



스타일이란

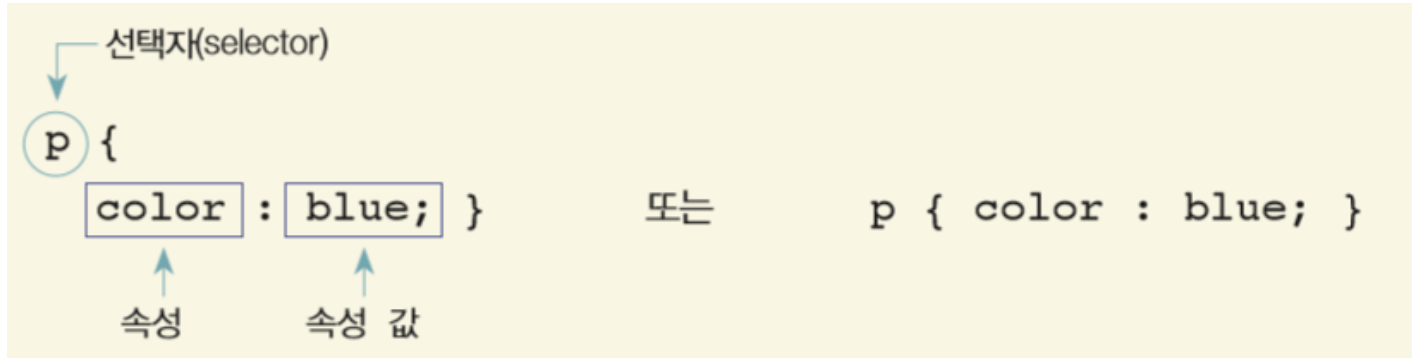
- **스타일(style)이란** : HTML 문서에서 자주 사용하는 글꼴이나 색상, 정렬, 각 요소들의 배치 방법 등 문서의 겉모습을 결정짓는 내용들
예) 텍스트 단락의 줄 간격, 표의 점선 테두리
- **스타일시트(styleshet)** : 스타일을 관리하기 쉽도록 한 군데 모아놓은 것

왜 스타일을 사용해야 할까?

- 1) 웹 문서의 디자인과 내용을 분리한다 : 내용은 HTML을 이용해 구성 하고, 디자인은 CSS를 이용해 꾸미기 때문에, 사이트의 내용을 수정해야 할 경우 디자인에 전혀 영향을 주지 않고 내용만 수정할 수 있다.
- 2) 다양한 매체에 적합한 문서를 만들 수 있다 : HTML로 작성된 내용은 그대로 두고 대상 기기에 맞게 CSS만 바꿔 주면 같은 내용을 여러 매체에서 사용할 수 있다.



스타일의 형식



- 선택자(selector) : 앞으로 만들 스타일 규칙이 어디에 적용될 것인가를 알려준다.
- 속성과 속성 값 : '속성 : 속성 값'과 같은 형식으로 함께 표시하며, 속성/속성 값 쌍이 여럿일 경우에는 세미콜론(;)으로 구분한다.

예)

```
p { color:blue; }  
h2 { font-size:20px; color:purple; }
```



스타일 적용 방법

태그에 직접 적용

Html 태그에 style 속성을 이용하여 적용

Ex) `<div style="background : #ff0000;">`

문서 내부에 적용

작성중인 Html 문서 내부에 `<style>` 태그를 이용하여 적용

Ex) `<style> div { background : #ff0000 } </style>`

외부 문서를 삽입

`<link href="외부 CSS 파일 경로" rel="stylesheet" type="text/ css">`



스타일 적용 방법 - 1 (태그 직접 적용)

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습</title>
</head>
<body>
  <div style="background:red;">
    첫번째 DIV
  </div>
  <div style="background:blue;">
    두번째 DIV
  </div>
</body>
</html>
```

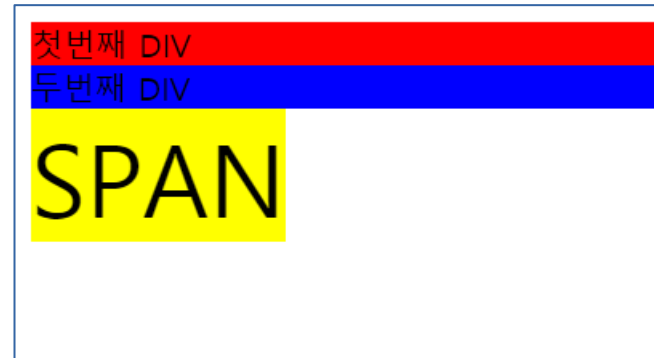
첫번째 DIV

두번째 DIV



스타일 적용 방법 – 2 (문서 내부 적용)

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습</title>
  <style>
    span {
      background : yellow;
      font-size : 50px;
    }
  </style>
</head>
<body>
  <div style="background:red;">첫 번째 DIV</div>
  <div style="background:blue;">두 번째 DIV</div>
  <span>SPAN</span>
</body>
</html>
```





스타일 적용 방법 – 3 (외부 문서 삽입)

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습</title>
  <link rel="stylesheet" href="external.css">
  <style>
    span {
      background : yellow; font-size : 50px;
    }
  </style>
</head>
<body>
  <div style="background:red;">첫번째 DIV</div>
  <div style="background:blue;">두번째 DIV</div>
  <span>SPAN</span>
  <a>A태그</a>
</body>
</html>
```

```
/* external.css */

a {
  cursor : move;
}
```





스타일의 우선 순위

스타일은 상속된다

CSS에서는 부모 요소에 있는 속성들이 자식 요소로 전달된다.

예제 Samples\6장\inherit.html

```
<style>
ul{
  color:blue; /* 글자색은 파랑 */
  list-style:square; /* 목록 항목은 square 모양으로 */
}
</style>
```

ul 태그의 스타일시트

