

Understanding HTML/CSS

스타일링 기초

을수



eulsoo.com
eulsoo.jung@gmail.com



이 저작물은 크리에이티브 커먼즈 저작자표시-변경금지 4.0 국제 라이선스에 따라 이용할 수 있습니다.

1. 이름짓고 선택하기
2. 구조적 선택
3. 가상선택자
4. 상속
5. 브라우저측 CSS
6. 하향우선 법칙
7. 명시도
8. 우선순위 정리

이름짓고 선택하기

Naming & Selecting

다양한 선택을 하기 위해 이름이 필요합니다

HTML안의 모든 <div>가 선택되었습니다.
태그이름만 가지고는 원하대로 선택할 수 없습니다.

```
div {border:2px solid red;}
```

```
<div>
  <h2>네이버서비스</h2>
  <div>
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는 가장 손쉽고 간편한 방법은
    무엇이 있을까? 아마도 전력을 아껴 쓰는
    일일 것이다.</p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
    이 생산되는 과정을 보여주는 스티커,
    전력 소비로 인한 환경 오염을 노골적으로
    보여주는 스티커 등 다양하다.</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

소설에 관하여
역사적 인물들
패션에 대하여

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

두가지 방법으로 이름을 지어 부를수 있습니다.

아이디 선택 **#service {border:2px solid black;}**

클래스 선택 **.castbox {border:2px solid red;}**

```
<div id="service">
  <h2>네이버서비스</h2>
  <div class="serv_item">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는...중략</p>
  </div>
  <div class="serv_item">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다.
    전 과정이 생산되는 과정을...중략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경오염을 노골적으로 보여주는 스티커등 다양하다.

단 하나뿐인 이름을 사용하기 위해서 id속성을 사용합니다

```
#service {border:2px solid red;}
```

“#”을 이용해 id이름인것을 표시합니다.
id를 이용해서 붙여진 이름은 단 한번만
사용할 수 있습니다.

```
<div id="service">
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는 ... 중략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다.
    전 과정이 생산되는... 중략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

id와 다르게 class는 여러개의 요소에 한가지 이름을 주어 한번에 명령을 내릴때 사용합니다

물론 id처럼 한가지 요소에만도 사용할 수 있습니다

```
#service {border:2px solid red;}  
.castbox {background-color:lime;}
```

“.”을 이름에 붙여 castbox를 클래스이름으로 가진 모든 요소를 선택하여 한번에 여러요소에 표현을 선언합니다.

```
<div id="service">  
  <h2>네이버서비스</h2>  
  <div class="castbox">  
    <h3>네이버 캐스트</h3>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 ... 중략</p>  
  </div>  
  <div class="castbox">  
    <h3>오픈 캐스트</h3>  
    <p>스티커의 디자인이 참 독특하다. 전  
    과정이 생산되는 ... 중략</p>  
  </div>  
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

한가지 요소가 여러 class에 속할 수 도 있습니다.

```
.service {border:2px solid black;}  
.castbox {background-color:lime;}  
.red {color:red;}
```

용도가 다른 castbox와 red라는 class를 만
들고 두가지 용도가 모두 필요한 요소에는
두개의 이름을 모두 붙여줄 수 있습니다.

```
<div class="service">  
  <h2 class="red">네이버서비스</h2>  
  <div class="castbox red">  
    <h3>네이버 캐스트</h3>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 ... 종략</p>  
  </div>  
  <div class="castbox">  
    <h3>오픈 캐스트</h3>  
    <p>스티커의 디자인이 참 독특하다. 전  
    이 생산되는... 종략</p>  
  </div>  
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

이름을 지을때는 몇가지 규칙이 있습니다

<p id="3service">

숫자로 시작하면 안됩니다.

<p class="service3">

숫자가 뒤에 붙는것은 괜찮습니다.

<p class="Service">

XHTML은 대소문자를 구분합니다. 대소문자를 구분하지 않는 HTML4.01이나 HTML5를 사용한다고 해도 습관적으로 소문자만 사용하는 것을 원칙으로 하는 것이 좋습니다.

<p class="service_three">

“_”를 사용해서 의미를 구분지어 줄 수 있습니다.

<p class="service-three">

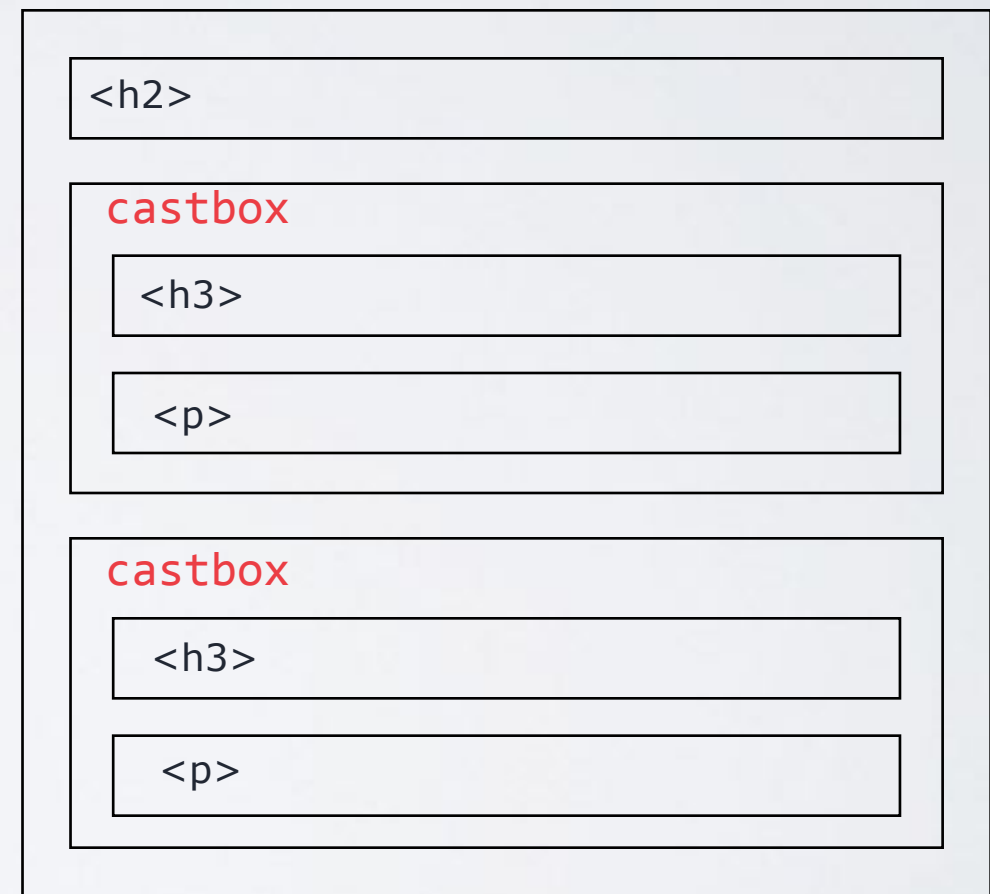
“-”를 사용해서 의미를 구분지어 줄 수 있습니다.

의미있는 이름을 부여할때 비로소 구조적으로 생각할 수 있습니다

“네이버 서비스안에 두개의 서비스가 있어” 라는 구조를 만들기 위해 castbox라는 이름을 부여하였습니다.
지금은 두개의 서비스가 있지만 계속해서 새로운 서비스가 추가 되더라도 문제없이 한번에 CSS로 컨트롤 할 수 있습니다.

```
<div class="service">
  <h2>네이버서비스</h2>
  <div class="serv_item">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는 가장 손쉽고 간편한 방법은
    무엇이 있을까? 아마도 전력을 아껴 쓰는
    일일 것이다.</p>
  </div>
  <div class="serv_item">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
    과정이 생산되는 과정을 보여주는 스티커,
    전력 소비로 인한 환경 오염을 노골적으로
    보여주는 스티커등 다양하다.</p>
  </div>
</div>
```

service



이렇듯 이름을 지을때는 구조와 연관성을 생각해야 합니다.

아래의 구조에서 cast_menu 부분에서 원하는 콘텐츠를 클릭하면 cast_body 부분이 바뀌면서 해당하는 콘텐츠를 보여주게 됩니다. 이렇듯 콘텐츠를 보여주는 방법에 따라 태그로 구조를 만들고 그에 걸맞는 이름을 구조와 연관성을 생각해서 지어야 합니다. 이런 과정을 통해서 우리는 좀 더 콘텐츠를 구조적으로 이해하는 생각의 훈련을 할 수 있습니다.

navercast



cast_menu

cast_game

game_hot

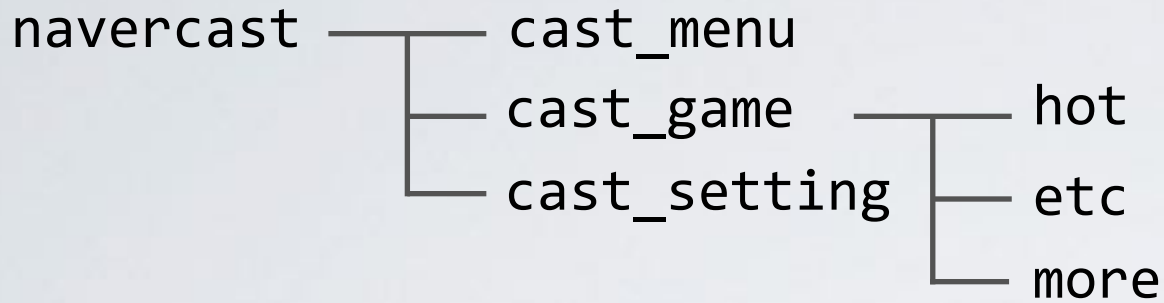
game_etc

game_recom

game_more

cast_setting

navercast라는 콘텐츠를 이루고 있는 세가지 박스이므로 모두 cast라고 붙이고 “_”뒤로는 용도에 따라 menu, body, setting이라고 이름을 주었습니다.



컨텐츠 성격에 맞는 hot, etc, more의 이름을 주어서 navercast(네이버캐스트)안에 cast_game(게임캐스트) 안에 more(더보기)라는 구조를 설계하게 되었습니다.



```

<div class="navercast">
  <div class="cast_menu">...</div>
  <div class="cast_game">
    <div class="hot">...</div>
    <div class="ect">...</div>
    <div class="more">...</div>
  </div>
  <div class="cast_setting">...</div>
</div>
  
```

콤마를 이용해 여러가지 요소를 한번에 선택할 수 있습니다.

h2, .castbox {border:2px solid red;}

“, ”콤마를 이용해서 여러가지 선택에 대해 한번에 보더를 주었습니다.

```
<div class="service">
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는 ... 중략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
    이 생산되는 ... 중략</p>
  </div>
</div>
```

네이버서비스


네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경오염을 노골적으로 보여주는 스티커등 다양하다.

한가지 요소에 지정된 아이디, 클래스, 태그이름을 역어서(체이닝) 선택할 수 도 있습니다.

 칸을 붙였습니다.
`div#service.service {border:2px solid lime;}` div라는 태그 이름, service라는 아이디, service 라는 클래스 모두 가지고 있는 요소를 선택함.

`h2.red {border:2px solid blue;}` h2라는 태그 이름에 클래스 red를 동시에 가지고 있는 요소를 선택함.

`.castbox.red {border:2px solid red;}`

클래스 castbox와 red를 동시에 가지고
있는 요소의 자손인 h3를 선택함.

```
<div id="service" class="service">
  <h2 class="red">네이버서비스</h2>
  <div class="castbox red">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는 ... 중략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
    이 생산되는 ... 중략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

속성을 이용해 선택할 수 있습니다.

```
input[value] {color:red;}
```


<input>중에 value속성이 있으면 선택합니다.



```
input[type="text"] {  
  background-color:yellow;  
}
```

.person의 자손인 <input>중에 type이라는 속성의 값을 text로 가지고 있는 요소를 선택합니다.

```
<div class="person">  
  <input type="text" id="email" />  
  <input type="submit" value="입력하기" />  
</div>
```




```
[class~="red"] {
  background-color:yellow;
}
```

service 자손중에서 class속성에 red이라는 단어로 값을 하나라도 가지고 있는 요소를 선택합니다.

```
div[class^="cast"] {
  border:2px solid blue;
}
```

service 자손중에서 class속성에 cast이라는 글자로 시작하고 있는 요소를 선택합니다.

```
div[class$="02"] {
  color:red;
}
```

service 자손중에서 class속성에 02이라는 글자로 끝나는 요소를 선택합니다.

```
[href*="naver.com"] {
  color:lime;
}
```

service 자손중에서 href속성에 naver.com이라는 글자를 가지고 있는 요소를 선택합니다

```
<div class="service" >
  <h2 class="top red">네이버서비스</h2>
  <div class="castbox01 red">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는 ... 중략</p>
  </div>
  <div class="castbox02">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
    과정이 생산되는 ... 중략</p>
    <a href="http://blog.naver.com">더보기</a>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

더보기

구조적 선택

Structural selecting

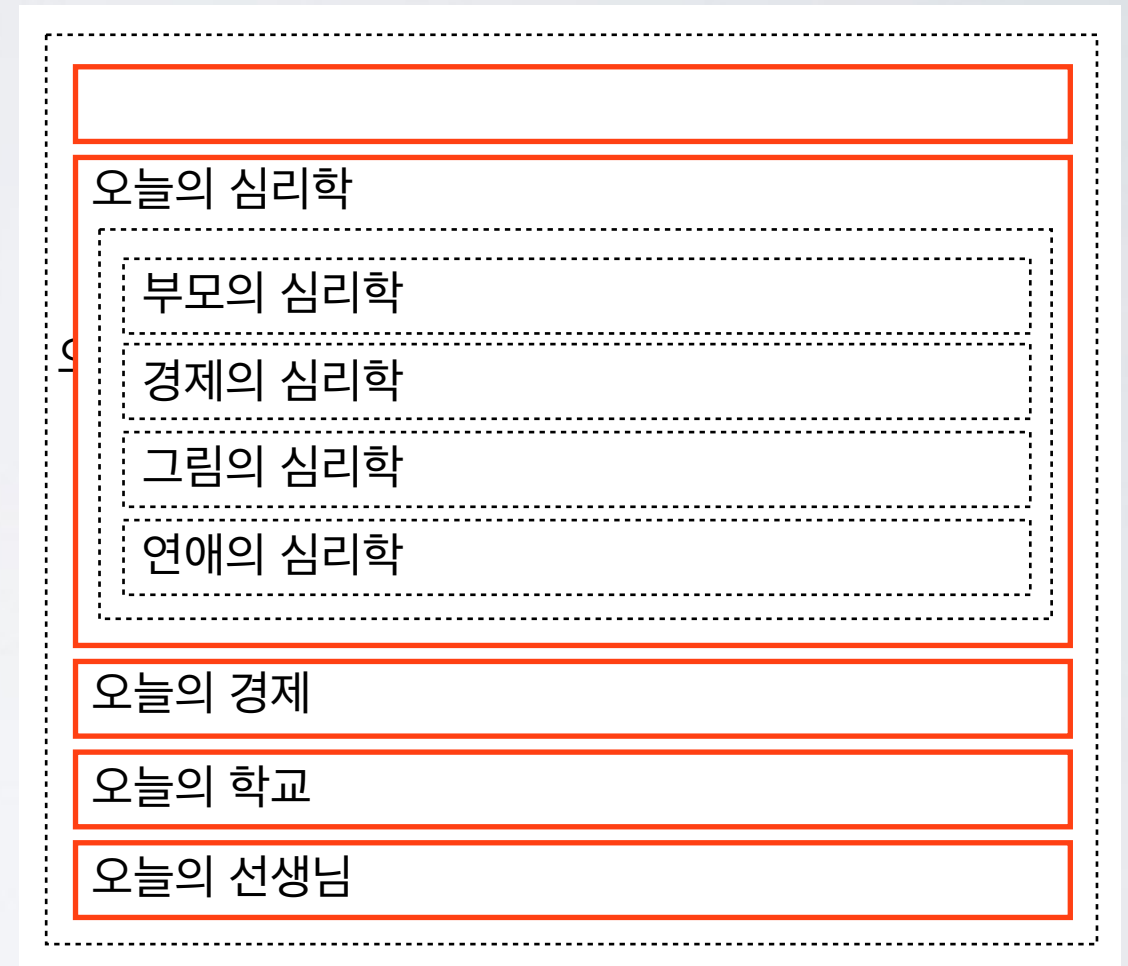
1. 직계자식을 선택합니다.

기호가 들어갈 때는 기호 양쪽으로 붙여도 되고 띄어도 됩니다.

`.category >li {border:2px solid red;}`

category의 직계 자식들만
선택하여 보더를 주었습니다.

