

## ❖ 하이퍼링크

### ● <a> 태그

- 다른 사이트 혹은 같은 문서 내 다른 위치로 이동할 때 사용
- href 속성값에 URL을 입력해 원하는 곳으로 이동

**<a href="URL">**

표기 유형	표기 예
절대경로 URL 표기	href=" <u>http://www.w3.org/index.html</u> "
상재경로 URL 표기	href="hello.html" * <base> 태그에 기본 URL 지정
페이지 내 요소 위치 표기	href="#top"
프로토콜 URL 표기	https://, ftp://, mailto:, file: 등
스크립트 표기	href="javascript:alert('Hello');"

URL 표기 유형과 표기 예

## • 인터넷의 시작

- 미국의 신기술을 연구하는 ARPA에서 현재 웹의 모태가 되는 ARPANET을 1969년에 개발

### ARPA

- 고등 연구 계획국 (ARPA 또는 ARP, Advanced Research Projects Agency)
- 이후 방위 고등 연구 계획국(DARPA , Defense Advanced Research Projects Agency)로 개칭
- 미국 국방성의 연구, 개발 부문을 담당

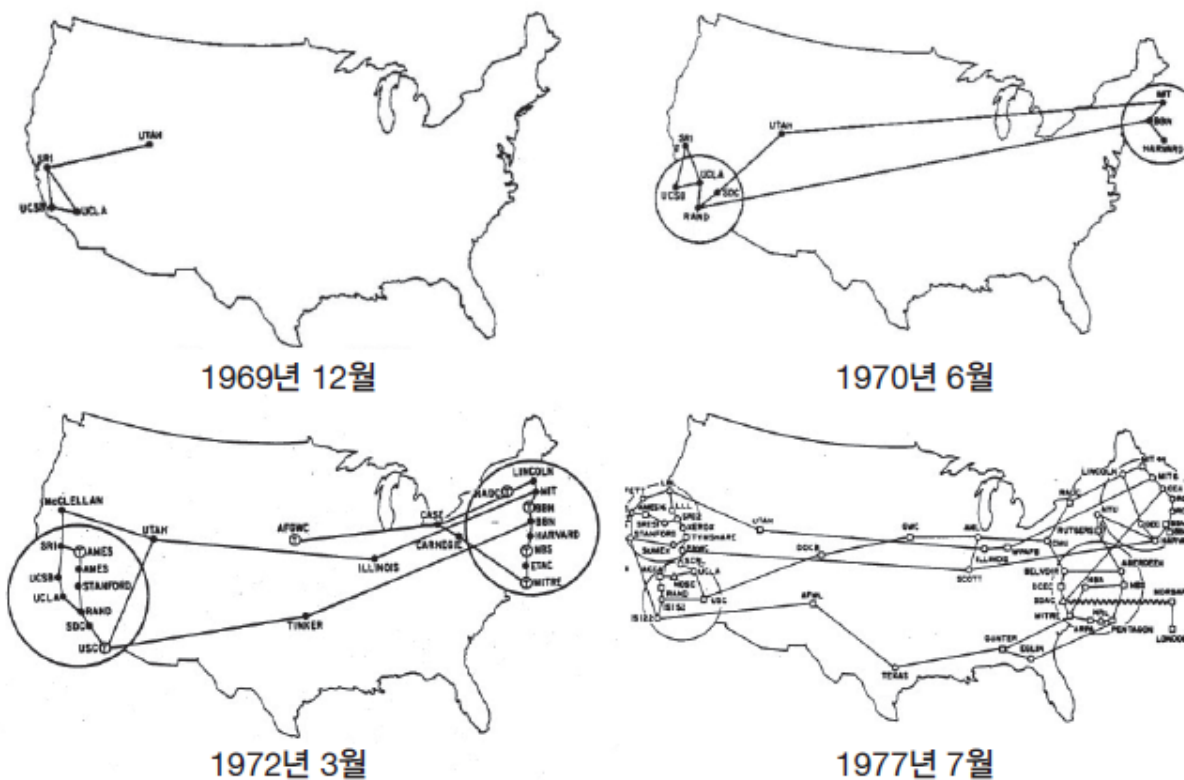


그림 1-1 ARPANET 연결 상태 변화

# HTML



## HTML5

## • 인터넷의 시작

- 미국의 신기술을 연구하는 ARPA에서 현재 웹의 모태가 되는 ARPANET을 1969년에 개발

### ARPA

- 고등 연구 계획국 (ARPA 또는 ARP, Advanced Research Projects Agency)
- 이후 방위 고등 연구 계획국(DARPA, Defense Advanced Research Projects Agency)로 개칭
- 미국 국방성의 연구, 개발 부문을 담당

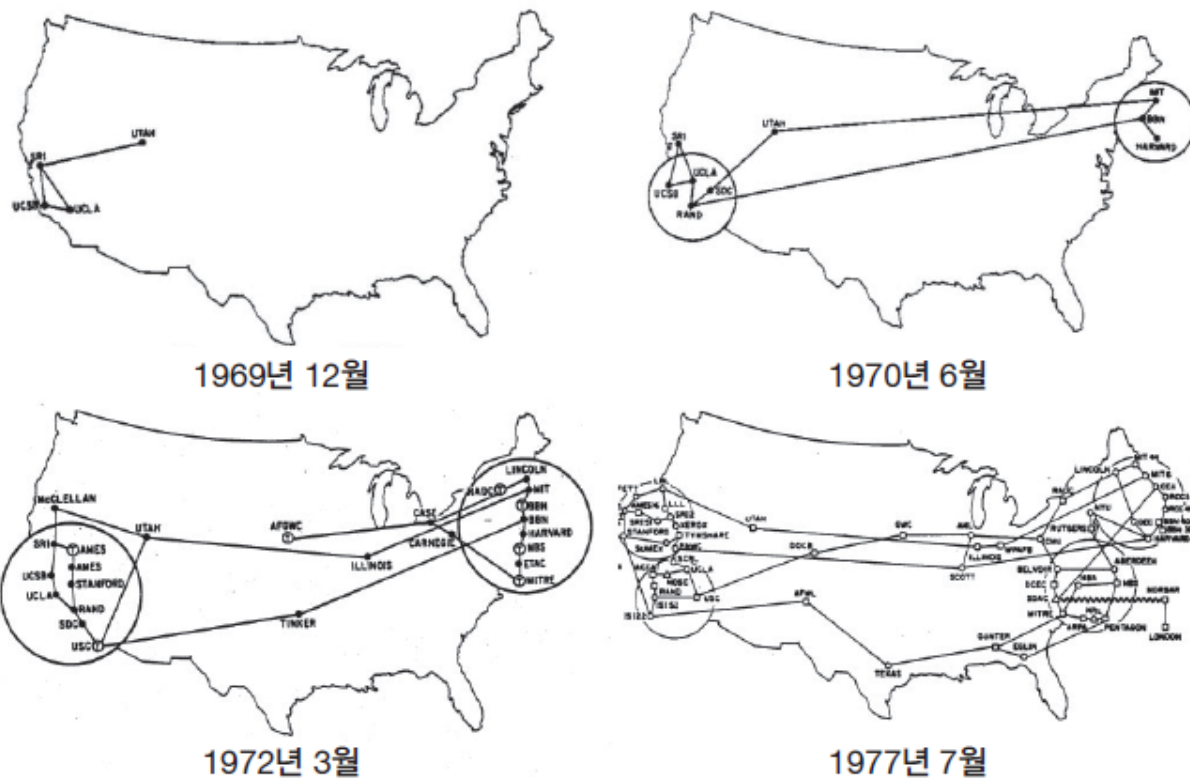


그림 1-1 ARPANET 연결 상태 변화

아파넷(ARPAnet)은 미국 국방부의 고등 연구 계획국 (Advanced Research Project Agency : 약칭 ARPA, 문화어: 현대연구계획국망)의 주도하에 만들어진 **세계 최초의 패킷 스위칭 네트워크**로 1969년에 시작되었으며, 현재의 인터넷의 원형으로 알려져 있다. 아파넷이라는 글자는 모두 대문자로 ARPANET로 표시되는 경우가 많다.