```
<body>
<section>
  <h2>예약 방법 및 요금</h2>
  <p>요안도라 예약의 가장 빠른 방법은?</p>
  <ul>
    <li>예약 방법</li>
    <ul>
      <li>직접 통화</li>
      <li>문자 남기기</li>
    </ul>
    <li>요금</li>
    <ul>
      <li>1인 : 40,000원</li>
      <li>2인 : 60,000원</li>
      <li>3인 : 80,000원</li>
      <li>4인 : 100,000원</li>
      <li>4인 이상일 경우 상의해 주세요.</li>
    </ul>
  </ul>
</section>
</body>
```

ul 태그의 스타일이 li 태그에도 상속됨

구현된 결과

예약 방법 및 요금

요안도라 예약의 가장 빠른 방법은?

- 예약 방법
 - 직접 통화
 - 문자 남기기
- 요금
 - 1인 : 40,000원
 - 2인 : 60,000원
 - 3인 : 80,000원
 - 4인 : 100,000원
 - 4인 이상일 경우 상의해 주세요.



스타일의 우선 순위

스타일은 단계적으로 적용된다

하나의 요소에 여러 스타일 규칙이 정의되어 있다면 가장 나중에 정의된 스타일을 적용한다.

스타일이 중복 되면 마지막 스타일이 적용된다.

스타일 규칙에 !important를 추가하면 해당 스타일 규칙이 가장 높은 우선순위를 갖게 된다.

```
<style>
.accent {color:red;}
.accent {color:blue;}
</style>

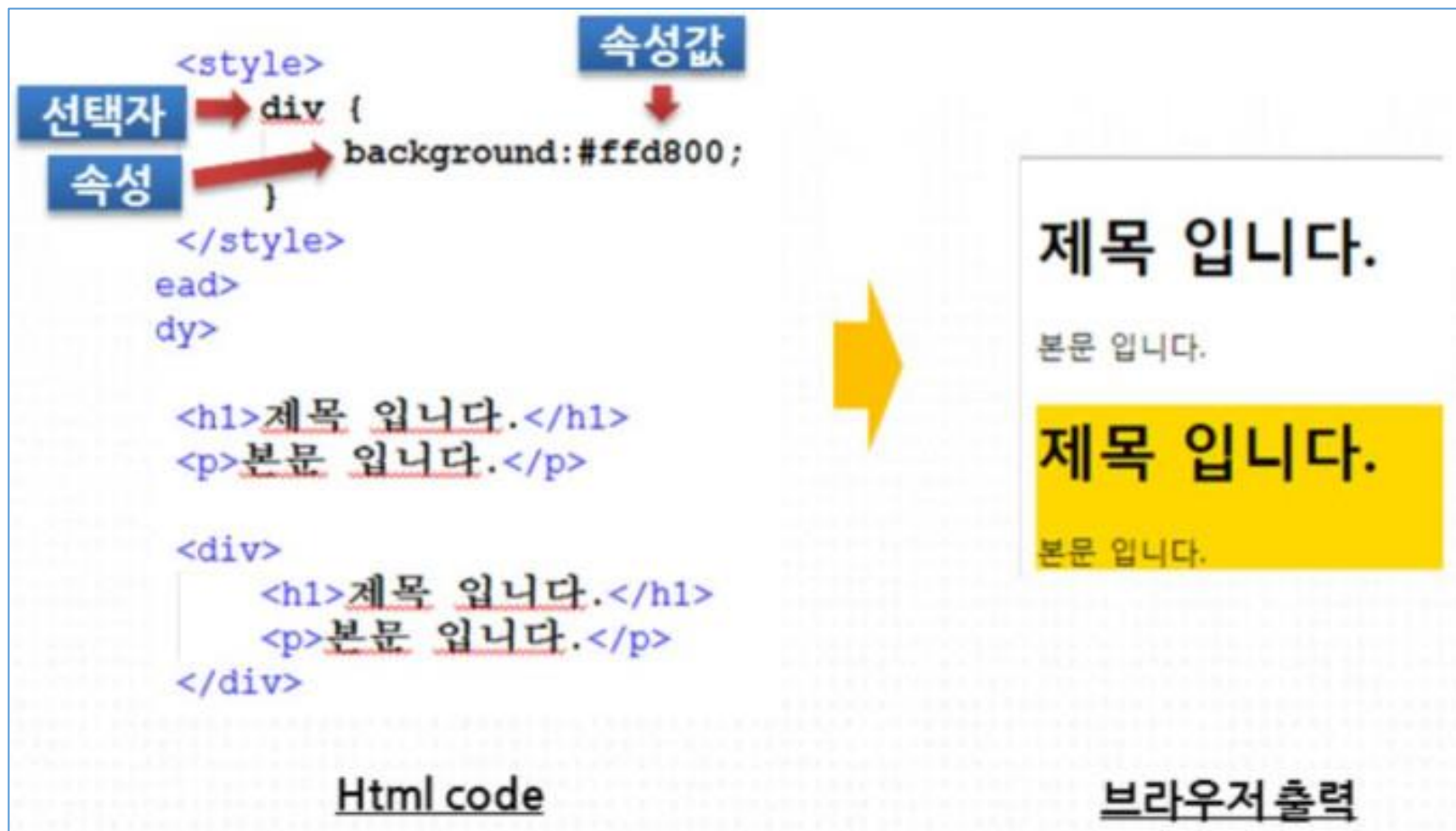
<p class="accent">성산의 날씨는 다음주 내내 높은 구름에 햇살 가득이라고 합니다.</p>
```

성산의 날씨는 다음주 내내 높은 구름에 햇살 가득이라고 합니다.

