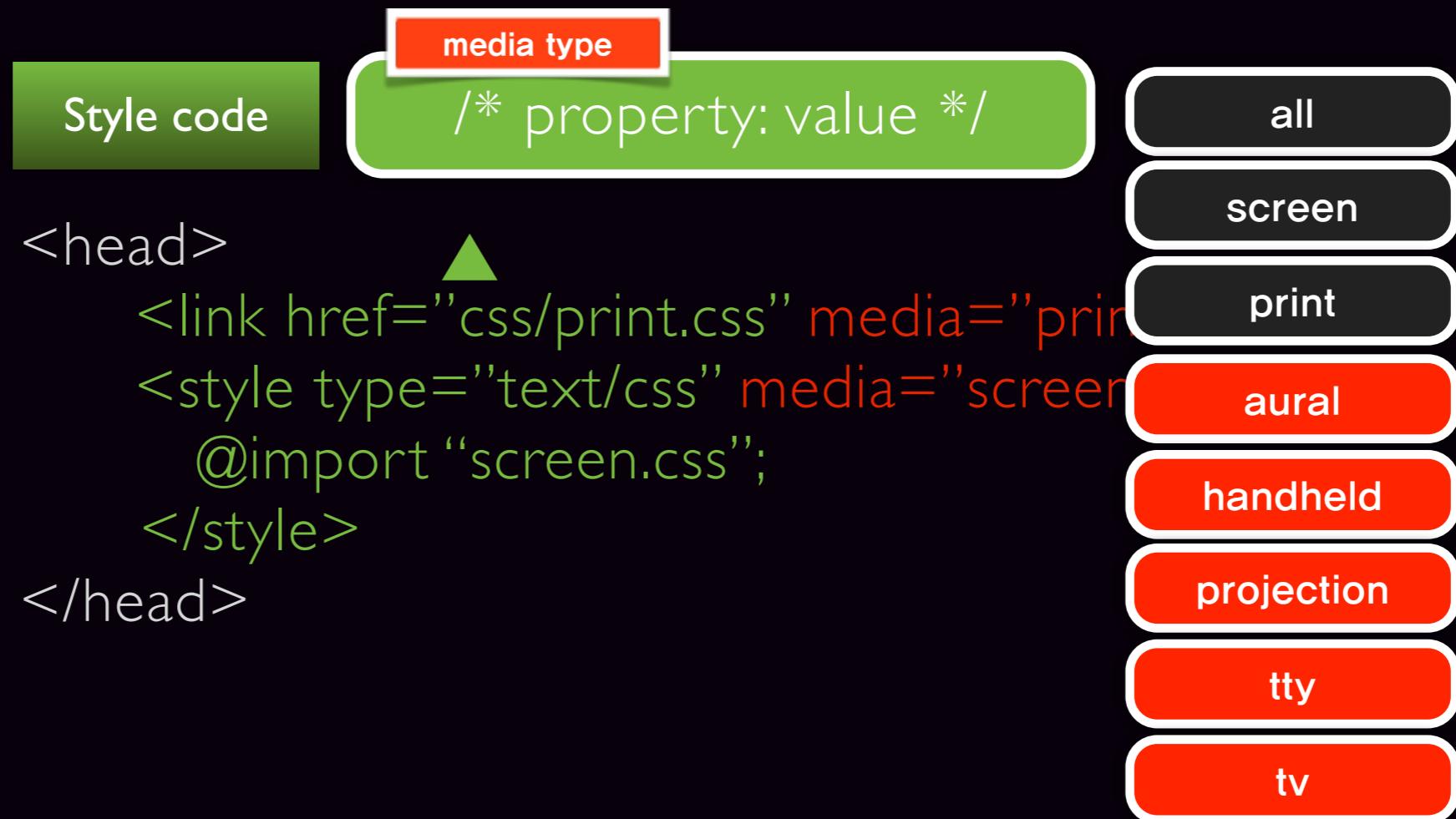
CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.