당시엔 회로 스위칭이 많이 쓰이고 있었지만, **아파넷의 발족한 뒤로 이제까지 패킷 스위칭이 데이터와 음성을 주고 받는 데에 널리 쓰이고 있다.**

발족 당시의 노드로는 UCLA, 캘리포니아 대학교 산타바바라, 스탠퍼드 연구소(SRI), 유타 대학교의 네 군데가 있었다.

아파넷은 **NCP(Network Control Program)라는 전송 통신규약(프로토콜)을 사용**하였으나, **1983년 TCP/IP가 이를 대체**하며 지금의 인터넷으로 진화하게 되었다.

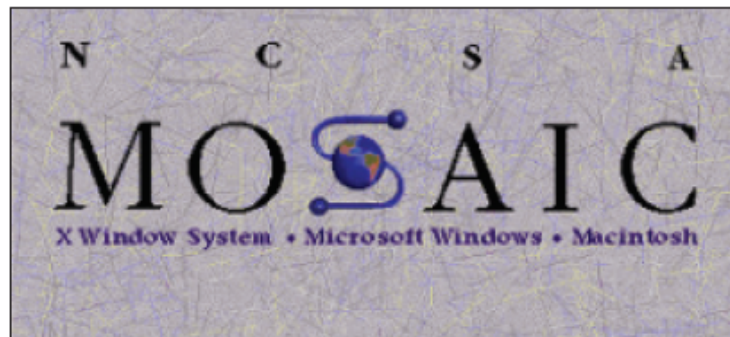
- **HTML** (Hyper Text Markup Language)
  - 유럽 입자 물리 연구소(CERN)의 계약자였었던 물리학자 팀 버너스리가 1980년 원형(인콰이어) 제안
  - HTML 최초의 일반 공개 설명은 1991년 말에 버너스리가 처음으로 인터넷에서 문서를 "HTML 태그"(HTML tag)로 부르면서 시작 - HTML 1
- **팀 버너스리 경** (Sir Tim Berners-Lee, OM, KBE, FRS, 1955년 6월 8일 ~ )
  - 영국의 물리학자, 컴퓨터 과학자
  - **W3C**<sup>World Wide Web Consortium</sup> 창설
    - HTML 표준을 비롯한 웹 표준안을 제작, 제안하는 일을 하는 국제적인 웹 표준화 단체
  - Hyper Text 라는 합성어의 시초
  - 1989년 월드 와이드 웹의 하이퍼텍스트 시스템을 고안하여 개발했다.
  - 인터넷의 기반을 닦은 여러 공로로 웹의 아버지라고 불리는 인물 중 하나이다.
  - URL, HTTP, HTML 최초 설계

표 1-1 HTML 버전(표준안) 발표 시기

버전	발표 연도
HTML1	1991년 10월
HTML2	1995년 11월
HTML3	1997년 1월
HTML4	1997년 12월
HTML5	2014년 10월

## • 1차 웹 브라우저 전쟁

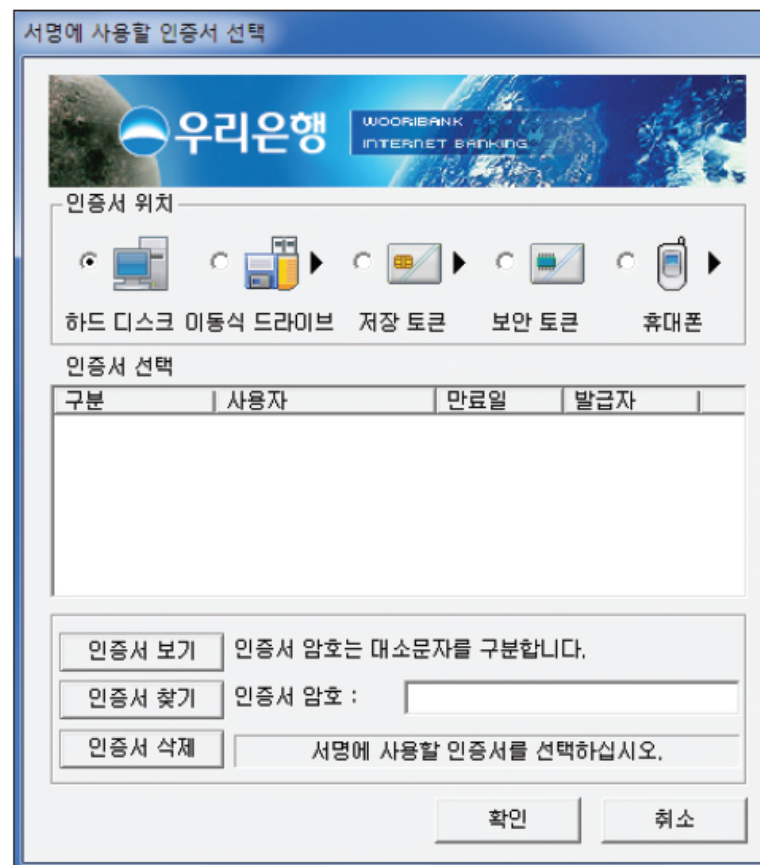
- NCSA (국립 슈퍼 컴퓨터 응용 센터 혹은 미국 국립 슈퍼컴퓨팅 응용 연구소, 전미 슈퍼컴퓨터 응용 연구소)  
: 웹 브라우저 모자이크를 최초로 발표
  - 팀 버너스리가 만든 기존의 WorldWideWeb 브라우저가 텍스트 위주였던 것과 달리, 모자이크는 이미지를 표시할 수 있는 최초의 그래픽 웹 브라우저였다.



- 넷스케이프 커뮤니케이션(마크 안데르센) : 웹 브라우저 넷스케이프 발표
- 마이크로소프트 : 인터넷 익스플로러 발표
  - 인터넷 익스플로러를 윈도우 운영체제에 강제로 설치해 점유율을 높임
- 아메리카 온라인AOL : 넷스케이프 커뮤니케이션을 인수, 점차 하락 후 붕괴  
넷스케이프 출신의 일부 개발자들은 별도로 모질라 커뮤니티를 만들고, 2002년 파이어폭스 웹 브라우저를 출시했다.

## • 플러그인

- 웹 브라우저와 연동되는 프로그램을 사용자의 PC에 추가로 설치해 웹 브라우저의 기능을 확장하는 방법
- Future Splash Animator
- 마이크로소프트의 액티브X



## • 2000년 초반

- 액티브X 기반으로 웹 응용 프로그램 제작
- 플래시 기반으로 애니메이션 제작

❖ W3C 와 마이크로소프트가 정한 기능 - 기타 브라우저에서는 작동 안함

- **2차 웹 브라우저 전쟁**

- 플러그인이 인터넷 익스플로러만 지원
- 점유율이 높은 인터넷 익스플로러가 표준 웹 브라우저가 되나 액티브X 같은 플러그인이 들어가면서 웹 사이트는 점점 무거워짐
- **WHATWG** Web Hypertext Application Technology Working Group
  - 웹 브라우저 제공 기업 (애플, 모질라, 오페라 소프트웨어) 에서 새로운 웹 표준 기관을 설립
  - Web Application 1.0 표준 작성
  - W3C는 웹 표준으로 Web Application 1.0 표준 채택, HTML5 표준으로 변경
  - 인터넷 익스플로러만 최신 표준을 지원하지 못하는 현상 초래
    - > 모든 웹 브라우저가 빠른 속도로 업데이트 되고 있음



## ❖ 용어 정리

### ◆ TCP/IP :

Transmission Control Protocol (전송 제어 규약) 과 Internet Protocol (인터넷 규약) 은 데이터가 어떻게 웹을 건너 여행해야 하는지 정의하는 통신 규약

- 주문을 하고, 상점에 가고, 또 상품을 살 수 있게 해주는 운송 장치와 같다.

ex] 차 또는 자전거 (또는 여러분의 두 다리) 와 같다.