```
<style>
.accent {color:red; !important;}
.accent {color:blue;}
</style>
```

성산의 날씨는 다음주 내내 높은 구름에 햇살 가득이라고 합니다.

선택자와 속성





전체 선택자, 태그 선택자

선택자, 셀렉터(selector)

스타일을 적용하기 위해 특정한 부분을 선택하는 방법

전체 선택자

- 페이지에 있는 모든 요소를 대상으로 스타일을 적용할 때 사용
- 다른 선택자와 함께 모든 하위 요소에 한꺼번에 스타일을 적용하려고 할 때 주로 사용

예) `* { margin:0; padding:0;}`

태그 선택자

- 문서 안의 특정 태그에 스타일이 모두 적용됨

예) `p { font-size:12px; font-family:"돋움"; }`



태그 선택자

예제 Samples\6장\css2.html

```
<head>
```

```
<style>
```

```
h2 {  
  color:blue;  
}
```

① <h2> 태그가 적용된 부분의 텍스트 색상을 파란색으로 지정

```
p {  
  font-size:12px; /* 글자 크기 12px */  
  color:#ff0000; /* 글자색 : 빨강(##0000) */  
  margin-left:20px; /* 왼쪽 여백 : 20px */  
}
```

② <p> 태그가 적용되는 곳에 스타일 지정

```
</style>
```

```
</head>
```

```
<body>
```

```
<section>
```

<h2> 태그 스타일 적용

```
<h2>요안도라 소개</h2>
```

<p> 태그 스타일 적용

```
<h3>ㄹ 요안도라는 게스트 하우스(Guest House) 형식의 농어촌 민박입니다</h3>
```

```
<p>성산의 날씨는 다음주 내내 높은 구름에 햇살 가득이라고 합니다.</p>
```

```
<p>목요일이면 섭씨 27도까지 오른다고 하지만,</p>
```

```
<p>늘 부는 바람이 쾌적한 균형을 잡아 마당에 마당에 나가 앉아 있는 시간이  
많아질 듯합니다. </p>
```

```
<p>오늘은 사진에 보이는 긴 돌담을 따라 들어오는 요안도레 올레 입구에 특색의  
대문을 달았습니다.</p>
```

```
</section>
```

```
</body>
```

요안도라 소개 ①

ㄹ 요안도라는 게스트 하우스(Guest House) 형식의 농어촌 민박입니다

성산의 날씨는 다음주 내내 높은 구름에 햇살 가득이라고 합니다.

목요일이면 섭씨 27도까지 오른다고 하지만,

늘 부는 바람이 쾌적한 균형을 잡아 마당에 마당에 나가 앉아 있는 시간이 많아질 듯합니다.

오늘은 사진에 보이는 긴 돌담을 따라 들어오는 요안도레 올레 입구에 특색의 대문을 달았습니다.

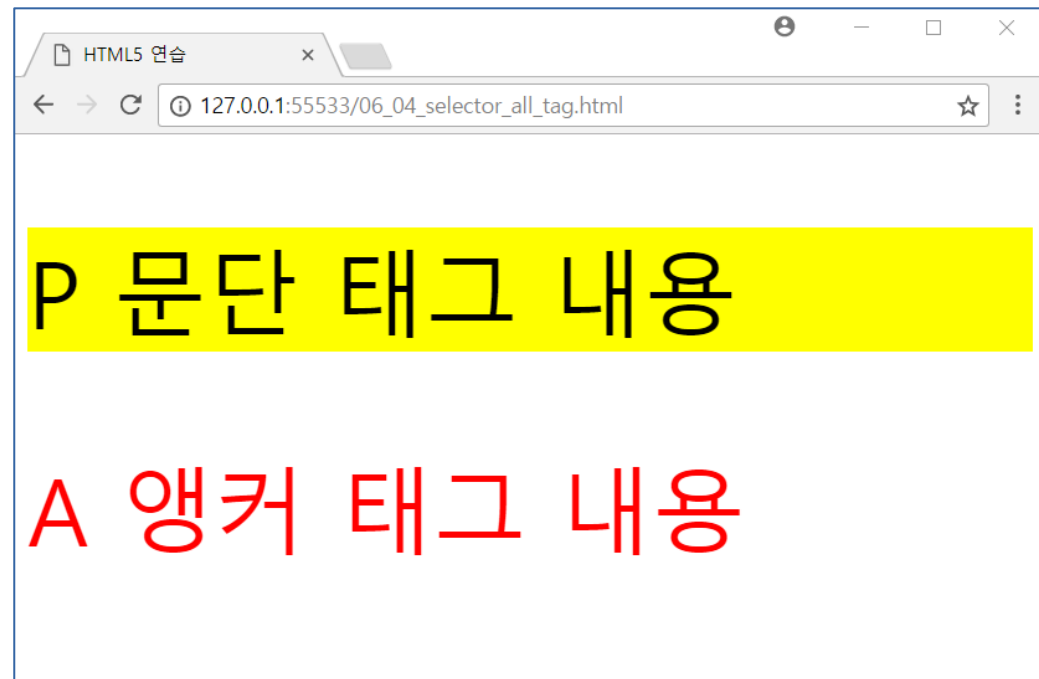
태그 선택자를 적용한 결과

전체 선택자, 태그 선택자

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습</title>
  <style>
    /* 전체 선택자 */
    * {
      font-size : 60px;
    }

    /* 태그 선택자 */
    p {
      background-color : yellow;
    }
    a {
      color : red;
    }
  </style>
</head>
```

```
<body>
  <p>P 문단 태그 내용</p>
  <a>A 앵커 태그 내용</a>
</body>
</html>
```





class 선택자, id 선택자

- 공통점 : 요소의 특정 부분에만 스타일 적용
- 차이점
 - 클래스 선택자 : 문서 안에서 여러 번 반복할 스타일이라면 클래스 선택자로 정의.
마침표(.) 다음에 클래스 이름 지정
 - id 선택자 : 문서 안에서 한번만 사용한다면 id 선택자로 정의.
파운드(#) 다음에 id 이름 지정

예)

```
.redtext { color:red; }
```

```
#pic2 { clear:both; float:left; }
```



class 선택자

예제_ Samples\6장\class.html

```
<style>
```

```
p{
  font-size:12px; /* 글자 크기 12px */
  margin-left:20px; /* 왼쪽 여백 20px */
}
```

① <p> 태그 선택자 지정, <p> 태그를 사용한 부분에 스타일 적용

```
.accent {
  color:blue; /* 글자색 파랑(blue) */
  font-weight:bold; /* 글자 굵게(bold) */
}
```

