

HTML5

(Web Programming)

1장 웹 개요와 실습 환경 구축

1.1 인터넷과 웹소개

- 1) 월드 와이드 웹(WWW:World Wide Web)은 인터넷에 연결된 컴퓨터를 통해 사람들이 정보를 공유할 수 있는 정보 공간을 말한다. 인터넷은 전 세계를 연결하고 있는 국제 정보 통신망이고 웹은 인터넷에서 작동하는 서비스이다.
- 2) W3C(World Wide Web Consortium)는 1994년 팀 버너스 리가 만든 HTML 표준을 비롯한 웹 표준안을 제작하거나 제안하는 일을 하는 국제적인 웹 표준화 단체이다.

1.2 웹의 동작 원리

- 1) 웹 브라우저에 원하는 웹 서버 주소(웹 페이지 URL)를 입력하면 클라이언트의 웹 브라우저를 통하여 해당 웹 페이지를 제공한다.
- 2) 네이버 또는 웹 서비스를 제공하는 기업을 서버라고 하고 서버로부터 웹 페이지를 제공 받는 일반 사용자를 클라이언트라고 한다.
- 3) 서버 프로그램은 자바, C#, 루비, 파이썬, 자바스크립트 같은 프로그래밍 언어로 개발하며 기본 웹 프레임워크부터 MVC 프레임워크, 비동기 프레임워크까지 다양한 프레임워크로 개발한다.

1.3 웹브라우저 종류

웹브라우저 종류	웹브라우저 특징
인터넷 익스 플로러	시장점유율이 가장 높고 호환성이 좋으나 비교적 속도가 느리다.
오페라	유럽 브라우저로 구글 크롬이 나오기 전에 가장 빠른 브라우저였다. 스마트폰에 어플을 만들면서 발전 가능성이 보인다.
사파리	애플사의 브라우저로 생동감 있게 보여 지기는 하나 단점은 다른 브라우저에 비해 무겁다.
구글 크롬	가장 최근에 만들어진 브라우저로 속도가 가장 빠르다는 것이 장점이다.

1.4 보조 웹 표준 기술 소개

- 1) HTML5는 큰 의미는 웹 표준 기술을 총칭하고 작은 의미는 HTML5는 웹 문서의 문법을 의미한다.
- 2) CSS3는 HTML 문서를 표현하는 방법을 기술하는 언어이다. 웹 문서에 스타일시트를 적용해야 흔히 접해왔던 네이버 메인 페이지가 된다.
- 3) 자바스크립트 HTML 페이지에서 사용자 반응 등을 처리하는 데 사용하는 프로그래밍 언어이다.

1.5 HTML5 주요 기능 소개

- 1) HTML5에서는 플러그인의 도움 없이 스스로 음악과 동영상을 재생할 수 있다.
- 2) HTML5는 하드웨어의 가속을 받아 2차원 그래픽 3차원 그래픽을 구현 할 수 있다.
- 3) 서버에 데이터를 요청해서 받아만 오는 단방향 방식으로 구현되었지만 HTML5에서는 서버와 실시간으로 쌍방향 통신을 수행할 수 있다.
- 4) HTML5는 디바이스에 접근해 디바이스의 정보와 기능을 사용할 수 있다. 예를 들어 스마트폰의 배터리 잔량과 같은 정보를 가져오거나 스마트폰의 GPS를 사용할 수 있고 진동 벨을 울리기도 한다.
- 5) HTML5는 인터넷이 연결되지 않은 상태에서도 응용 프로그램을 동작시킬 수 있다.
- 6) HTML5 표준에서는 시맨틱 웹을 구현하는 데 필요한 시맨틱 태그를 새로 추가하였다. 시맨틱 웹이란 검색 엔진 같은 프로그램이 정보의 의미를 분석하고 자료를 검색 및 처리하여 제공하는 지능형 웹을 말한다.
- 7) HTML5는 CSS3 스타일시트를 완벽하게 지원한다. CSS3 스타일시트를 사용하면 3차원 변환은 물론 애니메이션 효과를 적용할 수 있다.
- 8) HTML5의 추가 기능을 사용해 기존에 이용하던 웹의 성능을 극대화할 수 있다.

1.6 실습환경 구축하기

1) 웹브라우저 설치하기

구글 크롬의 홈페이지(<http://www.google.com/chrome>)에서 [Chrome 다운로드]버튼을 클릭하면 설치가 진행된다.

2) Visual Studio Code 설치 후 html 결과를 Browser로 바로 실행하기

- ctrl + shift + x로 마켓플레이스를 실행한 후 open in browser 확장프로그램 설치한다.
- alt+b: 디폴트 설정되어 있는 웹브라우저에서 html 파일이 열린다.
- alt+shift+b: 여러 웹브라우저 중에서 선택된 웹브라우저에서 html 파일 열린다.
- HTML 코드 자동완성: !키를 누른 후 Tab키를 누른다.

2장 HTML5 기본 구조와 작성법

2.1 태그와 요소

- 1) 요소는 제목, 본문, 이미지 등 HTML 페이지에 존재하는 것이고 태그는 이러한 요소를 만들 때 사용하는 기호이다.
- 2) `<h1>Hello HTML5</h1>` 시작 태그와 끝 태그를 별도로 입력한다.
- 2) `

` 시작 태그와 끝 태그를 함께 하는 요소도 존재한다.

```
<article>
  <h1>Article Header</h1>
  <p>안녕하세요!</p>
</article>
```

- 3) 아래와 같이 일부 태그는 태그 내부에 다른 태그를 넣을 수도 있다.

2.2 속성

`<h1 title="header">Hello HTML5</h1>` title은 속성이름이고 header는 속성 값이 된다.

`` src는 속성 이름이고 image.png는 속성 값이 된다.

2.3 주석

주석은 프로그램 실행에 영향을 미치지 않고 설명을 위한 목적으로 사용하는 코드를 말한다.

```
ex0200.html

<html>
  <head>
    <title>TITLE</title>
  </head>
  <body>
    <!-- h1 태그 -->
    <h1>Hello HTML5</h1>
  </body>
</html>
```

2.4 HTML5 페이지 기본 구조

```
ex0201.html

<!DOCTYPE html>
<html>
  <head>
    <title>HTML5 Basic Page</title>
  </head>
  <body>
    ...
  </body>
</html>
```

- !DOCTYPE html 태그는 웹 브라우저에 HTML5문서임을 알려준다.
- html 태그는 모든 HTML 페이지의 루트 요소로 모든 HTML 태그는 html 태그의 내부에 작성한다.
- body 태그는 사용자에게 실제로 보여 지는 부분이며
- head 태그는 body 태그에서 필요한 스타일시트와 자바스크립트를 제공하는데 사용한다. head 태그 내부에는 아래에 해당하는 태그만 입력할 수 있다 그 밖에 태그를 넣으면 웹 브라우저가 자동으로 해당 태그를 body 태그로 옮긴다.

태그 이름	설명
meta	웹 페이지에 추가 정보 전달
title	웹 페이지의 제목 지정
script	웹 페이지에 스크립트 추가
link	웹 페이지에 다른 파일 추가
style	웹 페이지에 스타일시트 추가
base	웹 페이지의 기본 경로 지정

2.5 CSS3의 종류와 기본 작성법

웹 페이지는 스타일시트를 사용해 스타일을 적용한다. 스타일이 적용되지 않을 경우 [크롬]-[설정]-[보안 및 개인정보 보호]-[인터넷 사용 기록 삭제]에서 쿠키와 캐시 등을 삭제한다. HTML 페이지에는 아래와 같이 두 가지 방법으로 스타일 시트를 적용할 수 있다.

- 쿠키: 내가 방문하는 사이트에서 만드는 파일로 인터넷 사용 기록을 저장하여 인터넷을 더 쉽게 사용할 수 있도록 해 준다.
- 캐시: 다음 방문 시 페이지를 빠르게 열 수 있도록 이미지 등 페이지의 일부를 기억한다.

1) 스타일시트를 가장 손쉽게 사용할 수 있는 방법으로 style 태그를 사용해 HTML 페이지 내부에 스타일시트를 직접 입력한다. 내부 스타일시트를 사용하는 방법은 아래와 같이 굉장히 간단하다.

ex0202.html

```
<head>
  <title>CSS Basic</title>
  <style>
    h1 {
      color: white;
      background: black;
    }
  </style>
</head>
<body>
  <h1>Hello World..!</h1>
</body>
```

2) 스타일시트가 짧다면 HTML 페이지 내부에 입력해도 괜찮다. 하지만 여러 사람이 함께 협업해야 하고 프로젝트의 규모가 커지면 HTML 페이지 내부에 스타일시트를 삽입하는 것보다 외부 스타일시트를 사용하는 것이 좋다. 외부 스타일시트는 스타일시트를 별도로 생성하고 link 태그의 href 속성을 이용해 HTML 페이지로 불러온다.

style.css

```
h1 {
  color: white;
  background:black;
}
```

ex0203.html

```
<head>
  <title>External CSS Basic</title>
  <link rel="stylesheet" href="style.css"/>
</head>
<body>
  <h1>Hello World..!</h1>
</body>
```

2.6 자바스크립트의 종류와 기본 작성법

웹 페이지에서 사용자 반응 등을 처리하기 위하여 자바스크립트를 사용한다. 웹 페이지에서 자바스크립트를 사용하는 방법은 앞서 살펴본 스타일시트 처럼 두 가지 방법으로 웹 페이지에 자바스크립트를 추가할 수 있다.

1) 내부 자바스크립트는 HTML 페이지 내부에 자바스크립트 코드를 곧바로 입력한다.

ex0204.html

```
<head>
  <title>JavaScript Basic</title>
  <script>
    alert("Hello JavaScript ..!");
  </script>
</head>
<body>
  ...
</body>
```

2) 실제로 자바스크립트 코드를 작성하다보면 1,000행이 넘어가는 경우가 많다. 이러한 코드를 한 파일에 적는다면 파일이 굉장히 복잡해지기 때문에 자바스크립트 파일을 분리해서 사용한다. 외부 자바스크립트 파일은 script 태그와 src 속성에 파일의 경로를 입력하여 HTML 페이지로 불러온다. 외부 스타일시트를 작성할 때처럼 새 파일을 생성하고 아래와 같이 작성한다.

```
script.js
```

```
alert("Hello JavaScript ..!");
```

```
ex0205.html
```

```
<html>
<head>
  <title>JavaScript Basic</title>
  <script src="script.js"> </script>
</head>
<body>
  ...
</body>
</html>
```

HTML5 전에는 아래와 같이 script 태그의 type 속성을 입력하여 외부 자바스크립트 파일을 추가 했다. 하지만 이제는 type 속성을 사용하지 않아도 자바스크립트를 기본으로 인식한다.

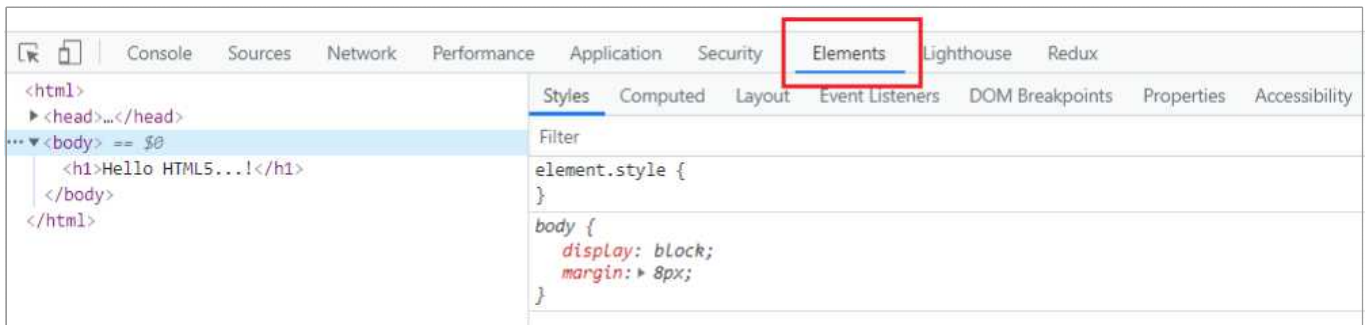
```
<head>
  <title>Outer JavaScript</title>
  <script type="text/javascript" src="script.js"></script>
</head>
```

- 외부 스타일시트를 추가할 때는 link 태그를 사용하고, 외부 자바스크립트 파일을 추가할 때는 script 태그를 사용한다는 것을 꼭 기억하자.

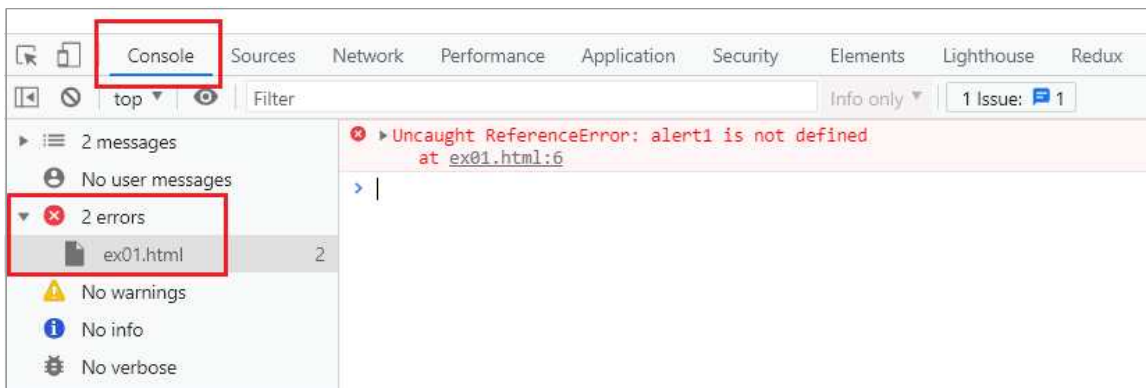
2.7 요소 검사를 이용한 에러 확인

프로그램이 원하지 않는 방향으로 동작하는 것을 일반적으로 버그라고 부른다. 그리고 이러한 버그를 잡는(수정하는) 행위를 디버그라고 한다. 과거의 HTML 페이지는 디버그를 하기 굉장히 어려웠다. 하지만 지금은 웹 브라우저에 요소 검사 기능이 추가되어 쉽게 디버그를 수행할 수 있다. 크롬을 열고 F12를 누르거나 웹페이지에서 마우스 오른쪽 버튼을 클릭하여 [요소 검사]메뉴를 선택한다.

- [Elements] 탭을 사용하면 현재 HTML 페이지의 계층 구조를 알 수 있다. 또한 각각의 태그에 적용된 스타일도 한눈에 파악할 수 있다.



[Console] 탭은 에러를 확인하거나 자바스크립트 코드를 추가로 입력할 경우 사용한다. 만약 자바스크립트 코드가 정상적으로 동작하지 않았다면 무엇이 문제인지 요소 검사를 통하여 직접 확인해 본다.



3.3 본문 태그

문서에서 제목 다음에 나오는 본문은 p 태그를 사용하여 입력한다. p 태그는 Paragraph를 의미하며 문단 하나를 생성한다. 웹 페이지 디자인 관련된 글을 검색하다보면 Lorem Ipsum으로 시작하는 글을 많이 볼 수 있다. 그럴듯한 글자를 아래 사이트에서 쉽게 생성할 수 있어 웹페이지를 디자인할 때 많이 사용한다.

<http://lipsum.com>

다음은 앞에서 다룬 제목 글자 태그와 본문 글자 태그를 Lorem Ipsum 사이트의 글을 이용한 예제이다.

```
<body>
  <h1>제목 글자</h1>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting ... random text.</p>
  <P>It has roots in a piece of classical Latin ... comes from a line in section 1.10.32.</P>
</body>
```

제목 글자

Lorem Ipsum is simply dummy text of the printing and typesetting industry. ... typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it ... The first line of Lorem Ipsum, "Lorem ipsum dolor sit amet..", comes from a line in section 1.10.32.

p 태그 외에도 HTML5는 아래와 같이 본문 태그를 제공한다. br 태그는 다른 글자 태그 내부에 삽입할 수 있지만 hr 태그는 안 된다. 물론 hr 태그가 다른 글자 태그 내부에 들어가도 웹 브라우저가 유연하게 대처하므로 정상적으로 출력된다. 하지만 이는 웹 표준에 어긋난다.

태그 이름	설명
br	개행 태그
hr	수평 줄 태그

아래는 앞에서 익힌 본문 태그를 모두 활용하여 간단한 문서를 작성한 예제이다.

ex0303.html

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML5 + CSS3 Text</title>
</head>
<body>
  <h1>홍차</h1>
  <hr/>
  <h2>정의</h2>
  <p>
    홍차는 백차, 녹차, 우롱차보다 더 많이 발효된 차의 일종이다. 동양에서는 차물의 빛이 붉기 때문에 홍차라고 부르지만,
    서양에서는 차물의 색깔 때문에 'black tea'라고 부른다.</p>
  <br/>
  <h2>등급</h2>
  <p>홍차는 여러 가지로 등급이 매겨진다. 일반적으로 차물의 모양에 따른 등급과 가공 상태에 따른 등급을 조합하여 표시한다.</p>
  <p>- 브로큰 페코</p>
  <p>- 브로큰 페코 수송</p>
  <p>- 브로큰 오렌지 페코 패닝</p>
</body>
</html>
```

3.4 하이퍼링크 태그

HTML에서 가장 중요한 단어는 H를 의미하는 하이퍼텍스트이다. 하이퍼텍스트는 사용자의 선택에 따라 관련된 특정 정보로 이동할 수 있도록 조작된 문서를 의미한다. HTML 페이지가 조직화된 문서 형태가 될 수 있는 이유는 a 태그 덕분이다.

a 태그로만은 어떤 페이지로 이동할지 웹 브라우저에 알려줄 수 없고 아래처럼 href 속성을 사용해 알려준다. Hyper Reference를 의미한다.

```
<a href="http://hanb.co.kr">한빛미디어</a>
```

a 태그의 href 속성에는 다음에 해당하는 내용을 입력할 수 있다.

하이퍼링크 태그 경로	설명
절대경로	현재 웹 페이지의 위치와 상관없이 웹페이지 또는 파일의 위치를 나타내는 경로 (예: /animal.jpg)
상대경로	현재 웹 페이지의 위치에 따라 결정되는 웹페이지 또는 파일의 경로 (예: ../animal.jpg)
아이디 경로	현재 웹 페이지 내부에 있는 다른 태그의 id 속성 (예: #name)
메일 경로	메일 주소 (예: mailto:hanb@hanb.co.kr)

다음은 a 태그를 사용해 특정 웹 페이지나 웹 페이지 내부의 특정 위치로 이동하는 링크를 생성한 예제이다. id 속성이 중복되면 먼저 나오는 태그로 이동한다. 하지만 id 속성을 중복해서 사용하는 것은 웹 표준에 어긋나는 행위이다.

1) 특정 웹 페이지에 연결하기 : 아래와 같이 코드를 작성하고 실행한 후, 하이퍼링크를 설정한 글자를 클릭하면 해당 페이지로 이동한다.

ex0304.html

```
<html>
<head>
  <title>HTML TEXT Basic Page</title>
</head>
<body>
  <a href="http://naver.com/">네이버</a><br/>
  <a href="http://daum.com/">다음</a><br/>
  <a href="http://daum.com/">다음</a><br/>
</body>
</html>
```

2) 웹 페이지 내부에 연결하기: 페이지 내부의 특정 태그로 이동하려면 id 속성을 사용한다. 우선 아래와 같이 h1 태그에 id 속성을 입력한다. 이어서 a 태그의 href 속성에 '#id속성' 형태의 문자열을 입력한다. 파일을 실행해서 [Alpha 부분] 링크를 클릭하면 해당 내용으로 이동한다.

ex0304.html

```
<html>
<head>
  <title>HTML TEXT Basic Page</title>
</head>
<body>
  <a href="#alpha">Alpha 부분</a>
  <a href="#beta">Beta 부분</a>
  <a href="#gamma">Gamma 부분</a>
  <hr/>
  <h1 id="alpha">Alpha</h1>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting ... random text.</p>
  <h1 id="beta">Beta</h1>
  <p>It has roots in a piece of classical Latin ... comes from a line in section 1.10.32.</p>
  <h1 id="gamma">Gamma</h1>
  <p>It is a long established fact that a reader will be distracted by ... when looking at its layout.</p>
</body>
</html>
```


3.5 글자 모양 태그

HTML5는 아래와 같은 글자 모양 태그를 사용하여 웹 페이지의 글자에 형태와 의미를 부여한다. 다만 HTML4 버전까지 존재했던 `big` 태그가 없어졌다. `small` 태그가 존재하기 때문에 `big` 태그가 있다고 생각할 수 있으므로 주의해야 한다.

태그 이름	설명
<code>b</code>	굵은 글자 태그
<code>i</code>	기울어진 글자 태그
<code>small</code>	작은 글자 태그
<code>sub</code>	아래 첨자 태그
<code>sup</code>	위 첨자 태그
<code>ins</code>	밑줄 글자 태그
<code>del</code>	가운데 줄이 그려진 글자(취소선) 태그

글자 모양 태그는 모두 단독으로 사용할 수 있다. 제목 글자 태그와 본문 글자 태그 내부에 입력하거나 글자 모양 태그 내부에 다른 글자 모양 태그를 넣을 수도 있다. 하지만 글자 모양 태그 내부에 제목 글자 태그와 본문 글자 태그는 넣을 수 없다.

ex0305.html

```
<html>
<body>
  <h1><b>다음은 글자 모양 태그를 활용하는 예제입니다.</b></h1>
  <h1><i>다음은 글자 모양 태그를 활용하는 예제입니다.</i></h1>
  <h1><small>다음은 글자 모양 태그를 활용하는 예제입니다.</small></h1>
  <h1>다음은 글자 모양 태그를 활용하는 <sub>예제</sub>입니다.</h1>
  <h1>다음은 글자 모양 태그를 활용하는 <sup>예제</sup>입니다.</h1>
  <h1><ins>다음은 글자 모양 태그를 활용하는 예제입니다.</ins></h1>
  <h1><del>다음은 글자 모양 태그를 활용하는 예제입니다.</del></h1>
  <hr>
  <b>다음은 글자 모양 태그를 활용하는 예제입니다.</b><br>
  <i>다음은 글자 모양 태그를 활용하는 예제입니다.</i><br>
  <small>다음은 글자 모양 태그를 활용하는 예제입니다.</small><br>
  <sub>다음은 글자 모양 태그를 활용하는 <sub>예제</sub>입니다.<br>
  <sup>다음은 글자 모양 태그를 활용하는 <sup>예제</sup>입니다.<br>
  <ins>다음은 글자 모양 태그를 활용하는 예제입니다.</ins><br>
  <del>다음은 글자 모양 태그를 활용하는 예제입니다.</del><br>
</body>
</html>
```

3.6 목록 태그

현대 웹 페이지에서 빠지지 않고 등장하는 요소가 있다면 바로 내비게이션 목록이다. 내비게이션 목록은 웹 사이트의 다른 웹 페이지로 이동할 수 있는 버튼을 모아둔 것이다. 목록 태그는 내비게이션 메뉴를 만들 때 주로 사용한다.

1) `ul` 태그는 unordered list를 의미하고 `li` 태그는 list item을 의미한다.

태그 이름	설명
<code>ul</code>	글머리 기호 목록 태그
<code>li</code>	목록 요소

2) 순서가 있는 목록을 생성할 때는 `ol` 태그를 사용한다. `ol` 태그는 ordered list를 의미한다.

태그 이름	설명
<code>ol</code>	순서 번호 목록 태그
<code>li</code>	목록 요소

```

<html>
<head><title>HTML List Basic Page</title></head>
<body>
  <!--글머리 기호 목록 만들기-->
  <ul>
    <li>사과</li>
    <li>바나나</li>
    <li>오렌지</li>
  </ul>
  <!--순서 번호 목록 만들기-->
  <ol>
    <li>사과</li>
    <li>바나나</li>
    <li>오렌지</li>
  </ol>
</body>
</html>

```

3.7 테이블 태그

표를 만들 때는 테이블 태그를 사용한다. 아래는 다양한 테이블 태그의 종류이다. 간단히 작성할 경우 `thead`, `tbody`, `tfoot`는 생략 가능하다.

태그 이름	설명
<code>table</code>	표 (table)
<code>thead</code>	표의 머리 (table head)
<code>tbody</code>	표의 본문 (table body)
<code>tfoot</code>	표의 꼬리 (table foot)
<code>tr</code>	표의 행 (table row)
<code>th</code>	표의 제목 요소 (table header)
<code>td</code>	표의 일반 요소 (table data)

```

<html>
<head>
  <title>HTML Table Basic Page</title>
</head>
<body>
  <table border="1">
    <thead>
      <tr>
        <th></th><th>월</th><th>화</th><th>수</th><th>목</th><th>금</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>1교시</td><td>영어</td><td>국어</td><td>과학</td><td>미술</td><td>기술</td>
      </tr>
      <tr>
        <td>2교시</td><td>도덕</td><td>체육</td><td>영어</td><td>수학</td><td>사회</td>
      </tr>
    </tbody>
  </table>
</body>
</html>

```

th와 td 태그에는 아래 표의 속성을 입력할 수 있다. colspan과 rowspan 속성을 사용하면 th와 td 태그가 표에서 차지하는 영역이 늘어난다. 일종의 셀 병합이다. colspan 속성을 사용하면 영역이 가로 방향으로 늘어나고 rowspan 속성을 사용하면 세로 방향으로 늘어난다.

속성 이름	설명
colspan	셀의 너비 지정
rowspan	셀의 높이 지정

다음은 colspan과 rowspan 속성을 적용한 예제이다.

ex0308.html

```
<body>
  <table border="1">
    <tr><th colspan="2">지역별 홍차</th></tr>
    <tr>
      <th rowspan="3">중국</th><td>정산소종</td>
    </tr>
    <tr><td>기문</td></tr>
    <tr><td>운남</td></tr>
    <tr>
      <th rowspan="4">인도 및 스리랑카</th><td>아삼</td>
    </tr>
    <tr><td>실론</td></tr>
    <tr><td>다질링</td></tr>
    <tr><td>날기리</td></tr>
  </table>
</body>
```

3.8 이미지 태그

이미지를 삽입할 때는 img 태그를 사용한다. 이미지 표시에 필요한 정보(경로, 크기 등)는 img 태그의 속성에 입력한다. img 태그에서는 속성이 중요하지만 요즘은 이미지 크기도 대개 스타일시트로 설정하기 때문에 width와 height 속성도 잘 사용하지 않는다.

속성 이름	설명
src	이미지 경로 지정
alt	이미지가 없을 때 나오는 글자 지정
width	이미지의 너비 지정
height	이미지의 높이 지정

표시하고 싶은 이미지나 준비 파일을 HTML 페이지와 같은 폴더에 저장해 두고 아래와 같은 코드를 작성한다.

ex0309.html

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

위 예제에서는 이미지 파일을 미리 준비해서 넣어두었지만 img 태그의 src 속성에는 다음과 같이 웹에 존재하는 이미지의 경로를 넣어도 된다.

```
img src="http://www.hanb.co.kr/img.gnb/logo_hanbimedia.gif"
```

웹 페이지를 개발하다 보면 이미지를 만들지 않고 공간을 확보해야하는 경우가 있다. 이런 경우에는 아래 주소로 이미지의 공간을 만들 수 있다.

```
img src="http://placeholder.it/너비x높이" 또는 "http://placekitten.com/너비/높이"
```

3.9 공간 분할 태그

HTML5의 공간 분할 태그는 div와 span이다 div 태그는 공간을 블록(block) 형식으로 분할하고 span 태그는 공간을 인라인(inline) 형식으로 분할한다. 블록 형식과 인라인 형식은 레이아웃 구성에 가장 중요한 요소이다.

태그 이름	설명
div	블록 형식으로 공간 분할
span	인라인 형식으로 공간 분할

1) 블록 형식으로 공간 분할하기: 아래 코드는 div 태그를 사용해 공간을 분할한다. 실행 결과를 보면 각 태그가 한 행을 모두 차지한다.

ex0310.html

```
<body>
  <div>div 태그 - block 형식</div>
  <div>div 태그 - block 형식</div>
  <div>div 태그 - block 형식</div>
</body>
```

2) 인라인 형식으로 공간 분할하기: div 태그를 span 태그로 바꿔보았다. 코드를 실행하면 각 태그가 자신의 글자 크기만큼 영역을 차지한다.

ex0310.html

```
<body>
  <span>span 태그 - inline 형식</span>
  <span>span 태그 - inline 형식</span>
  <span>span 태그 - inline 형식</span>
</body>
```

이러한 블록 형식과 인라인 형식은 일반 태그에도 적용된다.