### ◆ DNS :

Domain Name System Servers (도메인 이름 시스템 서버) 는 웹사이트를 위한 주소록  
브라우저에 웹 주소를 입력할 때, 브라우저는 그 웹사이트를 검색하기 전에 DNS 를 살펴본다.

브라우저는 HTTP 메시지를 올바른 장소로 전송하기 위해

그 웹사이트가 있는 서버가 어떤것인지 찾아야 한다.

### ◆ HTTP:

Hypertext Transfer Protocol (하이퍼텍스트 전송 규약) 은

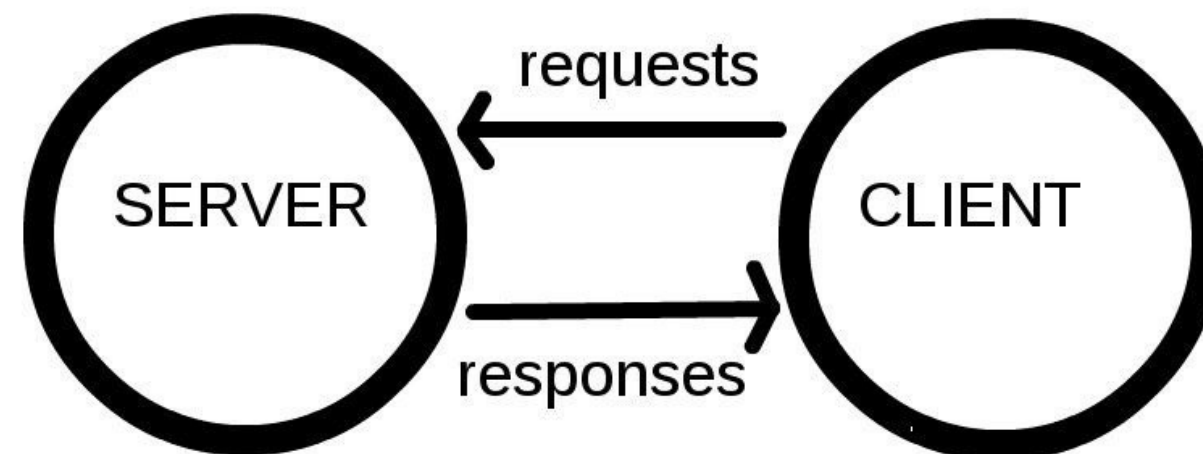
클라이언트와 서버가 서로 통신할 수 있게 하기 위한 언어를 정의하는 어플리케이션 규약 입니다.

상품을 주문하기 위해 여러분이 사용하는 언어와 같다.

## ❖ 웹의 동작 원리

### ◆ 요청(클라이언트)과 응답(서버)의 과정

- ❖ 클라이언트(사용자) : 서버에 요청하는 쪽
- ❖ 서버(제공자) : 요청에 응답하는 쪽



## ❖ 웹의 동작 방식

- ◆ 브라우저는 DNS 서버로 가서 웹사이트가 있는 **서버의 진짜 주소를 찾는다.**
- ◆ 그 다음 브라우저는 서버에게 웹사이트의 사본을 클라이언트에게 보내달라는 **HTTP 요청 메시지를 서버로 전송**합니다.  
이 메시지, 그리고 클라이언트와 서버 사이에 전송된 **모든 데이터는 TCP/IP 연결을 통해서 전송**됩니다.
- ◆ 이 메시지를 받은 서버는 클라이언트의 요청을 승인하고,  
허가 메시지를 클라이언트에게 전송합니다.  
그 다음 서버는 웹사이트의 파일들을 **데이터 패킷**이라 불리는 작은 일련의 덩어리들로 브라우저에 전송하기 시작합니다.
- ◆ 브라우저는 패킷들을 완전한 웹 사이트(HTML 문서)로 조립하고, 보여준다.

## ❖ 웹의 동작 방식

### ◆ 서버 프로그램

- 자바, C#, 루비, 파이썬, 자바스크립트
  - 웹 프레임워크(ASP.NET, JSP, PHP 등)
  - MVC 프레임워크(ASP.NET MVC, Spring MVC, Ruby on Rails 등)
  - 비동기 프레임워크(Node.js Express, Jetty )

### ◆ 클라이언트 프로그램

- 반드시 HTML, CSS, 자바스크립트로 개발

## ❖ 보조 웹 표준 기술

### HTML



#### \* HTML5

- ◆ 큰 의미 : 웹 표준 기술을 총칭
- ◆ 작은 의미 : 웹 문서의 문법을 의미

### CSS



#### \* CSS3

- ◆ CSSCascading Style Sheets : HTML 문서를 표현하는 방법을 기술하는 언어
- ◆ 스타일시트는 현대 웹 페이지에서 매우 중요한 역할임

### JS



#### \* 자바스크립트

- ◆ 표준 명칭 : ECMAScript5
- ◆ HTML에서 사용자 반응 등을 처리하는 데 사용하는 프로그래밍 언어
- ◆ 현대에는 서버는 물론 로봇 개발에도 사용됨

## ❖ WEB 2.0

- ◆ 웹 2.0(Web 2.0)이란 개방, 참여, 공유의 정신을 바탕으로 사용자가 직접 정보를 생산하여 쌍방향으로 소통하는 웹 기술을 말한다. 웹 1.0이 인터넷을 통해 일방적으로 정보를 보여주었다면, 웹 2.0은 사용자가 직접 콘텐츠를 생산하여 쌍방향으로 소통할 수 있다. 게시판, 댓글, 블로그, UCC, 지식백과 등이 있다.
- ◆ 단순한 웹사이트의 집합체를 웹 1.0으로 보고, 웹 애플리케이션을 제공하는 하나의 플랫폼으로의 발전을 웹 2.0이 데스크톱 컴퓨터의 응용 프로그램을 대체할 것으로 예견하고 있다. 사용자들의 참여, 공유, 개방을 유도산출물을 공유하고, 외부에 개방할 수 있도록 하는 서비스라고 할 수 있다.

