웹프로그래밍

- Web Programming -

3. HTML/CSS

정보보호학과 이병천 교수

3. HTML & CSS

- 3. HTML 기본문서 만들기
- 4. 다양한 내용 입력하기
- 5. 입력양식 작성하기
- 6. CSS 기본
- 7. 텍스트를 표현하는 다양한 스타일
- 8. 레이아웃을 구성하는 CSS 박스모델
- 9. 이미지와 그러데이션 효과로 배경 꾸미기
- 10. CSS 고급 선택자
- 11. 트랜지션과 애니메이션
- 12. 반응형 웹과 미디어 쿼리
- 블로그 예제

03. HTML 기본 문서 만들기

03-1 HTML과 첫 만남

03-2 HTML 구조 파악하기

03-3 HTML 파일 만들기

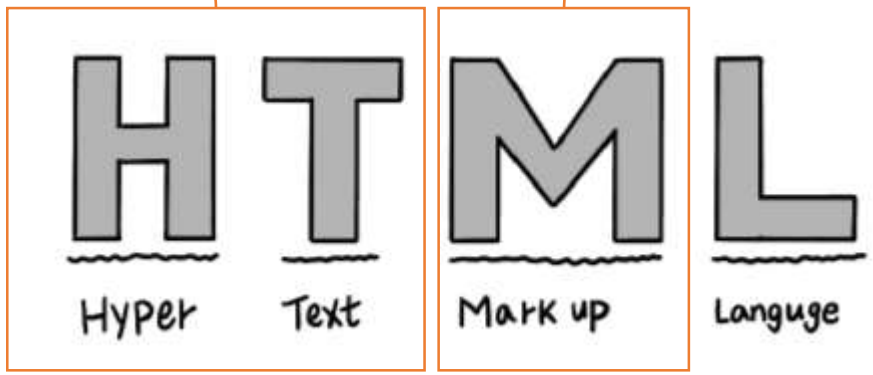
03-4 웹 문서 구조를 만드는 시맨틱 태그

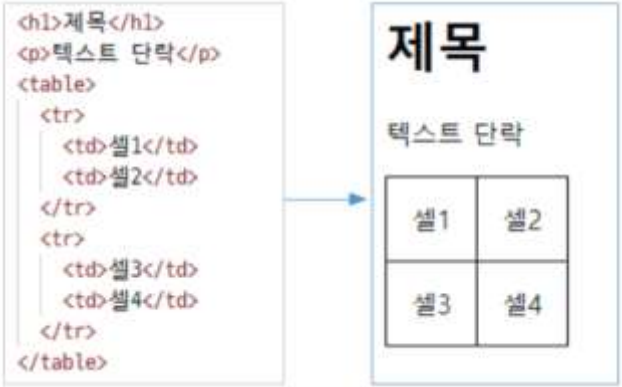
HTML이란

웹에서 자유롭게 오갈 수 있는 링크

텍스트뿐만 아니라 이미지, 영상 등을 표시함

마크업에 사용하는 꼬리표를 '태그(tag)'라고 함

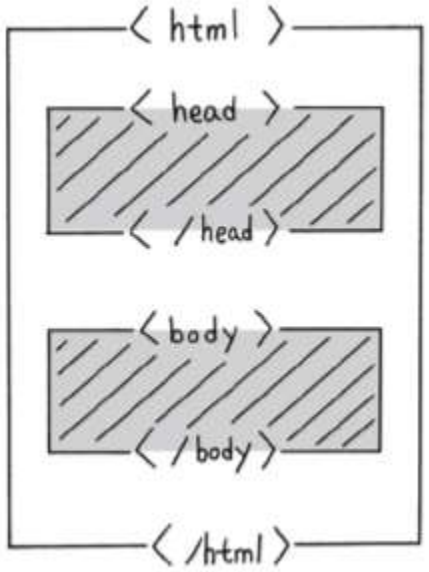




HTML은 웹에서 자유롭게 오갈 수 있는 웹 문서를 만드는 언어

HTML 문서의 기본 구조

```
<!DOCTYPE html>
<html lang="ko">
  <head>
    <meta charset="UTF-8">
    <title>HTML 기본 문서</title>
  </head>
  <body>
    <h1> 프런트엔드 웹 개발 </h1>
    <hr>
    <p>HTML</p>
    <p>CSS</p>
    <p>자바스크립트</p>
  </body>
</html>
```



❶ **<!doctype html>**: 현재 문서가 HTML5 언어로 작성한 웹 문서라는 뜻입니다.

❷ **<html> ~ </html>**: 웹 문서의 시작과 끝을 나타내는 태그입니다. 웹 브라우저가 <html> 태그를 만나면 </html>까지 소스를 읽어 화면에 표시합니다.

❸ **<head> ~ </head>**: 웹 브라우저가 웹 문서를 해석하는 데 필요한 정보를 입력하는 부분입니다.

❹ **<body> ~ </body>**: 실제로 웹 브라우저 화면에 나타나는 내용입니다. 앞으로 우리가 공부할 HTML 태그는 대부분 <body> 태그 안에 들어 있습니다.

HTML 문서의 기본 구조

<!DOCTYPE html> 또는 <!doctype html>

웹 브라우저에게 '이제부터 처리할 문서는 HTML 문서라고 알려주는 것

웹 문서 시작을 알리는 <html> 태그

- 실제 문서 정보와 내용이 시작되고 끝나는 것을 표시하는 태그
- lang 속성을 사용해 문서에서 사용할 언어 지정
- 문서 정보를 지정하는 <head> 부분과 문서 내용을 입력하는 <body> 부분 포함

HTML 문서의 기본 구조

브라우저에게 정보를 주는 <head> 태그

- 문서 관련 정보 입력, 웹 브라우저 화면에는 보이지 않음.
- 문서에서 사용할 외부 파일 링크

문자 세트 등 문서 정보가 들어 있는 <meta> 태그

한글로 된 내용을 표시하기 위해서 UTF-8 문자 세트를 사용

```
<meta charset="UTF-8">
```

이외에도 다양한 문서 정보를 지정

```
<meta name="keywords" content="html의 구조">
<meta name="description" content="html의 구조를 알아봅시다">
<meta name="author" content="Kyunghee Ko">
```

웹 문서의 키워드

웹 문서의 설명

웹 문서의 소유자나 제작자

문서 제목을 나타내는 <title> 태그

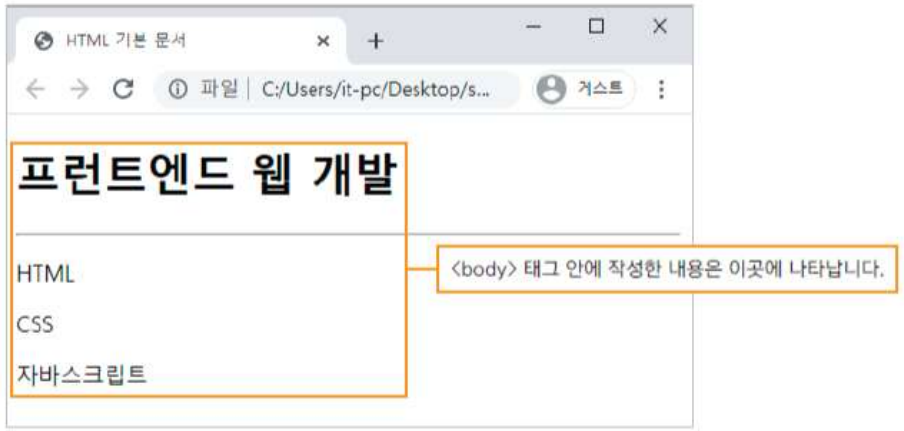
```
<title> HTML 기본 문서 </title>
```

HTML 문서의 기본 구조

웹 브라우저에 내용을 표시하는 <body> 태그

- 실제 브라우저에 표시될 내용 입력
- 대부분의 태그가 <body> 태그와 </body> 태그 사이에서 사용하는 태그들

```
<body>
  <h1> 프런트엔드 웹 개발 </h1>
  <hr>
  <p>HTML</p>
  <p>CSS</p>
  <p>자바스크립트</p>
</body>
```



웹 문서 구조를 만드는 시맨틱 태그

시맨틱(semantic) : 의미론적인, 의미가 통하는

이름만 봐도 의미를 알 수 있는 HTML 태그

왜 사용할까?

- 화면 낭독기 같은 보조 기기에서 사이트의 구조를 이해
- 할 수 있다
- 문서 구조가 정확히 나눠지므로 PC나 모바일 등 다양한 화면에서 웹 문서를 표현하기가 쉽다
- 웹 사이트를 검색할 때 필요한 내용을 정확히 찾을 수 있다

```
<div id="container">
  <header>
    ......
    <nav>
      ......
    </nav>
  </header>
  <main class="contents">
    <section id="headling">
      <h2>몸과 마음이 치유되는 섬</h2>
      ......
    </section>
    <section id="activity">
      <h2>다양한 액티비티가 기다리는 섬</h2>
      ......
    </section>
  </main>
  <footer>
    <section id="bottomMenu">
      ......
    </section>
  </footer>
</div>
```

헤더 영역

내비게이션 영역

본문 영역

푸터 영역

웹 문서 구조를 만드는 시맨틱 태그

헤더 영역을 나타내는 <header> 태그

- 헤더 영역
- 사이트 전체의 헤더 or 특정 영역의 헤더
- 검색 창이나 사이트 메뉴 삽입

내비게이션 영역을 나타내는 <nav> 태그

- 내비게이션 영역
- 웹 문서 위치에 영역을 받지 않음
- 문서 안에 여러 개 만들 수 있음(id로 구분)

```
<header>
  <div id="logo">
    <a href="#"><h1>Dream Jeju</h1></a>
  </div>
  <nav>
    <ul id="topMenu">
      <li><a href="#">단체 여행</a></li>
      <li><a href="#">맞춤 여행</a></li>
      <li><a href="#">갤러리</a></li>
      <li><a href="#">문의하기</a></li>
    </ul>
  </nav>
</header>
```

로고 영역

내비게이션 영역

그림 3-31 로고와 내비게이션을 포함하고 있는 헤더

웹 문서 구조를 만드는 시맨틱 태그

핵심 콘텐츠를 담는 <main> 태그

- 웹 문서에서 핵심이 되는 내용
- 웹 문서마다 다르게 보여주는 내용으로 구성
- 웹 문서에서 한 번만 사용

독립적인 콘텐츠를 담는 <article> 태그

- 독립된 웹 콘텐츠 항목
  (따로 떼어도 콘텐츠가 되는 내용)
- <section> 태그를 포함할 수 있음

콘텐츠 영역을 나타내는 <section> 태그

- 콘텐츠 영역
- 몇 개의 콘텐츠를 묶는 용도로 사용
- **css 적용을 위해 묶는 용도로 쓰지 말 것**

 태그

```
<main class="contents">
  <section id="headling">
    <h2>몸과 마음이 치유되는 섬</h2>
      ……
  </section>
  <section id="activity">
    <h2>다양한 액티비티가 기다리는 섬</h2>
      ……
  </section>
</main>
```

두 개의 <section> 태그 영역으로 나누었습니다.

웹 문서 구조를 만드는 시맨틱 태그

사이드 바 영역을 나타내는 <aside>

- 본문 내용 외에 왼쪽이나 오른쪽, 혹은 아래쪽에 사이드 바 표시
- 필수 요소가 아니므로 필요할 경우에만 사용

푸터 영역을 나타내는 <footer>

- 사이트 제작 정보나 저작권 정보, 연락처 등
- 다른 시맨틱 태그 사용해 다양한 정보 포함

**여러 소스를 묶는 **

- 소스를 묶는 용도
- 영역을 구별하거나 스타일 적용하기 위해

```
<footer>
  <div id="bottomMenu">
    <ul>
      <li><a href="#">회사 소개</a></li>
      <li><a href="#">개인정보처리방침</a></li>
      <li><a href="#">여행약관</a></li>
      <li><a href="#">사이트맵</a></li>
    </ul>
  </div>
</footer>
```

```
<header>
  <div id="logo">
    <a href="#"> <h1>Dream Jeju</h1>
    </a>
  </div>
  <nav>
    <ul id="topMenu">
      ……
    </ul>
  </nav>
</header>
```

04. 다양한 내용 입력하기

04-1 텍스트 태그 알아보기

04-2 목록 만들기

04-3 표 만들기

04-4 이미지 삽입하기

04-5 오디오와 비디오 삽입하기

04-6 하이퍼링크 삽입하기

04-1 텍스트 태그 알아보기

<h*n*> 태그 – 제목 표시

기본형 : **<h*n*> 제목 </h*n*>**

제목을 표시할 때 사용하는 태그

크기 : h1 > h2 > h3 > h4 > h5 > h6

태그 – 텍스트 단락

기본형 : ** 텍스트 **

입력한 내용 앞뒤로 빈 줄이 생기면서 텍스트 단락이 만들어짐

 태그 – 줄 바꾸기

기본형 : **
**

줄을 바꿀 위치에
 태그를 사용. 닫는 태그가 없음

```
<h1>레드향</h1>
<p>껍질에 붉은 빛이 돌아 레드향이라 불린다.</p>
<p>레드향은 한라봉과 귤을 교배한 것으로 <br>일반 귤보다 2~3배 크고, 과육이 붉고 통통하다.</p>
<p>비타민 C와 비타민 P가 풍부해<br> 혈액순환, 감기예방 등에 좋은 것으로 알려져 있다.</p>
<h2>레드향 샐러드 레시피</h2>
<h2>상품 구성</h2>
```

레드향

껍질에 붉은 빛이 돌아 레드향이라 불린다.

레드향은 한라봉과 귤을 교배한 것으로
일반 귤보다 2~3배 크고, 과육이 붉고 통통하다.

비타민 C와 비타민 P가 풍부해
혈액순환, 감기예방 등에 좋은 것으로 알려져 있다.

레드향 샐러드 레시피

상품 구성

04-1 텍스트 태그 알아보기

<blockquote> 태그 – 인용문 넣기

기본형 : **<blockquote> 인용 내용 </blockquote>**

다른 텍스트보다 안으로 들여 써짐.

```
<h1>레드향</h1>
<p>껍질에 붉은 빛이 돌아 레드향이라 불린다.</p>
<p>레드향은 한라봉과 귤을 교배한 것으로 <br>일반 귤보다 2~3배 크고, 과육이 붉고 통통하다.</p>
<blockquote>비타민 C와 비타민 P가 풍부해<br> 혈액순환, 감기예방 등에 좋은 것으로 알려져 있
다.</blockquote>
```

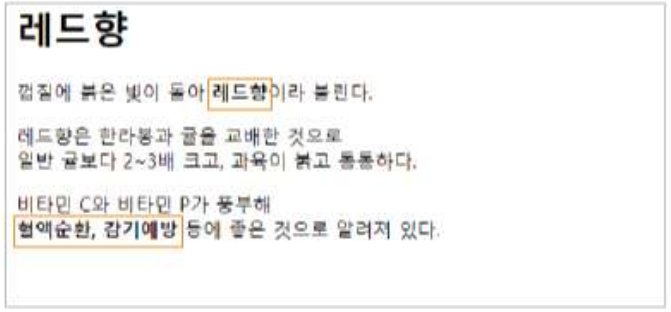
04-1 텍스트 태그 알아보기

, – 굵게 표시

기본형 : ** 내용 **, ** 내용 **

- - 중요한 내용이라서 강조해야 할 때
- - 단순히 굵게 표시할 때

```
<h1>레드향</h1>
<p>껍질에 붉은 빛이 돌아 <b>레드향</b>이라 불린다.</p>
<p>레드향은 한라봉과 귤을 교배한 것으로<br>일반 귤보다 2~3
배 크고, 과육이 붉고 통통하다.</p>
<p>비타민 C와 비타민 P가 풍부해<br> <strong>혈액순환, 감기예
방</strong> 등에 좋은 것으로 알려져 있다.</p>
```



, – 이탤릭체로 표시하기

기본형 : ** 내용 **, ** 내용 **

- - 흐름상 특정 부분을 강조하고 싶을 때
- - 단순히 이탤릭체로 표시할 때

```
<h1>레드향</h1>
<p>껍질에 붉은 빛이 돌아 <b>레드향</b>이라 불린다.</p>
<p>레드향은 <em>한라봉과 귤을 교배</em>한 것으로<br>일반 귤보
다 2~3배 크고, 과육이 붉고 통통하다.</p>
<p><i>비타민 C</i>와 <i>비타민 P</i>가 풍부해<br> <strong>혈액순
환, 감기예방</strong> 등에 좋은 것으로 알려져 있다.</p>
```

04-2 목록 만들기

<ol>, <li> - 순서 목록

기본형 : <ol>

<li>항목 1</li>

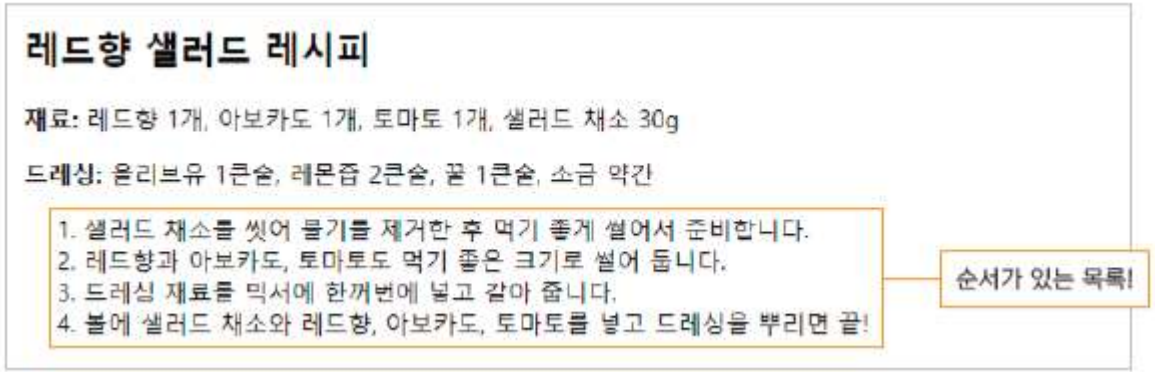
<li>항목 2</li>

.....

</ol>

- 각 항목 앞에 숫자가 붙여짐
- type 속성 : 순서 목록의 숫자 조정
  (1: 숫자, a: 소문자, A:대문자, i:로마소문자, I:로마대문자)
- start 속성 : 목록의 시작 번호 수정

```
  <h2>레드향 샐러드 레시피</h2>
  <p><b>재료 : </b>레드향 1개, 아보카도 1개, 토마토 1개, 샐러드 채소 30g</p>
  <p><b>드레싱 : </b>올리브유 1큰술, 레몬즙 2큰술, 꿀 1큰술, 소금 약간</p>
  <ol>
    <li>샐러드 채소를 씻고 물기를 제거한 후 준비합니다.</li>
    <li>레드향과 아보카도, 토마토를 먹기 좋은 크기를 썰어둡니다.</li>
    <li>드레싱 재료를 믹서에 갈아줍니다.</li>
    <li>볼에 샐러드 채소와 썰어 둔 레드향, 아보카도, 토마토를 넣고 드레싱을 뿌리
면 끝!</li>
  </ol>
```



04-2 목록 만들기

<ul>, <li> - 순서 없는 목록

기본형 : <ul>

<li>항목 1</li>

<li>항목 2</li>

.....

</ul>

- 각 항목 앞에 불릿이 붙여짐

```
<h2>레드향 샐러드 레시피</h2>
<p><b>재료 : </b>레드향 1개, 아보카도 1개, 토마토 1개, 샐러드 채소 30g</p>
<p><b>드레싱 : </b>올리브유 1큰술, 레몬즙 2큰술, 꿀 1큰술, 소금 약간</p>
<ul>
  <li>샐러드 채소를 씻고 물기를 제거한 후 준비합니다.</li>
  <li>레드향과 아보카도, 토마토를 먹기 좋은 크기를 썰어둡니다.</li>
  <li>드레싱 재료를 믹서에 갈아줍니다.</li>
  <li>볼에 샐러드 채소와 썰어 둔 레드향, 아보카도, 토마토를 넣고 드레싱을 뿌리
면 끝!</li>
</ul>
```

레드향 샐러드 레시피

재료 : 레드향 1개, 아보카도 1개, 토마토 1개, 샐러드 채소 30g

드레싱 : 올리브유 1큰술, 레몬즙 2큰술, 꿀 1큰술, 소금 약간

- 샐러드 채소를 씻어 물기를 제거한 후 먹기 좋게 썰어서 준비합니다.
- 레드향과 아보카도, 토마토도 먹기 좋은 크기로 썰어 둡니다.
- 드레싱 재료를 믹서에 한꺼번에 넣고 갈아줍니다.
- 볼에 샐러드 채소와 레드향, 아보카도, 토마토를 넣고 드레싱을 뿌리면 끝!

04-2 목록 만들기

<dl>, <dt>, <dd> - 설명 목록

기본형 : <dl>

```
<dt>이름(제목) 1</dt>
<dd>값(설명) 1</dd>
<dt>이름(제목) 2</dt>
<dd>값(설명) 2</dd>
.......
</dl>
```

- '이름(제목)'과 '값(설명)' 형태로 된 목록
- <dt> 태그 – 이름(제목) 지정
- <dd> 태그 – 값(설명) 지정
- 하나의 <dt>에 여러 개의 <dd> 값을 가질 수 있다

```
<h2>상품 구성</h2>
 <dl>
    <dt>선물용 3kg</dt>
    <dd>소과 13~16과</dd>
    <dd>중과 10~12과</dd>
  </dl>
  <dl>
    <dt>선물용 5kg</dt>
    <dd>중과 15~19과</dd>
  </dl>
```

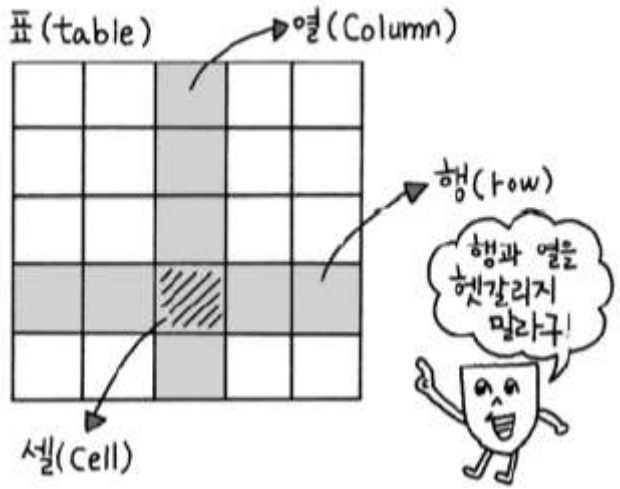
상품 구성

선물용 3kg
소과 13~16과
중과 10~12과

선물용 5kg
중과 15~19과

04-3 표 만들기

표의 구성 요소



표를 만드는 태그

<caption> : 표 제목 , **<table>** : 표 전체

<tr> : 행, **<td>** : 셀, **<th>** : 제목 셀

```
<table>
  <caption>선물용과 가정용 상품 구성</caption>
  <tr>
    <th>용도</th>
    <th>중량</th>
    <th>갯수</th>
    <th>가격</th>
  </tr>
  <tr>
    <td>선물용</td>
    <td>3kg</td>
    <td>11~16과</td>
    <td>35,000원</td>
  </tr>
  ……
  <tr>
    <td>가정용</td>
    <td>5kg</td>
    <td>18~26과</td>
    <td>47,000원</td
  </tr>
</table>
```

선물용과 가정용 상품 구성

용도	중량	개수	가격
선물용	3kg	11~16과	35,000원
선물용	5kg	18~26과	52,000원
가정용	3kg	11~16과	30,000원
가정용	5kg	18~26과	47,000원

04-3 표 만들기

표의 구조 정의하기

- 표의 구조를 '제목'과 '본문', '요약' 부분으로 나눈다.
- <thead>, <tbody>, <tfoot> 태그 사용
- 웹 브라우저 화면에서 보이지 않지만, 화면 낭독기나 자바스크립트 등에서 읽을 수 있다. → 시각 장애인도 표의 구조를 쉽게 이해할 수 있다.
- 표의 본문이 길 경우 자바스크립트를 이용해 제목과 바닥 부분을 고정하고 본문만 스크롤되도록 할 수 있다.

상품 구성

선물용과 가정용 상품 구성

용도	중량	개수	가격
선물용	3kg	11~16과	35,000원
선물용	5kg	18~26과	52,000원
가정용	3kg	11~16과	30,000원
가정용	5kg	18~26과	47,000원

이곳은 <thead>를 나타냅니다.

이곳은 <tbody>를 나타냅니다.

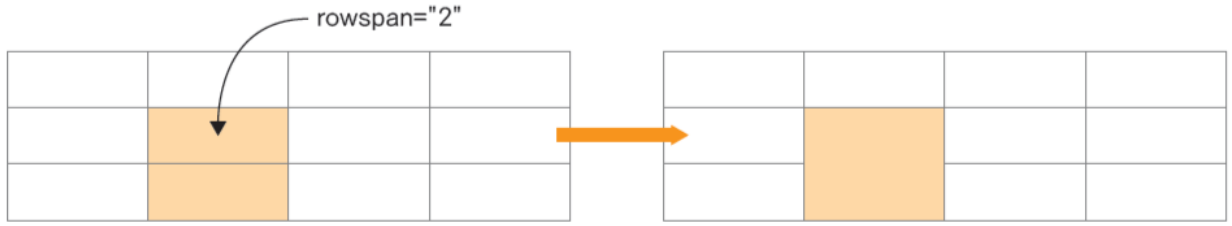
```
<table>
  <caption>선물용과 가정용 상품 구성</caption>
  <thead>
    <tr>
      <th>용도</th>
      <th>중량</th>
      <th>갯수</t>
      <th>가격</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>선물용</td>
      <td>3kg</td>
      <td>11~16과</td>
      <td>35,000원</td>
    </tr>
    ……
    <tr>
      <td>가정용</td>
      <td>5kg</td>
      <td>18~26과</td>
      <td>47,000원</td>
    </tr>
  </tbody>
</table>
```

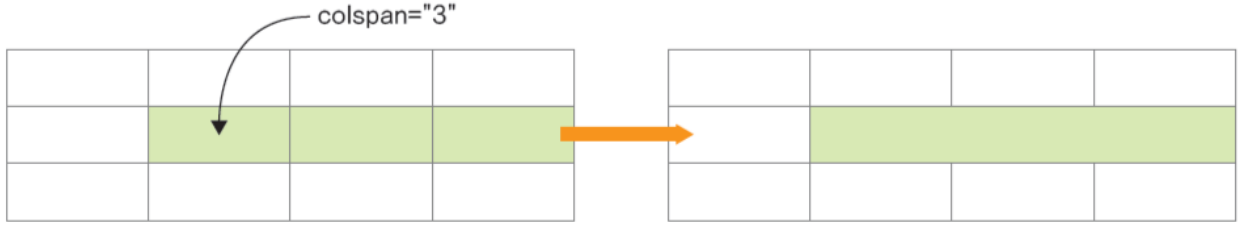
04-3 표 만들기

colspan, rowspan 속성 – 행 또는 열 합치기

기본형

```
<td rowspan="합칠 셀의 개수">셀의 내용</td>
<td colspan="합칠 셀의 개수">셀의 내용</td>
```






```
<table>
  <caption>선물용과 가정용 상품 구성</caption>
  <thead>
    <tr>
      <th>용도</th>
      <th>중량</th>
      <th>갯수</t>
      <th>가격</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td rowspan="2">선물용</td>
      <td>3kg</td>
      <td>11~16과</td>
      <td>35,000원</td>
    </tr>
    <tr>
      <td>5kg</td>
      <td>18~26과</td>
      <td>52,000원</td>
    </tr>
    <tr>
      <td rowspan="2">가정용</td>
      <td>3kg</td>
      <td>11~16과</td>
      <td>30,000원</td>
    </tr>
    <tr>
      <td>5kg</td>
      <td>18~26과</td>
      <td>47,000원</td>
    </tr>
  </tbody>
</table>
```

선물용과 가정용 상품 구성

용도	중량	갯수	가격
선물용	3kg	11~16과	35,000원
	5kg	18~26과	52,000원
가정용	3kg	11~16과	30,000원
	5kg	18~26과	47,000원

04-4 이미지 삽입하기

<img> 태그

웹 문서에 이미지를 삽입할 때 사용

기본형

```
<img src="이미지 파일 경로" alt="대체용 텍스트">
```

```
<img src="images/tangerines.jpg" alt="레드향">
<h1>레드향</h1>
```

레드향

<참고> 웹 문서와 이미지

웹 페이지에서 사용할 수 있는 이미지 파일은 파일 크기가 크지 않으면서도 화질은 좋게 유지해야 하기 때문에 주로 몇 가지 파일 형식만 사용합니다.

파일 형식	설명
GIF (graphic interchange format)	표시할 수 있는 색상 수가 최대 256가지뿐이지만 다른 이미지 파일 형식에 비해 파일 크기가 작아서 아이콘이나 불릿 등 작은 이미지에 주로 사용합니다.
JPG/JPEG (joint photographic experts group)	색상과 명암을 GIF보다 다양하게 표현할 수 있습니다. 이미지를 수정하고 저장하는 작업을 반복할 경우 화질이 떨어질 수도 있습니다.
PNG (portable network graphics)	네트워크용으로 개발된 파일 형식입니다. 색상을 다양하게 표현하면서 투명한 배경도 만들 수 있으므로 웹에서 가장 많이 사용합니다.

04-4 이미지 삽입하기

<img> 태그의 src 속성

- 웹 문서 파일의 위치를 기준으로 이미지 경로 지정
- HTML과 이미지 파일이 같은 폴더에 있다면 src 속성에 파일 이름만 적음

  예) <img src="tangerines.jpg">
- 하위 폴더에 이미지 파일이 있다면 하위 폴더까지 같이 적음

  예) <img src="images/tangerines.jpg">

<img> 태그의 alt 속성

- 이미지를 설명하는 대체 텍스트
- 화면 낭독기에서 이미지 대신 대체 텍스트를 읽어 줌. (웹 접근성)
- 이미지를 표시할 수 없는 상황일 때 대체 텍스트 표시
- 텍스트 자체를 이미지로 만들었을 경우 대체 텍스트 안에 이미지 파일의 내용을 그대로 넣어주어야 함.
- 불릿 이미지나 작은 아이콘 등의 경우에는 alt=" " 처럼 지정

04-4 이미지 삽입하기

<img> 태그의 width, height 속성

- 이미지 크기 조정하기
- width나 height 중 1개만 지정해도 나머지 속성은 자동으로 비율을 계산

단위	설명	예시
%	이미지 크기 값을 퍼센트(%)로 지정하면 현재 웹 브라우저 창의 너비와 높이를 기준으로 이미지 크기를 결정합니다.	width="50%"
px	이미지 크기 값을 픽셀(px)로 지정하면 이미지의 너비나 높이를 해당 픽셀 크기만큼 표시합니다.	width="150"

```
<p>원래 크기의 이미지</p>
<img src="images/salad.jpg" alt="레드향">
<p>width="50%", height="50%"로 지정한 이미지</p>
<img src="images/salad.jpg" alt="레드향" width="50%">
<p>width="150"으로 지정한 이미지</p>
<img src="images/salad.jpg" alt="레드향" width="150">
```

04-5 오디오와 비디오 삽입하기

<object> 태그

- 음악 파일 뿐만 아니라 동영상이나 자바 애플릿, PDF 파일 등 다양한 개체를 삽입
- 기본형

  <object width="너비" height="높이" data="파일"></object>

  - data – 재생할 파일 지정
  - width, height – 플레이어 화면 크기 지정

```
<h1>웹 문서에 PDF 파일 삽입하기</h1>
<object width="900" height="800" data="product.pdf"></object>
```

04-5 오디오와 비디오 삽입하기

<embed> 태그

- 대부분 브라우저에서 사용 가능

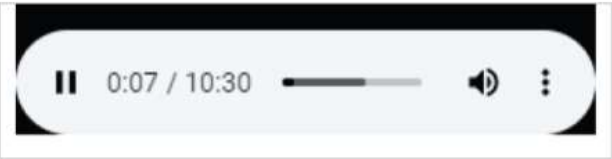
  → <object>, <audio>, <video> 태그를 지원하는 브라우저에서 멀티미디어 삽입할 때 사용

- 기본형

  <embed src="파일 경로" width="너비" height="높이">

  - 닫는 태그 없음
  - src : 멀티미디어 파일 지정
  - width, height : 플레이어 크기 지정

```
<embed src="medias/spring.mp3">
```



04-5 오디오와 비디오 삽입하기

웹 브라우저에 지원하는 멀티미디어 파일의 종류

종류	확장자	설명
비디오	mp4	많이 사용하는 비디오 형식으로, 라이선스가 있지만 웹에서는 무료로 사용할 수 있습니다.
	webm	화질이 우수하면서도 무료라서 mp4와 함께 많이 사용합니다.
오디오	mp3	대부분의 음원에서 사용하는 형식입니다. 2017년 이후 라이선스가 종료되면서 무료로 사용할 수 있습니다.
	mp4/ m4a	mp3의 문제점을 개선한 AAC 코덱을 사용한 파일 형식입니다. 확장자는 mp4뿐만 아니라 m4a를 사용하기도 합니다.

웹 브라우저별 오디오, 비디오 파일 지원 여부

웹 브라우저	오디오	설명	
	mp3	mp4	webm
인터넷 익스플로러	O(9+)	O(9+)	X
크롬	O(all)	O(all)	O(25+)
파이어폭스	O(22+)	O(35+)	O(28+)
사파리	O(4+)	O(3.2+)	X
오페라	O(15+)	O(25+)	O(16+)
iOS 사파리	O(4.1+)	O(all)	X
안드로이드 브라우저	O(2.3+)	O(4.4+)	X

04-5 오디오와 비디오 삽입하기

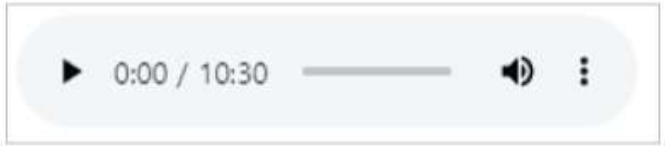
<audio> 태그, <video> 태그

기본형

```
<audio src="오디오 파일 경로"></audio>
<video src="비디오 파일 경로"></video>
```

```
<audio src="medias/spring.mp3" controls></audio>
```

재생막대 표시



재생막대 표시 플레이어 너비 지정

```
<video src="medias/salad.mp4" controls width="700"></video>
```

04-5 오디오와 비디오 삽입하기

<audio> 태그, <video> 태그의 속성

속성	설명
controls	플레이어 화면에 컨트롤 바를 표시합니다.
autoplay	오디오나 비디오를 자동으로 실행합니다.
loop	오디오나 비디오를 반복 재생합니다.
muted	오디오나 비디오의 소리를 제거합니다.
preload	페이지를 불러올 때 오디오나 비디오 파일을 어떻게 로딩할 것인지 지정합니다. 사용할 수 있는 값은 auto, metadata, none입니다. 기본적으로 preload="auto"가 사용됩니다.
width, height	비디오 플레이어의 너비와 높이를 지정합니다. width나 height의 값 중에서 하나만 지정하면 나머지는 자동으로 계산해서 표시합니다.
poster="파일 이름"	<video> 태그에서 사용하는 속성으로, 비디오가 재생되기 전까지 화면에 표시될 포스터 이미지를 지정합니다.