② .accent 클래스 선택자 스타일 지정, class="accent"라고 사용한 부분에만 스타일 적용

```
</style>
```

```
</head>
```

```
<body>
```

```
<section>
```

```
<h2>요안도라 소개</h2>
```

<p> 태그 단락에 accent 클래스 선택자 스타일 적용

```
<h3>☞ 요안도라는 게스트 하우스(Guest House) 형식의 농어촌 민박입니다</h3>
```

```
<p class="accent">성산의 날씨는 다음주 내내 높은 구름에 햇살 가득이라고  
합니다.</p>
```

```
<p>목요일이면 섭씨 27도까지 오른다고 하지만, </p>
```

```
<p>늘 부는 바람이 쾌적한 균형을 잡아 마당에 마당에 나가 앉아 있는 시간이 많아질  
듯합니다. </p>
```

```
<p>오늘은 사진에 보이는 긴 돌담을 따라 들어오는 <span class="accent">
```

```
요안도레 올레 입구</span>에 특근색의 대문을 달았습니다.</p>
```

```
</section>
```

```
</body>
```

 태그를 사용한 '요안도레 올레 입구'에 accent 클래스 선택자 스타일 적용

구현된 결과





id 선택자

예제 Samples\6장\id.html

<style>

```
#title{
  clear:both; /* 이전 요소와 줄바꿈서 표시 */
  float:left; /* 왼쪽으로 플로팅 */
  margin-left:10px; /* 왼쪽 여백 10px */
  width:200px; /* 너비 200px */
  height:110px; /* 높이 110px */
  background-color:orange; /* 배경색 오렌지색 */
  border:1px solid black; /* 테두리 1px짜리 검은색 실선 */
  text-align:center; /* 텍스트 가운데 정렬 */
}
```

title id 선택자
스타일 지정

```
#pic1{
  clear:none; /* 이전 요소와 연결해서 표시 */
  float:left; /* 왼쪽으로 플로팅 */
  margin-left:10px; /* 왼쪽 여백 10px */
  width:200px; /* 너비 200px */
  height:110px; /* 높이 110px */
}
```

pic1 id 선택자
스타일 지정

```
#pic2{
  clear:both; /* 이전 요소와 줄바꿈서 표시 */
  float:left; /* 왼쪽으로 플로팅 */
  margin-left:10px; /* 왼쪽 여백 10px */
  margin-top:20px; /* 위쪽 여백 20px */
  width:200px; /* 너비 200px */
  height:120px; /* 높이 120px */
}
```

pic2 id 선택자
스타일 지정

```
#pic3{
  clear:none; /* 이전 요소와 연결해서 표시 */
  float:left; /* 왼쪽으로 플로팅 */
  margin-left:10px; /* 왼쪽 여백 10px */
  margin-top:20px; /* 위쪽 여백 20px */
  width:200px; /* 너비 200px */
  height:120px; /* 높이 120px */
}
```

pic3 id 선택자
스타일 지정

id 선택자

```
</style>
```

```
</head>
```

```
<body>
```

```
<div id="title"> <h2>YOANDORA</h2> </div>
```

1 title 스타일 적용

```
<div id="pic1">  </div>
```

2 pic1 스타일 적용

```
<div id="pic2">  </div>
```

3 pic2 스타일 적용

```
<div id="pic3">  </div>
```

```
</body>
```

4 pic3 스타일 적용

구현된 결과



class 선택자, id 선택자

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습</title>
  <style>
    /* 아이디 선택자 */
    #first { background : red; }
    #second { background : blue; }

    /* 클래스 선택자 */
    span.title {
      background : yellow;
      font-size : 50px;
    }
    span.desc {
      background : gray;
      font-size : 150px;
    }
  </style>
</head>
```

```
<body>
  <div id="first">
    첫째 DIV
  </div>
  <div id="second">
    둘째 DIV
  </div>
  <span class="title">
    SPAN1
  </span>
  <span class="desc">
    SPAN2
  </span>
</body>
</html>
```





속성 선택자

표기	설명
[속성]	지정한 '속성'을 가지고 있는 요소를 찾아 스타일을 적용
[속성 = 값]	'속성'의 '값'이 일치하는 요소를 찾아 스타일을 적용
[속성 ~= 값]	'속성'과 '값'을 체크해 여러 개의 값 중 하나만 일치해도 스타일을 적용
[속성 ^= 값]	'속성'의 '값'이 지정한 문자로 시작하는 속성값에 대해서만 스타일을 적용
[속성 \$= 값]	'속성'의 '값'이 지정한 문자로 끝나는 속성에 대해서만 스타일을 적용
[속성 *= 값]	속성 값 중에 '값'의 일부가 포함되어 있는 속성에 스타일을 지정

속성 선택자

```
<style>
```

```
input {  
  color:green;  
  font-size:30px;  
  font-weight:bold;  
}
```

```
input[type=text] {  
  color:orange;  
  font-size:50px;  
  width:200px;  
}
```

```
input[type=tel] {  
  color:red;  
}
```

```
img[src] {  
  border:5px solid green;  
}
```

```
</style>
```

이름 :

홍길동

아이디 :

abcde

비밀번호 :

●●●●●

전화번호 :

01012345678

꿈+Story

속성 선택자 [속성] 형식

예제 Samples\6장\attr1.html

```
<style>
ul li { /* ul 요소 아래의 모든 li 요소 */
    font-size:16px; /* 글자 크기 16px */
    line-height:25px; /* 줄간격 25px */
}
a[href] {
    background:yellow; /* 배경색 노랑 */
}
</style>
</head>
```

〈a〉 태그 중 href 속성이 있는 곳에 스타일 지정