```
<body>
  <h1>h1 태그</h1>
  <p>p 태그 </p>
  <hr/>
  <a href="#">a 태그
  <i>i 태그</i>
</body>
```

코드를 실행하면 아래처럼 출력한다. h1과 p 태그는 블록 형식이고, a와 i 태그는 인라인 형식이다.

h1 태그

p 태그

[a 태그 / 태그](#)

각 형식에 해당하는 태그를 간단하게 아래처럼 정리할 수 있다.

블록 형식 태그	인라인 형식 태그
div 태그	span 태그
h1 ~ h6 태그	a 태그
p 태그	input 태그
목록 태그	글자 형식 태그
테이블 태그	img 태그
입력 양식 태그	

3.10 시맨틱 태그

태그에 의미를 부여하는 태그이며 시맨틱 태그를 사용해 페이지를 구성하면 각 태그에 의미를 부여할 수 있기 때문에 코드 가독성이 좋다. HTML5는 주로 아래 표와 같은 시맨틱 태그를 사용해 시맨틱 웹을 구현한다.

태그 이름	설명
header	헤더
nav	내비게이션
aside	사이드에 위치하는 공간
section	여러 중심 내용을 감싸는 공간
article	글자가 많이 들어가는 부분
footer	푸터

ex0311.html

```
<body>
  <header><h1>HTML5 기본</h1></header>
  <nav>
    <ul>
      <li><a href="#">메뉴 - 1</a></li>
      <li><a href="#">메뉴 - 2</a></li>
      <li><a href="#">메뉴 - 3</a></li>
    </ul>
  </nav>
  <section>
    <article>
      <h1>What is Lorem Ipsum?</h1>
      <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. but also ...</p>
    </article>
    <article>
      <h1>Where does it come from?</h1>
      <p>Contrary to popular belief, making it over 2000 years old, Richard... Hampden-Sydney College</p>
    </article>
  </section>
  <footer>
    <address>서울특별시 강서구 내발산동</address>
  </footer>
</body>
```

웹 브라우저에서 특정 시맨틱 태그 아래에 있는 글자 크기를 변경하므로 h1 태그의 글자 크기가 조금 작아지는 차이가 미세하게 발생한다.

HTML5 기본

- [메뉴 - 1](#)
- [메뉴 - 2](#)
- [메뉴 - 3](#)

What is Lorem Ipsum?

Lorem Ipsum is simply dummy text of the printing and typesetting industry. but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

Where does it come from?

Contrary to popular belief, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source.

서울특별시 강서구 내발산동

4장 HTML5 멀티미디어 태그와 입력 양식 태그

4.1 오디오 기본 태그

오디오를 삽입할 때는 <audio>태그를 사용한다. 태그와 달리 시작 태그와 끝 태그를 모두 사용한다. 오디오 실행에 필요한 추가 정보는 아래 표의 속성을 사용해 입력한다.

속성 이름	설명
src	음악 파일의 경로 지정
preload	음악을 재생하기 전에 데이터를 모두 불러올지 여부 지정
autoplay	음악의 자동 재생 여부 지정
loop	음악의 반복 여부 지정
controls	음악 재생 도구 출력 여부 지정

ex0401.html

```
<body>
  <audio controls="controls">
    <source src="Kalimba.mp3" type="audio/mp3"/>
    <source src="Kalimba.ogg" type="audio/ogg"/>
  </audio>
</body>
```

- XHTML5 표기 방법으로 autoplay, preload, loop 속성을 사용한 경우

```
<audio src="Kalimba.mp3" controls="controls" autoplay="autoplay"></audio>
```

- HTML5 표기 방식을 사용한 경우

```
<audio src="Kalimba.mp3" controls autoplay></audio>
```

4.2 비디오 기본 태그

HTML5 시대가 오면서 웹 브라우저만으로 동영상을 볼 수 있게 되었다. 웹 페이지에 비디오를 삽입할 때는 video 태그를 사용한다. audio 태그와 사용 방법이 같다. 비디오 실행에 필요한 추가 정보는 video 태그에 아래 표의 속성을 사용해 입력한다.

속성 이름	설명
src	비디오 파일의 경로 지정
poster	비디오 준비 중일 때 표시할 이미지의 파일 경로 지정
preload	비디오를 재생하기 전에 데이터를 모두 불러올지 여부 지정
autoplay	비디오의 자동 재생 여부 지정
loop	비디오의 반복 재생 여부 지정
controls	비디오 재생 도구의 출력 여부 지정
width	비디오의 너비 지정
height	비디오의 높이 지정

ex0402.html

```
<body>
  <video width="640" controls="controls" poster="http://placeholder.it/640x360">
    <source src="Wildlife.mp4" type="video/mp4" />
    <source src="Wildlife.webm" type="video/webm" />
  </video>
```

4.3 Video.js 플러그인

video 태그를 사용하면 웹브라우저마다 표시되는 video 태그의 형태(화면의 크기, 컨트롤 모양 등)가 일관되지 않으며 웹 페이지를 디자인할 때 문제가 될 수 있으나 Video.js 플러그인을 사용하면 이러한 문제를 간단하게 해결할 수 있다.

우선 head 태그에 link 태그와 script 태그를 입력한다. 이는 다른 개발자가 만든 외부 스타일시트와 자바스크립트를 사용하는 방법이다. 이어서 video 태그를 생성하고 class 속성과 data-setup 속성을 입력하면 끝이다. 스타일 시트와 자바 스크립트 경로는 video.js 플러그인의 홈페이지에서 <Embed This Player>를 클릭하면 복사할 수 있다.

ex0403.html

```
<html>
<head>
  <title>Video.js Basic</title>
  <link href="http://vjs.zencdn.net/c/video-js.css" rel="stylesheet">
  <script src="http://vjs.zencdn.net/c/video.js"></script>
</head>
<body>
  <video controls="controls" width="640" height="360" class="video-js vjs-default-skin" data-setup="{}">
    <source src="Wildlife.mp4" type="video/mp4" />
    <source src="Wildlife.webm" type="video/webm" />
  </video>
</body>
</html>
```

4.4 입력 양식 태그

입력 양식은 사용자에게 정보를 입력받는 공간을 말하며 입력 양식 태그를 사용해 만든다. 일반적으로 서버에서 클라이언트로 데이터가 전달된다. 하지만 회원가입과 로그인, 검색과 관련된 버튼을 클릭하는 순간에는 클라이언트가 서버로 데이터를 보내야 한다. 입력 양식을 사용하는 것은 클라이언트가 서버로 데이터를 보내는 가장 기본적인 방법이다.

입력 양식은 form 태그를 사용해 생성한다. 입력 양식에 데이터를 입력하고 [submit] 쿼리전송 버튼을 클릭하면 지정된 방법으로 지정된 장소에 데이터가 전달된다. 장소와 방법은 form 태그에 아래 표의 속성을 사용해 지정한다.

속성 이름	설명
action	전송 위치(데이터를 전달할 목적지)
method	전송 방식

action 속성은 주소이고 method 속성은 일반 편지와 택배 등의 배송 방식을 의미한다. 웹에서 데이터를 전송할 때 가장 많이 사용되는 전송 방식은 아래 표와 같다.

전송 방식	설명
GET	주소에 데이터를 직접 입력해 전달한다.
POST	별도의 방법을 사용해 데이터를 해당 주소로 전달한다.

데이터를 입력하고 [submit] 전송 버튼을 클릭한 후 웹 브라우저에 표시되는 URL을 살펴보면 차이점을 확인 할 수 있다. GET 방식으로 데이터를 전달하면 주소창에 URL이 '파일이름?search=데이터값' 형태로 변경되었을 것이다. GET 방식은 이렇게 주소에 데이터를 입력해서 보낸다.

반면에 POST 방식은 주소가 변경되지 않는다. GET 방식과는 달리 비밀스럽게 데이터를 전달한다. GET 방식은 주소에 데이터를 입력하므로 크기가 한정되어 있다. 하지만 POST 방식은 별도로 택배를 붙여 데이터를 전송하기 때문에 데이터 용량에 제한이 없다.

GET 전송 방식	POST 전송 방식
<pre><body> <form method="get"> <input type="text" name="search"/> <input type="submit"/> </form> </body></pre>	<pre><body> <form method="post"> <input type="text" name="search"/> <input type="submit"/> </form> </body></pre>

- 기본 입력 양식 태그 출력

ex0404.html

```
<body>
  <form>
    <!-- 사용자가 입력하는 입력 양식 -->
    <input type="text" name="text" value="text"/><br/>
    <input type="password" name="password" value="password"/><br/>
    <input type="file" name="file" value="file"/><br/>
    <input type="checkbox" name="checkbox" value="checkbox"/><br/>
    <input type="radio" name="radio" value="radio"/><br/>
    <!-- 보이지 않는 입력 양식 -->
    <input type="hidden" name="hidden" value="hidden"/><br/>

    <!-- 버튼 -->
    <input type="button" value="button"/><br/>
    <input type="reset" value="reset"/><br/>
    <input type="submit" value="submit"/><br/>
    <input type="image" src="http://placeholder.it/100x100"/>

    <!-- Textarea -->
    <textarea>Textarea 태그 Textarea 태그</textarea>
  </form>
</body>
```

라디오 버튼의 name 속성을 같게 줘서 대상 중 하나만 선택하도록 하는 형태를 손쉽게 구현할 수 있다. 코드를 실행하고 두 개의 라디오 버튼을 번갈아 가며 클릭해 보자. 한 번에 하나만 선택된다. 반면에 체크박스는 name 속성을 다르게 줘서 여러 개를 동시에 체크할 수 있다.

ex0405.html

```
<body>
  <form>
    <table>
      <tr>
        <!-- label 태그의 for 속성에 input 태그의 id 속성을 입력하여 label 태그가 어떤 input 태그를 설명하는지 알려준다. -->
        <td><label for="username">이름</label></td>
        <td><input id="username" type="text" name="username" placeholder="이름"></td>
      </tr>
      <tr>
        <td>성별</td>
        <td>
          <input id="man" type="radio" name="gender" value="m" checked/>
          <label for="man">남자</label>
          <input id="woman" type="radio" name="gender" value="w"/>
          <label for="woman">여자</label>
        </td>
      </tr>
      <tr>
        <td>취미</td>
        <td>
          <input id="movie" type="checkbox" name="movie" value="movie"/>
          <label for="movie">영화감상</label>
          <input id="game" type="checkbox" name="game" value="game"/>
          <label for="game">컴퓨터게임</label>
        </td>
      </tr>
    </table>
    <input type="submit" value="가입"/>
  </form>
</body>
```


- select 태그 익히기

1) 한 항목만 선택하게 하기: select 태그는 목록으로 보여주는 항목 중 하나 또는 여러 개를 선택할 수 있는 입력 양식 요소이다. 기본적으로 하나만 선택할 수 있다. select 태그를 사용할 때는 optgroup, option 태그를 함께 사용한다.

ex0406.html

```
<body>
  <select>
    <option>김밥</option>
    <option>떡볶이</option>
    <option>순대</option>
    <option>어묵</option>
  </select>
</body>
```

2) 여러 개의 항목 선택하게 하기: 여러 개의 항목을 선택하게 하려면 select 태그의 multiple 속성을 사용한다. 데스크톱에서는 아름답게 나오지 않으므로 잘 사용하지 않는다.

ex0407.html

```
<body>
  <select multiple="multiple">
    <option>김밥</option>
    <option>떡볶이</option>
    <option>순대</option>
    <option>어묵</option>
  </select>
</body>
```

3) 선택 옵션 묶기: 선택 옵션은 다음 코드와 같이 optgroup 태그를 사용해 그룹으로 묶을 수 있다.

ex0408.html

```
<body>
  <select>
    <optgroup label="HTML5">
      <option>Multimedia Tag</option>
      <option>Connectivity</option>
      <option>Device Access</option>
    </optgroup>
    <optgroup label="CSS3">
      <option>Animation</option>
      <option>3D Transform</option>
    </optgroup>
  </select>
</body>
```

4) fieldset과 legend 태그를 사용하면 입력 양식을 그룹으로 묶고 이름을 지정할 수 있다. legend 태그는 다음과 같이 fieldset 태그 안에 사용한다. 물론 다른 곳에도 쓸 수 있지만 아무 효과가 없다.

ex0409.html

```
<body>
  <form>
    <fieldset>
      <legend>입력 양식</legend>
      <table>
        <tr><td><label for="name">이름</label></td><td><input id="name" type="text"/></td></tr>
        <tr><td><label for="mail">이메일</label></td><td><input id="mail" type="email"/></td></tr>
      </table>
      <input type="submit"/>
    </fieldset>
  </form>
</body>
```

5장 CSS3 선택자

5.1 선택자 개요

선택자를 사용해 특정 HTML 태그를 선택하면 선택한 태그에 우리가 원하는 스타일 또는 스크립트를 적용할 수 있다. 아래는 선택자를 사용해 h1 태그를 선택하고 color 스타일 속성에 red 키워드를 적용한 예이다.

```
h1 { color: red; }
```

위와 같은 코드를 CSS 블록이라고 부르는데 CSS 블록은 다음과 같이 style 태그 내부에 입력한다. 그리고 style 태그 내부에 입력하는 코드들을 스타일시트라고 부른다.

ex0500.html

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Selector Basic</title>
  <style>
    h1 {
      color: red;
      background-color: orange;
    }
  </style>
</head>
<body>
  <h1>CSS 선택자 기본</h1>
</body>
</html>
```

5.2 기본 선택자

기본 선택자는 너무 중요한 내용이므로 반드시 기억해야 한다. 기본 선택자에 해당하는 내용을 정리하면 아래와 같다.

선택자 종류	선택자 형태	설명
전체 선택자	*	HTML 문서 안에 있는 태그를 모두 선택한다.
태그 선택자	태그	HTML 페이지 내부의 특정 태그를 모두 선택한다.
아이디 선택자	#아이디	특정id 속성이 있는 태그를 선택한다. 웹 표준에 id속성은 웹 페이지 내부에서 중복되면 안 된다는 규정이 있으므로 아이디 선택자는 특정 태그 하나를 선택할 때 사용한다.
클래스 선택자	.클래스	특정 클래스가 있는 태그를 선택한다.

1) 전체 선택자: 전체 선택자를 사용해 모든 태그의 color 스타일 속성을 red로 변경 한다

ex0501.html

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Selector Basic Page</title>
  <style>
    * { color: red; }
  </style>
</head>
<body>
  <h1>제목 글자 태그</h1>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
</body>
</html>
```

2) 태그 선택자: 태그 선택자를 사용하여 h1 태그의 color 스타일 속성에 red를 적용하고 p태그의 color 스타일 속성에 blue를 적용한다.

ex0502.html

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Selector Basic Page</title>
  <style>
    h1 { color: red }
    p { color: blue }
  </style>
</head>
<body>
  <h1>태그 선택자 적용하기</h1>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
  <p>Contrary to popular belief, Lorem Ipsum is not simply random text.</p>
</body>
</html>
```

한꺼번에 여러 개의 선택자를 입력하여 스타일 속성을 적용할 때는 쉼표를 사용한다. 다음은 태그 선택자 여러 개를 쉼표로 연결해 margin 스타일 속성과 padding 스타일 속성을 적용한 코드이다.

```
<style>
  body p, h1, h2, h3, h4, h5, h6 { margin:0; padding:0 }
</style>
```

3) 공간 분할 태그에 아이디 선택자를 사용하여 id 속성을 적용하고 레이아웃을 구성한다.

ex0503.html

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Selector Basic Page</title>
  <style>
    #header {
      width: 800px; margin: 0 auto; background: red; }
    #wrap {
      width: 800px; margin: 0 auto; overflow: hidden; }
    #aside {
      width: 200px; float: left; background: blue; }
    #content {
      width: 600px; float: left; background: green; }
  </style>
</head>
<body>
  <div id="header">
    <h1>#header 태그</h1>
  </div>
  <div id="wrap">
    <div id="aside">
      <h1>#aside 태그</h1>
    </div>
    <div id="content">
      <h1>#content 태그</h1>
    </div>
  </div>
</body>
</html>
```

4) 클래스 선택자는 특정한 클래스를 가지고 있는 태그를 선택할 때 사용한다. 웹 페이지를 개발할 때 가장 많이 이용한다.

ex0504.html

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS3 Selector Basic</title>
  <style>
    .select { color: red }
  </style>
</head>
<body>
  <ul>
    <li class="select">사과</li>
    <li>바나나</li>
    <li class="select">오렌지</li>
    <li>감</li>
  </ul>
</body>
</html>
```

- 여러 개의 클래스 선택자 사용: class 속성은 아래 코드처럼 공백으로 구분하여 클래스를 여러 개 사용할 수 있다. 아래와 같은 경우에는 item 클래스와 header 클래스가 함께 적용된다.

```
<html>
<head>
  <title>CSS3 Selector Basic</title>
  <style>
    .item { color: red; }
    .header { background-color: blue;}
  </style>
</head>
<body>
  <h1 class="item header">동해물과 백두산이</h1>
</body>
</html>
```

- 태그 선택자와 클래스 선택자: 웹 페이지 내부에서 id 속성은 중복되지 않아야 하지만 class 속성은 중복될 수 있다. class 속성이 서로 다른 태그에 적용되는 경우 태그 선택자와 클래스 선택자를 함께 사용해서 태그를 더 정확하게 선택할 수 있다.

```
<html>
<head>
  <title>CSS3 Selector Basic</title>
  <style>
    li.select {color: red;}
  </style>
</head>
<body>
  <h1 class="select">제목 글자</h1>
  <ul>
    <li class="select">사과</li>
    <li>바나나</li>
    <li>오렌지</li>
    <li>감</li>
  </ul>
</body>
</html>
```

5.3 속성 선택자

속성 선택자는 특정 속성과 값을 가지는 태그를 선택할 때 사용한다. input 태그는 이름이 모두 같지만 type 속성에 따라 형태가 달라지므로 input 태그를 선택할 때는 속성 선택자를 많이 사용한다.

선택자 형태	설명
선택자[속성]	특정한 속성이 있는 태그를 선택한다.
선택자[속성 = 값]	특정한 속성 안의 값이 특정 값과 같은 태그를 선택한다.

ex0505.html

```
<head>
  <title>CSS3 Selector Basic Page</title>
  <style>
    input[type="text"] { background: red ;
    input[type="password"] { background: blue ;
  </style>
</head>
<body>
  <form>
    <input type="text" />
    <input type="password" />
  </form>
</body>
```

5.4 후손 선택자와 자손 선택자

후손 선택자와 자손 선택자는 특정 태그의 후손과 자손을 선택할 때 사용한다. 아래 코드는 div 태그를 기준으로 삼는다면 바로 한 단계 아래에 위치한 h1, h2, ul 태그를 자속이라고 부르고 div 태그 아래에 위치한 모든 태그를 후손이라고 부른다.

```
<body>
  <div>
    <h1>CSS3 선택자 기본</h1>
    <h2>기본 선택자</h2>
    <ul>
      <li>전체 선택자</li>
      <li>태그 선택자</li>
      <li>아이디 선택자와 클래스 선택자</li>
    </ul>
  </div>
</body>
```

1) 후손 선택자는 특정한 태그의 후손을 선택할 때 사용한다.

선택자 형태	설명
선택자A 선택자B	선택자A의 후손인 선택자B 선택

```
<style>
  #header h1, h2 { color: red; }
</style>
```

이 선택자는 header 태그의 후손인 h1 태그를 선택하고 일반적인 h2 태그를 선택한다. header 태그의 후손인 h1 태그와 header 태그의 후손인 h2 태그를 선택하고 싶다면 다음과 같이 사용해야 한다.

```
<style>
  #header h1, #headerh2 { color: red; }
</style>
```

아래 코드는 h1 태그를 선택하지만 후손 선택자를 사용해 특정한 대상의 후손이라는 것을 구별한다.

ex0506.html

```
<head>
  <style>
    #header h1 { color: red }
    #section h1 { color: orange }
  </style>
</head>
<body>
  <div id="header">
    <h1 class="title">What is Lorem Ipsum?</h1>
    <div id="nav">
      <h1>Navigation</h1>
    </div>
  </div>
  <div id="section">
    <h1 class="title">What is Lorem Ipsum?</h1>
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
  </div>
</body>
```

2) 자손 선택자는 특정한 태그의 자손을 선택할 때 사용한다.

선택자 형태	설명
선택자A > 선택자B	선택자A의 자손인 선택자B 선택

다음 코드는 앞서 살펴본 예제와 구성이 같다. 다만 #nav 태그 아래에 있는 h1 태그는 후손이지 자손은 아니기 때문에 스타일이 적용되지 않는다.

ex0507.html

```
<head>
  <style>
    #header > h1 { color: red }
    #section > h1 { color: orange }
  </style>
</head>
<body>
  <div id="header">
    <h1 class="title">What is Lorem Ipsum?</h1>
    <div id="nav">
      <h1>Navigation</h1>
    </div>
  </div>
  <div id="section">
    <h1 class="title">What is Lorem Ipsum?</h1>
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
  </div>
</body>
```

5.5 반응 선택자

반응 선택자는 사용자의 반응으로 생성되는 특정한 상태를 선택한다. 사용자가 특정 태그 위에 마우스를 커서를 올리면 hover 상태이고 특정 태그를 마우스로 클릭하면 active 상태이다.

선택자 형태	설명
:active	사용자가 마우스로 클릭한 태그 선택
:hover	사용자가 마우스 커서를 올린 태그 선택

글자 위에 마우스 커서를 올리면 붉은색이 적용되고 클릭하면 파란색이 적용된다.

ex0508.html

```
<head>
  <style>
    hl:hover {
      color:red;
    }
    hl:active {
      color:blue;
    }
  </style>
</head>
<body>
  <hl>반응 선택자</hl>
</body>
```

5.6 상태 선택자

상태 선택자는 입력 양식의 상태를 선택할 때 사용된다.

선택자 형태	설명
:checked	체크 상태의 input 태그 선택
:focus	초점이 맞추어진 input 태그 선택
:enabled	사용 가능한 input 태그 선택
:disabled	사용 불가능한 input 태그 선택

ex0509.html

```
<head>
  <style>
    /*input 태그가 사용 가능할 경우에 background-color 속성에 white 키워드를 적용합니다.*/
    input:enabled { background-color: white }
    /*input 태그가 사용 불가능할 경우에 background-color 속성에 gray 키워드를 적용합니다.*/
    input:disabled { background-color: gray }
    /*input 태그에 초점이 맞추어진 경우에 background-color 속성에 orange 키워드를 적용합니다.*/
    input:focus { background-color: orange }
  </style>
</head>
<body>
  <h2>사용 가능</h2>
  <input />
  <h2>사용 불가능</h2>
  <input disabled="disabled"/>
</body>
```

5.7 구조 선택자

구조 선택자는 특정한 위치에 있는 태그를 선택할 때 사용한다.

선택자 형태	설명
:first-child	형제 관계에서 첫 번째로 등장하는 태그 선택
:last-child	형제 관계에서 마지막으로 등장하는 태그 선택
:nth-child(수열)	형제 관계에서 앞에서 수열 번째로 등장하는 태그 선택
:nth-last-child(수열)	형제 관계에서 뒤에서 수열 번째로 등장하는 태그 선택

아래는 구조 선택자를 사용해 홀수 위치와 짝수 위치를 판별하고 스타일을 적용한다. 첫 번째 태그와 마지막 태그는 둥근 테두리가 적용되고 홀수와 짝수 번째에 각각 다른 색상의 스타일이 적용된다.

ex0511.html

```
<head>
  <style>
    ul {
      overflow: hidden ;
    }
    li {
      list-style: none; float: left; padding: 15px; }
    li:first-child {
      border-radius: 10px 0 0 10px ;
    }
    li:last-child {
      border-radius: 0 10px 10px 0 ;
    }
    li:nth-child(2n) {
      background-color: #FF0003 ;
    }
    li:nth-child(2n+1) {
      background-color: #800000 ;
    }
  </style>
</head>
<body>
  <ul>
    <li>1 번째</li>
    <li>2 번째</li>
    <li>3 번째</li>
    <li>4 번째</li>
    <li>5 번째</li>
    <li>6 번째</li>
    <li>7 번째</li>
  </ul>
</body>
```

구조 선택자를 사용할 때 자주 실수하는 부분을 간단히 살펴보겠다. 우선 다음 코드를 읽고 실행 결과를 예측해 보자

ex0512.html

```
<head>
  <title>CSS3 Selector Basic</title>
  <style>
    li > a:first-child { color: red; }
  </style>
</head>
<body>
  <ul>
    <li><a href="#">주의 사항</a></li>
    <li><a href="#">주의 사항</a></li>
    <li><a href="#">주의 사항</a></li>
    <li><a href="#">주의 사항</a></li>
    <li><a href="#">주의 사항</a></li>
  </ul>
</body>
```

일부는 첫 번째 a 태그에 붉은 색이 적용될 것으로 예측했을 것이다. 하지만 코드를 실행하면 모든 a 태그에 붉은색이 적용된다. li 태그의 첫 번째 자손에 해당하는 a 태그를 선택했으므로 다섯 개의 a 태그 전체가 이 조건에 해당된다. 따라서 모든 글자가 붉게 표시된다. 첫 번째의 a 태그에만 붉은색을 적용하려면 아래와 같이 입력해야 한다.

```
<style>
  li:first-child > a { color: red; }
</style>
```


6장 CSS3 속성

6.1 키워드 단위

키워드는 W3C에서 미리 정의한 단어이다. 키워드를 스타일 값으로 입력하면 키워드에 해당하는 스타일이 자동으로 적용된다.

```
body {  
    display: block; /*block이 display의 키워드이다 */  
}
```

6.2 크기 단위

크기는 CSS에서 가장 많이 사용하는 단위며 종류에는 %, em, cm, mm, inch, px가 있다. 이 중에서도 아래의 단위를 많이 사용한다.

단위	설명
%	백분율 단위
em	배수 단위
px	픽셀

크기 단위를 적용할 HTML 페이지를 생성한다. 본문 글자의 크기가 모두 동일하다.

```
ex0601.html  
  
<html>  
<head>  
    <title>CSS3 Style Property Basic</title>  
    <style>  
    </style>  
</head>  
<body>  
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>  
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>  
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>  
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>  
</body>  
</html>
```

1) % 단위 적용하기: % 단위는 기본으로 설정된 크기를 기준으로 상대적인 크기를 지정한다. 이때 100%가 초기에 설정된 크기이다.

```
p:nth-child(1) { }  
p:nth-child(2) { font-size: 100%; }  
p:nth-child(3) { font-size: 150%; }  
p:nth-child(4) { font-size: 200%; }
```

2) em 단위 적용하기: em은 굉장히 유용한 크기 단위로 배수를 나타내는 단위이다. 1배=1em=100%이며 1.5배=1.5em=150%이다.

```
p:nth-child(1) { }  
p:nth-child(2) { font-size: 1.0em; }  
p:nth-child(3) { font-size: 1.5em; }  
p:nth-child(4) { font-size: 2.0em; }
```

3) px 단위 적용하기: % 단위와 em 단위는 크기를 상대적으로 지정했지만 px 단위는 절대적인 크기를 지정한다. 지금 살펴본 font-size 속성을 포함해 width 속성, height 속성 등에 크기 단위를 적용할 수 있다.

```
p:nth-child(1) { }  
p:nth-child(2) { font-size: 16px; }  
p:nth-child(3) { font-size: 24px; }  
p:nth-child(4) { font-size: 32px; }
```

6.3 색상 단위

색상을 입력하는 가장 간단한 방법은 다음 예처럼 색상에 해당하는 영단어를 입력하는 것이다. 하지만 영단어로 표현할 수 있는 색상은 제한되어 있기 때문에 CSS3에서는 색상을 더 다양하게 표현할 수 있도록 아래의 색상 단위를 제공한다.

```
<style>
  h1 { background-color: red; }
  h2 { background-color: orange; }
  h3 { background-color: blue; }
</style>
```

단위 형태	설명
rgb(red, green, blue)	RGB색상 단위
rgba(red, green, blue, alpha)	RGBA색상 단위
#000000	HEX 코드 단위

RGB 단위는 R(붉은색), G(초록색), B(파란색)를 조합하여 색상을 표현하며 다음과 같이 0부터 255사이의 숫자를 입력한다.

```
<style>
  h1 { background-color: rgb(255, 255, 255); }
</style>
```

RGBA 색상 단위는 RGB 색상 단위에 알파 값을 추가한 형태이다. 알파 값은 투명도를 나타내며 0.0부터 1.0사이의 숫자를 입력한다. 0.0은 완전 투명 상태고, 1.0은 완전 불투명 상태이다.

```
<style>
  h1 { background-color: rgba(255, 255, 255, 0.5); }
</style>
```

HEX 코드 단위는 RGB 색상 단위를 짧게 입력하는 방법으로 RGB 색상 조합을 16진수로 입력한다. 색상을 16진수로 표현할 때는 각색상 요소에 00에서 FF 사이의 값을 입력한다. 색상 선택이 힘들다면 아래 사이트의 ColorPicker와 같은 색상 선택 도구를 사용한다.

<http://www.colorpicker.com/>

6.4 URL 단위

이미지나 글꼴 파일을 불러올 때는 URL 단위를 사용한다. 아래는 background-image 속성을 사용하여 배경 이미지를 지정하는 예제이다.

```
ex0602.html

<head>
  <style>
    body{background-image: url('Penguins.jpg');}
  </style>
</head>
<body>
  <h1>What is Lorem Ipsum?</h1>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry.</p>
</body>
```

이미지가 존재하는 위치에 따라 URL 경로를 다음과 같이 입력하여 파일에 접근한다. 운영체제에서 파일 경로를 표시하는 방법과 일치한다.

```
/* 현재 폴더의 Desert.jpg */
background-image: url('Desert.jpg');
/* other 폴더의 Desert.jpg */
background-image: url('other/Desert.jpg');
/* 루트 폴더의 Desert.jpg */
background-image: url('/Desert.jpg');
```

6.5 박스 속성

- width와 height, margin과 padding 속성

width와 height 속성은 글자를 감싸는 영역의 크기를 지정한다. margin은 테두리와 다른 태그와의 간격을 지정하고 padding은 테두리와 글자 사이의 간격을 지정하는 속성이다.

1) width와 height 속성 적용하기: div 태그로 만든 박스의 너비와 높이가 100px로 지정한다.

```
ex0603.html

<head>
  <style>
    div {
      width: 300px; height: 100px; background-color: red;
    }
  </style>
</head>
<body>
  <div class="box"></div>
</body>
```

2) margin과 padding 속성 적용하기: margin 속성에 10px을 적용하고 padding 속성에 30px을 적용한다.

```
<style>
  div {
    width: 100px; height: 100px;
    background-color: red;
    border: 20px solid black;
    margin: 10px; padding: 30px;
  }
</style>
```

3) width와 height 속성은 글자가 들어가는 영역의 너비와 높이를 지정 한다 따라서 태그 전체 크기는 다음 공식으로 나타낼 수 있다.

```
전체 너비 = width + 2 x (margin + border + padding)
전체 높이 = width + 2 x (margin + border + padding)
```

4) 다음 순서대로 크기 단위를 띄어쓰기로 구분하면 각각의 너비를 별도로 지정할 수 있다.

```
div {
  width: 100px; height: 100px;
  background-color: red;
  /*margin: 위쪽 오른쪽 아래쪽 왼쪽 */
  margin: 0px 30px 0px 30px;
  /*padding: 위쪽 오른쪽 아래쪽 왼쪽 */
  padding: 0px 30px 0px 30px;
}
```

5) margin과 padding 속성에 두 개의 값을 입력하는 경우도 있다. 네 개의 방향을 두 개씩 묶어 지정하는 것이다.

```
div {
  width: 100px; height: 100px;
  background-color: red;
  /*margin: 위쪽아래쪽 왼쪽오른쪽 */
  margin: 0px 30px;
  /*padding: 위쪽아래쪽 왼쪽오른쪽 */
  padding: 0px 30px;
}
```

- border-width, border-style, border-color 속성

테두리를 넣을 때는 두께, 형태, 색상에 해당하는 border-width, border-style, border-color 속성을 모두 사용해야 한다.

1) 테두리 속성 익히기

ex0604.html

```
<html>
<head>
  <style>
    .box {
      border-width: thick;
      border-style: dashed;
      background-color: black;
    }
  </style>
</head>
<body>
  <div class="box">
    <h1>What is Lorem Ipsum?</h1>
  </div>
</body>
</html>
```

2) border 속성

border-width, border-style, border-color 속성은 border 속성을 사용해 한 번에 입력할 수 있다. 또한 border 속성도 margin과 padding 속성처럼 왼쪽, 오른쪽, 위쪽, 아래쪽 속성을 각각 입력할 수 있다.

```
<style>
  .box {
    border: thick dashed black;
  }
</style>
```

3) border-radius 속성 적용하기: border-radius 속성을 적용하면 모서리가 둥근 테두리가 만들어진다.

```
<style>
  box {
    border-width: thick;
    border-style: dashed;
    background-color: black;

    border-radius: 20px;
  }
</style>
```

4) 모서리마다 다른 둥글기 적용하기: border-radius 속성에 크기 단위 네 개를 입력하면 각 테두리에 서로 다른 둥글기가 적용된다.

```
<style>
  box {
    border-width: thick;
    border-style: dashed;
    background-color: black;

    /*border-radius: 왼쪽위 오른쪽위 오른쪽아래 왼쪽아래*/
    border-radius: 50px 40px 20px 10px;
  }
</style>
```

6.6 display 속성

HTML5 시대가 되면서 아래 표와 같이 display 속성에 입력할 수 있는 다양한 키워드가 등장했다. 공간분할 태그에서 div 태그는 블록 형식이고 span 태그는 인라인 형식이다. display 속성을 이용하면 인라인 형식, 블록형식, 인라인-블록 형식을 설정할 수 있다.

키워드 이름	설명
none	태그를 화면에서 보이지 않게 만든다.
block	태그를 블록 형식으로 지정한다.
inline	태그를 인라인 형식으로 지정한다.
inline-block	태그를 인라인-블록 형식으로 지정한다.