플레이어 표시 없이 배경 음악 넣기

```
<audio src="medias/spring.mp3" autoplay loop></audio
>
```

비디오 자동 재생하기

```
<video src="medias/salad.mp4" width="700" autoplay muted loop></video>
```

크롬, 파이어폭스 브라우저에서는 음소거해야 비디오를 자동 재생할 수 있음

04-6 하이퍼링크 삽입하기

<a> 태그

- 다른 문서, 혹은 다른 사이트로 바로 연결해 주는 기능
- 기본형

```
<a href="링크할 주소">텍스트 또는 이미지</a>
```

<a> 태그의 target 속성

- target="blank"로 지정하면 연결된 문서가 새 탭으로 열림

텍스트 링크

```
<a href="../05/order.html" target="_blank">주문서 작성하기</a>
```

이미지 링크

```
<a href="../05/order.html">
  <img src="images/tangerines.jpg" alt="레드향">
</a>
```

05. 입력 양식 작성하기

05-1 폼 삽입하기

05-2 사용자 입력을 위한 input 태그

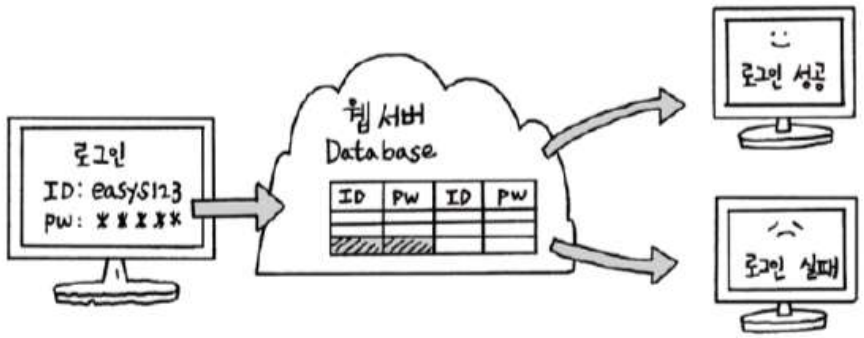
05-3 input 태그의 주요 속성

05-4 폼에서 사용하는 여러 가지 태그

05-1 폼 삽입하기

웹에서 만나는 폼

- 사용자가 웹 사이트로 정보를 보낼 수 있는 요소들은 모두 폼(form)
- 폼과 관련된 대부분의 작업들은 정보를 저장하거나 검색하거나 수정하는 일들
- 이런 작업은 모두 데이터베이스를 기반으로 한다.
- 아이디를 입력하는 텍스트 필드나 버튼 같은 폼의 형태를 만드는 것은 HTML 태그
- 폼에 입력한 사용자 정보를 처리하는 것은 ASP나 PHP 같은 서버 프로그래밍 이용



로그인 폼의 동작 과정

05-1 폼 삽입하기

<form> 태그

- 폼을 만드는 기본 태그. <form>과 </form> 사이에 여러 폼 요소 삽입
- 기본형

  <form [속성="속성값"]> 여러 폼 요소 </form>

<form> 태그의 속성

속성	설명
method	사용자가 입력한 내용을 서버 쪽 프로그램으로 어떻게 넘겨줄 것인지 지정합니다. method에서 사용할 수 있는 속성값은 get과 post입니다. • get: 데이터를256~4,096byte까지만 서버로 넘길 수 있습니다. 주소 표시줄에 사용자가 입력한 내용이 그대로 드러나는 단점이 있습니다. • post: 입력한 내용의 길이에 제한받지 않고 사용자가 입력한 내용도 드러나지 않습니다.
name	자바스크립트로 폼을 제어할 때 사용할 폼의 이름을 지정합니다.
action	<form> 태그 안의 내용을 처리해 줄 서버 프로그램을 지정합니다.
target	action 속성에서 지정한 스크립트 파일을 현재 창이 아닌 다른 위치에서 열도록 합니다.

폼에 내용을 입력하고 서버로 전송했을 때
서버에 있는 register.php를 실행한다

```
<form action="register.php">
  /* 여러 가지 폼 요소 */
</form>
```

05-1 폼 삽입하기

<fieldset> 태그

- 폼 요소를 그룹으로 묶는 태그
- 기본형

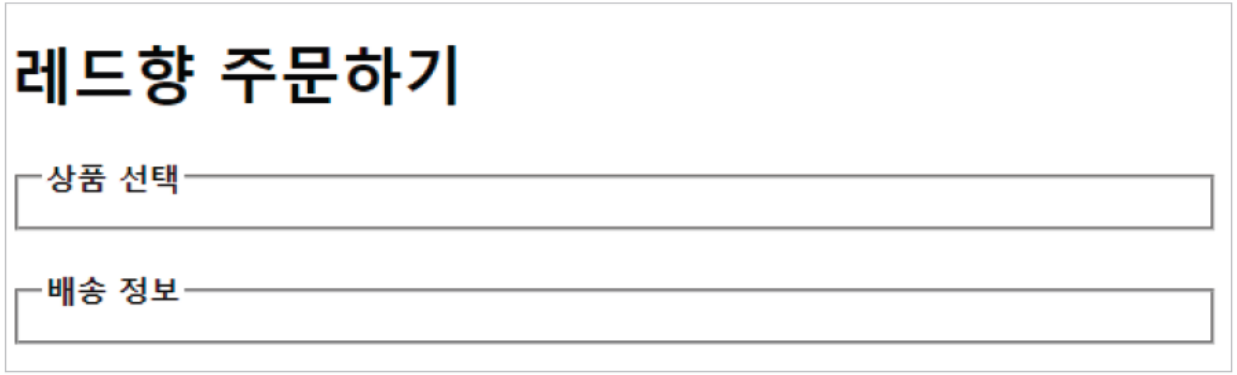
```
<fieldset [속성="속성값"]> ~ </fieldset>
```

<legend> 태그

- 그룹으로 묶는 구역에 제목을 붙이는 태그
- 기본형

```
<fieldset>
  <legend>그룹 이름</legend>
</fieldset>
```

```
<form action="">
  <fieldset>
    <legend>상품 선택</legend>

  </fieldset>
  <fieldset>
    <legend>배송 정보</legend>

  </fieldset>
</form>
```



05-1 폼 삽입하기

<label> 태그

- <input> 태그와 같은 폼 요소에 레이블을 붙일 때 사용

기본형 <label>레이블명<input></label>

<label> 태그 안에 <input> 태그 넣기

```
(... 생략 ...)
<label>아이디(6자 이상)<input type="text"></label>
(... 생략 ...)
```

기본형 <label for="id명">레이블명<input id="id명"></label>

Do it! <label> 태그와 폼 요소를 따로 쓰고 연결하기 예제 파일 05\label.html

```
(... 생략 ...)
<label for="user-id">아이디(6자 이상)</label>
<input type="text" id="user-id">
(... 생략 ...)
```

아이디(6자 이상)

<input> 태그의 type 속성 한 눈에 살펴보기

종류	설명
text	한 줄짜리 텍스트를 입력할 수 있는 텍스트 박스를 넣습니다.
password	비밀번호를 입력할 수 있는 필드를 넣습니다.
search	검색할 때 입력하는 필드를 넣습니다.
url	URL 주소를 입력할 수 있는 필드를 넣습니다.
email	이메일 주소를 입력할 수 있는 필드를 넣습니다.
tel	전화번호를 입력할 수 있는 필드를 넣습니다.
checkbox	주어진 여러 항목에서 2개 이상 선택할 수 있는 체크 박스를 넣습니다.
radio	주어진 여러 항목에서 1개만 선택할 수 있는 라디오 버튼을 넣습니다.
number	숫자를 조절할 수 있는 스핀 박스를 넣습니다.
range	숫자를 조절할 수 있는 슬라이드 막대를 넣습니다
date	사용자 지역을 기준으로 날짜(연, 월, 일)를 넣습니다.
month	사용자 지역을 기준으로 날짜(연, 월)를 넣습니다.
week	사용자 지역을 기준으로 날짜(연, 주)를 넣습니다.
time	사용자 지역을 기준으로 시간(시, 분, 초, 분할 초)을 넣습니다.
datetime	국제 표준시(UTC)로 설정된 날짜와 시간(연, 월, 일, 시, 분, 초, 분할 초)을 넣습니다.
datetime-local	사용자가 있는 지역을 기준으로 날짜와 시간(연, 월, 일, 시, 분, 초, 분할 초)을 넣습니다.
submit	전송 버튼을 넣습니다.
reset	리셋 버튼 을 넣습니다.
image	submit 버튼 대신 사용할 이미지를 넣습니다.
button	일반 버튼을 넣습니다
file	파일을 첨부할 수 있는 버튼을 넣습니다.
hidden	사용자에게는 보이지 않지만 서버로 넘겨주는 값이 있는 필드를 만듭니다.

<input> 태그

type = “text”

- 한 줄짜리 텍스트 입력 필드
- 주로 아이디나 이름, 주소 등 텍스트 입력

type = “password”

- 비밀번호 입력란
- 사용자가 입력하는 내용이 ' * '나 '•'로 표시된다

텍스트 필드와 비밀번호 필드의 속성

종류	설명
size	텍스트와 비밀번호 필드의 길이를 지정합니다. 즉, 화면에 몇 글자가 보이도록 할 것인지를 지정합니다. 예를 들어 최대로 입력할 수 있는 글자 수가 10개인데 size 속성값을 5로 지정하면 필드 크기는 5개 글자 크기에 맞추고 나머지 5개 글자는 화면에 보이지 않습니다.
value	텍스트 필드 요소가 화면에 표시될 때 텍스트 필드 부분에 보여 주는 내용입니다. 이 속성을 사용하지 않으면 빈 텍스트 필드가 표시됩니다. 비밀번호 필드에서 사용하지 않는 속성입니다.
maxlength	텍스트 필드와 비밀번호 필드에 입력할 수 있는 최대 문자 수를 지정합니다.

(단, 비밀번호 필드에는 value 속성이 없음)

기본형
```
<input type="text">
<input type="password">
```

Do it! 간단한 로그인 폼 만들기 예제 파일 05\login.html

```
(... 생략 ...)
<form>
  <fieldset>
    <label>아이디: <input type="text" id="user_id" size="10"></label>
    <label>비밀번호: <input type="password" id="user_pw" size="10"></label>
    <input type="submit" value="로그인">
  </fieldset>
</form>
(... 생략 ...)
```

아이디: 비밀번호: 로그인

<input> 태그

type = “search”

- 검색 필드
- 검색 창에 x 표시가 되어 검색어 삭제 쉬움

type = “url”

- 웹 주소 필드
- http://'로 시작하는 사이트 주소 입력

type = “email”

- 메일 주소 입력 필드
- 메일 주소 형식 자동 체크

type = “tel”

- 전화번호 입력 필드
- 사용자 입력을 체크하지는 않음.

Do it! 배송 정보 입력하는 폼 만들기 예제 파일 05\t-field.html

```
(... 생략 ...)
<ul>
  <li>
    <l for="user-name">이름</label>
    <input type="text" id="user-name">
  </li>
  <li>
    <label for="addr">배송 주소</label>
    <input type="text" id="a">
  </li>
  <li>
    <label for="m">이메일 주소</label>
    <input type="email" id="mail">
  </li>
  <l>
    <label for="phone">연락처</label>
    <input type="tel" id="phone">
  </li>
</ul>
(... 생략 ...)
```

정답: label, addr, mail, ll

배송 정보

- 이름
- 배송 주소
- 이메일
- 연락처

<input> 태그

type = “radio”

여러 항목 중 하나만 선택할 때

type = “checkbox”

여러 항목 중 둘 이상을 선택할 때

```
기본형  <input type="checkbox">
        <input type="radio">
```

라디오 버튼, 체크 박스에서 사용하는 속성

속성	설명
name	라디오 버튼이나 체크박스가 여러 개 있을 경우, 서버의 폼 프로그램에서 라디오 버튼이나 체크박스를 구분하기 위해 이름을 지정합니다. 라디오 버튼은 여러 개 중에서 한 가지만 선택하는 것이기 때문에 관련 있는 그룹끼리는 name 속성 값을 똑같이 만듭니다.
value	선택한 라디오 버튼이나 체크박스를 서버로 알려 줄 때 넘길 값을 지정합니다. 이 값은 영문이거나 숫자여야 하며 필수 속성입니다.
checked	라디오 버튼의 항목들은 처음에 아무 것도 선택되지 않은 상태로 화면에 표시되는데 기본으로 선택해 놓을 항목이 있다면 checked 속성을 사용합니다.

Do it! 체크 박스와 라디오 버튼을 삽입하는 폼 만들기 — 예제 파일 05\radio-check.html

```
(... 생략 ...)
<fieldset>
  <legend>상품 선택</legend>
  <p><b>주문할 상품을 선택해 주세요.</b></p>
  <label><input type="c          " value="s_3">선물용 3kg</label>
  <label><input type="checkbox" value="s_5">선물용 5kg</label>
  <label><input type="checkbox" value="f_3">가정용 3kg</label>
  <label><i          type="checkbox" value="f_3">가정용 5kg</label>
  <p><b>포장 선택</b></p>
  <label><input type="r          " name="gift" value="yes">선물 포장</label>
  <label><input type="radio" name="gift" value="no">선물 포장 안 함</label>
</f          >
(... 생략 ...)
```

정답: input, radio, fieldset

상품 선택

주문할 상품을 선택해 주세요.

☐ 선물용 3kg ☐ 선물용 5kg ☐ 가정용 3kg ☐ 가정용 5kg

포장 선택

○ 선물 포장 ○ 선물 포장 안 함

<input> 태그

type = “number”

숫자 입력 필드. 브라우저에 따라 스핀 박스로 표시됨

type = “range”

숫자 입력 필드. 슬라이드 막대를 이용해 숫자 입력

```
기본형  <input type="number">
        <input type="range">
```

숫자 입력 필드에서 사용하는 속성

속성	설명
min	필드에 입력할 수 있는 최솟값을 지정합니다. 기본 최솟값은 0입니다.
max	필드에 입력할 수 있는 최댓값을 지정합니다. 기본 최댓값은 100입니다.
step	숫자 간격을 지정할 수 있습니다. 기본값은 1입니다.
value	필드에 표시할 초깃값입니다.

Do it! 스핀 박스를 사용해 숫자 입력하기 — 예제 파일 05\number.html

```
(... 생략 ...)
<ul>
  <li>
    <label><input type="checkbox" value="s_3">선물용 3kg</label>
    <input type="n                " min="0" max="5">개 (최대 5개)
  </li>
  <li>
    <label><input type="checkbox" value="s_5">선물용 5kg</label>
    <input type="number" min="0" m              ="3" value="1">개 (최대 3개)
  </li>
</ul>
(... 생략 ...)
```

정답: number, max

- ☐ 선물용 3kg [ ] 개 (최대 5개)
- ☐ 선물용 5kg [1] 개 (최대 3개)

Do it! 슬라이드 막대를 사용해 숫자 입력하기 — 예제 파일 05\number.html

```
(... 생략 ...)
<ul>
  <li>
    <label><input type="checkbox" value="f_3">가정용 3kg</label>
    <input type="r                " min="0" max="5">개 (최대 5개)
  </li>
  <li>
    <label><input type="checkbox" value="f_3">가정용 5kg</label>
    <input type="range" m              ="0" max="3" value="1">개 (최대 3개)
  </li>
</ul>
(... 생략 ...)
```

정답: range, min

- ☐ 가정용 3kg [———●———] 개 (최대 5개)
- ☐ 가정용 5kg [——●————] 개 (최대 3개)

<input> 태그

type = “date”, type=“month”, type=“week”

달력 이용해 날짜 입력

기본형 `<input type="date | month | week">`

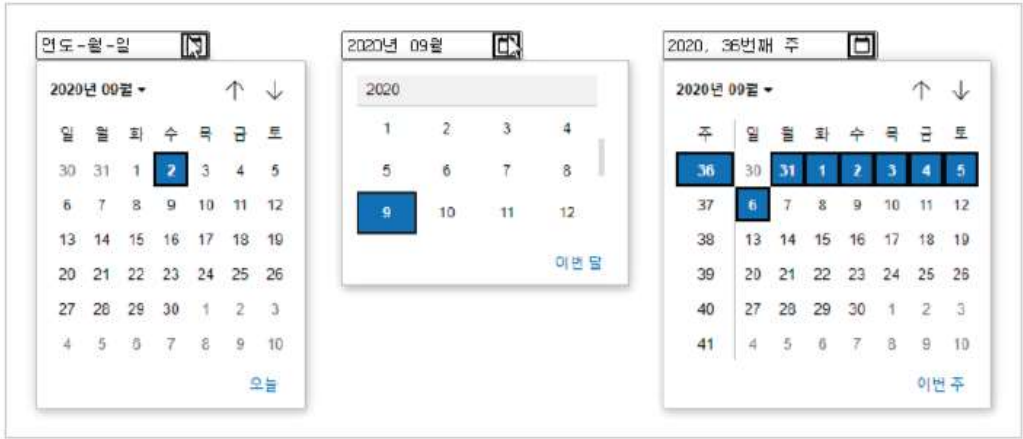
❶ type="date"로 지정하면 달력에서 날짜를 선택해서 입력할 수 있습니다. 날짜를 선택하면 필드에 "yyyy-mm-dd" 형식으로 연도, 월, 일이 표시됩니다.

❷ type="month"로 지정하면 달력에서 월을 선택해서 입력할 수 있습니다. 월을 선택하면 "yyyy-mm" 형식으로 연도, 월까지만 입력됩니다.

❸ type="week"로 지정하면 달력에서 주를 선택해서 입력할 수 있습니다. 주를 선택하면 1월 첫째 주를 기준으로 몇 번째 주인지 표시됩니다.

Do it! 3가지 형식의 날짜 입력하기 — 예제 파일 05\datetime.html

```
(... 생략 ...)
<h1>날짜 지정하기</h1>
<input type="date">
<input type="month">
<input type="week">
(... 생략 ...)
```



<input> 태그

type = “time”, type=“datetime-local”

시간 입력

기본형 `<input type="time ¦ datetime ¦ datetime-local">`

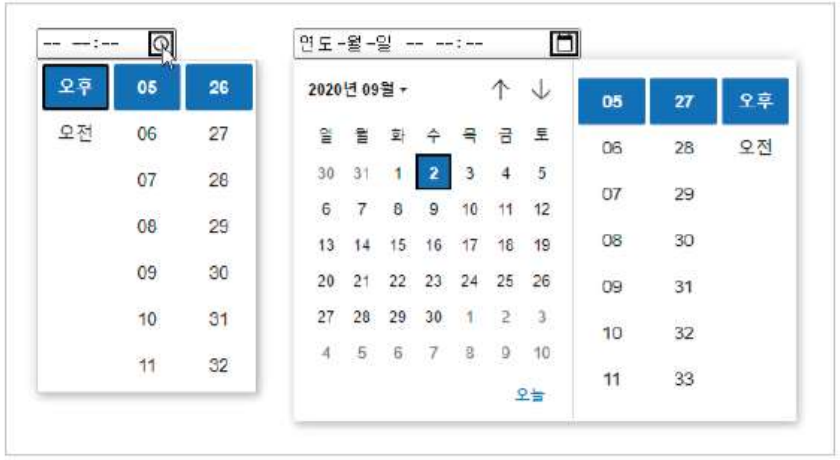
❶ type="time"은 폼에서 시간을 입력하게 합니다. 시간 입력 필드는 웹 브라우저마다 조금씩 다르게 나타나는데 항목이 3개로 구성된다는 점은 같습니다. 첫 번째 항목부터 '오전'과 '오후'를 의미하고 나머지는 '시'와 '분'을 의미합니다.

❷ type="datetime" 또는 type="datetime-local" 유형을 사용하면 사용자가 웹 문서를 보고 있는 지역에 맞는 날짜와 시간을 함께 입력할 수 있습니다.

Do it! 2가지 형식의 시간 입력하기 — 예제 파일 05\datetime.html

```
(... 생략 ...)
<h1>시간 지정하기</h1>
<input type="time">
<input type="datetime-local">
(... 생략 ...)
```

정답: time, type



<input> 태그

type = “submit”, type=“reset”

- 폼 전송/리셋 버튼
- 전송(submit) 버튼 : 사용자 입력 내용을 서버로 전송
- 리셋(reset) 버튼 : 사용자 입력 내용 전부 삭제
- value 속성을 이용해 버튼 표시 내용 지정

기본형 `<input type="submit ¦ reset " value="버튼에 표시할 내용">`

Do it! 전송과 리셋 버튼 추가하기 — 예제 파일 05\button.html

```
(... 생략 ...)
<div>
  <input type="submit" value="주문하기">
  <input type="reset" value="취소하기">
</div>
(... 생략 ...)
```

배송 정보

- 이름
- 배송 주소
- 이메일
- 연락처
- 배송 지정 (주문일로부터 최소 3일 이후)
  연도-월-일

주문하기 취소하기

type = “image”

submit 버튼 대신 이미지 삽입

기본형 `<input type="image" src="이미지 경로" alt="대체 텍스트">`

Do it! 이미지 버튼 추가하기 — 예제 파일 05\login-image.html

```
(... 생략 ...)
<fieldset>
  <label>아이디: <input type="text" id="user_id" size="10"></label>
  <label>비밀번호: <input type="password" id="user_pw" size="10"></label>
  <input t          ="image" s          ="images/login.png" alt="로그인">
</fieldset>
(... 생략 ...)
```

정답: type, src

아이디: 비밀번호: 로그인

<input> 태그

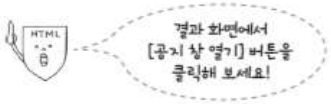
type = “button”

- 기능 없이 버튼 형태만 삽입
- 주로 버튼 클릭해서 자바스크립트 실행할 때 사용
- value 속성을 이용해 버튼 표시 내용 지정

기본형 `<input type="button" value="버튼에 표시할 내용">`

Do it! 버튼을 삽입해 자바스크립트 실행하기 — 예제 파일 05\button-2.html

```
(... 생략 ...)
<form>
  <input type="button" value="공지 창 열기" onclick="window.open('notice.html')">
</form>
(... 생략 ...)
```



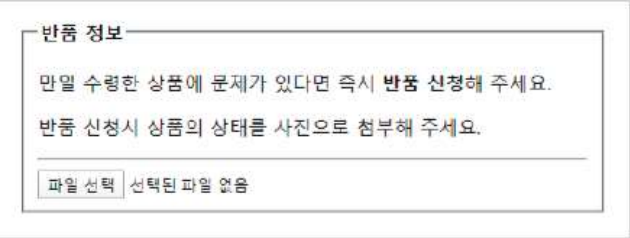
type = “file”

- 파일 첨부
- ‘파일 선택 ' 이나 ‘찾아보기’ 버튼으로 표시됨

기본형 `<input type="file">`

Do it! 파일 첨부 버튼 추가하기 — 예제 파일 05\file.html

```
(... 생략 ...)
  <input type="file">
</fieldset>
(... 생략 ...)
```



<input> 태그

type = “hidden”

- 화면 상의 폼에는 보이지 않는다.
- 폼을 서버로 전송할 때 서버로 함께 전송되는 요소

기본형 `<input type="hidden" name="이름" value="서버로 넘길 값">`

Do it! 히든 필드를 사용해 서버로 정보 보내기 예제 파일 05\hidden.html

```
(... 생략 ...)
<input type="hidden" value="사이트를 통한 직접 로그인">
<label>아이디: <input type="text" id="user_id" size="10"></label>
<label>비밀번호: <input type="password" id="user_pw" size="10"></label>
<input type="submit" value="로그인">
(... 생략 ...)
```

아이디: [ ] 비밀번호: [ ] [로그인]

<input> 태그의 다양한 속성

autofocus

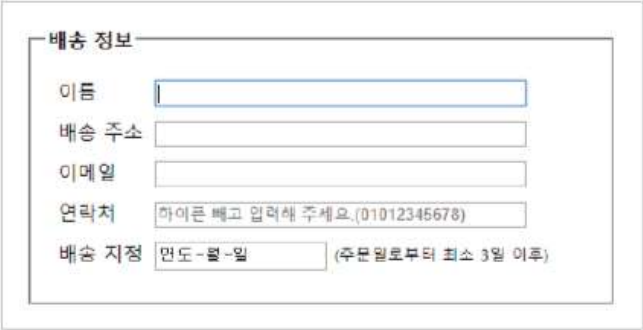
페이지를 불러오자마자 원하는 폼 요소에 마우스 커서 표시

placeholder

입력란에 표시하는 힌트로, 필드를 클릭하면 사라짐

Do it! autofocus와 placeholder 속성 사용하기 — 예제 파일 05\ placeholder.html

```
(... 생략 ...)
<label for="user-name">이름</label>
<input type="text" id="u - n " autofocus>
......
<label for="phone">연락처</label>
<input type="tel" id="phone" p ="하이픈 빼고 입력해 주세요.(01012345678)">
(... 생략 ...)
```

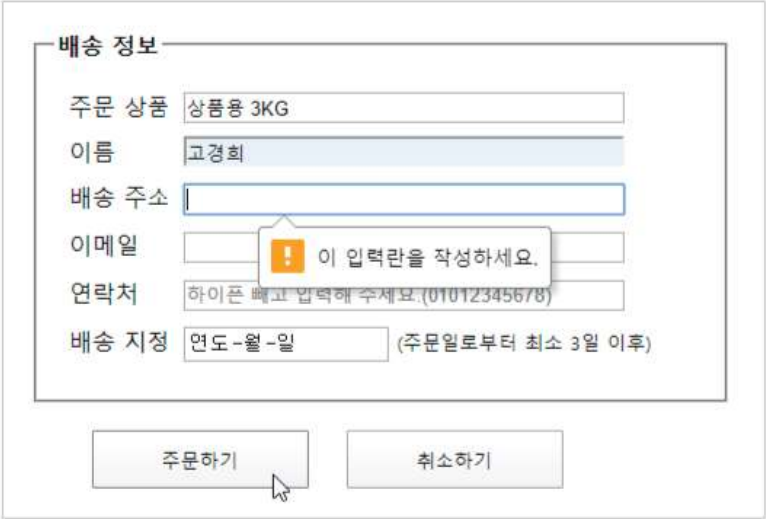


readonly

내용을 보기만 하고 입력하지 못하게 함.

required

- 필수 필드 체크
- 필수 필드는 브라우저에서 직접 체크하는 것이므로 오류 메시지 내용은 브라우저들마다 다르게 나타남



폼에서 사용하는 여러 가지 태그

텍스트 영역 <textarea>

- 여러 줄의 텍스트 입력
- 게시판 글 입력 양식, 사용자 약관 등

기본형 `<textarea>내용</textarea>`

<textarea> 태그의 속성

종류	설명
cols	텍스트 영역의 가로 너비를 문자 단위로 지정합니다.
rows	텍스트 영역의 세로 길이를 줄 단위로 지정합니다. 지정한 숫자보다 줄 개수가 많아지면 스크롤 막대가 생깁니다.

Do it! 텍스트를 여러 줄 입력하기 예제 파일 05\textarea.html

```
(... 생략 ...)
<label for="memo">메모</label>
<textarea id="memo" cols="40" rows="4"></textarea>
(... 생략 ...)
```

폼에서 사용하는 여러 가지 태그

<select>, <optgroup>, <option>

- 여러 옵션 중에서 선택 – 드롭다운 목록
- 공간을 최소한으로 사용하면서 여러 옵션 표시 가능

기본형
```
<select>
    <option value="값1">내용1</option>
    <option value="값2">내용2</option>
    ……
</select>
```

<select> 태그의 속성

종류	설명
size	화면에 표시할 드롭다운 항목의 개수를 지정합니다.
multiple	드롭다운 목록에서 둘 이상의 항목을 선택할 때 사용합니다.

<option> 태그의 속성

종류	설명
value	해당 항목을 선택할 때 서버로 넘겨줄 값을 지정합니다.
selected	드롭다운 메뉴를 삽입할 때 기본적으로 선택해서 보여 줄 항목을 지정합니다.

<datalist>, <option>

- 데이터 목록에 제시한 값 중에서 선택하면
  그 값이 자동으로 텍스트 필드에 입력됨
- 데이터 목록에 id를 이용해 이름을 붙이고,
- <input> 태그의 list 속성에 데이터 목록 id를 지정함.

기본형
```
<input type="text" list="데이터 목록 id">
<datalist id="데이터 목록 id">
  <option value="서버로 넘길 값1">선택 옵션1</option>
  <option value="서버로 넘길 값2">선택 옵션2</option>
</datalist>
```

<button>

- 다양한 형태의 버튼 삽입
- 화면 낭독기에서 버튼임을 정확히 전달할 수 있음
- CSS를 이용해 원하는 형태로 꾸밀 수 있음

기본형
```
<button type="submit">내용</button>
<button type="reset">내용</button>
<button type="button">내용</button>
```

06. CSS 기본

06-1 웹 문서에 디자인 입히기

06-2 스타일과 스타일 시트

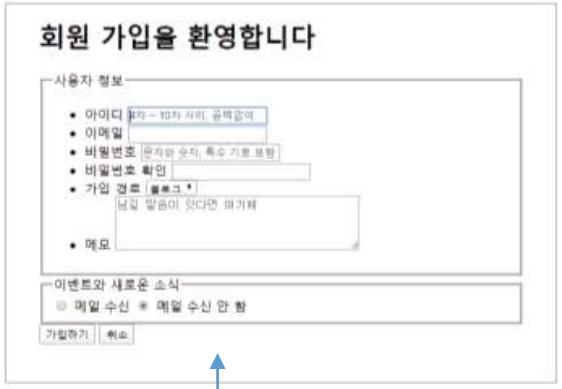
06-3 CSS 기본 선택자

06-4 캐스케이딩 스타일 시트

웹 문서에 디자인 입히기

왜 스타일을 사용할까

- HTML로는 웹 사이트의 내용을 나열하고 CSS로는 웹 문서의 디자인을 구성
  → 스타일을 사용하면 웹 문서의 내용과 상관없이 디자인만 바꿀 수 있다



마크업만 했을 때

스타일을 적용했을 때

- 다양한 기기에 맞게 탄력적으로 바뀌는 문서를 만들 수 있다
  → 반응형 웹 디자인

PC 화면

모바일 화면

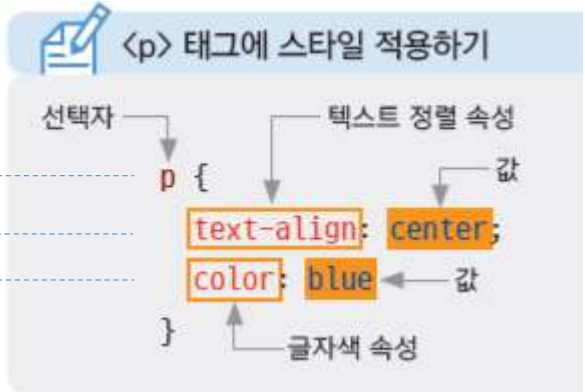
스타일과 스타일 시트

스타일 형식 알아보기

기본형 선택자 { 속성1: 속성값1; 속성2: 속성값2 }

스타일 규칙

텍스트 단락의
글자를 가운데로 정렬
글자색을 파랑으로
지정하고 싶다면



스타일 규칙 작성 방법

- 중괄호({ }) 사이에 스타일 규칙 나열
- 규칙이 여러 개일 경우 세미콜론(;)으로 구분

CSS 여러 줄로 표기하기

```
p {
  text-align: center;
  color: blue;
}
```

CSS 한 줄로 표기하기

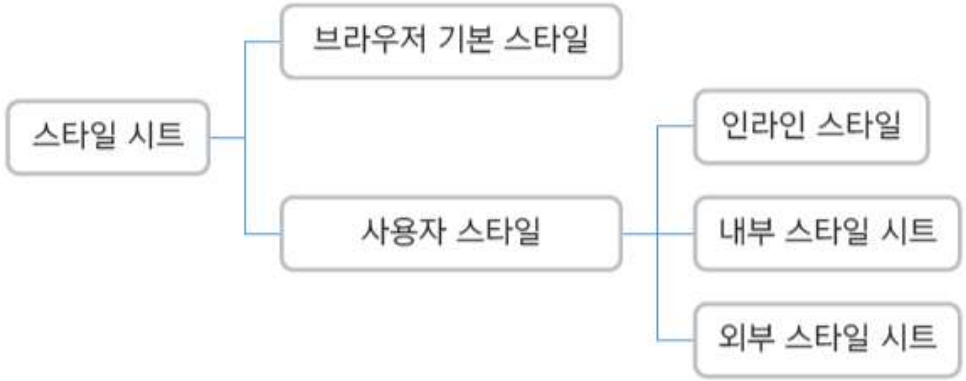
```
p { text-align: center; color: blue; }
```

스타일 주석

- /*와 */ 사이에 주석 내용 입력
- 한 줄 또는 여러 줄을 입력 가능

스타일과 스타일 시트

스타일 시트



브라우저 기본 스타일

- 브라우저에서 기본으로 적용하는 스타일
- 웹 문서에서 아무 스타일도 적용하지 않고 HTML만 사용해도 그 기능에 따라 크기에 맞게 보여줌

인라인 스타일

- 스타일 시트를 사용하지 않고 스타일을 적용할 대상에 직접 표시
- 스타일을 적용하고 싶은 태그에 style 속성을 사용해 **style="속성: 속성 값;"** 형태로 스타일 적용

내부 스타일 시트

- 웹 문서 안에서 사용할 스타일을 문서 안에 정리한 것
- 모든 스타일 정보는 <head> 태그와 </head> 태그 안에서 정의
- <style> 태그와 </style> 태그 사이에 작성

외부 스타일 시트

- 여러 웹 문서에서 사용할 스타일을 별도 파일로 저장해 놓고 필요할 때마다 파일에서 가져와 사용
- <style> 태그 없이 <link> 태그만 사용해 미리 만들어 놓은 외부 스타일 시트 파일 연결

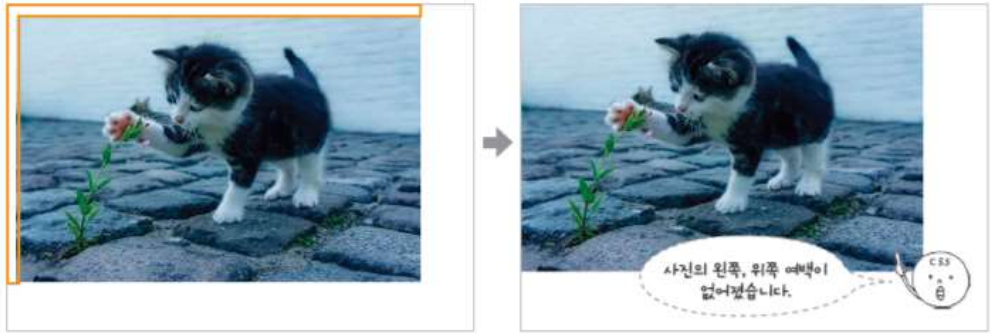
CSS 기본 선택자

전체 선택자

- 페이지에 있는 모든 요소를 대상으로 스타일을 적용할 때 사용
- 웹 브라우저의 기본 스타일을 초기화할 때 자주 사용

기본형 `* { 속성: 값; ..... }`

```
<style>
  * {
    margin: 0;
  }
</style>
</head>
<body>
  <img src="images/cat.jpg">
</body>
```



타입 선택자

- 문서에서 특정 태그를 사용한 모든 요소에 스타일이 적용됨

기본형 `태그명 { 스타일 규칙 }`

```
<style>
  p {
    font-style: italic; /* 이탤릭체 */
  }
</style>

<div>
  <h1>레드향</h1>
  <p>껍질에 붉은 빛이 돌아 레드향이라 불린다.</p>
  <p>레드향은 한라봉과 귤을 교배한 것으로 ...</p>
  <p>비타민 C와 비타민 P가 풍부해 ...</p>
</div>
```

레드향

껍질에 붉은 빛이 돌아 레드향이라 불린다.