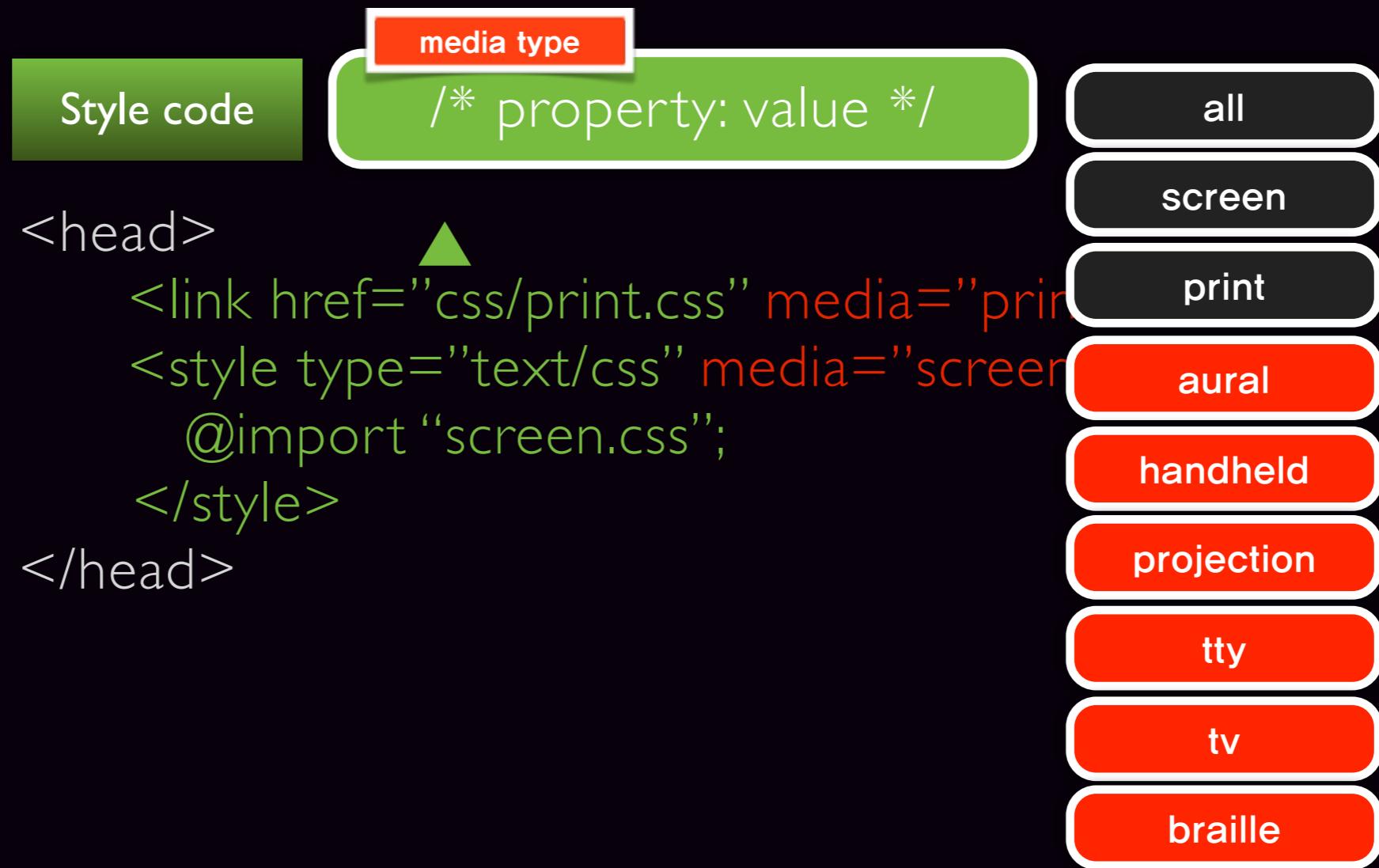
CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.

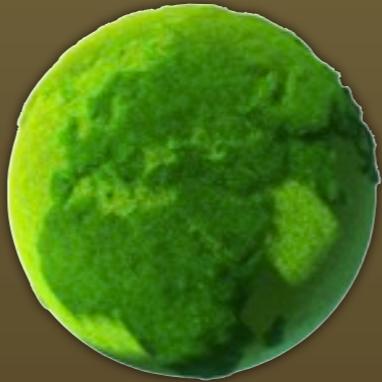


CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.



CSS는 각종 미디어 유형에 대처하는
별도의 CSS 코드를 작성할 수 있습니다.





Browser Style - reset

많은 브라우저는 각기 다른 브라우저의 기본 스타일(Default Style)을 가지고 있습니다.
CSS Design 시작에 있어 이는 다소 방해가 되기 때문에 브라우저 스타일 초기화
(Browser Style Reset)가 필요합니다.

Browser는
기본적으로 화면을 스타일하는 문서를
포함하고 있습니다.



Default style sheet for HTML 4

Appendix D. Default style sheet for HTML 4

This appendix is informative, not normative.

This style sheet describes the typical formatting of all HTML 4 ([\[HTML4\]](#)) elements based on extensive research of existing implementations. It is not a normative default style sheet in their implementations.

The full presentation of some HTML elements cannot be expressed in CSS 2.1, including [replaced](#) element, [img](#), [input](#), [area](#), [object](#), [embed](#), [img](#), [input](#), [area](#), [object](#), and frame elements.

For other elements, the legacy presentation can be described in CSS but the solution removes the element declarations to other elements (e.g., DIV). Likewise, legacy presentation of presentational attributes (e.g., the border attribute) is removed from the source document.

```
html, address,  
blockquote,  
body, dd, div,  
dl, dt, fieldset, form,  
frame, frameset,  
h1, h2, h3, h4,  
h5, h6, noframes,  
ol, p, ul, center,  
dir, hr, menu, pre { display: block; unicode-bidi: embed }  
li { display: list-item }  
head { display: none }  
table { display: table }  
tr { display: table-row }  
thead { display: table-header-group }  
tbody { display: table-row-group }  
tfoot { display: table-footer-group }  
col { display: table-column }  
colgroup { display: table-column-group }  
td, th { display: table-cell }  
caption { display: table-caption }  
th { font-weight: bolder; text-align: center }  
caption { text-align: center }  
body { margin: 8px }  
h1 { font-size: 2em; margin: .67em 0 }  
h2 { font-size: 1.5em; margin: .75em 0 }  
h3 { font-size: 1.17em; margin: .83em 0 }  
h4, p,  
blockquote, ul,  
fieldset, form,  
ol, dl, dir,  
menu { margin: 1.12em 0 }  
h5 { font-size: .83em; margin: 1.5em 0 }
```

<http://www.w3.org/TR/CSS21/sample.html>

CSS2.1 User Agent Style Sheet Defaults

CSS2.1 User Agent Style Sheet Defaults

Notes: [Values for IE8b](#) are based on information from a post-Beta1 build so some values might be different as the feature has since been updated.