```
<ul class="category">
  <li>오늘의 집</li>
  <li>오늘의 심리학
    <ul>
      <li>부모의 심리학</li>
      <li>경제의 심리학</li>
      <li>그림의 심리학</li>
      <li>연애의 심리학</li>
    </ul>
  </li>
  <li>오늘의 경제</li>
  <li>오늘의 학교</li>
  <li>오늘의 선생님</li>
</ul>
```



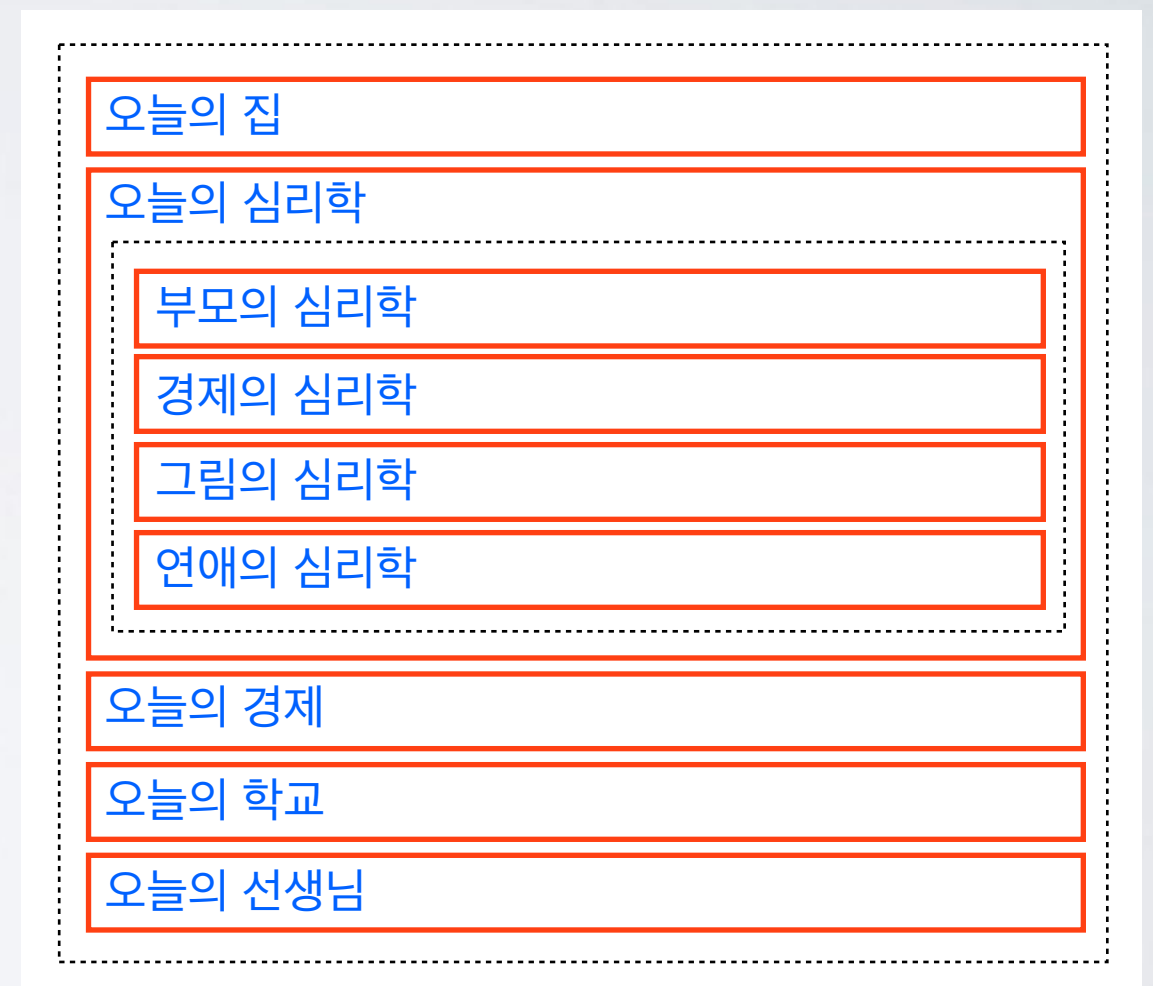
2. 자손을 선택합니다.

한 칸을 띄워서 자손을 표기합니다

```
.category li {border:2px solid red;}
```

category의 자손중에 모든 를 찾아
선택하였습니다.

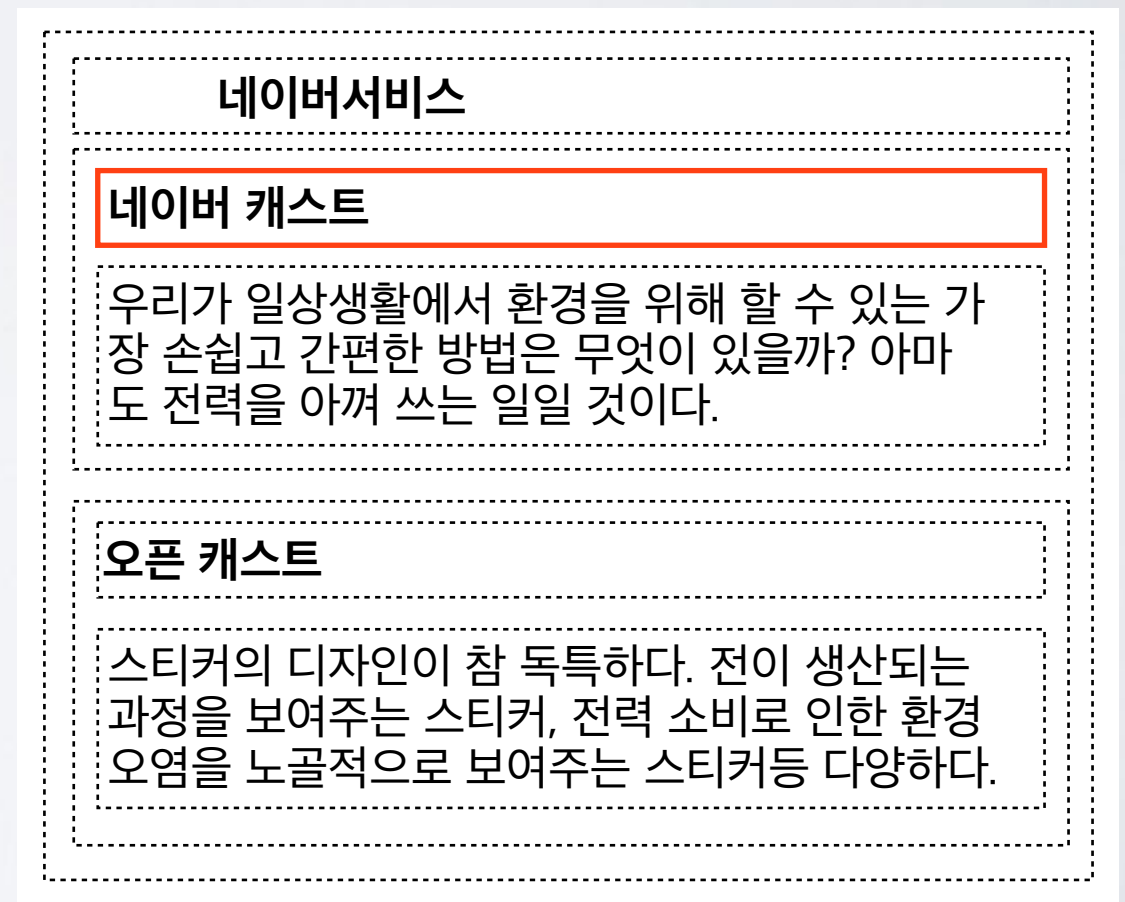
```
<ul class="category">  
  <li>오늘의 집</li>  
  <li>오늘의 심리학  
    <ul>  
      <li>부모의 심리학</li>  
      <li>경제의 심리학</li>  
      <li>그림의 심리학</li>  
      <li>연애의 심리학</li>  
    </ul>  
  </li>  
  <li>오늘의 경제</li>  
  <li>오늘의 학교</li>  
  <li>오늘의 선생님</li>  
</ul>
```



```
.castbox h3 {border:2px solid red;}
```

castbox의 자손중에 <h3>가 있으면 모두 선택하여서 결국 castbox안에 있는 <h3>만 선택하게 되었습니다. 이렇게 해서 <h3>를 선택하기 위해 class이름을 또 만들지 않고 castbox를 이용한 구조적인 선택이 되었습니다

```
<div class="service">
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는... 중략</p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
    이 생산되는 ... 중략</p>
  </div>
</div>
```



3. 바로뒤에 붙어있는 형제를 선택할 수 있습니다.

↙ + 기호를 사용합니다. 기호 양쪽으로 붙여쓸 수 있습니다.

```
h2 + .castbox {border:2px solid red;}  
.service > div + div {border:2px solid lime;}
```

service의 직계 자식들중에 <div>의 바로뒤에 붙어있는 <div>를 선택해서 보더를 선언하였습니다.

```
<div class="service" >  
  <h2>네이버서비스</h2>  
  <div class="castbox">  
    <h3>네이버 캐스트</h3>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 ... 중략</p>  
  </div>  
  <div>  
    <h3>오픈 캐스트</h3>  
    <p>스티커의 디자인이 참 독특하다. 전  
    이 생산되는 ... 중략</p>  
  </div>  
</div>
```

네이버서비스

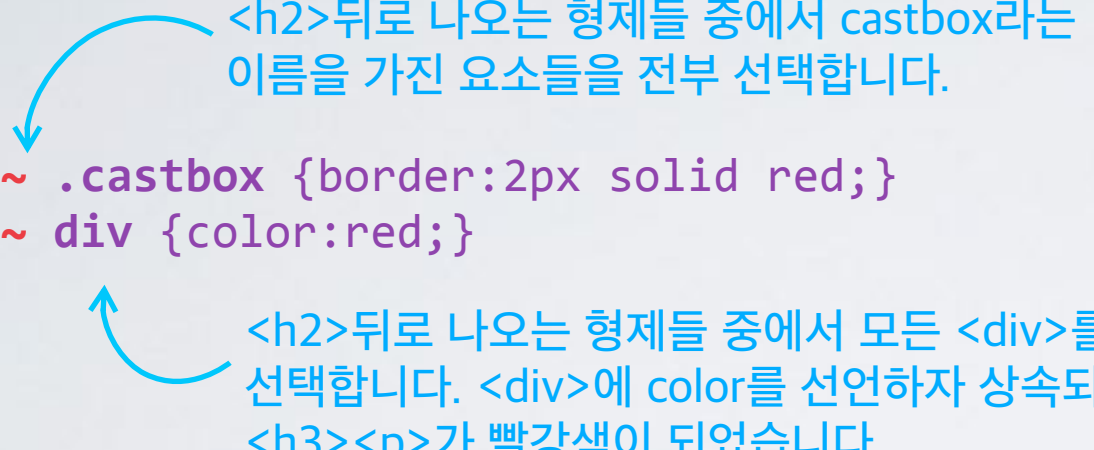
네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경오염을 노골적으로 보여주는 스티커등 다양하다.

4. 뒤로 나오는 원하는 모든 형제를 선택할 수 있습니다.



`h2 ~ .castbox {border:2px solid red;}`
`h2 ~ div {color:red;}`

```
<div class="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해
    할 수 있는 ... 중략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
    과정이 생산되는 ... 중략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경오염을 노골적으로 보여주는 스티커등 다양하다.

가상 선택자

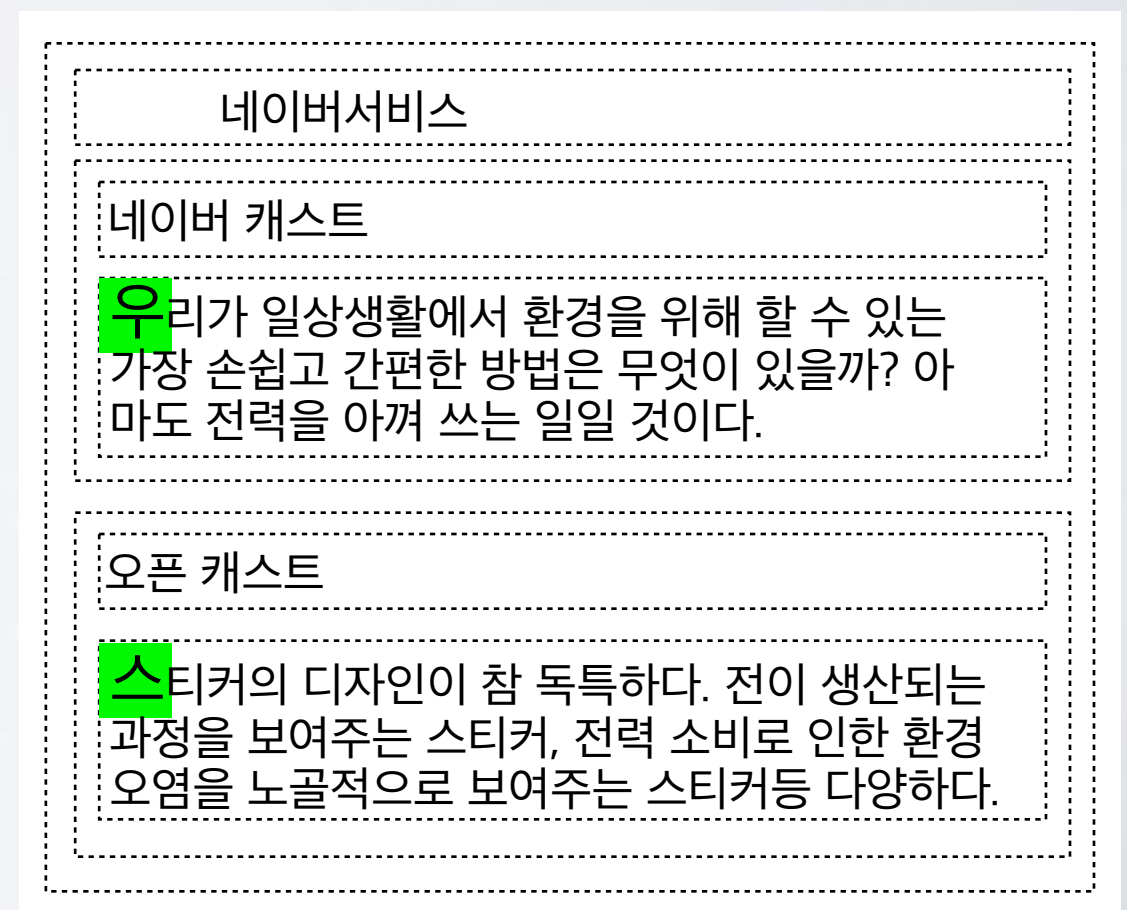
Pseudo-classes

1. 첫 글자만 선택합니다.

```
.castbox p:first-letter {  
  background-color:lime;  
  font-size:24px;  
}
```

castbox의 모든 자손중에서 <p>를 찾아
첫번째 글자를 마크업을 한 것처럼 가상의
클래스를 주어 선택합니다.

```
<div class="service" >  
  <h2>네이버서비스</h2>  
  <div class="castbox">  
    <h3>네이버 캐스트</h3>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 ... 중략</p>  
  </div>  
  <div class="castbox">  
    <h3>오픈 캐스트</h3>  
    <p>스티커의 디자인이 참 독특하다. 전  
    과정이 생산되는... 중략</p>  
  </div>  
</div>
```