레드향은 한라봉과 귤을 교배한 것으로 일반 귤보다 2~3배 크고, 과육이 붉고 풍부하다.

비타민 C와 비타민 P가 풍부해 혈액순환, 감기예방 등에 좋은 것으로 알려져 있다.

CSS 기본 선택자

class 선택자

- 요소의 특정 부분에만 스타일 적용
- 마침표(.) 다음에 클래스 이름 지정
- 문서 안에서 여러 번 반복할 스타일이라면 클래스 선택자로 정의

기본형 .클래스명 { 스타일 규칙 }

```
<style>
  p {
    font-style: italic;      /* 이탤릭체 */
  }
  .accent {
    border: 1px solid #000; /* 테두리 */
    padding: 5px;           /* 테두리와 내용 사이의 여백 */
  }
  .bg {
    background-color: #ddd; /* 배경색 */
  }
</style>
</head>
<body>
  <div>
    <h1 c              ="accent bg">레드향</h1>
    <p>껍질에 붉은 빛이 돌아 <span class="  a              ">레드향</span>이라 불린다.</p>
    <p>레드향은 한라봉과 귤을 교배한 것으로 ...</p>
    <p>비타민 C와 비타민 P가 풍부해 ...</p>
  </div>
```

accent와 bg 클래스 선택자를 사용

accent 클래스 선택자만 사용

레드향

껍질에 붉은 빛이 돌아 레드향이라 불린다.

레드향은 한라봉과 귤을 교배한 것으로 일반 귤보다 2~3배 크고, 과육이 붉고 튼튼하다.

비타민 C와 비타민 P가 풍부해 혈액순환, 감기예방 등에 좋은 것으로 알려져 있다.

id 선택자

- 요소의 특정 부분에만 스타일 적용
- 파운드(#) 다음에 id 이름 지정
- 문서 안에서 한번만 사용한다면 id 선택자로 정의

기본형 #아이디명 { 스타일 규칙 }

```
(... 생략 ...)
<style>
  #container {
    width: 500px;            /* 너비 */
    margin: 10px auto;       /* 중앙 배치 */
    padding: 10px;           /* 테두리와 내용 사이 여백 */
    border: 1px solid #000;  /* 테두리 굵기와 색깔 */
  }
</style>
.....
<div id="  c              ">
  <h1>레드향</h1>
(... 생략 ...)
```

레드향

껍질에 붉은 빛이 돌아 레드향이라 불린다.

레드향은 한라봉과 귤을 교배한 것으로 일반 귤보다 2~3배 크고, 과육이 붉고 튼튼하다.

비타민 C와 비타민 P가 풍부해 혈액순환, 감기예방 등에 좋은 것으로 알려져 있다.

CSS 기본 선택자

그룹 선택자

- 같은 스타일을 사용하는 선택자를 한꺼번에 정의
- 쉼표(,)로 구분해 여러 선택자를 나열

기본형 선택자1, 선택자2 { 스타일 규칙 }

<h1>, <p> 태그를 중앙에 정렬하기

```
h1 {
  text-align: center; /* 중앙 정렬 */
}
p {
  text-align: center; /* 중앙 정렬 */
}
```

그룹 선택자를 사용하여 <h1>, <p> 태그를 중앙에 정렬하기

```
(... 생략 ...)
h1, p {
  text-align: c          ; /* 중앙 정렬 */
}
(... 생략 ...)
```

정답: center

캐스케이딩 스타일 시트

캐스케이딩의 의미

- 캐스캐이딩(Cascading) : '위에서 아래로 흐른다'는 뜻. 즉 계단식으로 적용된다는 의미로 사용.
- 선택자에 여러 스타일이 적용될 때 스타일 충돌을 막기 위해 우선순위에 따라 적용할 스타일을 결정함.

스타일 충돌을 막는(캐스캐이딩)의 원칙

① **스타일 우선순위** - 스타일 규칙의 중요도와 적용 범위에 따라 우선순위가 결정되고 그 우선순위에 따라 위에서 아래로 스타일 적용

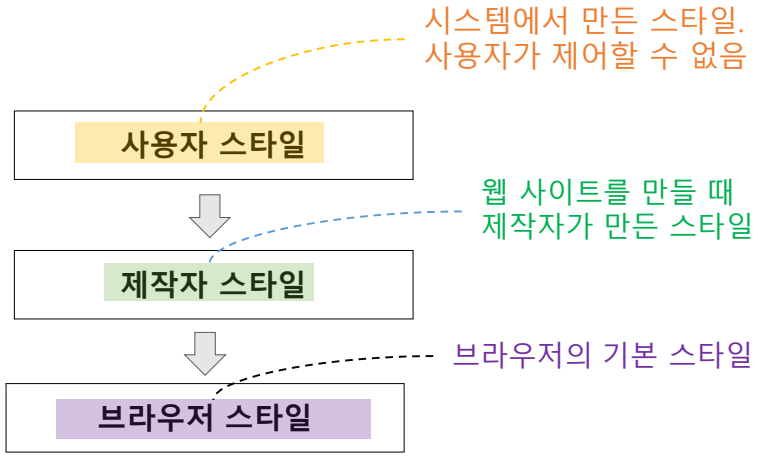
② **스타일 상속** - 태그들의 포함 관계에 따라 부모 요소의 스타일을 자식 요소로, 위에서 아래로 전달

※ 스타일 시트에서 '캐스캐이딩'은 가장 기본적인 개념이기 때문에 일반적으로 '스타일 시트'는 '캐스캐이딩 스타일 시트(CSS)'와 같은 의미로 사용됨

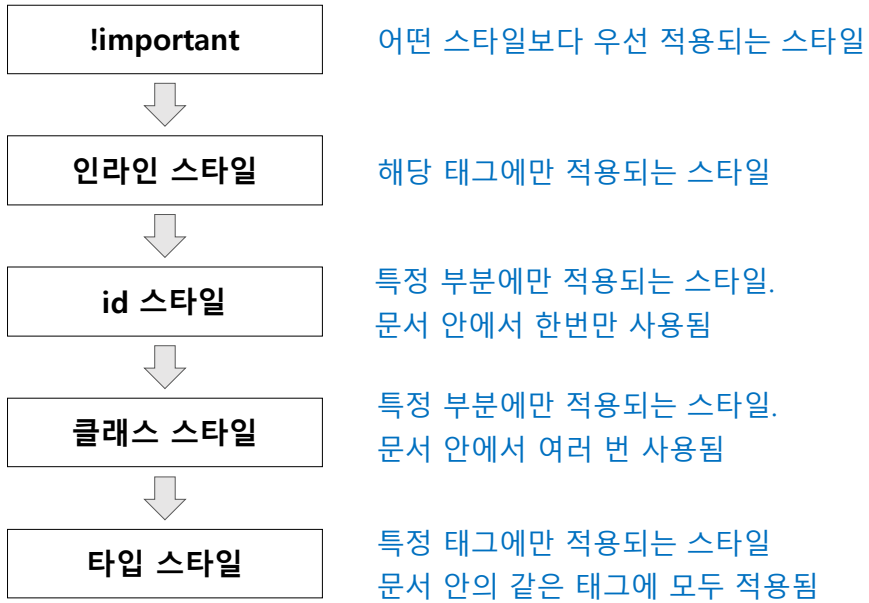
캐스케이딩 스타일 시트

원칙 1: 스타일 우선 순위

1. 얼마나 중요한가에 따라



2. 얼마나 한정지을 수 있는가에 따라



3. 소스 순서에 따라

- 중요도와 명시도가 같다면 소스 순서에 따라 결정
- 소스에서 나중에 온 스타일이 먼저 온 스타일을 덮어씀

캐스케이딩 스타일 시트

원칙 2: 스타일 상속

- 자식 요소에서 별도로 스타일을 지정하지 않으면 부모 요소에 있는 스타일 속성들이 자식 요소로 전달됨.
- 상속을 이용하면 스타일 시트를 효과적으로 만들 수 있다
- 주의할 것은 스타일의 모든 속성이 부모 요소에서 자식 요소로 상속되는 것은 아니라는 점
  (예, 글자 색은 상속되지만 배경 색은 상속되지 않음)

07. 텍스트를 표현하는 다양한 스타일

07-1 글꼴 관련 스타일

07-2 웹 폰트 사용하기

07-3 텍스트 관련 스타일

07-4 목록 스타일

07-5 표 스타일

글꼴 관련 스타일

font-family 속성

기본형 font-family:<글꼴 이름> | [<글꼴 이름>, <글꼴 이름>]

- 웹 문서에서 사용할 글꼴 지정
- <body> 태그를 비롯해 <p> 태그나 <h*n*> 태그처럼 텍스트를 사용하는 요소들에서 사용
- 지정한 글꼴이 없을 경우에 대비해 두 번째, 세 번째 글꼴까지 지정.
- 둘 이상의 글꼴 이름을 지정할 때는 쉼표(,)로 글꼴 구분
- <body> 태그 스타일에서 한 번 정의하면 문서 전체에 적용되고 문서 안의 모든 자식 요소에 계속 같은 글꼴이 사용됨.

Do it! 글꼴 적용하기

```
body { font-family: "맑은 고딕", 돋움, 굴림 }
```

font-size 속성

기본형 font-size: <절대 크기> | <상대 크기> | <크기> | <백분율>

❶ 브라우저에서 지정한 글자 크기
❷ 부모 요소의 글자 크기를 기준으로 상대적인 글자 크기를 지정
❸ 브라우저와 상관없이 글자 크기를 직접 지정
❹ 부모 요소의 글자 크기를 기준으로 백분율(%)로 표시

키워드를 사용해 글자 크기 지정하기

xx-small < x-small < small < medium < large < x-large < xx-large

단위를 사용해 글자 크기 지정하기

종류	설명
em	부모 요소에서 지정한 글꼴의 대문자 M의 너비를 기준(1em)으로 한 후 비율값을 지정합니다.
rem	문서 시작 부분(root)에서 지정한 크기를 기준(1rem)으로 한 후 비율값을 지정합니다.
ex	해당 글꼴의 소문자 x의 높이를 기준(1ex)으로 한 후 비율값을 지정합니다.
px	모니터의 1픽셀을 기준(1px)으로 한 후 비율값을 지정합니다.
pt	포인트라고 하며, 일반 문서에서 많이 사용합니다.

글꼴 관련 스타일

font-style 속성

글자를 이탤릭체로 표시하는 속성

기본형 `font-style: normal | italic | oblique`

❶ 기본값으로 일반적인 형태로 표시
❷ 이탤릭체로 표시
❸ 이탤릭체로 표시

font-weight 속성

글자 굵기를 조절하는 속성

기본형 `font-weight: normal | bold | bolder | lighter | 100 | 200 | … | 800 | 900`

❶ 기본값, 보통 굵기
❷ 굵게
❸ 원래보다 더 굵게
❹ 원래보다 더 가늘게
❺ 100~900 사이의 굵기를 표현하며 100이 가장 가늘게, 900이 가장 굵게

Do it! 글꼴 관련 스타일 사용하기 예제 파일 07\tstyle-1.html

```
(... 생략 ...)
  <style>
    body {
      font-s          : 20px;    /* 전체 글자 크기 */
    }
    h1 {
      font-f          : 바탕;    /* 글꼴 */
      font-size: 3em;            /* 글자 크기 */
    }
    .accent {
      font-size: 150%;           /* 글자 크기 */
      font-w          : 800;     /* 글자 굵기 */
    }
    .italic {
      font-style: italic;        /* 글자 스타일 */
    }
  </style>
.....
  <h1>레드향</h1>
  <p>껍질에 붉은 빛이 돌아 <span class="accent">레드향</span>이라 불린다.</p>
  <p>레드향은 한라봉과 귤을 교배한 것으로</p>
  <p class="italic">일반 귤보다 2~3배 크고, 과육이 붉고 통통하다.</p>
(... 생략 ...)
```

정답: size, family, weight

레드향

껍질에 붉은 빛이 돌아 **레드향**이라 불린다.

레드향은 한라봉과 귤을 교배한 것으로

일반 귤보다 2~3배 크고, 과육이 붉고 통통하다.

웹 폰트 사용하기

웹 폰트란

- 웹 문서 안에 글꼴 정보도 함께 저장했다가 사용자가 웹 문서에 접속하면 글꼴을 사용자 시스템으로 다운로드시켜 사용하는 글꼴.
- 사용자 시스템에 없는 글꼴이더라도 웹 제작자가 의도한 대로 텍스트를 표시할 수 있다.
- @font-face 속성 사용

직접 웹 폰트 업로드해 사용하기

1. 웹 폰트 파일 준비
   - eot 파일, woff 파일
   - 기존 ttf 파일을 변환해서 사용할 수도 있음
2. 다운로드하기 전에 사용자 시스템에 있는지 확인
   - local(*글꼴이름*)
   (IE8 이하 고려해야 하면 eot 파일 선언)
3. woff 파일 선언
4. 용량이 큰 ttf 파일을 마지막에 선언

Do it! 웹 폰트 사용하기 예제 파일 07\webfont.html

```
(... 생략 ...)
<style>
  @font-face {
    font-family: 'Ostrich';
    src: local('Ostrich Sans'),
         url('fonts/ostrich-sans-bold.woff') format('woff'),
         u          ('fonts/ostrich-sans-bold.ttf') format('truetype'),
         url('fonts/ostrich-sans-bold.svg') format('svg');
  }
  .wfont {
    font-family: 'Ostrich', sans-serif;  /* 웹 폰트 지정 */
  }
  p {
    font-s          : 30px;  /* 글자 크기 지정 */
  }
</style>
......
  <p>Using Default Fonts</p>
  <p class="wfont">Using Web Fonts</p>
(... 생략 ...)
```

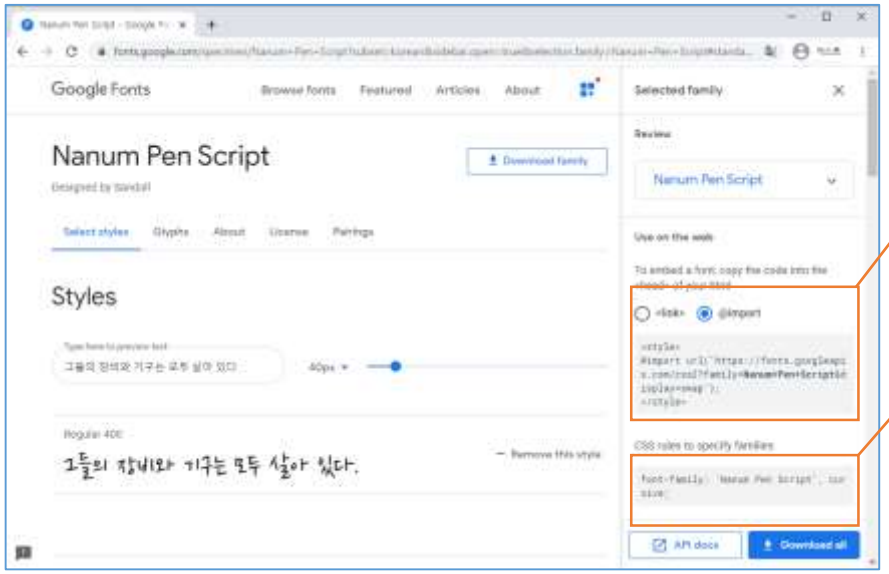
Using Default Fonts

USING WEB FONTS

웹 폰트 사용하기

구글 폰트 사용하기

1. https://fonts.google.com/ 로 접속
2. 한글 폰트 검색
3. Link 항목에 있는 소스 복사 & 글꼴 이름 기억
4. 웹 문서의 <style> 태그 안에 붙여넣음
5. font-family 속성에서 웹 폰트 글꼴 이름 사용




```
(... 생략 ...)
<style>
  @import url('https://fonts.googleapis.com/css2?family=Nanum+Pen+Script&dis-
play=swap');

  h1 {
    font-size: 60px;
    font-weight: bold;
    font-family: 'Nanum Pen Script', cursive;
  }
</style>
```

HTML+CSS+JavaScript

HTML+CSS+JavaScript

텍스트 관련 스타일

color 속성

- 글자 색 지정
- 16진수 값이나 rgb 값, hsl 값, 색상 이름 중에서 사용

색상 이름 표기법

- 잘 알려진 색상 이름으로 표시. 기본 색상 16가지
- 웹 안전 색상(web-safe color)
  - 모든 브라우저에서 표현할 수 있는 색상
  - 기본 16가지 색상을 포함해 모두 216가지

hsl/hsla 표기법

- color:hsl(240, 100%, 50%) 처럼 세 자리의 숫자로 표시.
- 앞의 숫자부터 색상(hue), 채도(saturation), 밝기(light)의 양.
- 투명도를 조절할 때는 마지막에 알파값 추가
  예) hsla(240,100%,50%,0.3)
- 알파값은 불투명도를 나타내는 값으로 0~1 값 중에서 사용
  (1은 불투명, 0은 완전 투명)

16진수 표기법

- #ffffff 처럼 #과 함께 6자리의 16진수로 표시.
- 앞에서부터 두자리씩 묶어 빨강, 초록, 파랑의 양.
- 하나도 섞이지 않았을 때는 00, 가득 섞였을 때는 ff로 표시.
- 000000(검은색) ~ ffffff(흰색)
- 두자리씩 중복될 경우 줄여 사용할 수 있음.
  예) #ffff00 → #ff0, #cccccc → #ccc

rgb/rgba 표기법

- color:rgb(255,0,0)처럼 세 자리의 숫자로 표시.
- 앞의 숫자부터 빨강, 초록, 파랑의 양.
- 하나도 섞이지 않았을 때는 0, 가득 섞였을 때는 255
- 투명도를 조절할 때는 마지막에 알파값 추가
  예) color:rgba(255,0,0,.3)
- 알파값은 불투명도를 나타내는 값으로 0~1 값 중에서 사용
  (1은 불투명, 0은 완전 투명)

텍스트 관련 스타일

text-align 속성

텍스트 정렬 방법 지정

```
기본형 text-align: start | end | left | right | center | justify |
       match-parent
```

종류	설명
start	현재 텍스트 줄의 시작 위치에 맞추어 문단을 정렬합니다.
end	현재 텍스트 줄의 끝 위치에 맞추어 문단을 정렬합니다.
left	왼쪽에 맞추어 문단을 정렬합니다.
right	오른쪽에 맞추어 문단을 정렬합니다.
center	가운데에 맞추어 문단을 정렬합니다.
justify	양쪽에 맞추어 문단을 정렬합니다.
match-parent	부모 요소를 따라 문단을 정렬합니다.

line-height 속성

- 문단의 줄 간격 지정
- <숫자>와 <백분율>은 부모 요소를 기준으로 몇 배인지 지정
- 보통 글자 크기의 1.5~2배 정도면 적당

- 줄 간격은 텍스트를 세로 정렬할 때도 유용함
  → line-height의 속성 값을 영역의 height의 값과 똑같이 지정

Do it! 텍스트를 세로로 가운데 정렬하기 예제 파일 07\line-height-col-2.html

```
(... 생략 ...)
.heading {
  width: 100%;
  height: 100px;
  background: #222;
  color: rgb(255,255,255);
  text-align: center;
  l          : 100px;
}
(... 생략 ...)
```

정답: line-height

텍스트 관련 스타일

text-decoration 속성

- 텍스트에 밑줄을 긋거나 가로지르는 줄 표시
- 텍스트 링크의 밑줄을 없앨 때도 사용

기본형 `text-decoration : none | underline | overline | line-through`

속성 값	설명
none	밑줄을 표시하지 않습니다.*
underline	밑줄을 표시합니다.
overline	영역 위로 선을 그립니다.
line-through	영역을 가로지르는 선(취소 선)을 그립니다.

Do it! 텍스트에 줄 표시하기 — 예제 파일 07\text-decoration.html

```
(... 생략 ...)
<body>
  <h1>text-decoration 속성</h1>
  <p style="text-decoration:none">none</p>
  <p style="text-decoration:underline">underline</p>
  <p style="text-decoration:overline">overline</p>
  <p style="text-decoration:line-through">line through</p>
</body>
(... 생략 ...)
```

텍스트에 줄을 표시하지 않음

밑줄 표시

윗줄 표시

취소선 표시

text-decoration 속성

none

underline

overline

~~line through~~

텍스트 관련 스타일

text-shadow 속성

텍스트에 그림자 효과를 추가하는 속성

기본형 text-shadow: none | <가로 거리> <세로 거리> <번짐 정도> <색상>

종류	설명
<가로 거리>	텍스트부터 그림자까지의 가로 거리로 필수 속성입니다. 양숫값은 글자의 오른쪽, 음숫값은 글자의 왼쪽에 그림자를 만듭니다.
<세로 거리>	텍스트부터 그림자까지의 세로 거리로 필수 속성입니다. 양숫값은 글자의 아래쪽, 음숫값은 글자의 위쪽에 그림자를 만듭니다.
<번짐 정도>	그림자가 번지는 정도입니다. 양숫값을 사용하면 그림자가 모든 방향으로 퍼져 나가므로 그림자가 크게 표시됩니다. 반대로 음숫값은 그림자가 모든 방향으로 축소되어 보입니다. 기본값은 0입니다.
<색상>	그림자 색상을 지정합니다. 한 가지만 지정할 수도 있고 공백으로 구분해 여러 색상을 지정할 수도 있습니다. 기본값은 현재 글자색입니다.

Do it! 텍스트에 그림자 효과 추가하기 예제 파일 07\text-shadow.html

```
(... 생략 ...)
  .shadow1 {
    color: red;
    text-shadow: 1px 1px black;  /* 가로, 세로 거리, 색상 지정 */
  }
  .shadow2 {
    text-shadow: 5px 5px 3px #ffa500;  /* 가로, 세로 거리, 번짐 정도, 색상 지정 */
  }
  .shadow3 {
    color: #fff;
    text-shadow: 7px -7px 20px #000;  /* 가로, 세로 거리, 번짐 정도, 색상 지정 */
  }
</style>
.....
  <h1 class="shadow1">HTML</h1>
  <h1 class="shadow2">CSS</h1>
  <h1 class="shadow3">자바스크립트</h1>
(... 생략 ...)
```

텍스트 관련 스타일

text-transform 속성

영문 텍스트의 대문자나 소문자를 바꾸는 속성

```
기본형  text-transform : none | capitalize | uppercase | lowercase | full-width;
```

속성 값	설명
none	변환하지 않습니다.*
capitalize	시작하는 첫 번째 글자를 대문자로 변환합니다.
uppercase	모든 글자를 대문자로 변환합니다.
lowercase	모든 글자를 소문자로 변환합니다.
full-width	가능한 모든 문자를 전각 문자로 변환합니다.

Do it! 글자 일부 또는 전체를 대문자로 바꾸기 예제 파일 07\text-transform.html

```
(... 생략 ...)
  .trans1{ text-transform: c          ; } /* 첫 번째 글자를 대문자로 */
  .trans2{ text-transform: upper      ; } /* 전체 글자를 대문자로 */
  .trans3{ text-transform: lowercase; }   /* 전체 글자를 소문자로 */
</style>
......
  <p class="trans1">html</p>
  <p class="trans2">css</p>
  <p class="trans3">JAVASCRIPT</p>
(... 생략 ...)
```

letter-spacing, word-spacing 속성

글자간 간격, 단어간 간격

```
기본형  letter-spacing : normal | <크기>
        word-spacing : normal | <크기>
```

Do it! 글자 간격 조절하기

```
(... 생략 ...)
  .spacing1 {
    letter-spacing: 0.2em;          /* 글자 간격 0.2em */
  }
  .spacing2 {
    letter-s          : 0.5em;  /* 글자 간격 0.5em */
  }
</style>
......
  <p>CSS</p>
  <p class="spacing1">CSS</p>
  <p class="spacing2">CSS</p>
(... 생략 ...)
```

목록 스타일

list-style-type 속성

순서 없는 목록의 불릿이나 순서 목록의 숫자를 바꾸는 속성

종류	설명	예시
disc	채운 원 모양입니다.	●
circle	빈 원 모양입니다.	○
square	채운 사각형 모양입니다.	■
decimal	1부터 시작하는 10진수입니다.	1, 2, 3, …
decimal-leading-zero	앞에 0이 붙는 10진수입니다.	01, 02, …
lower-roman	로마 숫자 소문자입니다.	i, ii, iii, …
upper-roman	로마 숫자 대문자입니다.	I, II, III, …
lower-alpha 또는 lower-latin	알파벳 소문자입니다.	a, b, c, …
upper-alpha 또는 upper-latin	알파벳 대문자입니다.	A, B, C, …
none	불릿이나 숫자를 없앱니다.	

Do it! 순서 없는 목록과 알파벳 대문자 목록 지정하기 — 예제 파일 07\list-1.html

```
(... 생략 ...)
  .book1 {
    list-style-type: none;  /* 순서 없는 목록 */
  }
  .book2 {
    list-style-type: u                ;  /* 알파벳 대문자 목록 */
  }
</style>
......
  <h1>도서 시리즈</h1>
  <ul class="book1">
    <li>Do it! 시리즈</li>
    <li>첫 코딩 시리즈</li>
    <li>된다 시리즈</li>
  </ul>
  <ol class="book2">
    <li>Do it! 시리즈</li>
    <li>첫 코딩 시리즈</li>
    <li>된다 시리즈</li>
  </ol>
(... 생략 ...)
```

upper-alpha :답은

도서 시리즈

- Do it! 시리즈
- 첫 코딩 시리즈
- 된다 시리즈

A. Do it! 시리즈
B. 첫 코딩 시리즈
C. 된다 시리즈

목록 스타일

list-style-image 속성

순서 없는 목록의 불릿을 이미지로 바꾸는 속성

기본형 `list-style-image: <url(이미지 파일 경로)> | none`

list-style-position 속성

불릿이나 번호를 들여쓰거나 내어쓸 수 있음

기본형 `list-style-position: inside | outside;`

종류	설명
inside	불릿이나 번호를 기본 위치보다 안으로 들여 씁니다.
outside	기본값입니다.

list-style 속성

list-style-type, list-style-position, list-style-image 속성을 한꺼번에 표기

불릿 없애기

```
ul {list-style-type: none;}
```

속성 줄여서 표시하기

```
ul {list-style: none;}
```

소문자 목록 만들고 들여쓰기

```
ol {
  list-style-type: lower-alpha;
  list-style-position: inside;
}
```

속성 줄여서 표시하기

```
ol {
  list-style: lower-alpha inside;
}
```

표 스타일

caption-side

- 캡션(설명글)은 기본으로 표 위쪽에 표시됨.
- 이 속성을 이용해 아래쪽에 표시 가능

기본형 `caption-side: top | bottom`

border-collapse

표 테두리와 셀 테두리를 합칠 것인지 설정

기본형 `border-collapse: collapse | separate`

속성 값	설명
collapse	테두리를 하나로 합쳐 표시합니다.
separate	테두리를 따로 표시합니다. 기본 값입니다.*

border

표의 바깥 테두리와 셀 테두리 모두 지정해야 함.

```
  table {
    caption-side: bottom;      /* 표 캡
    border: 1px solid black;
  }
  td, th {
    border: 1px dotted b          ;
    padding: 10px;
    text-align: center;
  }
</style>
```

상품 구성

용도	중량	개수	가격
선물용	3kg	11~16과	35,000원
	5kg	18~26과	52,000원
가정용	3kg	11~16과	30,000원
	5kg	18~26과	47,000원

선물용과 가정용 상품 구성

```
(... 생략 ...)
  table {
    caption-side: b          ;
    border: 1px solid black;
    border-collapse: collapse;
  }
  td, th {
    border: 1px dotted black;
    padding: 10px;
    text-align: center;
  }
</style>
```

상품 구성

용도	중량	개수	가격
선물용	3kg	11~16과	35,000원
	5kg	18~26과	52,000원
가정용	3kg	11~16과	30,000원
	5kg	18~26과	47,000원

선물용과 가정용 상품 구성

08. 레이아웃을 구성하는 CSS 박스 모델

08-1	CSS와 박스 모델
08-2	테두리 스타일 지정하기
08-3	여백을 조절하는 속성
08-4	웹 문서 레이아웃 만들기
08-5	웹 요소 위치 지정하기

https://www.w3schools.com/css/default.asp

https://github.com/Eun-Sook-Kim/doit_html_css_javascript

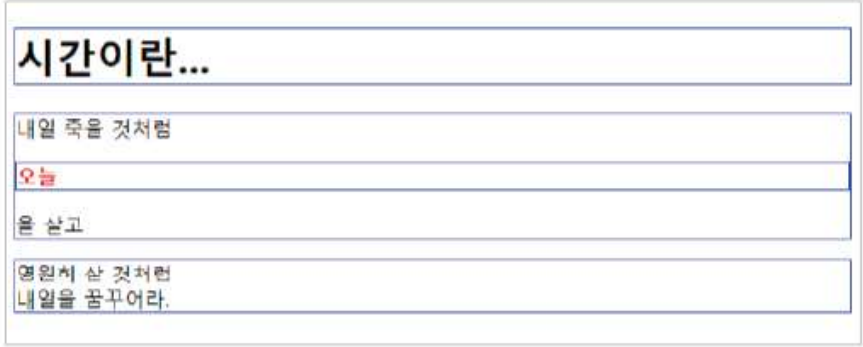
CSS와 박스 모델

블록 레벨 요소

- 요소를 삽입했을 때 혼자 한 줄을 차지하는 요소
- 요소의 너비가 100%
  예) , <p> 등

Do it! 블록 레벨 요소 사용하기 예제 파일 08\block.html

```
(... 생략 ...)
<h1>시간이란...</h1>
<div>내일 죽을 것처럼 <p class="accent">오늘</p>을 살고</div>
<p>영원히 살 것처럼 <br>내일을 꿈꾸어라.</p>
(... 생략 ...)
```

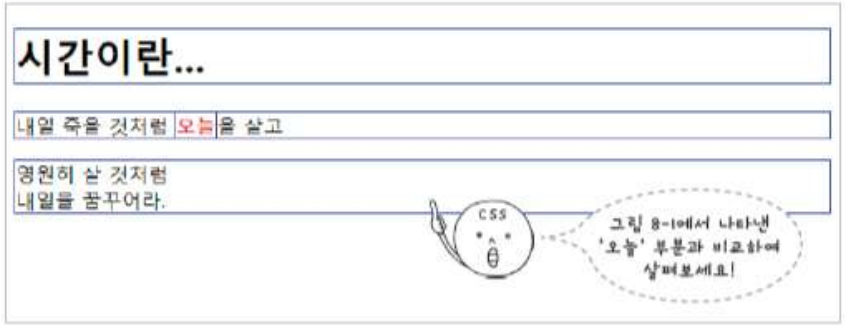


인라인 레벨 요소

- 줄을 차지하지 않는 요소
- 화면에 표시되는 콘텐츠만큼만 영역을 차지하고 나머지 공간에는 다른 요소가 올 수 있음
  예) <img>, 등

Do it! 인라인 레벨 요소 예제 파일 08\inline.html

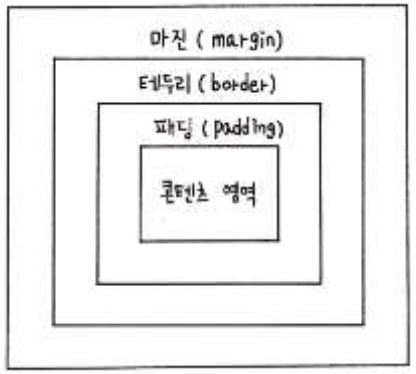
```
(... 생략 ...)
<h1>시간이란...</h1>
<div>내일 죽을 것처럼 <span class="accent">오늘</span>을 살고</div>
<p>영원히 살 것처럼 <br>내일을 꿈꾸어라.</p>
(... 생략 ...)
```



CSS와 박스 모델

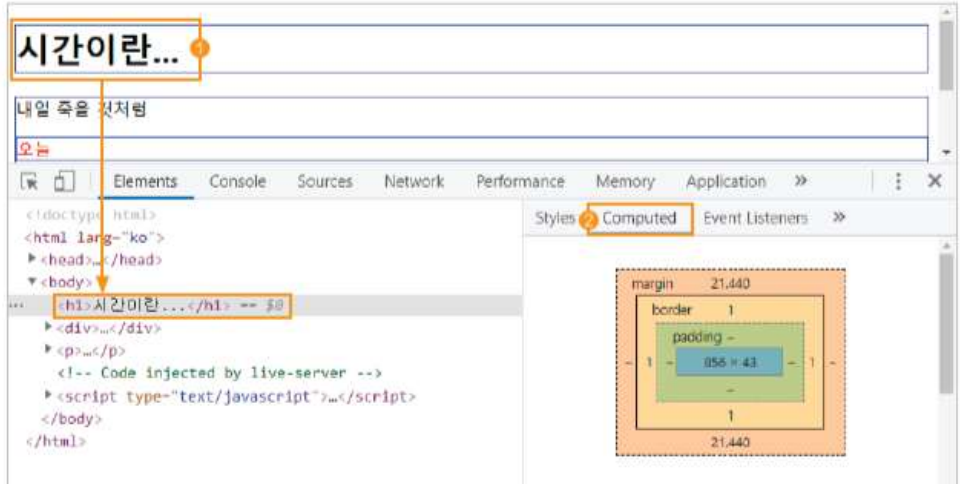
박스 모델

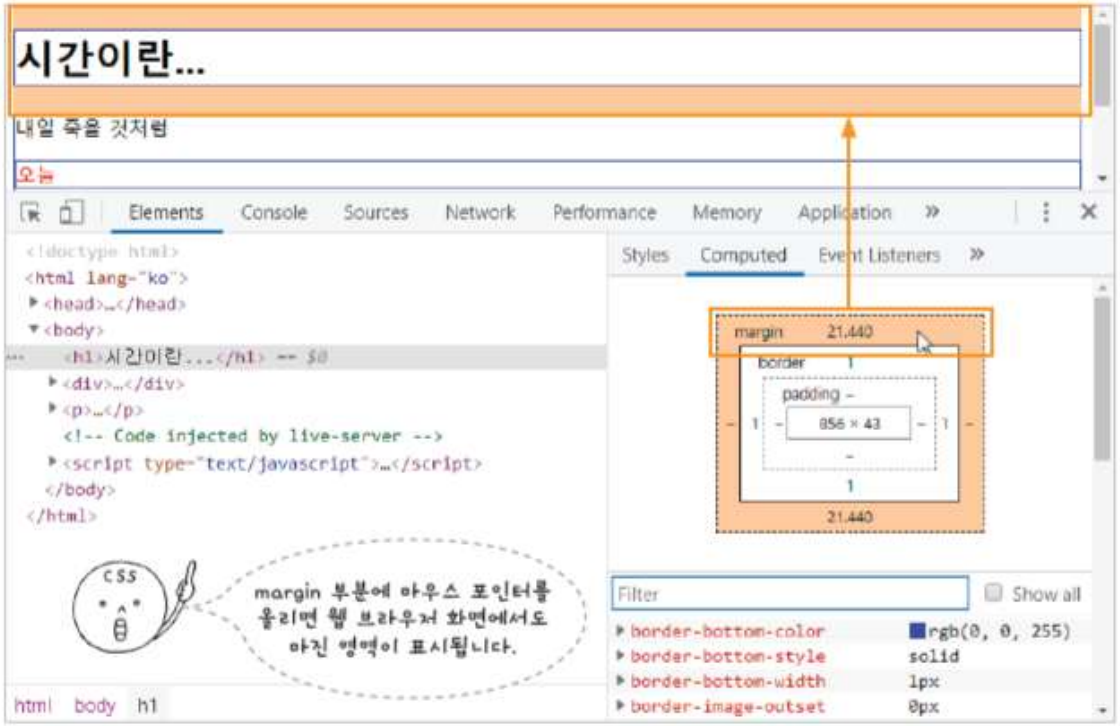
실제 콘텐츠 영역, 패딩(padding), 박스의 테두리(border), 그리고 마진(margin) 등의 요소로 구성됨.