- Rows with **Green** backgrounds showing differences between implementations.
- Rows with **Blue** backgrounds showing consistency between current implementations.
- Rows with **Light Grey** backgrounds showing deprecated elements.
- Rows with **Pink** shows a row has been added (not in Appendix D).
- Cells with **Red** shows values that are under questioning.
- Cells with **Orange** shows values that have been amended (corrected from values provided by Microsoft).
- [Mozilla default style sheet](#)
- [WebKit default style sheet](#)

CSS2.1 User Agent Style Sheet Defaults

| Element | Property | Appendix D | IE7 | IE8 | FF2 | FF3 | Opera |
|------------|--------------|----------------------------|--------------|---------------|---------------|---------------|---------------|
| Address | display | block | block | block | block | block | block |
| | font-style | italic | italic | italic | italic | italic | italic |
| B | font-weight | bolder | 700 | bolder | 401 | 401 | 700 |
| Big | font-size | 1.17em | 13.55pt | 1.17em | larger - 18px | 18px | 18px |
| Blockquote | display | block | block | block | block | block | block |
| | margin | 1.12em 0 | 14.25pt 30pt | 1em 40px | 1em 40px | 1em 40px | 1em 10px 1em |
| Body | margin-left | 40px | 30pt | 40px | 40px | 40px | 40px |
| | margin-right | 40px | 30pt | 40px | 40px | 40px | 10px |
| | display | block | block | block | block | block | block |
| | margin | 8px | 15px 10px | 8px | 8px | 8px | 8px |
| BR | br:before | content: "A" | undefined | content: "A" | null | content: "A" | content: "A" |
| Button | display | inline-block | inline | inline-block | inline | inline | inline-block |
| Caption | display | table-caption | block | table-caption | table-caption | table-caption | table-caption |
| | text-align | center | center | center | center | center | center |
| Center | display | block | block | block | block | block | block |

<http://css-class.com/test/css/defaults/UA-style-sheet-defaults.htm>

User Agent Style Sheets: Basics and Samples

- [Firefox 0.8](#)
- [Firefox 0.9](#)
- [Firefox 2.0.0.6](#)
- [Firefox 2.0.0.12](#)
- [Firefox 3.0b3](#)
- [Firefox 3.0.1](#)
- [Firefox 3.0.8](#)
- [Firefox 3.6.13](#)
- [Flock 0.9.0.2](#)
- [Flock 1.2.4](#)
- [Flock 1.2.7](#)
- [Internet Explorer 6](#)
- [Internet Explorer 7](#)
- [Internet Explorer 8](#)
- [Internet Explorer 9](#)
- [Konqueror 3.2.0](#)
- [Mozilla 1.0.1](#)
- [Mozilla 1.5](#)
- [Navigator 6.1](#)
- [Navigator 7.1](#)
- [Navigator 8.1](#)
- [Navigator 9.0b3](#)
- [Safari 3.0.4](#)
- [Safari x](#)

<http://meiert.com/en/blog/20070922/user-agent-style-sheets/>

Killer Collection of CSS Resets

The screenshot shows the Perishable Press website with a dark brown header. The header features a stylized sun icon and the text "Perishable Press" with "DIGITAL DESIGN & DIALOGUE" below it. The main content area has a dark background with white text. The title "Killer Collection of CSS Resets" is at the top, followed by the author "by JEFF STARR on TUESDAY, OCTOBER 23, 2007 – 51 RESPONSES". A sidebar on the right contains links like "Upgrade Your Feed Reader", "Welcome to Perishable Press" (with a photo of Jeff Starr), "Awesome Design Resources" (with links to "PHOTOGRAPHY WORDPRESS THEME" and "THEME SHIFT"), and an "Advertise Here" button.

Killer Collection of CSS Resets

by JEFF STARR on TUESDAY, OCTOBER 23, 2007 – 51 RESPONSES

UPDATE! CHECK OUT [CSSRESETR](#) FOR AN EASY WAY TO TEST AND DOWNLOAD THE BEST RESET styles for your next design.

 Using CSS to style semantically meaningful (X)HTML markup is an important key to modern web design practices. In a perfect world, every browser would interpret and apply all CSS rules in exactly the same way. However, in the imperfect world in which we live, quite the opposite frequently happens to be the case: many CSS styles are displayed differently in virtually every browser.

Many, if not all, major modern browsers (e.g., Firefox, Opera, Internet Explorer, Netscape, et al) implement their own generalized CSS rules, which often conflict with designer-applied styles. Other browsers fail to implement properly various CSS rules, making a mess of pages that happen to display perfectly in other browsers. Even worse, some browsers completely ignore specific aspects of CSS altogether, entirely ignoring widely accepted CSS attributes and properties.

Needless to say, the unpredictable manner in which browsers implement CSS requires us to seek realistic browser equalization strategies. Without relying on JavaScript to synthetically embellish default browser functionality, we focus our browser neutralization efforts entirely on CSS itself. By using an appropriately defined set of CSS "reset" rules, it is possible to override default browser styles and effectively neutralize behavior, allowing us to build our CSS on a uniform foundation.

Using a well-crafted set of global CSS reset styles enables designers to make assumptions about the default behavior of browsers. These presentational assumptions greatly simplify the process of

<http://perishablepress.com/press/2007/10/23/a-killer-collection-of-global-css-reset-styles/>