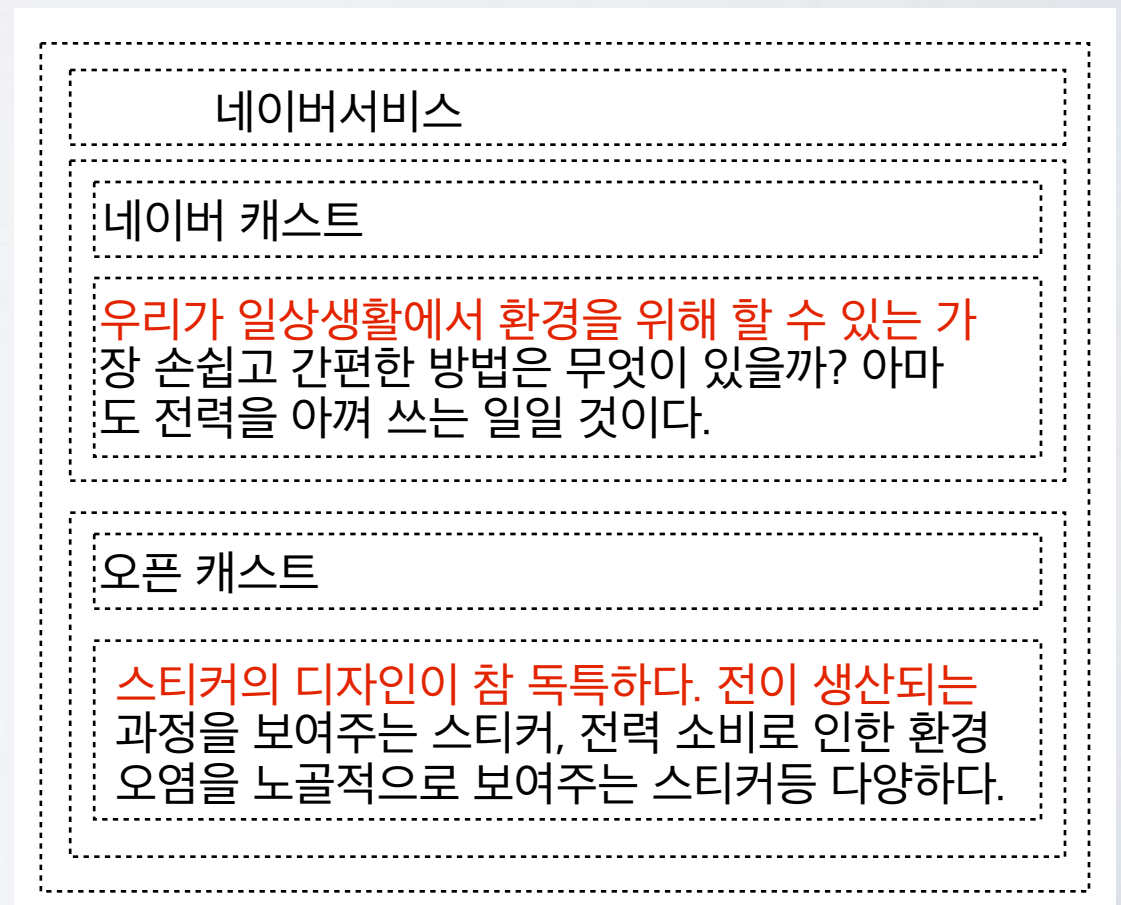


2. 첫 줄만 선택합니다.

```
.service p:first-line {  
  color:red  
}
```

service의 모든 자손중에서 <p>를 찾아
첫째줄을 마크업한것 처럼 가상선택자로
선택합니다.

```
<div class="service" >  
  <h2>네이버서비스</h2>  
  <div class="castbox">  
    <h3>네이버 캐스트</h3>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 가... 중략</p>  
  </div>  
  <div class="castbox">  
    <h3>오픈 캐스트</h3>  
    <p>스티커의 디자인이 참 독특하다. 전  
    과정이 생산되는... 중략</p>  
  </div>  
</div>
```



3. 해당 요소이면서 부모의 첫번째 자식인 요소(장남,장녀)를 선택합니다.

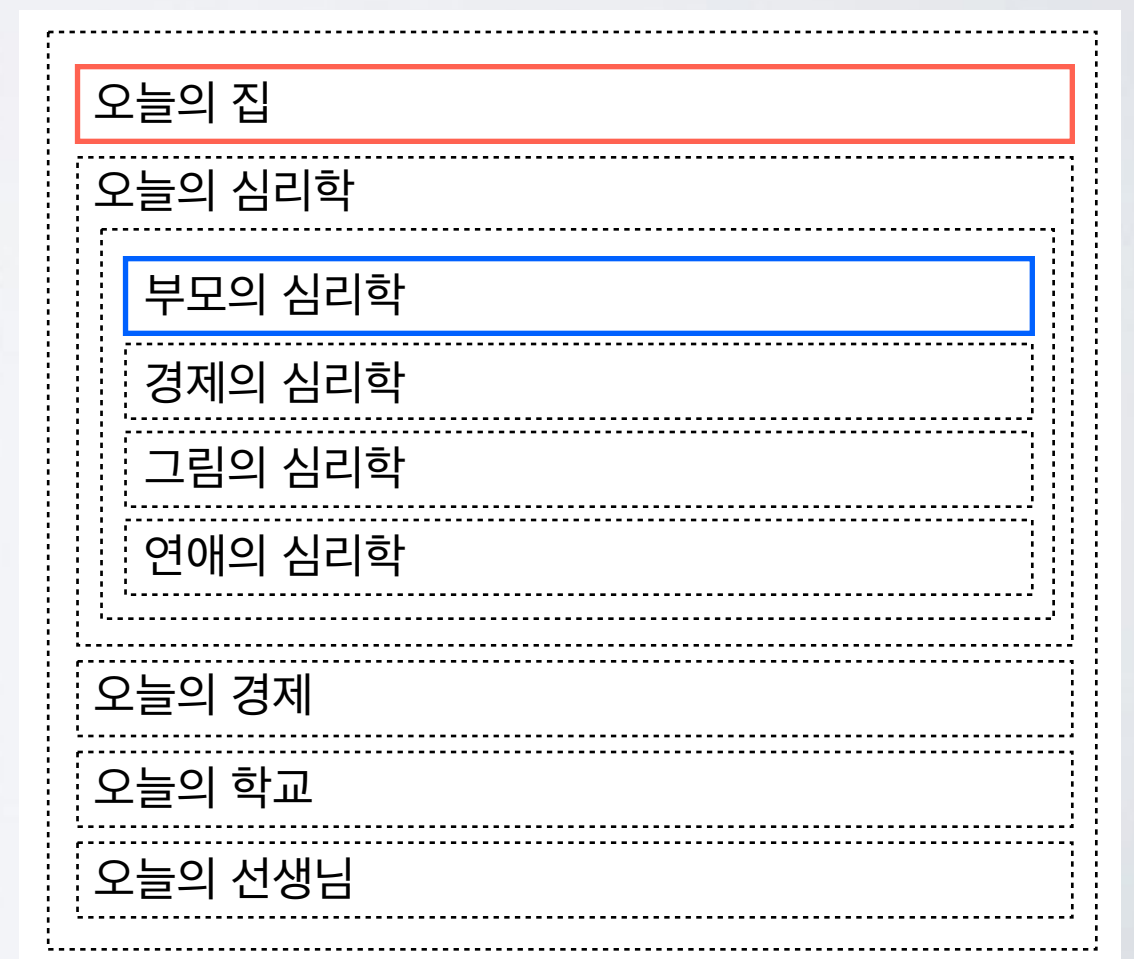
```
.category > li:first-child {  
  border:2px solid red;  
}
```

category의 직계 자식 이면서 category의 첫번째 자식인 요소를 찾아서 선택합니다.

```
.category li li:first-child {  
  border:2px solid blue;  
}
```

category의 자손 중에서 안에 들어있는 들을 찾아서 그중에 첫번째를 선택합니다.

```
<ul class="category">  
  <li>오늘의 집</li>  
  <li>오늘의 심리학  
    <ul>  
      <li>부모의 심리학</li>  
      <li>경제의 심리학</li>  
      <li>그림의 심리학</li>  
      <li>연애의 심리학</li>  
    </ul>  
  </li>  
  <li>오늘의 경제</li>  
  <li>오늘의 학교</li>  
  <li>오늘의 선생님</li>  
</ul>
```



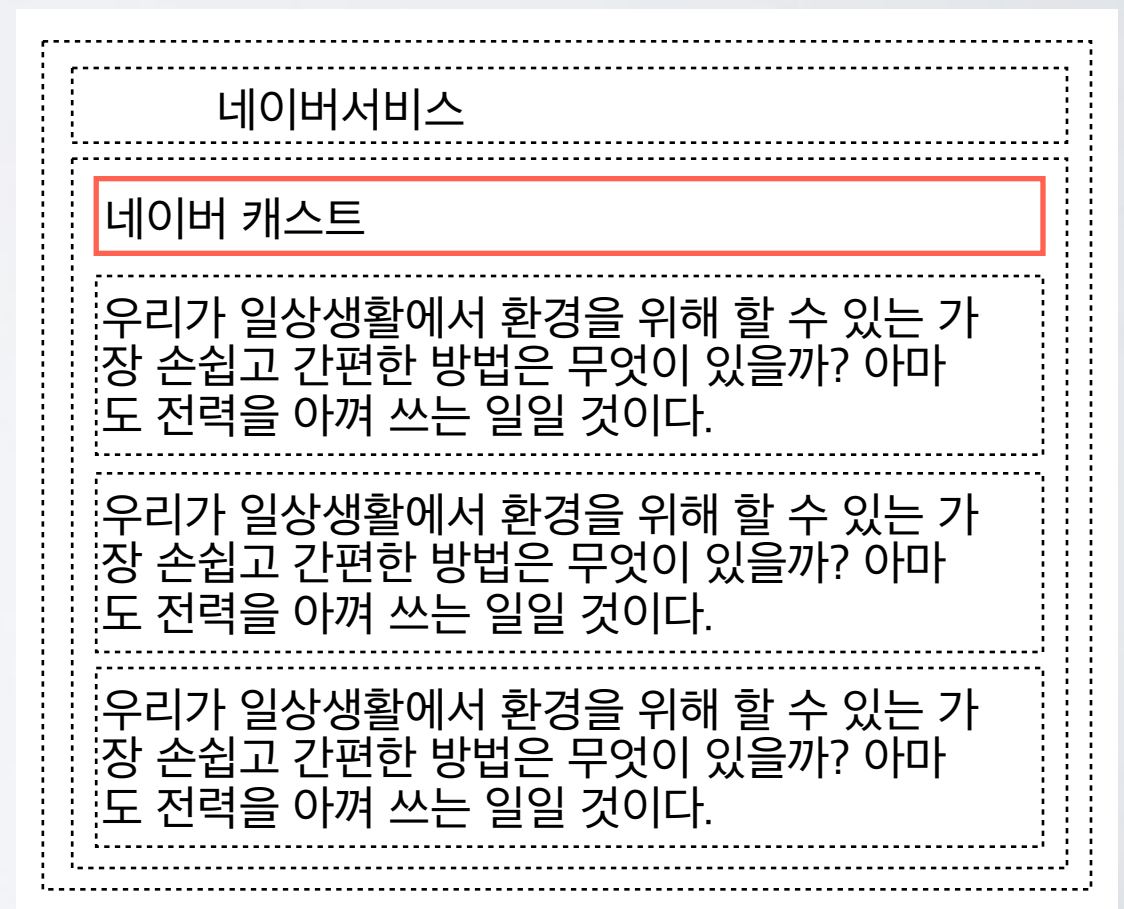
```
.service p:first-child {  
  border:2px solid blue;  
}
```

<p>들 중에서는 첫번째일지 모르지만 부모(castbox)입장에서는 둘째이기 때문에 선택되지 않습니다.

```
.service h3:first-child {  
  border:2px solid red;  
}
```

비록 <h3>는 혼자지만 부모(castbox)입장에서 첫번째 자식이기 때문에 선택됩니다.

```
<div class="service" >  
  <h2>네이버서비스</h2>  
  <div class="castbox">  
    <h3>네이버 캐스트</h3>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 ... 중략</p>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 ... 중략</p>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 ... 중략</p>  
  </div>  
</div>
```



4. 해당 요소이면서 부모의 막내 자식인 요소를 선택합니다.

IE7지원안함

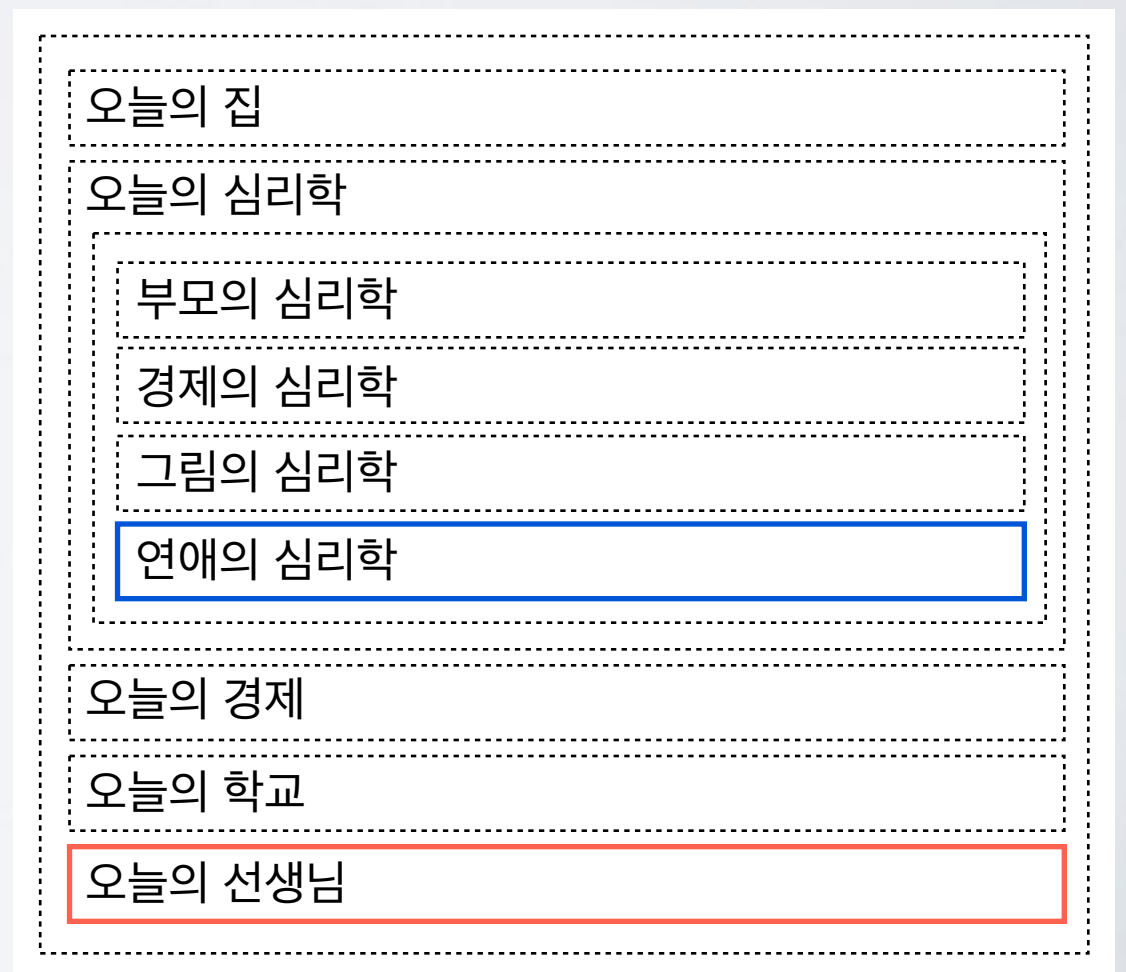
```
.category > li:last-child {  
    border:2px solid red;  
}
```

category의 직계 자식 들 중에서
마지막 요소를 선택합니다.

```
.category li li:last-child {  
    border:2px solid blue;  
}
```

category의 자손 중에서 안에 들어있는
들을 찾아서 그중에 마지막요소를 선택합니다.

```
<ul class="category">  
    <li>오늘의 집</li>  
    <li>오늘의 심리학  
        <ul>  
            <li>부모의 심리학</li>  
            <li>경제의 심리학</li>  
            <li>그림의 심리학</li>  
            <li>연애의 심리학</li>  
        </ul>  
    </li>  
    <li>오늘의 경제</li>  
    <li>오늘의 학교</li>  
    <li>오늘의 선생님</li>  
</ul>
```



5. 해당 요소의 첫째 자식과 마지막 자식을 가상으로 만들어 넣어 줄 수 있습니다.