개발자 도구 창에서 박스 모델 확인 가능

① 웹 브라우저에서 웹 문서 열기
② 박스 모델 확인할 부분을 마우스 오른쪽 버튼으로 누른 뒤 [검사] 선택
③ 개발자 도구 창 위에서 [Computed] 탭 클릭
④ 해당 부분의 박스 모델이 그림으로 표시됨





CSS와 박스 모델

width, height 속성

실제 콘텐츠 영역의 크기 지정

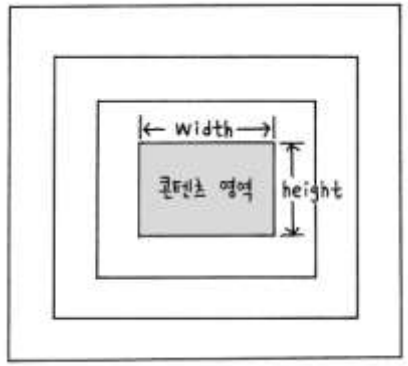
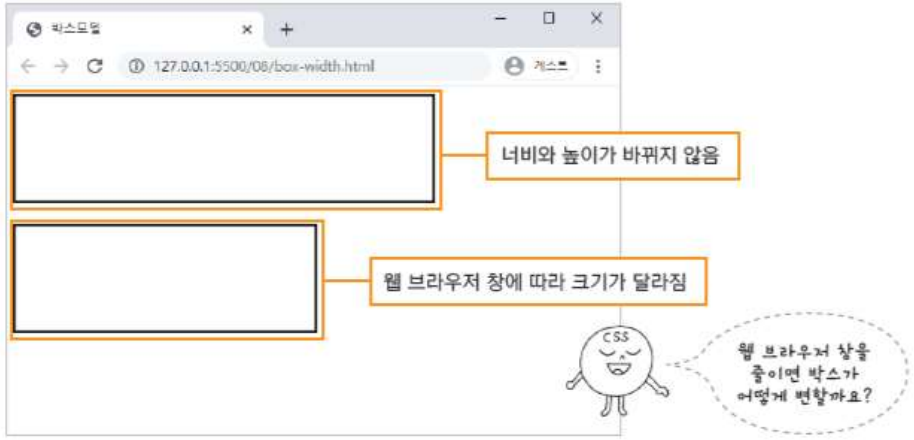


표 8-1 width와 height의 속성값

종류	설명
<크기>	너비나 높이의 값을 px이나 em 단위로 지정합니다.
<백분율>	박스 모델을 포함하는 부모 요소를 기준으로 너빗값이나 높잇값을 백분율(%)로 지정합니다.
auto	박스 모델의 너빗값과 높잇값이 콘텐츠 양에 따라 자동으로 결정됩니다. 기본값입니다.

Do it! 박스 모델 너비 지정하기 Box-width.html

```
(... 생략 ...)
  .box1 {
    width: 400px;   /* 고정 너비*/
    height: 100px;  /* 고정 높이*/
  }
  .box2 {
    w          : 50%;    /* 가변 너비 - 웹 브라우저의 50% */
    h          : 100px;  /* 고정 높이 */
  }
</style>
......
  <div class="box1"></div>
  <div class="box2"></div>
(... 생략 ...)
```



CSS와 박스 모델

box-sizing 속성

실제 박스 모델의 너비를 계산할 때 어디까지 포함할지 결정하는 속성

종류	설명
border-box	테두리까지 포함해서 너빗값을 지정합니다.
content-box	콘텐츠 영역만 너빗값을 지정합니다. 기본값입니다.

Do it! 박스 모델에 그림자 효과 주기 — 예제 파일 08\box-shadow.html

```
(... 생략 ...)
  .box1 { box-shadow: 2px -2px 5px 0px; } ❶
  .box2 { box-s                : 5px 5px 15px 5px blue; } ❷
</style>
(... 생략 ...)
```

box-shdow 속성

선택한 요소에 그림자 효과 내기

수평 거리와 수직 거리는 필수, 기타 속성은 옵션

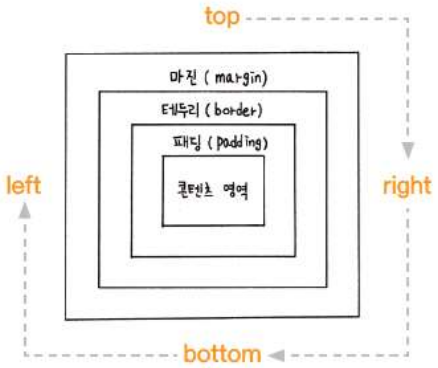
기본형 box-shadow: <수평 거리> <수직 거리> <흐림 정도> <번짐 정도> <색상> inset

종류	설명
<수평 거리>	그림자가 가로로 얼마나 떨어져 있는지를 나타냅니다. 양숫값은 요소의 오른쪽에, 음숫값은 요소의 왼쪽에 그림자를 만듭니다. 필수 속성입니다.
<수직 거리>	그림자가 세로로 얼마나 떨어져 있는지를 나타냅니다. 양숫값은 요소의 아래쪽에, 음숫값은 요소의 위쪽에 그림자를 만듭니다. 필수 속성입니다.
<흐림 정도>	이 값을 생략하면 0을 기본값으로 하여 진한 그림자를 표시합니다. 이 값이 커질수록 부드러운 그림자를 표시하며, 음숫값은 사용할 수 없습니다.
<번짐 정도>	양숫값을 사용하면 모든 방향으로 그림자가 퍼져서 박스보다 그림자가 크게 표시됩니다. 반대로 음숫값은 모든 방향으로 그림자가 축소되어 보입니다. 기본값은 0입니다.
<색상>	한 가지만 지정할 수도 있고, 공백으로 구분해서 여러 개의 색상을 지정할 수도 있습니다. 기본값은 현재 검은색입니다.
inset	이 키워드를 함께 표시하면 안쪽 그림자로 그립니다.

테두리 스타일 지정하기

박스 모델의 값 지정 방향

- 4개 방향의 값을 한꺼번에 지정할 때는 방향 순서를 지켜야 함
- top → right → bottom → left



border-style 속성

- 기본 값이 none → 화면에 테두리 표시안됨
- 테두리를 그리기 위해서는 맨 먼저 테두리 스타일부터 지정

기본형 `border-style: none | hidden | dashed | dotted | double | groove | inset | outset | ridge | solid`

종류	설명	예시
none	테두리가 없습니다. 기본값입니다.	
hidden	테두리를 감춥니다. 표에서 border-collapse: collapse일 경우 다른 테두리도 표시되지 않습니다.	
solid	테두리를 실선으로 표시합니다.	
dotted	테두리를 점선으로 표시합니다.	
dashed	테두리를 짧은 직선으로 표시합니다.	
double	테두리를 이중선으로 표시합니다. 두 선 사이의 간격이 border-width값이 됩니다.	
groove	테두리를 창에 조각한 것처럼 표시합니다. 홈이 파인 듯 입체 느낌이 납니다.	
inset	표에서 border-collapse: seperate일 경우 전체 박스 테두리가 창에 박혀 있는 것처럼 표시되고, 표에서 border-collapse: collapse일 경우 groove와 똑같이 표시됩니다.	
outset	표에서 border-collapse: seperate일 경우 전체 박스 테두리가 창에서 튀어나온 것처럼 표시되고, 표에서 border-collapse: collapse일 경우 ridge와 똑같이 표시됩니다	
ridge	테두리를 창에서 튀어나온 것처럼 표시합니다.	

Border-1.html

테두리 스타일 지정하기

border-width 속성

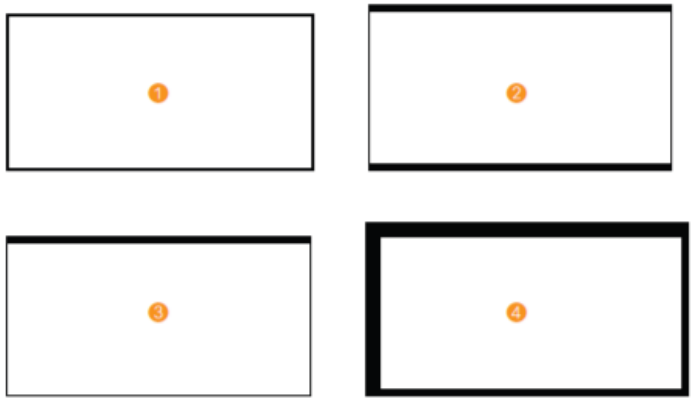
thin < medium < thick순으로 두꺼워짐

테두리 두께 지정

```
기본형 border-width: <크기> | thin | medium | thick
```

Do it! 박스 모델의 테두리 두께 지정하기 Border-2.html

```
(... 생략 ...)
#box1 { border-width: 2px; } ❶
#box2 { border-width: thick thin; } ❷
#box3 { border-width: thick thin thin; } ❸
#box4 { border-width: 10px 5px 5px 10px; } ❹
(... 생략 ...)
```



① 4개 방향 테두리 모두 같은 값이 적용됨

② top → right → bottom으로 적용하다가
마지막 left 속성값이 빠짐.
left 속성값은 마주 보는 right 속성값과 똑같이 적용됨

③ 첫 번째 값인 thick이 top, bottom 값이 되고,
두 번째 값 thin은 left, right 값이 됨

④ top → right →bottom → left순으로 적용됨

테두리 스타일 지정하기

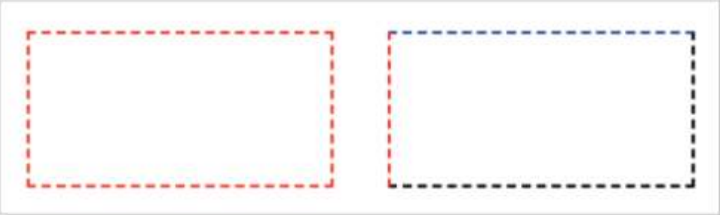
border-color 속성

테두리 색상 지정

Do it! 박스 모델의 테두리 색상 지정하기 — 예제 파일 08\border-3.html

```
(... 생략 ...)
#box1 {
  border-color: red;                /* 전체 테두리 빨강 */
}
#box2 {
  border-t                -color: blue; /* 위쪽 테두리 파랑 */
  border-left-color: red;           /* 왼쪽 테두리 빨강 */
}
(... 생략 ...)
```

정답: top



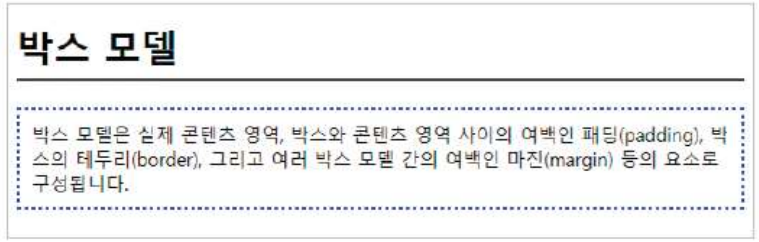
border 속성

- 테두리 스타일과 두께, 색상 등을 묶어 표기
- border-top이나 border-right처럼 방향을 함께 써서 4개 방향의 스타일을 따로 지정할 수 있음
- 순서는 상관없음

Do it! 테두리 스타일 묶어서 지정하기 — 예제 파일 08\border-4.html

```
(... 생략 ...)
h1 {
  padding-bottom: 5px;
  border-b          : 3px solid rgb(75, 70, 70); /* 아래쪽 테두리 3px 회색 실선 */
}
p {
  padding: 10px;
  border: 3px dotted blue;  /* 모든 테두리를 3px 파란색 점선 */
}
(... 생략 ...)
```

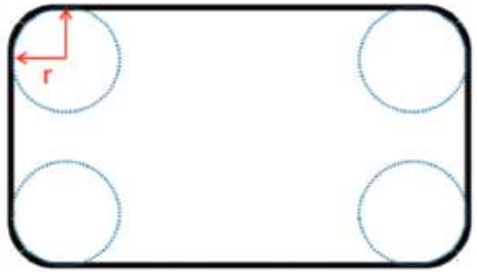
정답: bottom



테두리 스타일 지정하기

border-radius 속성

- 박스 모델의 테두리를 둥글게 처리
- 박스 모델의 꼭짓점 부분에 원(반지름 r)이 있다고 가정해서 둥글게 처리



기본형 `border-radius: <크기> | <백분율>`

종류	설명
<크기>	반지름 크기를 px, em의 단위와 함께 수치로 표시합니다
<백분율>	현재 요소의 크기를 기준으로 비율(%)로 지정합니다.

Border-5.html

Do it! 이미지의 꼭짓점을 둥글게 표시하기

```
(... 생략 ...)
  #round { border-r          : 25px; }
</style>
......
<img src="images/cat.jpg">
<img id="round" src="images/cat.jpg">
(... 생략 ...)
```

Border-6.html

Do it! 이미지를 원형으로 표시하기

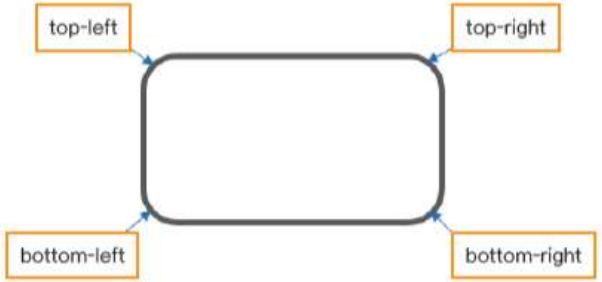
```
(... 생략 ...)
  #circle { border-r          : 50%; }
</style>
......
<img src="images/photo.jpg">
<img id="circle" src="images/photo.jpg">
(... 생략 ...)
```

테두리 스타일 지정하기

border-radius 속성

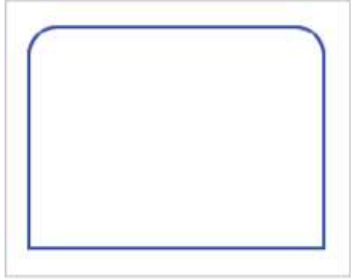
border와 radius 사이에 위치를 나타내는 예약어를 사용하면 꼭짓점마다 다르게 처리 가능

각 꼭짓점을 나타내는 예약어



Do it! 원하는 꼭짓점만 둥글게 처리하기 Border-7.html

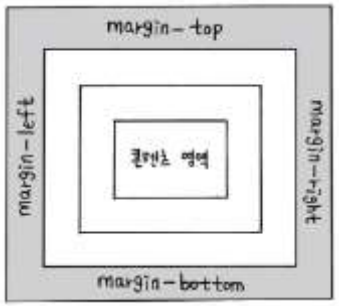
```
(... 생략 ...)
#round1 {
  border: 2px solid blue;
  border-top-left-radius: 20px;
  border-top-right-radius: 20px;
}
(... 생략 ...)
```



여백을 조절하는 속성

margin 속성

- 현재 요소 주변의 여백
- 마진을 이용하면 요소와 요소 간의 간격 조절 가능



기본형 `margin: <크기> ¦ <백분율> ¦ auto`

종류	설명	예시
<크기>	너빗값이나 높잇값을 px이나 em 같은 단위와 함께 수치로 지정합니다.	margin: 50px;
<백분율>	박스 모델을 포함한 부모 요소를 기준으로 너빗값이나 높잇값을 퍼센트(%)로 지정합니다.	margin: 0.1%;
auto	display 속성에서 지정한 값에 맞게 적절한 값을 자동으로 지정합니다.	

margin 속성을 사용해 가운데 정렬하기

- 배치할 요소의 너빗값이 정해져 있어야 함
- margin-left와 margin-right의 속성값을 auto로 지정

Do it! 웹 문서 전체를 웹 브라우저 화면 중앙에 배치하기 — 예제 파일 08\margin-3.html

```
(... 생략 ...)
<style>
  body { background-color: #222; }  /* 배경색은 어둡게 지정 */
  #container {
    background-color: #fff;
    width: 600px;                   /* 가로 너비 600px */
    m                : 20px auto;   /* 위아래 마진은 20px씩, 좌우 마진은 자동 계산 */
    padding: 20px;
  }
......
<body>
  <div id="container">
    <h1>레드향</h1>
    <p>껍질에 붉은 빛이 돌아 <b>레드향</b>이라 불린다.</p>
    ......
 </div>
(... 생략 ...)
```

정답: margin

여백을 조절하는 속성

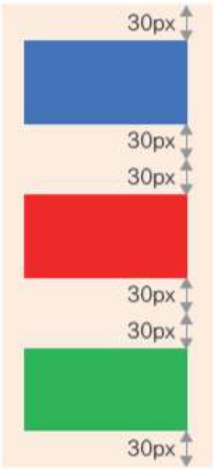
마진 중첩 현상

- 요소를 세로로 배치할 경우,
  마진과 마진이 만날 때 마진 값이 큰 쪽으로 겹쳐지는 것
- 요소를 가로로 배치할 경우에는 상관없음

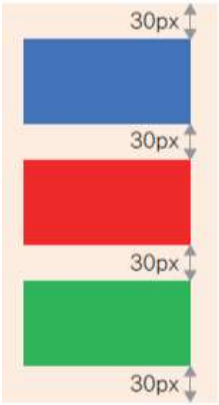
```
div {
    width:200px; /* 너비 */
    height:100px; /* 높이 */
    margin:30px; /* 마진 */
}
#box1 { background:rgb(0, 77, 243); }
#box2 { background:rgb(255, 72, 0); }
#box3 { background:rgb(18, 219, 0); }

<div id="box1"></div>
<div id="box2"></div>
<div id="box3"></div>
```

Margin-overlab.html

예상 모습

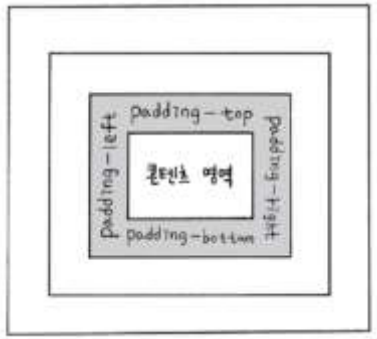


실제 화면



padding 속성

- 콘텐츠 영역과 테두리 사이의 여백
- 마진을 지정하는 방법과 같다



웹 문서 레이아웃 만들기

display 속성

사용할 수 있는 속성 값이 아주 많다

요소의 배치 방법 결정.

블록 레벨 요소, 인라인 레벨 요소를 바꿀 때 사용하는 속성 값

종류	설명
block	인라인 레벨 요소를 블록 레벨 요소로 만듭니다.
inline	블록 레벨 요소를 인라인 레벨 요소로 만듭니다.
inline-block	인라인 레벨 요소와 블록 레벨 요소의 속성을 모두 가지고 있으며 마진과 패딩을 지정할 수 있습니다.
none	해당 요소를 화면에 표시하지 않습니다.

Display.html

```
<style>
  * { box-sizing: border-box; }  /* 모든 요소에 테두리까지 포함한 너빗값 지정 */
  nav ul { list-style: none; }   /* 불릿이나 숫자 번호 없앰 */
  nav ul li {
    display: inline-block;   /* 인라인 레벨 요소와 블록 레벨 요소 모두 지정 */
    padding: 20px;           /* 패딩값 20px */
    margin: 0 20px;          /* 마진값 20px */
    border: 1px solid #222;  /* 메뉴의 테두리 지정 */
  }
```

웹 문서 레이아웃 만들기

float 속성

요소를 왼쪽이나
오른쪽에 떠 있게 만듦

기본형 `float: left | right | none`

속성 값	설명
left	해당 요소를 문서의 왼쪽으로 배치합니다.
right	해당 요소를 문서의 오른쪽으로 배치합니다.
none	좌우 어느 쪽으로도 배치하지 않습니다.

Do it! 텍스트 왼쪽에 이미지 배치하기 Float-1.html

```
(... 생략 ...)
<style>
  img {
    float: l          ;   /* 왼쪽에 떠 있게 */
    margin-right: 40px;   /* 오른쪽 마진 40px */
  }
......
<img src="images/tree.png">
<p>Ex et adipisicing ......</p>
(... 생략 ...)
```

clear 속성

float 속성을 무효화 시키는 속성

기본형 `clear: none | left | right | both`

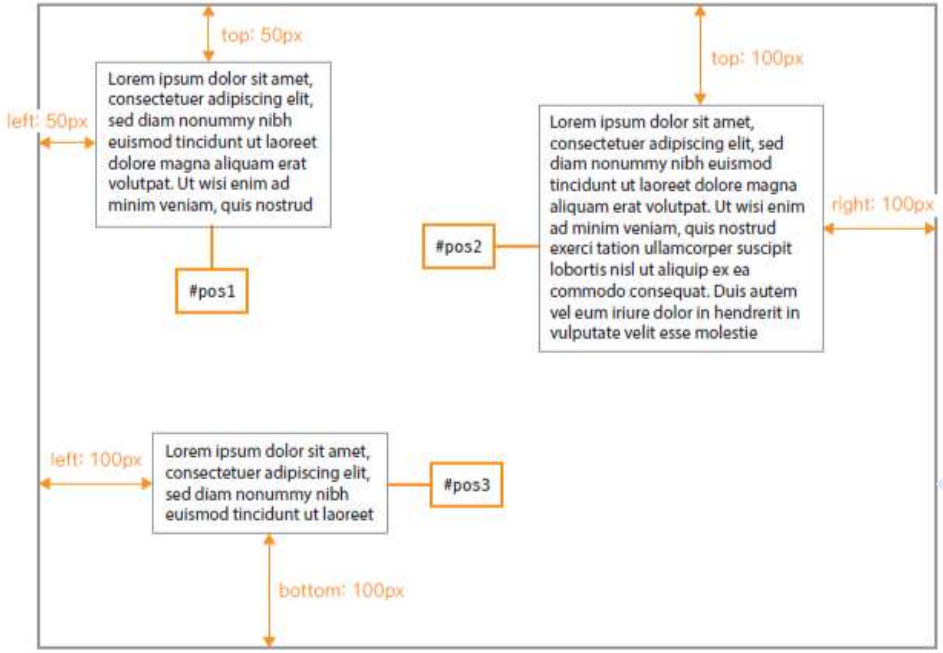
종류	설명
left	float: left를 해제합니다.
right	float: right를 해제합니다.
both	float: left와 float: right를 해제합니다.

웹 요소의 위치 지정하기

left, right, bottom, top 속성

웹 요소의 위치를 지정하는 속성

종류	설명
left	기준 위치와 요소 사이에 왼쪽으로 얼마나 떨어져 있는지 지정합니다.
right	기준 위치와 요소 사이에 오른쪽으로 얼마나 떨어져 있는지 지정합니다.
top	기준 위치와 요소 사이에 위쪽으로 얼마나 떨어져 있는지 지정합니다.
bottom	기준 위치와 요소 사이에 아래쪽으로 얼마나 떨어져 있는지 지정합니다.



Do it! 텍스트 요소 자유롭게 배치하기 — 예제 파일 08\left-top.html

```
(... 생략 ...)
 #pos1 {
    position: absolute;  /* 포지셔닝 - absolute */
    left: 50px;          /* 왼쪽에서 50px 떨어지게 */
    top: 50px;           /* 위쪽에서 50px 떨어지게 */
 }
 #pos2 {
    position: absolute;  /* 포지셔닝 - absolute */
    right: 100px;        /* 오른쪽에서 100px 떨어지게 */
    top: 100px;          /* 위쪽에서 100px 떨어지게 */
 }
 #pos3 {
    position: absolute;  /* 포지셔닝 - absolute */
    left: 100px;         /* 왼쪽에서 100px 떨어지게 */
    bottom: 100px;       /* 아래쪽에서 100px 떨어지게 */
 }
......
  <p id="pos1">Ex et adipisicing ......</p>
  <p id="pos2">Lorem ipsum reprehenderit ......</p>
  <p id="pos3">Excepteur voluptate ......</p>
(... 생략 ...)
```

https://www.w3schools.com/css/css_positioning.asp

웹 요소의 위치 지정하기

position 속성

웹 문서 안에 요소들을 배치하기 위한 속성

기본형 `position: static | relative | absolute | fixed`

속성 값	설명
static	요소를 문서의 흐름에 맞추어 배치합니다.*
relative	이전 요소에 자연스럽게 연결해 배치하되 위치를 지정할 수 있습니다.
absolute	원하는 위치를 지정해 배치합니다.
fixed	지정한 위치에 고정해 배치합니다. 화면에서 요소가 잘릴 수도 있습니다.

The position Property

The `position` property specifies the type of positioning

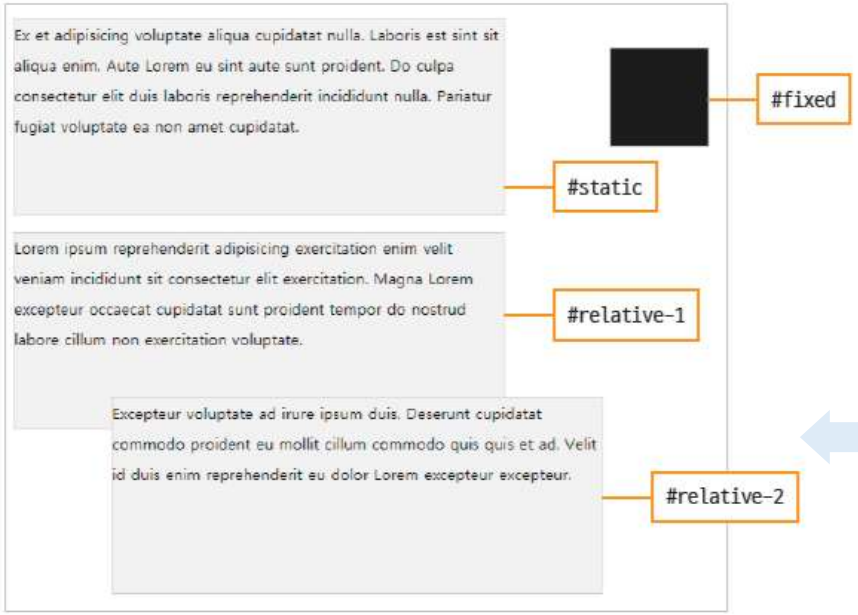
There are five different position values:

- `static`
- `relative`
- `fixed`
- `absolute`
- `sticky`

Do it! position 속성으로 요소 위치 지정하기 — 예제 파일 08\position-1.html

```
(... 생략 ...)
  #static { position: static; }
  #relative-1 { position: relative; }
  #relative-2 {
    p                    : relative;  /* 포지셔닝 - absolute */
    left: 100px;                      /* 왼쪽에서 100px 떨어지게 */
    top: -50px;                       /* 위쪽에서 -50px 떨어지게(위로 이동) */
  }
  #fixed {
    width: 100px;
    height: 100px;
    background-color: #222;
    position: fixed;                  /* 포지셔닝 - fixed */
    right: 30px;                      /* 오른쪽에서 30px 떨어지게 */
    top: 30px;                        /* 위쪽에서 30px 떨어지게 */
  }
</style>
......
  <p id="static">Ex et adipisicing ..... </p>
  <p id="relative-1">Lorem ipsum reprehenderit ..... </p>
  <p id="relative-2">Excepteur voluptate ad ..... </p>
  <p id="fixed"></p>
(... 생략 ...)
```

정답: position



Display 속성

https://www.w3schools.com/cssref/pr_class_display.asp

inline : inline element로 취급
block : block element로 취급
flex : block-level flex 컨테이너로 취급
grid : block-level grid 컨테이너로 취급
inline-block : inline-level block 컨테이너로 취급
inline-flex : inline-level flex 컨테이너로 취급
inline-grid : inline-level grid 컨테이너로 취급
……
매우 많음

CSS Syntax

```
display: value;
```

Property Values

Value	Description
inline	Displays an element as an inline element (like). Any height a
block	Displays an element as a block element (like <p>). It starts on a new
contents	Makes the container disappear, making the child elements children of
flex	Displays an element as a block-level flex container
grid	Displays an element as a block-level grid container
inline-block	Displays an element as an inline-level block container. The element its can apply height and width values
inline-flex	Displays an element as an inline-level flex container
inline-grid	Displays an element as an inline-level grid container
inline-table	The element is displayed as an inline-level table
list-item	Let the element behave like a <li> element
run-in	Displays an element as either block or inline, depending on context
table	Let the element behave like a <table> element
table-caption	Let the element behave like a <caption> element
table-column-group	Let the element behave like a <colgroup> element
table-header-group	Let the element behave like a <thead> element
table-footer-group	Let the element behave like a <tfoot> element
table-row-group	Let the element behave like a <tbody> element

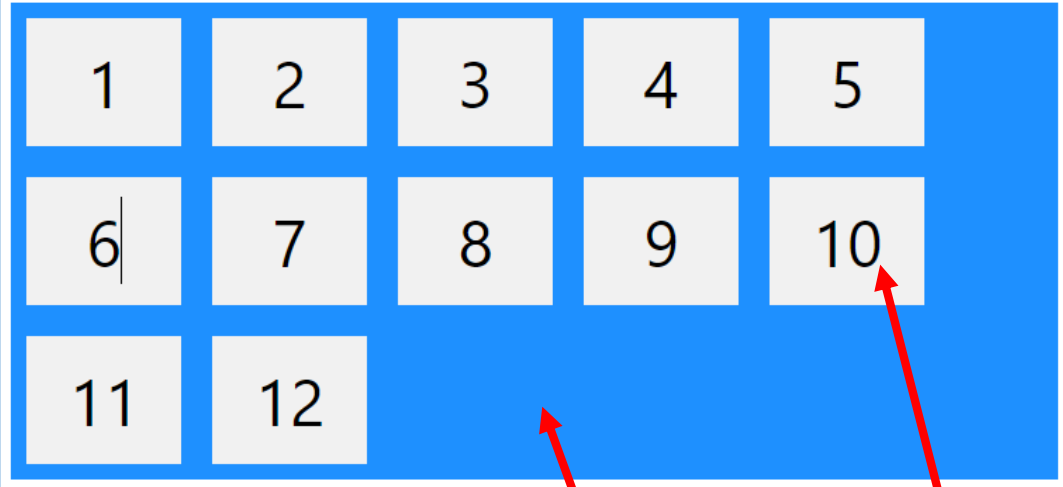
Flexbox Layout

Flexbox란?
- 유연한 반응형 레이아웃을 쉽게 디자인할 수 있도록 개발된 레이아웃

Display 속성을 flex로 지정한 컨테이너 내부에 child element들을 넣으면 flexible item으로 취급

```
<!DOCTYPE html>
<html>
<head>
<style>
.flex-container {
  display: flex;
  background-color: DodgerBlue;
}
.flex-container > div {
  background-color: #f1f1f1;
  margin: 10px;
  padding: 20px;
  font-size: 30px;
}
</style>
</head>
<body>

<div class="flex-container">
  <div>1</div>
  <div>2</div>
  <div>3</div>
</div>

<p>A Flexible Layout must have a parent element with the <em>display</em> property set to <em>flex</em>.</p>
<p>Direct child elements(s) of the flexible container automatically becomes flexible items.</p>
</body>
</html>
```

https://www.w3schools.com/css/css3_flexbox.asp



Flex 아이템

Flex 컨테이너

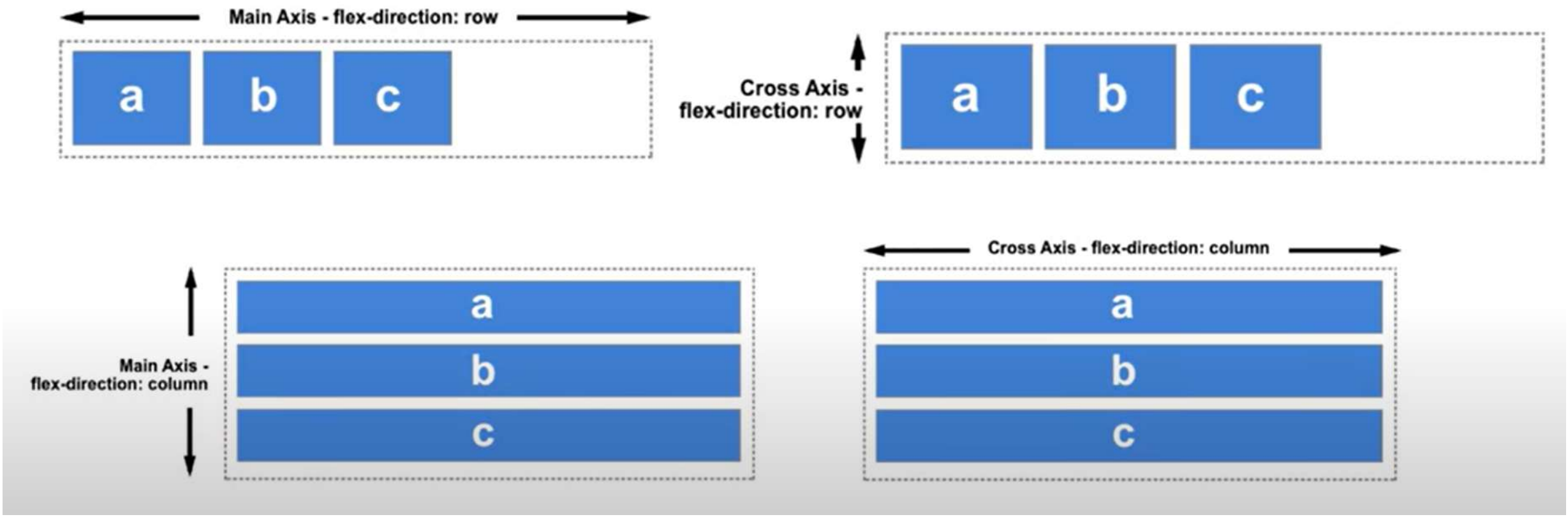
Flex Containers & Flex Items

```
<div class="flex-container">
    <div>Flex Item 1</div>
    <div>Flex Item 2</div>
    <img src="pic.png" alt="also a flex item" />
</div>
```