```
<body>
<section id="container">
<header><h1>객실 소개</h1></header>
<h3>객실 안내</h3>
<ul>
<li><a href="one.html"><b>2인실(유채방, 동백방)</b></a>
동성 또는 가족 2인 공동 취사실 및 욕실(화장실) 사용</li>
<li><a href="two.html"><b>4인실(펜션)</b></a>
취사 가구 완비, 실내 욕실(화장실)</li>
</ul>
<h3>객실 요금</h3>
```

〈a〉 태그 중 href 속성이 있는 요소에 스타일 적용

구현된 결과

객실 소개

객실 안내

- 2인실(유채방, 동백방) 동성 또는 가족 2인, 공동 취사실 및 욕실(화장실) 사용
- 4인실(펜션) 취사 가구 완비, 실내 욕실(화장실)

객실 요금

- 1인 : 40,000원
- 2인 : 60,000원
- 3인 : 80,000원
- 4인 : 100,000원



속성 선택자 [속성=값] 형식

예제_ Samples\6장\attr2.html

```
<style>
input[type="text"] {
    width:150px; /* 너비 150px */
    background-color:skyblue; /* 배경색 하늘색 */
    border:1px solid #000; /* 1px짜리 검은색 실선 */
}
</style>
```

⟨input⟩ 태그 중 type="text"인 input 요소에 스타일 지정

```
<ul>
<li>
    <label for="name">예약자</label>
    <input type="text" name="name" id="name">
</li>
```

① ⟨input⟩ 태그 중 type 속성 값이 "text"인 요소에 스타일 적용

```
<li>
    <label for="bank">입금 은행</label>
    <input type="text" name="bank">
</li>
</ul>
</fieldset>
```

② ⟨input⟩ 태그 중 type 속성 값이 "text"인 요소에 스타일 적용

구현된 결과

예약하기

예약자 정보

예약자 ①
연락처
예약 인원
입금 은행 ②



속성 선택자 [속성~=값] 형식

예제 Samples\6장\attr3.html

```
<style>
```

```
h4[title ~= "4인실"] {  
    font-style:italic; /* 글자를 이탤릭체로 */  
}
```

title 속성에 "4인실" 값이 있는
(h4) 태그에 적용할 스타일 지정

```
h4[title ~= "욕실"] {  
    color:red; /* 글자색 빨강 */  
    bold-weight:bold; /* 글자 굵게 */  
}
```

title 속성에 "욕실" 값이 있는 (h4)
태그에 적용할 스타일 지정

```
</style>
```

구현된 결과

객실 소개

객실 안내

남자 도미토리 ①

공동 취사실 및 욕실(화장실) 사용
1인 20,000원(비수기), 25,000원(성수기)

여자 도미토리 ②

취사 가구 완비, 실내 욕실(화장실)
1인 30,000원(비수기), 40,000원(성수기)

펜션 ③

취사 가구 완비, 실내 욕실(화장실)
1박 100,000원(비수기), 150,000원(성수기)

```
<h3> 객실 안내</h3>
```

```
<h4 title="4인실">남자 도미토리</h4>
```

① title 속성이 "4인실"인 선택자 적용

```
<ul>
```

```
<li>공동 취사실 및 욕실(화장실) 사용</li>
```

```
<li>1인 20,000원(비수기), 25,000원(성수기)</li>
```

```
</ul>
```

```
<h4 title="욕실">여자 도미토리</h4>
```

② title 속성이 "실내"이거나
"욕실"인 선택자 적용

```
<ul>
```

```
<li>취사 가구 완비, 실내 욕실(화장실)</li>
```

```
<li>1인 30,000원(비수기), 40,000원(성수기)</li>
```

```
</ul>
```

```
<h4 title="4인실 욕실">펜션</h4>
```

③ title 속성이 "4인실"이거나
"실내", "욕실"인 선택자 적용

```
<ul>
```

```
<li>취사 가구 완비, 실내 욕실(화장실)</li>
```

```
<li>1박 100,000원(비수기), 150,000원(성수기)</li>
```

```
</ul>
```



속성 선택자 [속성^=값] 형식

예제_ Samples\6장\attr4.html

```
<style>
```

```
h4[title ^= "남자"] {  
  background :url(man.png) no-repeat; /* man.png를 배경 이미지로 */  
  padding: 10px 30px; /* h4 요소의 패딩(안 여백) 상하 10px 좌우 30px */  
}
```

title 속성 값이 "남자"로 시작하는 <h4> 태그에 적용할 스타일 지정

```
h4[title ^= "여자"] {  
  background :url(woman.png) no-repeat; /* woman.png를 배경 이미지로 */  
  padding: 10px 30px; /* h4 요소의 패딩(안 여백) 상하 10px 좌우 30px */  
}
```

```
</style>
```

```
</head>
```

title 속성 값이 "여자"로 시작하는 <h4> 태그에 적용할 스타일 지정

```
<h3> 객실 안내</h3>
```

```
<h4 title="남자 도미토리">남자 도미토리</h4>
```

1 title 속성 값이 "남자"로 시작하는 <h4> 태그

```
<ul>
```

```
<li>공동 취사실 및 욕실(화장실) 사용</li>
```

```
<li>1인 20,000원(비수기), 25,000원(성수기)</li>
```

```
</ul>
```

```
<h4 title="여자 도미토리">여자 도미토리</h4>
```

2 title 속성 값이 "여자"로 시작하는 <h4> 태그

```
<ul>
```

구현된 결과

객실 소개

객실 안내

남자 도미토리 1

공동 취사실 및 욕실(화장실) 사용
1인 20,000원(비수기), 25,000원(성수기)

여자 도미토리 2

취사 거구 준비, 실내 욕실(화장실)
1인 30,000원(비수기), 40,000원(성수기)



속성 선택자 [속성\$=값] 형식

예제_ Samples\6장\attr5.html

```
<style>
```

```
[href $= "hwp"] {  
    background:url(hwp_icon.gif) center right no-repeat;  
    /* 배경으로 hwp 아이콘 표시 */  
    padding-right:25px; /* 아이콘을 표시할 수 있도록 오른쪽에 25px 여백 */  
}
```

연결한 파일의 확장자가 hwp인 링크에 적용할 스타일 지정

```
a[href $= "xls"] {  
    background:url(excel_icon.gif) center right no-repeat;  
    /* 배경으로 excel 아이콘 표시 */  
    padding-right:25px; /* 아이콘을 표시할 수 있도록 오른쪽에 25px 여백 */  
}
```