- none 키워드 적용하기 : display 속성에 none 키워드를 적용하면 태그가 화면에서 보이지 않는다. 즉 div 태그 전체가 화면에서 사라진다.

ex0605.html

```
<head>
  <style>
    #box { display: none; }
  </style>
</head>
<body>
  <span>데미 객체</span>
  <div id="box">대상 객체</div>
  <spna>데미 객체</spna>
</body>
```

- block 키워드 적용하기: #box 태그의 display 속성을 block 키워드로 바꾸고 요소 검사에서 각 태그를 클릭하면 해당 요소가 선택된다.

```
<style>
  #box {
    display: block; }
</style>
```

- inline 키워드와 inline-block 키워드 적용하기 : width, height 속성, margin 속성을 주어 차이점을 확인해 보자

```
<style>
  #box {
    display: inline;
    background-color: red;
    /*width와 height 속성이 지정되지 않고 margin 속성이 div 태그의 좌우로만 적용된다.*/
    width: 100px;
    height: 100px;
    margin: 10px;
  }
</style>
```

```
<style>
  #box {
    display: inline-block;
    background-color: red;
    /*width와 height 속성이 지정되고 margin 속성도 상하좌우로 적용된다.*/
    width: 100px;
    height: 100px;
    margin: 10px;
  }
</style>
```

6.7 배경 속성

아래 표와 같이 배경 속성을 사용하여 해당 태그의 배경 이미지 또는 색상을 지정한다.

속성 이름	설명
background-image	배경 이미지 지정
background-size	배경 이미지의 크기 지정
background-repeat	배경 이미지의 반복 형태 지정
background-attachment	배경 이미지의 부착 형태 지정
background-position	배경 이미지의 위치 지정
background	한 번에 모든 배경 속성 입력

- background-image 속성: 배경 이미지를 지정하는 속성으로 스타일 값에는 URL단위 또는 그라데이션트를 입력한다.

1) 배경 이미지 한 개 삽입하기

```
<style>
  body {
    background-image: url('./image/BackgroundFront.png');
  }
</style>
```

2) 배경 이미지 두 개 삽입하기

```
<style>
  body {
    /*각각의 그림을 레이어라고 부른다.*/
    background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
  }
</style>
```

- background-size 속성: 배경 이미지 크기를 조절하는 속성이다. CSS3에 추가된 속성이므로 인터넷 익스플로러8 이하에는 사용할 수 없다.

1) 배경 이미지의 너비 조절하기

```
<style>
  body {
    background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
    background-size: 100%;
  }
</style>
```

2) 배경 이미지 높이 조절하기: background-size 속성에서 두 번째 입력하는 크기 단위는 높이를 의미한다.

```
<style>
  body {
    background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
    background-size: 100% 250px;
  }
</style>
```

3) background-size 속성과 관련된 주의 사항: 쉽표를 사용하면 각 배경에 서로 다른 크기를 입력하는 것으로 인식한다.

```
<style>
  body {
    background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
    /*첫 번째 이미지의 너비: 100%, 두 번째 이미지의 너비: 250px*/
    background-size: 100%, 250px;
  }
</style>
```

- **background-repeat** 속성: repeat를 입력하면 이미지가 패턴을 이루고, repeat-x를 입력하면 X축 방향으로만 이미지가 반복된다. 마찬가지로 repeat-y를 입력하면 Y축 방향으로 이미지가 반복된다.
- **background-attachment** 속성: 배경이미지를 화면에 고정하는 속성으로 스타일 값에는 여러 키워드(fixed, inherit, initial, local, scroll)를 입력할 수 있다. 기본 키워드는 scroll이다. 화면 스크롤에 따라 배경 이미지가 함께 이동한다.
- **background-position** 속성: 아이콘(스프라이트 이미지)을 만들 때 자주 사용하는 속성이다. 크기 단위를 다음과 같이 사용한다.

속성	속성 값
background-position	키워드
background-position	X축 위치
background-position	X축 위치 Y축 위치

• 배경 이미지 배치하기

1) 배경 이미지 반복 없애기: background-repeat 속성에 no-repeat 키워드를 입력한다. 반복 없이 배경 이미지가 적용된다.

ex0606.html

```
<html>
<head>
  <style>
    body {
      background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
      background-size: 100%;
      background-repeat: no-repeat;
    }
  </style>
</head>
<body>
</body>
</html>
```

2) 스크롤이 가능할 분량의 텍스트 추가하기: 스크롤을 형성할 수 있을 정도의 글을 입력한 후 스크롤을 내리면 이미지가 위로 올라가 사라진다.

ex0607.html

```
<html>
<head>
  <style>
    body {
      background-color: #E7E7E8;
      background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
      background-size: 100%;
      background-repeat: no-repeat;
      background-attachment: scroll;
    }
  </style>
</head>
<body>
  <h1>What is Lorem Ipsum?</h1>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. ... Lorem Ipsum.</p>
  <p>Lorem Ipsum is not simply random text. It has ... comes from a line in section 1.10.32.</p>
  <p>The standard chunk of Lorem Ipsum used since the 1500s is reproduced below ... translation by H. Rackham.</p>
  <p>There are many variations of passages of Lorem Ipsum available, but the majority ... words etc.</p>
  <p>It is a long established fact that a reader will be distracted by ...on purpose (injected humour and the like).</p>
  <p>The standard chunk of Lorem Ipsum used since the 1500s is reproduced below ... translation by H. Rackham.</p>
</body>
</html>
```

3) 배경 이미지 화면에 고정시키기: background-attachment 속성 값에 fixed를 입력한다. 스크롤을 이동해도 이미지는 고정된다.

```
<style>
  body {
    background-color: #E7E7E8;
    background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
    background-size: 100%;
    background-repeat: no-repeat;
    background-attachment: fixed;
  }
</style>
```

4) 배경 이미지 하단에 배치하기: background-position 속성에 bottom 키워드를 입력하면 이미지가 아래에 위치한다.

```
<style>
  body {
    background-color: #E7E7E8;
    background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
    background-size: 100%;
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-position: bottom;
  }
</style>
```

5) 배경 이미지 위치 고정하기: 다음 코드는 배경 이미지의 X축 위치를 0px로 지정하고 Y축 위치를 50%로 지정한다.

```
<style>
  body {
    background-color: #E7E7E8;
    background-image: url('./image/BackgroundFront.png'), url('./image/BackgroundBack.png');
    background-size: 100%;
    background-repeat: no-repeat;
    background-attachment: fixed;
    background-position: 0px 50%;
  }
</style>
```

6.8 글자 속성

HTML 페이지는 대학에서 원격으로 논문을 제출하고 확인하는 용도로 만들어 졌다. 따라서 CSS도 글자와 관련된 속성이 굉장히 많다.

- font-size 속성: 크기 단위나 키워드(inherit, initial, large, larger, medium, small, smaller, x-large, x-small)를 입력할 수 있다.

ex0608.html

```
<head>
  <style>
    .a { font-size: 32px; }
    .b { font-size: 2rem; }
    .c { font-size: 1large;}
    .d { font-size: small;}
  </style>
</head>
<body>
  <h1>What is Lorem Ipsum?</h1>
  <p class="a">What is Lorem Ipsum?</p>
  <p class="b">What is Lorem Ipsum?</p>
  <p class="c">What is Lorem Ipsum?</p>
  <p class="d">What is Lorem Ipsum?</p>
</body>
```


p태그의 기본 크기는 16px이다. 따라서 32px은 지정한 글자와 2배수를 적용한 글자의 크기가 같다. 또한 h1태그의 기본 크기는 32px이다.

- font-family 속성: font-family 속성의 스타일 값에는 사용자 컴퓨터에 설치된 글꼴을 입력할 수 있다. 개발에 사용하는 컴퓨터에는 글꼴이 설치되어 있지만 개발한 웹 페이지를 이용하는 사용자의 컴퓨터에는 글꼴이 없을 수 있을 수 있다. 이러한 경우를 고려하여 font-family 속성을 여러 개 입력한다. 그리고 다국적 사용자를 대상으로 웹 페이지를 제공할 경우 사용자에게 무슨 글꼴이 있는지 일일이 확인할 수 없으므로 웹 브라우저에서 지정하는 generic-family 글꼴을 입력한다.

1) 글꼴 한 개 지정하기: 일반적으로 한 단어로 이루어진 글꼴 이름은 따옴표를 사용하지 않지만 두 단어 이상인 경우 따옴표 사용을 이용한다.

ex0609.html

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .font_arial { font-family: Arial; }
    .font_roman { font-family: 'Times New Roman'; }
  </style>
</head>
<body>
  <h1 class="font_arial">What is Lorem Ipsum?</h1>
  <p class="font_roman">What is Lorem Ipsum?</p>
</body>
</html>
```

2) 글꼴 여러 개 지정하기: 웹 페이지를 사용할 사용자의 컴퓨터에는 없을 경우 font-family 속성을 여러 개 입력한다.

```
<style>
  .font_arial { font-family: '없는 글꼴', Arial; }
  .font_roman { font-family: 'Times New Roman', Arial; }
</style>
```

3) 웹 브라우저의 고유 글꼴 적용하기: 글꼴이 사용자의 컴퓨터에 설치되어있지 않으므로 사용자의 웹 브라우저에 설정된 고유의 글꼴을 출력한다.

```
<style>
  /*인터넷 익스플로러의 경우 sans-serif 글꼴과 serif 글꼴에 따옴표를 적용하면 안 된다*/
  .font_arial { font-family: '없는 글꼴', 'sans-serif'; }
  .font_roman { font-family: '없는 글꼴', serif; }
</style>
```

- font-style과 font-weight 속성: 글자의 두께와 기울기를 조정하는 속성이다. font-weight 속성에 400을 입력할 경우 일반 글자의 두께로 출력되고, 700을 입력하면 두껍게 출력된다. 두께를 지원하지 않는 글꼴은 font-weight 속성을 사용해 두께를 조절할 수 없다.

ex0610.html

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .font_big { font-size: 2em; }
    .font_italic { font-style: italic; }
    .font_bold { font-weight: bold; }
  </style>
</head>
<body>
  <p class="font_big font_italic font_bold">What is Lorem Ipsum?</p>
</body>
</html>
```

- text-align 속성: 글자의 정렬과 관련된 속성이다. text-align 속성에는 키워드(center, end, inherit, initial, justify, left, match-parent, right, start)를 입력할 수 있다.

ex0611.html

```
<head>
  <style>
    .font_big { font-size: 2em;}
    .font_italic { font-style: italic; }
    .font_bold { font-weight: bold; }
    .font_center { text-align: center; }
    .font_right { text-align: right; }
  </style>
</head>
<body>
  <p class="font_big font_italic font_bold font_center">What is Lorem Ipsum?</p>
  <p class="font_bold font_right">2013. 12. 09</p>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. ... versions of Lorem Ipsum.</p>
</body>
```

- line-height 속성: 글자의 높이를 지정하지만 현대의 HTML 페이지는 문서보다 애플리케이션 형태로 많이 사용되기 때문에 line-height 속성을 사용하여 글자의 높이를 지정하기보다는 글자를 수직 중앙 정렬할 때 사용한다.

1) 간단한 버튼 만들기: 다음은 글자가 수평으로만 중앙 정렬된다. block 속성에 수직정렬을 위한 속성이 없기 때문이다.

ex0612.html

```
<head>
  <style>
    .font_big { font-size: 2em;}
    .font_italic { font-style: italic; }
    .font_bold { font-weight: bold; }
    .font_center { text-align: center; }
    .button {
      width: 150px;
      height: 70px;
      background-color: #FF6A00;
      border: 10px solid #FFFFFF;
      border-radius: 30px;
      box-shadow: 5px 5px 5px #A9A9A9;
    }
    .button > a { display: block;}
  </style>
</head>
<body>
  <div class="button">
    <a href="#" class="font_big font_italic font_bold font_center">Click</a>
  </div>
</body>
```

2) 글자를 수직 중앙 정렬하기

```
.button {
  width: 150px;
  height: 70px;
  background-color: #FF6A00;
  border: 10px solid #FFFFFF;
  border-radius: 30px;
  box-shadow: 5px 5px 5px #A9A9A9;
}

.button > a { display: block; line-height: 70px; } /*글자를 감싸는 박스의 높이와 같은 크기의 line-height속성을 적용했다*/
```

- text-decoration 속성: a 태그에 href 속성을 지정하면 글자에 밑줄이 생기고 파란색으로 변경된다. text-decoration 속성으로 밑줄만 제거하고 색상은 color 속성을 사용해 별도로 지정한다.

ex0613.html

```
<html>
<head>
  <style>
    a {
      text-decoration: none;
    }
  </style>
</head>
<body>
  <a href="#">Where does it come from?</a>
</body>
</html>
```

6.8 위치 속성

프로그램을 개발 할 때 요소의 위치는 다음 두 가지 방법으로 설정한다.

- 절대 위치 좌표: 요소의 X좌표와 Y좌표를 설정해 절대 위치를 지정한다.
- 상대 위치 좌표: 요소의 입력한 순서에 따른 상대 위치를 지정한다.

- position 속성: HTML 태그의 위치 설정 방법을 변경할 때 사용한다. position 속성에는 아래와 같은 키워드를 입력할 수 있다.

키워드	설명
static	상대 위치 좌표 설정
relative	초기 위치에서 상하좌우로 이동
absolute	절대 위치 좌표 설정
fixed	화면을 기준으로 절대 위치 좌표 설정

- 1) absolute 속성 적용하기: 실행결과는 브라우저에 따라 약간 다르지만 파란색 상자만 보이거나 세 개의 상자가 모두 겹쳐보인다.

ex0613.html

```
<html>
<head>
  <style>
    .box {
      width:100px; height:100px;
      position:absolute;
    }
    .box:nth-child(1) { background-color:red; }
    .box:nth-child(2) { background-color:green; }
    .box:nth-child(3) { background-color:blue; }
  </style>
</head>
<body>
  <div>
    <div class="box"></div>
    <div class="box"></div>
    <div class="box"></div>
  </div>
</body>
</html>
```

2) left와 top 속성 적용하기: 실행하면 붉은색 상자가 맨 뒤에 있고 파란 색 상자가 맨 앞에 있다. 뒤에 작성한 코드가 상위에 출력된다.

```
<style>
  .box {
    width:100px; height:100px;
    position:absolute;
  }
  .box:nth-child(1) {
    background-color:red;
    left: 10px; top: 10px;
  }
  .box:nth-child(2) {
    background-color:green;
    left: 50px; top: 50px
  }
  .box:nth-child(3) {
    background-color:blue;
    left: 90px; top: 90px;
  }
</style>
```

- z-index 속성: 실행 결과에서 도형의 순서를 변경하고 싶을 때는 z-index 속성을 사용한다. z-index 속성에는 숫자를 입력하며 숫자가 클수록 앞에 위치한다.

ex0614.html

```
<html>
  <head>
    <style>
      .box {
        width:100px; height:100px;
        position:absolute;
      }
      .box:nth-child(1) {
        background-color:red;
        left: 10px; top: 10px;
        z-index: 100;
      }
      .box:nth-child(2) {
        background-color:green;
        left: 50px; top: 50px;
        z-index: 10;
      }
      .box:nth-child(3) {
        background-color:blue;
        left: 90px; top: 90px;
        z-index: 1;
      }
    </style>
  </head>
  <body>
    <div>
      <div class="box"></div>
      <div class="box"></div>
      <div class="box"></div>
    </div>
  </body>
</html>
```

일반적으로 "너는 반드시 앞에 있어야 한다."라는 태그의 z-index 속성에 9999 같이 터무니없게 큰 값을 입력하기도 한다.

- 위치 속성 공식: 위치 속성을 사용할 때 필요한 공식 몇 가지를 살펴보겠다. 굉장히 중요한 공식들이므로 따로 적어 기억해 둔다.

1) HTML 페이지 만들기: 위치 속성 공식을 적용하지 않은 HTML 페이지를 다음과 같이 작성한다.

```
<body>
  <h1>What is Lorem Ipsum?</h1>
  <div>
    <div class="box"></div>
    <div class="box"></div>
    <div class="box"></div>
  </div>
  <h1>What is Lorem Ipsum?</h1>
</body>
```

2) 문제 파악하기 : 실행 결과에 두 가지 문제가 있다.

- h1 태그 두 개가 붙어있다. (div 태그가 영역을 차지하지 않는다.)
- 색상이 적용된 상자가 자신의 부모를 기준으로 위치를 잡지 않는다.

```
body > div {
  width: 480px; height: 100px;
  border: 3px solid black;
}
```

3) 문제 해결하기

- 자손의 position 속성에 absolute 키워드를 적용하면 부모에게 height 속성을 입력한다.
- 자손의 position 속성을 absolute 키워드로 적용하면 부모의 position 속성을 relative 키워드로 적용한다.

```
body > div {
  ...
  position: relative;
}
```

- overflow 속성: overflow는 내부의 요소가 부모의 범위를 벗어날 때 요소를 처리하는 방법을 지정하는 속성이다. overflow 속성에는 아래의 키워드를 입력할 수 있다.

키워드	설명
hidden	영역을 벗어나는 부분을 보이지 않게 만든다.
scroll	영역을 벗어나는 부분을 스크롤로 만든다.

1) hidden 속성 적용하기: div 태그의 overflow 속성에 hidden 키워드를 입력하면 부모의 영역을 벗어난 부분이 모두 숨겨진다.

```
body > div {
  ...
  overflow: hidden;
}
```

2) scroll 속성 적용하기: overflow 속성에 scroll 키워드를 입력하면 div 태그에 스크롤이 생긴다.

```
body > div {
  ...
  overflow: scroll;
}
```

3) overflow-x와 overflow-y 속성 적용하기: 특정한 방향으로만 스크롤을 생성하고 싶을 때는 사용한다.

```
body > div {
  ...
  overflow-y: scroll;
}
```

6.9 float 속성

float는 웹 페이지를 만들 때 가장 많이 사용하는 속성이다. 원래는 부유하는 대상을 만들기 위해 개발되었지만 현대에는 레이아웃을 잡을 때 많이 사용한다. float 속성에는 아래의 키워드를 입력할 수 있다.

키워드	설명
left	태그를 왼쪽에 붙인다.
right	태그를 오른쪽에 붙인다.

- float 기본 속성 익히기

1) 기본 HTML 페이지 만들기: 코드를 실행하면 이미지와 글이 분리된 결과 화면이 출력된다. img는 인라인형식, p는 블록형식 태그이다.

ex0615.html

```
<html>
  <head>
    <style>

    </style>
  </head>
  <body>
    
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry... </p>
  </body>
</html>
```

2) img 태그에 float 속성 적용하기: img 태그에 float 속성을 적용하면 이미지와 글이 어우러진 형태로 출력된다.

```
<style>
  img { float: left; }
</style>
```

원래 float 속성은 부유하는 대상을 만들기 위해 만들었지만 현대에는 웹 페이지의 레이아웃을 만들 때에 더 많이 사용한다.

- 수평 정렬하기

1) float 속성을 적용할 대상 만들기: float 속성을 사용하면 태그를 수평으로 정렬할 수 있다.

ex0616.html

```
<html>
  <head>
    <style>

    </style>
  </head>
  <body>
    <div class="box">1</div>
    <div class="box">2</div>
  </body>
</html>
```

2) float 속성이 left 키워드인 경우 확인하기: float 속성에 left 키워드를 적용해 본다. 첫 번째에 위치한 div 태그가 맨 왼쪽에 붙는다.

```
<style>
  .box {
    width: 100px; height: 100px;
    background-color: red;
    margin: 10px; padding: 10px;
    /*태그를 왼쪽으로 붙인다.*/
    float: left;
  }
</style>
```

3) float 속성이 right 키워드인 경우 확인하기: float 속성을 right 키워드로 수정하고 실행하면 1번 div 태그가 오른쪽에 위치한다.

```
<style>
  .box {
    width: 100px; height: 100px;
    background-color: red;
    margin: 10px; padding: 10px;
    /*태그를 오른쪽으로 붙인다.*/
    float: right;
  }
</style>
```

6.10 그림자 속성

그림자 속성은 태그에 그림자를 부여하는 속성이다. 글자에 그림자를 부여하는 text-shadow와 박스에 그림자를 부여하는 box-shadow이 있다.

1) text-shadow 속성: text-shadow는 글자에 그림자를 부여하는 속성으로 아래의 형태로 값을 입력한다.

```
text-shadow: 5px(왼쪽) 5px(아래) 5px(흐름도) black(색상)
```

2) box-shadow: 박스에 그림자를 부여하는 속성으로 아래의 형태로 값을 입력한다.

```
box-shadow: 5px(왼쪽) 5px(아래) 5px(흐름도) black(색상)
```

• 글자와 박스에 그림자 만들기

1) 글자에 그림자 만들기: 아래처럼 입력하여 글자에 그림자를 생성한다.

ex0617.html

```
<html>
<head>
  <style>
    h1 {
      text-shadow: 5px 5px 5px black;
    }
  </style>
</head>
<body>
  <h1>What is Lorem Ipsum?</h1>
</body>
</html>
```

2) 박스에 그림자 만들기: 아래처럼 입력하여 박스에 그림자를 생성한다.

ex0618.html

```
<html>
<head>
  <style>
    div {
      border: 3px solid black;
      box-shadow: 10px 10px 30px black;
      text-shadow: 5px 5px 5px black;
    }
  </style>
</head>
<body>
  <div class="box">
    <h1>What is Lorem Ipsum?</h1>
  </div>
</body>
</html>
```

3) 중첩 그림자 만들기: 쉼표를 사용하면 여러 개의 그림자 속성을 적용할 수 있다. 다음 코드는 글자와 박스에 삼중 그림자를 적용한다.

```
<style>
div {
border: 3px solid black;
box-shadow: 10px 10px 30px black, 10px 10px 20px orange, 10px 10px 30px red;
text-shadow: 5px 5px 5px black, 10px 10px 20px orange, 10px 10px 30px red;;
}
</style>
```

4) 아래 주소의 CSS3 Generator를 사용하면 box-shadow와 text-shadow 속성으로 쉽게 생성할 수 있다.

<http://css3generator.com/>

6.11 그라데이션트

그라데이션트는 두 가지 이상의 색상을 혼합하는 채색 기능이다. CSS3 이전의 웹은 그라데이션트를 지원하지 않아 포토샵과 그래픽 툴을 이용했지만 CSS3부터는 그라데이션트 기능을 지원한다. 그라데이션트 코드 생성은 아래의 주소로 접속하여 이용한다.

<http://www.colorzilla.com/gradient-editor/> (Ultimate CSS Gradient Generator)

• 그라데이션트 만들기

- 1) 툴 이용해 그라데이션트 코드 생성하기: 위 주소로 접속하여 페이지의 왼쪽에서 원하는 형태의 그라데이션트를 생성한 후 CSS 코드를 복사한다.
- 2) 복사한 CSS 코드를 이용해 HTML 페이지 만들기: 복사한 CSS 코드를 붙여넣어 아래 코드처럼 HTML 페이지를 구성한다.

ex0610.html

```
<html>
<head>
<title>CSS3 Style Property Basic</title>
<style>
div {
font-family: 'Times New Roman';
height: 60px;
line-height: 60px;
text-align: center;

/*Old browsers*/
background: #e6f0a3;
/* FF3.6+ */
background: -moz-linear-gradient(left, #e6f0a3 0%, #d2e638 50%, #c3d825 51%, #dbf043 100%);
/* Chrome,Safari4+ */
background: -webkit-gradient(linear, left top, right top, color-stop(0%,#e6f0a3),
color-stop(50%,#d2e638),color-stop(51%,#c3d825), color-stop(100%,#dbf043));
/* Chrome10+,Safari5.1+ */
background: -webkit-linear-gradient(left, #e6f0a3 0%,#d2e638 50%,#c3d825 51%,#dbf043 100%);
/* Opera 11.10+ */
background: -o-linear-gradient(left, #e6f0a3 0%,#d2e638 50%,#c3d825 51%,#dbf043 100%);
/* IE10+ */
background: -ms-linear-gradient(left, #e6f0a3 0%,#d2e638 50%,#c3d825 51%,#dbf043 100%);
/* W3C */
background: linear-gradient(to right, #e6f0a3 0%,#d2e638 50%,#c3d825 51%,#dbf043 100%);
/* IE6-9 */
filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#e6f0a3',
endColorstr='#dbf043',GradientType=1 );
}
</style>
</head>
<body>
<div>
<h1>CSS3 그라데이션트</h1>
</div>
</body>
</html>
```


7장 CSS3 응용과 레이아웃

7.1 수평 정렬하기

수평으로 요소를 정렬할 때는 반드시 다음 규정을 기억해야 한다.

자손에게 float 속성을 지정하고 부모의 overflow 속성을 hidden으로 적용한다.

아래 코드는 div.item 태그에 float 속성을 left로 적용하고 부모의 overflow 속성을 hidden으로 적용하여 수평을 정렬하였다.

ex0701.html

```
<html>
<head>
  <style>
    div.container {
      overflow: hidden;
    }
    div.item {
      float: left;
      margin: 0px 3px;
      padding: 10px;
      border: 1px solid black;
    }
  </style>
</head>
<body>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has ...</p>
  <div class="container">
    <div class="item">메뉴 - 1</div>
    <div class="item">메뉴 - 2</div>
    <div class="item">메뉴 - 3</div>
    <div class="item">메뉴 - 4</div>
  </div>
  <p>It is a long established fact that a reader will be distracted by the readable content ...</p>
</body>
</html>
```

7.2 중앙 정렬 레이아웃 구성하기

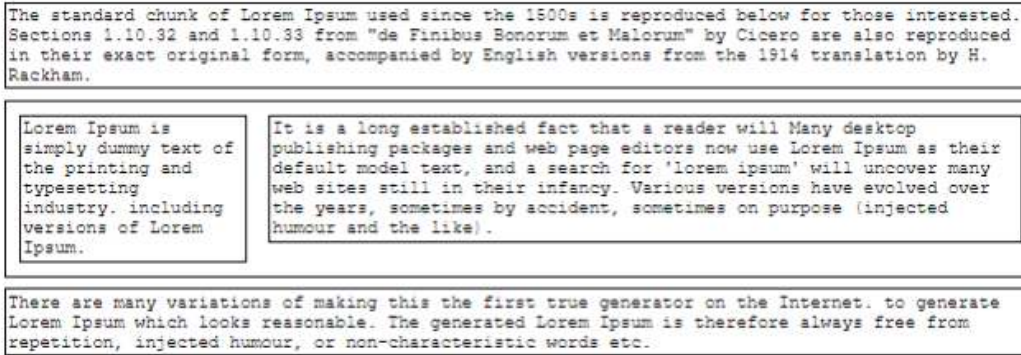
중앙 정렬은 현대에 가장 많이 사용되는 레이아웃 구성 방법이다. 포털사이트를 살펴보면 특정한 너비를 가지고 내용이 중앙에 위치한다. 이러한 구성을 중앙 정렬 에어아웃이라고 부른다. 중앙 정렬 요소에 해당하는 태그에 width 속성을 부여하고 margin 속성을 '0 auto'로 입력한다.

ex0702.html

```
<html>
<head>
  <style>
    /*초기화*/
    * { margin: 0px; padding: 0px;}
    body {
      margin: 0px auto;
      width: 960px; }
  </style>
</head>
<body>
  <h1>Lorem ipsum</h1>
  <h2>What is Lorem Ipsum?</h2>
  <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry ...</p>
  <p>It is a long established fact that a reader will be distracted by the readable ...</p>
</body>
</html>
```

7.3 One true 레이아웃 구성하기

One True 레이아웃은 아래와 같은 페이지를 만들 때 사용하는 기본 레이아웃 구성 방법이다. 국내 모든 포털 사이트의 메인페이지는 One True 레이아웃을 사용해 개발되었다.



1) One True 레이아웃의 행 구성: 행을 독립적으로 생각해서 공간을 나눈다.

```
<div id="top"></div>
<div id="middle"></div>
<div id="bottom"></div>
```

2) One True 레이아웃의 열 구성: 이어서 열을 구분한다. 아래에서는 중간에 있는 공간이 두 개로 나누어진다.

```
<div id="top"></div>
<div id="middle">
  <div id="left"></div>
  <div id="right"></div>
</div>
<div id="bottom"></div>
```

3) 부모 태그에 고정된 너비를 입력한다.

4) 수평 정렬하는 부모 태그의 overflow 속성에 hidden을 적용한다.

5) 자손 태그에 적당한 너비를 입력하고 float 속성을 적용한다.

중앙 정렬 레이아웃을 사용해 body 태그에 고정된 너비를 설정하고 요소를 수평으로 정렬해야 하는 #middle 태그의 overflow 속성에 hidden을 적용했다. 마지막으로 #left 태그와 #right 태그에 적당한 너비를 설정하고 float 속성을 적용한다. 또한 배경 색상을 넣었다.

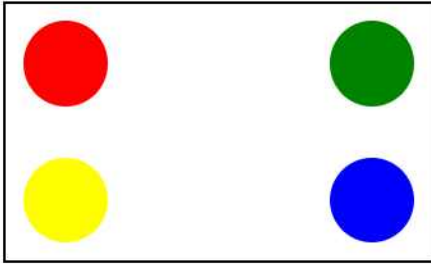
ex0703.html

```
<!DOCTYPE html>
<html>
<head>
  <style>
    body { width: 500px; margin: 10px auto; }
    #middle { overflow: hidden;}
    #left { float: left; width: 150px; background: red; }
    #right { float: left; width: 350px; background: blue; }
    #top { background: green; }
    #bottom { background: purple; }
  </style>
</head>
<body>
  <div id="top">The standard chunk of Lorem Ipsum used since the 1500s is reproduced...</div>
  <div id="middle">
    <div id="left">Lorem Ipsum is simply dummy text of the printing and typesetting industry...</div>
    <div id="right">It is a long established fact that a reader will Many desktop publishing packages ...</div>
  </div>
  <div id="bottom">There are many variations of making this the first true generator on the Internet ...</div>
</body>
</html>
```

7.4 절대 위치를 사용해 요소 배치하기

절대 위치를 사용한 요소 배치로 직접 아래와 같은 페이지를 만들어보자. 참고로 원은 div 태그에 border-radius 속성을 적용해 만든다.

Dummy Text



Dummy Text

절대 위치를 사용해 요소를 배치하려면 아래 내용을 이용 한다.