Flex Container Properties	Flex Item Properties
flex-direction	order
flex-wrap	flex-grow
flex-flow	flex-shrink
justify-content	flex-basis
align-items	align-self
align-content	

https://www.youtube.com/watch?v=3YW65K6LcIA

Axes


Main Axis - flex-direction: row
a
b
c
Cross Axis - flex-direction: row
a
b
c
Main Axis - flex-direction: column
a
b
c
Cross Axis - flex-direction: column
a
b
c

Flexbox

Flex 컨테이너의 속성

flex-direction : column, row, column-reverse, row-reverse (아이템들을 쌓는 방향)
flex-wrap : wrap, nowrap, wrap-reverse (컨테이너를 넘치는 경우의 처리 방법)
Flex-flow : row wrap, … (flex-direction과 flex-wrap을 함께 설정하는 방법)
Justify-content : center, flex-start, flex-end, space-around, space-between (flex 아이템들의 정렬 방법)
align-items : center, flex-start, flex-end, stretch, baseline (flex 아이템들의 정렬 방법)
align-content : space-around, space-between, stretch, center (flex 라인의 정렬 방법)

Flex 아이템의 속성

order : 아이템의 순서 지정
flex-grow : 아이템의 크기 지정
flex-shrink : 축소 필요시의 축소 정도 지정
flex-basis : 아이템의 초기 길이 지정
flex : flex-grow, flex-shrink, flex-basis 를 한번에 지정
align-self : 한 아이템의 정렬 지정

Grid layout

Grid란?
- 열(row)과 행(column)으로 이루어진 그리드 기반의 레이아웃
Display 속성을 grid로 지정한 컨테이너 내부에 child element들을 넣으면 grid item으로 취급

```
<!DOCTYPE html>
<html>
<head>
<style>
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto;
  background-color: #2196F3;
  padding: 10px;
}
.grid-item {
  background-color: rgba(255, 255, 255, 0.8);
  border: 1px solid rgba(0, 0, 0, 0.8);
  padding: 20px;
  font-size: 30px;
  text-align: center;
}
</style>
</head>
<body>
<h1>The display Property:</h1>

<div class="grid-container">
  <div class="grid-item">1</div>
  <div class="grid-item">2</div>
  <div class="grid-item">3</div>
  <div class="grid-item">4</div>
  <div class="grid-item">5</div>
  <div class="grid-item">6</div>
  <div class="grid-item">7</div>
  <div class="grid-item">8</div>
  <div class="grid-item">9</div>
</div>
<p>Set the <em>display</em> property to <em>grid</em> to make a block-level grid container.</p>
</body>
</html>
```

https://www.w3schools.com/css/css_grid.asp

Grid Elements

A Grid Layout must have a parent element with the *display* property set to *grid* or *inline-grid*.

Direct child element(s) of the grid container automatically becomes grid items.

1	2	3
4	5	6
7	8	9

Grid Containers & Grid Items

```
<div class="grid-container">
    <div>Grid Item 1</div>
    <div>Grid Item 2</div>
    <img src="pic.png" alt="also a grid item" />
</div>
```

Grid Container Properties	Grid Item Properties
grid-template columns	grid-column
grid-template-rows	grid-row
grid-gap	grid-area
grid-auto-rows	align-self
grid-auto-columns	justify-self
grid-template-areas	
justify-content	
align-items	

https://www.youtube.com/watch?v=0xMQfnTU6oo

Grid layout

All CSS Grid Properties

Property	Description
column-gap	Specifies the gap between the columns
gap	A shorthand property for the *row-gap* and the *column-gap* properties
grid	A shorthand property for the *grid-template-rows*, *grid-template-columns*, *grid-template-areas*, *grid-auto-rows*, *grid-auto-columns*, and the *grid-auto-flow* properties
grid-area	Either specifies a name for the grid item, or this property is a shorthand property for the *grid-row-start*, *grid-column-start*, *grid-row-end*, and *grid-column-end* properties
grid-auto-columns	Specifies a default column size
grid-auto-flow	Specifies how auto-placed items are inserted in the grid
grid-auto-rows	Specifies a default row size
grid-column	A shorthand property for the *grid-column-start* and the *grid-column-end* properties
grid-column-end	Specifies where to end the grid item
grid-column-gap	Specifies the size of the gap between columns
grid-column-start	Specifies where to start the grid item
grid-gap	A shorthand property for the *grid-row-gap* and *grid-column-gap* properties
grid-row	A shorthand property for the *grid-row-start* and the *grid-row-end* properties

Grid 컨테이너 속성

grid-template-columns : 컬럼의 개수, 넓이
grid-template-rows : 열의 높이
justify-content : 컨테이너 내 그리드의 가로 정렬
  space-evenly, space-around, space-between, center, start, end
align-content : 그리드의 세로 정렬

09. 이미지와 그러데이션 효과로 배경 꾸미기

09-1 배경색과 배경 범위 지정하기

09-2 배경 이미지 지정하기

09-3 그러데이션 효과로 배경 꾸미기

배경색과 배경 범위 지정하기

background-color 속성

- 웹 문서의 요소에 배경색 지정
- 16진수나 rgb 값, rgba 값 또는 색상 이름 사용
- 배경색은 상속되지 않는다

background-clip 속성

배경을 어디까지 적용할지 지정

기본형 `background-clip: border-box | padding-box | content-box`

속성 값	설명
border-box	박스 모델의 가장 외곽인 테두리(border)까지 적용합니다.
padding-box	박스 모델에서 테두리를 뺀 패딩(padding) 범위까지 적용합니다.
content-box	박스 모델에서 내용 부분에만 적용합니다.

Do it! 배경 적용 범위 지정하기 예제 파일 09\bg-clip.html

```
(... 생략 ...)
<style>
  #clip-border { background-clip: border-box; }     /* 테두리까지 배경 지정 */
  #clip-padding { background-clip: p          ; } /* 테두리를 뺀 패딩 범위까지 배경 지정 */
  #clip-content { b           : content-box; } /* 콘텐츠에만 배경 지정 */
</style>
......
<div id="container">
  <div class="desc" id="clip-border"> ..... </div>
  <div class="desc" id="clip-padding"> ..... </div>
  <div class="desc" id="clip-content"> ..... </div>
</div>
(... 생략 ...)
```

정답: padding-box, background-clip

배경색과 배경 범위 지정하기

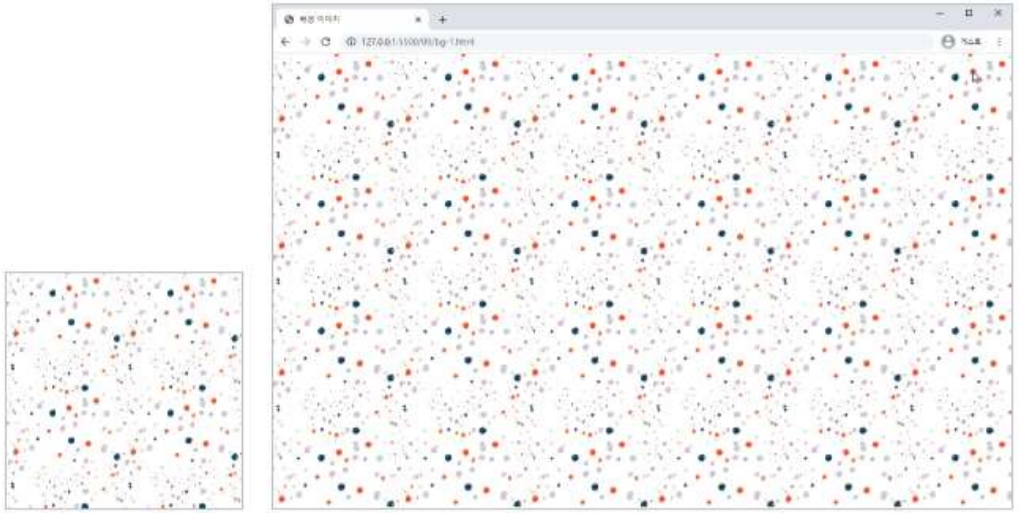
background-image 속성

배경 이미지 파일 경로 지정

기본형 `background-image: url(파일 경로)`

Do it! 기본적인 배경 이미지 넣기 예제 파일 09\bg-1.html

```
(... 생략 ...)
<style>
  body { background-image: url('images/bg1.jpg'); }
</style>
(... 생략 ...)
```



background-repeat 속성

배경 이미지 반복 여부 및 반복 방향 지정

기본형 `background-repeat : repeat | repeat-x | repeat-y | no-repeat`

속성 값	설명
repeat	브라우저 화면에 가득 찰 때까지 배경 이미지를 가로와 세로로 반복합니다. ★
repeat-x	브라우저 창 너비와 같아질 때까지 배경 이미지를 가로로 반복합니다.
repeat-y	브라우저 창 높이와 같아질 때까지 배경 이미지를 세로로 반복합니다.
no-repeat	배경 이미지를 한 번만 표시하고 반복하지 않습니다.

배경색과 배경 범위 지정하기

background-position 속성

배경 이미지를 반복하지 않을 경우, 배경 이미지를 표시할 위치 지정

```
기본형  background-position: <수평 위치> <수직 위치>;
        수평 위치 : left | center | right | <백분율> | <길이 값>
        수직 위치 : top | center | bottom | <백분율> | <길이 값>
```

① **백분율 :** 배경 이미지의 가로 위치와 세로 위치를 %로 나타냄.
예) background-postion: 0% 0% , background-position : 30% 60%

② **길이 길이 :** 배경 이미지의 위치를 직접 길이로 지정
예) background-position:30px 20px;

③ **키워드** - top, left, center, right, top, middle, bottom
가로 배치는 **left와 center, top** 중에서 선택
세로 배치는 **top**과 **bottom, center** 중에서 선택
예) background-position:center bottom

Do it! 불릿 대신 배경 이미지 사용하기 — 예제 파일 09\bg-2.html

```
(... 생략 ...)
<style>
  ul {
    list-style-type: none; /* 불릿 없앰 */
    margin-left: -30px;    /* 왼쪽 여백 -30px */
  }
  li {
    background-image: u          ('images/book-icon.png'); /* 배경 이미지 삽입 */
    background-repeat: no-repeat;                          /* 배경 이미지 반복 안 함 */
    background- p                : left center; /* 배경 이미지 위치 */
    padding-left: 50px; /* 왼쪽 패딩 */
    line-height: 40px;  /* 줄간격 */
  }
</style>
......
<h1>이지스퍼블리싱</h1>
<ul>
  <li>회사소개</li>
  <li>도서</li>
  <li>자료실</li>
  <li>동영상강의</li>
</ul>
(... 생략 ...)
```

정답: url, position

배경색과 배경 범위 지정하기

background-origin 속성

배경 이미지를 배치하기 위한 기준 설정

기본형 `background-origin: border-box | padding-box | content-box`

속성 값	설명
border-box	박스 모델의 가장 외곽인 테두리(border)가 기준이 됩니다.
padding-box	박스 모델에서 테두리를 뺀 패딩(padding)이 기준이 됩니다.
content-box	박스 모델에서 내용 부분이 기준이 됩니다.

background-attachment 속성

배경 이미지를 고정하는 속성

기본형 `background-attachment: scroll | fixed`

속성	설명
scroll	화면 스크롤과 함께 배경 이미지도 스크롤됩니다. 기본 값입니다.*
fixed	화면이 스크롤되더라도 배경 이미지는 고정됩니다.

background 속성

- 배경 관련 속성을 줄여서 표기
- 각 속성 값이 다르므로 표기 순서는 상관없음

배경 이미지 정의하기

```
body {
  background-image: url('images/bg4.png');
  background-repeat: no-repeat;
  background-position: center bottom;
  background-attachment: fixed;
}
```

↓

```
background: url('images/bg4.png') no-repeat center bottom fixed;
```

TIP 속성값이 다르므로 입력 순서는 상관없습니다.

배경색과 배경 범위 지정하기

Do it! 배경 이미지 배치하고 고정하기 — 예제 파일 09\bg-3.html

```
(... 생략 ...)
<style>
  body {
    background-image: url('images/bg2.png'); /* 문서 전체 배경 이미지 */
    background-repeat: no-repeat;            /* 배경 이미지 반복 안 함 */
    background-position: right top;          /* 배경 이미지를 오른쪽 상단에 위치 */
    background-a          : fixed;           /* 배경 이미지 고정 */
  }
  div {
    ......
    background-image: url('images/bg3.png'); /* 텍스트 상자의 배경 이미지 */
    background-repeat: no-repeat;            /* 배경 이미지를 반복 안 함 */
    background-position: right top;          /* 배경 이미지를 오른쪽 상단에 위치 */
  }
  #bg1 { background-origin: padding-box; }        /* 패딩까지 배경 이미지 표시 */
  #bg2 { background-origin: border-box; }         /* 테두리까지 배경 이미지 표시 */
  #bg3 { background-o          : content-box; }   /* 내용 영역에만 배경 이미지 표시 */
</style>
(... 생략 ...)
```

정답: attachment, origin

배경색과 배경 범위 지정하기

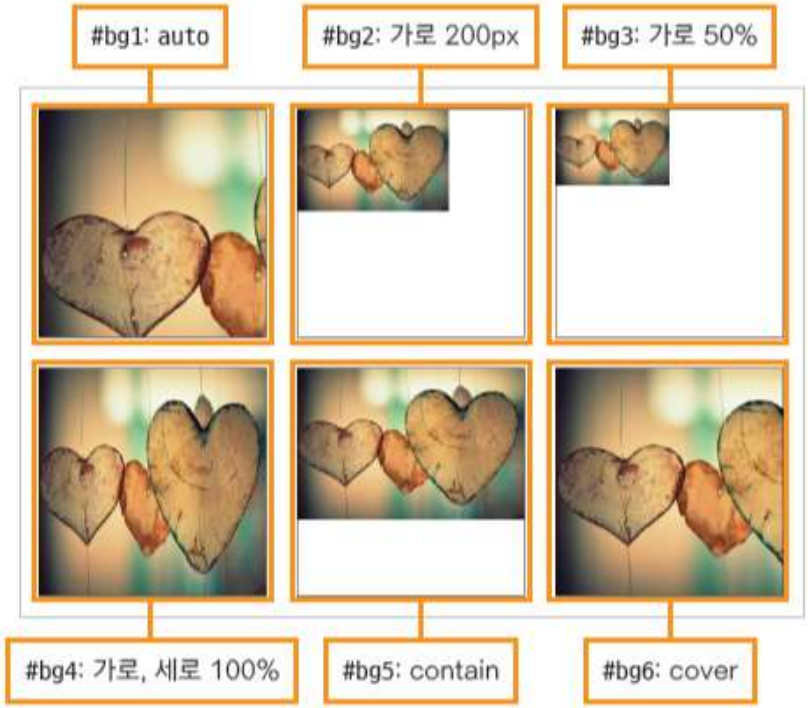
background-size 속성

배경 이미지 크기 조절

기본형 `background-size: auto | contain | cover | <크기 값> | <백분율>`

종류	설명
auto	원래 배경 이미지 크기만큼 표시합니다. 기본값입니다.
contain	요소 안에 배경 이미지가 다 들어오도록 이미지를 확대·축소합니다.
cover	배경 이미지로 요소를 모두 덮도록 이미지를 확대·축소합니다.
<크기>	이미지의 너비와 높이를 지정합니다. 값이 하나만 주어질 경우 너빗값으로 인식하며, 이미지의 너비와 너빗값에 맞춘 높잇값도 자동 계산합니다.
<백분율>	배경 이미지가 들어갈 요소의 크기를 기준으로 값을 백분율로 지정하고 그 크기에 맞도록 배경 이미지를 확대·축소합니다.

```
#bg1 { background-size: auto; }        /* 원래 배경 이미지 크기로 표시 */
#bg2 { background-size: 200px; }       /* 너비는 200px, 높이는 자동 계산 */
#bg3 { background-size: 50%; }         /* 너비는 요소 기준으로 50%, 높이는 자동 계산 */
#bg4 { background-size: 100% 100%; }   /* 요소의 너비와 높이를 100% 맞춤 */
#bg5 { background-size: contain; }     /* 요소의 배경 이미지가 다 보이게 표시 */
#bg6 { background-size: cover; }       /* 요소를 완전히 덮도록 배경 이미지 표시 */
</style>
(... 생략 ...)
```



그러데이션 효과로 배경 꾸미기

선형 그러데이션

- 수직 방향이나 수평 방향으로, 혹은 대각선 방향으로 색상이 일정하게 변하는 것
- 선형 그러데이션을 지정할 때는 방향과 색상이 필요하다

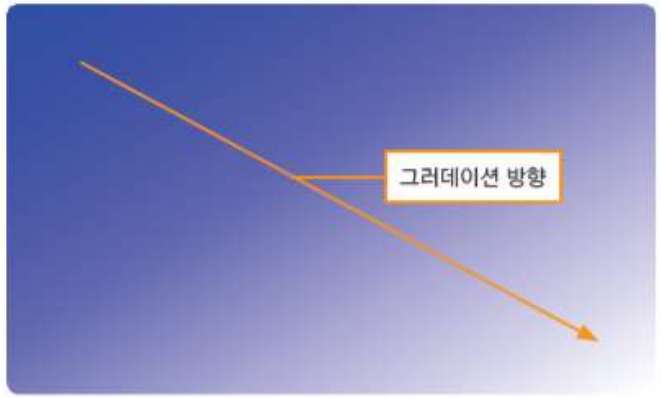
기본형 `linear-gradient(to <방향> 또는 <각도>, <색상 중지점>, [<색상 중지점>, .....]);`

1) 방향

끝 지점을 기준으로 'to' 키워드와 함께 사용

Do it! 방향을 사용해 선형 그러데이션 만들기 예제 파일 09\linear-1.html

```
(... 생략 ...)
  .grad {
    background: blue; /* CSS3를 지원하지 않는 웹 브라우저용 */
    background: linear-gradient(to right bottom, blue, white); /* 왼쪽 위에서 오른쪽 아래 방향으로, 파란색에서 흰색으로 */
  }
</style>
(... 생략 ...)
```

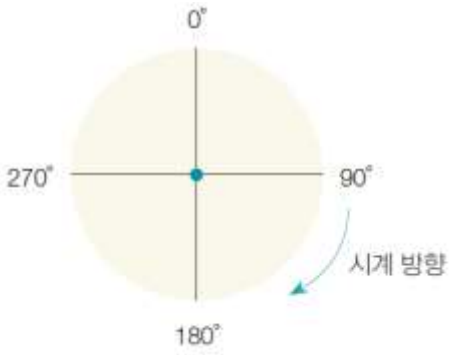


그러데이션 효과로 배경 꾸미기

선형 그러데이션

2) 각도

- 그러데이션이 끝나는 각도
- 단위는 deg



Do it! 각도를 사용해 선형 그러데이션 만들기 예제 파일 09\linear-2.html

```
(... 생략 ...)
  .grad {
    background: #f00; /* CSS3를 지원하지 않는 웹 브라우저용 */
    background: linear-gradient(45deg, #f00, #fff); /* 45도 방향, 빨간색에서 흰색으로 */
  }
</style>
(... 생략 ...)
```

그러데이션 효과로 배경 꾸미기

선형 그러데이션

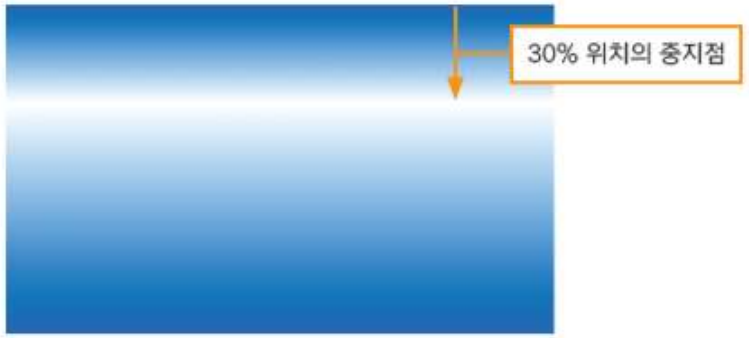
3) 색상 중지점

- 색상이 바뀌는 지점
- 색상만 지정할 수도 있고 색상과 함께 중지 점의 위치도 함께 지정할 수도 있음.

Do it! 선형 그러데이션의 색상 중지점 지정하기 — 예제 파일 09\linear-3.html

```
(... 생략 ...)
<style>
  .grad {
    background: #06f; /* CSS3를 지원하지 않는 웹 브라우저용 */
    background: linear-gradient(to bottom, #06f, white 30%, #06f); /* 위에서부터
30% 위치에 색상 중지점 지정 */
  }
</style>
(... 생략 ...)
```

시작 색상 — 끝 색상 — 중지점의 색상과 위치



그러데이션 효과로 배경 꾸미기

원형 그러데이션

- 원이나 타원의 중심부터 동심원을 그리며 바깥 방향으로 색상이 바뀌는 그러데이션
- 색상이 바뀌기 시작하는 원의 중심과 크기를 지정하고 그러데이션의 모양을 지정해야 함.

기본형 radial-gradient(<모양> <크기> at <위치>, <색상 중지점>, [<색상 중지점>, ...])

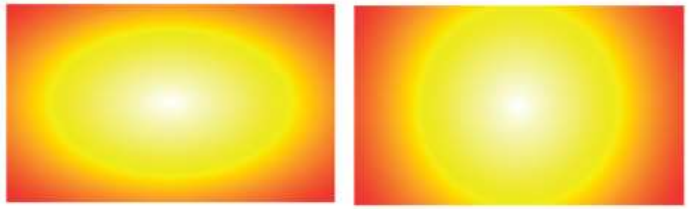
1) 모양

- 원형 그러데이션에서 만들어지는 모양은 circle(원형)과 ellipse(타원형)
- 따로 지정하지 않으면 ellipse로 인식

Do it! 원형 그러데이션의 모양 지정하기 예제 파일 09\radial-1.html

```
(... 생략 ...)
<style>
  .grad1 {
    background: red; /* CSS3를 지원하지 않는 웹 브라우저용 */
    background: radial-gradient(white, yellow, red); /* 모양을 지정하지 않으면 타원형 */
  }
  .grad2 {
    background: red; /* CSS3를 지원하지 않는 웹 브라우저용 */
    background: radial-g                (circle, white, yellow, red); /* 원형 */
  }
</style>
(... 생략 ...)
```

정답: gradient

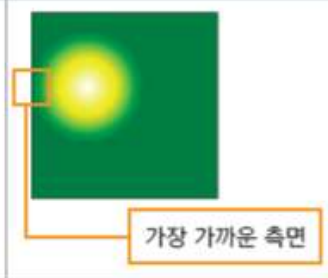
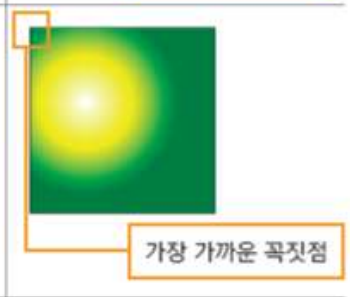
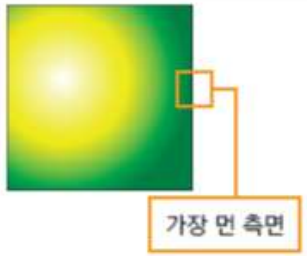


그러데이션 효과로 배경 꾸미기

원형 그러데이션

2) 크기

그러데이션 원의 크기 지정

속성값	설명	결과 화면
closest-side	• 원형이라면 그러데이션 가장자리가 그러데이션 중심에서 가장 가까운 요소의 측면에 닿을 때까지 그립니다. 타원형이라면 그러데이션 중심에서 가장 가까운 요소의 가로 측면이나 세로 측면에 닿을 때까지 그립니다. • 결과 화면은 그러데이션을 적용하는 요소가 변이 4개인 사각형입니다. 여기에서는 오른쪽 측면에 그러데이션의 가장자리가 닿을 때까지 그러데이션을 그립니다.	
closest-corner	• 그러데이션 가장자리가 중심에서 가장 가까운 요소의 코너에 닿도록 그립니다. • 결과 화면은 그러데이션이 적용되는 사격형 요소에 꼭짓점(coner)이 4개 있는데, 그러데이션의 중심에서 가장 가까운 꼭짓점에 닿을 때까지 그러데이션을 그립니다.	
farthest-side	• 원형이라면 그러데이션 가장자리가 그러데이션 중심에서 가장 멀리 떨어져 있는 측면에 닿을 때까지 그립니다. 타원형이라면 그러데이션 가장자리가 그러데이션 중심에서 가장 멀리 떨어져 있는 가로 또는 세로 측면과 만나도록 합니다.	
farthest-corner	• 그러데이션의 가장자리가 그러데이션의 중심에서 가장 멀리 떨어져 있는 코너에 닿도록 합니다. • 기본값입니다.	

그러데이션 효과로 배경 꾸미기

원형 그러데이션

3) 위치

- 그러데이션이 시작하는 원의 중심 지정
- '모양'과 '크기' 속성 다음에 at 키워드와 함께 위치 값 지정
- 사용할 수 있는 값 : 키워드나 백분율

left, center, right 중 하나, top, center, bottom 중 하나

Do it! 위치 키워드를 사용해 원형 그러데이션 만들기 예제 파일 09\radial-3.html

```
(... 생략 ...)
  .grad {
    background: blue; /* CSS3를 지원하지 않는 웹 브라우저용 */
    background: radial-gradient(circle at 20% 20%, white, blue); /* 원형 그러데이션,
20% 20% 위치, 흰색에서 파란색 지정 */
  }
</style>
(... 생략 ...)
```

그러데이션 효과로 배경 꾸미기

원형 그러데이션

4) 색상 중지점

- 색상이 바뀌는 지점
- 색상만 지정할 수도 있고
  색상과 함께 중지 점의 위치도 함께 지정할 수도 있음.

Do it! 원형 그러데이션의 색상 중지점 지정하기 — 예제 파일 09\radial-4.html

```
(... 생략 ...)
  .grad1 {
    background: skyblue; /* CSS3를 지원하지 않는 웹 브라우저용 */
    background: radial-gradient(yellow, white, orange); /* 색상 중지점 지정 */
  }
  .grad2 {
    background: skyblue;
    background: radial-g                (yellow, white 10%, orange 60%);  /* 색
상 중지점의 위치까지 지정 */
  }
</style>
......
  <div class="grad1"></div>
  <div class="grad2"></div>
(... 생략 ...)
```

정답: gradient

그러데이션 효과로 배경 꾸미기

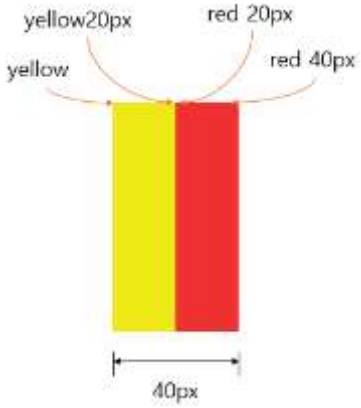
그러데이션을 사용한 패턴 만들기

단순히 그러데이션을 반복하는 것이 아니라 '패턴'을 만들어 반복시킴

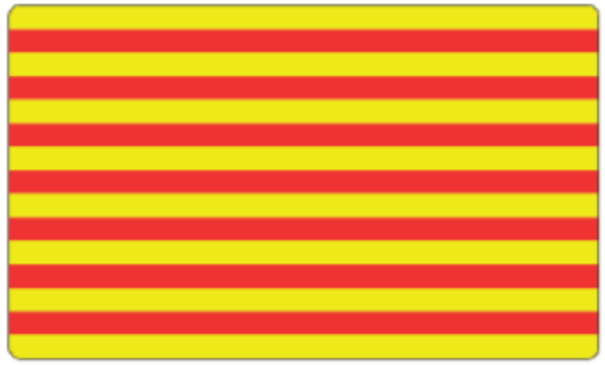
그러데이션을 단순 반복했을 경우

```
.grad1 {
  background: red; /* CSS3를 지원하지 않는 웹 브라우저용 */
  background: repeating-linear-gradient(yellow, red 20px); /* 선형 그러데이션 반복 */
}
```

그러데이션이 겹치는 부분을 조절했을 경우




```
.grad1 {
  background: red; /* CSS3를 지원하지 않는 웹 브라우저용 */
  background: repeating-linear-gradient(yellow, yellow 20px, red 20px, red 40px
);
}
```



10. CSS 고급 선택자

10-1 연결 선택자

10-2 속성 선택자

10-3 가상 클래스와 가상 요소

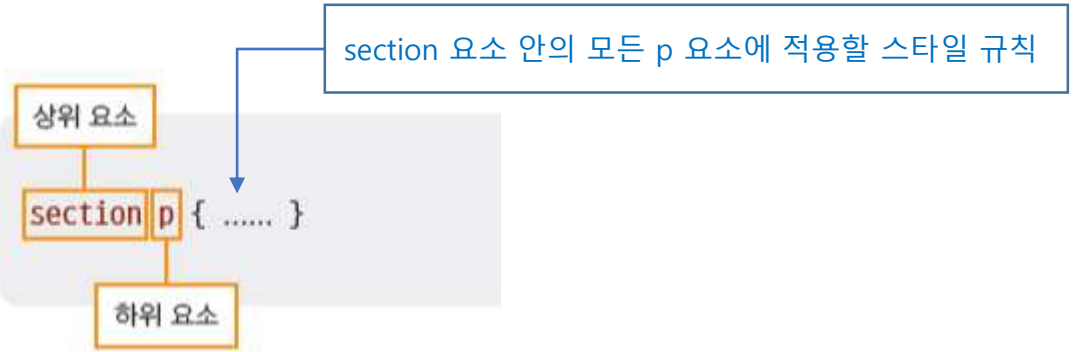
연결 선택자

- 연결 선택자 : 선택자와 선택자를 연결해 적용 대상을 제한하는 선택자.
- **컴비네이션 선택자**(combination selector) 또는 '**조합 선택자**'라고도 함

하위 선택자(descendant selector)

- 부모 요소에 포함된 모든 하위 요소에 스타일이 적용된다
- 자식 요소뿐만 아니라 손자 요소, 손자의 손자 요소 등 모든 하위 요소까지 적용
- 하위 선택자를 정의할 때는 상위 요소와 하위 요소를 나란히 쓴다.

기본형 상위요소 하위요소



Do it! 하위 선택자를 사용하여 글자색 적용하기 예제 파일 10\descendant.html

```
(... 생략 ...)
  section p { color: blue; } /* section 요소의 모든 하위 p 요소를 파란색 글자로 지정 */
</style>
.....
<section>
  <h1>예약 방법 & 사용 요금</h1>
  <p>아직 온라인 예약 신청이.....</p>
  <div>
    <p>가족실(2~4인) : 60,000원/일</p>
    <p>도미토리(4인 공용) : 25,000원/일</p>
  </div>
</section>
(... 생략 ...)
```

section의 모든 하위(자식, 손자) p 요소에 적용

예약 방법 & 사용 요금

아직 온라인 예약 신청이 준비되지 않았습니다.
전화(xxx-xxxx-xxxx)로 문의하시기 바랍니다.