```
</style>
```

```
</head>
```

연결한 파일의 확장자가 xls인 링크에 적용할 스타일 지정

```
<ul>
```

```
<li><a href="intro.hwp">hwp 파일</a></li>
```

```
<li><a href="intro.xls">엑셀 파일</a></li>
```

```
</ul>
```

연결 파일의 확장자가
hwp와 xls인 링크

구현된 결과

객실 요금 및 사진 다운 받기

- [hwp 파일](#) 
- [엑셀 파일](#) 



속성 선택자 [속성*=값] 형식

예제_ Samples\6장\attr7.html

```
<style>
  li[title *= "2011"] {
    font-style:italic; /* 글자는 이탤릭체로 */
    font-weight:700; /* 글자 조금 굵게 */
  }
  a[hreflang |= "en"] {
    background:url(en.jpg) center right no-repeat;
                                /* 오른쪽에 en.jpg 표시 */

    padding-right:30px; /* 배경 그림과 내용 사이의 여백 */
  }
</style>
</head>
```

hreflang 속성이 en일 경우 적용할 스타일 지정

```
<header><h1>CSS 참고 사이트</h1></header>
<ul>
  ① <li title="last update:2011-11">
  ② <a href="#" hreflang="en">사이트1</a></li>
    <li title="last update:2010-01">
  ③ <a href="#" hreflang="en">사이트2</a></li>
    <li title="last update:2010-08">
    <a href="#" hreflang="ko">사이트3</a></li>
  ④ <li title="last update:2011-01">
    <a href="#" hreflang="fr">사이트4</a></li>
</ul>
```

title 속성 값에 2011 문자열이 있는 요소에 스타일 적용(이탤릭 체)

hreflang 속성이 en으로 시작하는 요소에 스타일 적용(en.png 이미지)

구현된 결과

CSS 참고 사이트

사이트1  ①, ②
사이트2  ③
사이트3
사이트4  ④



속성 선택자

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습</title>
  <style>
    h5[title] { background-color : red; }
    h5[class~=header] { background-color : yellow; }
    h5[class^=con] { background-color : blue; }
    h5[class$=esc] { background-color : orange; }
    h5[class*=tiv] { background-color : purple; }

    input[type=text] { width : 50px; }
    input[type=password] { background-color : red; }
    input[type=tel] { font-size : 30px; }
    input[type=search] { font-style : italic; }
  </style>
</head>
```

```
<body>
  <h5>h5 제목</h5>
  <h5 title="제목1">h5 제목 1</h5>
  <h5 class="title header">h5 제목 2</h5>
  <h5 class="content">h5 제목 3</h5>
  <h5 class="desc">h5 제목 4</h5>
  <h5 class="active">h5 제목 5</h5>

  <input type="text"> <br>
  <input type="password"> <br>
  <input type="tel"> <br>
  <input type="search">
</body>
</html>
```

하위 선택자(descendant selector)

- 부모 요소에 포함된 모든 하위 요소에 스타일이 적용된다
- 하위 선택자를 정의할 때는 상위 요소와 하위 요소를 나란히 쓴다.

상위 요소 하위 요소 {속성:속성 값; ...}

기본형

예) `body p { color:blue; }`

```
1  <!doctype html>
2  <html>
3
4  <body>
5      <p>CSS 선택자</p>
6      <div>
7          <h1>제목</h1>
8          <ul>
9              <li>1번</li>
10             <li>2번</li>
11          </ul>
12          <p>내용</p>
13      </div>
14 </body>
15
16 </html>
```

자식 선택자(child selector)

부모 요소의 자식 요소에만 스타일이 적용된다.

부모 요소 > 자식 요소 {속성:속성 값;}

기본형

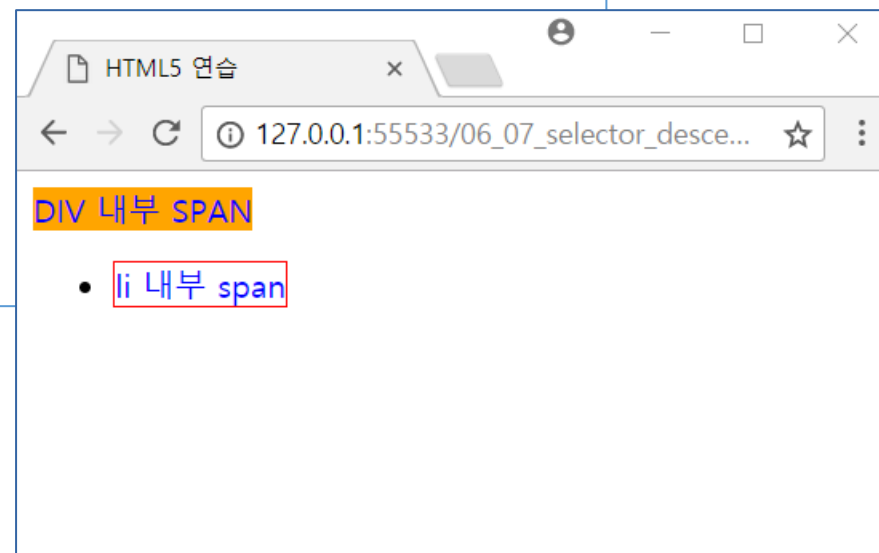
예) `body > p { color:blue; }`

```
1  <!doctype html>
2  <html>
3
4  <body>
5      <p>CSS 선택자</p>
6      <div>
7          <h1>제목</h1>
8          <ul>
9              <li>1번</li>
10             <li>2번</li>
11         </ul>
12         <p>내용</p>
13     </div>
14 </body>
15
16 </html>
```