## ❖ HTML5 주요 기능

### \* 멀티미디어 기능

- ◆ HTML5는 플러그인 없이 음악과 동영상을 재생

### \* 그래픽 기능

- ◆ HTML5 : 2차원, 3차원 그래픽 구현 가능
  - 2차원 그래픽을 구현 방법
    - HTML 태그를 사용해 2차원 벡터 그래픽 구현
    - 자바스크립트 캔버스를 사용해 2차원 래스터 그래픽 구현
  - 3차원 그래픽 구현 방법
    - CSS3를 사용해 3차원 그래픽 구현
    - 자바스크립트 WebGL을 사용해 3차원 그래픽 구현

### \* 통신

- ◆ 서버와 실시간으로 쌍방향 통신 가능  
웹에서 실시간 채팅이나 실시간 온라인 게임을 할 수 있음
- ◆ 온라인 게임 퀘이크 2
  - WebGL과 HTML5 웹 소켓을 사용 – 다른 사용자와 실시간으로 온라인 게임 가능

### \* 장치 접근

- ◆ 장치에 접근해 정보와 기능을 사용
  - 예) 스마트폰 배터리 잔량 정보, 스마트폰 GPS 사용, 진동 벨
  - AT&T HTML5 SDK : 스마트폰의 내장 기능을 이용하는 응용 프로그램 개발 가능

## ❖ HTML5 주요 기능

### \* 시맨틱

- ◆ 검색 엔진 같은 프로그램이 정보를 분석, 자료를 검색 및 처리해서 제공하는 지능형 웹

### \* CSS3 스타일시트

- ◆ CSS3 스타일시트를 사용하면 3차원 변환과 애니메이션 효과 적용 가능

### \* 웹의 성능 극대화 및 통합

- ◆ HTML5의 추가 기능으로 기존에 이용하던 웹의 성능을 극대화할 수 있음
  - 웹 워커를 이용하면 사용자 화면을 멈추지 않으면서 연산도 처리 가능
  - HTML5 표준을 적용한 웹 브라우저를 사용해 간단한 워드 작업이나 게임가능

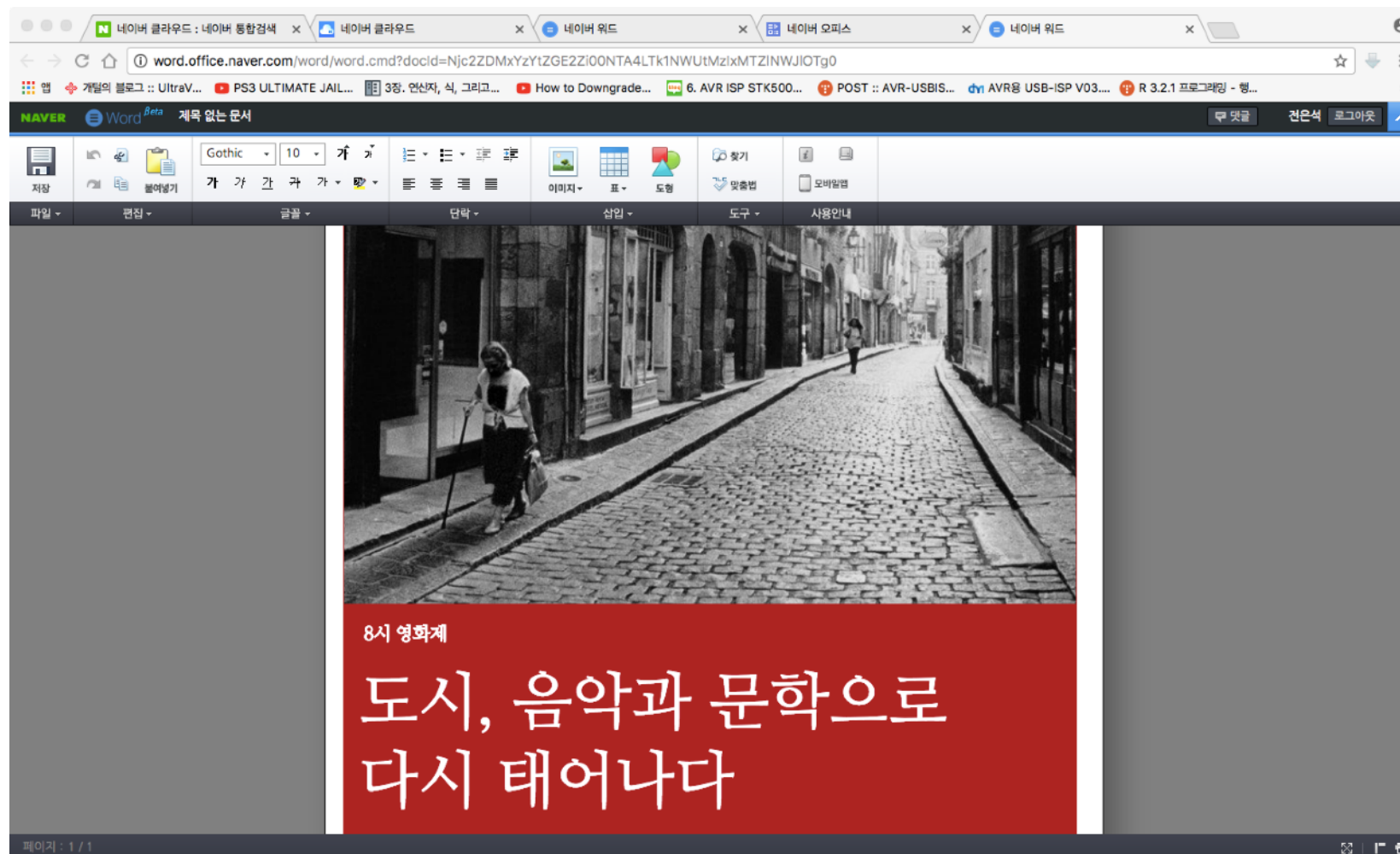
### \* 오프라인 및 저장소

- ◆ HTML5는 인터넷이 연결되지 않아도 응용 프로그램 동작 가능
- ◆ 오프라인 Gmail



## ❖ HTML5 주요 기능

- ◆ HTML5부터는 웹 페이지 자체가 하나의 응용 프로그램이 됨
  - 예) 네이버오피스



- ◆ 하이브리드 응용 프로그램

## ❖ HTML5 개발 환경 구축

### ◆ Google Chrome 설치

<https://www.google.com/chrome/browser/desktop/index.html>

### ◆ Firefox 설치

<https://www.mozilla.org/ko/firefox/new/>

### ◆ Font 설치 - 네이버 코딩용 폰트

<https://github.com/naver/d2codingfont>

<https://github.com/naver/nanumfont>

### ◆ Atom 설치( 또는 Eclipse )

<https://atom.io/>

## ❖ HTML (Hyper Text Markup Language) 문서 형식

```

1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2 <HTML>
3 <HEAD>
4   ...
5 </HEAD>
6 <BODY>
7   ...
8 </BODY>
9 </HTML>

```

### 1. 문서 형식 선언

<!DOCTYPE	HTML	PUBLIC	"-//W3C//DTD HTML 4.0 Transitional//EN"	[URI] >
문서형식 선언	최상위요소 (문서형태)	구조키워드 (PUBLIC/ SYSTEM)	FPI ( Formal Public Identifier ) - : DTD와 공식 표준과의 연결 내용을 지정 자신만의 DTD에서는 "-" 지정 단체에서 승인한 DTD라면 "+" 지정 W3C : 그룹 또는 담당자 DTD HTML 4.0 Transitionl : 사용하고 있는 문서의 형식 EN : 사용언어 지정 FPI의 필드들은 반드시 // 로 구분	DTD 정의 URI URI를 생략할 경우 브라우저의 기본 셋팅으로 해석한다. 따라서 브라우저마다 해석 방법이 달라질 수 있다.

## ❖ HTML (Hyper Text Markup Language) 문서 형식

```
1 <!DOCTYPE HTML>
2 <HTML>
3 <HEAD>
4   ...
5 </HEAD>
6 <BODY>
7   ...
8 </BODY>
9 </HTML>
```

### 1. 문서 형식 선언

<!DOCTYPE	HTML>
문서형식 선언	최상위요소 (문서형태)

HTML5은 구조상 DTD로 충분히 정의될 수 없으나 표준 문서를 명시할 필요성이 있으므로 위와 같은 문서 형식 선언을 갖는다.