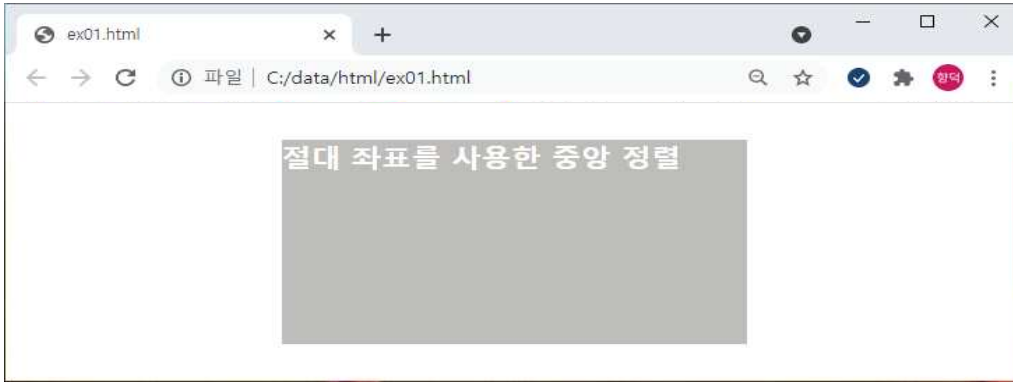
자신의 position 속성에 absolute를 적용하면 부모의 position 속성에 relative를 적용한다.

ex0704.html

```
<!DOCTYPE html>
<html>
<head>
  <style>
    #container {
      width: 500px; height: 300px;
      border: 3px solid black;
      overflow: hidden;
      position: relative; }
    .circle {
      position: absolute;
      width: 100px; height: 100px;
      border-radius: 50% 50%; }
    #red {
      background: red;
      left: 20px; top: 20px; }
    #green {
      background: green;
      right: 20px; top: 20px; }
    #blue {
      background: blue;
      right: 20px; bottom: 20px; }
    #yellow {
      background: yellow;
      left: 20px; bottom: 20px; }
  </style>
</head>
<body>
  <h1>Dummy Text</h1>
  <div id="container">
    <div id="red" class="circle"></div>
    <div id="green" class="circle"></div>
    <div id="blue" class="circle"></div>
    <div id="yellow" class="circle"></div>
  </div>
  <h1>Dummy Text</h1>
</body>
</html>
```

7.5 절대 위치를 사용해 요소를 중앙에 배치하기

다음 단계를 거쳐 요소를 화면 가운데에 배치할 수 있다. 그러면 요소가 왼쪽 위로 끌어당겨지면서 요소의 중심이 페이지 중앙에 위치한다. 이때 입력하는 값은 div 태그의 너비와 높이의 정확히 반이어야 한다.



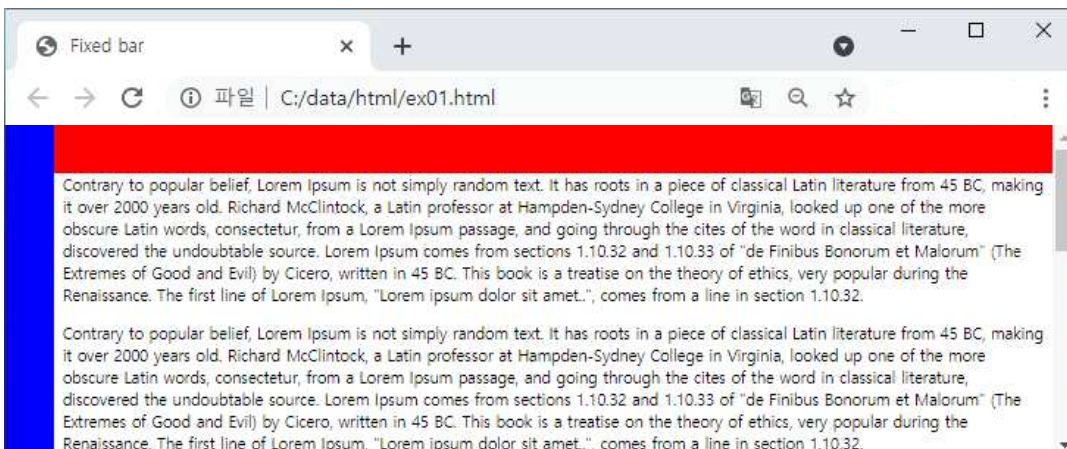
- 중앙 정렬하려는 div 태그의 position 속성을 absolute로 지정한다.
- left 속성과 top 속성을 모두 50%로 지정한다.
- 중앙에 정렬하려는 div 태그의 margin-left 속성과 margin-top 속성에 음수를 입력한다.

ex0705.html

```
<head>
  <style>
    * { margin:0; padding:0; }
    body { background: red; }
    #container {
      width:500px; height:250px;
      background: orange;
      position: absolute;
      left: 50%; top: 50%;
      margin-left: -250px;
      margin-top: -125px; }
  </style>
</head>
<body>
  <div id="container">
    <h1>절대 좌표를 사용한 중앙 정렬</h1>
  </div>
</body>
```

7.6 고정 위치를 사용해 바 생성하기

최근 만들어지는 웹 페이지에는 아래처럼 바가 있다. 바는 웹 페이지의 왼쪽, 오른쪽, 위, 아래에 붙어서 사용자를 따라다니는 요소로 인터넷 쇼핑몰이나 웹 응용 프로그램에서 많이 사용된다.



이러한 고정 바를 만드는 방법은 굉장히 간단하다. 우선 position 스타일 속성에 fixed를 적용하고 left, top, right, bottom 속성에 값을 입력해 위치를 설정한다. 이어서 width 속성과 height 속성으로 크기를 설정하면 된다. 이를 구현하면 아래 예제와 같다.

ex0706.html

```
<html>
<head>
  <title>Fixed bar</title>
  <style>
    .container {
      margin-top:50px; margin-left:50px;
    }
    .top_bar {
      background: red;
      position: fixed;
      left: 0; top: 0; right: 0;
      height: 50px;
    }
    .left_bar {
      background: blue;
      position:fixed;
      left:0; top:0px; bottom:0;
      width:50px;
    }
  </style>
</head>
<body>
  <div class="top_bar"></div>
  <div class="left_bar"></div>
  <div class="container">
    <p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has ...</p>
    <p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has ...</p>
    <p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has ...</p>
    <p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has ...</p>
  </div>
</body>
</html>
```

7.7 글자 생략하기

인터넷을 돌아다니다 보면 게시판의 목록 페이지나 모바일 페이지에서 글자가 생략된 모습을 자주 볼 수 있다. 이는 한 행을 넘어가는 글자를 자동으로 생략하는 기능이다.

ex0707.html

```
<head>
  <title>Ellipsis</title>
  <style>
    h1, p {
      width: 300px;
    }
    .ellipsis {
      white-space: nowrap;
      overflow: hidden;
      text-overflow: ellipsis;
    }
  </style>
</head>
<body>
  <h1 class="ellipsis">Lorem Ipsum is simply dummy text of the printing and typesetting industry.</h1>
  <p class="ellipsis">Lorem Ipsum is simply dummy text of the printing and typesetting industry...</p>
</body>
```

Lorem Ipsum is si...

Lorem Ipsum is simply dummy text of ...

• 드롭다운 메뉴바 레이아웃 예제

HTML5	CSS3	JavaScript
데이터	데이터	데이터
데이터	데이터	데이터
	데이터	데이터
		데이터
		데이터
		데이터

```

<script src="http://code.jquery.com/jquery-1.10.2.js"></script>
<script>
    $(document).ready(function() {
        $('.outer-menu-item').hover(function() {
            $(this).find('.inner-menu').show();
        }, function() {
            $(this).find('.inner-menu').hide();
        });
    });
</script>

```

ex0708.html

```

<head>
    <link rel="stylesheet" href="./navigation.css"/>
</head>
<body>
    <div id="page-wrapper">
        <nav id="main-navigation">
            <div class="pull-left">
                <ul>
                    <li class="outer-menu-item">
                        <span class="menu-title">HTML5</span>
                        <ul class="inner-menu">
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                        </ul>
                    </li>
                    <li class="outer-menu-item">
                        <span class="menu-title">CSS3</span>
                        <ul class="inner-menu">
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                        </ul>
                    </li>
                    <li class="outer-menu-item">
                        <span class="menu-title">JavaScript</span>
                        <ul class="inner-menu">
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                            <li class="inner-menu-item"><a href="#">데이터</a></li>
                        </ul>
                    </li>
                </ul>
            </div>
            <div class="pull-right">
                <div class="search-bar">
                    <form>
                        <input type="text" class="input-search">
                        <input type="submit" class="input-search-submit" value="검색"/>
                    </form>
                </div>
            </div>
        </nav>
    </div>
</body>

```

```

/*초기화-----*/
* {
    margin: 0; padding: 0;
    font-family: '맑은 고딕', 'Malgun Gothic', Gothic, sans-serif;
}
a {text-decoration: none; }
li {list-style: none; }
/*페이지 구성하기-----*/
body{
    width: 960px; margin:0 auto;
    background: #E6E6E6;
}
#page-wrapper {
    background: white;
    margin: 40px 0; padding: 10px 20px;
    border-radius: 5px;
    box-shadow: 0 2px 6px rgba(100, 100, 100, 0.3);
}
/*네비게이션 풀다운 메뉴-----*/
#main-navigation {
    border-top: 1px solid #C8C8C8;
    border-bottom: 1px solid #C8C8C8;
    height: 40px;
}
.pull-left { float:left; }
.outter-menu-item {
    float: left;
    position: relative;
}
.outter-menu-item:hover {
    background: black;
    color: white;
}
.menu-title {
    display: block;
    height: 30px; line-height: 30px;
    text-align: center;
    padding: 5px 20px;
}
.inner-menu {
    display: block;
    position: absolute;
    top: 40px; left: 0;
    width: 100%;
    background: white;
    box-shadow: 0 2px 6px rgba(5, 5, 5, 0.9);
    z-index: 1000;
    text-align: center;
}
.inner-menu-item > a {
    display: block;
    padding: 5px 10px;
    color: black;
}
.inner-menu-item > a:hover {
    background: black;
    color: white;
}

```

```
.pull-right { float:right; }
.search-bar {
    height: 26px;
    padding: 7px;
}
.input-search {
    display: block;
    float: left;
    background-color: #FFFFFF;
    border: 1px solid #CCCCCC;
    border-radius: 15px 0 0 15px;
    box-shadow: inset 0 1px 1px rgba(0, 0, 0, 0.05);

    width: 120px; height: 24px;
    padding: 0 0 0 10px;
    font-size: 12px;
    color: #555555;
}
.input-search:focus{
    border-color: rgba(82, 168, 236, 0.8);
    outline: 0;
    box-shadow: inset 0 1px 1px rgba(0, 0, 0, 0.05);
}
.input-search-submit{
    display: block;
    float: left;
    width: 50px; height: 24px;
    border-radius: 0 15px 15px 0;
    border: 1px solid #CCCCCC;
    margin-left: -1px;
    vertical-align: top;
    display: inline-block;
}
```

• 문서 레이아웃 예제

HTML5 개요와 활용

2013년 02월 14일

240 x 160

HTML5는 큰 의미와 작은 의미로 나누어 생각할 수 있습니다. 큰의미의 HTML5는 웹 표준 기술을 총칭하고, 작은 의미의 HTML5는 웹 문서의 문법을 의미합니다. 대개 HTML5라고 하면 큰 의미의 HTML5를 말합니다.

카테고리

데이터
데이터
데이터
데이터
데이터

최근 글

데이터
데이터
데이터
데이터

웹 페이지에서는 글자 태그가 가장 많은 비중을 차지합니다. 원래 논문과 관련된 목적으로 작성한 웹 문서라는 개념에서 시작되었는데, 문서는 대부분 글자로 구성되므로 웹 페이지에는 다양한 글자 태그가 담겨 있습니다.


```

<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="./content.css"/>
</head>
<body>
  <div id="page-wrapper">
    <div id="content">
      <section id="main-section">
        <article>
          <div class="article-header">
            <h1 class="article-title">HTML5 개요와 활용</h1>
            <p class="article-date">2013년 02월 14일</p>
          </div>
          <div class="article-body">
            <br/><br/>
            <p>HTML5는 큰 의미와 작은 의미로 나누어 생각할 수 있습니다. 큰 의미의 HTML5는 웹 표준 기술을 총칭하고,
              작은 의미의 HTML5는 웹 문성의 문법을 의미합니다. 대개 HTML5라고 하면 큰 의미의 HTML5를 말합니다.
            </p>
          </div>
        </article>
        <article>
          <div class="article-header">
            <h1 class="article-title">HTML5 응용과 실습</h1>
            <p class="article-date">2012년 2월 14일</p>
          </div>
          <div class="article-body">
            <br/><br/>
            <p>웹 페이지에서는 글자 태그가 가장 많은 비중을 차지합니다. 원래 논문과 관련된 목적으로 작성한 웹 문서라는
              개념에서 시작되었는데, 문서는 대부분 글자로 구성되므로 웹 페이지에는 다양한 글자 태그가 담겨 있습니다.
            </p>
          </div>
        </article>
      </section>
      <aside id="main-aside">
        <div class="aside-list">
          <h3>카테고리</h3>
          <ul>
            <li><a href="#">데이터</a></li>
            <li><a href="#">데이터</a></li>
            <li><a href="#">데이터</a></li>
            <li><a href="#">데이터</a></li>
            <li><a href="#">데이터</a></li>
          </ul>
        </div>
        <div class="aside-list">
          <h3>최근 글</h3>
          <ul>
            <li><a href="#">데이터</a></li>
            <li><a href="#">데이터</a></li>
            <li><a href="#">데이터</a></li>
            <li><a href="#">데이터</a></li>
          </ul>
        </div>
      </aside>
    </div>
  </div>
</body>
</html>

```

```

/*초기화-----*/
* {
    margin: 0; padding: 0;
    font-family: '맑은 고딕', 'Malgun Gothic', Gothic, sans-serif;
}
a { text-decoration: none; }
li { list-style: none; }
/*페이지 구성하기-----*/
body{
    width: 960px; margin:0 auto;
    background: #E6E6E6;
}
#page-wrapper {
    background: white;
    margin: 40px 0; padding: 10px 20px;
    border-radius: 5px;
    box-shadow: 0 2px 6px rgba(100, 100, 100, 0.3);
}
/*본문 (중앙내용)-----*/
#content { overflow: hidden; }
#main-section {
    float: left;
    width: 710px;
}
#main-aside {
    float: right;
    width: 200px;
}
/*본문 (포스트 상단)-----*/
article{
    padding: 0 10px 20px 10px;
    border-bottom: 1px solid #C8C8C8;
}
.article-header { padding: 20px 0 }
.article-title {
    font-size: 25px;
    font-weight: 500;
    padding-bottom: 10px;
}
.article-date {
    font-size: 13px;
}
.article-body {
    font-size: 14px;
}
/*본문 (사이드메뉴)-----*/
.aside-list {
    padding: 10px 0 30px 0
}
.aside-list > h3 {
    font-size: 15px;
    font-weight: 600;
}
.aside-list li a {
    margin-left: 8px;
    font-size: 13px;
    color: #6C6C6C;
}

```

8장 자바스크립트 기본

8.1 자바스크립트 기본 용어

자바스크립트에서 사용하는 기본 용어부터 알아보겠다. 앞으로도 계속 나오는 용어이므로 주의 깊게 살펴보자.

- 표현식과 문장

값을 만들어내는 간단한 코드를 표현식이라고 부르고 표현식이 하나 이상 모이면 문장이 된다. 자바스크립트에서는 코드 맨 마지막에 세미콜론을 적어 종결을 나타낸다. 표현식 하나도 세미콜론을 찍으면 문장이라고 부른다. 그리고 문장이 모이면 프로그램이 된다.

```
273;
10 + 20 + 30 * 2;
var name = '홍' + '길' + '동';
alert('Hello JavaScript');
```

- 키워드: 키워드는 특별한 의미가 부여된 단어로 자바스크립트가 처음 만들어질 때 정해졌다. 현재 브라우저에서는 다음과 같은 키워드를 지원한다.

break	else	instanceof	true
case	false	new	try
catch	finally	null	typeof
continue	for	return	var
default	function	switch	void
delete	if	this	while
do	in	throw	with

- 식별자: 식별자는 자바스크립트에서 이름을 붙일 때 사용하는 단어이다. 변수나 함수 이름 등으로 사용되며 다음과 같은 규칙에 맞게 생성한다.

- 1) 키워드를 사용하면 안 된다.
- 2) 숫자로 시작하면 안 된다.
- 3) 특수문자는 _과 \$만 허용한다.
- 4) 공백은 입력하면 안 된다.
- 5) 생성자 함수의 이름은 항상 대문자로 시작한다.
- 6) 변수, 인스턴스, 함수, 메서드의 이름은 항상 소문자로 시작한다.
- 7) 여러 단어로 이뤄진 식별자는 각 단어의 첫 글자를 대문자로 한다.

자바스크립트의 식별자는 크게 변수, 속성, 함수, 메서드 네 종류로 나눌 수 있다.

구분	단독으로 사용	다른 식별자와 함께 사용
식별자 뒤에 괄호 없음	변수	속성
식별자 뒤에 괄호 있음	함수	메서드

다음은 위에 표를 이용하여 식별자를 변수, 속성, 함수, 메서드로 구분한 예제이다.

```
alert('Hello Word') →함수
Array.lenth →속성
input →변수
prompt('Message', 'Defstr') →함수
Math.PI →속성
Math.abs(-273) →메서드
```

- 주석

주석은 프로그램의 진행에 전혀 영향을 주지 않는 코드로 프로그램을 설명하는데 사용한다. 다음은 자바스크립트에서 주석을 사용한 예제이다.

```
<script>
  //한줄 주석이다. 코드의 실행에 영향을 주지 않는다.
  /*
  alert('Hello JavaScript...!');
  alert('Hello JavaScript...!');
  alert('Hello JavaScript...!');
  */
</script>
```

8.2 자바스크립트 출력

자바스크립트의 가장 기본적인 출력 방법은 `alert()` 함수를 사용하여 웹 브라우저에 경고창을 띄우는 것이다. 사용방법은 아래와 같다.

```
alert("메시지");
```

- 자바스크립트 기본 출력 익히기

ex0801.html

```
<head>
  <script>
    alert('Hello JavaScript .. !');
  </script>
</head>
<body>
</body>
```

- 크롬 개발자 도구를 이용한 실행 결과 확인

간단한 내용을 살펴볼 때 파일을 하나하나 만들고 `alert()` 함수를 사용하는 것은 비효율적이다. 더 간단하게 자바스크립트 실행 결과를 확인할 수 있는 방법은 크롬 웹 브라우저에서 [F12]키를 눌러 개발자 도구의 [Console] 탭을 사용하는 것이다.

[F12] -> [Console] -> [자바스크립트 코드작성] -> [Enter]

8.3 자료형

프로그래밍 언어는 자료를 손쉽게 다루기 위해 만들어졌다. 따라서 어떤 프로그래밍 언어를 공부하든 해당 프로그래밍 언어에서 사용하는 자료형을 먼저 알아보아야 한다. 자바스크립트에는 숫자, 문자열, 불리언, 함수, 객체, 정의되지 않은 자료형이 있다.

- 숫자

가장 기본적인 자료형은 숫자이다. 자바스크립트는 정수와 실수를 구분하지 않으므로 숫자를 굉장히 쉽게 다룰 수 있다. 자바스크립트에서는 숫자를 사용하여 다음과 같은 사칙 연산을 수행할 수 있다.

연산자 기호	설명
+	덧셈연산
-	뺄셈연산
*	곱셈연산
/	나눗셈연산
%	나머지 연산자

- 문자열

문자의 집합을 문자열이라고 부른다. 자바스크립트에서는 큰따옴표나 작은따옴표 안에 문자를 넣어 문자열을 생성한다.

```
"문자열입니다." → 문자열 입니다
"'문자열 입니다'" → '문자열 입니다'
```

문자열은 아래와 같이 문자열 연결 연산자를 사용하여 합칠 수도 있다.

연산자	설명
+	문자열 연결 연산자

"가나다" + "라마" + "바사아" + "자차카타" + "파하" → 가나다라마바사아자차카타파하

- 불리언

불리언은 결과를 참과 거짓으로 표현할 때 사용한다. 숫자와 문자열은 만드는 방법에 따라 수많은 종류를 만들 수 있지만 불리언은 true와 false 두 가지만 만들 수 있다.

```
52 < 273 → true
52 > 273 → false
```

두 대상을 비교할 수 있는 연산자를 비교 연산자라고 부른다. 자바스크립트는 다음과 같은 비교 연산자를 지원한다.

연산자	설명
>=	좌변이 우변보다 크거나 같다.
<=	우변이 좌변보다 크거나 같다.
>	좌변이 우변보다 크다.
<	우변이 좌변보다 크다.
==	좌변과 우변이 같다.
!=	좌변과 우변이 다르다.

비교 연산자를 사용해 숫자는 물론 문자열도 비교할 수 있다. 문자열은 국어 사전의 앞쪽에 위치할 수록 값이 작다. 다음 표현식에서는 가방이 하마보다 앞쪽에 있으므로 false의 값을 갖는다.

```
"가방" > "하마" → false
```

유니코드 문자를 사용하여 전 세계의 모든 언어를 비교할 수 있고 방금 살펴본 비교 연산자를 사용하면 불리언을 만들 수 있다. 불리언끼리는 논리연산자를 사용한다. 자바스크립트는 다음과 같은 세 가지 논리 연산자를 제공한다.

연산자	설명
!	논리 부정 연산자
	논리곱 연산자
&&	논리합 연산자

논리 부정 연산자는 단순히 참을 거짓으로 거짓을 참으로 바꾼다.

```
!true → false
!(10 == 10) → false
```

이제 논리합과 연산자와 논리곱 연산자를 알아보겠다. 논리합 연산자는 좌변과 우변이 모두 거짓일 때만 거짓이다. 반면 논리곱 연산자는 좌변과 우변이 모두 참일 경우에만 참이다. 이 둘을 표로 정리하면 다음과 같다.

좌변	우변	논리합 연산결과	논리곱 연산 결과
true	true	true	true
true	false	true	false
false	true	true	false
false	false	false	false

논리 연산자는 다음과 같은 방식으로 사용한다.

```
true || false → true
(10 < 20) && (20 < 30) → true
```

8.4 변수

변수는 값을 저장할 때 사용하는 식별자이다. 이름은 변수지만 숫자뿐만 아니라 모든 자료형을 저장할 수 있다. 수학자들이 3.14159265...라는 원주율 값을 π 라고 간단히 표시하는 것과 비슷하다. 다음 두 단계를 거쳐 사용할 수 있다.

- 1) 변수를 선언한다.
- 2) 변수를 초기화한다.

변수를 만드는 것을 "변수를 선언한다."라고 표현한다. 변수는 var 키워드 뒤에 식별자를 입력하는 방법으로 선언한다.

```
var pi;
```

변수에 값을 저장하는 것을 "변수에 값을 할당한다."라고 표현한다. 다음과 같이 변수 pi에 값을 할당해 보자.

```
var pi;  
pi = 3.14159265;
```

변수를 선언한 후 처음 값을 할당하는 것을 "변수를 초기화한다."라고 표현한다. 일반적으로 변수의 선언과 초기화를 다음과 같이 한 번에 처리한다.

```
var pi = 3.14159265;
```

변수에 값이 저장되면 다음과 같이 변수를 사용해 저장된 값을 활용할 수 있다. 다음과 같이 입력하면 변수 pi에 저장된 값이 출력된다. 숫자가 저장된 변수는 산술 연산자를 이용하고 문자열이 저장된 변수는 문자열 연결 연산자와 같이 문자열과 관련된 연산자를 이용한다.

```
var pi = 3.14159265;  
alert(pi);
```

- 변수 사용법(선언과 초기화) 익히기

- 1) 기본 HTML 페이지 만들기: 먼저 자바스크립트를 실행할 수 있는 기본 HTML 페이지를 만든다. 이후 예제에서도 다른 태그를 별도로 수정하지 않은 이상 script 태그 부분만 표시한다.

```
ex0802.html
```

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>JavaScript Basic</title>  
  <script>  
  
  </script>  
</head>  
<body>  
  
</body>  
</html>
```

- 2) 변수 사용하기: 변수를 사용하여 자바 스크립트 코드를 script 태그 사이에 입력한다. 코드를 실행하면 다음과 같이 반지름이 10인 원의 둘레가 구해진다.

```
<script>  
  //변수를 선언 및 초기화한다.  
  var radius = 10;  
  var pi = 3.14159265;  
  
  //출력한다.  
  alert(2 * radius * pi);  
</script>
```

8.5 조건문

프로그램은 선택의 기로에 서는 순간이 있다. 바로 조건문을 만날 때이다. 이번에는 프로그램의 흐름을 변화시킬 수 있는 조건문을 살펴보자.

- if 조건문

if 조건문은 가장 기본적인 조건문으로 불리언 표현식이 true면 문장을 실행하고 false면 문장을 무시한다. 조건문을 만족해 실행되는 문장이 한 행이라면 중괄호를 생략할 수 있지만 문장이 여러 행이라면 중괄호로 감싸야 한다.

1) if 조건문으로 참, 거짓 구분하기

아래 코드를 실행하면 if 조건문 표현식이 거짓이므로 내부의 문장을 실행하지 않는다.

ex0802.html

```
<script>
  //조건문
  if (273 < 52) {
    //표현식 "273 < 52"가 참일 경우 실행한다.
    alert('273 < 52 => true');
  }

  //프로그램 종료
  alert('프로그램 종료');
</script>
```

2) 현재 시간 구하기

```
<script>
  //Date 객체를 선언한다. 현재시간 측정
  var date = new Date();

  //요소를 추출한다.
  var year = date.getFullYear();
  var month = date.getMonth() + 1;
  var day = date.getDay();
  var hours = date.getHours();
  var minutes = date.getMinutes();
  var seconds = date.getSeconds();
</script>
```

3) 오전과 오후 구분하기

현재 시간을 구하는 방법만 알면 쉽게 오전과 오후를 구분할 수 있다. 앞의 방법을 사용하여 현재 시간을 구하고 오전과 오후를 판단하는 조건문 두 개와 경고창으로 메시지를 표시하는 코드를 추가한다. 다른 시간 요소를 구하는 방법도 살펴보았으므로 어떻게 계절을 구분할지도 생각해보자.

ex0802.html

```
<script>
  //변수를 선언한다.
  var date = new Date();
  var hours = date.getHours();

  //조건문
  if (hours < 12) {
    //표현식이 참일 경우 실행한다.
    alert('오전입니다.');
```

```
  }

  if (12 <= hours) {
    //표현이 참일 경우 실행한다.
    alert('오후입니다.');
```

```
  }
</script>
```

- if else 조건문

오전과 오후같이 조건에 따라 두 가지 경우로 나뉘는 상황이 아주 많다. 자바스크립트는 두 가지로 분명하게 나뉘지는 상황에서 편리하게 사용할 수 있는 if else 조건문을 제공한다. if 조건문처럼 실행되는 문장이 하나면 중괄호를 생략할 수 있지만 문장이 여러 행이라면 중괄호로 감싸줘야 한다.

```
if (불리언 표현식) {  
    문장 1  
} else {  
    문장 2  
}
```

다음은 if else 조건문으로 조건에 따라 분기하는 예제이다. if else 조건문을 사용하면 한 번만 비교하므로 효율적이다.

ex0802.html

```
<script>  
    //변수를 선언한다.  
    var date = new Date();  
    var hours = date.getHours();  
    //조건문  
    if (hours < 12) {  
        //표현식이 참일 경우 실행한다.  
        alert('오전입니다.');    }else {  
        //표현식이 거짓일 경우 실행한다.  
        alert('오후입니다.');    }  
</script>
```

- 중첩 조건문과 if else if 조건문

1) 중첩 조건문: 조건문은 중첩해서 사용할 수 있다. 아래 코드는 오전과 오후가 아니라 하루의 일정을 표현하기위해 다음처럼 중첩 조건문을 사용하였다.

ex0803.html

```
<script>  
    //Date 객체를 선언한다. 현재 시간 측정  
    var date = new Date();  
    var hours = date.getHours();  
    //조건문  
    if (hours < 5) {  
        alert('잠을 자렴 .....');    } else {  
        if (hours < 7) {  
            alert('준비');        } else {  
            if (hours < 9) {  
                alert('출근');            } else {  
                if (hours < 12) {  
                    alert('빈둥빈둥');                } else {  
                    if (hours < 1) {  
                        alert('식사');                    } else {  
                        // 여러 가지 업무를 수행합니다.  
                    }  
                }  
            }  
        }  
    }  
</script>
```



```

if (불리언 표현식) {
    if (불리언 표현식) {
        문장
    } else {
        문장
    }
} else {
    if (불리언 표현식) {
        문장
    } else {
        문장
    }
}

```

앞서 조건에 따라 실행되는 문장이 하나라면 중괄호를 생략해도 된다고 했다. 위 코드에서 중첩 조건문의 중괄호를 생략했을 경우 만들어지는 조건문 if else if 조건문이다. if else if 조건문은 중복되지 않는 조건 세 가지 이상을 구분할 때 사용한다.

```

if (불리언 표현식) {
    문장
} else if (불리언 표현식) {
    문장
} else {
    문장
}

```

2) if else if: 중첩 조건문의 중괄호를 생략했을 경우 만들어지는 조건문으로 조건 세 가지 이상을 구분할 때 사용한다.

ex0804.html

```

<script>
    //Date 객체를 선언한다. 현재 시간 측정
    var date = new Date();
    var hours = date.getHours();
    //조건문
    if (hours < 5) {
        alert('잠을 자렴....');
    } else if (hours < 7) {
        alert('준비');
    } else if (hours < 9) {
        alert('출근');
    } else if (hours < 12) {
        alert('빈둥빈둥');
    } else {
        // 여러 가지 업무를 수행합니다.
    }
</script>

```

3) 논리 연산자와 조건문: 논리 연산자도 사용할 있다. 예를 들어 계절을 구분하려면 조건문을 다음과 같이 작성한다.

```

<script>
    var date = new Date();
    var month = date.getMonth() + 1;
    if (3 <= month && month <= 5) {
        alert('봄입니다.');
```

8.6 반복문

컴퓨터는 인간보다 연산 능력이 월등히 뛰어나 작업을 재빠르게 반복할 수 있다. 자바스크립트에서는 배열과 반복문을 사용해 반복처리를 한다. 다음 코드는 경고창을 1,000번 출력한다. 단 인터넷 익스플로러는 경고창 출력을 중간에 종료할 수 없기 때문에 유의해서 실행한다.

```
<script>
  for ( var i =0; i < 1000; i++ ) {
    alert('출력');
  }
</script>
```

• 배열

배열은 변수 여러 개를 한꺼번에 다룰 수 있는 자료형이다. 자바스크립트에는 숫자, 문자열, 불리언, 함수, 객체, 정의되지 않은 자료형까지 총 여섯 개의 자료형이 있다. 배열은 객체의 일종이며 다음과 같이 대괄호([])를 사용해 생성한다. 대괄호 내부의 각 자료는 쉼표로 구분한다.

```
//변수를 선언한다.
var array = [273, 32, 103, 57, 52];
```

배열 내부에 입력된 자료 하나하나를 요소라고 부른다. 다음 코드는 요소가 여섯 개인 배열이다. 배열 내부에 다양한 자료형을 입력할 수 있고 배열 전체를 출력하면 다음처럼 요소가 순서대로 표시된다.

```
<script>
  //변수를 선언한다.
  var array = [273, '문자열', true, function() {}, {}, [32, 103]];
  alert(array);
</script>
```

배열의 각 요소에 접근하려면 배열 이름 다음에 대괄호를 사용하여 순번을 입력한다. 예를 들어 위의 예시 중 true에 접근하려면 array[2]를 입력한다. 배열의 요소에 접근하기 위해 사용하는 대괄호 안의 숫자는 인덱스(index)라고 부르며 0부터 시작한다.

1) 배열 생성하고 접근하기: 다음 코드는 배열을 생성하고 인덱스를 사용해 요소에 접근한다. 인덱스의 시작 숫자에 주의하여 살펴본다.

ex0805.html

```
<script>
  //변수를 선언한다.
  var array = ['가', '나', '다', '라'];

  //배열의 요소를 변경한다.
  array[0] = '윤';

  //요소를 출력한다.
  alert(array[0]);
  alert(array[1]);
  alert(array[2]);
  alert(array[3]);
</script>
```

2) 배열의 length 속성: 배열 변수의 length 속성을 사용하면 배열 요소의 개수를 간단하게 알 수 있다. 다음 코드를 실행하면 배열의 요소 개수인 5를 출력한다. 사실 객체를 배운 이후에 다뤄야 하는 내용이지만 간단한 내용이므로 지금 다뤘다.