하위 및 자식 선택자 - 1

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습 </title>
  <style>
    div > span {
      background : orange;
    }
    div span {
      color : blue;
    }
    li > span {
      border : 1px solid red;
    }
  </style>
</head>
```

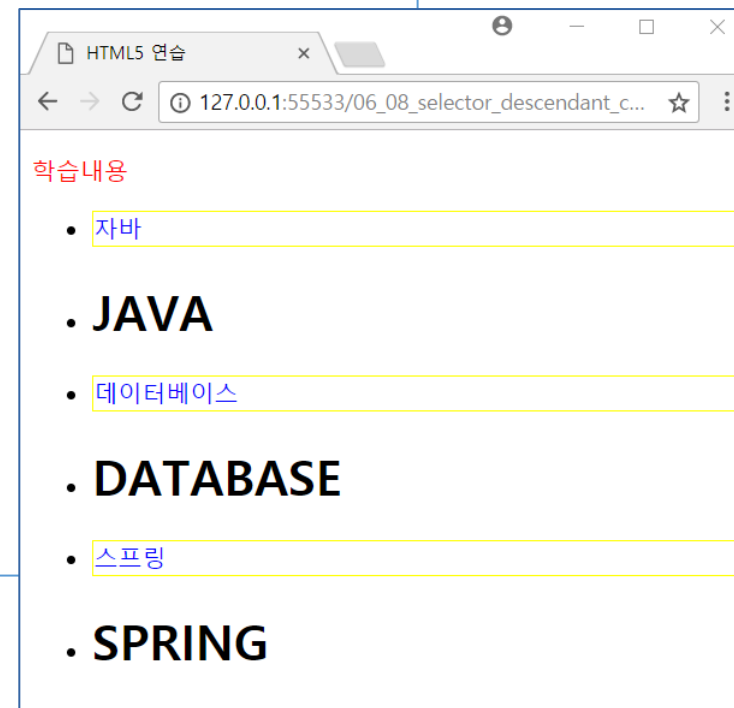
```
<body>
  <div>
    <span>DIV 내부 SPAN</span>
    <ul>
      <li>
        <span>li 내부 span</span>
      </li>
    </ul>
  </div>
</body>
</html>
```



하위 및 자식 선택자 - 2

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습 </title>
  <style>
    div p {
      color : red;
    }
    li > p {
      color : blue;
      border : 1px solid yellow;
    }
  </style>
</head>
```

```
<body>
  <div>
    <p> 학습내용 </p>
    <ul>
      <li> <p> 자바 </p> </li>
      <li> <h1> JAVA </h1> </li>
      <li> <p> 데이터베이스 </p> </li>
      <li> <h1> DATABASE </h1> </li>
      <li> <p> 스프링 </p> </li>
      <li> <h1> SPRING </h1> </li>
    </ul>
  </div>
</body>
</html>
```



인접 형제 선택자(adjacent selector)

- 문서 구조상 같은 부모를 가진 형제 요소 중 첫 번째 동생 요소에만 스타일 적용.
- 같은 부모 요소를 가지는 요소들을 형제 관계라고 부른다.

먼저 나오는 요소를 '형 요소', 나중에 나오는 요소를 '동생 요소'라고 한다.

요소1 + 요소2 {속성:속성 값; ...}

기본형

예) `h1 + p { text-decoration : underline; }`



인접 형제 선택자(adjacent selector)

예제 Samples\6장\adjacent.html

```
<style>
  h1+ul {
    color:blue; /* 글자색 파랑 */
    font-weight:bold; /* 글자 굵게 */
  }
</style>
```

〈h1〉 태그 다음의 첫 번째 〈ul〉
태그에 적용할 스타일 지정

```
</head>
<body>
  <section id="container">
    <h1>예약 방법 및 요금</h1>
```

```
    <ul>
      <li>직접 통화</li>
      <li>문자 남기기</li>
```

〈h1〉 태그 다음에 오는
첫 번째 〈ul〉 태그

```
    <ul>
      <li>1인 : 40,000원</li>
      <li>2인 : 60,000원</li>
      <li>3인 : 80,000원</li>
      <li>4인 : 100,000원</li>
    </ul>
  </section>
</body>
```

구현된 결과

예약 방법 및 요금

- 직접 통화
- 문자 남기기
- 1인 : 40,000원
- 2인 : 60,000원
- 3인 : 80,000원
- 4인 : 100,000원

형제 선택자(sibling selector)

형제 선택자(sibling selector)

- 형제 요소들에 스타일 적용
- 인접 형제 선택자와 다른 점은 모든 형제 요소에 다 적용된다는 것이다.

요소1 ~ 요소2 {속성:속성 값; ...}

기본형

예) `h1~p { text-decoration : underline; }`



형제 선택자(sibling selector)

예제_ Samples\6장\sibling.html

```
<style>
h1~ul {
  color:blue; /* 글자색 파랑 */
  font-weight:bold; /* 글자 굵게 */
}
```

〈h1〉 태그 다음에 오는 모든 형제 요소
〈ul〉 태그에 적용할 스타일 지정

```
</style>
</head>

<body>
  <section id="container">
```

```
    <h1>예약 방법 및 요금</h1>
```

```
    <ul>
      <li>직접 통화</li>
      <li>문자 남기기</li>
    </ul>
    <ul>
      <li>1인 : 40,000원</li>
      <li>2인 : 60,000원</li>
      <li>3인 : 80,000원</li>
      <li>4인 : 100,000원</li>
    </ul>
```

〈h1〉 태그 다음에 오는 〈ul〉 태그

```
  </section>
</body>
```

구현된 결과

예약 방법 및 요금

- 직접 통화
- 문자 남기기
- 1인 : 40,000원
- 2인 : 60,000원
- 3인 : 80,000원
- 4인 : 100,000원



인접 형제 및 형제 선택자

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습</title>
  <style>
    div > h3 + input {
      background : yellow;
    }

    div > h3 ~ p {
      color : purple;
      font-weight : bold;
      border : 1px solid red;
    }

    p + p {
      font-size : 2em;
    }
  </style>
</head>
```

```
<body>
  <div>
    <h3>div 자식 h3</h3>
    <input type="text" value="1" />
    <input type="text" value="2" />