# HTML

## (HyperText Markup Language)

제목, 단락, 목록 등과 같은 본문을 위한 구조적 의미를 나타내는 것뿐만 아니라  
링크, 인용과 그 밖의 항목으로 구조적 문서를 만들 수 있는 방법을 제공

- 1980년, 유럽 입자 물리 연구소(CERN)의 계약자였었던 물리학자 팀 버너스리가 HTML의 원형인 인콰이어를 제안 (연구원들이 문서를 이용하고 공유하기 위한 체계)
- 1989년에 팀 버너스리는 인터넷 기반 하이퍼텍스트 체계를 제안하는 메모를 작성
- HTML - 월드 와이드 웹을 개발해 1991년 배포  
1991년 말에 버너스리가 처음으로 인터넷에서 문서를 "HTML 태그"(HTML tag)로 부르면서 시작
- 1993년 - 월드 와이드 웹 사용인구가 폭발적으로 급증하자 팀 버너스 리가 소스코드를 모든 사람이 사용할 수 있도록 공개
- 1994년 - 팀 버너스리 W3C(World Wide Web Consortium) 창설
- 1995년 - HTML 2.0을 완성
- 2008년 - 1월 HTML 5 W3C에 의해 발표

- \* 팀 버너스 리의 이러한 업적을 기리기 위해 주소 시작은 www 로 시작하기로...
- \* 참고 사이트 - <http://www.w3schools.com/>

## DTD

- 문서 형식 정의(Document Type Definition, DTD)는 컴퓨터 용어
- SGML 계열의 마크업 언어에서 문서 형식을 정의하는 것

## 마크업 언어 (Markup Language)

- 태그 등을 이용하여 문서나 데이터의 구조를 명기하는 언어의 한 가지
- 종류 :

SGML, HTML, XML, XHTML, SVG, MathML, MXML, XAML, SAMI 등 ...

## DTD

SGML(Standard Generalized Markup Language)은 문서용 마크업 언어를 정의하기 위한 메타 언어이다.

IBM에서 1960년대에 개발한 GML(Generalized Markup Language)의 후속이며, ISO 표준이다.

많은 응용이 가능하도록 다양한 마크업 구문을 제공한다.

SGML 선언을 변경함으로써 꺾쇠 괄호('<', '>')를 사용하지 않는 것도 가능하기 때문에,

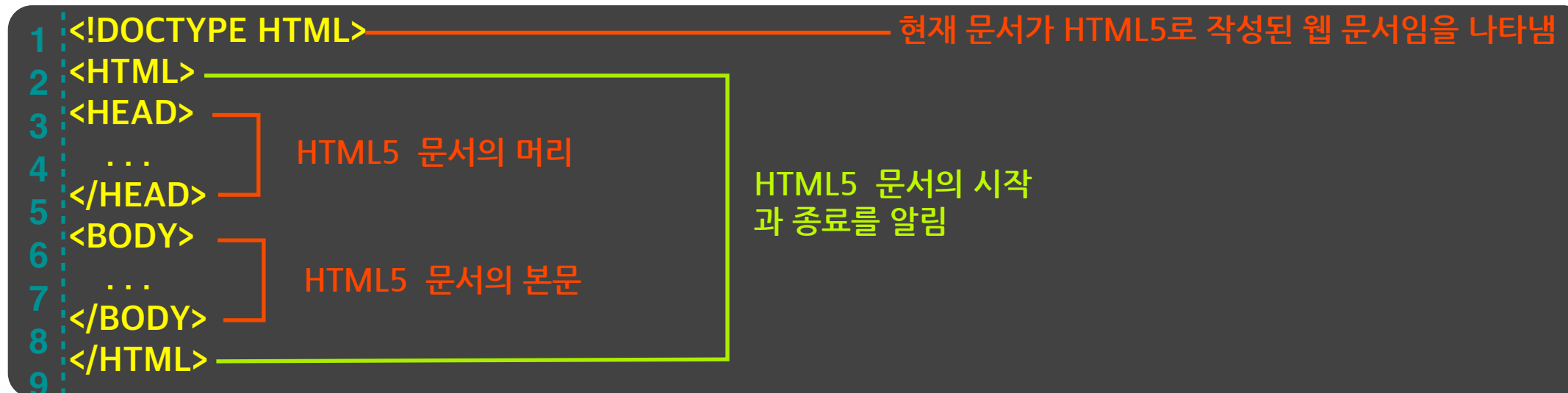
상세 참조 구문(concrete reference syntax)로도 불린다.

SGML은 정부나 항공우주 기업의 대규모 계획 사업 과정에서 기계 판독형(machine-readable) 문서를 공유할 목적으로, 몇 십 년 이상의 기간 동안 판독 가능하도록 설계되었다.

SGML은 인쇄와 출판 산업에 광범위하게 사용되었지만,

너무 복잡한 이유로 소규모 범용 목적으로 사용하는데 걸림돌이 되었다.

## ❖ HTML5 문서의 구조



### HTML5 문서의 기본 구조

- **<!DOCTYPE html>**
  - HTML5 문서를 선언하는 구문
  - 생략할 수 있지만 하위 호환성을 위해 작성할 것을 권장
- **<html> ~ </html>**
  - HTML5 문서의 시작과 종료를 알림
  - 언어(lang) 속성을 사용하여 주된 언어 값을 설정할 수 있음
- **<head> ~ </head>**
  - 웹 페이지의 정보를 정의
  - <title> 태그와 자바스크립트, 스타일시트(CSS) 등을 정의
- **<body> ~ </body>**
  - HTML5 문서의 본문을 작성

## ❖ <meta> 태그

- ◆ <head>와 </head> 사이에 입력한다.

<meta NAME="Generator" CONTENT="Eclipse" />

-> 웹사이트를 제작할 때 어떤 툴을 사용했는지 표시

<meta NAME="Author" CONTENT="홍길동" />

-> 웹사이트 제작자 표시

<meta NAME="Keywords" CONTENT="HTML meta Tag" />

-> 검색엔진에서 검색할 웹 사이트의 키워드 표시

<meta NAME="Description" CONTENT="메타 태그 설명 페이지" />

-> 웹사이트와 주제를 표시

<meta http-equiv="Content-Type" CONTENT="text/html; charset=UTF-8" />

-> 문자 세트 표시



## ❖ <meta> 태그 기능

### ◆ 웹 브라우저 캐시 기능 무시

`<meta http-equiv="Pragma" CONTENT="No-Cache" />`

또는

`<META http-equiv="Expires" content="-1" />`

또는

`<META http-equiv="Cache-Control" content="No-Cache">`

### ◆ 시간을 설정하여 자동으로 페이지 이동

`<meta http-equiv="refresh" CONTENT="대기시간; URL=웹문서" />`

## ❖ <meta> 태그 기능

### ◆ 장면 전환 효과

<meta http-equiv="Page-Enter"