가족실(2~4인) : 60,000원/일

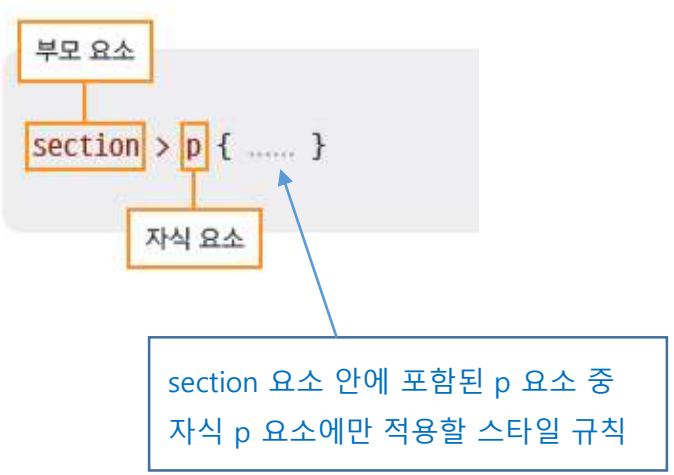
도미토리(4인 공용) : 25,000원/일

연결 선택자

자식 선택자(child selector)

- 자식 요소에 스타일을 적용하는 선택자
- 두 요소 사이에 '>(부등호)'를 표시해 부모 요소와 자식 요소를 구분

기본형 부모요소 > 자식요소



Do it! 자식 선택자를 사용하여 글자색 적용하기 — 예제 파일 10\child.html

```
(... 생략 ...)
  section > p { color: blue; }  /* section 요소의 자식인 p 요소를 파란색 글자로 지정 */
</style>
......
<section>
  <h1>예약 방법 & 사용 요금</h1>
  <p>아직 온라인 예약 신청이.....</p>
  <div>
    <p>가족실(2~4인): 60,000원/일</p>
    <p>도미토리(4인 공용): 25,000원/일</p>
  </div>
</section>
(... 생략 ...)
```

section 요소의 자식인 p 요소에만 스타일 적용

연결 선택자

인접 형제 선택자(adjacent selector)

- 같은 부모를 가진 형제 요소 중 첫 번째 동생 요소에만 스타일 적용
- 요소1과 요소2 사이에 '+' 기호 사용
- 요소1과 요소2는 같은 레벨이면서 요소1 이후 첫번째 요소2에 적용

기본형 요소1 + 요소2

h1의 형제인 p 요소 중 첫 번째 p

```
h1 + p { color: blue; }
```

h1 요소 다음에 오는 p 요소들 중
첫번째 p 요소에만 파란색 글씨 적용

Do it! 인접 형제 요소에 스타일 적용하기 예제 파일 10\adj.html

```
(... 생략 ...)
  h1 + p {
    background-color: #222;  /* 배경색은 검은색으로 */
    color: #fff;             /* 글자색은 흰색으로 */
  }
</style>
......
  <h1>예약 방법 & 이용 요금</h1>
  <p>아직 온라인 예약.....</p>
  <p>가족실(2~4인) : 60,000원/일</p>
  <p>도미토리(4인 공용) : 25,000원/일</p>
(... 생략 ...)
```

h1 요소의 형제 요소 중에서 첫 번째 형제인 p 요소에 적용

예약 방법 & 이용 요금

아직 온라인 예약 신청이 준비되지 않았습니다.
전화(xxx-xxxx-xxxx)로 문의하시기 바랍니다.

가족실(2~4인) : 60,000원/일

도미토리(4인 공용) : 25,000원/일

연결 선택자

형제 선택자(sibling selector)

- 형제 요소들에 스타일 적용
- 인접 형제 선택자와 다른 점은 모든 형제 요소에 다 적용된다는 것
- 요소1과 요소2 사이에 기호 '~' 사용

기본형 요소1 ~ 요소2

h1의 형제인 모든 p 요소

```
h1 ~ p { color: blue; }
```

h1 요소 다음에 오는
모든 형제 p 요소에 밑줄 적용

Do it! 모든 형제 요소의 스타일 지정하기 예제 파일 10\sibling.html

```
(... 생략 ...)
  h1 ~ p {
    background-color: #222; /* 배경색은 검은색으로 */
    color: #fff;            /* 글자색은 흰색으로 */
  }
</style>
......
  <h1>예약 방법 & 이용 요금</h1>
  <p>아직 온라인 예약.....</p>
  <p>가족실(2~4인) : 60,000원/일</p>
  <p>도미토리(4인 공용) : 25,000원/일</p>
(... 생략 ...)
```

h1 요소의 형제인 p 요소에 모두 적용

예약 방법 & 이용 요금

아직 온라인 예약 신청이 준비되지 않았습니다.
전화(xxx-xxxx-xxxx)로 문의하시기 바랍니다.

가족실(2~4인) : 60,000원/일

도미토리(4인 공용) : 25,000원/일

속성 선택자

[속성] 선택자

지정한 속성을 가진 요소를 찾아 스타일 적용

Do it! 요소 중에서 링크가 있는 요소만 스타일 적용하기 — 예제 파일 10\attr-1.html

```
(... 생략 ...)
  a[ h          ] {          /* href 속성이 있는 a 요소를 찾는 선택자 */
    background: yellow;       /* 배경색을 노란색으로 */
    border: 1px solid #ccc;   /* 테두리 지정 */
    font-weight: normal;      /* 글꼴 굵기 지정 */
  }
</style>
......
<ul>
  <li><a>메인 메뉴 : </a></li>
  <li><a href="#">메뉴 1</a></li>
  <li><a href="#">메뉴 2</a></li>
  <li><a href="#">메뉴 3</a></li>
  <li><a href="#">메뉴 4</a></li>
```

a 요소 중에서 href 속성값이 있는 요소만 적용

메인 메뉴 : 메뉴 1 메뉴 2 메뉴 3 메뉴 4

[속성 = 값] 선택자

주어진 속성과 속성 값이 일치하는 요소를 찾아 스타일 적용

Do it! 새 탭으로 열리는 링크에만 아이콘 추가하기 — 예제 파일 10\attr-2.html

```
(... 생략 ...)
  a[ t              ="_blank"] {  /* target 속성값이 _blank인 a 요소를 선택하는 선택자 */
    padding-right: 30px;          /* 오른쪽에 패딩 지정 */
    background: url(images/newwindow.png) no-repeat center right;  /* 배경 이미지 지정 */
  }
</style>
......
<ul>
  <li><a href="https://html.spec.whatwg.org" target=" _b          ">HTML</a></li>
  <li><a href="https://www.w3.org/TR/selectors">CSS Selector Level 3</a></li>
  <li><a href="https://www.w3.org/TR/css3-mediaqueries">미디어쿼리</a></li>
</ul>
(... 생략 ...)
```

a 요소 중에서 href 속성값이 _blank인 요소에만 적용

정답: _blank

HTML

CSS Selector Level 3

미디어쿼리

속성 선택자

[속성~=값] 선택자

여러 속성 값 중에 해당 값이 포함되어 있는 요소를 찾아 스타일 적용

Do it! 특정 속성값이 포함된 요소에 스타일 적용하기 — 예제 파일 10\attr-3.html

class 속성에 button 값이 있는 요소를 찾는 선택자

```
(... 생략 ...)
  [class ~= "b                ] {
    box-shadow: rgba(0, 0, 0, 0.4) 4px 4px;  /* 그림자 지정 */
    border-radius: 5px;  /* 모서리 둥글게 지정 */
  }
</style>
.....
<ul>
  <li><a href="#" class="flat">메뉴 1</a></li>
  <li><a href="#" class="flat">메뉴 2</a></li>
  <li><a href="#" class="button">메뉴 3</a></li>
  <li><a href="#" class="flat button" >메뉴 4</a></li>
</ul>
(... 생략 ...)
```

속성값이 button인 요소에 적용

속성값으로 button을 포함하는 요소에 적용

정답: "button"

[속성 |= 값] 선택자

- 특정 값이 포함된 속성을 가진 요소를 찾아 스타일 적용
- 하이픈으로 연결해 한 단어 값을 이루는 요소에도 적용

Do it! title 속성값에 따라 아이콘 다르게 표시하기 — 예제 파일 10\attr-4.html

```
(... 생략 ...)
❶a[title |= us] {  /* 속성값이 "us"이거나 "us-"로 시작하는 a 요소를 찾는 선택자 */
    background: url(images/us.png) no-repeat left center;  /* 아이콘 적용 */
  }
❷a[title |= jap] {  /* 속성값이 "jap"이거나 "jap-"로 시작하는 a 요소를 찾는 선택자 */
    background: url(images/jp.png) no-repeat left center;  /* 아이콘 적용 */
  }
❸a[title |= chn] {  /* 속성값이 "chn"이거나 "chn-"으로 시작하는 a 요소를 찾는 선택자 */
    background: url(images/ch.png) no-repeat left center;  /* 아이콘 적용 */
  }
</style>
.....
<ul>
  <li>외국어 서비스 : </li>
  <li><a href="#" title="us-english">영어</a></li>
  <li><a href="#" title="ja">일본어</a></li>
  <li><a href="#" title="chn">중국어</a></li>
</ul>
(... 생략 ...)
```

속성값이 "us-"로 시작하는 한 단어이므로 ❶ 스타일이 적용됨

속성값이 "jap"도, "jap-"도 아니므로 ❷ 스타일이 적용되지 않음

속성값이 "chn"과 일치하므로 ❸ 스타일이 적용됨

속성 선택자

[속성^=값] 선택자

특정 값으로 시작하는 속성을 가진 요소를 찾아 스타일 적용

Do it! 속성값의 시작 부분이 일치하는 요소에 스타일 적용하기 — 예제 파일 10\attr-5.html

```
(... 생략 ...)
❶a[title ^="eng"] { /* 속성값이 "eng"로 시작하는 a 요소를 찾는 선택자 */
  background: url(images/us.png) no-repeat left center;  /* 아이콘 적용 */
  padding: 5px 25px;  /* 패딩 지정 */
 }
❷a[title ^="jap"] { /* 속성값이 "jap"로 시작하는 요소를 찾는 선택자 */
  background: url(images/jp.png) no-repeat left center;  /* 아이콘 적용 */
  padding: 5px 25px;  /* 패딩 지정 */
 }
❸a[title ^="chin"] { /* 속성값이 "chin"으로 시작하는 요소를 찾는 선택자 */
  background: url(images/ch.png) no-repeat left center;  /* 아이콘 적용 */
  padding: 5px 25px;  /* 패딩 지정 */
 }
</style>
......
<ul>
  <li>외국어 서비스 : </li>
  <li><a href="#" title="english">영어</a></li>
  <li><a href="#" title="japanese">일본어</a></li>
  <li><a href="#" t              ="chinese">중국어</a></li>
</ul>
(... 생략 ...)
```

속성값이 "eng"로 시작하므로 ❶ 스타일 적용

속성값이 "jap"로 시작하므로 ❷ 스타일 적용

속성값이 "chin"으로 시작하므로 ❸ 스타일 적용

정답: title

외국어 서비스 : 영어 일본어 중국어

[속성 $= 값] 선택자

특정 값으로 끝나는 속성을 가진 요소를 찾아 스타일 적용

Do it! 파일 확장자에 따라 아이콘 다르게 표시하기 — 예제 파일 10\attr-6.html

```
(... 생략 ...)
❶a[href $= hwp] {  /* 링크 속성값이 hwp로 끝나는 a 요소를 찾는 선택자 */
  background: url(images/hwp_icon.gif) center right no-repeat;
  padding-right: 25px;  /* 오른쪽 패딩 지정 */
 }
❷a[href $= xls] {  /* 링크 속성값이 xls로 끝나는 a 요소를 찾는 선택자 */
  background: url(images/excel_icon.gif) center right no-repeat;
  padding-right: 25px; /* 오른쪽 패딩 지정 */
 }
</style>
......
<h3>회사 소개 파일 내려받기</h3>
<ul>
  <li><a href="intro.hwp">hwp 파일</a></li>
  <li><a h              ="intro.xls">엑셀 파일</a></li>
</ul>
(... 생략 ...)
```

속성값이 "hwp"로 끝나므로 ❶ 스타일이 적용됨

속성값이 "xls"로 끝나므로 ❷ 스타일이 적용됨

정답: href

회사 소개 파일 내려받기

- hwp 파일
- 엑셀 파일

속성 선택자

[속성*=값] 선택자

값의 일부가 일치하는 속성을 가진 요소를 찾아 스타일 적용

Do it! 속성값의 일부가 일치하는 요소에 스타일 적용하기 — 예제 파일 10\attr-7.html

```
(... 생략 ...)
  a[href *= w3] {   /* 속성값 중 일부에 w3가 있는 a 요소를 찾는 선택자 */
    background: blue;
    color: white;
  }
</style>
.....
<p>(아래 링크 중에서 배경색이 파란색인 링크는 W3C 사이트로 연결됩니다.)</p>
<ul>
  <li><a href="https://html.spec.whatwg.org/">HTML 표준안 사이트</a></li>
  <li><a href="https://caniuse.com/">HTML 지원 여부 체크</a></li>
  <li><a href="https://www.w3.org/TR/css3-mediaqueries">미디어쿼리</a></li>
</ul>
(... 생략 ...)
```

href의 속성값 중에서 "w3"가 일치하므로 스타일을 적용

HTML5 참고 사이트

(아래 링크 중에서 배경색이 파란색인 링크는 W3C 사이트로 연결됩니다.)

- HTML 표준안 사이트
- HTML 지원 여부 체크
- 미디어쿼리

속성 선택자 정리

종류	선택 요소	예시
[속성]	해당 속성이 있는 요소	[required]
[속성 = 값]	지정한 속성값이 있는 요소	[target = _blank]
[속성 ~= 값]	지정한 속성값이 포함된 요소(단어별)	[class ~= button]
[속성 |= 값]	지정한 속성값이 포함된 요소(하이픈 포함, 단어별)	[title |= us]
[속성 ^= 값]	지정한 속성값으로 시작하는 요소	[title ^= eng]
[속성 $= 값]	지정한 속성값으로 끝나는 요소	[href $= xls]
[속성 *= 값]	지정한 속성값의 일부가 일치하는 요소	[href *= w3]

가상 클래스와 가상 요소

사용자 동작에 반응하는 가상 클래스 선택자

:link - 방문하지 않은 링크에 스타일 적용
:visited - 방문한 링크에 스타일 적용
:active - 웹 요소를 활성화했을 때의 스타일 적용
:hover - 웹 요소에 마우스 커서를 올려놓을 때의 스타일 적용
:focus - 웹 요소에 초점이 맞추어졌을 때의 스타일 적용

순서 중요 :link → :visited → :hover → :active

Do it! 가상 클래스 선택자를 사용해 링크 스타일 적용하기 — 예제 파일 10\navi.html

```
(... 생략 ...)
  .navi ul li {
    float: left;
    width: 150px;
    padding: 10px;
  }
  .navi a:link, .navi a:visited {  /* 방문한 링크와 방문하지 않은 링크 설정 */
    display: block;
    font-size: 14px;
    color: #000;
    padding: 10px;
    text-decoration: none; /* 밑줄을 없앰 */
    text-align: center;
  }
  .navi a:hover, .navi a:focus {  /* 마우스 포인터를 올렸을 때와 초점을 맞췄을 때 설정 */
    background-color: #222;       /* 배경색을 짙은 회색으로 변경 */
    color: #fff;
  }
  .navi a:active {                /* 마우스 포인터로 클릭했을 때 지정 */
    background-color: #f00;       /*배경색을 빨간색으로 변경 */
  }
</style>
......
  <nav class="navi">
    <ul>
      <li><a href="#">이용 안내</a></li>
      <li><a href="#">객실 소개</a></li>
      <li><a href="#">예약 방법 및 요금</a></li>
      <li><a href="#">예약하기</a></li>
    </ul>
  </nav>
```

가상 클래스와 가상 요소

요소 상태에 따른 가상 클래스 선택자

:target – 앵커로 연결된 부분에 스타일 적용

Do it! 앵커 대상에 스타일 적용하기 — 예제 파일 10\ps-1.html

```
(... 생략 ...)
  #intro:target {
    background-color: #0069e0;  /* 배경색을 파란색으로 지정 */
    color: #fff;                /* 글자색을 흰색으로 지정 */
  }
</style>
......
  <nav class="navi">
    <ul>
      <li><a href="#intro">이용 안내</a></li>
      <li><a href="#room">객실 소개</a></li>
      <li><a href="#reservation">예약 방법 및 요금</a></li>
      <li><a href="ps-2.html">예약하기</a></li>
    </ul>
  </nav>
  <div id="intro" class="contents">
    <h2>이용 안내</h2>
    ......
  </div>
  <div id="room" class="contents">
    <h2>객실 소개</h2>
    ......
  </div>
  <div id="reservation" class="contents">
    <h2>예약 방법 및 요금</h2>
    ......
  </div>
(... 생략 ...)
```

가상 클래스와 가상 요소

요소 상태에 따른 가상 클래스 선택자

:target – 앵커로 연결된 부분에 스타일 적용

:enabled, :disabled – 요소의 사용 여부에 따라 스타일 적용

:checked – 라디오 버튼이나 체크 박스에 체크했을 때 스타일 적용

Do it! 선택된 라디오 버튼의 스타일 적용하기 예제 파일 10\ps-2.html

```
(... 생략 ...)
  #signup input:checked + label {  /* input 요소에 checked 속성이 추가되었을 때 label
요소의 스타일 */
    color: red;        /* 글자색을 빨간색으로 지정 */
    font-weight: bold; /* 글꼴을 굵게 지정 */
  }
</style>
......
  <ul>
    <li>
      <input type="radio" name="room" id="basic">
      <label for="basic">기본형(최대 2인)</label>
    <li>
    <li>
     <input type="radio" name="room" id="family">
      <label for="family">가족형(최대 8인)</label>
    </li>
  </ul>
(... 생략 ...)
```

가상 클래스와 가상 요소

요소 상태에 따른 가상 클래스 선택자

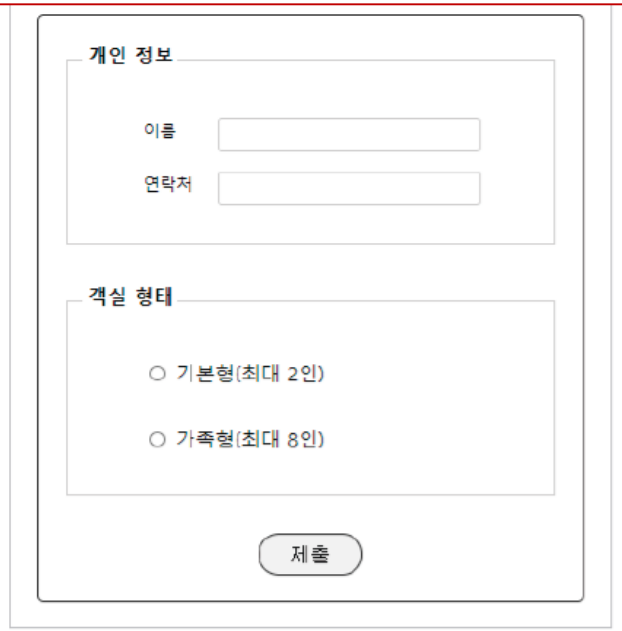
:target – 앵커로 연결된 부분에 스타일 적용

:enabled, :disabled – 요소의 사용 여부에 따라 스타일 적용

:checked – 라디오 버튼이나 체크 박스에 체크했을 때 스타일 적용

:not – 특정 요소를 제외하고 스타일 적용

type="text" 필드와 type="tel" 필드에 스타일 적용하기



일반적인 방법

Do it! 일반적으로 텍스트 필드 선택하기 예제 파일 10\ps-2.html

```
(... 생략 ...)
  #signup input[type=text], input[type=tel] {
    border: 1px solid #ccc;
    border-radius: 3px;
    padding: 5px;
    width: 200px;
  }
(... 생략 ...)
```

:not 선택자 사용

Do it! not 선택자로 라디오 필드를 제외한 텍스트 필드 선택하기 예제 파일 10\ps-2.html

```
(... 생략 ...)
  #signup input:not([type=radio]) {
    border: 1px solid #ccc;
    border-radius: 3px;
    padding: 5px;
    width: 200px;
  }
(... 생략 ...)
```

정답: radio

가상 클래스와 가상 요소

문서 구조에 따른 가상 클래스 선택자

- 웹 문서의 구조를 기준으로 특정 위치에 있는 요소를 찾아 스타일 적용
- 위치가 계속 바뀐다면 a*n*+b 처럼 수식을 사용할 수도 있음.
  이 때 n 값은 0부터.

종류	설명
:only-child	부모 안에 자식 요소가 하나뿐일 때 자식 요소를 선택합니다.
A:only-type-of	부모 안에 A 요소가 하나뿐일 때 선택합니다.
:first-child	부모 안에 있는 모든 요소 중에서 첫 번째 자식 요소를 선택합니다
:last-child	부모 안에 있는 모든 요소 중에서 마지막 자식 요소를 선택합니다
A:first-of-type	부모 안에 있는 A 요소 중에서 첫 번째 요소를 선택합니다.
A:last-of-type	부모 안에 있는 A 요소 중에서 마지막 요소를 선택합니다.
:nth-child(*n*)	부모 안에 있는 모든 요소 중에서 n번째 자식 요소를 선택합니다.
:nth-last-child(*n*)	부모 안에 있는 모든 요소 중에서 끝에서 n번째 자식 요소를 선택합니다.
A:nth-of-type(*n*)	부모 안에 있는 A 요소 중에서 n번째 요소를 선택합니다.
A:nth-last-of-type(*n*)	부모 안에 있는 A 요소 중에서 끝에서 n번째 요소를 선택합니다.

Do it! 위치에 따라 자식 요소 선택하기 — 예제 파일 10\ps-3.html

```
(... 생략 ...)
❶.contents :nth-child(3) {    /* .contents의 세 번째 자식 요소에 스타일 적용 */
  background-color: #ffff00; /* 배경색을 노란색으로 */
}
❷.contents p:nth-of-type(3) { /* .contents의 p 요소 중에서 세 번째 자식 요소에 스타일 적용 */
  background-color: #00b900; /* 배경색을 초록색으로 */
}
</style>
.....
<div class="contents">
  <h2>이용 안내</h2>
  <p>Excepteur do ..... </p>
  <p>Qui magna culpa ..... </p>
  <h2>객실 소개</h2>
  <p>Irure incididunt ..... </p>
  <h2>예약 방법 및 요금</h2>
  <p>Fugiat aliquip ..... </p>
</div>
```

가상 클래스와 가상 요소

가상 요소

- 화면 꾸미기용 요소를 웹 문서에 포함시키지 않기 위해 가상 요소 사용

- ::first-line : 특정 요소의 첫번째 줄에 스타일 적용
- ::first-letter : 특정 요소의 첫번째 글자에 스타일 적용
- ::before : 특정 요소의 앞에 지정한 콘텐츠 추가
- ::after : 특정 요소의 뒤에 지정한 콘텐츠 추가

Do it! 가상 요소를 사용해 스타일 적용하기

```
(... 생략 ...)
  li.new::after {
    content: "NEW!!";
    font-size: x-small;
    padding: 2px 4px;
    margin: 0 10px;
    border-radius: 2px;
    background: #f00;
    color: #fff;
  }
</style>
......
<div class="container">
  <h1>제품 목록</h1>
  <ul>
    <li class="new">제품 A</li>
    <li>제품 B</li>
    <li>제품 C</li>
    <li class="new">제품 D</li>
  </ul>
</div>
(... 생략 ...)
```

11. 트랜지션과 애니메이션

11-1 변형 알아보기

11-2 트랜지션 알아보기

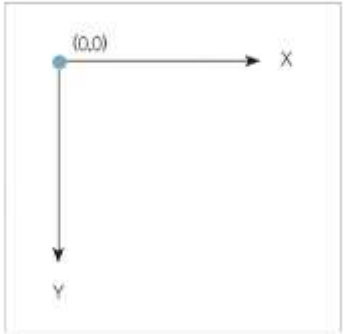
11-3 애니메이션 알아보기

변형 알아보기

변형(transform, 트랜스폼) : 특정 요소의 크기나 형태 등 스타일이 바뀌는 것

2차원 변형

- 수평이나 수직으로 웹 요소 변형
- 크기나 각도만 지정하면 됨
- 2차원 좌표 사용



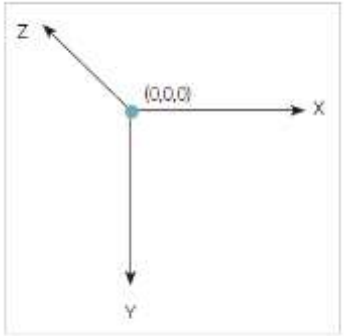
2차원 좌표계

표 11-1 2차원 변형 함수

종류	설명
translate(tx, ty)	지정한 크기만큼 x축, y축으로 이동합니다.
translateX(tx)	지정한 크기만큼 x축으로 이동합니다.
translateY(ty)	지정한 크기만큼 y축으로 이동합니다.
scale(sx, sy)	지정한 크기만큼 x축과 y축으로 확대·축소합니다.
scaleX(sx)	지정한 크기만큼 x축으로 확대·축소합니다.
scaleY(sy)	지정한 크기만큼 y축으로 확대·축소합니다.
rotate(각도)	지정한 각도만큼 회전합니다.
skew(ax, ay)	지정한 각도만큼 x축과 y축으로 왜곡합니다.
skewX(ax)	지정한 각도만큼 x축으로 왜곡합니다.
skewY(ay)	지정한 각도만큼 y축으로 왜곡합니다.

3차원 변형

- x축과 y축에 원근감 추가
- z축은 앞뒤로 이동. 보는 사람 쪽으로 다가올 수록 값이 더 커짐



3차원 좌표계

표 11-2 3차원 변형 함수

종류	설명
translate3d(tx, ty, tz)	지정한 크기만큼 x축, y축, z축으로 이동합니다.
translateZ(tz)	지정한 크기만큼 z축으로 이동합니다.
scale3d(sx, sy, sz)	지정한 크기만큼 x축, y축, z축으로 확대·축소합니다.
scaleZ(sz)	지정한 크기만큼 z축으로 확대·축소합니다.
rotate(rx, ry, 각도)	지정한 각도만큼 회전합니다.
rotate3d(rx, ry, rz, 각도)	지정한 각도만큼 회전합니다.
rotateX(각도)	지정한 각도만큼 x축으로 회전합니다.
rotateY(각도)	지정한 각도만큼 y축으로 회전합니다.
rotateZ(각도)	지정한 각도만큼 z축으로 회전합니다.
perspective(길이)	입체적으로 보일 수 있도록 깊잇값을 지정합니다.

변형 알아보기

translate 함수

지정한 방향으로 이동할 거리를 지정하면 해당 요소를 이동시킴

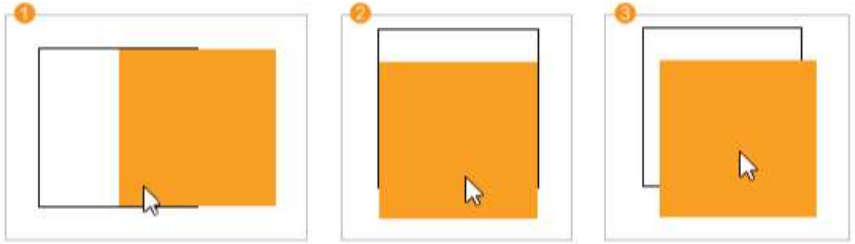
기본형
```
transform: translate(tx, ty)
transform: translate3d(tx, ty, tz)
transform: translateX(tx)
transform: translateY(ty)
transform: translateZ(tz)
```

- **transform:translate(tx, ty)** - x축 방향으로 tx만큼, y축 방향으로 ty만큼 이동 tx와 ty 두 가지 값을 사용하지만 ty 값이 주어지지 않으면 0으로 간주.
- **transform:translate3d(tx, ty, tz)** - x축 방향으로 tx만큼, y축 방향으로 ty만큼, z축 방향(앞뒤)으로 tz만큼 이동.
- **transform:translateX(tx)** - x축 방향으로 tx만큼 이동.
- **transform:translateY(ty)** - y축 방향으로 ty만큼 이동.
- **transform:translateZ(tz)** - z축 방향으로 tz만큼 이동.

Do it! translate( ) 함수를 사용해 웹 요소 이동하기 — 예제 파일 11\translate.html

```
(... 생략 ...)
  #movex:hover { transform: translateX(50px); }  /* x축으로 50px 이동 */ ❶
  #movey:hover { transform: translateY(20px); }  /* y축으로 20px 이동 */ ❷
  #movexy:hover { transform: t                (10px, 20px); }  /* x축으로 10px, y축
으로 20px 이동 */ ❸
</style>
(... 생략 ...)
```

정답: translate



변형 알아보기

scale 함수

지정한 크기만큼 요소를 확대/축소

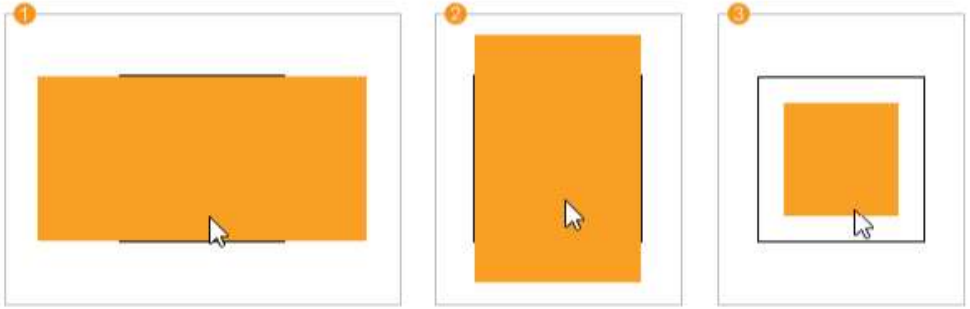
기본형
```
transform: scale(sx, sy)
transform: scale3d(sx, sy, sz)
transform: scaleX(sx)
transform: scaleY(sy)
transform: scaleZ(sz)
```

- **transform:scale(sx, sy)** - x축 방향으로 sx만큼, y축 방향으로 sy만큼 확대.
  sy 값이 주어지지 않는다면 sx 값과 같다고 간주.
  예) scale(2.0)는 scale(2,2)와 같은 함수이며 요소를 두 배로 확대.
- **transform:scale3d(sx, sy, sz)** - x축 방향으로 sx만큼, y축 방향으로 sy만큼, z축 방향으로 sz만큼 확대.
- **transform:scaleX(sx)** – x축 방향으로 sx만큼 확대.
- **transform:scaleY(sy)** - y축 방향으로 sy만큼 확대.
- **transform:scaleZ(sz)** -z축 방향으로 sz만큼 확대.

Do it! scale( ) 함수를 사용해 확대·축소하기 — 예제 파일 11\scale.html

```
(... 생략 ...)
<style>
  #scalex { transform: scaleX(2); }          /* x축으로 2배 확대 */ ❶
  #scaley { transform: scaleY(1.5); }        /* y축으로 1.5배 확대 */ ❷
  #scale { transform: s          (0.7); }  /* x, y축으로 0.7배 확대 */ ❸
</style>
(... 생략 ...)
```

정답: scale



변형 알아보기

rotate 함수

- 각도만큼 웹 요소를 시계 방향이나 시계 반대 방향으로 회전
- 일반 각도(degree)나 래디안(radian) 값 사용(1래디안=1/180°)

2차원 rotate( ) 함수

기본형 `transform: rotate(각도)`

Do it! 함수를 사용해 2차원에서 회전하기 예제 파일 11\rotate.html

```
(... 생략 ...)
<style>
  #rotate1:hover { transform: rotate(40deg); }        /* 오른쪽으로 40도 회전 */ ❶
  #rotate2:hover { transform: r          (-40deg); } /* 왼쪽으로 40도 회전 */ ❷
</style>
(... 생략 ...)
```

정답: rotate

변형 알아보기

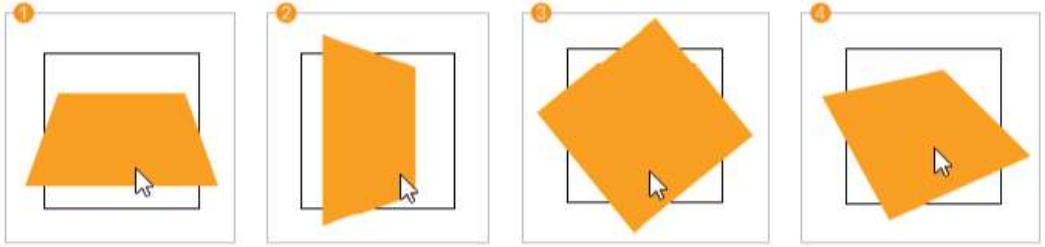
rotate 함수

3차원 rotate( ) 함수

```
기본형  transform: rotate(rx, ry, 각도)
        transform: rotate3d(rx, ry, rz, 각도)
        transform: rotateX(각도)
        transform: rotateY(각도)
        transform: rotateZ(각도)
```

perspective 속성

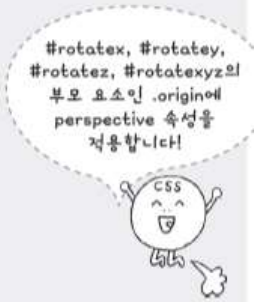
- 원근감을 표현하기 위해 사용하는 속성
- 원래 있던 위치에서 사용자가 있는 쪽으로 얼마나 이동하는지 나타냄.
- 값(픽셀 단위)은 0보다 커야 하며 값이 클수록 사용자로부터 멀어짐.
- perspective 속성은 변형하는 요소의 부모 요소에 정의해야 한다.



Do it! rotate( ) 함수를 사용해 3차원에서 회전하기 — 예제 파일 11\rotate3d.html

```
(... 생략 ...)
  .origin {
    ......
    perspective: 200px;  /* 원근감 추가 */
  }
  .origin > div {
    ......
    transition: all 3s;  /* 3초 동안 회전하도록 트랜지션 적용 */
  }
  #rotatex:hover { transform: rotateX(55deg); }        /* x축으로 55도 회전 */ ❶
  #rotatey:hover { transform: rotateY(55deg); }        /* y축으로 55도 회전 */ ❷
  #rotatez:hover { transform: rotateZ(55deg); } /* z축으로 55도 회전 */ ❸
  #rotatexyz:hover {transform: rotate3d(2.5, 1.2, -1.5, 55deg); } /* x, y, z축에 방향 벡터를 지정하고 55도 회전 */ ❹
</style>
......
<div class="origin">
  <div id="rotatex"></div>
</div>
<div class="origin">
  <div id="rotatey"></div>
</div>
<div class="origin">
  <div id="rotatez"></div>
</div>
<div class="origin">
  <div id="rotatexyz"></div>
</div>
(... 생략 ...)
```

정답: rotateZ, rotateY



변형 알아보기

skew 함수

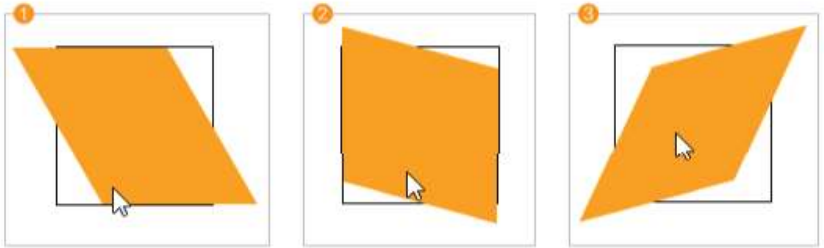
요소를 지정한 각도만큼 비틀어 왜곡

```
기본형 transform: skew(x각도, y각도)
       transform: skewX(x각도)
       transform: skewY(y각도)
```

- **transform:skewX(ax)** – x축을 따라 당김.
- **transform:skewY(ay)** – y축을 따라 당김.
- **transform:skew(ax, ay)** – 첫 번째 각도는 x축을 따라 당기는 각도이고 두 번째 각도는 y축을 따라 당기는 각도. 두 번째 값이 주어지지 않으면 y축에 대한 각도를 0으로 간주함.