```
<script>
  //변수를 선언한다.
  var array = [10, 20, 30, 40, 50];

  //출력한다.
  alert(array.length);
</script>
```

- while 반복문

while 반복문은 가장 기본적인 반복문이다. if 조건문과 형식이 비슷하지만 if 조건문과 달리 불리언 표현식이 참이면 중괄호 안의 문장을 계속 실행한다. 조건이 변하지 않으면 무한히 반복하므로 반드시 조건을 거짓으로 만들 수 있는 문장이 포함되어야 한다.

```
<script>
    //반복을 수행한다.
    while(true) {
        alert('무한반복');
    }
</script>
```

다음 코드는 특정한 숫자를 증가시켜 불리언 표현식을 거짓으로 만들어 반복문을 벗어난다.

ex0806.html

```
<script>
    //변수를 선언한다.
    var i = 0;
    var array = ['가', '나', '다'];

    //반복을 수행한다. I가 배열 원소 개수인 3보다 작은 동안 반복한다.
    while (i < array.length) {
        //출력한다.
        alert(i + '번째 출력: ' + array[i]);
        //탈출을 위해 변수를 더한다.
        i++;
    }
</script>
```

- for 반복문

for 반복문은 원하는 횟수만큼 반복하고 싶을 경우에 사용한다. for 반복문은 맨 먼저 초기식을 실행하고 조건식을 확인한다. 이때 조건식이 거짓이면 반복문을 빠져나가고 참이면 문장과 종결식을 차례로 실행한 후에 다시 조건식이 참인지 확인한다. for 반복문은 이와 같은 일을 계속 반복 진행한다. for 반복문을 각 단계별로 실행하면 다음과 같다.

- 1) 초기식을 비교한다.
- 2) 조건식을 비교한다. 조건이 거짓이면 반복문을 종료한다.
- 3) 문장을 실행한다.
- 4) 종결식을 실행한다.
- 5) 2단계로 이동한다.

굉장히 어렵게 설명했지만 for 반복문은 일반적으로 다음과 같은 형식을 사용하여 특정한 횟수만큼 반복한다.

```
for ( var i = 0; i < length; i++ ) {
}
```

- 1) for 반복문 이용하기: 다음 코드는 while 반복문을 살펴보면서 다루었던 예제와 같은 기능을 수행한다.

ex0807.html

```
<script>
    //변수를 선언한다.
    var array = ['가', '나', '다'];

    //반복을 수행한다.
    for (var i = 0; i < 3; i++) {
        //출력한다.
        alert(i + '번째 출력: ' + array[i]);
    }
</script>
```

2) 외부 요인으로 조건 변경: 대개 for 반복문을 많이 사용하고 while 반복문은 외부 요인으로 조건이 변경되는 경우에 사용한다.

```
<script>
//변수를 선언한다.
var start = new Date().getTime();
var count = 0;

//반복을 수행한다.
while( start + 1000 > new Date().getTime());
    count++;
}

//출력한다.
alert(count + '만큼 반복했습니다.');
```

3) for 반복문을 사용하여 덧셈하기

```
<script>
//변수를 선언한다.
var output = 0;

//반복을 수행한다.
for ( var i = 0; i <= 100; i++) {
    output += i; //100까지 더해야 하므로 <= 연산자를 사용한다.
}

//출력한다.
alert(output);
</script>
```

8.7 함수

자바스크립트에서 함수는 코드의 집합이다. 지금까지는 함수를 사용하는 방법만 살펴보았다. 이제 이 함수를 직접 만드는 방법을 알아보자.

• 선언과 호출 및 실행 우선순위

1) 선언과 호출 : 함수는 코드의 집합을 나타내는 자료형이다. 1과 같이 함수 이름을 입력하지 않고 만드는 익명 함수 생성 방법과 2와 같이 함수 이름을 입력해서 만드는 선언적 함수 생성 방법이 있다.

방법	표현
1. 익명 함수	function() { }
2. 선언적 함수	function 함수() { };

2) 익명 함수 선언하기: 아래 코드는 두 문장을 포함하는 함수를 생성하고 출력한다.

```
ex0808.html

<script>
//함수를 선언한다.
var 함수 = function() {
    alert('함수_01');
    alert('함수_02');
}

//출력한다.
alert(typeof(함수) + ' : ' + 함수);
</script>
```

3) 선언적 함수 선언하기: 아래코드는 위 코드의 익명 함수를 선언적 함수로 바꾼 것이다.

ex0809.html

```
<script>
    //함수를 선언한다.
    function() {
        alert('함수_01');
        alert('함수_02');
    }

    //출력한다.
    alert(typeof(함수) + ' : ' + 함수);
</script>
```

익명함수와 선언적 함수 모두 많이 사용되므로 용어와 생성 방법을 꼭 기억해두자. 이렇게 만들어진 함수는 alert() 함수를 사용했던 것처럼 함수 이름 뒤에 괄호를 사용해 실행할 수 있다. 이렇게 함수를 실행하는 것을 "함수를 호출한다."라고 표현한다.

```
//함수를 호출한다.
함수();
```

4) 실행 우선순위

함수도 변수이므로 다음과 같이 사용할 경우 가장 마지막에 입력된 값이 저장된다. 따라서 코드를 실행하면 문자열 함수 '함수_B'를 출력한다.

ex0810.html

```
<script>
    //함수를 선언한다.
    var 함수 = function() { alert('함수_A'); };
    var 함수 = function() { alert('함수_B'); };

    //함수를 호출한다.
    함수();
</script>
```

하지만 선언적 함수와 익명 함수를 함께 사용할 경우에는 실행 순서가 달라진다. 자바스크립트는 모든 코드를 읽기 전에 선언적 함수를 먼저 읽는다. 따라서 선언적 함수가 익명 함수 뒤에 있지만 먼저 읽기 때문에 다음을 실행하면 나중에 읽은 익명 함수를 실행한다.

ex0811.html

```
<script>
    //함수를 선언한다.
    var 함수 = function() { alert('함수_A'); };
    function 함수() { alert('함수_B'); };

    //함수를 호출한다.
    함수();
</script>
```

• 매개변수와 반환값

alert() 함수의 괄호 안에 문자열 또는 숫자 등을 집어넣었다. 이렇게 함수의 괄호 안에 집어넣는 것을 매개변수라고 표현한다.

```
//함수를 호출한다.
alert('매개변수');
```

시간과 관련된 정보를 추출할 때는 다음과 같이 함수의 실행 결과를 변수에 입력했다. 이렇게 함수를 실행한 결과를 반환값(리턴값)이라고 부른다.

```
var minutes = date.getMinute();
var seconds = date.getSeconds();
```

매개변수와 반환값을 가지는 함수는 다음과 같은 형식으로 작성한다. 단 모든 함수에 매개변수와 반환값을 사용해야 하는 것은 아니다.

```
function 함수이름( 매개변수, 매개변수, 매개변수 ) {  
    //함수 코드  
    //함수 코드  
    //함수 코드  
    return 반환값;  
}
```

다음은 매개변수로 x를 넣으면 x^2 를 반환하는 함수를 정의하고 리턴된 값을 출력하는 예제 코드이다.

ex0809.html

```
<script>  
    //함수를 선언한다.  
    function f(x) {  
        return x * x;  
    }  
    //함수를 호출한다.  
    alert(3);  
</script>
```

- 콜백함수

매개 변수를 전달되는 함수를 콜백함수라고 부른다. 다음은 callTenTimes()와 callback() 함수를 선언했다. callback()은 문자열 '함수 호출'을 출력하는 간단한 함수이다. 그리고 callTenTimes()는 매개변수로 전달된 함수를 열 번 호출하는 함수이다.

ex0810html

```
<script>  
    //함수를 선언한다.  
    function callTenTimes(callback) {  
        //10회 반복한다.  
        for (var i = 0; i < 10; i++) {  
            callback(); //매개변수로 전달한 함수를 호출한다.  
        }  
    }  
    //변수를 선언한다.  
    var callback = function () {  
        alert('함수 호출');  
    };  
    callTenTimes(callback); //함수를 호출한다.  
</script>
```

콜백함수는 익명 함수로 사용하는 경우가 많다. 다음은 앞서 실행한 코드와 결과가 같다. 매개변수에 익명 함수를 곧바로 입력할 수도 있다.

ex0810html

```
<script>  
    //함수를 선언한다.  
    function callTenTimes(callback) {  
        //10회 반복한다.  
        for (var i = 0; i < 10; i++) {  
            callback(); //매개변수로 전달한 함수를 호출한다.  
        }  
    }  
    //함수를 호출한다.  
    callTenTimes(function () {  
        alert('함수 호출');  
    });  
</script>
```

8.8 객체

객체는 여러 개의 자료형을 한 번에 저장하는 자료형이다. 배열은 인덱스를 기반으로 자료를 저장하지만 객체는 키를 기반으로 자료를 저장한다.

• 객체 개요

객체를 살펴보기 전에 배열과 관련된 기본 내용을 다시 정리해보겠다. 배열은 다음과 같은 방법으로 생성한다.

```
<script>
  //배열을 선언한다.
  var array = ['사과', '바나나', '망고', '딸기'];
</script>
```

배열에는 인덱스와 요소가 있다. 각 배열 요소를 사용하려면 다음과 같이 배열 이름 뒤에 인덱스를 입력해 요소에 접근한다.

```
array[0] → '사과'
array[1] → '바나나'
array[2] → '망고'
array[3] → '딸기'
```

배열은 객체를 기반으로 만들어졌다. 따라서 배열과 객체는 상당히 유사하다. 다른점이 있다면 배열은 요소에 접근할 때 인덱스를 사용하지만 객체는 키를 사용하는 것이다. 우선 객체를 만드는 방법을 알아보자.

```
<script>
  //객체를 선언한다.
  var product = {
    제품명: '7D 건조 망고',
    유형: '당절임',
    성분: '망고, 설탕, 메타주아황산나트륨, 치자황색소',
    원산지: '필리핀'
  }
</script>
```

이처럼 생성한 객체의 키와 속성은 다음과 같다.

키	속성
제품명	7D 건조 망고
유형	당절임
성분	망고, 설탕, 메타주아황산나트륨, 치자 황색소
원산지	필리핀

객체의 요소에 접근하는 것도 배열과 비슷하다. 다음과 같이 객체 뒤에 대괄호를 사용해 키를 입력하면 객체의 속성에 접근할 수 있다.

```
product[제품명] → '7D 건조 망고'
product[유형] → '당절임'
product[성분] → '망고, 설탕, 메타주아황산나트륨, 치자황색소'
product[원산지] → '필리핀'
```

또는 다음과 같은 방법으로 객체의 속성에 접근한다. 깔끔하고 입력하기도 쉽기 때문에 이 방법을 더 많이 사용한다.

```
product.제품명 → '7D 건조 망고'
product.유형 → '당절임'
product.성분 → '망고, 설탕, 메타주아황산나트륨, 치자황색소'
product.원산지 → '필리핀'
```

for in 반복문을 사용하면 객체의 요소를 하나하나 살펴볼 수 있다. 다음과 같은 형식으로 for in 반복문을 작성하여 객체를 순환한다. 코드를 실행하면 객체의 요소가 하나씩 차례로 출력된다.

```
ex0811.html
```

```
<script>
//객체를 선언한다.
var product = {
    제품명: '7D 건조 망고',
    유형: '당절임',
    성분: '망고, 설탕, 메타중아황산나트륨, 치자황색소',
    원산지: '필리핀'
};
//출력한다.
for (var i in product) {
    alert(i + ':' + product[i]);
}
</script>
```

• 속성과 메서드

배열 내부에 있는 값 하나하나를 요소(element)라고 한다. 하지만 객체 내부에 있는 값 하나 하나는 속성(property)라고 한다. 배열의 요소처럼 객체의 속성에도 다음과 같이 다양한 자료형을 입력할 수 있다.

```
//객체를 선언한다.
var object = {
    number: 273,
    string: 'rintiantta',
    boolean: true,
    array: [52, 273, 103, 32],
    method: function() {
    }
};
```

객체의 속성 중 자료형이 함수인 속성을 특별히 메서드(method)라고 부른다. 아래 코드에서 개체 person에 name과 eat 속성이 있다. 이중 eat 속성의 자료형이 함수이므로 특별히 eat() 메서드라고 부른다.

```
//객체를 선언한다.
var person = {
    name: '홍길동',
    eat: function (food) {
        alert(food + '을/를 먹습니다.');
```

```
    }
};
```

```
//메서드를 호출한다.
```

```
person.eat('밥');
```

객체에 있는 함수 속성을 메서드에서 사용하고 싶을 때는 자신이 가지고 있는 속성임을 분명하게 표시해야 한다. 자기 자신이 가지고 있는 속성이라는 것을 표시할 때는 this 키워드를 사용한다. 지금까지 살펴본 내용을 정리하면 다음과 같다.

```
<script>
//객체를 선언한다.
var person = {
    name: '홍길동',
    eat: function(food) {
        //eat()메서드와 name 속성 모두 person 객체 안에 있다. 같은 객체에 있는 속성을 이용하므로 this 키워드를 사용했다.
        alert(this.name + '이' + food + '을/를 먹습니다.');
```

```
    }
}
```

```
// 메서드를 호출한다.
```

```
person.eat('밥');
```

```
</script>
```


9장 문서 객체 모델

웹 브라우저가 HTML 파일을 분석하고 표시하는 방식을 문서 객체 모델(DOM: Document Object Model)이라고 한다.

9.1 웹 페이지 생성 순서

- 웹 브라우저는 HTML 코드를 위에서 아래로 실행한다.
- script를 읽는 시점에 태그가 생성되지 않았다면 스크립트 태그를 태그 아래에 삽입하거나 이벤트 기능을 사용한다.

다음 코드에서는 HTML 페이지 내부에 alert() 함수를 사용하여 중간중간 실행 흐름을 끊었다.

ex0901.html

```
<html>
<head>
  <script>alert('Process-0')</script>
</head>
<body>
  <h1>Process-1</h1>
  <script>alert('Process-2')</script>
  <h1>Process-2</h1>
  <script>alert('Process-3')</script>
</body>
</html>
```

- 문서 객체 조작 방법 익히기

- 1) 문서 객체 사용 예리: script 태그를 읽는 시점에 h1과 h2 태그는 생성되지 않았으므로 순서대로 코드를 작성하면 예리가 발생한다.

ex0902.html

```
<html>
<head>
  <script>
    //h1 태그의 배경 색상을 변경한다.
    document.querySelector('h1').style.backgroundColor = 'red';
    //h1 태그의 배경 색상을 변경한다.
    document.querySelector('h2').style.color = 'red';
  </script>
</head>
<body>
  <h1>Process-1</h1>
  <h2>Process-2</h2>
</body>
</html>
```

- 2) 해결방법1: script 태그 위치이동: 가장 간단한 해결 방법은 다음과 같이 스크립트 태그를 제목 태그들 아래에 삽입하는 것이다. 하지만 이 경우 HTML 페이지가 방대해졌을 때 유지보수가 어렵다.

```
<html>
<body>
  <h1>Process-1</h1>
  <h2>Process-2</h2>
  <script>
    //h1 태그의 배경 색상을 변경한다.
    document.querySelector('h1').style.backgroundColor = 'red';
    //h1 태그의 배경 색상을 변경한다.
    document.querySelector('h2').style.color = 'red';
  </script>
</body>
</html>
```

3) 해결방법2: 이벤트 활용: 다음과 같이 다음절에서 배울 이벤트라는 기능을 사용한다.

```
<head>
  <script>
    window.onload = function() {
      //h1 태그의 배경 색상을 변경한다.
      document.querySelector('h1').style.backgroundColor = 'red';
      //h1 태그의 배경 색상을 변경한다.
      document.querySelector('h2').style.color = 'red';
    };
  </script>
</head>
<body>
  <h1>Process-1</h1>
  <h2>Process-2</h2>
</body>
```

9.2 문서 객체 선택

이미 존재하는 HTML 태그를 자바스크립트에서 문서 객체로 변환하는 것을 문서 객체를 선택한다고 한다. 문서 객체를 선택하면 자바스크립트를 사용하여 실행 중에 내부 글자를 변경하거나 스타일을 변경할 수 있다.

• 한 개의 문서 객체 선택

문서 객체를 선택하는 가장 기본적인 방법은 아래의 메서드를 사용하는 것이다. 아래의 `querySelector()` 메서드는 HTML5에서 추가되었다.

메서드 이름	설명
<code>document.getElementById(아이디)</code>	아이디를 사용해 문서 객체 선택
<code>document.querySelector(선택자)</code>	선택자를 사용해 문서 객체 선택

1) `getElementById()` 메서드 사용하기: id속성을 사용해 문서 객체를 선택하고 스타일과 내부 글자를 조작한다.

ex0903.html

```
<head>
  <script>
    window.onload = function () { //이벤트를 연결한다.

      var header = document.getElementById('header'); //문서 객체를 선택한다.
      header.style.color = 'orange';
      header.style.background = 'red';
      header.innerHTML = 'From JavaScript'; //innerHTML은 태그의 내부를 의미하는 속성이다.
    };
  </script>
</head>
<body>
  <h1 id="header">Header</h1>
</body>
```

2) `QuerySelector()` 메서드 사용하기: id 속성 대신에 선택자를 입력한다.

```
<script>
  window.onload = function () {
    var header = document.querySelector('h1');
    header.style.color = 'orange';
    header.style.background = 'red';
    header.innerHTML = 'From JavaScript';
  };
</script>
```

- 여러 개의 문서 객체 선택

여러 개의 문서 객체를 선택할 때는 아래의 메서드를 사용한다.

메서드 이름	설명
document.getElementsByName(이름)	name 속성으로 여러 개의 문서 객체 선택
document.getElementsByClassName(클래스)	class 속성으로 여러 개의 문서 객체 선택
document.querySelectorAll(선택자)	선택자로 여러 개의 문서 객체 선택

다음 예제는 문서 객체를 querySelectorAll() 메서드를 이용하여 문서 객체를 여러 개 선택하여 조작할 수 있다.

ex0904.html

```
<head>
  <script>
    window.onload = function () { //이벤트를 연결한다.
      var headers = document.querySelectorAll('hl'); //문서 객체를 선택한다.
      for (var i = 0; i < headers.length; i++) {
        var header = headers[i]; //변수를 선언한다.
        //문서 객체를 조작한다.
        header.style.color = 'orange';
        header.style.background = 'red';
        header.innerHTML = 'From JavaScript';
      }
    };
  </script>
</head>
<body>
  <hl>Header</hl>
  <hl>Header</hl>
  <hl>Header</hl>
</body>
```

9.3 문서 객체 조작

- 글자 조작

문서 객체 내부에 있는 글자를 조작할 때는 아래의 속성을 사용한다. 이 중에서 textContent 속성은 익스플로러 9버전 이상에 사용할 수 있다.

속성 이름	설명
textContent	일반 글자 형식의 문서 객체 내부 글자
innerHTML	HTML 형식의 문서 객체 내부 글자

- 1) 내부 글자 변경하기: textContent 속성을 사용하여 문서 객체 내부의 글자를 변경할 수 있다. 코드를 실행하면 입력한 글자가 body태그에 바로 들어간다.

ex0905.html

```
<head>
  <script>
    window.onload = function () { //이벤트를 연결한다.
      var output = ''; //변수를 선언한다.
      for (var i = 0; i < 10; i++) {
        output += '<hl>Header - ' + i + '</hl>';
      }
      document.body.textContent = output; //문서 객체 내부의 글자를 변경한다.
    };
  </script>
</head>
<body>
</body>
```

2) 태그 내부 글자 변경하기: 다음은 innerHTML 속성으로 생성한 문자열을 넣었다. textContent 속성과 달리 HTML 태그가 적용된다.

ex0906.html

```
<head>
  <script>
    //이벤트를 연결한다.
    window.onload = function () {
      //변수를 선언한다.
      var output = '';
      for (var i = 0; i < 10; i++) {
        output += '<h1>Header - ' + i + '</h1>';
      }
      //문서 객체 내부의 글자를 변경한다.
      document.body.innerHTML = output;
    };
  </script>
</head>
<body>
</body>
```

- 스타일 조작

태그의 스타일 속성을 조작하는 것은 이전 절에서 잠시 살펴보았다. 다음과 같이 문서 객체의 style 속성을 변경하면 된다.

```
var heaer = document.getElementById('header');
header.style.color = 'red';
```

위 방법을 적용하여 아래처럼 스타일시트의 스타일 속성을 자바스크립트의 스타일 식별자로 변경할 수 있다.

스타일시트의 스타일 속성	자바스크립트의 스타일 식별자
background-image	backgroundImage
background-color;	backgroundColor
box-sizing	boxSizing
list-style	listStyle

다음 코드를 실행하면 256개의 div 태그가 위에서 아래로 그레이디언트를 형성한다.

ex0907.html

```
<head>
  <script>
    window.onload = function () { //이벤트를 연결한다.
      var output = ''; //문서 객체를 추가한다.
      for (var i = 0; i < 256; i++) {
        output += '<div></div>';
      }
      document.body.innerHTML = output;
      var divs = document.querySelectorAll('div'); //문서 객체를 선택한다.
      for (var i = 0; i < divs.length; i++) {
        var div = divs[i]; //변수를 선언한다.
        div.style.height = '2px'; //스타일을 적용한다.
        div.style.background = 'rgb(' + i + ', ' + i + ', ' + i + ')';
      }
    };
  </script>
</head>
<body>
</body>
```

- 속성 조작

이제 문서 객체의 속성을 조작하는 방법을 알아보겠다. 문서 객체의 속성을 조작할 때는 아래의 메서드를 사용한다.

메서드 이름	설명
setAttribute(속성이름, 속성값)	속성 지정
getAttribute(속성이름)	속성 추출

1) img 태그 속성 조작하기: 간단한 속성 조작 방법을 사용해 img 태그의 src, width, height 속성을 변경한다. img 태그에는 id 속성만을 부여했다.

```
ex0908.html

<html>
<head>
  <script>
    //이벤트를 연결한다.
    window.onload = function () {
      //변수를 선언한다.
      var image = document.getElementById('image');

      //속성을 변경한다.
      image.src = 'http://placeholder.it/300x200';
      image.width = 300;
      image.height = 200;
    };
  </script>
</head>
<body>
  <img id="image" />
</body>
</html>
```

2) body 태그의 속성 조작하기: body 태그에 data-custom 속성을 지정한 후 속성을 다시 추출해서 출력한다.

```
<html>
<head>
  <script>
    //이벤트를 연결한다.
    window.onload = function () {
      //속성을 지정한다.
      document.body.setAttribute('data-custom', 'value');

      //속성을 추출한다.
      var dataCustom = document.body.getAttribute('data-custom');
      alert(dataCustom);
    };
  </script>
</head>
<body>

</body>
</html>
```

코드를 실행하고 요소 검사를 사용해서 살펴보면 아래처럼 body 태그에 data-custom 속성이 추가된 것을 확인할 수 있다.

```
<html>
  <head>...</head>
  <body data-custom="value">
</html>
```

지금까지 다뤘던 예제들이 정적이었다면 문서 객체를 정리해보는 의미로 1초 단위로 현재 시간을 표시하는 시계를 만들어보았다. setInterval() 함수를 사용해 1초마다 clock 문서 객체의 innerHTML 속성을 현재 시간으로 변경한다. setInterval() 함수는 단위 시간마다 함수를 실행한다.

ex0909.html

```
<head>
  <script>
    //이벤트를 연결한다.
    window.onload=function(){
      //문서 객체를 선택한다.
      var clock = document.getElementById('clock');

      //1초마다 함수를 실행한다.
      setInterval(function(){
        var now=new Date();
        clock.innerHTML = now.toString();
      }, 1000);
    };
  </script>
</head>
<body>
  <h1 id="clock"><h1>
</body>
```

9.4 이벤트

이벤트는 키보드를 누르거나 마우스를 클릭하는 것과 같이 어떠한 현상이 프로그램에게 영향을 미치는 것을 의미한다. 이벤트는 사용자가 직접 발생시킬 수도 있고 응용 프로그램에서 자체적으로 발생시킬 수도 있다.

코드에서 onload를 이벤트 속성이라고 부른다. 그리고 on을 제외한 load를 이벤트이름 또는 이벤트 타입이라고 부른다. 그리고 이벤트 속성에 넣는 함수를 이벤트 리스너 또는 이벤트 핸들러라고 부른다.

• 이벤트 모델

문서 객체에 이벤트를 연결하는 방법을 이벤트 모델이라고 부른다.

1) 인라인 이벤트 모델 사용하기: button 태그 내부에 onclick 속성을 사용하여 button 태그를 클릭하면 경고창을 출력한다.

ex0910.html

```
<html>
<head>
  <title>Inline Event</title>
</head>
<body>
  <button onclick="alert('click')">버튼</button>
</body>
</html>
```

HTML 태그의 문자열에서 on으로 시작하는 속성은 모두 이렇게 이벤트와 관련된 속성이다. 이벤트 속성의 종류는 아래와 같다.

onblur	onfocus	onfocusin	onfocusout	onload
onresize	onscroll	onunload	onclick	ondblclick
onmousedown	onmouseup	onmousemove	onmouseover	onmouseout
onmouseenter	onmouseleave	onchange	onselect	onsubmit
onkeydown	onkeypress	onkeyup	onerror	

인라인 코드 내부에서 세미콜론으로 문장을 구분하면 여러 행을 입력할 수 있지만 HTML 페이지가 너무 지저분해진다. 그래서 인라인 이벤트 모델을 사용할 경우에는 script 태그 내부에 함수를 선언한 후 인라인 이벤트 속성 내부에서 해당 함수를 실행하는 형태를 취한다.

2) script 태그를 이용해 인라인 이벤트 모델 사용하기: 인라인 이벤트 모델에서 script 태그 내부에 있는 함수를 호출한다.

```
<html>
<head>
  <title>Inline Event</title>
  <script>
    function buttonClick() {
      alert('click');
    }
  </script>
</head>
<body>
  <button onclick="buttonClick()">버튼</button>
</body>
</html>
```

3) 고전 이벤트 모델: 문서 객체의 이벤트 속성에 함수를 지정하여 작성한다. 이때 지정된 함수를 이벤트 핸들러 또는 이벤트 리스너라고 부른다.

ex0911.html

```
<html>
<head>
  <title>Traditional Event</title>
  <script>
    //이벤트를 연결한다.
    window.onload = function () {
      //문서 객체를 선택한다.
      var button = document.getElementById('button');

      //이벤트를 연결한다. (이벤트 리스너)
      button.onclick = function () {
        alert('click');
      };
    };
  </script>
</head>
<body>
  <button id="button">버튼</button>
</body>
</html>
```

4) 이벤트 발생 객체: 이벤트 리스너 내부에서 this 키워드를 사용하면 이벤트를 발생한 자기 자신을 의미한다. 따라서 다음 코드처럼 이벤트 리스너를 사용하고 내부에서 this 키워드의.textContent 속성을 사용하면 자기 자신의 문자를 변경한다.

ex0912.html

```
<html>
<head>
  <title>Traditional Event</title>
</head>
<body>
  <button id="button">버튼-</button>
  <script>
    //이벤트를 연결한다.
    document.getElementById("button").onclick=function(){
      this.setContent = this.textContent + '★';
    };
  </script>
</body>
</html>
```

- 이벤트 객체

현실 세계에서 사건(이벤트)이 발생하면 누가, 언제, 어디서, 무엇을, 어떻게, 왜에 해당하는 정보를 알 수 있다. 자바스크립트도 발생한 이벤트와 관련된 정보를 이벤트 객체를 사용하여 알아 낼 수 있다. 아래 예제 코드는 이벤트 리스너의 첫 번째 매개변수로 이벤트 객체가 들어온다.

```
<head>
  <title>Event Object</title>
  <script>
    window.onload = function (event) { //이벤트 객체
      alert(event);
    };
  </script>
</head>
<body>

</body>
```

인터넷 익스플로러 9버전부터는 두가지 방법을 모두 사용할 수 있다. 모든 브라우저에서 이벤트 객체를 사용하려면 다음과 같이 입력한다.

```
<script>
  window.onload = function (e) {
    var event;
    if (e) {
      //매개변수 e가 있는 경우: 현대 웹 브라우저
      event = e;
    } else {
      //매개변수 e가 없는 경우: 구 버전의 인터넷 익스플로러
      event = window.event;
    }
  };
</script>
```

- 기본 이벤트의 제거

HTML 태그는 기본적으로 몇 가지 이벤트를 가지고 있다. a 태그를 클릭하면 href 속성에 입력한 위치로 이동하고 form 태그로 생성한 제출 버튼을 클릭하면 자동으로 입력 양식을 제출한다. 이를 기본 이벤트라고 부른다. 웹 페이지는 가끔 이러한 기본 이벤트를 막아야 하는 경우가 있다. 다음 예제는 a 태그의 click 이벤트 리스너의 반환값에 false를 입력하여 이동을 막았다.

ex0913.html

```
<html>
<head>
  <title>Traditional Event - Default Event</title>
  <script>
    //이벤트를 연결한다.
    window.onload = function () {
      //문서 객체를 선택한다.
      var button = document.getElementById('button');
      //이벤트를 연결한다.
      button.onclick = function () {
        //기본 이벤트를 제거한다.
        return false;
      };
    };
  </script>
</head>
<body>
  <a id="button" href="http://hanb.co.kr">버튼</a>
</body>
</html>
```


10장 jQuery 라이브러리

jQuery는 라이브러리는 모든 웹 브라우저에서 동작하는 클라이언트용 자바스크립트 라이브러리이다. 무료로 사용 가능한 오픈 소스 라이브러리이며 사용 목적은 다음과 같다.

- 쉬운 문서 객체 모델과 관련된 처리 구현
- 쉬운 일관된 이벤트 연결 구현
- 쉬운 시각적 효과 구현
- 쉬운 Ajax 응용 프로그램 개발

10.1 jQuery 라이브러리 설정

- 1) 방법1: <http://jquery.com>에 접속한 후 jquery 라이브러리 파일을 다운받아 외부 자바스크립트 파일로 추가한다.
- 2) 방법2: CDN 호스트를 사용하는 방법으로 다운로드 페이지에 들어가 script 태그 부분을 복사해 사용한다. CDN은 파일을 여러 서버에 분산시키고 사용자가 접속하는 지역과 가장 가까운 곳의 서버에서 파일을 전송하는 기술이다.

다음 예제는 jQuery를 사용하도록 기본 설정하고 window.onload 이벤트와 같은 기능을 수행하도록 작성하였다.

ex1001.html

```
<html>
<head>
  <title>jQuery Basic</title>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {

    });
  </script>
</head>
<body>
  ...
</body>
</html>
```

10.2 문서 객체 선택

jQuery 라이브러리는 다음과 같은 형식으로 사용한다.

`$(선택자).메서드(매개변수, 매개변수)`

다음 예제는 문서 객체를 선택하여 스타일을 지정한다. `querySelector()`메서드와 달리 문서 객체 여러 개를 한 번에 선택한다.

ex1002.html

```
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    //이벤트를 연결한다.
    $(document).ready(function () {
      //스타일 속성을 변경한다.
      $('hl').css('color', 'red');
      $('hl').css('background', 'black');
    });
  </script>
</head>
<body>
  <hl>Header</hl>
  <hl>Header</hl>
  <hl>Header</hl>
</body>
```

10.3 문서 객체 조작

jQuery 라이브러리를 사용하면 문서 객체도 굉장히 쉽게 조작할 수 있다. 문서 객체를 조작하는 메서드의 사용 방법은 서로 비슷하므로 속성 조작 메서드에서 대부분의 내용을 소개한다. 속성조작과 관련된 내용을 주의 깊게 살펴보기 바란다.

- 속성 조작

jQuery 라이브러리를 사용해서 문서 객체의 속성을 조작할 때는 `attr()` 메서드를 사용한다. `attr()` 메서드는 매개변수를 넣는 방법에 따라 속성을 지정할 수도 있고 속성을 추출할 수도 있다.

1) 속성 추출: 다음과 같이 `attr()` 메서드에 매개변수를 하나만 입력하면 해당 속성을 추출한다. jQuery 라이브러리는 문서 객체 여러 개를 한 번에 선택한다. 여러 개의 문서 객체를 선택하고 속성을 추출하면 첫 번째 문서 객체의 속성을 추출한다.