CONTENT="RevealTrans(Duration=속도, Transition=번호)" />

효과 설정 번호	효과 설명	효과 설정 번호	효과 설명
0	문서 내용을 사각형 모양으로 바깥쪽에서부터 안쪽으로	12	문서 내용을 점을 찍듯이
1	문서 내용을 사각형 모양으로 안쪽에서부터 바깥쪽으로	13	문서 내용을 좌우에서 가운데로
2	문서 내용을 원형 모양으로 바깥쪽에서부터 안쪽으로	14	문서 내용을 가운데서 좌우 쪽으로
3	문서 내용을 원형 모양으로 안쪽에서부터 바깥쪽으로	15	문서 내용을 위아래에서 가운데로
4	문서 내용을 아래쪽에서 위쪽으로	16	문서 내용을 가운데에서 위아래로
5	문서 내용을 위쪽에서 아래쪽으로	17	문서 내용을 오른쪽 위에서 왼쪽 아래로
6	문서 내용을 왼쪽에서 오른쪽으로	18	문서 내용을 오른쪽 아래에서 오른쪽 위로
7	문서 내용을 오른쪽에서 왼쪽으로	19	문서 내용을 왼쪽 위에서 오른쪽 아래로
8	문서 내용을 왼쪽에서 오른쪽으로 부분적으로 끊어서	20	문서 내용을 왼쪽 아래에서 오른쪽 위로
9	문서 내용을 위쪽에서 아래쪽으로 부분적으로 끊어서	21	문서 내용을 가로 방향으로 줄을 긋듯이
10	문서 내용을 왼쪽에서 오른쪽으로 타일 모양으로	22	문서 내용을 세로 방향으로 줄을 긋듯이
11	문서 내용을 위쪽에서 아래쪽으로 타일 모양으로	23	모든 효과를 사용하여 문서 내용을 랜덤으로

## ❖ 메타정보를 정의한 웹 문서 작성하기

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>HTML5 메타정보</title>
5   <meta charset="UTF-8">
6   <meta name="author" content="Hong Gil Dong">
7   <meta name="keywords" content="HTML5, CSS3, JavaScript, JQuery">
8   <meta name="description" content="Web Programming">
9   <meta http-equiv="refresh" content="10, http://github.com/eunsjun/">
10  <base href="http://www.w3.org/" target="_blank">
11 </head>
12 <body>
13   <p>헤드 태그 내 메타정보에는 웹 문서를 만든 이, 검색 시 키워드, 문서에 대한 설명,
14     문서 내 기본 디렉터리 등이 포함됩니다.</p>
15   <p>이 문서는 10초 후 저자 카페로 이동합니다.</p>
16   <a href="">여기를 클릭하면 기본 디렉터리로 설정된 www.w3.org 사이트로 이동합니다.</a>
17 </body>
18 </html>
```

HTML5\_meta01.html

## ❖ HTML5 문서의 작성 규칙

- 태그 이름은 대소문자를 구분하지 않음 - 소문자로 작성할 것을 권장
- 본문 내 연속된 공백이나 줄 바꿈은 하나의 공백으로 처리
- 태그의 포함 관계를 표현하기 위해 들여쓰기 적용
- 태그의 쌍을 겹치지 않고 완벽히 내포시킴

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>잘못된 태그 중첩 문서
5 </head>
6   </title>
7 <body>
8   <p>태그가 중첩되지 않도록 작성해야 합니다.
9   </p>
10 </body>
11 </html>
```

### 잘못된 태그 중첩 예

- 종료 태그를 반드시 사용
  - 일반적인 표기 형식 : <p>~</p>
  - 단축 표기 형식 : <p/>
- 주석은 ‘<!--’로 시작해서 ‘-->’로 끝냄

## ❖ 태그의 분류 - 영역에 의한 분류

- **Block Tag**

- 내용과 상관없이 웹브라우저의 가로폭을 100% 모두 사용하는 태그
- 뒤에 나열된 태그는 반드시 다음 행에 표현된다.
- 예 ] div, p, hr, pre 등

- **Inline Tag**

- 표현되는 내용에 따라 가로사이즈가 결정되는 태그
- 뒤에 나열된 태그는 바로 오른쪽에 표현된다.
- 예 ] span, a, img 등

## ❖ 태그의 분류 - 용도에 의한 분류

### ● 입력 태그

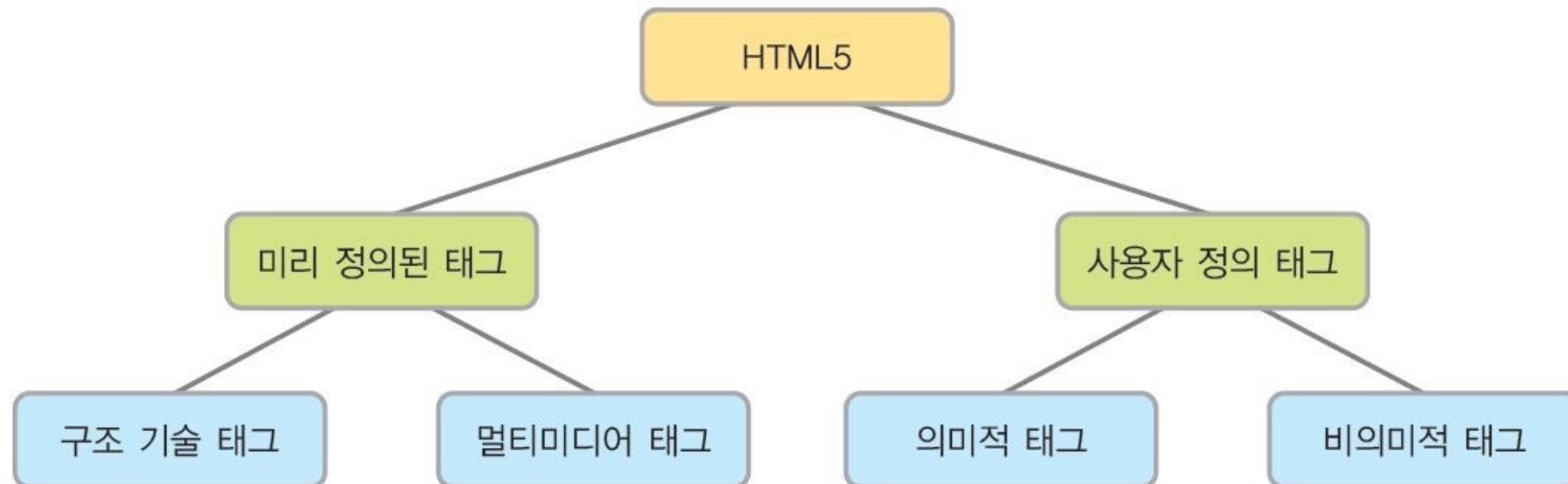
- 사용자가 내용을 입력하거나 선택할 수 있는 태그
- <form> 태그 하위에 기술하여 사용하는 것이 일반적이다.
- 데이터를 서버에 전달할 목적으로 사용된다.
- 반드시 name속성값을 붙여줘야 한다.
- 예 ] input, select, textarea 등

### ● 일반 태그

- 서버에서 내용을 결정해서 화면에 표현할 목적으로 사용되는 태그들
- 서버에 내용을 전달할 수 없는 태그들
- 예 ] span, a, img 등

## ❖ 태그의 분류

- 미리 정의되어 있는 태그
  - 사용자가 변경할 수 없으며 사용 용도에 맞게 사용
- 사용자가 직접 정의하여 사용하는 태그
  - 사용자가 태그를 직접 설계하고 HTML 문서에 포함하여 작성



HTML5 태그의 분류

## ❖ 태그의 분류

- 미리 정의되어 있는 태그

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>미리 정의된 태그</title>
5 </head>
6 <body>
7   <h1>문서의 제목</h1>
8   <hr>문서의 구분선</hr>
9   <p>단락 구분</p>
10  <strong>중요한 문장</strong>
11  <br>줄 바꿈
12 </body>
13 </html>
```

HTML5\_element01.html



## ❖ 태그의 분류

- 사용자 정의 태그 작성 시 유의점
  - 일반적인 의미로 해석할 수 있도록 포함 관계 설정
  - 동등 관계로도 설정 가능



(a) 정상적인 포함 관계



(b) 비정상적인 포함 관계

태그의 포함 관계



태그의 동등 관계

## ❖ 태그의 분류

- 사용자 정의 태그로 웹 문서 작성하기

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>사용자 정의 태그</title>
5 </head>
6 <body>
7   <company>글로벌아이티정보기술
8     <department>교무부</department>
9     <department>행정부
10    <team>전산팀</team>
11    </department>
12  </company>
13 </body>
14 </html>
```