IE7지원안함

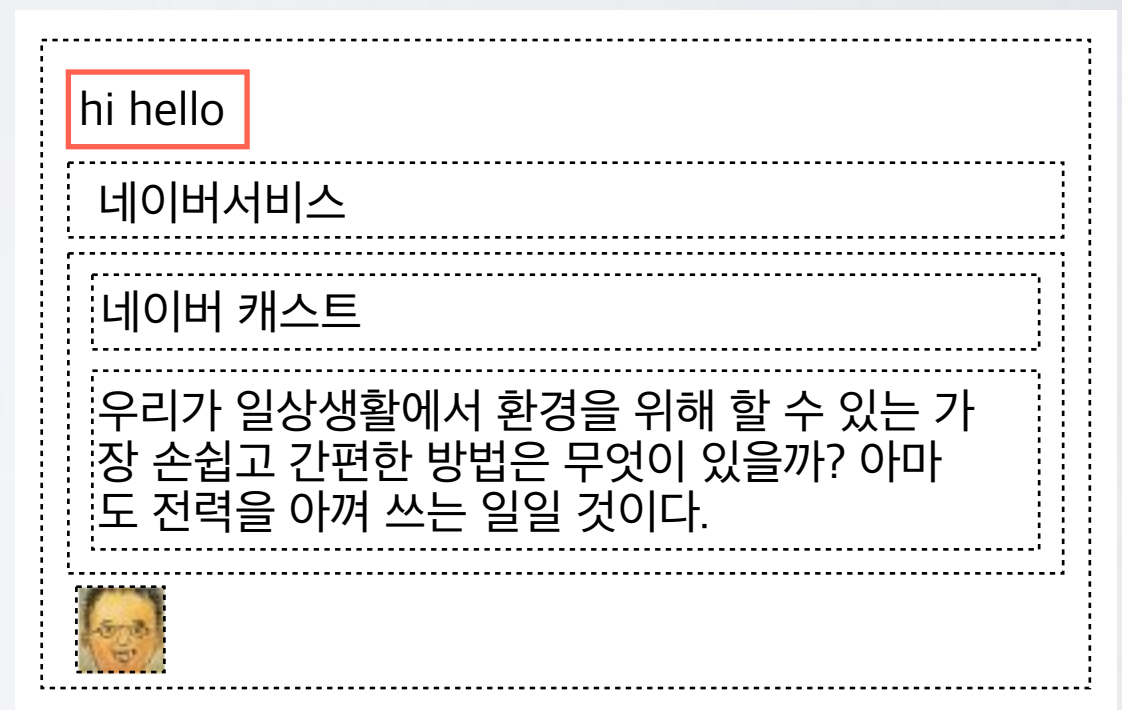
```
.service::before {  
  content: 'hi hello';  
  border: 2px solid red;  
}
```

:before선택자와 content속성을 이용해서
service의 직계자식들의 제일 앞에 hi hello라
는 콘텐츠를 넣어주고 상자를 선택합니다.

```
.service:after {  
  content: url('images/eulsoo.jpg');  
}
```

:after선택자와 content속성을 이용해서 이
미지를 넣었습니다.

```
<div class="service" >  
  <h2>네이버서비스</h2>  
  <div class="castbox">  
    <h3>네이버 캐스트</h3>  
    <p>우리가 일상생활에서 환경을 위해  
    할 수 있는 ... 중략</p>  
  </div>  
</div>
```



6. 포커스를 받을때 해당 요소를 선택합니다.

IE7지원안함

포커스 받았다는 것은 웹문서의 링크나 입력상자 체크박스 라디오버튼등 사용자와 상호작용이 일어나는 곳을 사용자가 클릭하거나 탭키를 눌러 해당 요소를 지나갈 때 포커스를 받았다고 말합니다.

```
.info input:focus {  
    background-color:yellow;  
}
```

```
<div class="info">  
    <label for="phone">전화번호 입력</label>  
    <input id="phone" type="text"/>  
</div>
```

일반상태

전화번호 입력

포커스를 받았을 때

전화번호 입력

<input>에 포커스를 받으면 선택하여
배경색을 바꿔 주었습니다.

```
.menu a:focus {  
    color:blue;  
}
```

```
<div class="menu">  
    <a href="http://test.com">공감각 마케팅</a>  
    <a href="http://test.com">무의식 소비</a>  
    <a href="http://test.com">디지털 치매증후군</a>  
</div>
```

일반상태

공감각 마케팅 무의식 소비 디지털 치매증후군

포커스를 받았을 때

공감각 마케팅 무의식 소비 디지털 치매증후군

<a>에 포커스를 받으면 a링크
글씨색을 바꿔줍니다.

7. 상태와 반응에 따라 해당 요소를 선택할 수 있습니다.

아직 방문하지 않은 링크

```
.menu a:link {color:lime;}
```

이미 방문했던 링크

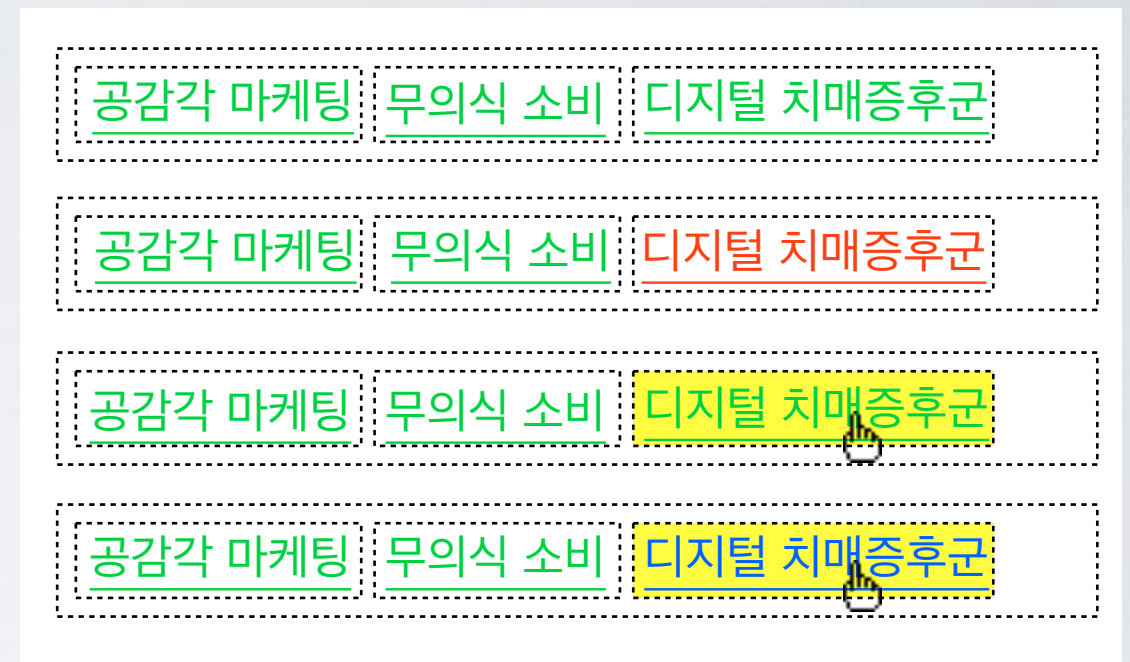
```
.menu a:visited {color:red;}
```

마우스를 오버할때

```
.menu a:hover {background-color:yellow;}
```

클릭하는 순간 (버튼이 눌렸을때)

```
.menu a:active {color:blue;}
```

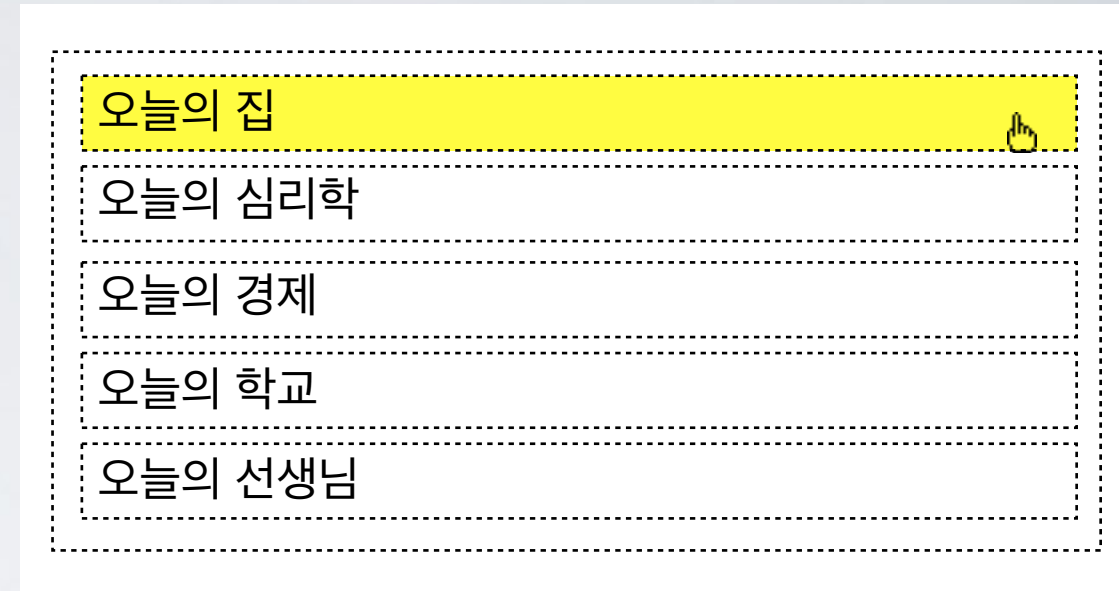


```
<div class="menu">  
  <a href="http://test.com">공감각 마케팅</a>  
  <a href="http://test.com">무의식 소비</a>  
  <a href="http://test.com">디지털 치매증후군</a>  
</div>
```


:hover와 :active는 <a>가 아니어도 사용할 수 있습니다.

에 마우스를 오버할때

```
.category li:hover {  
    background-color:yellow;  
}
```

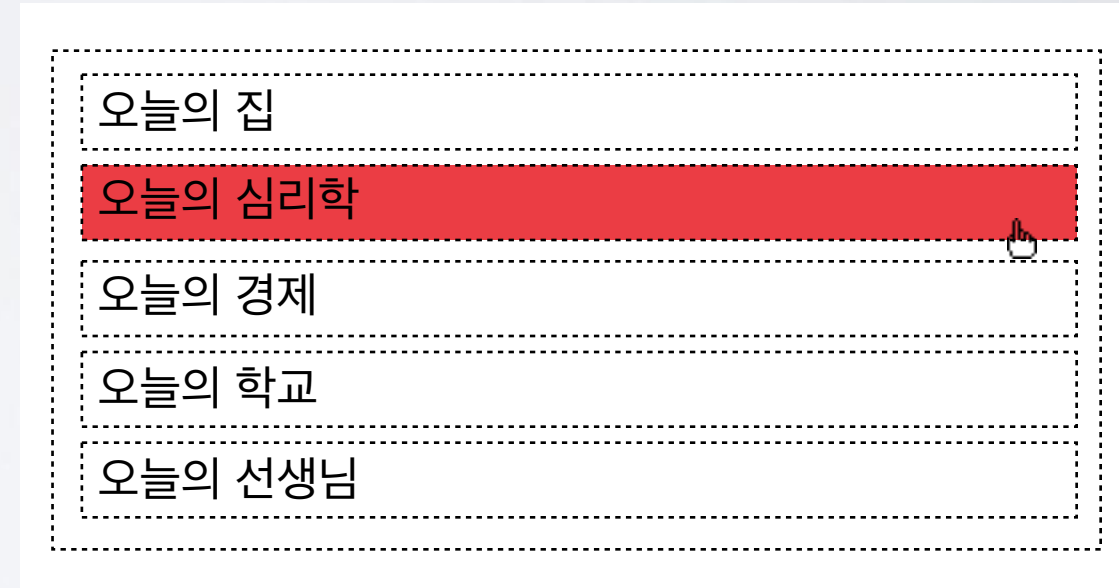


를 클릭하는 순간 (버튼이 눌렸을때)

```
.category li:active {  
    background-color:red;  
}
```

IE7지원안함

```
<ul class="category">  
    <li>오늘의 집</li>  
    <li>오늘의 심리학</li>  
    <li>오늘의 경제</li>  
    <li>오늘의 학교</li>  
    <li>오늘의 선생</li>  
</ul>
```



상속

Inheritance

CSS는 효과적으로 표현을 명령하기 위하여 상속을 이용합니다

글씨크기, 줄간격, 글씨색, 서체등등의 일부 명령들은 일일이 모든 태그에 명령 할 필요없이 부모상자에 적용하면 모든 자손들에 스며들듯이 명령이 적용되는 상속시스템이 적용되어 있습니다.

```
<html>
<head>
<meta charset="utf-8" />
<style>
  body {color:red;}
</style>
</head>
```

```
<body>
  <div>
    <h1>First style</h1>
    <p>style태그를 이용하여 스타일을 더한다.</p>
  </div>
</body>
</html>
```

First style

style속성을 이용해서 스타일을 더한다.



<body>에 적용한 글씨색 속성이 <div>를 거쳐 <h1><p>에 까지 각 각 상속되어 붉은색으로 되었습니다

대상요소 가까이에서 명령되어 상속될수록 우선합니다.

```
<html>
<head>
<meta charset="utf-8"/>
<style>
  body {color:red;}
  div {color:lime;}
</style>
</head>
<body>
  <div>
    <h1>First style</h1>
    <p>style태그를 이용하여
      스타일을 더한다.</p>
  </div>
</body>
</html>
```



First style

style속성을 이용해서 스타일을 더한다.