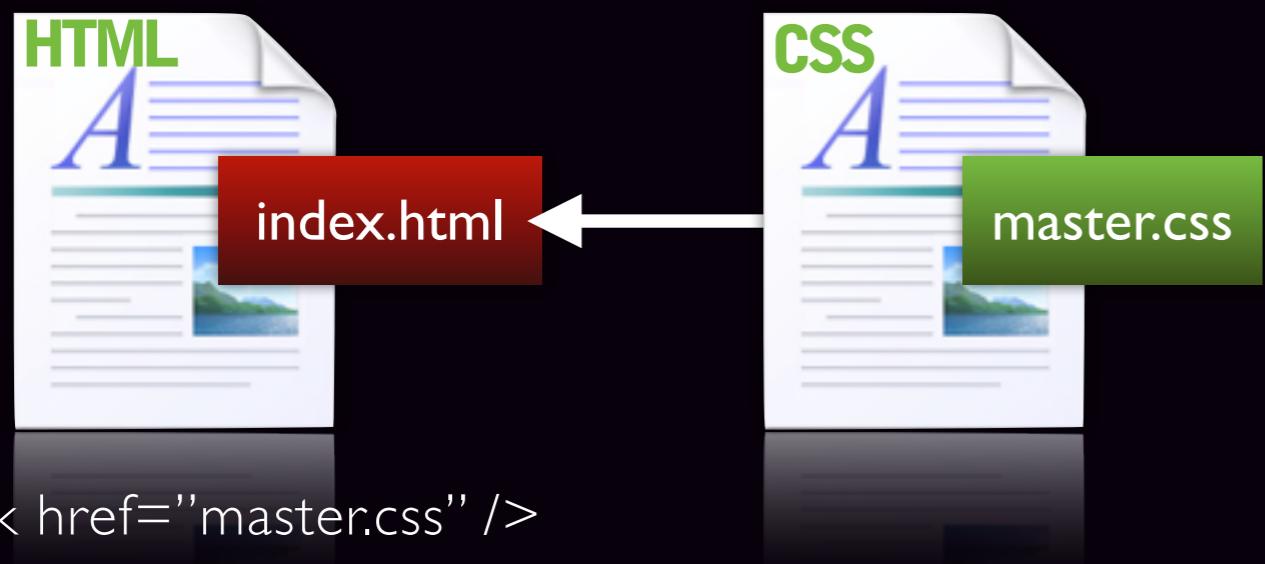
OOCSS

OOP CSS는 디자인 패턴을 분류하여
세부 스타일을 전문적으로 나눠 관리하는 시스템으로
고급 CSS 전문가의 기법입니다.

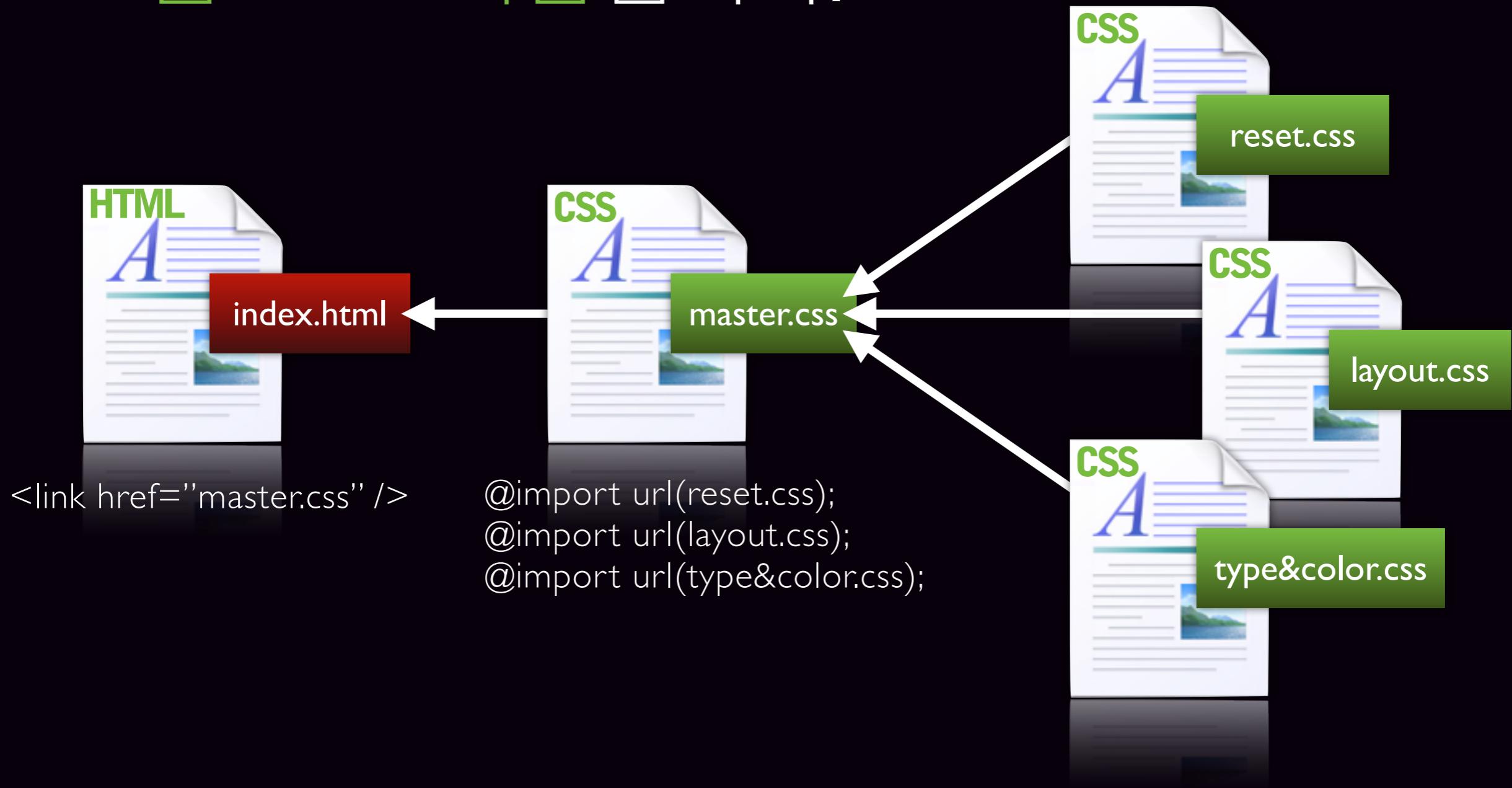
OOCSS는 스타일문서를
세부 역할로 쪼개어 관리하는 시스템으로
고급 CSS 기법입니다.