HTML5\_userTag01.html

## ❖ 태그의 분류

- 사용자 정의 태그로 웹 문서 작성하기

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>사용자 정의 태그</title> </head>
5 <body>
6   <company>글로벌아이티정보기술
7   <department>교무부</department>
8   <department>행정부
9     <team>전산팀</team>
10  </department>
11 </company>
12 <p/>
13 <university>한국대학교
14   <major>컴퓨터과학</major>
15   <major>경영정보학</major>
16 </university>
17 </body>
18 </html>
19
```

HTML5\_userTag02.html

## ❖ 태그의 속성

- 속성
  - 태그의 종속적인 정보를 표현하기 위해 사용
  - 태그없이 단독으로 사용할 수 없음
  - 속성="값" 형태로 작성

```
<시작 태그 속성="값">내용</종료 태그>
```

## ❖ 태그의 속성

- 글로벌 속성 - 모든 태그에 공통적으로 사용 가능한 속성

속 성	설 명
accesskey="단축키"	공백으로 구분하여 여러개의 단축키를 지정할 수 있다. 단축키를 작성한 순서대로 순위가 매겨지기 때문에 제일 처음 작성한 단축키가 적용
class="클래스명"	공백으로 구분하여 여러 개의 클래스 명을 지정할 수 있다. 클래스 속성은 동일한 문서 안에 있는 여러개의 태그에 동일한 이름으로 지정할 수 있다. 스타일 시트를 적용할 경우에는 셀렉터로도 이용 가능
contenteditable="편집가능여부"	편집할 수 있는 경우에는 true로 지정하고 빈 문자("")이거나 편집할 수 없는 경우에는 false로 지정
contextmenu="menu 요소의 id 속성값"	메뉴 요소의 id 속성 값을 지정한다.
dir="텍스트 표시 방향"	왼쪽에서 오른쪽인 경우에는 ltr, 오른쪽에서 왼쪽인 경우에는 rtl로 지정
draggable="드래그 가능 여부"	드래그할 수 있는 경우에는 true, 드래그할 수 없는 경우에는 false로 지정
dropzone="드롭한 아이템 처리 방법"	값을 copy로 지정하면 드래그한 데이터가 이 장소에 복사되고, move로 지정하면 드래그한 데이터가 이동한다. 또한 link라고 지정하면 오리지널 데이터와 드롭된 곳 사이에 어떤 연결이 만들어진다.
hidden="hidden"	이 속성이 지정된 요소는 브라우저에 표시되지 않는다. <p hidden> 또는 <p hidden="hidden"> 과 같이 지정
id="이름"	문서 내에서 해당 태그를 유일하게 식별하는 역할. 동일한 문서 내에서 중복 사용 불가
lang="언어코드"	한국어는 ko, 영어는 en, 미국 영어는 en-US, 프랑스어는 fr등 언어의 속성값 지정
spellcheck="철자 체크 여부"	철자 체크를 할 것인지 말 것인지 지정
style="CSS 선언"	세미콜론으로 구분하여 여러 개의 CSS 선언을 지정할 수 있다.
tabindex="이동 순서"	실제로 Tab 키를 이용하는지 아닌지는 환경에 따라 다르다. 값에는 정수를 지정하며, 값이 작은 것에서 큰것으로 이동된다. 값이 0으로 지정된 요소와 tabindex 속성이 지정되어 있 지 않은 요소는 1 이상의 값이 지정 되어 있는 요소 다음으로 포커스가 이동. 또한 마이너스값을 지정한 경우, 포커스는 가능하지만 Tab키에 의한 이동 대상은 되지 않는다.

## ❖ 태그의 속성

- 미리 정의된 속성
  - 특정 태그에만 사용할 수 있도록 미리 정의된 속성

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>속성 사용</title>
5 </head>
6 <body>
7   <h1>미리 정의된 속성을 사용한 사례</h1>
8   
10  <a href="http://github.com/eunsjun/" target="_blank" title="클릭하세요!">저자 카페 방문</a>
11 </body>
```

HTML5\_attr01.html

## ❖ 태그의 속성

### ● 사용자 정의 태그에 속성 추가 시 고려할 점

- 태그와 의미적으로 연관성이 있거나 부가적인 정보를 제공하는 속성인지 확인
- 태그 없이 단독으로 사용할 수 없기 때문에 태그와 관계가 있는지 확인
- 속성값은 대부분 한 가지 값으로 표현하는 것이 일반적
- 데이터 값의 단위가 여러 개일 경우 속성보다는 태그로 정의
- 속성값이 계속 변경되는 경우라면 속성보다는 태그로 정의
- 속성값은 웹 브라우저에 직접적으로 출력되지 않으므로 직접 출력되어야 하는 경우에는 태그로 정의

## ❖ 태그의 속성

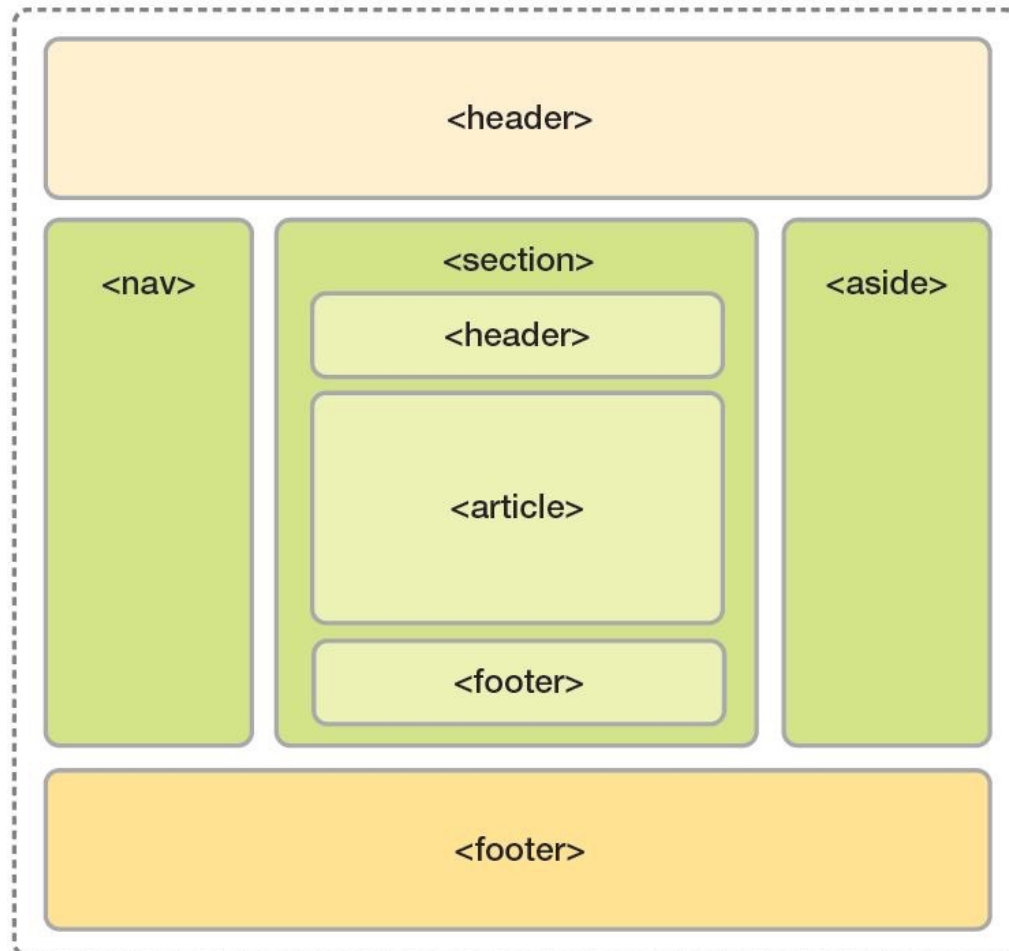
```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>사용자 정의 요소</title>
5 </head>
6 <body>
7   <company location="서울">글로벌아이티정보기술<br>
8     <department title="교무부 멤버 보기">
9       <a href="#mem1">교무부</a>
10    </department><br>
11    <department title="행정부 멤버 보기">
12      <a href="#mem2">행정부</a>
13    </department>
14  </company><p>-</p> <p>-</p>
15  <staff id="mem1">교무부 멤버 이름
16    <hr/>
17  <member position="원장">조재혁</member>
18  <member>전은석</member>
19  <member hidden="hidden">제니</member>
20  </staff>
21  <hr/><p>-</p> <p>-</p>
22  <staff id="mem2">행정부 멤버 이름
23    <hr/>
24  <member position="원장">조재혁</member>
25  <member hidden="hidden">제니</member>
26  <member>지수</member>
27  </staff>
28  <hr/>
29 </body>
30 </html>
```