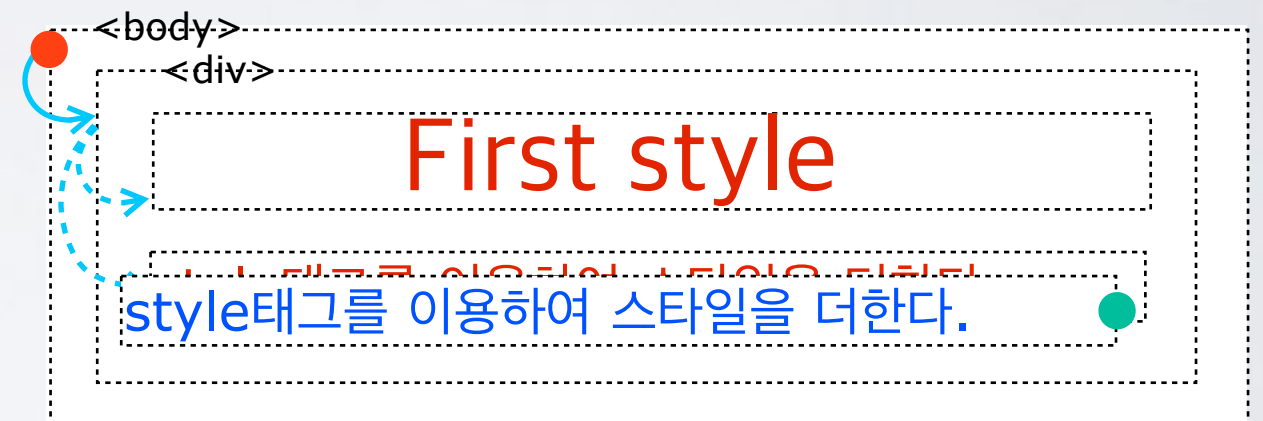
<body>에 지정한 color:red보다 <div>에 지정한 color:lime이 대상인 <h1><p>에서 더 가까이 지정되어 상속되었기 때문에 color:lime이 우선하여 보이게 됩니다

그러므로 당연히 상속된 명령보다 직접 선언된 명령이 우선합니다

```
<html>
<head>
<meta charset="utf-8"/>
<style>
  body {color:red;}
  p {color:blue;}
</style>
</head>
<body>
  <div>
    <h1>First style</h1>
    <p>style태그를 이용하여
      스타일을 더한다.</p>
  </div>
</body>
</html>
```

First style

style속성을 이용해서 스타일을 더한다.

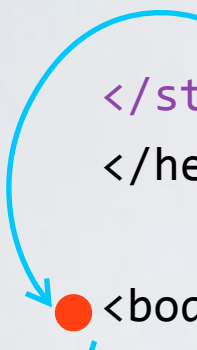


<body>에 지정되어 상속을 통해 <p>에 color:red가 적용된 후에 다시 <p>를 직접 선택하여 color:blue가 재지정 되었습니다

이렇듯 상속된 속성은 직접 선언한 속성보다 우선순위가 낮습니다

〈a〉에는 상속되지 않습니다.

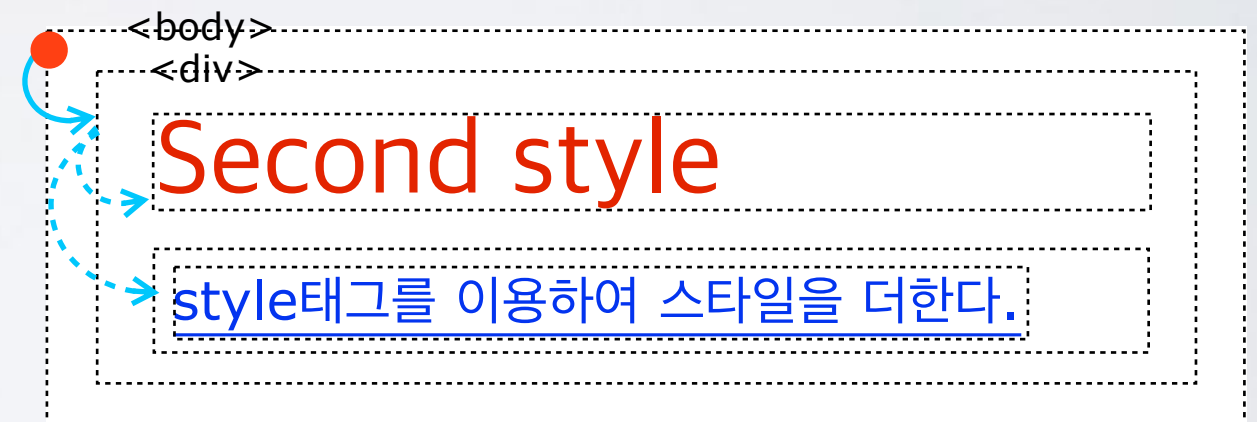
```
<html>
<head>
<meta charset="utf-8"/>
<style>
  body {color:red;}
</style>
</head>
```



```
<body>
  <div>
    <h1>Second style</h1>
    <p><a href="#">style태그를 이용하여
      스타일을 더한다.</a></p>
  </div>
</body>
</html>
```

Second style

style태그를 이용하여 스타일을 더한다.



<body>에 지정한 color:red가 <p>까지만 상속되고 <a>에는 상속되지 못합니다.

그래서 <a>는 별도로 지정합니다.

```
<html>
<head>
<meta charset="utf-8"/>
<style>
  body {color:red;}
  a {color:orange;}
</style>
</head>
<body>
  <div>
    <h1>Second style</h1>
    <p><a href="#">style태그를 이용하여
      스타일을 더한다.</a></p>
  </div>
</body>
</html>
```

Second style

style태그를 이용하여 스타일을 더한다.



<body>에 지정한 color:red가 <p>까지만 상속되고 <a>에는 상속되지 못하여 <a>에 별도로 색을 지정하였습니다

배경, 넓이, 높이, 여백, 간격, 위치값등 상속되면 않되는 속성들도 많습니다

상속 속성

color	글씨색
cursor	커서
font-family	서체
font-size	글씨크기
font-weight	글씨굵기
font	폰트관련 축약
letter-spacing	글씨 사이의 간격
line-height	글줄의 간격
list-style	목록 스타일
text-align	글 좌,우,가운데 정렬
text-indent	들여쓰기
text-transform	대소문자
word-spacing	단어의 간격
visibility	보이는 상태 조절
border-collapse	테이블 보더 병합
border-spacing	테이블 보더 간격
caption-side	테이블 제목 위치

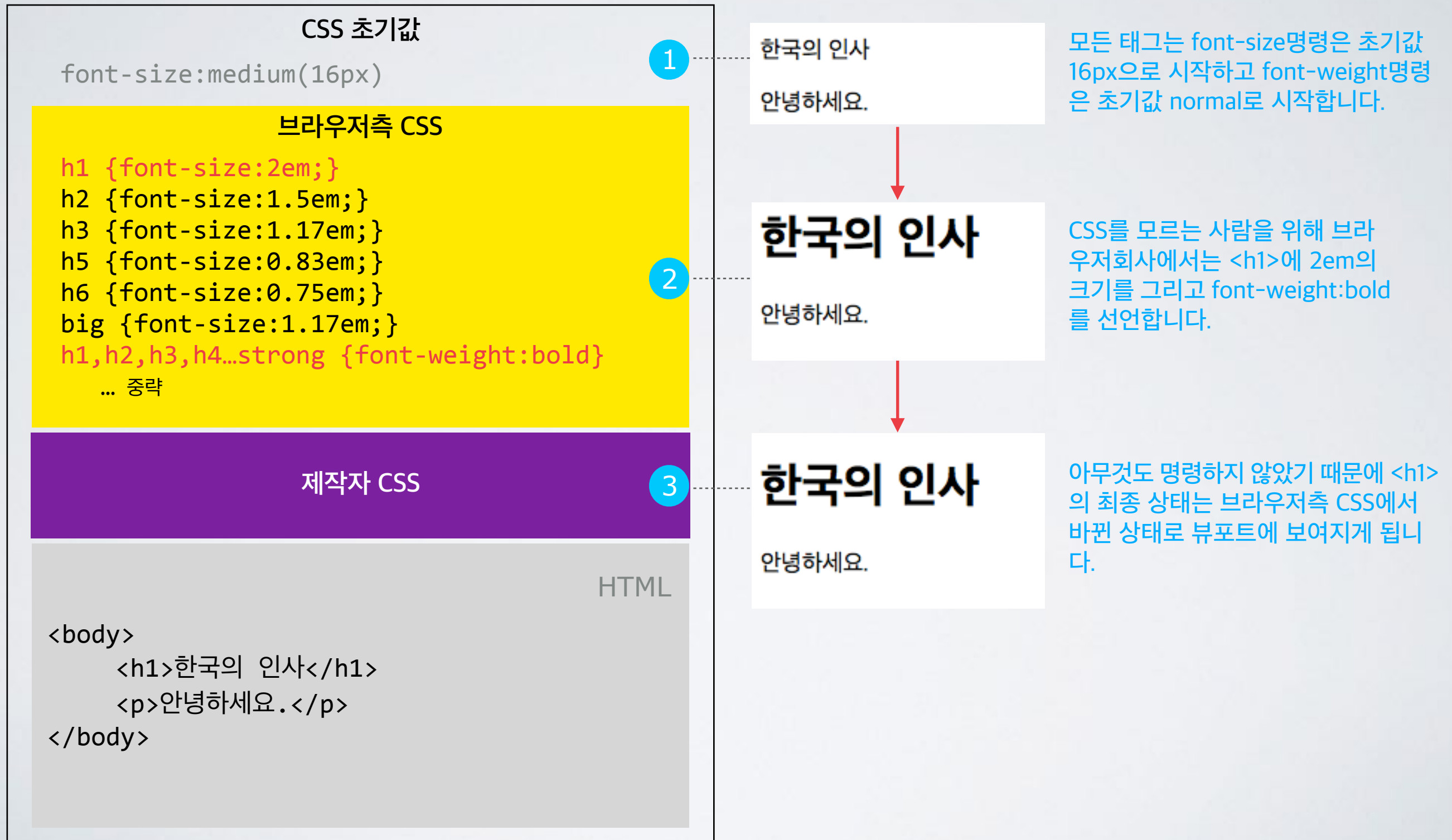
비상속 속성

background	배경
width	넓이지정
height	높이지정
padding	여백
margin	간격
border	선
top, left, right, bottom	offset값 지정
float	띄워서 좌우 배치
position	배치 상태 지정
display	박스타입 지정
overflow	넘치는 상황
vertical-align	글줄안에서 상하 정렬 기준
z-index	보이는 순서

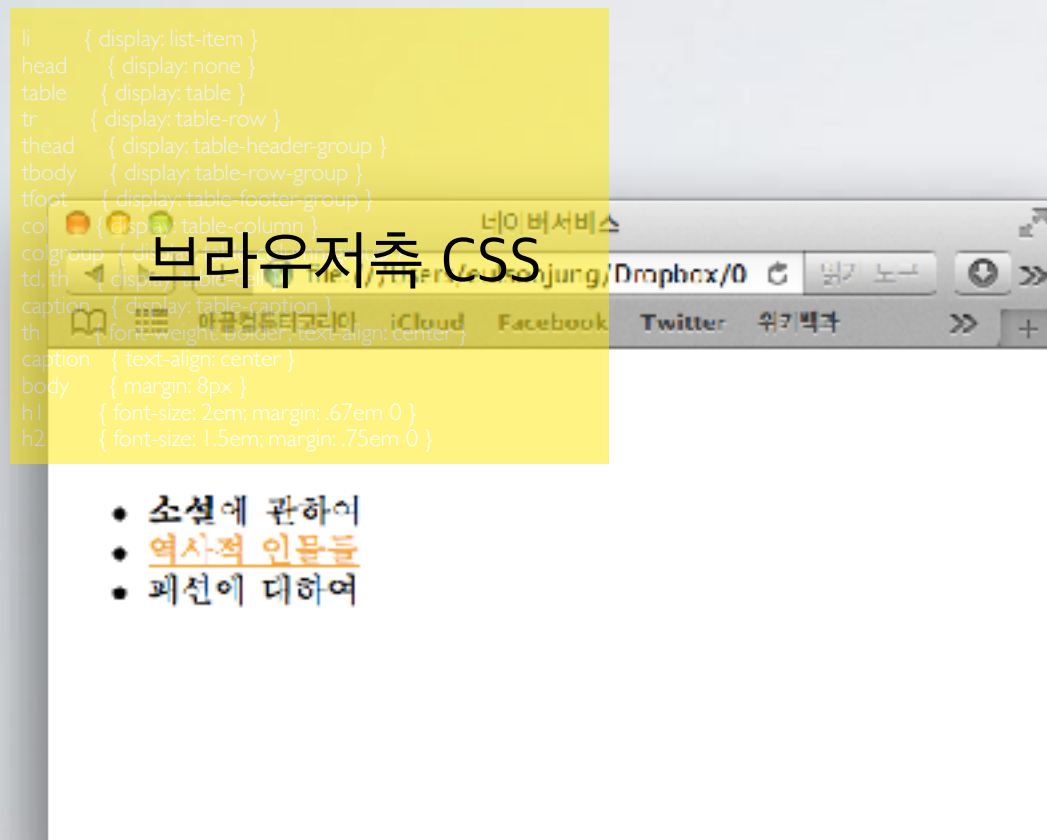
브라우저측 CSS

User agent CSS

브라우저는 우리(제작자)가 CSS 명령을 내리기 전에 2가지 단계를 거친다고 했습니다.

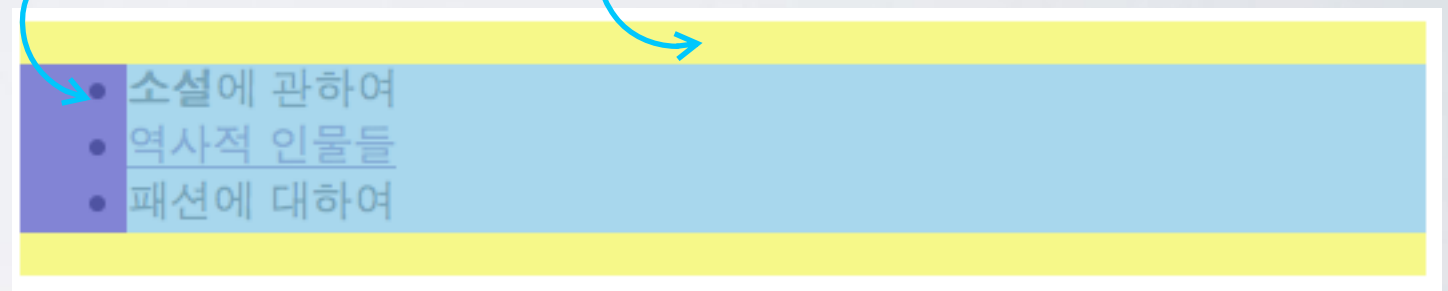


이렇게 CSS를 알지 못하는 사람들을 위해서
브라우저측 CSS에 기본적인 표현들을 지정해 두었습니다.