```
//img 태그의 src 속성을 추출한다.  
var src = $('img').attr('src');
```

다음은 jQuery 라이브러리를 사용해 `script` 태그를 선택하고 `src` 속성을 추출한다.

```
ex1003.html  
  
<head>  
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>  
  <script>  
    //이벤트를 연결한다.  
    $(document).ready(function () {  
      //속성을 추출한다.  
      var src = $('script').attr('src');  
      //출력한다.  
      alert(src);  
    });  
  </script>  
</head>  
<body>  
</body>
```

2) 속성 지정: 문서 객체의 속성은 세 가지 방법으로 지정할 수 있다. 첫 번째 방법은 첫 번째 매개변수에 속성의 이름을 입력하고 두 번째 매개변수에 속성 값을 입력하는 것이다.

```
//img 태그의 src 속성을 지정한다.  
$('img').attr('src', 'http://placeholder.it/300x200');
```

다음은 `attr()` 메서드를 사용해 문서 객체의 속성을 지정한다. `img` 태그를 선택하고 `alt`속성, `src`속성, `width`속성을 변경한다.

```
ex1003.html  
  
<head>  
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>  
  <script>  
    //이벤트를 연결한다.  
    $(document).ready(function () {  
      //속성을 지정한다.  
      $('img').attr('alt', 'jQuery 라이브러리를 사용한 속성 지정');  
      $('img').attr('src', 'http://placeholder.it/100x100');  
      $('img').attr('width', '100');  
    });  
  </script>  
</head>  
<body>  
  <img /><img /><img />  
</body>
```

- 스타일 조작

jQuery 라이브러리를 사용해 스타일을 조작할 때는 `css()` 메서드를 사용한다. `css()` 메서드는 `attr()` 메서드와 같은 방법으로 사용한다. 하지만 속성과 달리 하이픈(-)이 들어가는 스타일 속성은 다음과 같은 두가지 방법으로 입력해야 한다.

```
$('#body').css('backgroundColor');
$('#body').css('background-color');
```

1) 문서객체 스타일 조작하기: 다음 예제는 `css()` 메서드의 두 번째 매개변수에 속성값을 입력하여 태그의 스타일 속성을 조작한다. 코드를 생략하면 모든 `.box` 태그에 스타일이 적용된다.

ex1004.html

```
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {
      //스타일을 적용한다.
      $('.box').css('float', 'left');
      $('.box').css('margin', 10);
      $('.box').css('width', 100);
      $('.box').css('height', 100);
      $('.box').css('backgroundColor', 'red');
    });
  </script>
</head>
<body>
  <div class="box"></div>
  <div class="box"></div>
  <div class="box"></div>
</body>
</html>
```

2) `css()` 메서드의 매개변수에 객체를 넣은 형태로도 스타일을 조작한다.

ex1005.html

```
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {
      //스타일을 적용한다.
      $('.box').css({
        float: 'left',
        margin: 10,
        width: 100,
        height: 100,
        backgroundColor: 'red'
      });
    });
  </script>
</head>
<body>
  <div class="box"></div>
  <div class="box"></div>
  <div class="box"></div>
</body>
</html>
```

3) css() 메서드의 매개변수에 함수를 입력해 속성을 지정한다.

ex1006.html

```
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {
      //문서 객체를 추가한다.
      var output = '';
      for (var i = 0; i < 256; i++) {
        output += '<div> < /div>';
      }
      document.body.innerHTML = output;

      //스타일을 적용한다.
      $('div').css({
        height: 2,
        backgroundColor: function (i) {
          return 'rgb(' + i + ',' + i + ',' + i + ')';
        }
      });
    });
  </script>
</head>
<body>
  <div class="box"></div>
  <div class="box"></div>
  <div class="box"></div>
</body>
</html>
```

• 글자 조작

jQuery 라이브러리로 문서 객체 내부의 글자를 조작할 때는 아래의 메서드를 사용한다.

메서드 이름	설명
html()	문서 객체 내부의 HTML 태그 조작
text()	문서 객체 내부의 글자 조작

1) tex() 메서드로 내부 글자 변경하기: html()메서드를 사용하면 글자에 태그가 적용되지만 text() 메서드는 일반 글자처럼 출력된다.

ex1007.html

```
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {
      //내부 글자를 변경한다.
      $('h1:nth-child(1)').text('<a href="#">text()</a>');
      $('h1:nth-child(2)').html('<a href="#">html()</a>');
    });
  </script>
</head>
<body>
  <h1>Header - 0</h1>
  <h1>Header - 1</h1>
</body>
</html>
```

2) html() 메서드로 내부 글자 변경하기: text() 메서드를 사용하면 선택된 모든 문서 객체를 추출하고 html() 메서드는 지금까지 살펴본 메서드처럼 첫 번째 위치한 문서 객체의 글자를 추출한다.

ex1008.html

```
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {
      //변수를 선언한다.
      var text = $('hl').text();
      var html = $('hl').html();
      //출력한다.
      alert("text: " + text);
      alert("html: " + html);
    });
  </script>
</head>
<body>
  <hl>Header - 0</hl>
  <hl>Header - 1</hl>
</body>
```

• 클래스 조작

jQuery 라이브러리는 클래스를 추가하고 제거하는데 attr() 메서드 대신 클래스 조작 메서드를 별도로 제공한다.

메서드 이름	설명
addClass()	클래스를 추가 한다.
removeClass()	클래스를 제거 한다.
toggleClass()	클래스 전환. 메서드는 클래스가 있으면 제거하고 없으면 추가한다.

hover() 메서드는 첫 번째 매개변수에 마우스 커서를 올렸을 때 실행할 것을 입력하고 두 번째 매개변수에 마우스 커서를 내렸을 때 실행할 것을 입력한다. 코드를 실행하고 마우스 커서를 div 태그로 만든 도형 위에 올리거나 내리면 class 속성이 바뀌어 모양이 변경된다.

ex1009.html

```
<head>
  <style>
    #box {
      width: 100px; height: 100px;
      background-color: red; }
    #box.hover {
      background-color: blue;
      border-radius: 50px; }
  </style>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {
      $('#box').hover(function () {
        $('#box').addClass('hover'); //스타일을 변경한다.
      }, function () {
        $('#box').removeClass('hover'); //스타일을 변경한다.
      });
    });
  </script>
</head>
<body>
  <div id="box"></div>
</body>
```

10.4 이벤트

이전 장에서 순수하게 자바스크립트만 사용해 이벤트를 연결하는 방법을 살펴보았다. 웹 브라우저마다 조금씩 차이가 있으므로 어렵게 느꼈을 수도 있다. 하지만 jQuery 라이브러리를 사용하면 손쉽게 모든 웹 브라우저에서 동작하는 이벤트를 생성할 수 있다.

- 간단한 이벤트 연결

jQuery 라이브러리는 아래와 같은 간단한 이벤트 연결 메서드를 제공한다.

blur	focus	focusin	focusout	load
resize	scroll	unload	click	dblclick
mousedown	mouseup	mousemove	mouseover	mouseout
mouseenter	mouseleave	change	select	submit
keydown	keypress	keyup	error	ready

간단한 방식으로 이벤트를 연결할 때는 다음과 같은 방법을 사용한다.

```
$('#selector').method(function (event) { } );
```

다음 코드는 click 이벤트를 연결하는 예제이다. 코드를 실행하고 h1 태그로 생성한 Click을 클릭하면 경고창이 출력된다.

```
ex1010.html

<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () { //이벤트를 연결한다.
      $('h1').click(function () { //이벤트를 연결한다.
        alert('클릭!');
      });
    });
  </script>
</head>
<body>
  <h1>Click</h1>
</body>
```

hover() 메서드의 첫 번째 매개변수에는 mouseenter 이벤트 리스너를 입력하고 두 번째 매개변수는 mouseleave 이벤트 리스너를 입력한다. 코드를 실행하고 글자 위에 마우스 커서를 올리거나 내리면 색상이 변경된다.

```
ex1011.html

<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () { //이벤트를 연결한다.
      $('h1').hover(function () { //이벤트를 연결한다.
        $(this).css({ //색상을 변경한다.
          background: 'red',
          color: 'white'
        });
      }, function () {
        $(this).css({ //색상을 제거한다.
          background: '',
          color: ''
        });
      });
    });
  </script>
</head>
<body>
  <h1>Click</h1>
</body>
```

- 이벤트 연결 기본

jQuery 라이브러리로 일반 이벤트를 연결할 때는 on() 메서드를 사용하고 이벤트를 제거할 때는 off() 메서드를 사용한다.

메서드 이름	설명
on()	이벤트 연결
off()	이벤트 제거

1) 첫 번째 방법으로 이벤트 연결하기: 메서드 체이닝을 사용해 문서 객체에 여러 개의 이벤트를 적용한다. 코드를 실행하고 사각형에 이벤트를 적용하면 색상이 변경된다. 도형을 클릭하거나 마우스 커서를 움직여본다.

ex1012.html

```
<head>
<script src="http://code.jquery.com/jquery-1.9.1.js"></script>
<script>
    $(document).ready(function () {
        $('#box').css({ //스타일 변경 및 이벤트 연결
            width: 100,
            height: 100,
            background: 'orange'
        }).on('click', function () {
            $(this).css('background', 'red');
        }).on('mouseenter', function () {
            $(this).css('background', 'blue');
        }).on('mouseleave', function () {
            $(this).css('background', 'orange');
        });
    });
</script>
</head>
<body>
    <div id="box"></div>
</body>
```

2) 두 번째 방법으로 이벤트 연결하기: 문서 객체를 조작하는 방법과 비슷하므로 쉽게 이해할 수 있을 것이다.

ex1013.html

```
<head>
<script src="http://code.jquery.com/jquery-1.9.1.js"></script>
<script>
    $(document).ready(function () {
        $('#box').css({ width: 100, height: 100, background: 'orange'}) //스타일 변경 및 이벤트 연결
        .on({
            click: function () {
                $(this).css('background', 'red');
            },
            mouseenter: function () {
                $(this).css('background', 'blue');
            },
            mouseleave: function () {
                $(this).css('background', 'orange');
            }
        });
    });
</script>
</head>
<body>
    <div id="box"></div>
</body>
```

- 이벤트 객체와 기본 이벤트 제거

이벤트 객체의 `preventDefault()` 메서드를 사용하면 문서 객체의 기본 이벤트를 제거한다. 예를 들어 회원가입 페이지에서 사용자가 정보를 정확히 입력했는지 확인한 후 페이지를 이동하기 위해 기본 이벤트를 제거하는 상황에서 사용한다.

메서드 이름	설명
<code>event.preventDefault()</code>	기본 이벤트 제거

아래를 실행하고 a 태그로 생성한 링크를 클릭하면 경고창이 출력된다. 하지만 a 태그의 기본 이벤트를 제거했으므로 페이지는 이동하지 않는다.

```
ex1014.html

<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {
      $('a').click(function (event) {
        alert('click');
        event.preventDefault();
      });
    });
  </script>
</head>
<body>
  <a href="http://hanb.co.kr">Hanbit Media</a>
</body>
```

아래 코드를 실행한 후 이름으로 한글이 아닌 값을 입력하고 입력 양식을 제출하면 에러가 출력되고 이벤트를 제거했으므로 양식이 제출되지 않는다.

```
ex1015.html

<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    //이벤트를 연결한다.
    $(document).ready(function () {
      //이벤트를 연결한다.
      $('form').submit(function (event) {
        //input 태그의 값을 추출한다.
        var value = $('input').val();
        //유효성을 검사한다.
        if (value.replace(/[가-힣]/g, '').length == 0) {
          //유효성 검사 통과
          alert('과정을 진행합니다.');
```


10.5 효과

jQuery 라이브러리를 사용하면 손쉽게 문서 객체에 효과를 부여할 수 있다. 최근에는 CSS3를 사용해 애니메이션을 구현하는 방법도 등장했다. CSS3로 애니메이션 효과를 적용하면 성능은 좋지만 구 버전의 인터넷 익스플로러에서는 작동하지 않으므로 여기서는 jQuery 라이브러리를 사용해 문서 객체에 효과를 부여해보았다.

- 효과 메서드 소개

jQuery 라이브러리는 아래와 같이 문서 객체를 보여주거나 보이지 않게 만드는 간단한 효과 메서드를 제공한다.

메서드 이름	설명
show(속도)	문서 객체를 크게 하여 보여준다.
hide(속도)	문서 객체를 작게 하며 사라지게 한다.
toggle(속도)	show() 메서드와 hide() 메서드를 번갈아 실행한다.
slideDown(속도)	문서 객체를 슬라이드 효과와 함께 보여준다.
slideUp(속도)	문서 객체를 슬라이드 효과와 함께 사라지게 한다.
slideToggle(속도)	slideDown() 메서드와 slideUp() 메서드를 번갈아 실행한다.
fadeIn(속도)	문서 객체를 선명하게 하며 보여준다.
fadeOut(속도)	문서 객체를 흐리게 하며 사라지게 한다.
fadeToggle	fadeIn() 메서드와 fadeOut() 메서드를 번갈아 실행한다.

모든 간단한 효과 메서드는 속도를 매개변수로 입력할 수 있다. 속도를 입력하지 않으면 기본값이 500밀리 초가 적용된다. 속도에는 밀리 초 숫자나 아래의 문자열을 입력할 수 있다.

속도 문자열	속도
show	600밀리 초
normal	500밀리 초
fast	400밀리 초

다음 코드를 실행하고 버튼을 클릭하면 div 태그에 해당하는 내용이 서서히 사라진다. 사라진 이후에 버튼을 다시 클릭하면 다시 나타난다.

ex1016.html

```
<html>
<head>
  <title>jQuery Basic</title>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    //이벤트를 연결한다.
    $(document).ready(function () {
      //이벤트를 연결한다.
      $('button').click(function () {
        //간단한 효과를 적용한다.
        $('.page').fadeToggle('slow');
      });
    });
  </script>
</head>
<body>
  <button>Toggle Show</button>
  <div class="page">
    <h1>What is Lorem Ipsum?</h1>
    <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum...</p>
  </div>
</body>
</html>
```

- 애니메이션 기본

문서 객체에 애니메이션을 부여할 때는 animate() 메서드를 사용한다. animate() 메서드는 다음과 같이 세 가지 형태로 사용한다.

```
$(selector).animate(속성개체);
$(selector).animate(속성개체, 시간);
$(selector).animate(속성개체, 시간, 콜백함수);
```

animate() 메서드의 첫 번째 매개변수인 속성 개체에 입력할 수 있는 값은 다음과 같다.

opacity	right	width
top	bottom	margin
left	height	padding

다음은 사각형 크기를 변경하는 간단한 예제를 만들었다. animate() 메서드를 사용해 div 태그의 width속성과 height 속성을 변경한다.

```
ex1017.html

<html>
<head>
  <title>Animate Basic</title>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    //이벤트를 연결한다.
    $(document).ready(function () {
      //스타일을 변경한다.
      $('#box').css({
        width: 100,
        height: 100,
        background: 'red'
      }).animate({
        width: 300,
        opacity: 0.5,
      }, 500);
    });
  </script>
</head>
<body>
  <div id="box"></div>
</body>
</html>
```

left 속성과 top 속성과 같은 위치 속성을 사용하려면 position 속성을 absolute 또는 relative로 지정해야한다.

- 애니메이션 정지

jQuery 라이브러리는 animate() 메서드 또는 display() 메서드를 사용해 애니메이션을 예약한다. 애니메이션은 동시에 실행되는 것이 아니라 차례대로 실행된다. 아래의 애니메이션 정지 메서드를 사용하면 애니메이션을 잠시 멈추거나 중지시킬 수 있다.

메서드 이름	설명
delay()	애니메이션을 특정한 시간만큼 정지한다.
stop	애니메이션을 완전히 정지한다.

delay() 메서드를 사용하면 animate() 메서드 사이에 간격을 줄 수 있다. 다음과 같이 animate() 메서드 사이에 입력한다. 다른 곳에도 입력할 수 있지만 효과가 없어진다.

```
$('#hl').animate({ }) .delay(1000).animate({ });
```

delay() 메서드에는 매개변수로 숫자를 입력했지만 stop() 메서드에는 불리언을 입력한다. 매개변수 clearQueue의 기본값은 false로 현재 실행 중인 애니메이션만 멈추지만 true를 입력하면 예약된 모든 애니메이션을 제거한다. 매개변수 goToEnd를 사용하면 애니메이션이 멈추었을 경우 animate() 메서드에서 지정한 최종 수치로 변경해 정지한다.

```
$('#hl').stop(clearQueue, goToEnd);
```

다음은 간단한 입력 양식을 사용해 각각의 매개변수를 조절할 수 있게 만든 예제이다. 코드를 실행하면 붉은 상자의 길이가 변하는 애니메이션을 3초 주기로 반복한다. 애니메이션이 실행되는 동안 매개변수 clearQueue와 goToEnd를 조절하고 버튼을 클릭해 작동방식을 직접 확인해보자.

ex1018.html

```
<html>
<head>
  <title>Animation Basic</title>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <script>
    $(document).ready(function () {
      // 변수를 선언합니다.
      var clearQueue = false;
      var goToEnd = false;

      // 이벤트를 연결합니다.
      $('#clearQueue').change(function () {
        clearQueue = $(this).prop('checked')
      });
      $('#goToEnd').change(function () {
        goToEnd = $(this).prop('checked');
      });
      $('#stopButton').click(function () {
        $('#box').stop(clearQueue, goToEnd);
      });

      // 스타일을 변경합니다.
      $('#box').css({
        width: 100,
        height: 100,
        background: 'red'
      })

      // 타이머를 연결합니다.
      setInterval(function () {
        $('#box').animate({
          width: 200
        }, 1000).delay(1000).animate({
          width: 100,
          height: 100
        }, 1000);
      }, 3000);
    });
  </script>
</head>
<body>
  <!-- 입력 양식 -->
  <label>clearQueue</label>
  <input id="clearQueue" type="checkbox"/>
  <br/>
  <label>goToEnd</label>
  <input id="goToEnd" type="checkbox"/>
  <br/>
  <button id="stopButton">Stop</button>
  <!-- 애니메이션 -->
  <div id="box"></div>
</body>
</html>
```

11장 자바스크립트와 jQuery 라이브러리 응용

자바스크립트와 jQuery 라이브러리는 응용할 수 있는 분야가 굉장히 많다. 이전 장에서 기본적인 내용만 다루었다면 이번 장에서는 실제 웹 페이지 개발에 바로 활용할 수 있는 것들을 살펴보겠다.

- 입력 양식 자동 초점 설정하기

다음 코드를 실행하면 주민등록번호를 입력할 수 있는 입력양식이 나온다. 첫 번째 입력 칸을 선택하고 글자를 입력하다 보면 커서가 자동으로 두 번째 입력 칸으로 이동한다. 또한 두 번째 입력 칸에서 숫자를 지우다보면 커서가 자동으로 첫 번째 입력 칸으로 넘어간다.

ex1101.html

```
<html>
<head>
  <script>
    //이벤트를 연결한다.
    window.onload = function () {
      //문서 객체를 선택한다.
      var input_1 = document.querySelectorAll('input')[0];
      var input_2 = document.querySelectorAll('input')[1];

      //input_1
      input_1.onkeydown = function () {
        //글자 개수가 여섯 개 이상일 경우
        if (6 <= input_1.value.length) {
          //input_2 문서 객체로 초점을 이동한다.
          input_2.focus();
        }
      };

      //input_2
      input_2.onkeydown = function (event) {
        //이벤트 객체를 추출한다.
        var event = event || window.event;
        //사용자의 입력이 '백 스페이스'이고 입력된 글자가 없는 경우
        if (event.keyCode == 8 && input_2.value.length == 0) {
          input_1.focus();
        }
      };
    };
  </script>
</head>
<body>
  <input type="text" maxlength="6"/>
  <span>-</span>
  <input type="text" maxlength="7"/>
</body>
</html>
```

- 프레임 애니메이션 만들기

자바스크립트의 타이머 함수를 사용하면 특정 시간마다 코드를 실행시킬 수 있다. setInterval() 함수는 포털 사이트 메인 페이지에서 실시간 검색어 출력에 활용된다. 실시간 검색어는 일정 시간마다 서버에 데이터를 요청해서 사용자에게 보여주는 것이다.

함수이름	설명
setInterval(함수, 시간)	특정한 시간마다 함수 실행
setTimeout(함수, 시간)	특정한 시간 후에 함수 실행
clearInterval(식별번호)	setInterval() 함수로 설정한 타이머 제거
clearTimeOut(식별번호)	setTimeout() 함수로 설정한 타이머 제거

다음 코드를 실행하면 이미지가 1초에 20번 전환되며 사람이 달리는 애니메이션을 확인할 수 있다.

ex1102.html

```
<html>
<head>
  <script>
    window.onload = function () {
      //변수를 선언한다.
      var count = 0;
      var image = document.getElementById('image');
      var frames = [
        '0.png', '1.png', '2.png', '3.png', '4.png',
        '5.png', '6.png', '7.png', '8.png', '9.png',
        '10.png', '11.png', '12.png', '13.png', '14.png',
      ];
      //타이머 반복을 시작한다.
      setInterval(function () {
        image.src = frames[count % frames.length];
        count = count + 1;
      }, 1000 / 20);
    };
  </script>
</head>
<body>
  <img id="image"/>
</body>
</html>
```

• 문서 객체 생성하기

순수하게 자바스크립트만 사용해 문서 객체를 생성하는 것은 굉장히 복잡하다. 하지만 jQuery 라이브러리를 사용하면 문서 객체를 쉽게 생성할 수 있다. 아래 코드는 jQuery 라이브러리를 사용하여 문서 객체를 생성하는 예제이다.

ex1103.html

```
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"> </script>
  <script>
    //이벤트를 연결한다.
    $(document).ready(function () {
      //문서 객체를 생성한다.
      $('<h1>Create Document Object By jQuery</h1>')
    });
  </script>
</head>
<body>
</body>
</html>
```

위의 코드를 이용해 생성한 문서 객체는 아래의 메서드를 사용해 화면에 추가한다. 각각의 메서드를 사용하여 생성한 태그를 원하는 위치에 추가한다. prependTo() 메서드와 appendTo() 메서드는 대상의 내부에 추가되며 beforeTo() 메서드와 afterTo() 메서드는 대상의 외부에 추가된다.

메서드이름	설명
\$(객체).prependTo(대상)	객체를 대상의 앞부분에 추가
\$(객체).addendo(대상)	객체를 대상의 뒷부분에 추가
\$(객체).beforeTo(대상)	객체를 대상의 앞에 추가
\$(객체).afterTo(대상)	객체를 대상의 뒤에 추가

다음은 appendTo() 메서드를 사용하여 문서 객체를 body 태그에 추가해 보았다. 생성된 문서 객체가 화면에 표시된다.

ex1104.html

```
<head>
<title>jQuery DOM</title>
<script src="http://code.jquery.com/jquery-1.9.1.js"></script>
<script>
    //이벤트를 연결한다.
    $(document).ready(function () {
        //10회 반복한다.
        for (var i = 0; i < 10; i++) {
            //문서 객체를 생성한다.
            $('<h1>Create Document Object + ' + i + '</h1>').css({
                backgroundColor: 'black',
                color: 'red'
            }).appendTo('body');
        }
    });
</script>
</head>
<body>
</body>
```

jQuery 라이브러리에는 위 표의 추가 메서드 외에 아래 표의 추가 메서드도 있다.

메서드이름	설명
\$(대상).prepend(객체)	객체를 대상의 앞부분에 추가
\$(대상).append(객체)	객체를 대상의 뒷부분에 추가
\$(대상).before(객체)	객체를 대상의 앞에 추가
\$(대상).after(객체)	객체를 대상의 뒤에 추가

다음은 추가 메서드를 이용해 문서 객체를 추가하는 예제이다. jQuery 객체를 변수에 저장할 때는 변수 앞에 식별자 \$를 붙여 사용한다.

ex1105.html

```
<head>
<title>jQuery DOM</title>
<script src="http://code.jquery.com/jquery-1.9.1.js"></script>
<script>
    //이벤트를 연결한다.
    $(document).ready(function () {
        //10회 반복한다.
        for (var i = 0; i < 10; i++) {
            //문서 객체를 생성한다.
            var $dynamic = $('<h1>Create Document Object + ' + i + '</h1>').css({
                backgroundColor: 'black',
                color: 'red'
            });

            //문서 객체를 추가한다.
            $('body').append($dynamic);
        }
    });
</script>
</head>
<body>
</body>
```

- 무한 스크롤 기능 구현하기

페이스북, 핀터레스트, 쿠팡 등의 사이트를 살펴보면 스크롤을 맨 아래까지 내리더라도 무한 스크롤을 활용하여 정보를 계속 제공한다. 무한 스크롤은 사용자가 스크롤을 끝까지 내리면 자동으로 데이터를 추가하여 스크롤이 끝나지 않게 만들어주는 기능으로 현대 웹 사이트에 다양하게 활용한다.

다음 코드는 무한스크롤은 스크롤 이벤트가 발생할 경우 전체 문서의 높이가 스크롤 높이와 브라우저 창의 높이를 더한 값과 같은지 조건을 확인한다. 이 조건을 충족하는 경우에만 데이터를 추가하는 단순한 코드를 추하는 예제이다.

ex1106.html

```
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.10.2.js"></script>
  <style>
    article { padding: 20px; }
    article .block {
      padding: 20px;
      min-height: 200px;
    }
    article .block p {
      line-height: 20px;
      color: white;
      font-size: 16px; font-weight: 600;
    }
    article .block:nth-child(2n+1) { background: gray; }
    article .block:nth-child(2n) { background: black; }
  </style>
</head>
<body>
  <article>
    <div class="block">
      <p>
        Lorem ipsum dolor sit amet consectetur adipisicing elit. Voluptatem mollitia accusamus sequi ipsa,
        rerum nam laboriosam, ipsam aperiam deleniti beatae expedita id quisquam veritatis corporis,
        voluptates ducimus molestiae eum adipisci.
      </p>
    </div>
    <div class="block">
      <p>
        Lorem ipsum dolor sit amet consectetur adipisicing elit. Voluptatem mollitia accusamus sequi ipsa,
        rerum nam laboriosam, ipsam aperiam deleniti beatae expedita id quisquam veritatis corporis,
        voluptates ducimus molestiae eum adipisci.
      </p>
    </div>
    <!-- 반복될 임시 콘텐츠 추가 영역 -->
  </article>
</body>
<script>
  var count = 0; //추가될 임시 콘텐츠 수
  $(window).on('scroll', function() {
    var scroll = $(this).scrollTop() + $(this).prop('innerHeight'); //스크롤위치 + window창 높이
    var height=$(document).height(); //문서 전체 높이

    //스크롤의 위치가 문서의 마지막에 위치한 경우 페이지를 불러온다.
    if(scroll+1 >= height) {
      count++;
      var addContent = '<div class="block"><p>'+ count +'번째로 추가된 콘텐츠</p></div>';
      $('article').append(addContent); //article에 추가되는 콘텐츠를 append
    }
  });
</script>
</html>
```

- LightBox 구현하기

라이트박스는 웹 페이지 내부에서 특정한 정보를 보여주기 위해 배경을 어둡게 하고 정보는 밝게 보여주는 기능이다.

lightbox.css

```
#darken-background {
    position: fixed;
    top:0px; left:0px; right:0px;
    height: 100%;
    display: none;
    background: rgba(0, 0, 0, 0.5);
    z-index: 10000;
    overflow-y: scroll;
}

#lightbox {
    width: 700px;
    margin: 20px auto; padding:15px;
    border: 1px solid #333333;
    border-radius: 5px;
    background: white;
    box-shadow: 0px 5px 5px rgba(34, 25, 25, 0.4);
    text-align: center;
}
```

ex1107.html

```
<html>
<head>
    <script src="http://code.jquery.com/jquery-1.10.2.js"></script>
    <link rel="stylesheet" href="./lightbox.css">
    <style>
        .box {
            margin: 10px;
            float: left;
        }
    </style>
    <script>
        $(document).ready(function(){
            $("#darken-background").click(function(){
                $("#darken-background").hide(); //라이트박스를 숨기게 한다.
            });
            $('.box').click(function(){
                $("#darken-background").show(); //라이트박스를 보이게 한다.
            });
        });
    </script>
</head>
<body>
    <div class="box"></div>
    <div class="box"></div>
    <div class="box"></div>

    <!--라이트 박스-->
    <div id="darken-background">
        <div id="lightbox">
            
        </div>
    </div>
</body>
</html>
```


- 유효성 체크

아래 코드는 상품등록을 위한 내용들을 입력 받으며 유효성 체크(한글, 숫자, 길이, 공백 등)를 하는 예제 프로그램이다.

ex1108.html