Do it! skew( ) 함수를 사용해 도형 비틀기 — 예제 파일 11\skew.html

```
(... 생략 ...)
  #skewx:hover { transform: skewX(30deg); } /* x축 기준으로 30도 비틀기 */ ❶
  #skewy:hover { transform: skewY(15deg); } /* y축 기준으로 15도 비틀기 */ ❷
  #skewxy:hover { transform: skew(-25deg, -15deg); } /* x축 기준으로 -25도, y축 기준
으로 -15도 비틀기 */ ❸
</style>
(... 생략 ...)
```



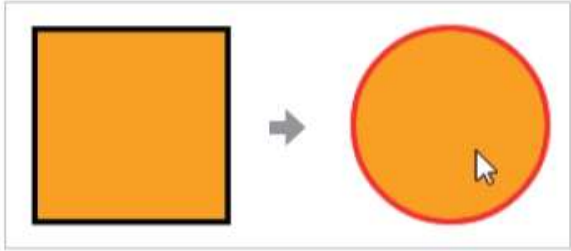
트랜지션 알아보기

트랜지션이란

웹 요소의 스타일 속성이 조금씩 자연스럽게 바뀌는 것

예) 하늘색 도형 위로 마우스를 올려놓으면 도형이 하늘색에서 파란색으로 바뀌고 마우스를 치우면 원래 배경 색으로 되돌아감.

예) 도형 위로 마우스를 올려놓으면 사각형의 테두리와 테두리색이 바뀌고 마우스를 치우면 원래 스타일로 되돌아감.



트랜지션의 속성

종류	설명
transition-property	트랜지션의 대상을 지정합니다.
transition-duration	트랜지션을 실행할 시간을 지정합니다.
transition-timing-function	트랜지션의 실행 형태를 지정합니다.
transition-delay	트랜지션의 지연 시간을 지정합니다.
transition	transition-property, transition-duration, transition-timing-function, transition-delay 속성을 한꺼번에 정합니다.

트랜지션 알아보기

transition-property 속성

- 트랜지션을 적용할 속성 선택
- 이 속성을 지정하지 않으면 모든 속성이 트랜지션 대상이 됨.

기본형 `transition-property: all | none | <속성 이름>`

종류	설명
all	all값을 사용하거나 transition-property를 생략할 경우 요소의 모든 속성이 트랜지션 대상이 됩니다. 기본값입니다.
none	트랜지션을 하는 동안 아무 속성도 바뀌지 않습니다.
속성 이름	트랜지션 효과를 적용할 속성을 지정합니다. 예를 들어 배경색만 바꿀 것인지, width값을 바꿀 것인지 원하는 대상만 골라 지정할 수 있습니다. 속성이 여럿일 경우 쉼표(,)로 구분하여 나열합니다.

```
transition-property: all;              /* 해당 요소의 모든 속성에 트랜지션 적용 */
transition-property: background-color; /* 해당 요소의 배경색에 트랜지션 적용 */
transition-property: width, height;    /* 해당 요소의 너비와 높이에 트랜지션 적용 */
```

transition-duration 속성

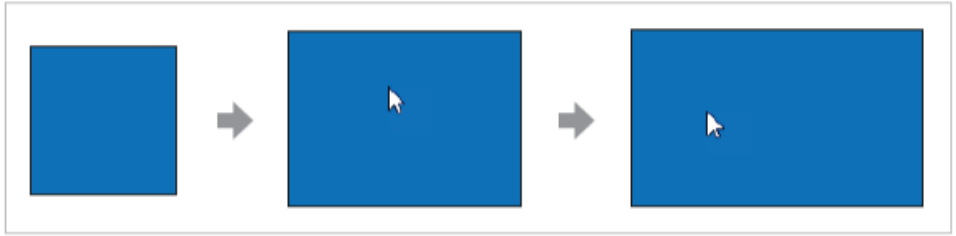
- 트랜지션 진행 시간 지정
- 시간 단위는 초(seconds) 또는 밀리초(milliseconds)
- 트랜지션이 여러 개라면 쉼표(,)로 구분해 진행 시간 지정

기본형 `transition-duration: <시간>`

Do it! 트랜지션 대상과 진행 시간 지정하기 예제 파일 11\tr-1.html

```
(... 생략 ...)
  .box {
    ......
    transition-property: width, height;   /* 트랜지션 대상 - 너비, 높이 */
    transition-duration: 2s, 1s;          /* 트랜지션 시간 - 2초, 1초 */
  }
  .box:hover {
    width: 200px;
    height: 120px;
  }
</style>
(... 생략 ...)
```

정답: duration



트랜지션 알아보기

transition-timing-function 속성

트랜지션의 시작과 중간, 끝에서의 속도 지정

기본형 `transition-timing-function: linear | ease | ease-in | ease-out | ease-in-out | cubic-bezier(n, n, n, n)`

종류	설명
ease	처음에는 천천히 시작하고 점점 빨라지다가 마지막엔 천천히 끝냅니다. 기본값입니다.
linear	시작부터 끝까지 똑같은 속도로 진행합니다.
ease-in	느리게 시작합니다.
ease-out	느리게 끝냅니다.
ease-in-out	느리게 시작하고 느리게 끝냅니다.
cubic-bezier(n, n, n, n)	베지에 함수를 정의해서 사용합니다. 이때 n값은 0~1 사이만 사용할 수 있습니다.

transition-delay 속성

- 트랜지션이 언제부터 시작될지 지연 시간 지정
- 시간 단위는 초(seconds) 또는 밀리초(milliseconds). 기본값 0

기본형 `transition-delay: <시간>`

transition 속성

- 트랜지션 관련 속성을 한꺼번에 지정
- 시간 값 속성이 2개이므로, 앞에 오는 시간값은 transition-duration, 뒤에 오는 시간 값은 transition-delay 속성으로 간주

기본형 `transition: <transition-property값> | <transtion-duration값> | <transition-timing-function값> | <transition-delay값>`

Do it! 트랜지션 속성 한꺼번에 지정하기 예제 파일 11\tr-2.html

```
(... 생략 ...)
  .box {
    ......
    transition: 2s ease-in;
  }
  .box:hover {
    width: 200px;
    height: 200px;
    background-color: #f50;
    transform: rotate(270deg);
  }
</style>
(... 생략 ...)
```

transition-property: 기본값 all
transition-duration: 2s
transition-timing-function: ease-in
transition-delay: 기본값 0

여기에 있는 속성이 모두 트랜지션 대상

애니메이션 알아보기

CSS와 애니메이션

- 웹 요소에 애니메이션 추가
- 애니메이션을 시작해 끝내는 동안 원하는 곳 어디서든 스타일을 바꾸며 애니메이션을 정의할 수 있다.
- 키프레임(keyframe) : 애니메이션 중간에 스타일이 바뀌는 지점

애니메이션 관련 속성

종류	설명
@keyframes	애니메이션이 바뀌는 지점을 지정합니다.
animation-delay	애니메이션의 시작 시간을 지정합니다.
animation-direction	애니메이션을 종료한 뒤 처음부터 시작할지, 역방향으로 진행할지 지정합니다.
animation-duration	애니메이션의 실행 시간을 지정합니다.
animation-iteration-count	애니메이션의 반복 횟수를 지정합니다.
animation-name	@keyframes로 설정해 놓은 중간 상태를 지정합니다.
animation-timing-function	키프레임의 전환 형태를 지정합니다.
animation	animation 속성을 한꺼번에 묶어서 지정합니다.

애니메이션 알아보기

@keyframes 속성

- 애니메이션의 시작과 끝을 비롯해 상태가 바뀌는 지점을 설정
- '이름'으로 애니메이션 구별

```
기본형 @keyframes <이름> {
          <선택자> { <스타일> }
        }
```

- @keyframes의 선택자에서 속성값이 바뀌는 지점을 가리킴
- 시작 위치는 0%, 끝 위치 100%로 놓고 위치 지정
- 시작과 끝 위치만 사용한다면 from, to 키워드 사용 가능

animation-name 속성

- 어떤 애니메이션을 사용할지 구별
- @keyframes 속성에서 만든 애니메이션 '이름'을 사용

```
기본형 animation-name: <키프레임 이름> | none
```

animation-duration 속성

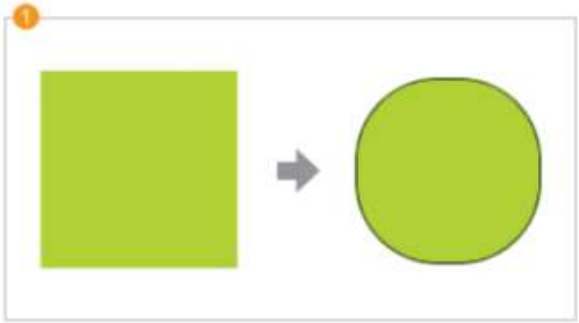
- 애니메이션 실행 시간 설정. 기본값 0
- 사용 가능한 값은 초(s)나 밀리초(ms)

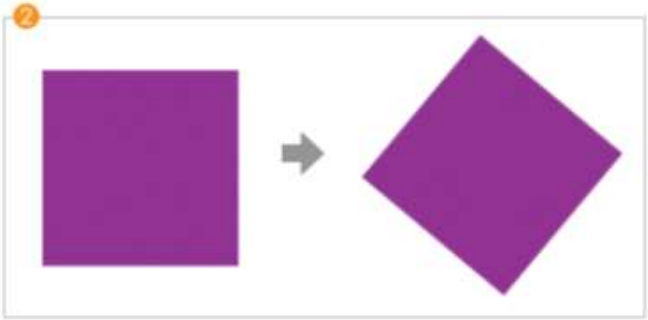
```
기본형 animation-duration: <시간>
```

애니메이션 알아보기

Do it! 애니메이션의 지점, 이름, 실행 시간 적용하기 — 예제 파일 11\ani-1.html

```
(... 생략 ...)
  #box1 {
    background-color: #4cff00;       /* 연두색 박스 */
    border: 1px solid transparent;   /* 1px짜리 투명 테두리 */
    animation-name: shape;           /* 애니메이션 지정 */
    animation-duration: 3s;          /* 애니메이션 실행 시간 */
  }
  #box2 {
    background-color: #8f06b0;       /* 보라색 박스 */
    border: 1px solid transparent;   /* 1px짜리 투명 테두리 */
    animation-name: rotate;          /* 애니메이션 지정 */
    animation-duration: 3s;          /* 애니메이션 실행 시간 */
  }
  @keyframes shape {
    from { border: 1px solid transparent; }  /* 1px짜리 투명 테두리에서 */
    to {
      border: 1px solid #000; /* 검은색 테두리로 변경 */
      border-radius: 50%;     /* 테두리를 둥글게 변경 */
    }
  }
  @keyframes rotate {
    from { transform: rotate(0deg); }  /* 0도에서 시작해서 */
    to { transform: rotate(45deg); }   /* 45도까지 회전하기 */
  }
</style>
......
<div class="box" id="box1"></div>
<div class="box" id="box2"></div>
(... 생략 ...)
```





애니메이션 알아보기

animation-direction 속성

애니메이션이 끝난 후 원래 위치로 돌아가거나 반대 방향으로 실행하도록 지정

기본형 `animation-direction: normal | alternate`

종류	설명
normal	애니메이션을 끝까지 실행하면 원래 위치로 돌아갑니다. 기본값입니다.
alternate	애니메이션을 끝까지 실행하면 왔던 방향으로 되돌아가면서 애니메이션을 실행합니다.

animation-iteration-count 속성

애니메이션 반복 횟수 지정하기

기본형 `animation-iteration-count: <숫자> | infinite`

종류	설명
숫자	애니메이션의 반복 횟수를 정합니다.
infinite	애니메이션을 무한 반복합니다.

Do it! 무한 반복하는 애니메이션 만들기 예제 파일 11\ani-2.html

```
(... 생략 ...)
  #box1 {
    background-color: #4cff00;
    border: 1px solid transparent;
    animation-name: shape;
    animation-duration: 3s;
    animation-iteration-count: infinite;          /* 애니메이션 무한 반복 */
  }
  #box2 {
    background-color: #8f06b0;
    border: 1px solid transparent;
    animation-name: rotate;
    animation-duration: 3s;
    animation-iteration-count: i                ;  /* 애니메이션 무한 반복 */
  }
  @keyframes shape { ..... }
  @keyframes rotate { ..... }
  }
</style>
(... 생략 ...)
```

애니메이션 알아보기

animation-timing-function 속성

애니메이션 속도 곡선 지정

animation 속성

- 여러 개의 애니메이션 속성을 하나의 속성으로 줄여서 사용
- 지정하지 않은 속성은 기본 값 사용.
- animation-duration 속성 값은 반드시 지정해야 함.

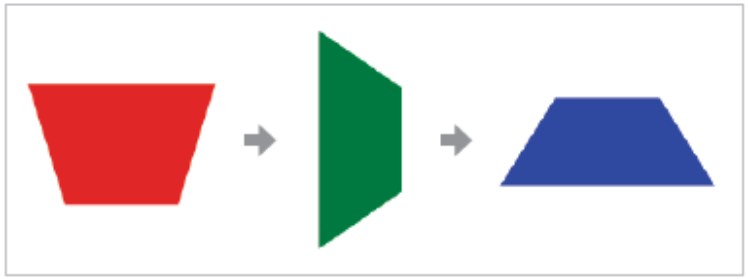
Do it! 애니메이션 2개를 한꺼번에 지정하기 — 예제 파일 11\ani-3.html

```
(... 생략 ...)
  .box {
    width: 100px;
    height: 100px;
    margin: 60px auto;
    animation: rotate 1.5s infinite, background 1.5s infinite alternate;
  }

  @keyframes rotate { /* 0도 -> x축 -180도 회전 -> y축 -180도 회전 */
    from { transform: perspective(120px) rotateX(0deg) rotateY(0deg); }
    50% { transform: perspective(120px) rotateX(-180deg) rotateY(0deg); }
    to { transform: perspective(120px) rotateX(-180deg) rotateY(-180deg); }
  }

  @k                    background {
    from { background-color: red; }   /* 시작 배경색은 빨강 */
    50% { background-color: green; }  /* 중간(50%) 배경색은 초록 */
    to { background-color: blue; }    /* 마지막(100%) 배경색은 파랑 */
  }
</style>
(... 생략 ...)
```

rotate 애니메이션 정의, 1.5초 진행, 무한 반복

background 애니메이션 정의, 1.5초 진행, 무한 반복

정답: @keyframes



12. 반응형 웹과 미디어 쿼리

12-1 반응형 웹 알아보기

12-2 미디어 쿼리 알아보기

12-3 그리드 레이아웃 알아보기

12-4 플렉스 박스 레이아웃 알아보기

12-5 CSS 그리드 레이아웃 알아보기

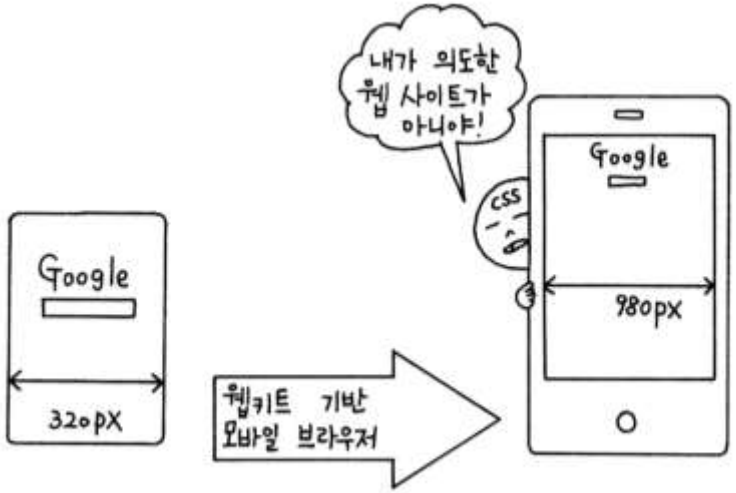
반응형 웹 알아보기

반응형 웹 디자인

- 웹 사이트의 내용을 그대로 유지하면서 다양한 화면 크기에 맞게 웹 사이트를 표시하는 방법
- 다양한 화면 크기의 모바일 기기들이 계속 쏟아져 나오는데 그 때마다 그 크기에 맞춘 사이트를 별도로 제작하는 것은 비효율적
  → 화면 크기에 '반응'해 화면 요소들을 자동으로 바꾸어 사이트를 구현하는 것이 바로 반응형 웹 디자인

모바일 기기를 위한 뷰포트(viewport)

- 뷰포트 : 실제 내용이 표시되는 영역
- PC 화면과 모바일 화면의 픽셀 표시 방법이 다르기 때문에
  모바일 화면에서 의도한대로 표시되지 않음
  → 뷰포트를 지정하면 기기 화면에 맞춰 확대/축소해서 내용 표시



반응형 웹 알아보기

뷰포트 지정하기

기본형 `<meta name="viewport" content="속성1=값1", "속성2=값2", .....">`

종류	설명	사용 가능한 값	기본값
width	뷰포트 너비	device-width 또는 크기	브라우저 기본값
height	뷰포트 높이	device-height 또는 크기	브라우저 기본값
user-scalable	확대·축소 가능 여부	yes 또는 no (yes는 1로, device-width와 device-height값은 10으로 간주)	yes
initial-scale	초기 확대·축소 값	1~10	1

일반적인 사용법

뷰포트의 너비를 스마트폰 화면 너비에 맞추고 초기 화면 배율을 1로 지정

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

뷰포트 단위

- vw(viewport width): 1vw는 뷰포트 너비의 1%와 같습니다.
- vh(viewport height): 1vh는 뷰포트 높이의 1%와 같습니다.
- vmin(viewport minimum): 뷰포트의 너비와 높이 중에서 작은 값의 1%와 같습니다.
- vmax(viewport maximum): 뷰포트의 너비와 높이 중에서 큰 값의 1%와 같습니다.

Do it! 뷰포트 단위를 사용해 글자 크기 조절하기 — 예제 파일 12\vw.html

```
(... 생략 ...)
<style>
  h1 {
    font-size: 5vw;
    text-align: center;
  }
</style>
.....
<h1>안녕하세요?</h1>
(... 생략 ...)
```

안녕하세요?

안녕하세요?

미디어 쿼리 알아보기

미디어 쿼리(media queries) : 접속하는 장치(미디어)에 따라 특정한 CSS 스타일을 사용하는 방법

미디어 쿼리를 이용해 제작된 사이트들을 모아놓은 http://mediaqueri.es 참고

미디어 쿼리 구문

- @media 속성을 사용해 특정 미디어에서 어떤 CSS를 적용할 것인지 지정함
- <style> 태그와 </style> 태그 사이에 사용

기본형 `@media [only ¦ not] 미디어 유형 [and 조건] * [and 조건]`

❶ only: 미디어 쿼리를 지원하지 않는 웹 브라우저에서는 미디어 쿼리를 무시하고 실행하지 않습니다.
❷ not: not 다음에 지정하는 미디어 유형을 제외합니다. 예를 들어 not tv라고 지정하면 TV를 제외한 미디어 유형에만 적용합니다.
❸ and: 조건을 여러 개 연결해서 추가할 수 있습니다.

예) 미디어 유형이 screen이면서 최소 너비가 768px이고
최대 너비는 1439px일 경우에 적용할 CSS

```
@media screen and (min-width: 768px) and (max-width: 1439px) {
 (... 생략 ...)
}
```

미디어 유형의 종류

종류	설명
all	모든 미디어 유형에서 사용할 CSS를 정의합니다.
print	인쇄 장치에서 사용할 CSS를 정의합니다.
screen	컴퓨터 스크린에서 사용할 CSS를 정의합니다. 스마트폰의 스크린도 포함됩니다.
tv	음성과 영상이 동시에 출력되는 TV에서 사용할 CSS를 정의합니다.
aural	음성 합성 장치(주로 화면을 읽어 소리로 출력해 주는 장치)에서 사용할 CSS를 정의합니다.
braille	점자 표시 장치에서 사용할 CSS를 정의합니다.
handheld	패드(pad)처럼 손에 들고 다니는 장치를 위한 CSS를 정의합니다.
projection	프로젝터를 위한 CSS를 정의합니다.
tty	디스플레이 기능이 제한된 장치에 맞는 CSS를 정의합니다. 이런 장치에서는 픽셀(px) 단위를 사용할 수 없습니다.
embossed	점자 프린터에서 사용할 CSS를 정의합니다.

미디어 쿼리 알아보기

미디어 쿼리의 다양한 조건

웹 문서의 가로 너비와 세로 높이(뷰포트)

실제 웹 문서 내용이 나타나는 영역의
너비와 높이를 조건으로 사용

종류	설명
width, height	웹 페이지의 가로 너비, 세로 높이
min-width, min-height	웹 페이지의 최소 너비, 최소 높이
max-width, max-height	웹 페이지의 최대 너비, 최대 높이

단말기의 가로 너비와 세로 높이

대부분의 단말기 해상도와
실제 브라우저의 너비가 다르다는 점에 주의

종류	설명
device-width, device-height	단말기의 가로 너비, 세로 높이
min-device-width, min-device-height	단말기의 최소 너비, 최소 높이
max-device-width, max-device-height	단말기의 최대 너비, 최대 높이

화면 회전

스마트폰이나 태블릿에서 기기를 가로나
세로로 돌려보는지 확인

종류	설명
orientation: portrait	단말기의 세로 모드
orientation: landscape	단말기의 가로 모드

예) 화면 너비가 1440px 이상일 때

```
@media screen and (min-width: 1440px) {  /* 너비가 최소 1440px인 화면용 스타일 */
  (... 생략 ...)
}
```

예) iPhoneX 를 가로로 돌릴 경우

```
@media screen and (min-device-width: 375px) and (min-device-height: 812px) {
  (... 생략 ...)
}
```

미디어 쿼리 알아보기

미디어 쿼리 중단점

- **중단점(breakpoint)** : 서로 다른 CSS를 적용할 화면 크기
- 대부분 기기의 화면 크기 기준.
- 모든 기기를 반영할 수 없기 때문에 스마트폰과 태블릿, 데스크톱 정도로 구분
- 모바일 퍼스트(mobile first) : 모바일 기기 레이아웃을 기본으로 작성 → 태블릿 & PC 레이아웃 작성

일반적인 사용 예

(미디어 쿼리 중단점은 개발자나 작업 조건에 따라 달라질 수 있다.)

- **스마트폰**: 모바일 페이지는 미디어 쿼리를 사용하지 않고 기본 CSS로 작성. 만일 스마트폰의 방향까지 고려해서 제작한다면 min-width의 세로와 가로를 각각 portrait 320px, landscape 480px로 지정.
- **태블릿**: 세로 크기가 768px 이상이면 태블릿으로 지정. 가로 크기는 데스크톱과 똑같이 1024px 이상으로 지정.
- **데스크톱**: 화면 크기가 1024px 이상이면 데스크톱으로 설정.

미디어 쿼리 알아보기

미디어 쿼리 적용하기

1. 외부 CSS 파일 연결

방법1) <link> 태그 사용하기

<head>와 </head> 사이에 삽입

```
기본형 <link rel="stylesheet" media="미디어 쿼리 조건" href="css 파일 경로">
```

예)
```
<link rel="stylesheet" media="print" href="css/print.css">
```

방법2) @import 구문 사용하기

<style>와 </style> 사이에 삽입

```
기본형 @import url(css 파일 경로) 미디어 쿼리 조건
```

예)
```
@import url("css/tablet.css") only screen and (min-width: 321px) and (max-width: 768px);
```

2. 웹 문서에서 직접 정의하기

방법1) <style> 태그 안에서 media 속성 사용하기

```
기본형 <style media="<조건>"> {

        }
        </style>
```

예)
```
<style media="screen and (max-width: 320px)">
  body {
    background-color: orange;
  }
</style>
```

방법2) @media문 사용하기

```
기본형 <style>
         @media <조건> {

         }
       </style>
```

예)
```
<style>
  @media screen and (max-width: 320px) {
    body {
      background-color: orange;
    }
  }
</style>
```

그리드 레이아웃 알아보기

그리드 레이아웃이란

- 반응형 웹 디자인에서 웹 문서 요소를 배치하는 기준
- 웹 사이트 화면을 여러 개의 칼럼(column)으로 나눈 후 웹 요소를 배치
- 화면을 규칙적으로 배열하므로 레이아웃을 일관성 있게 유지할 수 있음

그리드 레이아웃의 특징

- 시각적으로 안정된 디자인
- 업데이트가 편한 웹 디자인 구성
- 요소를 자유롭게 배치

그리드 레이아웃 알아보기

그리드 레이아웃을 만드는 방법

1) 플렉스 박스 레이아웃(플렉서블 박스 레이아웃)

- 수평 방향이나 수직 방향 중 하나를 주축으로 정하고 박스를 배치
- 여유 공간이 생길 경우 너비나 높이를 적절하게 늘리거나 줄일 수 있음

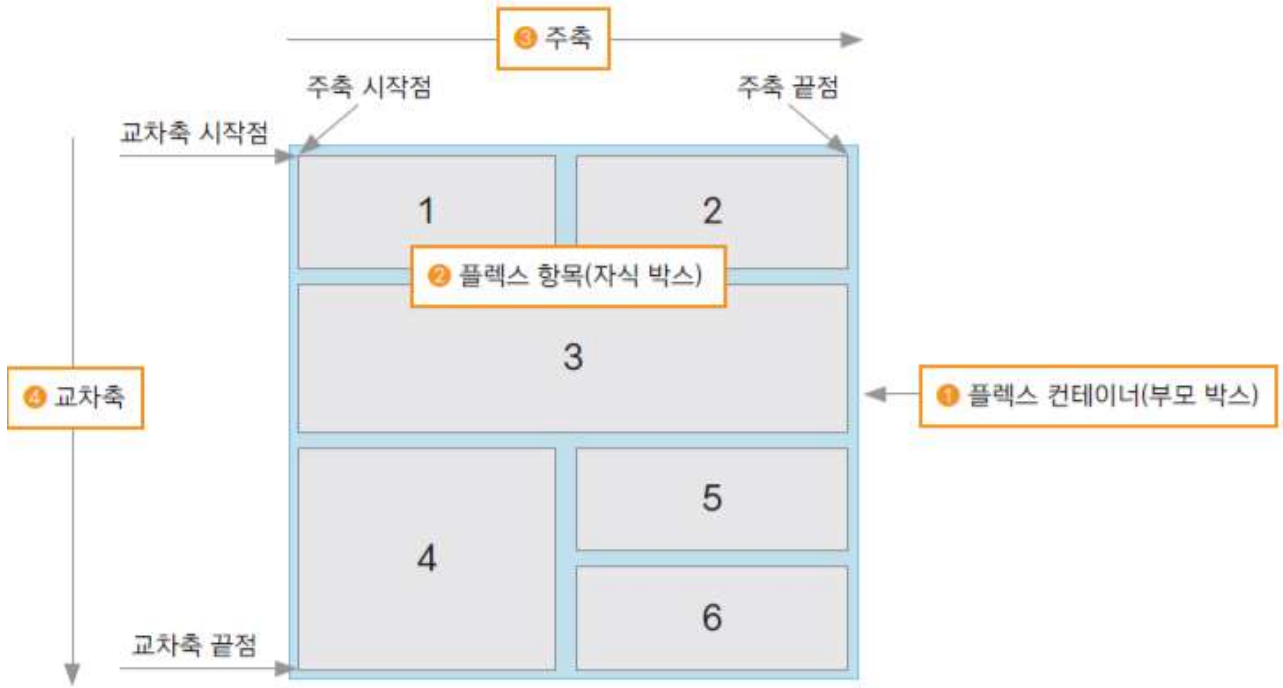
2) CSS 그리드 레이아웃

- 최근에 등장한 CSS 표준
- 수평 방향이나 수직 방향 어디로든 배치 가능
- 마치 레고 블록을 끼워 맞추듯 요소를 배치할 수 있음

플렉스 박스 레이아웃

플렉스 박스 레이아웃(flex box layout)

- 그리드 레이아웃을 기본으로, 플렉스 박스를 원하는 위치에 배치하는 것.
- 여유 공간에 따라 너비나 높이, 위치를 자유롭게 변형할 수 있음



❶ **플렉스 컨테이너(부모 박스):** 플렉스 박스 레이아웃을 적용할 대상을 묶는 요소입니다.

❷ **플렉스 항목(자식 박스):** 플렉스 박스 레이아웃을 적용할 대상으로 그림 12-18에서 1~6까지 작은 박스들이 모두 해당합니다.

❸ **주축(main axis):** 플렉스 컨테이너 안에서 플렉스 항목을 배치하는 기본 방향입니다. 기본적으로 왼쪽에서 오른쪽이며 수평 방향으로 배치합니다. 플렉스 항목의 배치가 시작되는 위치를 '주축 시작점', 끝나는 위치를 '주축 끝점'이라고 합니다.

❹ **교차축(cross axis):** 주축과 교차하는 방향을 말하며 기본적으로 위에서 아래로 배치합니다. 플렉스 항목의 배치가 시작되는 위치를 '교차축 시작점', 끝나는 위치를 '교차축 끝점'이라고 합니다.

플렉스 박스 레이아웃

플렉서블 박스 레이아웃 기본 속성

display 속성

배치 요소들을 감싸는 부모 요소를 플렉스 컨테이너로 지정

종류	설명
flex	컨테이너 안의 플렉스 항목을 블록 레벨 요소로 배치합니다.
inline-flex	컨테이너 안의 플렉스 항목을 인라인 레벨 요소로 배치합니다.

flex-direction 속성

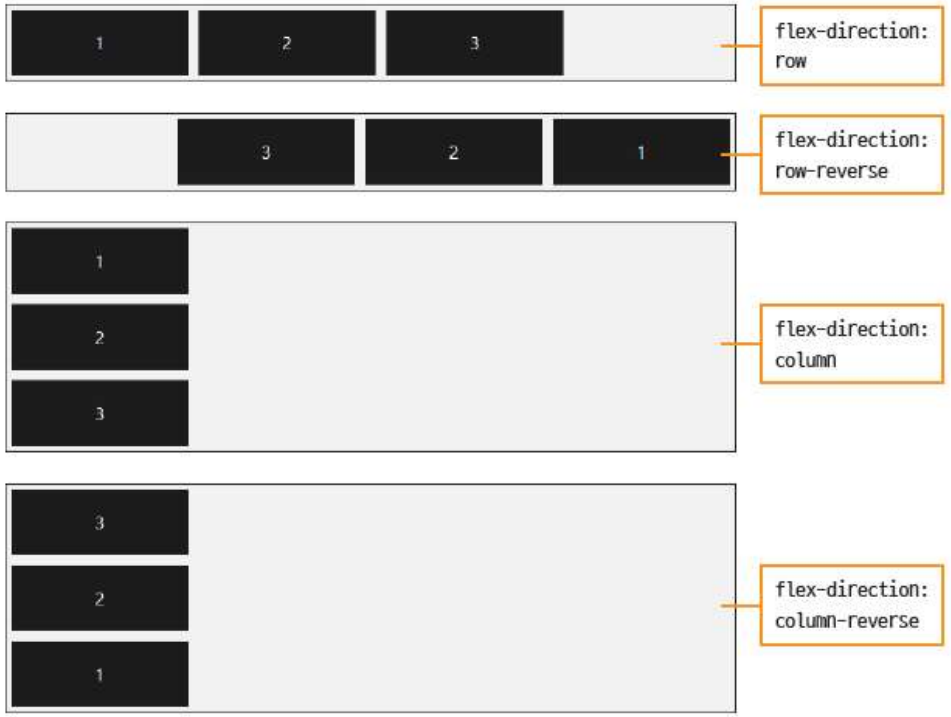
플렉스 항목의 배치를 위해 주축과 방향 지정.

종류	설명
row	주축을 가로로 지정하고 왼쪽에서 오른쪽으로 배치합니다. 기본값입니다.
row-reverse	주축을 가로로 지정하고 반대로 오른쪽에서 왼쪽으로 배치합니다.
column	주축을 세로로 지정하고 위쪽에서 아래쪽으로 배치합니다.
column-reverse	주축을 세로로 지정하고 아래쪽에서 위쪽으로 배치합니다.