박스 왼쪽에 여백이 설정되고
중점을 찍어 주었습니다

박스에 상하로 간격이 설정하였습니다



```
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=utf-8"/>
  <title>네이버서비스</title>
</head>

<body>
<ul>
  <li><strong>소셜</strong>에 관하여</li>
  <li><a href="">역사적 인물들</a></li>
  <li>패션에 대하여</li>
</ul>
</body>
```

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: 3em 0 }

브라우저측 CSS

네이버 캐스트

"우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까?" 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

- 소설에 관하여
- 역사적 인물들
- 패션에 대하여

 중점과 들여쓰기

<a>글씨색, 밑줄

<h1>~<h6>단계별사이즈, 볼드, 상/하 간격

<p> 상하마진
<q> 쌍따옴표
 이탤릭

<h2>네이버서비스</h2>

<div>

<h3>네이버 캐스트</h3>

<p><q>우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까?</q> 아마도 전력을 아껴 쓰는일일 것이다.</p>

</div>

<div>

<h3>오픈 캐스트</h3>

소설에 관하여
역사적 인물들
패션에 대하여

</div>

브라우저측 CSS

```
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 }
h6 { font-size: .75em; margin: 1.67em 0 }
h1, h2, h3, h4,
h5, h6, b,
strong { font-weight: bolder }
blockquote { margin-left: 40px; margin-right: 40px }
```

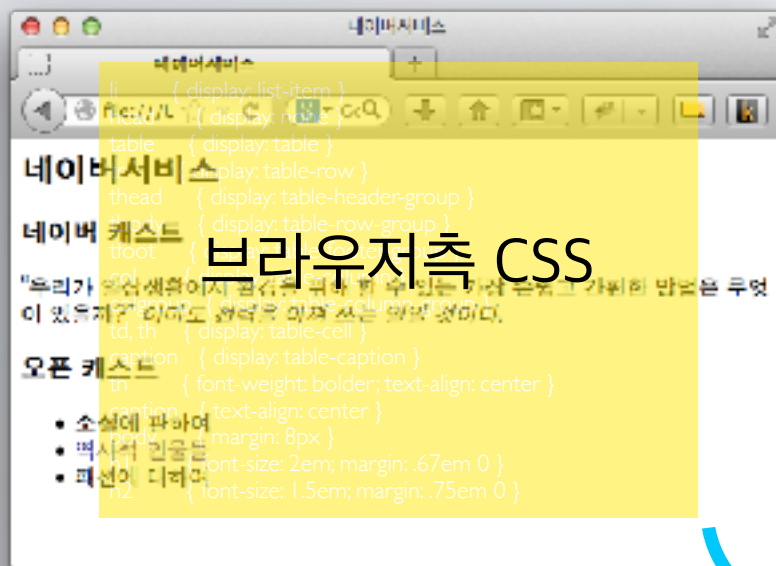
```
i, cite, em,
var, address { font-style: italic }
pre, tt, code,
kbd, samp { font-family: monospace }
pre { white-space: pre }
button, textarea,
input, select { display: inline-block }
big { font-size: 1.17em }
small, sub, sup { font-size: .83em }
sub { vertical-align: sub }
sup { vertical-align: super }
table { border-spacing: 2px; }
thead, tbody,
tfoot { vertical-align: middle }
td, th, tr { vertical-align: inherit }
s, strike, del { text-decoration: line-through }
hr { border: 1px inset }
ol, ul, dir,
menu, dd { margin-left: 40px }
ol { list-style-type: decimal }
ol ul, ul ol,
ul ul, ol ol { margin-top: 0; margin-bottom: 0 }
u, ins { text-decoration: underline }
br:before { content: "\A"; white-space: pre-line }
center { text-align: center }
:link, :visited { text-decoration: underline }
:focus { outline: thin dotted invert }

/* Begin bidirectionality settings (do not change) */
BDO[DIR="ltr"] { direction: ltr; unicode-bidi: bidi-override }
BDO[DIR="rtl"] { direction: rtl; unicode-bidi: bidi-override }

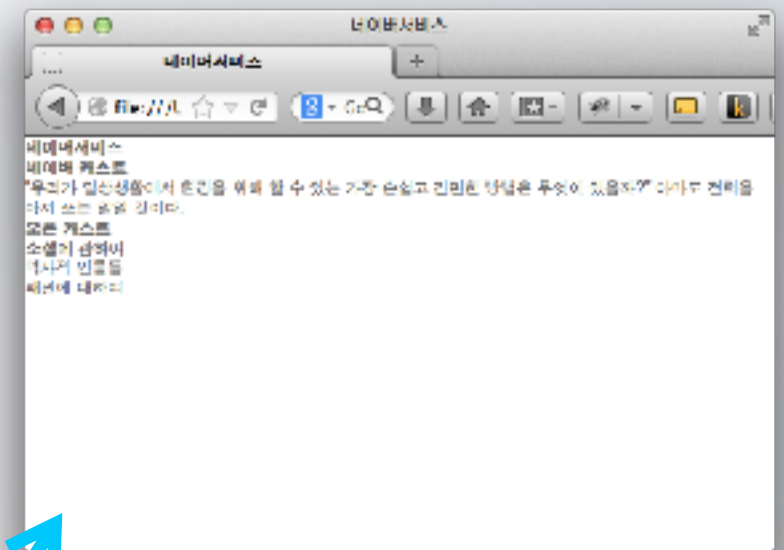
*[DIR="ltr"] { direction: ltr; unicode-bidi: embed }
*[DIR="rtl"] { direction: rtl; unicode-bidi: embed }

@media print {
  h1 { page-break-before: always }
  h1, h2, h3,
  h4, h5, h6 { page-break-after: avoid }
  ul, ol, dl { page-break-before: avoid }
}
```

전문적인 제작자들은 브라우저측 CSS의 배려가 필요없기 때문에 리셋용 CSS 명령을 만들어 씁니다.



작업자의
초기화 CSS



기본적인 인지를 위한 간격과 중점, 들여쓰기 제목의 글씨 크기들을 새로운 디자인을 적용하기 위해 깨끗하게 없애버린 상태로 만들기 위해 태그별로 간격과 글씨등의 CSS명령을 내리게 됩니다.
이런것을 흔히 리셋용 CSS라고 부르고 있습니다.

리셋용 CSS의 한예를 들어볼까요?

CSS 초기값

font-size:medium(16px)

0

브라우저측 CSS

```
h1 {font-size:2em;}
h2 {font-size:1.5em;}
h3 {font-size:1.17em;}
h5 {font-size:0.83em;}
h6 {font-size:0.75em;}
h1,h2,h3,h4...strong {font-weight:bold}
... 중략
```

1

제작자 CSS

```
/* 리셋용 스타일 */
body {font-size:12px}
h1, h2, h3, h4, h5, h6 {font-size:12px}
```

2

HTML

```
<body>
  <h1>한국의 인사</h1>
  <p>안녕하세요.</p>
</body>
```

한국의 인사
안녕하세요.

모든 태그는 font-size명령은 초기값 16px으로 시작하고 font-weight명령은 초기값 normal로 시작합니다.

한국의 인사
안녕하세요.

CSS를 모르는 사람을 위해 브라우저회사에서는 <h1>에 2em의 크기를 그리고 font-weight:bold를 선언합니다.

한국의 인사
안녕하세요.

이렇게 동일한 크기의 글씨로 시작하면 작업하기가 수월하기 때문에 각각 다른 제목과 본문 글씨크기를 하나의 크기로 맞추어 놓습니다.

작업자의 리셋용 CSS의 예시입니다.

reset.css

```
body {  
  font:normal 12px Arial, sans-serif;  
  line-height:1.3;  
  color:#666666;  
  background-color:#fff;  
  margin:0;  
  padding:0;  
}  
h1, h2, h3, h4, h5, h6 {font-size:1em;}  
h1, h2, h3, h4, h5, h6, div, p, dl, dt, dd, ul, ol, li, form, fieldset,  
blockquote, address, table, thead, tbody, tfoot, tr, td, caption {  
  margin:0;  
  padding:0;  
}  
ul, ol {list-style:none;}  
img {border:0;}  
a {text-decoration:none;}  
table {  
  border-collapse: collapse;  
  border-spacing: 0;  
}
```

<body>에 글씨크기 색, 줄간격등을 주어 모든 태그에 상속시켜 적용 되도록 하였습니다.

모든 제목에 1em을 주어 가장 부모인 <body>의 글씨크기 12px를 그대로 받아 적용하게 하였습니다. (em단위는 폰트와 텍스트에서 다릅니다)

기본적인 인지를 위해 박스의 간격과 여백이 들어가 있는 태 그들에 "0" 값을 선언함으로 모두 초기화 하였습니다.

리스트에 중점과 숫자표시를 없앴습니다.

ie에서 이미지에 기본으로 생기는 보더 초기화

링크에 기본으로 생기는 밑줄을 없앱니다.

테이블의 선을 합친 스타일로 초기화합니다.

reset.css

리셋용 명령들...

```
<html>
<head>
  <title>네이버서비스</title>
  <link href="css/reset.css" rel="stylesheet">
  <style>
    p {color:red;} 초기화 CSS가 브라우저의 기본표현을 초기화 시킨후 <style>안에 글씨색을 재지정 하였습니다.
  </style>
</head>

<body>
<div>
  <h2>네이버서비스</h2>
  <div>
    <h3>네이버 캐스트</h3>
    <p><q>우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까?</q> <em>아마도 전력을 아껴 쓰는 일일 것이다.</em></p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <ul>
      <li><strong>소설</strong>에 관하여</li>
      <li><a href="">역사적 인물들</a></li>
      <li>패션에 대하여</li>
    </ul>
  </div>
</div>
</body>
</html>
```

네이버서비스

네이버 캐스트

“우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까?” 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

소설에 관하여

역사적 인물들

패션에 대하여

네이버서비스

네이버서비스

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.


소설에 관하여

역사적 인물들

패션에 대하여

하향우선

HTML은 위에서 아래로 차례대로 처리합니다

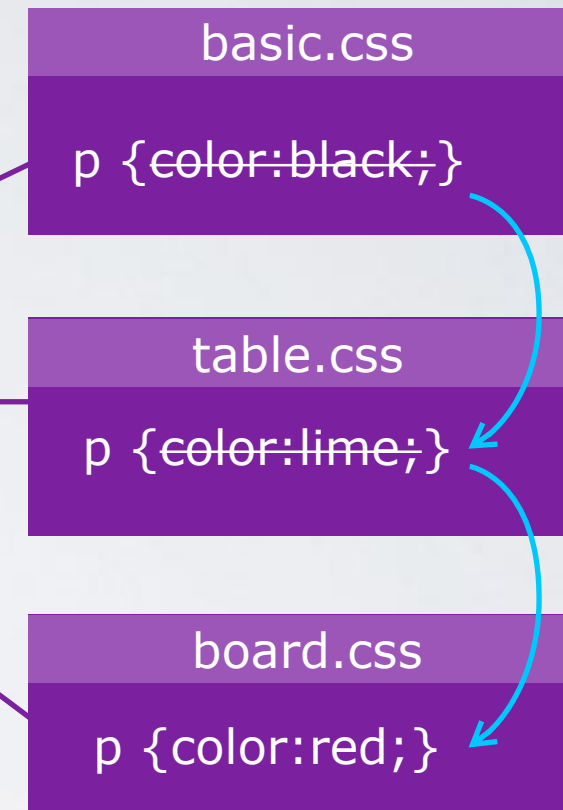


```
<html>
<head>
  <meta charset="utf-8" />
  <title>hello world!</title>
  <link href="css/basic.css" rel="stylesheet" />
  <link href="css/table.css" rel="stylesheet" />
  <link href="css/board.css" rel="stylesheet" />
</head>
<body>
  <h1>Second style</h1>
  <p>style태그를 이용하여 스타일을 더한다.</p>
</body>
</html>
```

그렇기 때문에 여러개의 CSS파일을
사용할 경우에도 차례대로 인식하게 됩니다.

하나의 요소에 여러곳에서 명령할 경우 덮어 씌워지게(override) 됩니다

```
<html>
<head>
  <meta charset="utf-8" />
  <title>hello world!</title>
  <link href="css/basic.css" rel="stylesheet" />
  <link href="css/table.css" rel="stylesheet" />
  <link href="css/board.css" rel="stylesheet" />
</head>
<body>
  <h1>Second style</h1>
  <p>style태그를 이용하여 스타일을 더한다.</p>
</body>
</html>
```



Second style

style태그를 이용하여 스타일을 더한다.

Second style

style태그를 이용하여 스타일을 더한다.

〈style〉안에서도 마찰가지고

```
<html>
<head>
<meta charset="utf-8" />
<title>hello world!</title>
<style type="text/css">
  h1 {color:black;}
  p {color:yellow;}
  p {color:red;}
</style>
</head>

<body>
  <h1>Second style</h1>
  <p>style태그를 이용하여 스타일을 더한다.</p>
</body>
</html>
```

Second style

style태그를 이용하여 스타일을 더한다.

Second style

style태그를 이용하여 스타일을 더한다.

동일한 방법으로 동일한 대상을 선택했을 경우
위에서 아래의 순서로 처리하는 브라우저에
의해서 뒷선언을 앞선언이 덮어 씌워(override)
결국 아래 선언이 표현됩니다.

CSS 파일 안에서도 마찰가지입니다.

```
<html>
<head>
<meta charset="utf-8" />
<title>hello world!</title>
<link href="css/basic.css" ... />
</head>
```

```
<body>
  <h1>Second style</h1>
  <p>style태그를 이용하여 스타일을 더한다.</p>
</body>
</html>
```

basic.css

```
h1 {color:black;}
p {color:black;}
p {color:lime;}
p {color:red;}
```

Second style

style태그를 이용하여 스타일을 더한다.

Second style

style태그를 이용하여 스타일을 더한다.
style태그를 이용하여 스타일을 더한다.
style태그를 이용하여 스타일을 더한다.

대상인 HTML에 가까울 수록 우선순위가 높아집니다.

```
<html>
<head>
<meta charset="utf-8"/>
<title>hello world!</title>
<link href="css/basic.css" ... />
<link href="css/table.css" ... />
<style type="text/css">
  h1 {color:black;}
  p {color:yellow;}
  p {color:red;}
</style>
</head>

<body>
  <h1>Second style</h1>
  <p>style태그를 이용하여 스타일을 더한다.</p>
</body>
</html>
```

basic.css

```
h1 {color:black;}
p {color:green;}
```

table.css

```
p {color:lime;}
```

Second style

style태그를 이용하여 스타일을 더한다.

Second style

style태그를 이용하여 스타일을 더한다.

@import 된 스타일도 Cascading됩니다

```
<html>
<head>
<meta charset="utf-8"/>
<title>hello world!</title>
<link href="css/basic.css" ... />
<style type="text/css">
  h1 {color:black;}
  p {color:yellow;}
  p {color:red;}
</style>
</head>

<body>
  <h1>Second style</h1>
  <p>style태그를 이용하여 스타일을 더한다.</p>
</body>
</html>
```

basic.css

```
@import url('table.css')
p {color:black;}
p {color:green;}
```

table.css

```
p {color:lime;}
```

Second style

style태그를 이용하여 스타일을 더한다.

Second style

style태그를 이용하여 스타일을 더한다.