```
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <style>
    table {border-collapse: collapse;}
    td { border-bottom: 1px solid gray; padding:10px;}
    .divButton { width: 500px; padding: 10px;}
  </style>
</head>
<body>
  <h1>[상품등록]</h1>
  <form name="frm" action="sqlInsert.jsp">
    <table>
      <tr><td><input type="text" name="pname" placeholder="상품명"></td></tr>
      <tr><td><input type="text" name="price" placeholder="가격"></td></tr>
      <tr>
        <td>
          <input type="text" name="num1" placeholder="num1">-<input type="text" name="num2" placeholder="num2">
        </td>
      </tr>
    </table>
    <div class="divButton">
      <input type="submit" value="상품등록"><input type="reset" value="등록취소">
    </div>
  </form>
</body>
<script>
  $(frm).submit(function(event){
    event.preventDefault(); //event 동작을 막는다.
    if(confirm("등록하시겠습니까?")==false) return;
    //입력받은 값을 변수에 저장
    var pname=$(frm.pname).val();
    var price=$(frm.price).val();
    var num1=$(frm.num1).val();
    var num2=$(frm.num2).val();
    if(pname.replace(/[가-힣]/g, '') || pname==""){ //한글이 아닌 문자가 있는지 체크
      alert("한글만 입력하세요!");
      $(frm.pname).val("");
      $(frm.pname).focus();
    }
    else if(price.replace(/[0-9]/g, '') || price==""){ //숫자가 아닌 문자가 있는지 체크
      alert("숫자만 입력하세요!");
      $(frm.price).val("");
      $(frm.price).focus();
    }
    else if(num1.replace(/[0-9]/g, '') || num1.length != 6){
      alert('숫자 6자리로 입력하세요!');
      $(frm.num1).val("");
      $(frm.num1).focus();
    }
    else if(num2.replace(/[0-9]/g, '') || num2.length != 7){
      alert('숫자 7자리로 입력하세요!');
      $(frm.num2).val("");
      $(frm.num2).focus();
    }
    else {
      alert('성공');
    }
  });
</script>
</html>
```

- 소문자가 아닌 문자가 있는지 체크
\$(frm.pname).val().replace(/[a-z]/g, '')
- 대문자가 아닌 문자가 있는지 체크
\$(frm.pname).val().replace(/[A-Z]/g, '')
- 소문자나 대문자가 아닌 문자가 있는지 체크
\$(frm.pname).val().replace(/[A-Z]/gi, '')
- 숫자에 컴마 출력
price.toString().replace(/B(?=(\d{3})+(?! \d))/g, ',');

- 문서객체 추가 삭제

다음 예제는 이름, 주소를 입력받아 저장 버튼을 클릭하면 아래 테이블에 추가되는 프로그램이다. 삭제버튼을 클릭하면 내용이 삭제된다.

ex1109.html

```
<html>
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <style>
    table {border-collapse: collapse; margin-top: 10px;}
    td {border: 0.5px solid gray; padding: 5px;}
  </style>
</head>
<body>
  이름: <input type="text" id="txtName"> <hr>
  주소: <input type="text" id="txtAddress" size=50><hr>
  <input type="button" value="저장" id="btnSave">
  <input type="button" value="취소" id="btnCancel"><hr>
  <button id="btnDel">선택삭제</button>
  <table width=500 id="tbl">
    <tr>
      <td width=50><input type="checkbox" id="chkAll"></td>
      <td width=100>이름</td><td width=300>주소</td><td width=50>삭제</td>
    </tr>
  </table>
</body>
<script>
  $("#btnSave").on("click", function() { //저장버튼 클릭한 경우
    var txtName=$("#txtName").val();
    var txtAddress=$("#txtAddress").val();
    var str="<tr>";
    str += "<td><input type='checkbox'></td>";
    str += "<td>" + txtName + "</td>";
    str += "<td>" + txtAddress + "</td>";
    str += "<td><button>삭제</button></td>";
    str += "</tr>";
    $("#tbl").append(str);
    $("#txtName").val("");
  });
  $("#tbl").on("click", "tr button", function() { //행 삭제버튼을 클릭한 경우
    var row=$(this).parent().parent();
    $(row).remove();
  });
  $("#chkAll").on("click", function() { //전체선택버튼을 클릭한 경우
    if($("#tbl tr input:checkbox").checked){
      $("#tbl tr input:checkbox").each(function(){
        $(this).prop("checked", true);
      });
    }else{
      $("#tbl tr input:checkbox").each(function(){
        $(this).prop("checked", false);
      });
    }
  });
  $("#btnDel").on("click", function() { //선택삭제버튼을 클릭한 경우
    $("#tbl tr input:checkbox:checked").each(function(){
      var row=$(this).parent().parent();
      $(row).remove();
    });
  });
</script>
</html>
```

이름:

주소:

<input type="checkbox"/>	이름	주소	삭제
<input type="checkbox"/>	홍길동	인천 서구 경서동 동문아파트	<input type="button" value="삭제"/>
<input type="checkbox"/>	심청이	인천 부평구 계산동 현대아파트	<input type="button" value="삭제"/>

```
//각행의 체크박스를 클릭한 경우
var all=$("#tbl .row .chk").length;
var checked=$("#tbl .row .chk:checked").length;

return false //반복문을 빠져나간다.
```

- ajax을 이용한 게시글 목록출력

ajax이란 자바스크립트를 통해서 서버에 데이터를 비동기 방식으로 요청하는 것이다. 다음 예제는 아래 주소에 있는 서버의 posts 데이터를 비동기적으로 호출해 한 페이지에 3개씩 출력하는 예제이다.

<https://jsonplaceholder.typicode.com/>

ex1110.html

```
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <style>
    #content { width: 800px; margin: 0 auto; }
    #pagination {
      width: 800px;
      margin: 0 auto; text-align: center; }
  </style>
</head>
<body>
  <div id="content"></div>
  <div id="pagination">
    <button id="prev">이전</button>
    <span id="page"></span>
    <button id="next">다음</button>
  </div>
</body>
<script>
  var page = 1;
  getList();
  function getList(){ //목록출력함수
    $.ajax({
      type: 'get',
      url: 'https://jsonplaceholder.typicode.com/posts',
      dataType: 'json',
      success: function(data){
        var str = '';
        $(data).each(function(){
          var id = this.id;
          var title = this.title;
          var body = this.body;
          if(id >=(page-1)*3+1 && id <=page*3) {
            str += '<h3>${id}:${title}</h3>';
            str += '<p>${body}</p>';
          }
        });
        $('#content').html(str);
        if(page==1) { $('#prev').attr('disabled', true); } //1페이지인 경우 prev 버튼의 disabled 속성을 true로 설정한다.
        else{ $('#prev').attr('disabled', false); }
      }
    });
    $('#page').html(page);
  }
  $('#next').on('click', function(){ //다음 버튼을 클릭한 경우
    page = page + 1;
    getList();
  });
  $('#prev').on('click', function(){ //이전 버튼을 클릭한 경우
    page = page - 1;
    getList();
  });
</script>
```

1:sunt aut facere repellat provident occaecati excepturi optio reprehenderit

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto

2:qui est esse

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui neque nisi nulla

3:ea molestias quasi exercitationem repellat qui ipsa sit aut

et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantium quis pariatur molestiae porro eius odio et labore et velit aut

이전 1 다음

- ajax을 이용한 photo 목록 출력

<https://jsonplaceholder.typicode.com/photos>



photo.css

```
#content {
  width: 850px;
  margin: 0 auto;
  overflow: hidden;
}

.box {
  width: 150px; height: 150px;
  float: left;
  margin: 5px;
  padding: 5px;
  border-radius: 5px 5px 5px 5px;
  box-shadow: 3px 3px 5px gray;
  text-align: center;
}

.box img {
  width: 120px;
}

.box .title {width: 120px; font-size: 13px;
  margin: 0 auto;
  text-align: center;
  white-space: nowrap;
  overflow: hidden;
  text-overflow: ellipsis;
}

#pagination {
  width: 850px;
  margin: 0 auto;
  text-align: center;
  margin-top: 10px;
}
```

```

<html>
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <link rel="stylesheet" href='photo.css'/>
</head>
<body>
  <div id="content"></div>
  <div id="pagination">
    <button id="prev">이전</button>
    <span id="page"></span>
    <button id="next">다음</button>
  </div>
</body>
<script>
  var page = 1;
  getList();
  //목록출력 함수
  function getList(){
    $.ajax({
      type: 'get',
      url: 'https://jsonplaceholder.typicode.com/photos',
      dataType: 'json',
      success: function(data){
        var str = '';
        $(data).each(function(){
          var id = this.id;
          var title = this.title;
          var url = this.url;
          if(id >=(page-1)*10+1 && id <=page*10) {
            str += '<div class="box">';
            str += '<div><img src='${url}'></div>';
            str += '<div class="title">${title}</div>';
            str += '</div>';
          }
        });
        $('#content').html(str);
        //1페이지인 경우 prev 버튼의 disabled 속성을 true로 설정한다.
        if(page==1) {
          $('#prev').attr('disabled', true);
        }else{
          $('#prev').attr('disabled', false);
        }
      }
    });
    $('#page').html(page);
  }

  //다음 버튼을 클릭한 경우
  $('#next').on('click', function(){
    page = page + 1;
    getList();
  });
  //다음 버튼을 클릭한 경우
  $('#prev').on('click', function(){
    page = page - 1;
    getList();
  });
</script>
</html>

```

- 게시물(post)에 대한 댓글(comment) 목록 출력

<https://jsonplaceholder.typicode.com/posts/2/comments>

1:sunt aut facere repellat provident occaecati excepturi optio reprehenderit

2:qui est esse

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui neque nisi nulla

6:et fugit eligendi deleniti quidem qui sint nihil autem[Presley.Mueller@myrl.com]

doloribus at sed quis culpa deserunt consectetur qui praesentium accusamus fugiat dicta voluptatem rerum ut voluptate autem voluptatem repellendus aspernatur dolorem in

7:repellat consequatur praesentium vel minus molestias voluptatum[Dallas@ole.me]

maiores sed dolores similique labore et inventore et quasi temporibus esse sunt id et eos voluptatem aliquam aliquid ratione corporis molestiae mollitia quia et magnam dolor

comment.css

```
#content {
  width: 800px;
  margin: 0 auto;
}

#comments {
  width: 800px;
  margin: 0 auto;
}

h3 {
  color: blue;
  cursor: pointer;
}

h5 {
  color: green;
}

.box {
  background: rgb(250, 246, 246);
  color: gray;
  padding: 5px;
  margin: 5px;
}

.body {
  display: none;
}
```

```

<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <link rel="stylesheet" href="comment.css"/>
</head>
<body>
  <div id="content"></div>
  <hr/>
  <div id="comments"></div>
</body>
<script>
  getList();
  //post 목록 출력 함수 정의
  function getList(){
    $.ajax({
      type: 'get',
      url: 'https://jsonplaceholder.typicode.com/posts',
      dataType: 'json',
      success: function(data){
        var str = '';
        $(data).each(function(){ //포스트 목록을 출력한다.
          var id = this.id;
          var title = this.title;
          var body = this.body;
          if(id <=2) {
            str += '<div class=box postid="'+id+'">';
            str += '<h3>'+id+':'+title+'</h3>';
            str += '<p class="body">'+body+'</p>';
            str += '</div>'
          }
        });
        $('#content').html(str);
      }
    });
  }
  $('#content').on('click', '.box', function(){
    $('#content .box .body').each(function(){ //모든 포스트 목록 내용을 숨긴다.
      $(this).hide();
    });
    var postid = $(this).attr('postid');
    $(this).find('.body').show(); //현재 클릭한 포스트 내용을 보여준다.
    $.ajax({
      type: 'get',
      url: 'https://jsonplaceholder.typicode.com/posts/'+postid+'/comments',
      dataType: 'json',
      success: function(data){
        var str="";
        $(data).each(function(){ //커멘트 목록을 출력한다.
          var name=this.name;
          var email=this.email;
          var body=this.body;
          var id=this.id;
          str += '<div class="box">';
          str += '<h5>'+id+':'+name+'['+email+'</h5>';
          str += '<p>'+body+'</p>';
          str += '</div>'
        });
        $('#comments').html(str);
      }
    });
  });
</script>

```

- Todos 목록 무한 스크롤

4:et porro tempora

5:laboriosam mollitia et enim quasi adipisci quia provident illum

6:qui ullam ratione quibusdam voluptatem quia omnis

7:illo expedita consequatur quia in

8:quo adipisci enim quam ut ab

ex1113.html

```
<head>
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <style>
    #content {
      width: 800px; height: 300px; margin: 0 auto; background: whitesmoke; padding: 10px; overflow: auto; }
  </style>
</head>
<body>
  <div id="content"></div>
</body>
<script>
  var page = 1;
  //마지막 페이지를 구한다.
  var lastPage = Math.ceil(200/10);
  getList();
  function getList(){
    $.ajax({
      type: 'get',
      url: 'https://jsonplaceholder.typicode.com/todos',
      dataType: 'json',
      success: function(data){
        var str = '';
        $(data).each(function(){
          var id = this.id;
          var title = this.title;
          var body = this.body;
          if(id >=(page-1)*10+1 && id <=page*10) {
            str += '<h4>${id}:${title}</h4>';
          }
        });
        $("#content").append(str);
      }
    });
  }
  $("#content").scroll(function() {
    if (page == lastPage) return;
    var scroll=$(this).scrollTop() + $(this).innerHeight(); //스크롤위치 + 보이는 문서의 높이
    var height=$(this).prop('scrollHeight'); //문서의 전체 높이

    //스크롤의 위치가 문서의 마지막에 위치한 경우 페이지를 불러온다.
    if(scroll+1 >= height){
      page++;
      getList();
    }
  });
</script>
```


- jQuery 라이브러리로 갤러리 구현하기 (slick slider)



ex1114.html

```
<html>
  <head>
    <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
    <style>
      #slider{ position: relative; overflow: hidden; width:550px; height: 150px; }
      #items { position: absolute; width:1100px; overflow: hidden; margin-left: -110px;}
      #items .item { width: 100px; float: left; margin-left:10px;}
      #items .item img {width:100px; height: 100px;}
    </style>
  </head>
  <body>
    <div id="slider">
      <div id="items"></div>
    </div>
    <div style="width:550px; text-align:center;margin-top: 10px;">
      <button id="prev">< </button>&nbsp;&nbsp;&nbsp;<button id="next">> </button>
    </div>
  </body>
</script>

//item생성 후 items에 추가
for(var i=0; i<10; i++){
  var str="";
  str += `<div class="item">`;
  str += `<h4>Kitten${i}</h4>`;
  str += ``;
  str += `</div>`;
  $('#items').append(str);
}

//슬라이드 반복 실행
var repeat=setInterval( function(){ $('#next').click(); }, 2500);

$('#next').on('click', function(){
  clearInterval(repeat); //슬라이드 반복 중지
  $('#items .item:first').animate({ marginLeft:-110 }, 300, function(){
    $(this).appendTo($('#items')).css({marginLeft:10});
  });
  repeat=setInterval(function(){ $('#next').click(); }, 2500); //슬라이드 반복 실행
});

$('#prev').on('click', function(){
  clearInterval(repeat); //슬라이드 반복 중지
  $('#items .item:first').animate({ marginLeft: 110 }, 300, function(){
    $(this).before($('#items .item:last')).css({marginLeft:10});
  });
  repeat=setInterval(function(){ $('#next').click(); }, 2500); //슬라이드 반복 실행
});
</script>
</html>
```

- 프레임 애니메이션

ex1115.html

```
<head>
<script src="http://code.jquery.com/jquery-1.9.1.js"></script>
<style>
    #slider {
        display: flex;
        width:400px; height: 250px; margin: 10px auto;
        border: 1px solid gray; box-shadow: 10px 10px 5px gray;
        align-items: center; justify-content: center;}
    #items {
        padding: 10px; margin: 10px;
        background-color: aqua;
    }
    #item {
        display: flex;
        align-items: center; justify-content: center; }
    #item img {
        width:150px; height: 150px; margin: 5px; }
    #item .title { color: white; font-size:20px; }
</style>
</head>
<body>
<div id="slider">
    <button id="prev">◀</button>
    <div id="items">
        <div id="item"></div>
    </div>
    <button id="next">▶</button>
</div>
</body>
<script>
    var index=0;
    var array= [];
    //item생성 후 배열에 저장
    for(var i=0; i<10; i++){
        var str = '';
        str += ``;
        str += `<div class="title">Kitten[${i}]</div>`;
        array.push(str);
    }

    $('#item').html(array[index]);
    //슬라이드 반복 실행
    var repeat = setInterval(function(){$('#next').click();}, 3000);

    $('#prev').on('click', function(){
        clearInterval(repeat); //슬라이드 반복 중지
        if(index == 0) index=9; else index--;
        $('#item').hide().html(array[index %10]).fadeIn(1000);
        repeat = setInterval(function(){$('#next').click();}, 3000); //슬라이드 반복 실행
    });

    $('#next').on('click', function(){
        clearInterval(repeat); //슬라이드 반복 중지
        if(index == 9) index=0; else index++;
        $('#item').hide().html(array[index %10]).fadeIn(1000);
        repeat = setInterval(function(){$('#next').click();}, 3000); //슬라이드 반복 실행
    });
</script>
```



12. 반응형 웹 페이지

12.1 반응형 웹 페이지 레이아웃

웹사이트의 레이아웃을 만들 때 사용하는 모니터의 화면 해상도를 고려하여 레이아웃이 PC인 경우와 모바일인 경우 적절한 레이아웃으로 변경되어야 한다. 이러한 문제를 해결하는 방법 중의 하나가 반응형 웹디자인입니다. 해상도에 따라 가로폭이나 배치를 변경하여 가독성을 높이는 것이다.

```
@media ( max-width: 768px ) {  
    body { color: red; } /* 웹브라우저의 가로 해상도가 768px 이하일 때 글자색을 빨간색으로 바꾼다. */  
}
```

가로폭 조정을 위해서 HTML 문서의 <head>와 </head> 사이에 다음의 코드를 넣어야 한다.

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

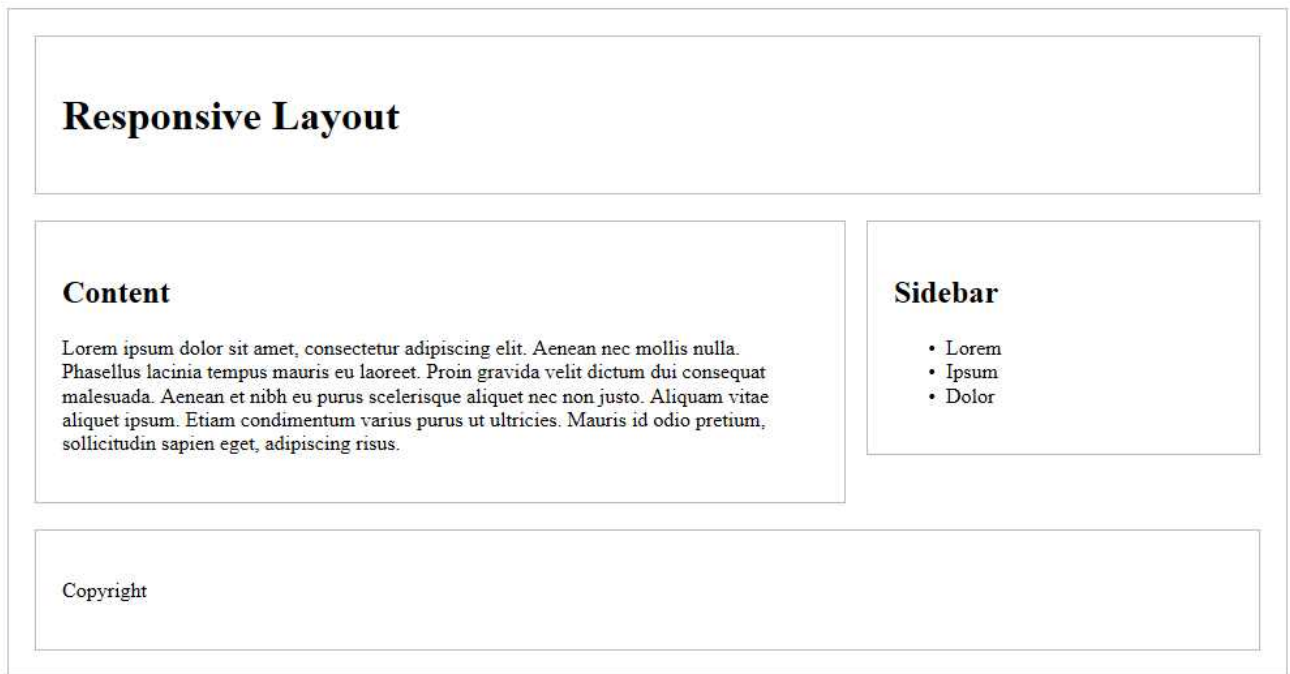
style.css

```
#container {  
    width: 940px;  
    margin: 10px auto;  
    padding: 20px;  
    border: 1px solid #bcbcbc;  
}  
#header {  
    padding: 20px;  
    margin-bottom: 20px;  
    border: 1px solid #bcbcbc;  
}  
#content {  
    width: 580px;  
    padding: 20px;  
    margin-bottom: 20px;  
    float: left;  
    border: 1px solid #bcbcbc;  
}  
#sidebar {  
    width: 260px;  
    padding: 20px;  
    margin-bottom: 20px;  
    float: right;  
    border: 1px solid #bcbcbc;  
}  
#footer {  
    clear: both;  
    padding: 20px;  
    border: 1px solid #bcbcbc;  
}  
  
@media ( max-width: 768px ) {  
    #container {  
        width: auto;  
    }  
    #content {  
        float: none;  
        width: auto;  
    }  
    #sidebar {  
        float: none;  
        width: auto;  
    }  
}
```

- 모니터의 해상도가 768px 이하인 경우



- 모니터의 해상도가 769px 이상인 경우



ex1201.html

```
<html>
<head>
  <title>CSS</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="./style.css"/>
</head>
<body>
  <div id="container">
    <div id="header">
      <h1>Responsive Layout</h1>
    </div>
    <div id="content">
      <h2>Content</h2>
      <p>
        Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean nec mollis nulla. Phasellus lacinia
        tempus mauris eu laoreet. Proin gravida velit dictum dui consequat malesuada. Aenean et nibh eu
        purus scelerisque aliquet nec non justo. Aliquam vitae aliquet ipsum. Etiam condimentum varius
        purus ut ultricies. Mauris id odio pretium, sollicitudin sapien eget, adipiscing risus.
      </p>
    </div>
    <div id="sidebar">
      <h2>Sidebar</h2>
      <ul>
        <li>Lorem</li>
        <li>Ipsum</li>
        <li>Dolor</li>
      </ul>
    </div>
    <div id="footer">
      <p>Copyright</p>
    </div>
  </div>
</body>
</html>
```

12.2 반응형 웹 네비게이션

- PC 메뉴



- 모바일 메뉴



ex1202.html

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <script src="http://code.jquery.com/jquery-1.9.1.js"></script>
  <link rel="stylesheet" href="./navi.css"/>
</head>
<body>
  <div id="wrap">
    <header class="header fixed">
      <h1 class="logo"><a href=</a></h1>
      <nav>
        <ul class="gnb">
          <li><a href="">MENU1</a></li>
          <li><a href="">MENU2</a></li>
          <li><a href="">MENU3</a></li>
          <li><a href="">MENU4</a></li>
          <li><a href="">MENU5</a></li>
        </ul>
      </nav>
      <span class="menu-toggle-btn">
        <span></span>
        <span></span>
        <span></span>
      </span>
    </header>
  </div>
</body>
<script>
  $(".menu-toggle-btn").click(function() {
    $(".gnb").stop().slideToggle("fast");
  });
</script>
</html>
```

12.2 반응형 웹 네비게이션

header.css

```
/* 컴퓨터 모바일 공용 CSS----- */
a {
    text-decoration: none;
}

.header {
    position: relative;
    padding: 18px 2%;
    height: 50px;
}

.header .logo {
    float: left;
    padding: 0;
    margin: 0;
    padding-left: 10px;
    margin-top: auto;
    margin-bottom: auto;
}

.header .logo img { width: 200px; }

.header .gnb {
    margin: 0;
    display: none;
    position: absolute;
    top: 100%;
    left: 60%;
    list-style-type: none;
    width: 40%;
    background: rgba(128, 128, 128, 0.4);
}

.header .gnb li a {
    display: block;
    padding: 20px 0 20px 10px;
    font-size: 14px;
    color: #fff;
}

.header .menu-toggle-btn {
    float: right;
    width: 40px;
    padding-top: 10px;
    cursor: pointer;
}

.header .menu-toggle-btn span {
    display: block;
    width: 100%;
    height: 3px;
    background: #000;
}

.header .menu-toggle-btn span:nth-child(2) {
    margin: 7px 0;
}
```

/* 태블릿용 CSS -----*/

```
@media all and (min-width:768px) {  
  .header {  
    display: grid;  
    grid-template-columns: 30% 70%;  
    padding: 0px;  
    height: auto;  
    margin-top: auto;  
    margin-bottom: auto;  
    background: #808080;  
  }  
  .header .logo {  
    padding: 10px 20px;  
    padding-left: 30px;  
  }  
  .header nav {  
    margin-left: 0px;  
    margin-right: 0px;  
    padding-left: 0px;  
    margin-right: 0px;  
    margin-top: auto;  
    margin-bottom: auto;  
    width: 100%;  
  }  
  .header .gnb {  
    /* !important 없으면 btn 접었을 시 display:none되어 gnb가 표시되지 않음 */  
    display: grid !important;  
    display: block;  
    height: 100%;  
    grid-template-columns: 15% 15% 15% 15% 15%;  
    position: static;  
    width: auto;  
    background: none;  
  }  
  .header .gnb li {  
    float: left;  
    margin-top: auto;  
    margin-bottom: auto;  
    margin-left: 10px;  
    margin-right: 10px;  
    border-bottom: 0;  
  }  
  .header .gnb li a {  
    color: #fff;  
    padding: 0;  
    font-size: 18px;  
  }  
  .header .menu-toggle-btn { display: none; }  
}
```

/* 컴퓨터용 CSS ----- */

```
@media all and (min-width:1132px) {  
  .header { grid-template-columns: 20% 80%; }  
  .header .logo {  
    padding: 10px 20px;  
    padding-left: 50px;  
  }  
  .header .gnb li a { font-size: 20px; }  
}
```

13. 모바일 웹페이지 만들기

13.1 미디어쿼리 사용하기

1) 웹 폰트 사용하기

ex1301.html

```
<html>
<head>
  <meta charset="euc-kr">
  <title>웹 폰트 사용하기</title>
  <style type="text/css">
    @font-face { font-family:SeoulNamsanFont; src:url('image/SeoulNamsan.ttf'); } /* 서울남산체(모바일전용)를 다운로드 받아 설정 */
    p { font-family:SeoulNamsanFont; } /* 웹폰트 이름 사용 */
  </style>
</head>
<body>
  <p>이 문장은 웹폰트로 <b>"서울남산체"</b>를 사용하였습니다.</p>
</body>
</html>
```

2) 미디어쿼리로 디바이스 맞춤형 스타일 준비하기

mobile.css (모바일 스타일)

```
@charset "euc-kr";
body { color:#FFFf;background-color:#000; }
#pc_view { display:none; }
```

pc.css (pc 스타일)

```
@charset "euc-kr";
body { background-color:#fff; }
#mobile_view { display:none; }
```

ex1302.html

```
<html>
<head>
  <meta name="viewport" content="user-scalable=no, width=device-width" charset="euc-kr"/>
  <title>미디어쿼리로 디바이스 맞춤형 스타일 준비하기</title>
  <link href="mobile.css" rel="stylesheet" type="text/css" media="screen and (max-width:799px)">
  <link href="pc.css" rel="stylesheet" type="text/css" media="screen and (min-width:800px)">
</head>
<body>
  <h1>미디어 쿼리 연습</h1>
  <!-- 데스크탑 PC 화면에서만 출력된다. -->
  <div id="pc_view">
    <h2>데스크탑 PC 화면</h2>
    <p>브라우저의 너비를 800 px보다 작게 줄어보세요.</p>
    <p>배경이 검정색에 글자가 흰색으로 바뀔 겁니다.</p>
  </div>
  <!-- 모바일 화면에서만 출력된다. -->
  <div id="mobile_view">
    <h2>모바일 화면</h2>
    <p>브라우저의 너비를 800 px이상으로 크게 늘어보세요.</p>
    <p>배경이 흰색에 글자가 검정색으로 바뀔 겁니다.</p>
  </div>
</body>
</html>
```


3) 모바일 PC 레이아웃 응용

ex1303.html

```
<html>
<head>
  <meta name="viewport" content="user-scalable=no, width=device-width" charset="euc-kr"/>
  <title>모바일과 PC 레이아웃 응용</title>
  <link href="common.css" rel="stylesheet" type="text/css">
  <link href="mobile1.css" rel="stylesheet" type="text/css" media="screen and (max-width:799px)">
  <link href="pcl.css" rel="stylesheet" type="text/css" media="screen and (min-width:800px)">
</head>
<body>
  <div class='pg'>
    <header class='head'><h1>미디어아트 인터랙션 디자인</h1></header>
    <article class='pg-main'>
      <section class='entries'>
        <h2>브라우저의 너비를 800px 미만으로 줄여보세요.</h2>
        <p class='pcview_only'>
          이 문장은 PC화면에서만 보입니다. 이 문장은 PC화면에서만 보입니다. 이 문장은 PC화면에서만 보입니다.
          이 문장은 PC화면에서만 보입니다. 이 문장은 PC화면에서만 보입니다. 이 문장은 PC화면에서만 보입니다.
          이 문장은 PC화면에서만 보입니다. 이 문장은 PC화면에서만 보입니다. 이 문장은 PC화면에서만 보입니다.
        </p>
        <p>
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
        </p>
        <hr/>
        <h2>반대로 브라우저의 너비를 800px 이상으로 늘려 보세요.</h2>
        <p class='pcview_only'>
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
        </p>
        <p class='pcview_only'>
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
          이 문장은 모든 디바이스 화면에서 보입니다. 이 문장은 모든 디바이스 화면에서 보입니다.
        </p>
      </section>
      <nav class='sidebar'>
        <h3>SNS 등록</h3>
        <ul>
          <li><a href='#'>페이스북</a></li>
          <li><a href='#'>트위터</a></li>
        </ul>
        <h3>카테고리</h3>
        <ul>
          <li><a href='#'>디지털 아트</a></li>
          <li><a href='#'>사운드 아트</a></li>
          <li><a href='#'>정보 가시화</a></li>
          <li><a href='#'>뉴 미디어 프로그래밍</a></li>
        </ul>
      </nav>
    </article>
    <footer class='foot'><p>Copyright &copy; 2011 All Rights Reserved</p></footer>
  </div>
</body>
</html>
```

common.css (공통 스타일)

```
@CHARSET "UTF-8";

@font-face {
    font-family:SeoulNamsanFont;
    src:url('image/SeoulNamsan.ttf');
}

* {
    margin:0px;
    padding:0px;
}

body {
    line-height:150%;
    font-family:SeoulNamsanFont;
}

a:link, a:visited {
    color:inherit;
    text-decoration:none;
}

a:hover, a:active {
    text-decoration: underline;
}

h1 {
    font-size:25px;
    color:#FFF;
}

h2 {
    font-size:19px;
    color:#333;
    margin-top:5px;
    margin-bottom:5px;
}

h3 {
    font-size:12px;
    color:#666;
    margin-top:5px;
    margin-bottom:5px;
}

p {
    margin-bottom:10px;
}

header, footer {
    display:block;
}
```

pcl.css (pc 스타일)

```
@charset "euc-kr";

.pg {
    width:800px;
    margin-right:auto;
    margin-left:auto;
}

.head {
    background-image:url(image/title.png);
    background-repeat:no-repeat;
    width:800px;
    height:264px;
}

.head h1 {
    display:none;
}

.entries {
    float:left;
    width:680px;
}

.sidebar {
    float:left;
    text-align: right;
    width: 100px;
    margin-left:20px;
}

.sidebar ul {
    list-style-type: none;
}

.foot {
    background-color:#333;
    border-top-width:1px;
    border-top-style:solid;
    border-top-color:#999;
    padding:10px;
    color: #FFF;
    margin-top:10px;
    clear:left;
    text-align:center;
}
```

```
@charset "euc-kr";

.head {
    background-color:#473E37;
    padding:1em;
}

.pg-main {
    padding-right:10px;
    padding-left:10px;
}

.entries p.pcview_only {
    display:none;
}

.sidebar ul {
    border-left-width:2px;
    border-left-style:solid;
    border-left-color:#ccc;
}

.sidebar ul li {
    display:inline;
    border-right-width:2px;
    border-right-style:solid;
    border-right-color:#ccc;
    padding-right:5px;
    padding-left: 5px;
}

.foot {
    background-color:#333;
    border-top-width:1px;
    border-top-style:solid;
    border-top-color:#999;
    padding:10px;
    color:#FFF;
    margin-top:10px;
    text-align:center;
}
```

13.2 스마트폰 UI 기법 익히기

JQuery Mobile은 아이폰이나 안드로이드 같은 모바일 디바이스에서 모바일 웹 UI를 빠르고 강력하게 만들 수 있는 프레임 워크이다. jquerymobile.com/download 주소로 이동하여 파일을 다운로드 받거나 URL 주소만 복사해서 사용한다.