    <p>div 형제1</p>
    <p>div 형제2</p>
  </div>
</body>
</html>
```

div 자식 h3

1 2

div 형제1

div 형제2



가상 클래스

표기	설명
:link	링크
:visited	사용자가 방문했던 링크
:active	대상 요소가 활성화되었을 때. 클릭했을 때 등
:hover	사용자가 대상 요소를 가리키고 있을 때. 롤 오버 등 (마우스 커서 진입)
:focus	대상 요소가 포커스 되었을 때
:checked	대상 요소가 체크 되었을 때 (checkbox, radio)
:selected	대상 요소가 선택 되었을 때 (select)



가상 클래스 - 1

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습 </title>
  <style>
    a { text-decoration : none; }
    a[id=honeyv]:link { color : green; }
    a[id=honeyv]:visited { color : red; }
    a[id=honeyv]:active { color : blue; }
    textarea[name=text]:hover { background-color : gray; }
    textarea[name=text]:focus {
      border : 3px double blue;
    }
  </style>
</head>
```

```
<body>
  <a id="honeyv" href="http://www.honeyv.com">허니비 </a>
  <br>
  <textarea name="text"> </textarea>
  <br>
  <a href="http://www.zdnet.co.kr" class="sel">ZDNET </a>
  <br>
  <a href="http://www.oss.kr" class="sel">공개SW포털 </a>
  <br>
  <a href="http://devmento.co.kr" class="sel">Dev Mento </a>
  <br>
</body>
</html>
```

허니비

ZDNET
공개SW포털
Dev Mento



가상 클래스

표기	설명
:root	문서 안의 루트 요소(html)에 스타일을 적용한다
:nth-child(n)	부모 요소의 n번째의 자식 요소에 스타일을 적용한다.
:nth-last-child(n)	부모 요소의 뒤로부터 n번째의 자식 요소에 스타일을 적용한다.
:nth-of-type(n)	같은 유형의 요소 중에서 n 번째 나타나는 요소에 스타일을 적용한다.
:nth-last-of-type(n)	같은 유형의 요소 중에서 끝에서부터 세어 n번째 나타나는 요소에 스타일을 적용한다.



가상 클래스

표기	설명
:first-child	첫번째 자식 요소에 스타일을 적용한다.
:last-child	마지막 자식 요소에 스타일을 적용한다.
:first-of-type	형제 중에서 첫번째 요소에 스타일을 적용한다.
:last-of-type	형제 중에서 마지막 요소에 스타일을 적용한다.
:only-child	자식 요소가 하나 뿐인 부모 요소에 스타일을 적용한다.
:only-of-type	해당 요소가 하나 뿐인 요소에 스타일을 적용한다.



가상 클래스 - 2

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습 </title>
  <style>
    :root {
      background-color : azure;
    }
    p:nth-child(3) {
      color : red;
    }
    p:nth-of-type(3) {
      color : blue;
    }
  </style>
</head>
```

```
<body>
  <div>
    <p>1번 </p>
    <span>2번 </span>
    <p>3번 </p>
    <span>4번 </span>
    <p>5번 </p>
    <span>6번 </span>
  </div>
</body>
</html>
```

1번
2번
3번
4번
5번
6번

가상 클래스 - 3

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습 </title>
  <style>
    td:nth-of-type(2n) {
      background : green;
    }
    td:nth-child(1) {
      background : red;
    }
    td:nth-of-type(5) {
      background : blue;
    }
  </style>
</head>
```

```
<body>
  <table border="1">
    <tr>
      <td>1 </td> <td>2 </td> <td>3 </td> <td>1 </td> <td>2 </td>
    </tr>
    <tr>
      <td>4 </td> <td>5 </td> <td>6 </td> <td>4 </td> <td>5 </td>
    </tr>
    <tr>
      <td>7 </td> <td>8 </td> <td>9 </td> <td>7 </td> <td>8 </td>
    </tr>
  </table>
</body>
</html>
```

1	2	3	1	2
4	5	6	4	5
7	8	9	7	8



가상 요소

표기	설명
::before	특정 요소의 내용 앞에 지정한 내용을 만든다.
::after	특정 요소의 내용 뒤에 지정한 내용을 만든다.
::selection	사용자가 드래그해서 선택하는 영역의 스타일을 지정한다.
::not()	괄호 안에 있는 요소를 제외한 요소에 스타일을 적용한다.



가상 클래스 - 4

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>HTML5 연습</title>
  <style>
    a::before {
      content : '앵커 태그 - ';
      color : red;
    }
    #honeyv:active::after {
      content : '-' attr(href);
    }
    a::selection {
      background-color : orange;
    }
    a:not(.sel) {
      background-color : yellow;
    }
  </style>
</head>
```

```
<body>
  <a id="honeyv" href="http://www.honeyv.com">허니비</a>
  <br>
  <a href="http://www.zdnet.co.kr" class="sel">ZDNET</a>
  <br>
  <a href="http://www.oss.kr" class="sel">공개SW포털</a>
  <br>
  <a href="http://devmento.co.kr" class="sel">Dev Mento</a>
  <br>
</body>
</html>
```

앵커 태그 - 허니비

앵커 태그 - ZDNET

앵커 태그 - 공개SW포털

앵커 태그 - Dev Mento



CSS3란

CSS3란

- CSS2나 CSS1에 비해 좀 더 정교하고 화려한 화면을 구성할 수 있고 애니메이션도 지원한다.
- 여러 모듈로 쪼개어 개발 중 → 일부는 완성되었지만 대부분 아직도 개발 중

CSS3와 브라우저 접두사(prefix)

- CSS3 표준 규약이 완성되지 않아 대부분의 CSS3 속성을 사용하려면 속성 이름 앞에 브라우저를 식별할 수 있는 접두사(prefix)를 붙여서 사용해야 한다.

-webkit-	웹킷(Webkit) 방식의 브라우저용(사파리, 크롬 등)
-moz-	게코(Gecko) 방식의 브라우저용(모질라, 파이어폭스 등)
-o-	오페라 브라우저
-ms	마이크로소프트 인터넷 익스플로러