그렇기 때문에 리셋용 CSS는 항상 제일 위에 넣어 가장 먼저 적용되게 만들어야 합니다

```
<html>
<head>
  <meta charset="utf-8"/>
  <title>네이버서비스</title>
  <link href="css/reset.css" ...>
  <link href="css/naver.css" ...>
</head>

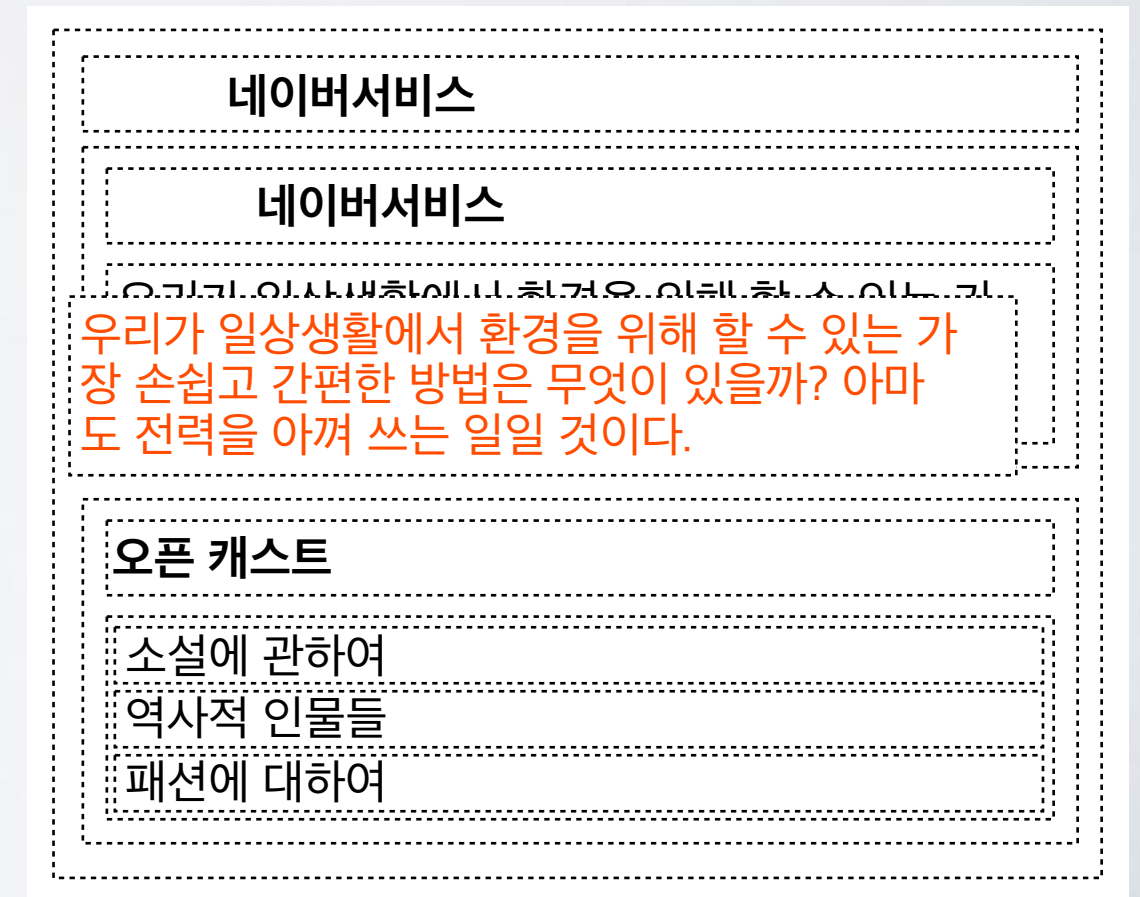
<body>
<div>
  <h2>네이버서비스</h2>
  <div>
    <h3>네이버 캐스트</h3>
    <p><q>우리가 일상생활에서 환경을 위해
    할 수 있는 가장 손쉽고 간편한 방법은
    무엇이 있을까?</q> <em>아마도 전력을 아껴 쓰는
    일일 것이다.</em></p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <ul>
      <li><strong>소설</strong>에 관하여</li>
      <li><a href="">역사적 인물들</a></li>
      <li>패션에 대하여</li>
    </ul>
  </div>
</div>
</body>
</html>
```



네이버서비스
네이버 캐스트

“우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까?” 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트
소설에 관하여
역사적 인물들
패션에 대하여



명시도

Specificity

하향우선법칙을 이기려면 더 구체적으로 부르면 됩니다.

```
div h2 {font-size:24px;}
```

0 0 2

```
h2 {font-size:16px;}
```

0 0 1

<h1>을 직접 부르는 것보다
<div>자손인 <h1>이라고 부르면
우선순위가 높습니다. 태그1개보다
2개를 이용해서 불렀기 때문입니다.

font-size:16px

font-size:24px

```
<div class="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해 할 수
      있는 가장 손쉽고 간편한 ...종략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
      이 생산되는 ... 종략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버 서비스

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

class이름이 태그이름보다 높습니다.

태그2개 클래스1개로 총12가 되었습니다.

div.service h2 {color:lime;}	0 1 2
.service h2 {color:yellow;}	0 1 1
h2 {color:red;}	0 0 1
div h2 {color:blue;}	0 0 2

```
<div class="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해 할 수
      있는 가장 손쉽고 간편한 ...중략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
      이 생산되는 ... 중략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버 서비스

네이버 서비스

네이버 서비스

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

class이름과 :after(가상선택자)는 명시도가 같습니다.

:first-child, :last-child, :before, :after, :hover, :active, :link, :visited

```
.castbox h3 {color:yellow;}
```

0 1 1

```
h3:first-child {color:lime;}
```

0 1 1

둘다 class1, 태그1로 총011이됩니다. 하지만 아래줄이 우선하기 때문에 lime색이 되었습니다.

```
<div class="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해 할 수
      있는 가장 손쉽고 간편한 ...중략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
      이 생산되는 ... 중략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버 서비스

네이버 캐스트

네이버캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.


```
<div class="menu">
  <a href="http://test.com">공감각 마케팅</a>
  <a href="http://test.com">무의식 소비</a>
  <a href="http://test.com">디지털 치매증후군</a>
</div>
```

↓

.menu a: hover {color: yellow}	0 2 1
.menu a: active {color: blue;}	0 2 1
.menu a: link {color: lime;}	0 2 1
.menu a: visited {color: red;}	0 2 1

가상선택자의 구체성이 같기 때문에
아래코드가 우선하여 :hover, :active가
적용되지 않는다.

↓

.menu a: link {color: lime;}	0 2 1
.menu a: visited {color: red;}	0 2 1
.menu a: hover {color: yellow}	0 2 1
.menu a: active {color: blue;}	0 2 1

순서를 바꾸어 모두 작동하게 만든다.

방문전

공감각 마케팅 무의식 소비 디지털 치매증후군

방문후

공감각 마케팅 무의식 소비 디지털 치매증후군

마우스오버

공감각 마케팅 무의식 소비 디지털 치매증후군

마우스클릭

공감각 마케팅 무의식 소비 디지털 치매증후군

방문전

공감각 마케팅 무의식 소비 디지털 치매증후군

방문후

공감각 마케팅 무의식 소비 디지털 치매증후군

마우스오버

공감각 마케팅 무의식 소비 디지털 치매증후군

마우스클릭

공감각 마케팅 무의식 소비 디지털 치매증후군

class 이름과 [type](속성선택자)는 같습니다.

[value] [type="text"] [class~="red"] [class^="cast"] [class\$="02"] [href*="naver.com"]

```
div.castbox > h3 {color:yellow;}
```

0 1 2

```
div[class="castbox"] > h3 {color:lime;}
```

0 1 2

둘다 class1, 태그2로 총012이 되고
">"는 영향을 주지 않았습니다.
하지만 아래줄이 우선하기 때문에
lime색이 되었습니다.

```
<div class="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해 할 수
      있는 가장 손쉽고 간편한 ...종략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
      이 생산되는 ... 종략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버 서비스

네이버 캐스트

네이버캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

id이름이 class이름보다 높습니다.

```
div div.castbox h3 {color:yellow;} 0 1 3
```

```
#service h2 {color:lime;} 1 0 1
```

아이디1 태그1로 총101이 되었습니다.

```
<div class="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해 할 수
      있는 가장 손쉽고 간편한 ...종략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
      이 생산되는 ... 종략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버 서비스

네이버 캐스트

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

">" "*" "+" ","는 명시도에 영향을 미치지 않습니다.

```
.castbox h3 + p {color:yellow;}
```

0 1 2

```
div.castbox > p {color:lime;}
```

0 1 2

"+"와 ">"는 영향을 줄 수 없고 둘 다 class1, 태그2로 총012가됩니다. 하지만 아래줄이 우선하기 때문에 lime색이 되었습니다.

```
<div class="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해 할 수
      있는 가장 손쉽고 간편한 ...종략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p>스티커의 디자인이 참 독특하다. 전
      이 생산되는 ... 종략</p>
  </div>
</div>
```

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

네이버 서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전 이 생산되는

스티커의 디자인이 참 독특하다. 전 이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커 등 다양하다.

아무리 명시도가 높아도 스타일 속성을 이용한 인라인스타일을 이길 수 없습니다.

```
.castbox h3 + p {color:yellow;}
```

0 1 2

```
div#service .castbox > p {color:lime;}
```

1 1 2

```
<div id="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해 할 수
      있는 가장 손쉽고 간편한 ...종략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p style="color:red">스티커의 디자인이
      참 독특하다. 전이 생산되는 ... 종략</p>
  </div>
</div>
```

아이디까지 사용하여 명시도를 높였지
만 결국 붉은색글이 되었습니다.

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한
방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는과정을 보여주는
스티커, 전력 소비로 인한 환경오염을 노골적으로 보여주는 스티
커등 다양하다.

네이버 서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가

장 손쉽고 간편한 방법은 무엇이 있을까? 아마
도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는

과정을 보여주는 스티커, 전력 소비로 인한 환경
오염을 노골적으로 보여주는 스티커등 다양하다.

!important 선언으로 인라인 스타일을 이길 수 있습니다.

```
.castbox h3 + p {color:yellow;}
```

0 1 2

```
.castbox > p {color:lime !important;}
```

1 1 2

```
<div id="service" >
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p>우리가 일상생활에서 환경을 위해 할 수
      있는 가장 손쉽고 간편한 ...중략</p>
  </div>
  <div class="castbox">
    <h3>오픈 캐스트</h3>
    <p style="color:red">스티커의 디자인이
      참 독특하다. 전이 생산되는 ... 중략</p>
  </div>
</div>
```

아이디까지 사용하여 명시도를 높였지만 결국 붉은색글이 되었습니다.

네이버서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는과정을 보여주는 스티커, 전력 소비로 인한 환경오염을 노골적으로 보여주는 스티커등 다양하다.

네이버 서비스

네이버 캐스트

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까? 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

스티커의 디자인이 참 독특하다. 전이 생산되는

스티커의 디자인이 참 독특하다. 전이 생산되는 과정을 보여주는 스티커, 전력 소비로 인한 환경 오염을 노골적으로 보여주는 스티커등 다양하다.

우선순위의 정리

Priority summary

!important

인라인 스타일

명시도 우선

하향우선

일반적인 선언

브라우저측 기본스타일

CSS를 넣지 않았으므로 브라우저 기본 스타일만 적용되어진 상태입니다.

```
<html>
<head>
  <meta charset="utf-8"/>
  <title>네이버서비스</title>
</head>

<body>
<div>
  <h2>네이버서비스</h2>
  <div>
    <h3>네이버 캐스트</h3>
    <p><q>우리가 일상생활에서 환경을 위해
할 수 있는 가장 손쉽고 간편한 방법은
무엇이 있을까?</q> <em>아마도 전력을 아껴 쓰는
일일 것이다.</em></p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <ul>
      <li><strong>소설</strong>에 관하여</li>
      <li><a href="">역사적 인물들</a></li>
      <li>패션에 대하여</li>
    </ul>
  </div>
</div>
</body>
</html>
```

네이버서비스

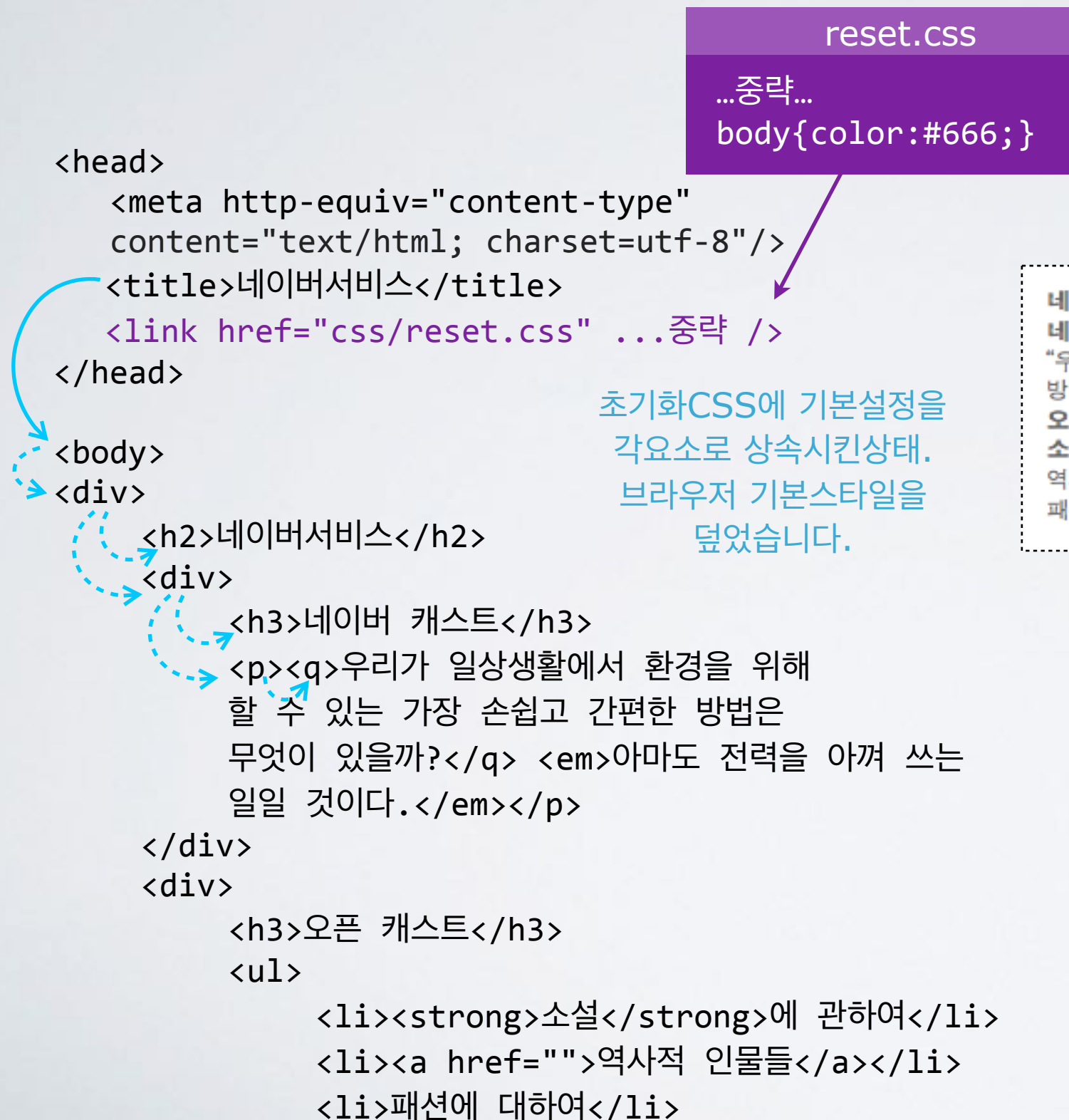
네이버 캐스트

“우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까?” 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

- 소설에 관하여
- 역사적 인물들
- 패션에 대하여

6위. 제작자가 원하는 스타일을 넣었음으로 브라우저 스타일보다 우선합니다.



네이버서비스

네이버 캐스트

“우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한 방법은 무엇이 있을까?” 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

소설에 관하여
역사적 인물들
패션에 대하여

장 손쉽고
껴 쓰는

오픈 캐스트

- 소설에 관하여
- 역사적 인물들
- 패션에 대하여

5위. 넣은 스타일중 직접 선택된 선언이 상속된 선언보다 우선합니다.

reset.css

...중략...
body{color:#666;}