OOCSS는 스타일문서를
세부 역할로 쪼개어 관리하는 시스템으로
고급 CSS 기법입니다.



OOCSS는 스타일문서를
세부 역할로 쪼개어 관리하는 시스템으로
고급 CSS 기법입니다.




```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/
2 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko" lang="ko">
3     <head>
4         <meta content="text/html; charset=utf-8" http-equiv="Content-Type" />
5         <title>캠퍼스라이프</title>
6         <link rel="stylesheet" type="text/css" href="css/style.css" media="screen" />
7     </head>
8     <body>
9         <!-- Gallery 영역 -->
10        <div id="gallery">
11            <h3>젊은 시각</h3>
12            <a href="#">
13                
14                <p><a href="#">여대생 '생리 공결제' 논란<span class="newspaper">정성엽 명예 기자</span></a>
15                <ul>
16                    <li><a href="#">안병만 교육 &quot;국민 체감할 수 있...</a></li>
17                    <li><a href="#">동국대 등 5곳 입학사정관제 미...</a></li>
18                    <li><a href="#">부실 사립대 8곳 강도높은 구조</a></li>
19                    <li><a href="#">인천대·인천 전문대 통폐합 승인</a></li>
20                    <li><a href="#">초등 영어기초 학력미달 학생</a></li>
21                </ul>
22            </div>
23            <!-- // Gallery 영역 -->
24        </body>
25    </html>
```

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/1999/xhtml">
2 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko" lang="ko">
3     <head>
4         <meta content="text/html; charset=utf-8" http-equiv="Content-Type"/>
5         <title>캠퍼스라이프</title>
6         <link rel="stylesheet" type="text/css" href="style.css"/>
7     </head>
8     <body>
9         <!-- Gallery 영역 -->
10        <div id="gallery">
11            <h3>젊은 시각</h3>
12            <a href="#">
13                
14                <p><a href="#">여대생 '생리 공결제' 논란<span class="newspaper">정성엽 명예 기자</span></a>
15                <ul>
16                    <li><a href="#">안병만 교육 &quot;국민 체감할 수 있...</a></li>
17                    <li><a href="#">동국대 등 5곳 입학사정관제 미...</a></li>
18                    <li><a href="#">부실 사립대 8곳 강도높은 구조</a></li>
19                    <li><a href="#">인천대·인천 전문대 통폐합 승인</a></li>
20                    <li><a href="#">초등 영어기초 학력미달 학생</a></li>
21                </ul>
22            </div>
23            <!-- // Gallery 영역 -->
24        </body>
25    </html>
```

style.css x default.css x color_font.css x layout.css x

```
1 @charset "utf-8";
2
3 /* 외부 제작자 스타일 문서 호출 */
4 @import "default.css";
5 @import "color_font.css";
6 @import "layout.css";
```

The screenshot shows a software interface for managing CSS files and an HTML document. At the top, there are tabs for five files: style.css, default.css, color_font.css, and layout.css. The 'default.css' tab is currently selected. Below the tabs, the code for 'default.css' is displayed:

```
1 @charset "utf-8";
2
3 /* 공통 스타일 */
4 h1, h2, h3, h4, h5, h6, ul{padding:0; margin:0;}
5 h1, h2, h3, h4, h5, h6{font-size:100%;}
6 img{border:0;}
7 a{text-decoration:none;}
8 a:link, a:visited{color:#333;}
```

On the right side of the interface, the HTML document is visible. It includes a head section with links to the CSS files and a body section containing a list of links:

```
<head>
    <meta charset="utf-8" />
    <title>국립현대미술관</title>
    <link href="style.css" type="text/css" rel="stylesheet" media="screen" />
    <link href="color_font.css" type="text/css" rel="stylesheet" media="screen" />
    <link href="layout.css" type="text/css" rel="stylesheet" media="screen" />
</head>
<body>
    <ul>
        <li><a href="#">안병만 교육 &quot;국민 체감할 수 있...</a></li>
        <li><a href="#">동국대 등 5곳 입학사정관제 미...</a></li>
        <li><a href="#">부실 사립대 8곳 강도높은 구조</a></li>
        <li><a href="#">인천대·인천 전문대 통폐합 승인</a></li>
        <li><a href="#">초등 영어기초 학력미달 학생</a></li>
    </ul>
</div>
<!-- // Gallery 영역 --&gt;
&lt;/body&gt;
&lt;/html&gt;</pre>
```

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/1999/xhtml1/
2 html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko" lang="ko">
3   <head> @charset "utf-8";
4     <meta content="text/html; charset=utf-8" http-equiv="Content-Type" language="UTF-8" type="text/html; charset=utf-8"/>
5     <title>캠퍼스라이프</title>
6     <link rel="stylesheet" type="text/css" href="css/style.css" media="screen"/>
7   </head> h1, h2, h3, h4, h5, h6, ul{padding:0; margin:0;}
8   <body> h1, h2, h3, h4, h5, h6{font-size:100%;}
9   <!-- Gallery 영역 -->
10  <div style="border:0; position: relative; width: 100%; height: 100%;">
11    <a href="#" style="text-decoration: none; color: inherit; display: block; width: 100%; height: 100%;">
12      <img alt="Gallery 이미지" style="width: 100%; height: 100%;"/>
13      <p><a href="#">여기</a></p>
14      <ul>
15        <li><a href="#">여기</a></li>
16        <li><a href="#">여기</a></li>
17        <li><a href="#">여기</a></li>
18        <li><a href="#">여기</a></li>
19        <li><a href="#">여기</a></li>
20      </ul>
21    </div>
22    <!-- // Gallery 영역 -->
23  </body>
24 </html>
```