Do it! 플렉스 항목 배치하기 — 예제 파일 12\flex-1.html

```
(... 생략 ...)
<style>
  .container { display: flex; }                    /* 플렉스 컨테이너 지정 */
  ......
  #opt1 { flex-direction: row; }                   /* 왼쪽에서 오른쪽으로 */
  #opt2 { flex-direction: row-reverse; }           /* 오른쪽에서 왼쪽으로 */
  #opt3 { flex-direction: column; }                /* 위에서 아래로 */
  #opt4 { flex-direction: column-reverse; }        /* 아래에서 위로 */
(... 생략 ...)
```

정답: reverse



플렉스 박스 레이아웃

플렉서블 박스 레이아웃 기본 속성

flex-wrap 속성

플렉스 항목을 한 줄 또는 여러 줄로 배치

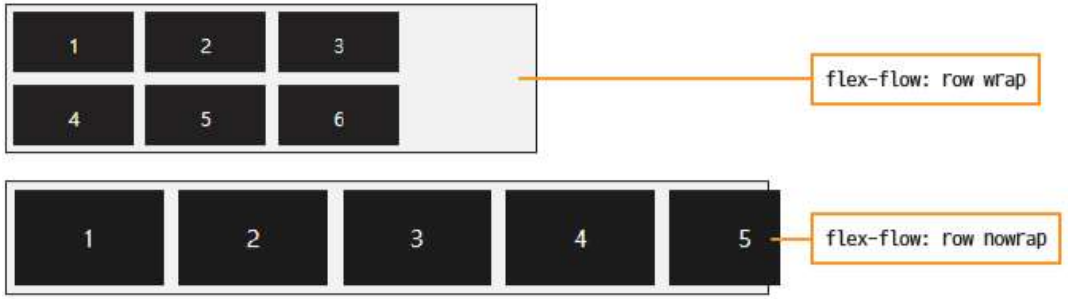
종류	설명
nowrap	플렉스 항목을 한 줄에 표시합니다. 기본값입니다.
wrap	플렉스 항목을 여러 줄에 표시합니다.
wrap-reverse	플렉스 항목을 여러 줄에 표시하되, 시작점과 끝점이 바뀝니다.

flex-flow 속성

- 플렉스 배치 방향과 여러 줄 배치를 한꺼번에 지정
- 기본 값은 flex-flow:row no-wrap

Do it! 플렉스 항목의 배치 방향과 줄바꿈 적용하기 — 예제 파일 12\flex-3.html

```
(... 생략 ...)
<style>
  ......
  #opt1 { flex-flow: row wrap; }    /* 왼쪽에서 오른쪽, 여러 줄 */
  #opt2 { flex-flow: row nowrap; }  /* 왼쪽에서 오른쪽, 한 줄 */
(... 생략 ...)
```



플렉스 박스 레이아웃

플렉스 항목 배치를 위한 속성

justify-content 속성

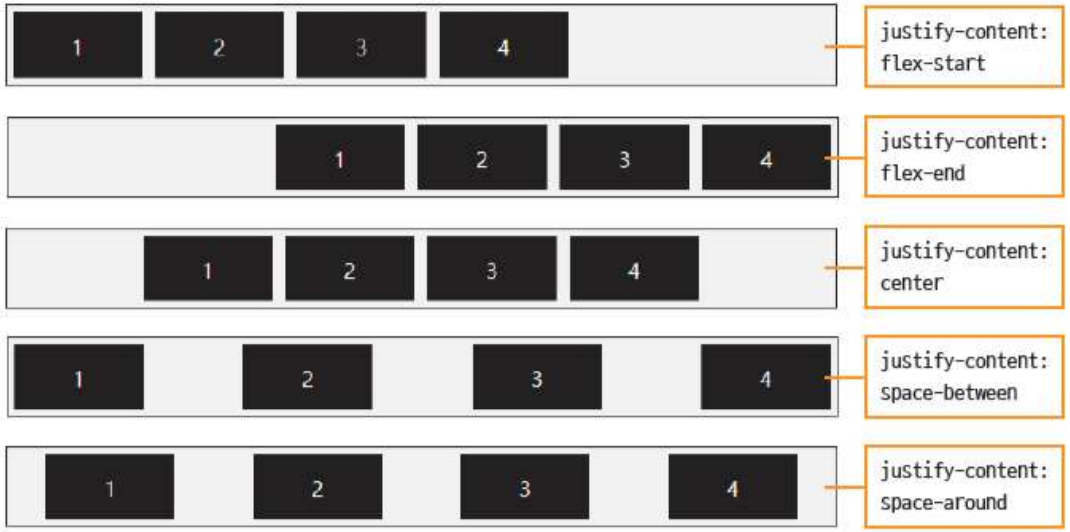
플렉스 항목을 주축 방향으로 배치할 때의 배치 기준

종류	설명
flex-start	주축의 시작점에 맞춰 배치합니다.
flex-end	주축의 끝점에 맞춰 배치합니다.
center	주축의 중앙에 맞춰 배치합니다.
space-between	첫 번째 항목과 끝 항목을 주축의 시작점과 끝점에 배치한 후 나머지 항목은 그 사이에 같은 간격으로 배치합니다.
space-around	모든 항목을 주축에 같은 간격으로 배치합니다.

Do it! 주축에서 플렉스 항목 간의 간격 적용하기 — 예제 파일 12\flex-4.html

```
(... 생략 ...)
<style>
  ......
  #opt1 { justify-content: flex-start; }      /* 주축 시작점 기준 */
  #opt2 { justify-content: flex-end; }        /* 주축 끝점 기준 */
  #opt3 { justify-content: center; }          /* 주축 중앙 기준 */
  #opt4 { justify-content: space-between; }   /* 시작점과 끝점 배치한 후 같은 간격 */
  #opt5 { justify-content: space-around; }    /* 전체 항목 같은 간격 */
(... 생략 ...)
```

정답: center



플렉스 박스 레이아웃

플렉스 항목 배치를 위한 속성

align-items 속성, align-self 속성

해당 항목 자체에 사용

교차축을 기준으로 하는 배치 방법 조절

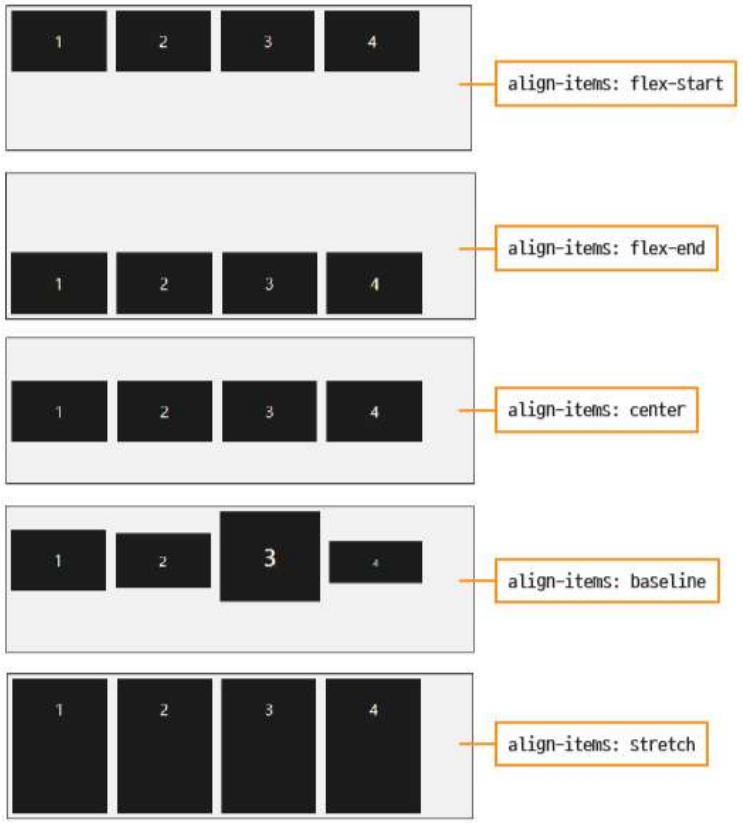
교차축에서 특정 항목만 지정하고 싶다면 align-self 속성 사용

종류	설명
flex-start	교차축의 시작점에 맞춰 배치합니다.
flex-end	교차축의 끝점에 맞춰 배치합니다.
center	교차축의 중앙에 배치합니다.
baseline	교차축의 문자 기준선에 맞춰 배치합니다.
stretch	플렉스 항목을 늘려 교차축에 가득 차게 배치합니다.

Do it! 플렉스 박스에서 교차축 정렬 방법 지정하기 — 예제 파일 12\flex-5.html

```
(... 생략 ...)
<style>
  ......
  #opt1 { align-items: flex-start; }      /* 주축 시작점 기준으로 배치 */
  #opt2 { align-items: flex-end; }        /* 주축 끝점 기준으로 배치 */
  #opt3 { align-items: center; }          /* 주축 중앙 기준으로 배치 */
  #opt4 { align-items: baseline; }        /* 문자 기준선에 맞춰 배치 */
  #opt5 { align-items: s          ; } /* 항목을 늘려 교차축에 가득 차게 배치 */
(... 생략 ...)
```

정답: stretch



플렉스 박스 레이아웃

플렉스 항목 배치를 위한 속성

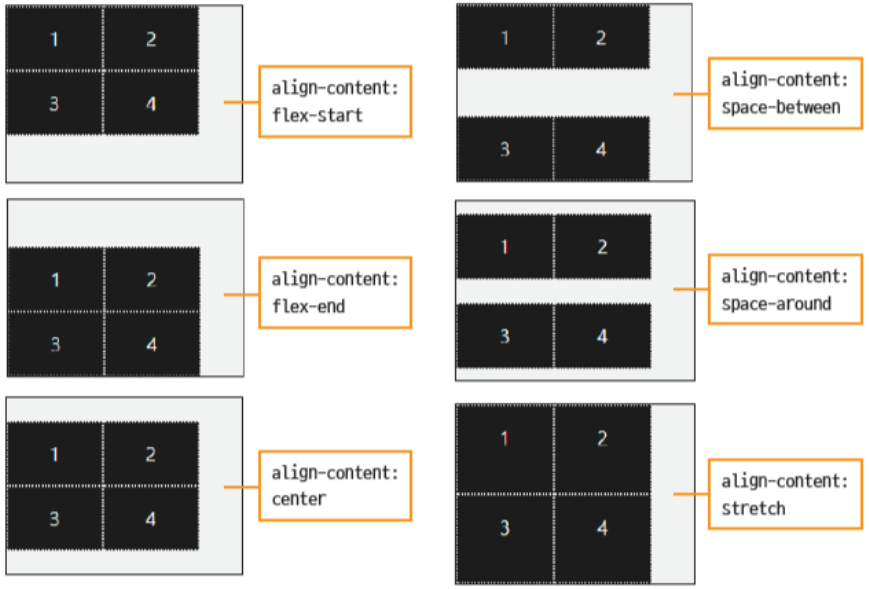
align-content 속성

플렉스 항목이 여러 줄로 표시될 때 교차 축 기준의 배치 방법 지정

종류	설명
flex-start	교차축의 시작점에 맞춰 배치합니다.
flex-end	교차축의 끝점에 맞춰 배치합니다.
center	교차축의 중앙에 맞춰 배치합니다.
space-between	첫 번째 항목과 끝 항목을 교차축의 시작점과 끝점에 맞추고 나머지 항목은 그 사이에 같은 간격으로 배치합니다.
space-around	모든 항목을 교차축에 같은 간격으로 배치합니다.
stretch	플렉스 항목을 늘려서 교차축에 가득 차게 배치합니다.

Do it! 여러 줄일 때 교차축에서 플렉스 항목 간의 간격 지정하기 — 예제 파일 12\flex-7.html

```
(... 생략 ...)
<style>
  .container {
    ......
    display: flex;          /* 플렉스 컨테이너 지정 */
    flex-flow: row wrap;    /* 왼쪽에서 오른쪽, 여러 줄 표시 */
  }
  #opt1 { align-content: flex-start; }        /* 교차축 시작점 기준 */
  #opt2 { align-content: flex-end; } /* 교차축 끝점 기준 */
  #opt3 { align-content: center; }            /* 교차축 중앙 기준 */
  #opt4 { align-content: space-between; }     /* 시작점과 끝점에 배치한 뒤 중간 항목
은 같은 간격으로 배치 */
  #opt5 { align-content: space-around; } /* 전체 항목을 같은 간격으로 배치 */
  #opt6 { align-content: stretch; }      /* 항목을 늘려 교차축에 가득 차게 배치 */
(... 생략 ...)
```



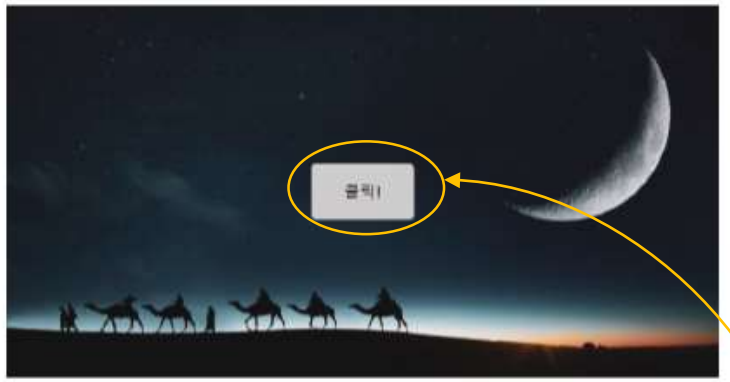
플렉스 박스 레이아웃

플렉스 박스 레이아웃을 사용해 화면 중앙에 배치하기

플렉스 박스 레이아웃을 가장 많이 사용하는 예

Do it! 플렉스 박스 레이아웃을 사용해 화면 중앙에 배치하기 — 예제 파일 12\center.html

```
(... 생략 ...)
<style>
  * {
    margin: 0;
    box-sizing: border-box;
  }
  body {
    background: url('images/bg5.jpg') no-repeat left top fixed;
    background-size: cover;
    display: flex;
    justify-content: center;
    align-items: center;
    min-height: 100vh;
  }
  button { ...... }
......
<button>클릭!</button>
(... 생략 ...)
```



화면 크기에 상관없이 버튼은 **항상 화면 중앙**에 표시됨

CSS 그리드 레이아웃

CSS 그리드 레이아웃

- 플렉스 박스 레이아웃은 주축/교차축 개념이 있지만 CSS 그리드 레이아웃은 양쪽 방향 모두 사용
  (플렉스 그리드 레이아웃은 1차원, CSS 그리드 레이아웃은 2차원이라고도 함)
- 줄(row)과 칼럼(column)으로 화면을 구성하고, 줄 사이의 여백, 칼럼 사이의 여백을 조절.

CSS 그리드 레이아웃

CSS 그리드 레이아웃 항목을 배치하는 속성

display 속성

배치 요소들을 감싸는 부모 요소를 그리드 컨테이너로 지정

종류	설명
gird	컨테이너 안의 항목을 블록 레벨 요소로 배치합니다.
inline-grid	컨테이너 안의 항목을 인라인 레벨 요소로 배치합니다.

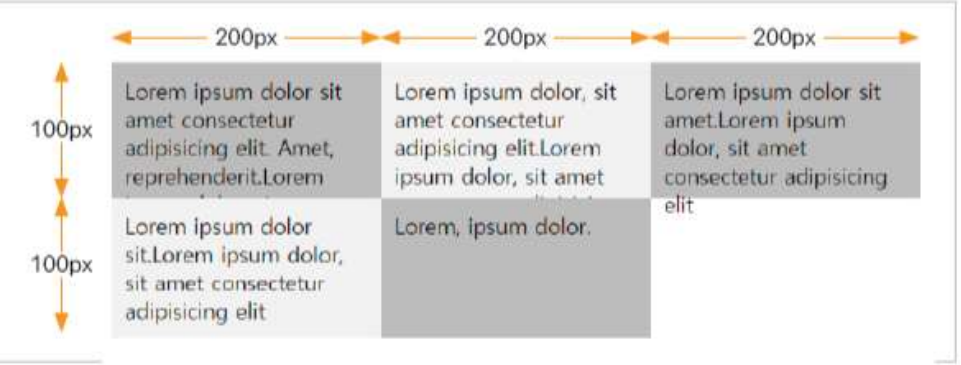
grid-template-columns, grid-template-rows 속성

칼럼/줄의 크기와 개수 지정

- grid-template-columns : 그리드 컨테이너 안의 칼럼 개수와 너빗값
- grid-template-rows: 그리드 컨테이너 안의 줄 개수와 너빗값

Do it! 그리드 박스에서 칼럼과 줄 지정하기 — 예제 파일 12\grid-1.html

```
(... 생략 ...)
<style>
  #wrapper {
    display: grid;                                  /* 그리드 컨테이너 지정 */
    grid-template-columns: 200px 200px 200px;       /* 너비가 200px인 칼럼 3개 */
    grid-template-rows: 100px;                      /* 줄 높이 100px */
  }
  ......
<div id="wrapper">
  <div class="items">Lorem ipsum dolor sit.....</div>
  <div class="items">Lorem .....</div>
  <div class="items">Lorem .....</div>
  <div class="items">Lorem .....</div>
  <div class="items">Lorem .....</div>
</div>
(... 생략 ...)
```



CSS 그리드 레이아웃

상대적인 크기를 지정하는 fr 단위

칼럼/줄의 크기를 지정할 때 px 단위는 반응형 웹 디자인에 적합하지 않음 → 상대적인 크기를 지정하는 **fr**(fraction) 단위 사용

예) 너비의 비율이 2:1:2인 칼럼 3개를 배치한다면

```
grid-template-columns: 2fr 1fr 2fr;
```

값이 반복된다면 repeat( ) 함수

똑같은 값을 여러 번 반복한다면 내장 함수 repeat( ) 함수 사용

```
grid-template-columns: 1fr 1fr 1fr;
```

↓

```
grid-template-columns: repeat(3, 1fr);
```

자동으로 칼럼 개수를 조절하는 auto-fill, auto-fit

칼럼 너비와 함께 auto-fit이나 auto-fill을 지정하면 화면 너비에 따라 칼럼 개수를 조절할 수 있음

- auto-fit : 남는 공간 없이 꽉 채우기
- auto-fill : 칼럼의 최소 너비까지만 표시하고 남는 공간은 그대로 둠

예) 너비 200px인 칼럼을 화면에 가득찰 때까지 배치

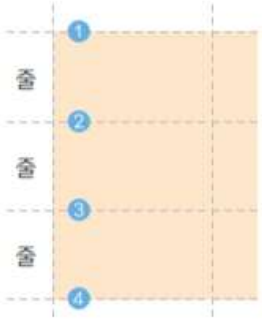
```
grid-template-columns: repeat(auto-fit, 200px);
```

```
<style>
  #wrapper1 {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
    margin-bottom: 20px;
  }
  #wrapper2 {
    display: grid;
    grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));
  }
```

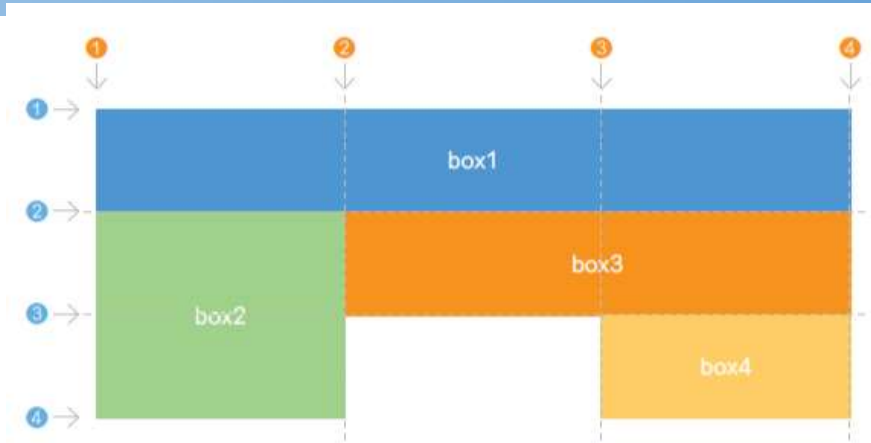
CSS 그리드 레이아웃

그리드 라인을 사용해 배치하기

- CSS 그리드 레이아웃에는 눈에 보이지 않는 그리드 라인이 포함되어 있음
- 그리드 라인을 사용해 그리드 항목을 배치할 수 있음



종류	설명	예시
grid-column-start	칼럼 시작의 라인 번호를 지정합니다.	grid-column-start: 1
grid-column-end	칼럼 마지막의 라인 번호를 지정합니다.	grid-column-end: 1
grid-column	칼럼 시작 번호와 칼럼 끝 번호 사이에 슬래시(/)를 넣어 사용합니다.	grid-column: 1/4
grid-end-start	줄 시작의 라인 번호를 지정합니다.	grid-end-start: 2
grid-row-end	줄 마지막의 라인 번호입니다.	grid-row-end: 4
grid-row	줄 시작 번호와 줄 끝 번호 사이에 슬래시(/)를 넣어 사용합니다.	grid-row: 2/4




```
<style>
  #wrapper {
    width: 700px;
    display: grid;
    grid-template-columns: repeat(3, 1fr);
    grid-template-rows: repeat(3, 100px);
  }
```

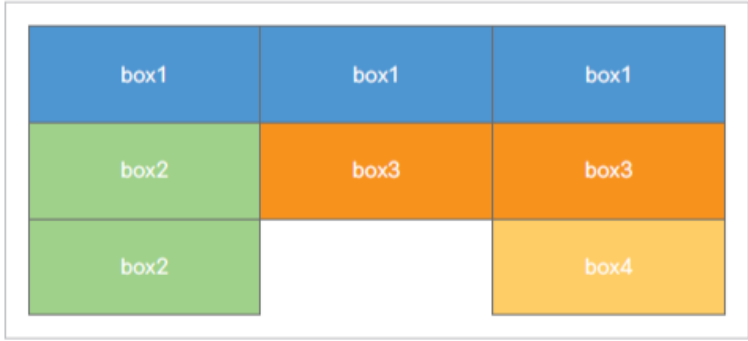
```
.box1 {
  background-color: #3689ff;
  grid-column: 1/4;
}
.box2 {
  background-color: #00cf12;
  grid-row: 2/4;
}
.box3 {
  background-color: #ff9019;

  grid-column: 2/4;
  grid-row-start: 2;
}
.box4 {
  background-color: #ffd000;
  grid-column-start: 3;
  grid-row-start: 3;
}
```

CSS 그리드 레이아웃

템플릿 영역을 만들어 배치하기

grid-area 속성을 사용해 템플릿 영역을 만든 후 배치

예) box1 ~ box4까지 네 개의 템플릿 영역을 만든 후 배치



1) 템플릿 영역 만들기

```
.box1 {
  background-color: #3689ff;
  grid-area: box1;
}
.box2 {
  background-color: #00cf12;
  grid-area: box2;
}
.box3 {
  background-color: #ff9019;
  grid-area: box3;
}
.box4 {
  background-color: #ffd000;
  grid-area: box4;
}
```

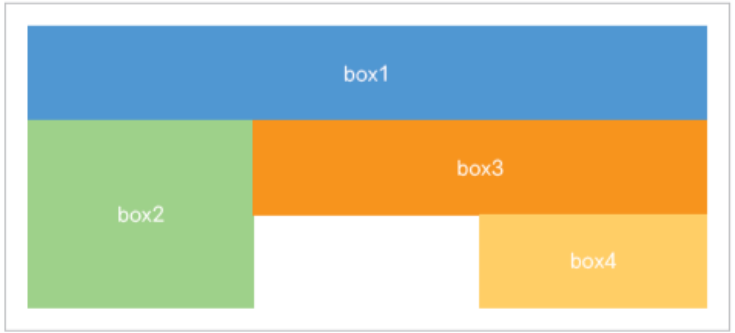
2) 템플릿 영역 배치하기

```
#wrapper {
  width: 700px;
  display: grid;
  grid-template-columns: repeat(3, 1fr);
  grid-template-rows: repeat(3, 100px);
  grid-template-areas:
    "box1 box1 box1"
    "box2 box3 box3"
    "box2 . box4";
}
```

한 줄에 들어갈 템플릿 영역을 큰따옴표(" ")로 묶습니다.

빈 영역은 마침표(.)를 넣습니다.

3) 결과 화면

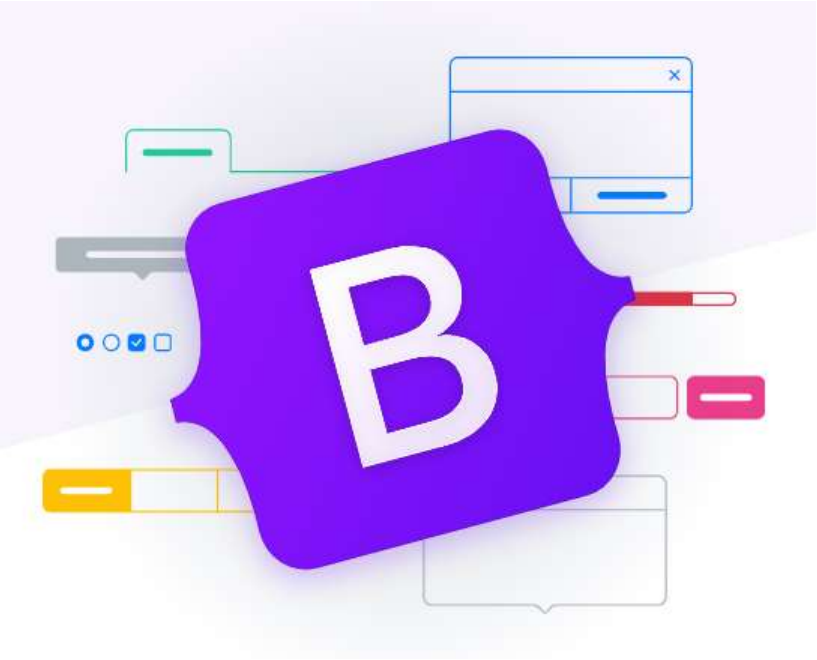


예제. 블로그 만들기 실습

- https://github.com/lbcsultan/blog
- https://lbcsultanblog.netlify.app/

Bootstrap 소개

- CSS 프레임워크
  - 복잡한 CSS 문법을 사용하지 않고도 CSS를 편리하게 사용할 수 있도록 함
  - Bootstrap, Tailwind, MaterialUI 등
- Bootstrap이란?
  - 반응형, 모바일 우선 웹페이지를 빠르게 작성할 수 있는 편리한 오픈소스 CSS 프레임워크
  - 트위터사에서 개발하여 공개
  - https://getbootstrap.com/
  - 현재버전: v5.3 (2023.3)



Bootstrap 실습

- Get Started (시작)
  - CSS setup
  - JS setup
- Layout (레이아웃 이용)
  - .container
  - .container-fluid
  - .container-{breakpoint}
  - Columns
- Content
  - Typography
  - Images
  - Tables
- Forms
  - Input, select, textarea

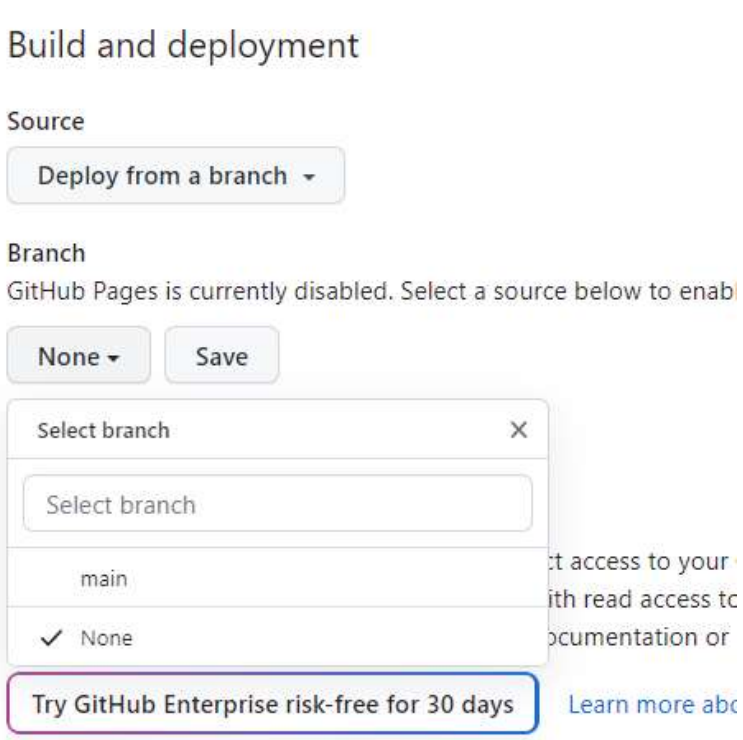
- Components
  - Button, card, carousel
  - Navbar (메뉴)
- Icons
- Themes

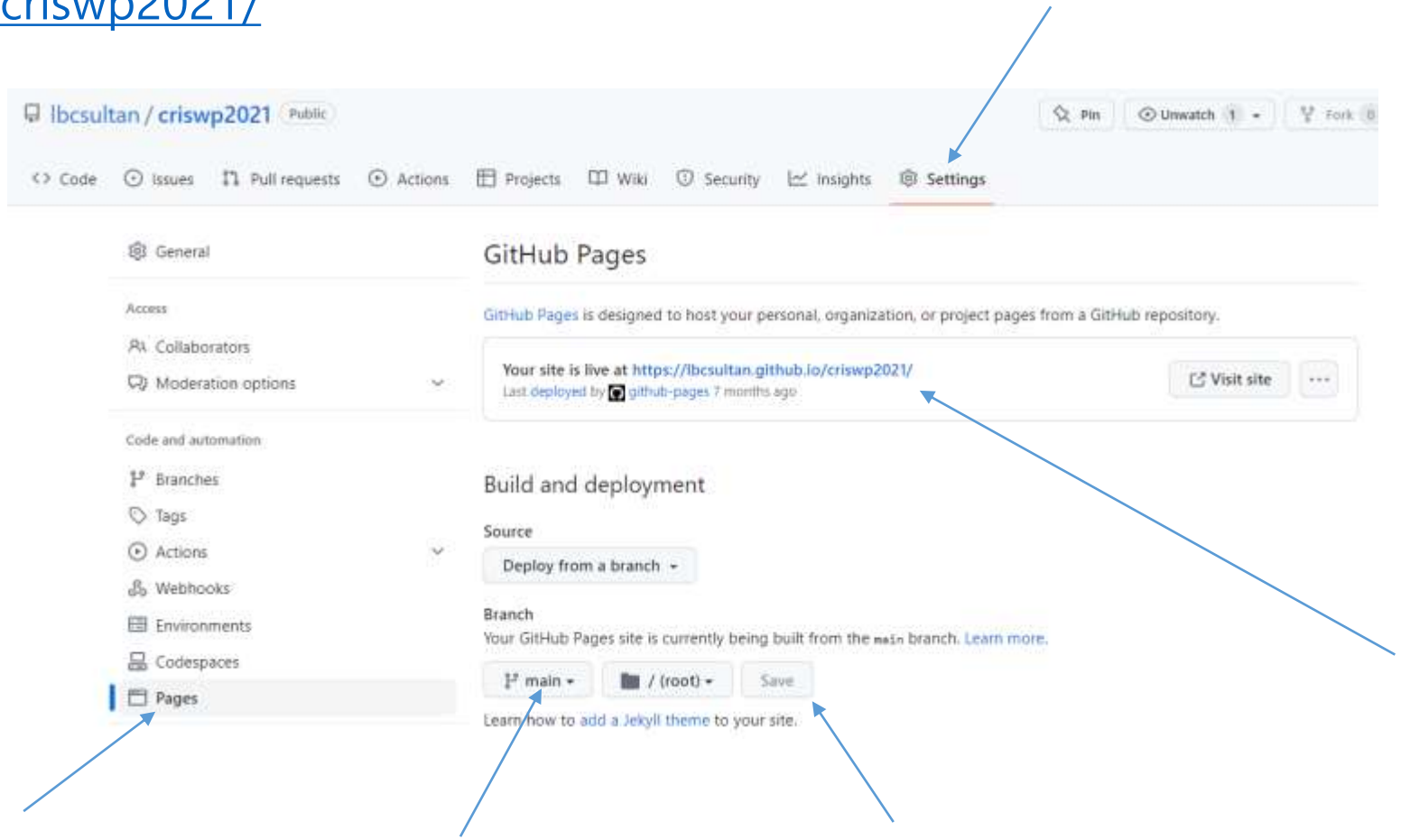
예제. Bootstrap 홈페이지 만들기

- https://github.com/lbcsultan/criswp2021
- https://criswp2021.netlify.app/index.html

Github pages

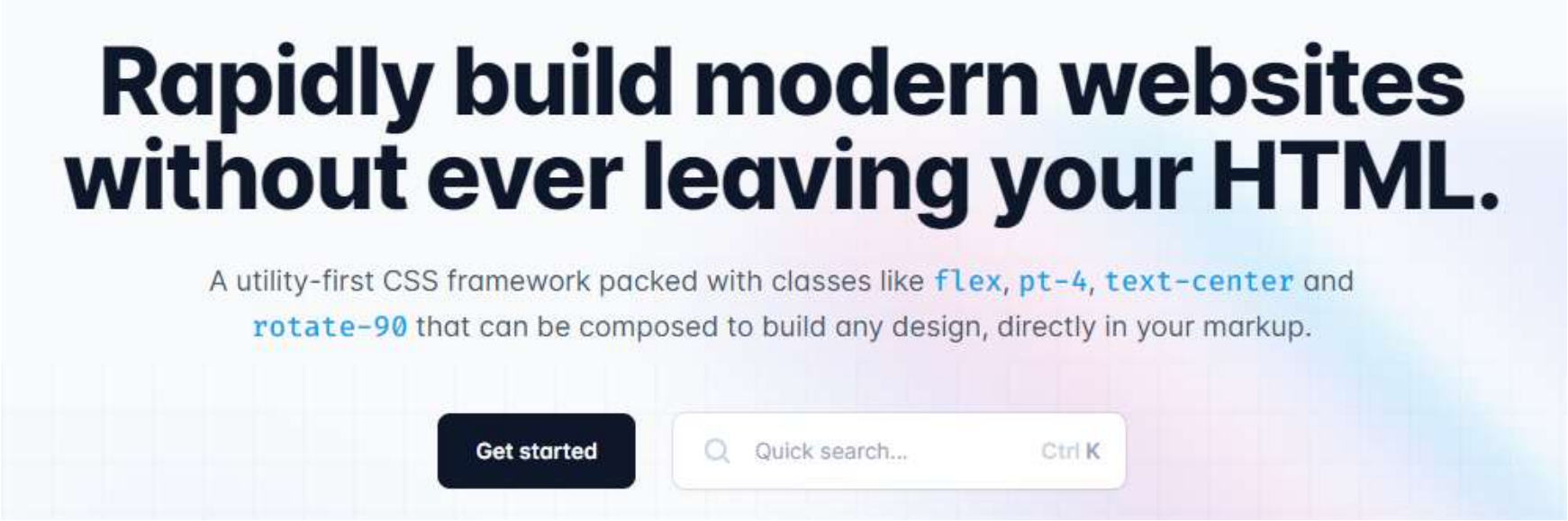
- 소스를 관리하는 Github.com에서 직접 정적 웹서비스를 제공
  - Settings > Pages > Active
  - https://lbcsultan.github.io/criswp2021/





Tailwind CSS

- Tailwind CSS
  - https://tailwindcss.com/
  - Utility-First CSS Framework: 세분화된 설정마다 외우기 쉬운 이름의 클래스를 미리 만들어 놓음.
  - HTML 페이지에서 클래스 이름을 지정하여 CSS를 직접 세밀하게 설정 가능
  - 많은 클래스 이름을 사용하면서 코드가 길어지는 단점
  - 유튜브 동영상 강좌 활용



Tailwind CSS

- Installation
  - https://tailwindcss.com/docs/installation
  - Play CDN
- Vscode 확장 프로그램
  - IntelliSense for VS Code 설치
- 각종 개념 숙지
  - Layout
  - Flexbox & Grid
  - Spacing, Sizing, Typography
  - Background, Borders

<head> 영역에 추가
<script src="https://cdn.tailwindcss.com"></script>

index.html

```html
<!doctype html>
<html>
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale
  <script src="https://cdn.tailwindcss.com"></script>
</head>
<body>
  <h1 class="text-3xl font-bold underline">
    Hello world!
  </h1>
</body>
</html>
```

Tailwind CSS

- Tailwind UI components / templates
  - https://tailwindui.com/

- 예제 사이트
  - https://github.com/bradtraversy/tailwind-landing-page

- 동영상 강의
  - https://www.youtube.com/watch?v=EoDYYPUooKY