```
<html>
<head>
  <meta charset="utf-8"/>
  <title>네이버서비스</title>
  <link href="css/reset.css" ...중략 />
  <style type="text/css">
    p {color:red;}
  </style>
</head>
<body>
  <div>
    <h2>네이버서비스</h2>
    <div>
      <h3>네이버 캐스트</h3>
      <p><q>우리가 일상생활에서 환경을 위해
        할 수 있는 가장...중략</p>
    </div>
    <div>
      <h3>오픈 캐스트</h3>
      <ul>
        <li><strong>소설</strong>에 관하여</li>
        <li><a href="">역사적 인물들</a></li>

```

네이버서비스

네이버서비스
네이버 캐스트

"우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한
방법은 무엇이 있을까?" 아마도 전력을 아껴 쓰는 일일 것이다.

소설에 관하여
역사적 인물들
패션에 대하여

손쉽고
아껴 쓰는

오픈 캐스트

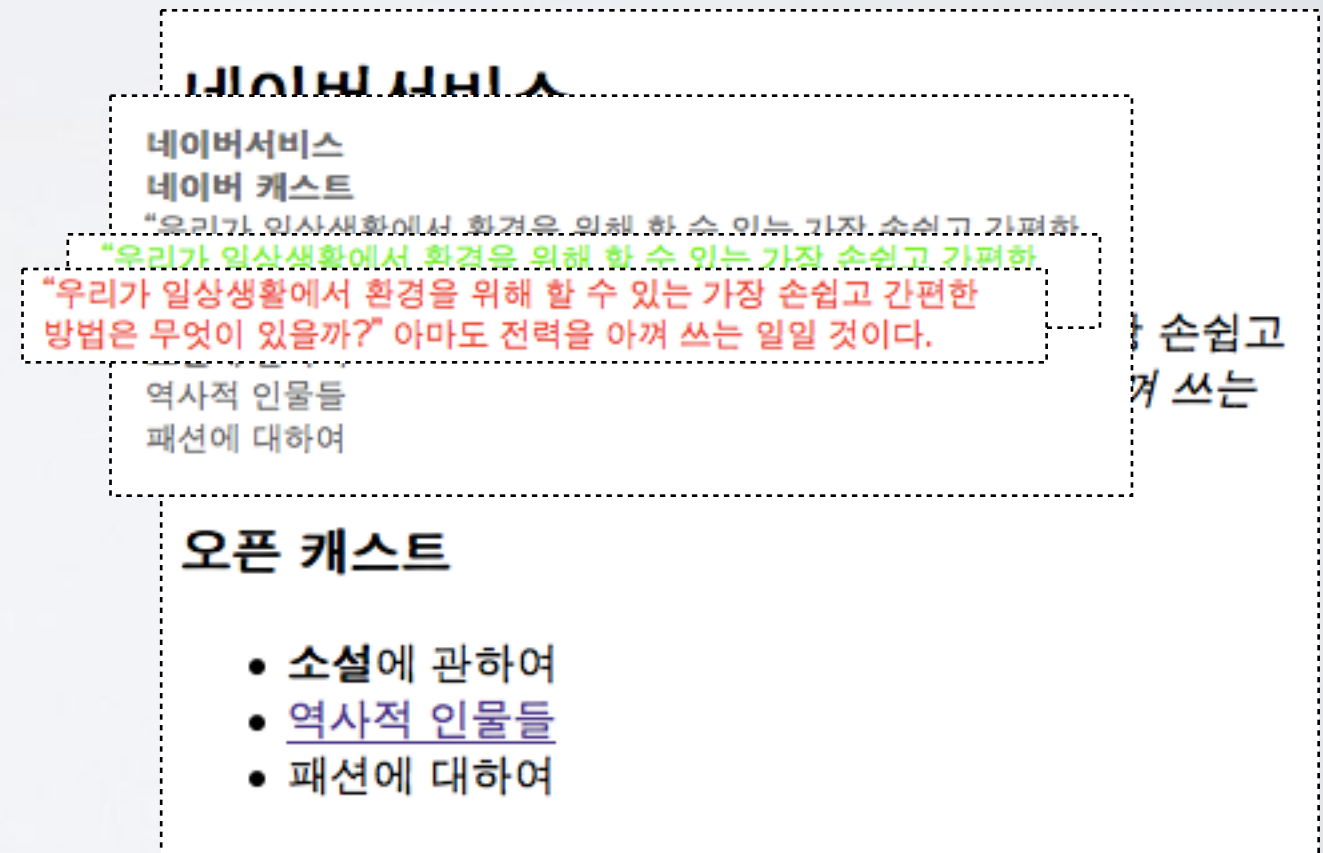
- 소설에 관하여
- 역사적 인물들
- 패션에 대하여

4위. 직접선택한 요소가 같을 경우 아래코드가 우선합니다.

```
<html>
<head>
  <meta charset="utf-8"/>
  <title>네이버서비스</title>
  <link href="css/reset.css"...중략 />
  <link href="css/basic.css"...중략 />
  <style type="text/css">
    p {color:red;} 001
  </style>
</head>

<body>
<div>
  <h2>네이버서비스</h2>
  <div>
    <h3>네이버 캐스트</h3>
    <p><q>우리가 일상생활에서 환경을 위해
    할 수 있는 가장..중략</p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <ul>
      <li><strong>소설</strong>에 관하여</li>
      <li><a href="">역사적 인물들</a></li>
      <li>패션에 대하여</li>
    </ul>
  </div>
</div>
</body>
</html>
```

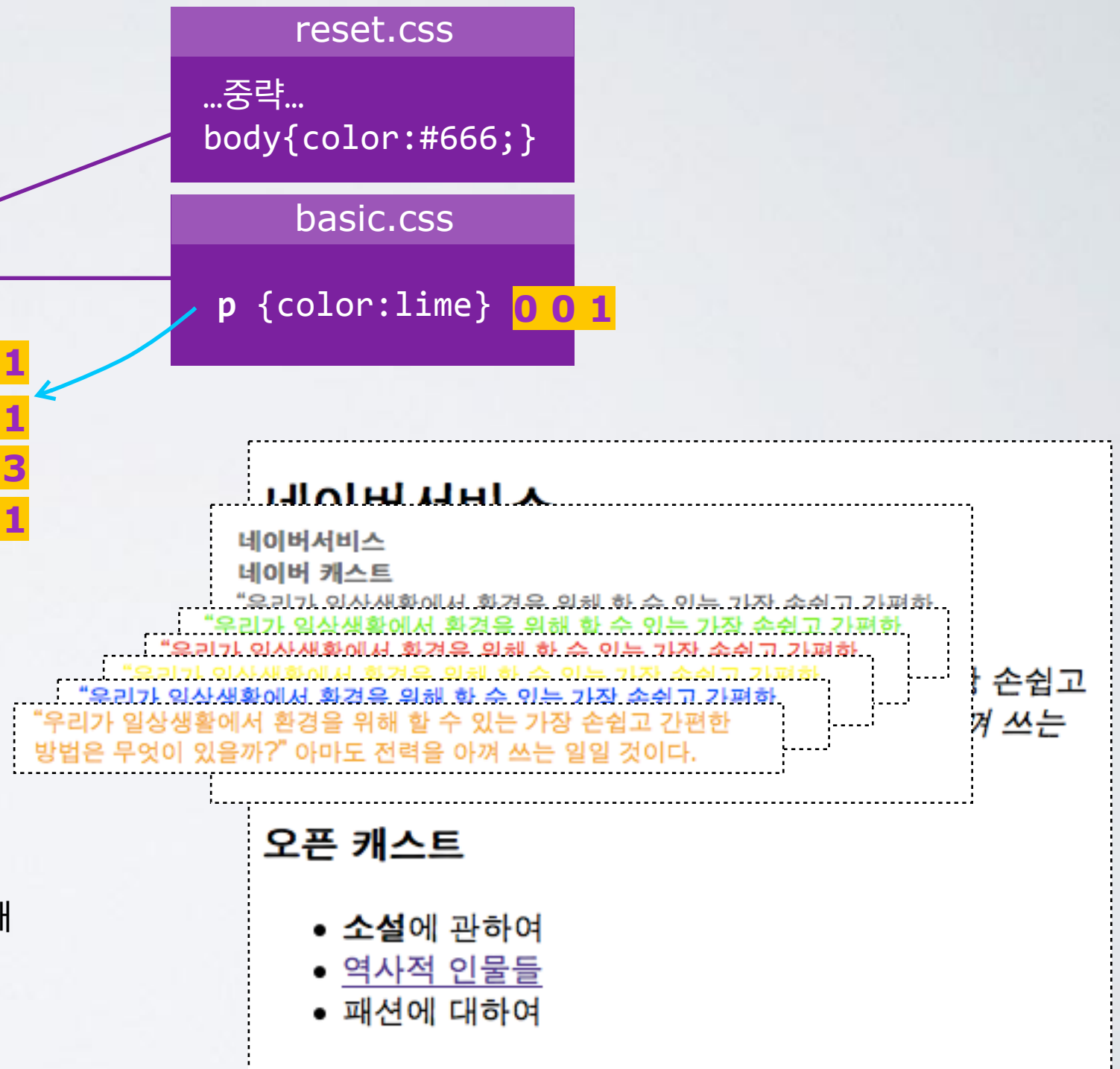
태그이름으로 같은 요소를
선택했기 때문에 아래요소가
우선한다.



3위. 아무리 위에 있더라도 구체적으로 부르면 우선합니다.

```
<html>
<head>
  <meta charset="utf-8"/>
  <title>네이버서비스</title>
  <link href="css/reset.css"...중략 />
  <link href="css/basic.css"...중략 />
  <style type="text/css">
    p {color:red;}
    #service p {color:orange;}
    div div > p {color:yellow;}
    .castbox p {color:blue;}
  </style>
</head>

<body>
<div>
  <h2>네이버서비스</h2>
  <div>
    <h3>네이버 캐스트</h3>
    <p><q>우리가 일상생활에서 환경을 위해
    할 수 있는 가장..중략</p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <ul>
      <li><strong>소설</strong>에 관하여</li>
```




```

<html>
<head>
  <meta charset="utf-8"/>
  <title>네이버서비스</title>
  <link href="css/reset.css"...중략 />
  <link href="css/basic.css"...중략 />
  <style type="text/css">
    p {color:red;}
    #service p {color:orange;}
    div div > p {color:yellow;}
    .castbox p {color:blue;}
  </style>
</head>

<body>
<div>
  <h2>네이버서비스</h2>
  <div>
    <h3>네이버 캐스트</h3>
    <p><q>우리가 일상생활에서 환경을 위해
    할 수 있는 가장..중략</p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <ul>
      <li><strong>소셜</strong>에 관하여</li>
      <li><a href="">역사적 인물들</a></li>
      <li>패션에 대하여</li>
    </ul>
  </div>
</div>

```

5	0	0	1
2	1	0	1
4	0	0	3
3	0	1	1

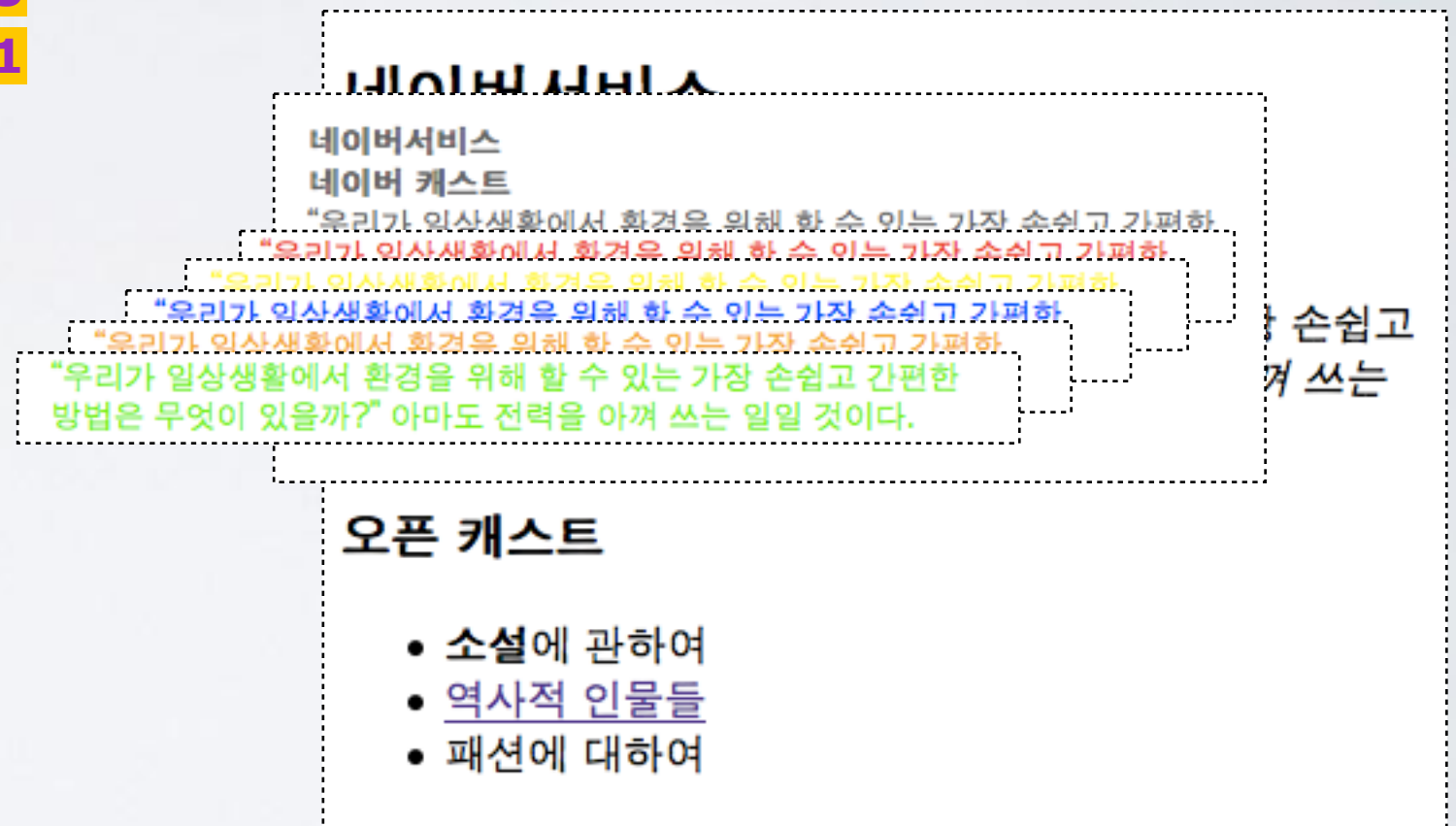
reset.css

...중략...
body{color:#666;}

basic.css

#service div p {
color:lime
}

구체성이 높아지면서 하향
우선법칙이 무시됩니다.



2위. 아무리 구체성이 높더라도 인라인스타일이 우선합니다.

```
<html>
<head>
  <meta charset="utf-8"/>
  <title>네이버서비스</title>
  <link href="css/reset.css" ...중략 />
  <link href="css/basic.css" ...중략 />
  <style type="text/css">
    p {color:red;}
    #service p {color:orange;}
  </style>
</head>

<body>
<div>
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p style="color:blue"><q>우리가 일상생활에
    서 환경을 위해 할 수 있는 가장..중략</p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
    <ul>
```

reset.css

...중략...
body{color:#666;}

basic.css

p {color:lime} 0 0 1

네이버서비스

네이버서비스 네이버 캐스트

"우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한
방법은 무엇이 있을까?" 아마도 전력을 아껴 쓰는 일일 것이다.

오픈 캐스트

- 소설에 관하여
- 역사적 인물들
- 패션에 대하여

손쉽고
껴 쓰는

1위. !important는 어떤것 보다 우선합니다.

```
<html>
<head>
  <meta charset="utf-8"/>
  <title>네이버서비스</title>
  <link href="css/reset.css" ...중략 />
  <link href="css/basic.css" ...중략 />
  <style type="text/css">
    p {color:red !important;}
    #service p {color:orange;} 1 0 1
  </style>
</head>

<body>
<div>
  <h2>네이버서비스</h2>
  <div class="castbox">
    <h3>네이버 캐스트</h3>
    <p style="color:blue"><q>우리가 일상생활에
      서 환경을 위해 할 수 있는 가장..중략</p>
  </div>
  <div>
    <h3>오픈 캐스트</h3>
```

reset.css

..중략...
body{color:#666;}

basic.css

p {color:lime} 0 0 1

네이버서비스

네이버서비스
네이버 캐스트

"우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한

"우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한

"우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한

"우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한

"우리가 일상생활에서 환경을 위해 할 수 있는 가장 손쉽고 간편한
방법은 무엇이 있을까?" 아마도 전력을 아껴 쓰는 일일 것이다.

패션에 대하여

오픈 캐스트

- 소설에 관하여
- 역사적 인물들
- 패션에 대하여

손쉽고
계 쓰는



eulsoo.com
eulsoo.jung@gmail.com



이 저작물은 크리에이티브 커먼즈 저작자표시-변경금지 4.0 국제 라이선스에 따라 이용할 수 있습니다.