style.css x default.css x color_font.css x layout.css x

```
1 @charset "utf-8";
2 /* body 영역 */
3 body{font:11px/1.3 몰움, 굴림;}
4
5 /* gallery 영역 - 컬러, 폰트 */
6 #gallery{border:2px solid #eee;}
7 #gallery p span{color:#999;}
8 #gallery ul li a:hover{color:#0c7ce0;}
```

OOP CSS는 스타일문서를
세부 역할로 쪼개어 관리하는 시스템으로
고급 CSS 기법입니다.

The image shows a code editor with two tabs open: 'style.css' and 'layout.css'. The 'style.css' tab contains the following CSS code:

```
1 @charset "utf-8";
2 /*gallery 영역 - 레이아웃*/
3 #gallery{width:184px; padding:6px 16px 11px;}
4 #gallery h3{margin:7px 0;}
5 #gallery img{float:left; margin:0 10px 12px 0;}
6 #gallery p span{display:block; margin-top:4px;}
7 #gallery ul{clear:both; margin-left:10px;}
```

The 'layout.css' tab contains the following CSS code:

```
1 !DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/1999/xhtml"
2 
3 <head> @charset "utf-8";
4   <meta content="text/html; charset=utf-8" href="default.css"/>
5   <title>캠퍼스라이프</title>
6   <link rel="stylesheet" type="text/css" href="color_font.css"/>
7 </head> h1, h2, h3, h4, h5, h6, ul{padding:0; margin:0; border:0; font-size:1em; font-weight:bold; color:#000; font-family:'돋움', sans-serif; line-height:1.2; letter-spacing:0.05em; text-decoration:none; transition:color 0.2s ease-in-out; position: relative; z-index: 1; }
8 <body> h1, h2, h3, h4, h5, h6{font-size:1em; font-weight:bold; color:#000; font-family:'돋움', sans-serif; line-height:1.2; letter-spacing:0.05em; text-decoration:none; transition:color 0.2s ease-in-out; position: relative; z-index: 1; }
9 <!-- Gallery 영역 -->
10 <div border:0; border-collapse:collapse; width:100%; height:100%;>
11   <a href="#"> 선택한 사진선택한 사진선택한 사진
```


OOP CSS로 분리된 스타일문서는 성능을 고려했을 때 배포시 하나로 합쳐주는 것이 효율적입니다.

The screenshot shows the CodePlex project page for "CSS Mixer". The top navigation bar includes links for Register, Sign In, and CodePlex Home. Below the navigation is a search bar with the placeholder "Search all CodePlex projects" and a "Search" button. A logo of a truck with "CSS" on its side is visible. The main content area has a heading "CSS Mixer" and a navigation menu with links for Home, Downloads, Documentation, Discussions, Issue Tracker, Source Code, People, License, RSS, and Wiki & Documentation. Below the menu, there are links for View All Comments, Print View, Page Info, and Change History (all pages). The "Home" tab is selected. The "Project Description" section contains text about the tool's purpose: "CSS Mixer provide a GUI that allow you to combine many css files into one. And use the "Yahoo! UI Library: YUI Compressor for .Net" to minify it." It also mentions using it to improve ASP.NET themes. The "Download" section shows the current version is V1.0, released on Mon Feb 15 2010 at 7:00 AM, with a status of Stable and 941 downloads. There is a "Download" button with a green arrow icon. To the right of the download info is an advertisement for the book "Clean Code: A Handbook of Agile Software Craftsmanship" by Robert C. Martin, featuring quotes from Adam R. and Shunichi K. The advertisement includes a "Get it now" button and a link to the book's page. At the bottom of the page, there is a note about using the YUI Compressor project and a mention of the last edit date and author.

OOP CSS로 분리된 스타일문서는
성능을 고려했을 때 배포시 하나로 합쳐
주는 것이 효율적입니다.