1) JQuery Mobile UI 시작하기

2) 하나의 페이지로 멀티구성

ex1304.html

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1" charset=euc-kr">
  <title>Jquery Mobile UI 시작하기</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.css"/>
  <script type="text/javascript" src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script type="text/javascript" src="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.js"></script>
</head>
<body>
  <div data-role="page">
    <div data-role="header">
      <h1>Header 입니다.</h1>
    </div>
    <div data-role="content">
      <p>Contents 입니다.</p>
    </div>
    <div data-role="footer"><h2>Footer입니다.</h2></div>
  </div>
</body>
</html>
```

ex1305.html

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1" charset=euc-kr">
  <title>하나의 페이지로 멀티구성</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.css"/>
  <script type="text/javascript" src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script type="text/javascript" src="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.js"></script>
</head>
<body>
  <div data-role="page" id="main">
    <!--a:접정색, b:파랑색, d:베이지색, e:노랑색-->
    <div data-role="header" data-theme="b">
      <h1>메인페이지</h1>
    </div>
    <div data-role="content">
      <h2>메인페이지입니다.</h2>
      <!--a태그안에 data-role="button" 추가해 보기-->
      <p><a href="#sub">서브페이지</a></p>
    </div>
    <div data-role="footer">
      <h2>Footer입니다.</h2>
    </div>
  </div>
  <div data-role="page" id="sub">
    <div data-role="header">
      <h1>서브페이지</h1>
    </div>
    <div data-role="content">
      <h2>서브페이지 입니다.</h2>
      <p><a href="#main" data-direction="reverse">메인페이지</a></p>
    </div>
    <div data-role="footer">
      <h2>Footer입니다.</h2>
    </div>
  </div>
</body>
</html>
```

13.3 스마트폰 페이지 전환 효과 익히기

1) 간단한 목록 만들기

ex1306.html

```
<html>
<head>
  <meta charset="euc-kr">
  <title>간단한 목록 만들기</title>
</head>
<body>
  <ul>
    <li>사과</li>
    <li>바나나</li>
    <li>수박</li>
    <li>딸기</li>
  </ul>
  <ol>
    <li>커피</li>
    <li>홍차</li>
    <li>녹차</li>
    <li>아이스크림</li>
  </ol>
</body>
</html>
```

2) 다양한 페이지 링크연결

ex1307-main.html

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1" charset="euc-kr">
  <title>다양한 페이지 링크연결 메인</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.css"/>
  <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.js"></script>
</head>
<body>
  <div data-role="page" id="page">
    <div data-role="header">
      <h1>다양한 링크 연습</h1>
    </div>
    <div data-role="content">
      <ul data-role="listview">
        <li data-role="list-divider">내부 도메인</li>
        <li>
          <a href="ex1307-sub.html">같은 내부 페이지</a>
        </li>
        <li>
          <a href="ex1307-sub.html" data-rel="dialog">대화상자로 열기</a>
        </li>
        <li data-role="list-divider">외부 도메인</li>
        <li>
          <a href="http://www.google.com">www.google.com</a>
        </li>
        <li>
          <a href="ex1307-sub.html" rel="external">rel="external" 사용</a>
        </li>
      </ul>
    </div>
    <div data-role="footer">
      <h4>Footer 입니다.</h4>
    </div>
  </div>
</body>
</html>
```

ex1307-sub.html

```
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1" charset="euc-kr">
  <title>다양한 페이지 전환 효과 메인</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.css"/>
  <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.js"></script>
</head>
<body>
  <div data-role="page" id="page">
    <div data-role="header"><h1>서브 페이지</h1></div>
    <div data-role="content">
      <h1>다양한 페이지 링크 연습</h1>
      <p>다양한 종류의 링크를 연결할 수 있습니다. </p>
      <a href="ex1307-main.html" data-role="button" data-rel="back" data-theme="b">돌아가기</a>
    </div>
    <div data-role="footer"><h4>Footer 입니다.</h4></div>
  </div>
</body>
```

ex1308-main.html

```
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1" charset="euc-kr">
  <title>다양한 페이지 전환 효과 메인</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.css"/>
  <script type="text/javascript" src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script type="text/javascript" src="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.js"></script>
</head>
<body>
  <div data-role="page" id="page">
    <div data-role="header" data-theme="b"><h1>장면전환 효과입니다.</h1></div>
    <div data-role="content" >
      <ul data-role="listview">
        <li><a href="ex1308-sub.html" data-transition="slide">Slide효과</a></li>
        <li><a href="ex1308-sub.html" data-transition="slideup">SlideUp효과</a></li>
        <li><a href="ex1308-sub.html" data-transition="slidedown">SlideDown효과</a></li>
        <li><a href="ex1308-sub.html" data-transition="pop">Pop효과</a></li>
        <li><a href="ex1308-sub.html" data-transition="fade">Fade효과</a></li>
        <li><a href="ex1308-sub.html" data-transition="flip">Flip효과</a></li>
      </ul>
    </div>
    <div data-role="footer"><h4>Footer입니다.</h4></div>
  </div>
</body>
```

4) 다양한 페이지 전환 효과 메인

ex1308-sub.html

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1" charset="euc-kr">
  <title>다양한 페이지 전환 효과 서브</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.css"/>
  <script type="text/javascript" src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script type="text/javascript" src="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.js"></script>
</head>
<body>
  <div data-role="page" id="page">
    <div data-role="header"><h1>장면전환효과</h1></div>
    <div data-role="content">
      <a href="ex10-03-main.html" data-role="button" data-rel="back" data-icon="back">돌아가기</a>
    </div>
    <div data-role="footer"><h4>Footer 입니다.</h4></div>
  </div>
</body>
</html>
```

13.4 메인 페이지 만들기

1) 아이콘 메뉴 페이지 만들기

ex1309.html (아이콘 메뉴 페이지 만들기)

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1.0" charset="ecu-kr"/>
  <title>버추얼대학교</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0b2/jquery.mobile-1.0b2.min.css"/>
  <script src="http://code.jquery.com/jquery-1.6.2.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0b2/jquery.mobile-1.0b2.min.js"></script>
  <style type="text/css">
    #logo {
      width:176px;
      height:132px;
      text-indent:-9999px;
      background-image:url('image/CI_TITLE.png');
      background-repeat:no-repeat;
      margin-top:0px;
      margin-right:auto;
      margin-left:auto;
      margin-bottom:10px;
    }
    .title_style {
      color: #3F3C30;
      padding-top:10px;
    }
  </style>
</head>
<body>
  <div data-role="page" id="main">
    <div data-role="header">
      <h2 id="logo">가상 대학교</h2>
    </div>
    <div data-role="content" data-theme="d">
      <ul data-role="listview" data-inset="true">
        <li>
          <a href="#" rel="external">
            
            <h3 class="title_style">캠퍼스 이미지</h3>
          </a>
        </li>
        <li>
          <a href="#">
            
            <h3 class="title_style">대학교소개</h3>
          </a>
        </li>
        <li>
          <a href="#">
            
            <h3 class="title_style">홍보동영상</h3>
          </a>
        </li>
        <li>
          <a href="#">
            
            <h3 class="title_style">캠퍼스맵</h3>
          </a>
        </li>
      </ul>
    </div>
  </div>
</body>
</html>
```


3) 메뉴와 툴바 적용하기

ex12-01.html (메뉴와 툴바 적용하기)

```
<html>
<head>
  <meta charset="euc-kr">
  <title>메뉴와 툴바 적용하기</title>
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.css"/>
  <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.js"></script>
</head>
<body>
  <div data-role="page" id="page" data-fullscreen="true">
    <!-- header입니다. ----->
    <div data-role="header" data-position="fixed">
      <a href="#" data-icon="back">Back</a>
      <a href="#" data-icon="home" data-iconpos="notext" data-theme="b">홈</a>
      <h1>Header 입니다.</h1>
      <div data-role="navbar">
        <ul>
          <li>
            <a href="#" class="ui-btn-active">상단1</a>
          </li>
          <li>
            <a href="#">상단2</a>
          </li>
          <li>
            <a href="#">상단3</a>
          </li>
          <li>
            <a href="#">상단4</a>
          </li>
          <li>
            <a href="#">상단5</a>
          </li>
        </ul>
      </div>
    </div>
    <!-- content 입니다. ----->
    <div data-role="content">
      
    </div>
    <!-- footer 입니다. ----->
    <div data-role="footer" data-position="fixed">
      <div data-role="navbar" >
        <ul>
          <li>
            <a href="#" class="ui-btn-active" data-icon="gear" >하단1</a>
          </li>
          <li>
            <a href="#" data-icon="grid" >하단2</a>
          </li>
          <li>
            <a href="#" data-icon="search">하단3</a>
          </li>
          <li>
            <a href="#" data-icon="info">하단4</a>
          </li>
        </ul>
      </div>
    </div>
  </div>
</body>
</html>
```

13.5 photoSwipe 사용법 익히기

1)www.photoswipe.com에 접속하여 필요한 파일을 다운로드 받는다.

2)압축 파일을 푼 후 'photoswipe'라는 폴더 이름으로 변경하여 작업 폴더에 복사해 놓는다.

ex12-02.html (갤러리 실습하기)

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
  <title>버추얼대학교</title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.css"/>
  <!-- Photo Swipe CSS 설정 추가 -->
  <link href="photoswipe/examples/jquery-mobile.css" type="text/css" rel="stylesheet"/>
  <link href="photoswipe/photoswipe.css" type="text/css" rel="stylesheet"/>
  <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0/jquery.mobile-1.0.min.js"></script>
  <!-- Photo Swipe 자바스크립트 설정 추가 -->
  <script src="photoswipe/lib/klass.min.js"></script>
  <script src="photoswipe/code.photoswipe.jquery-3.0.4.min.js"></script>
  <script src="photoswipe/create_ps_instance.js"></script>
</head>
<body>
  <!-- main 페이지입니다. -->
  <div data-role="page" id="main">
    <div data-role="content">
      <a href="#campus_images" >캠퍼스 이미지</a>
    </div>
  </div>
  <!-- campus images 페이지입니다. -->
  <div data-role="page" id="campus_images" class="gallery-page">
    <div data-role="header">
      <a href="#main" data-icon="home" data-direction="reverse" class="ui-btn-left">Home</a>
      <h1>캠퍼스 이미지</h1>
    </div>
    <div data-role="content" data-theme="d">
      <ul class="gallery">
        <li>
          <a href="image/full/cb01.jpg" rel="external">
            </a>
        </li>
        <li>
          <a href="image/full/cb02.jpg" rel="external">
            </a>
        </li>
        <li>
          <a href="image/full/cb03.jpg" rel="external">
            </a>
        </li>
        <li>
          <a href="image/full/cb04.jpg" rel="external">
            </a>
        </li>
        <li>
          <a href="image/full/cb05.jpg" rel="external">
            </a>
        </li>
        <li>
          <a href="image/full/cb06.jpg" rel="external">
            </a>
        </li>
        <li>
          <a href="image/full/cb07.jpg" rel="external">
            </a>
        </li>
      </ul>
    </div>
    <div data-role="footer"><h2>버추얼 대학교</h2></div>
  </div>
</body>
</html>
```

13.6 CSS3 트랜지션으로 애니메이션 효과 만들기

ex14-02.html (대학교 소개 페이지 만들기 실습)

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=euc-kr">
  <title>대학교 소개 페이지 만들기 실습</title>
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0b3/jquery.mobile-1.0b3.min.css"/>
  <script src="http://code.jquery.com/jquery-1.6.2.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0b3/jquery.mobile-1.0b3.min.js"></script>
  <style type="text/css">
    #pic_univ {
      float:left;
      margin-right:10px;
      margin-bottom:10px;
      -webkit-transition-property:width;
      -webkit-transition-duration:.3s; }
    .small { width:30%; }
    .large { width:100%; }
    .clear_both { clear:both; }
    .bold_style {
      font-weight:bold;
      color:#950000; }
  </style>
  <!-- 대학교 소개 자바스크립트 설정 추가-->
  <script type="text/javascript" >
    $(document).ready(function(){
      //처음 flag값을 false로 초기화 시킴
      $("#pic_univ").data("flag","false");
      //flag 값에 따라 스타일을 small과 large를 교환하며 적용
      $("#pic_univ").click(function(){
        var bflag = $("#pic_univ").data("flag");
        if(bflag == "false" ) {
          bflag = "true";
          $("#pic_univ").removeClass("small");
          $("#pic_univ").addClass("large");
        }
        else {
          bflag = "false";
          $("#pic_univ").removeClass("large");
          $("#pic_univ").addClass("small");
        }
        $("#pic_univ").data("flag",bflag);
      });
    });
  </script>
</head>
<body>
  <div data-role="page" id="intro" data-add-back-btn="true" >
    <div data-role="header" data-theme="a" >
      <a href="#" data-icon="back">Back</a>
      <a href="#" data-icon="home" data-iconpos="notext" data-theme="b">홈</a>
      <h1>대학교 소개</h1>
    </div>
    <div data-role="content" data-theme="c">
      
      <h2><span class="bold_style">시대정신</span>을 선도하는 열린 <span class="bold_style">학문 공동체</span>입니다.</h2>
      <div class="clear_both"></div>
      <p>안녕하십니까? 버추얼 대학교에 관심을 가져주셔서 감사합니다. 버추얼 대학교는 1세기의 역사를 지니며 그 동안 많은 발전을 거듭하였습니다.</p>
      <p>버추얼 대학교는 세계에서 가장 선도적인 대학교 중의 하나로 인정받고 있습니다. 설립목적은 사회와 개인에 직접적인 기여와 성공을 준비할 수 있도록 하는 것입니다. 궁극적으로는 인류문명에 유용한 인재를 양성하는 것입니다. 오늘날 버추얼 대학교는 현재 인류가 당면하고 있는 여러 문제를 해결하고 다음 세대를 선도할 수 있는 리더가 될 수 있도록 학생들을 준비시키는 것입니다.</p>
      <p>여러분의 미래를 버추얼 대학교에서 함께하시길 희망합니다.</p>
      <p>총장 Luke Andrew</p>
    </div>
    <div data-role="footer">
      <h2>버추얼 대학교</h2>
    </div>
  </div>
</body>
</html>
```

13.7 동영상 연결하기

ex15-01.html (HTML5로 동영상 연결하기 실습)

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1.0" charset="euc-kr"/>
  <title>HTML5로 동영상 연결하기 실습</title>
</head>
<body>
  <h1>동영상 예제</h1>
  <video width="320" height="240" controls autoplay>
    <source src="video/virtual_university.mp4" type="video/mp4"/>
    <source src="video/virtual_university.ogv" type="video/ogg"/>
    <source src="video/virtual_university.webm" type="video/webm"/>
    브라우저가 HTML5 video 엘리먼트를 지원하지 않습니다.
  </video>
</body>
</html>
```

ex15-02.html (HTML5로 동영상 연결하기 실습)

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1.0" charset="euc-kr"/>
  <title>HTML5로 동영상 연결하기 실습</title>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0rc1/jquery.mobile-1.0rc1.min.css"/>
  <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0rc1/jquery.mobile-1.0rc1.min.js"></script>
  <!-- Video JS 설정 추가 -->
  <link rel="stylesheet" type="text/css" href="video/video-js.css">
  <script src="video/video.js"></script>
  <style type="text/css">
    #video_canvas {margin-top:0px; margin-right:auto; margin-bottom:0px; margin-left:auto;
      width:320px; height:290px;}
    .content_style {padding-right:0px; padding-left:0px;}
  </style>
</head>
<body>
  <div data-role="page" id="video_value" data-add-back-btn="true">
    <div data-role="header" data-theme="a" >
      <a href="#" rel="external" data-icon="home" data-direction="reverse" class="ui-btn-left">Home</a>
      <h1>최고의 가치</h1>
    </div>
    <div data-role="content" data-theme="c" class="content_style" >
      <div id="video_canvas">
        <div>
          <video class="video-js vjs-default-skin" width="320" height="240" controls preload="auto"
            poster="video/most_value.jpg" data-setup="{}">
            <source src="video/virtual_university.mp4" type='video/mp4; codecs="avc1.42E01E,mp4a.40.2"'/>
            <source src="video/virtual_university.ogv" type='video/ogg; codecs="theora, vorbis"'/>
            <source src="video/virtual_university.webm" type='video/webm; codecs="vp8, vorbis"'/>
          </video>
          <!--브라우저가 지원안할 경우 대비-->
          <p class="vjs-no-video">
            <strong>비디오 다운로드:</strong><a href="video/virtual_university.mp4" target="_blank">MP4</a>,
            <a href="video/virtual_university.ogv" target="_blank">Ogg</a>,
            <a href="video/virtual_university.webm" target="_blank">WebM</a><br>
          </p>
        </div>
      </div>
    </div>
    <div data-role="footer">
      <h2>버추얼 대학교</h2>
    </div>
  </div>
</body>
</html>
```

13.8 구글맵 표시하기

1) <http://code.google.com/p/jquery-ui-map>에 접속하여 다운로드 받는다.

2) 압축을 풀고 'ux'폴더와 'web'폴더를 작업폴더에 복사한다.

ex16-01.html (구글맵 표시하기)

```
<html>
<head>
  <meta charset="euc-kr">
  <title>구글맵 표시하기</title>
  <meta name="viewport"
    content="width=device-width, initial-scale=1.0, minimum-scale=1.0, maximum-scale=1.0, user-scalable=no"/>

  <!-- JQuery Mobile을 사용하기 위해 반드시 필요한 태그-->
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0rc2/jquery.mobile-1.0rc2.min.css" />
  <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0rc2/jquery.mobile-1.0rc2.min.js"></script>

  <!-- JQuery-UI-Map을 사용하기 위해 반드시 필요한 태그-->
  <script src="http://maps.google.com/maps/api/js?sensor=true" type="text/javascript"></script>
  <script src="web/jquery.fn.gmap.js" type="text/javascript"></script>
  <script src="ui/jquery.ui.map.extensions.js" type="text/javascript"></script>

  <script type="text/javascript">
    $(document).ready(function(){
      //초기 위치를 안산대학교로 설정
      var StartLatLng = new google.maps.LatLng(37.30981, 126.87560);
      $('#map_canvas').gmap({ "center" : StartLatLng , "zoom" : 16});
    });
  </script>
  <style type="text/css">
    .map_style {
      padding:0px;
      height:480px;
      width:100%; }
  </style>
</head>
<body>
  <div data-role="page" id="page" class="map_style">
    <div data-role="header">
      <h1>캠퍼스 맵</h1>
    </div>
    <div data-role="content" class="map_style">
      <div id="map_canvas" class="map_style"></div>
    </div>
  </div>
</body>
</html>
```

ex16-02.html (현재 내 위치 찾아보기)

```
<html>
<head>
  <meta charset="euc-kr">
  <title>현재 내 위치 찾아보기</title>
  <meta name="viewport"
    content="width=device-width, initial-scale=1.0, minimum-scale=1.0, maximum-scale=1.0, user-scalable=no" />

  <!-- JQuery Mobile을 사용하기 위해 반드시 필요한 태그-->
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0rc2/jquery.mobile-1.0rc2.min.css"/>
  <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0rc2/jquery.mobile-1.0rc2.min.js"></script>
```

```

<!-- JQuery-UI-Map을 사용하기 위해 반드시 필요한 태그-->
<script src="http://maps.google.com/maps/api/js?sensor=true" type="text/javascript"></script>
<script src="web/jquery.fn.gmap.js" type="text/javascript"></script>
<script src="ui/jquery.ui.map.extensions.js" type="text/javascript"></script>

<script type="text/javascript">
    // $(document).ready() 시작
    $(document).ready(function(){
        //초기 위치를 안산대학교로 설정
        var StartLatLng = new google.maps.LatLng(37.30981, 126.87560);
        $("#map_canvas").gmap({ "center": StartLatLng, "zoom":16});

        // $("#current_location").click() 시작
        $("#current_location").click(function(){
            //$("#map_canvas").gmap() 끝
            $("#map_canvas").gmap("getCurrentPosition" , function(position, status) {
                if( status === "OK" ) {
                    var latlng = new google.maps.LatLng(position.coords.latitude, position.coords.longitude)
                    var markerBlue = "http://www.google.com/intl/en_us/mapfiles/ms/icons/blue-dot.png";
                    $("#map_canvas").gmap("get", "map").panTo(latlng);
                    $("#map_canvas").gmap("addMarker", { "position": latlng, "icon": markerBlue});
                }else {
                    alert("현재위치를 찾을 수 없습니다. ");
                }
            });
            //$("#map_canvas").gmap() 끝
        });
        // $("#current_location").click() 끝
    });
    // $(document).ready() 끝
</script>

<style type="text/css">
    .map_style {
        padding: 0px;
        height: 100%;
        width: 100%;}
</style>
</head>

<body>
    <!-- 페이지 시작-->
    <div data-role="page" id="page" data-fullscreen="true" class="map_style">
        <!-- header 시작-->
        <div data-role="header" data-position="fixed" >
            <h1>캠퍼스 맵</h1>
        </div>
        <!-- header 종료-->

        <!-- content 시작 -->
        <div data-role="content" class="map_style">
            <div id="map_canvas" class="map_style"></div>
        </div>
        <!-- content 종료-->

        <div data-role="footer" data-position="fixed">
            <!-- navbar 종료-->
            <div data-role="navbar">
                <ul>
                    <li>
                        <a href="#" data-icon="search" id="current_location">현재위치 찾기</a>
                    </li>
                </ul>
            </div>
            <!-- navbar 종료-->
        </div>
    </div>
    <!-- 페이지 종료-->
</body>
</html>

```

13.9 Web SQL Database

websql.html (Web SQL Database 실습)

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1" charset="euc-kr">
  <script type="text/javascript">
    var db;
    function winLoad() {
      if(window.openDatabase == undefined) {
        alert("이 웹브라우저는 Database를 지원하지 않습니다.");
        return;
      }
      db = window.openDatabase("Address","1.0","test",1024*1024);
      db.transaction(function(dbHelper) {
        dbHelper.executeSql("CREATE TABLE IF NOT EXISTS students
                               (num INTEGER primary key autoincrement, name TEXT not null, addr TEXT)");
      });
      showData();
    }
    function insert() {
      var nameObj = frm.name.value;
      var addrObj = frm.addr.value;
      if(nameObj == "" || nameObj == null) {
        alert("이름을 입력하세요!");
        return;
      }
      db.transaction(function(dbHelper) {
        dbHelper.executeSql("INSERT INTO students VALUES(null,?,?)",[nameObj, addrObj])
      });
    }
    function showData() {
      db.transaction(function(dbHelper) {
        dbHelper.executeSql("select * from students",[], success, function() {alert("실패");} );
        frm.name.focus()
      });
    }
    function success(dbHelper, rs) {
      var content="";
      var stdList=document.getElementById("stdList");
      for(var i=0;i<rs.rows.length;i++) {
        var row = rs.rows.item(i);
        content = content + "<tr><td width=40>" + row["num"] + "</td><td width=60>" + row["name"] +
                               "</td><td width=200>" + row["addr"] + "</td></tr>";
      }
      if(content=="") {
        stdList.innerHTML="<tr><td width=300>데이터가 존재하지 않습니다.</td></tr>";
      } else { stdList.innerHTML=content; }
    }
    function stdDelete() {
      var num=document.getElementById("num").value;
      db.transaction (function (dbHelper) {
        var sql = "DELETE FROM students WHERE num = ?";
        dbHelper.executeSql(sql,[num],undefined,function() { alert("삭제성공");},function() { alert("삭제실패"); });
      });
      showData();
    }
  </script>
</head>
<body onLoad="winLoad();">
  <h1>주소 관리 화면</h1>
  <form name="frm">
    이름: <input type="text" id="name"><br/>
    주소: <input type="text" id="addr"><br/>
    <button onclick="insert()">추가</button><br/>
    <table id="stdList" border=1></table><br/>
    <input type="text" id="num"><br/>
    <button onclick="stdDelete()">삭제</button>
  </form>
</body>
</html>
```

13.10 호스팅 업체 등록하기

1. www.dothome.co.kr에 접속하여 닷홈에 회원 가입을 한다.
2. 무료 호스팅을 신청한다.
3. 알FTP 프로그램을 설치하여 웹서버에 접속한다.
4. hd9536(아이디).dothome.co.kr/myadmin 으로 데이터베이스 서버에 접속한다.

13.11 전화관리 프로그램 (web sql 데이터베이스 사용)

tel_list.html

```
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-scale=1" charset="euc-kr">
  <title>전화번호 관리</title>
  <link rel="stylesheet" href="themes/my-theme.min.css"/>
  <link rel="stylesheet" href="http://code.jquery.com/mobile/1.0rc1/jquery.mobile-1.0rc1.min.css"/>
  <script src="http://code.jquery.com/jquery-1.6.4.min.js"></script>
  <script src="http://code.jquery.com/mobile/1.0rc1/jquery.mobile-1.0rc1.min.js"></script>
</style>
  #strlist {
    font-family:"맑은 고딕";
    width:100%;
    padding:0px;
    margin:0px;
  }
  #telList { font-family:"맑은 고딕"; }
  h5 { cursor:pointer; }
</style>
<script type="text/javascript">
  var db;

  function winLoad(){
    if(window.openDatabase == undefined) {
      alert("이 웹브라우저는 Database를 지원하지 않습니다.");
      return;
    }

    db = window.openDatabase("Telephone","1.0","Telephone Database",1024*1024);
    db.transaction(function(dbHelper) {
      dbHelper.executeSql("CREATE TABLE IF NOT EXISTS
                           tel (num INTEGER primary key autoincrement, name TEXT not null, tel TEXT)");
    });
    showData();
  }

  function showData(){
    db.transaction(function(dbHelper) {
      dbHelper.executeSql("select * from tel",[], successList, function() {
        alert("실패");
      });
    });
  }

  function successList(dbHelper, rs) {
    var content="<hr/>";
    var telList=document.getElementById("telList");

    for(var i=0;i<rs.rows.length;i++) {
      var row = rs.rows.item(i);
      content= content +
        "<h5 id='strlist' onClick='frmEdit(\"+row[\"num\"]+\")'>\" + row[\"name\"] + \" : \" + row[\"tel\"] + \"</h5><hr/>\";
    }

    if(rs.rows.length==0) {
      telList.innerHTML="<h3>데이터가 존재하지 않습니다.</h3>";
    } else {
      telList.innerHTML=content;
    }
  }
</script>
</body>
</html>
```



```

function insert(){
    var nameObj = frm.name.value;
    var telObj = frm.tel.value;
    if(nameObj == "" || nameObj == null) {
        alert("이름을 입력하세요!");
        return;
    }

    db.transaction(function(dbHelper){
        dbHelper.executeSql("INSERT INTO tel VALUES(null,?,?)",[nameObj, telObj])
    });
    alert("저장되었습니다.");
    showData();
    frm.name.value="";
    frm.tel.value="";
}

function frmEdit(num){
    db.transaction(function(dbHelper) {
        dbHelper.executeSql("select * from tel where num="+num,[], successEdit, function() {
            alert("실패");
        });
    });
}

function successEdit(dbHelper, rs){
    window.location.href="#edit";
    var row = rs.rows.item(0);

    frmedit.num.value=row["num"];
    frmedit.name.value=row["name"];
    frmedit.tel.value=row["tel"];
}

function telEdit(){
    num=frmedit.num.value;
    name=frmedit.name.value;
    tel=frmedit.tel.value;

    db.transaction(function(dbHelper){
        dbHelper.executeSql("update tel set name="+name+", tel="+tel+" where num="+num);
    });
    alert("수정되었습니다.");
    showData();
}

function telDelete(){
    var num=frmedit.num.value;
    db.transaction (function (dbHelper) {
        var sql = "DELETE FROM tel WHERE num = ?";
        dbHelper.executeSql(sql,[num], undefined,function() {
            alert("삭제 성공");
        }, function() {
            alert("삭제 실패");
        });
    });

    alert("삭제되었습니다.");
    showData();
    window.location.href="#main";
}
</script>
</head>

```

```

<body onLoad="winLoad();" data-theme="a">
  <div data-role="page" id="main" >
    <div data-role="header" data-theme="b" >
      <a href="#insert" data-icon="pluse" class="ui-btn-left" rel="external">입력</a>
      <h1>전화번호목록</h1>
    </div>
    <div data-role="content" data-theme="c" class="content_style" >
      <div id="telList"></div>
    </div>
    <div data-role="footer" data-theme="b">
      <h2>인천직업능력교육원</h2>
    </div>
  </div>
  <div data-role="page" id="insert" >
    <div data-role="header" data-theme="b" >
      <a href="#main" data-icon="grid" class="ui-btn-left" rel="external">목록</a>
      <h1>전화번호입력</h1>
    </div>
    <div data-role="content" data-theme="c" class="content_style" >
      <div id="telList">
        <form name="frm">
          <label>이름</label>
          <input type="text" name="name" value=""/>
          <label>전화</label>
          <input type="text" name="tel" value=""/><br/>
          <input type="button" value="저장하기" data-theme="b" onClick="insert();"/>
        </form>
      </div>
    </div>
    <div data-role="footer" data-theme="b">
      <h2>인천직업능력교육원</h2>
    </div>
  </div>
  <div data-role="page" id="edit" >
    <div data-role="header" data-theme="b" >
      <a href="#main" data-icon="grid" class="ui-btn-left">목록</a>
      <a href="#" data-icon="delete" class="ui-btn-right" onClick="telDelete()">삭제</a>
      <h1>전화번호수정</h1>
    </div>
    <div data-role="content" data-theme="c" class="content_style" >
      <div id="telList">
        <form name="frmedit">
          <label>번호</label>
          <input type="text" name="num" readonly/>
          <label>이름</label>
          <input type="text" name="name"/>
          <label>전화</label>
          <input type="text" name="tel"/><br/>
          <input type="button" value="저장하기" data-theme="b" onClick="telEdit();"/>
        </form>
      </div>
    </div>
    <div data-role="footer" data-theme="b"><h2>인천직업능력교육원</h2></div>
  </div>
</body>
</html>

```

14. 기타

14.1 부트스트랩 정의

- 다양한 웹 라이브러리를 모아놓은 웹 사이트 개발 프레임워크로서 디자인을 잘 못하는 프로그래머도 쉽게 홈페이지를 개발할 수 있도록 도와준다.
- 반응형 웹 페이지(테스탑, 모바일, 태블릿, 디바이스에 구애받지 않고 같은 내용을 출력) 제작가능하다
- 라이선스에 구애받지 않고 오픈소스를 제공받는다.

14.2 무료 템플릿 사이트

http://startbootstrap.com로 통하여 사용한다.