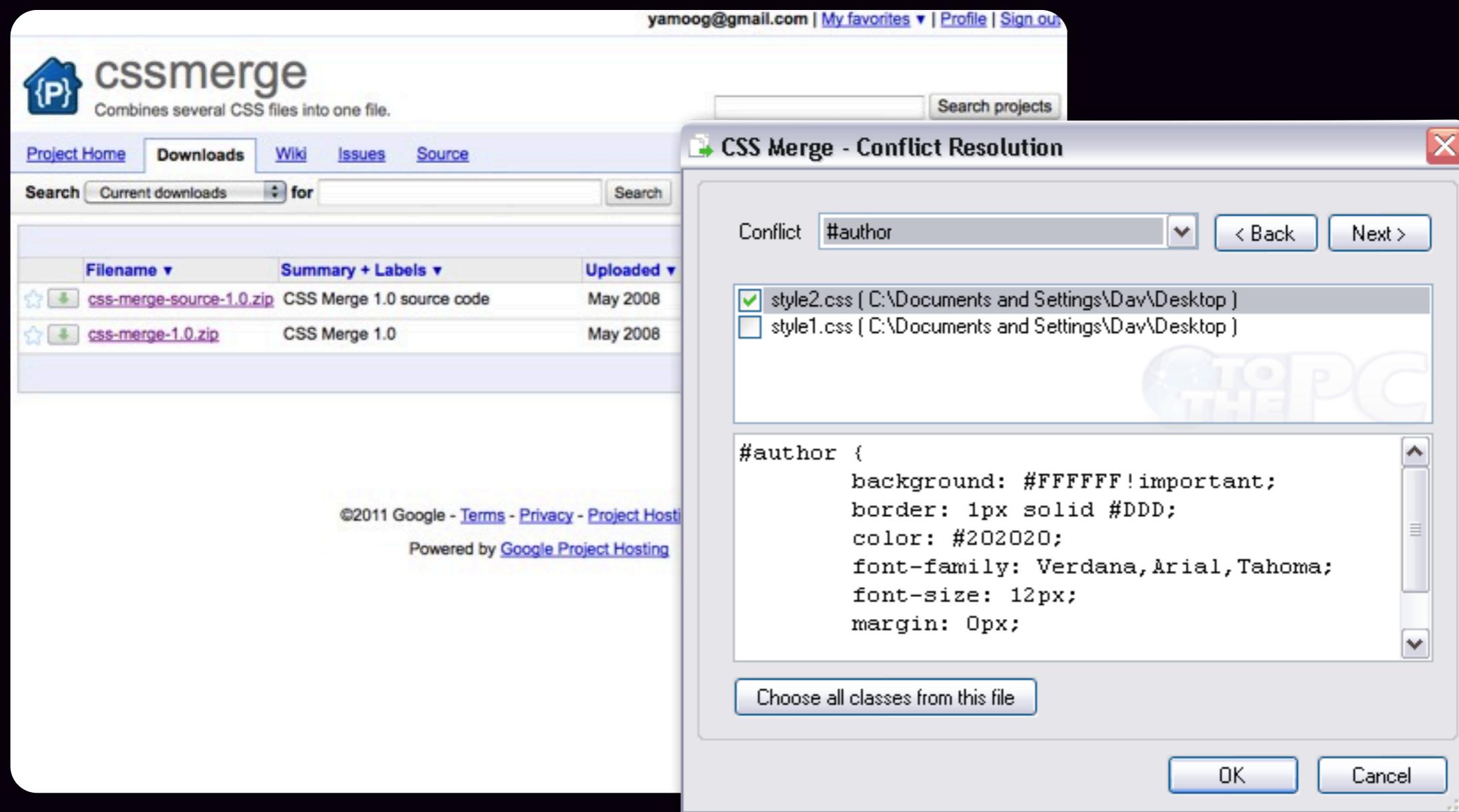
OOP CSS로 분리된 스타일문서는 성능을 고려했을 때 배포시 하나로 합쳐주는 것이 효율적입니다.

The screenshot shows the Google Project Hosting interface for the 'cssmerge' project. At the top, there's a navigation bar with links for 'yamoog@gmail.com', 'My favorites ▾', 'Profile', and 'Sign out'. Below the navigation is the project logo, 'cssmerge', which features a blue house icon with a white 'P' inside. A subtext below the logo reads 'Combines several CSS files into one file.' To the right of the logo is a search bar with the placeholder 'Search projects'. Below the header, there's a menu bar with tabs: 'Project Home' (which is selected and highlighted in blue), 'Downloads', 'Wiki', 'Issues', and 'Source'. Underneath the menu is a search bar with dropdown options 'Search' and 'Current downloads' followed by a 'for' field and a 'Search' button. The main content area is a table listing two files:

| Filename | Summary + Labels | Uploaded | ReleaseDate | Size | DownloadCount | ... |
|--|---------------------------|----------|-------------|--------|---------------|-----|
| css-merge-source-1.0.zip | CSS Merge 1.0 source code | | May 2008 | 123 KB | 710 | |
| css-merge-1.0.zip | CSS Merge 1.0 | | May 2008 | 229 KB | 2217 | |

At the bottom of the page, there are copyright notices: '©2011 Google - [Terms](#) - [Privacy](#) - [Project Hosting Help](#)' and 'Powered by [Google Project Hosting](#)'.

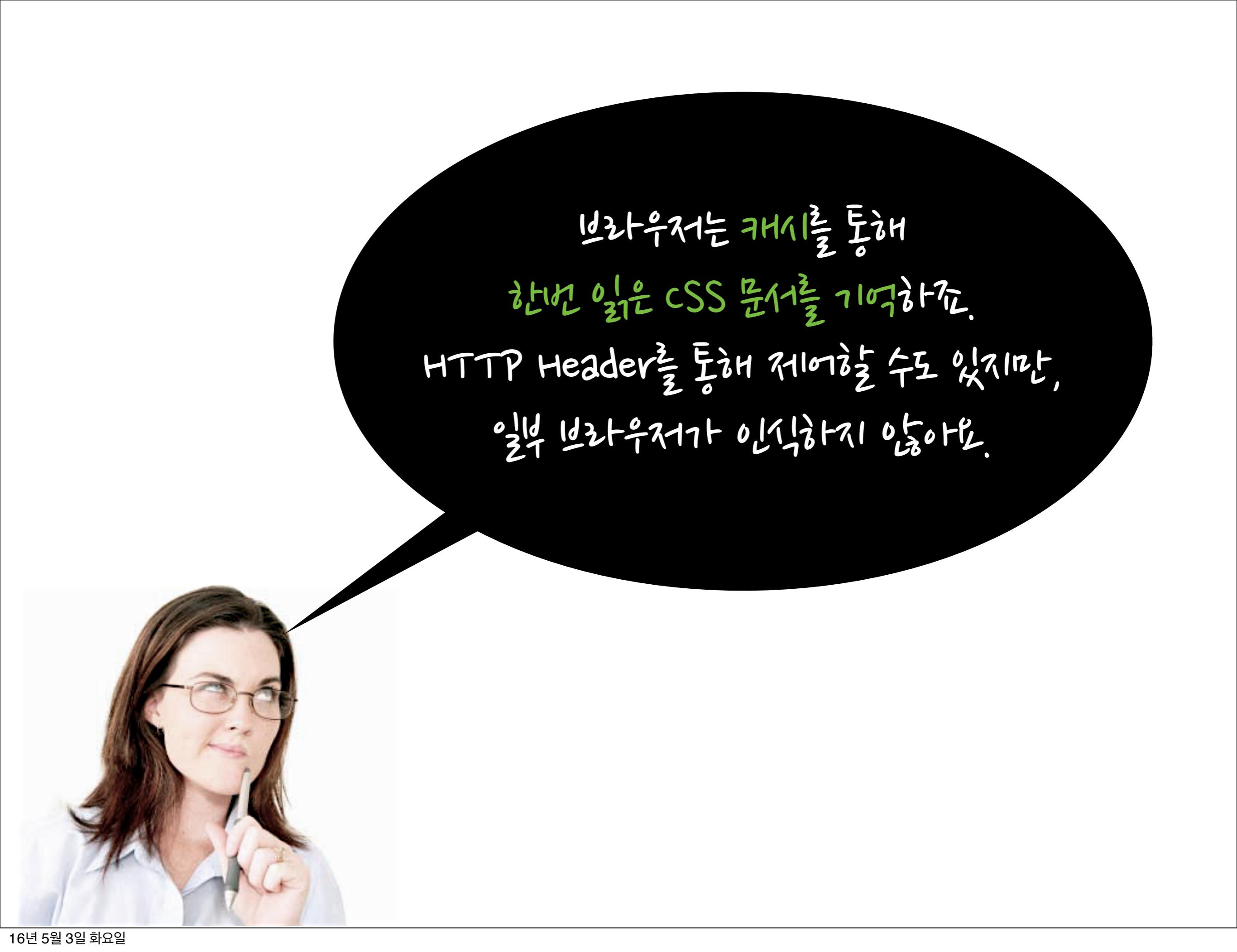
OOP CSS로 분리된 스타일문서는 성능을 고려했을 때 배포시 하나로 합쳐주는 것이 효율적입니다.



Browser cache & CSS

브라우저는 캐시(기억장치)를 통해
한번 읽은 적 있는 CSS 문서를 기억하고 있습니다.
이로 인해 수정된 CSS가 반영되지 않을 수 있습니다.





브라우저는 캐시를 통해
한번 읽은 CSS 문서를 기억하죠.
HTTP Header를 통해 제어할 수도 있지만,
일부 브라우저가 인식하지 않아요.

브라우저 개별 적용하지 않게 하려면

HTML 문서에 inline 으로 CSS 를 작성해야 한다.

하지만 이는 구조와 분리가 이루어지지 않아

내로 끌려야 한다!

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
<meta http-equiv="Content-Style-Type" content="text/css"/>
<title>Daum - 생활이 더 편리해! </title>
<link rel="alternate" type="application/rss+xml" href="http://www.daum.net/rss.xml"/>
<link rel="search" type="application/opensearchdescription+xml" href="http://www.daum.net/opensearch.xml" title="다음" />
```

```
<style type="text/css">
body,div,dl,dt,dd,ul,ol,li,h1,h2,h3,h4,form,fieldset,p,button{margin:0;padding:0;}
body,h1,h2,h3,h4,th,td,input{color:#333;font-family:"돋움",dotum,sans-serif;font-size:12px;font-weight:normal;}
body{background:#fff url("http://i1.daumcdn.net/deco/top/2010v2/bg_top_v01.gif") repeat-x;direction:ltr;text-align:center;}
hr{display:none;}
img,fieldset{border:0;}
ul,ol,li{list-style:none;}
img,input,select,textarea{vertical-align:middle;}
a{color:#30323f;text-decoration:none;}
a:hover{color:#4559E9;text-decoration:underline;}
a.ir:hover{text-decoration:none;}
#wrap .daumhide{visibility:hidden;position:absolute;top:-9999em;left:-9999em;width:1px;height:1px;line-height:0;overflow:hidden;}
.ir span{display:block;position:relative;z-index:-1; height:11px; overflow:hidden; font-size:11px;}
.paging{width:29px;position: absolute; top:0; right:0;}
.paging button{position: absolute; top:0; display: block; height:15px; border:0; overflow: hidden; background: url("http://i1.daumcdn.net/deco/tc/paging_bg.gif") no-repeat; width: 100%; height: 100%;}
.paging .prev{width:15px; left:0;}
```

다른 URL의 CSS 파일을 처리하는 것처럼
브라우저를 속여서 개시를 새로하게 만들면 되죠!

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=javaScript" />
<meta http-equiv="Content-Style-Type" content="text/css" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<title>네이버 :: 나의 경쟁력, 네이버</title>
<link rel="stylesheet" type="text/css" href="/css/navermain.css?20110609"/>

<link rel="stylesheet" type="text/css" href="/css/comcast.css?201100524"/>

<!-- (if IE) <style type="text/css">.cast_list li{white-space: normal;}</style> -->
```



```
yahoo9_example.html
+y yahoo9_example.html +☰

1 <link rel="stylesheet" type="text/css" href="main.css?<?=time()?>">

<link rel="stylesheet" type="text/css" href="main.css?ver=20110608">

<link rel="stylesheet" type="text/css" href="main.css?ver=1.9.3">
```