## ❖ 웹 문서의 레이아웃

### ● 레이아웃

- 화면을 분할하거나 배열하여 구성하는 것
- HTML5 웹 표준에서는 각 영역을 구분하는 구조적 태그 요소를 정의하여 사용



- **<header>**  
HTML5 문서의 머리말 영역으로 중요한 정보를 표시  
(예 : 사이트의 제목, 로고 등)
- **<nav>**  
내비게이션(navigation) 영역으로  
웹 사이트 내에 분류된 다른 영역으로 이동할 때 사용
- **<section>**  
문서의 영역을 구성할 때 사용.  
<header>, <article> 태그 등을 포함할 수 있음
- **<article>**  
독립된 주요 콘텐츠 영역을 정의.  
하나의 <section> 태그 내에 여러 개의 <article> 태그를 구성할 수 있음
- **<aside>**  
주요 콘텐츠 이외에 남은 콘텐츠를 표시  
(예 : 사이드 바(sidebar) 등).
- **<footer>**  
사이트의 자세한 정보를 표시  
(예 : 저작권 정보, 관리자 정보, 회사 정보 등)

구조적 태그 요소

## ❖ 웹 문서의 레이아웃

HTML5\_layout01.html

```
1 <!-- HTML5 문서의 기본 레이아웃 -->
2 <!DOCTYPE html>
3 <html>
4 <head>
5   <title>HTML5 기본 구조 태그1</title>
6 </head>
7 <body>
8   <!-- 헤더 부분 -->
9   <header>
10    <center>
11      <h1>문서의 주제목</h1>
12    </center>
13    <nav>메뉴1</nav>
14    <nav>메뉴2</nav>
15  </header>
16  <hr/>
```

## ❖ 웹 문서의 레이아웃

HTML5\_layout01.html

```
17 <!-- 본문 부분 -->
18 <section>
19   <article>
20     <header>
21       <h2>부제목1</h2>
22     </header>
23     <p>이곳은 부 세션 영역입니다.</p>
24   </article>
25   <article>
26     <header>
27       <h2>부제목2</h2>
28     </header>
29     <p>이곳은 부 세션 영역입니다.</p>
30   </article>
31   <article>
32     <hgroup>
33       <h2>대그룹</h2>
34       <h3>중그룹</h3>
35       <h4>소그룹</h4>
36     </hgroup>
37   </article>
38 </section>
```

38

## ❖ 웹 문서의 레이아웃

HTML5\_layout01.html

```
39 <!-- 부가 정보 부분 -->
40 <aside>
41   <h2>부가적 제목</h2>
42   <p>이곳은 부가적 정보를 제공하는 공간입니다.</p>
43 </aside>
44 <hr/>
45 <!-- 사이트 정보 부분 -->
46 <footer>
47   <small>사이트 정보 표시</small>
48 </footer>
49 </body>
50 </html>
```

## ❖ 웹 문서의 레이아웃

- <hr> 요소

- 구분선(horizontal line)을 그을 때 사용

속 성	값	설 명
align	left, center, right	수평선의 정렬을 지정
noshade	noshade	수평선의 입체감을 제거
size	pixels	수평선의 두께를 지정
width	pixels, %	수평선의 상대적 넓이를 지정

<hr> 요소에 사용할 수 있는 속성

## ❖ 웹 문서의 레이아웃

HTML5\_layout02.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>HTML5 기본 구조 요소2</title>
5 </head>
6 <body>
7   <!-- 헤더 부분 -->
8   <header>
9     <center>
10      <h1>문서의 주제목</h1>
11    </center>
12    <hr align="center" width=50%></hr>
13  </header>
14  <!-- 본문 부분 -->
15  <section>
16    <article>
17      <h2>문서 내용1</h2>
18      <h3>문서 내용2</h3>
19      <h4>문서 내용3</h4>
20      <h5>문서 내용4</h5>
21      <h6>문서 내용5</h6>
22    </article>
23  </section>
24  <hr size="10" align="left" width=90% noshade="noshade"></hr>
25  <!-- 사이트 정보 부분 -->
26  <footer>
27    <small>사이트 정보 표시</small>
28  </footer>
29 </body>
30 </html>
```

## ❖ 특수문자 처리

특수문자/키보드	특수이름	아스키코드
<	&lt;	&#60;
>	&gt;	&#62;
&	&amp;	&#38;
©	&copy;	&#169;
®	&reg;	&#174;
#	&num;	&#35;
스페이스바(spacebar)	&nbsp;	&#160;

특수문자/키보드 입력값에 매칭되는 특수이름과 아스키코드 값

## ❖ 특수문자 처리

- 특수문자 출력하기

```
<body>
  <header>
    <center>
      <h1>&lt; HTML 특수문자 &gt;</h1>
    </center>
  </header>
  <section>
    <article>
      &num; HTML5 / 웹 표준 &num;
      <p></p>
      HTML5 &#38; CSS3 &#38; JavaScript &#38; JQuery
    </article>
  </section>
  <hr size="10" align="left" width=90% noshade="noshade"></hr>
  <footer>
    <small>
      &nbsp;&nbsp;&nbsp;&copy; 글로벌아이티정보기술 &nbsp;&nbsp;&nbsp;
      &nbsp;&nbsp;&nbsp;&reg; 2018
    </small>
  </footer>
</body>
```



## ❖ 특수문자 처리

### ● <pre> 태그

- 공백, 특수문자, 줄 바꿈 등도 사용자가 입력한 그대로 화면에 표시

```
<body>
  <header>
    <center>
      <h2>&lt; pre &gt; 태그</h2>
    </center>
  </header>
  <section>
    <article>
      <pre>
### HTML5 / 웹 표준 ###

      HTML5 & CSS3 & JavaScript & JQuery

      < 시간 : 13:50 ~ 17:40 >
      </pre>
    </article>
  </section>
  <hr size="10" align="left" width=90% noshade="noshade"></hr>
  <footer>
    <small>&copy; 글로벌아이티정보기술 &reg; 2018</small>
  </footer>
</body>
```

## ❖ 텍스트

태 그	설 명
<b>	볼트체로 표시
<em>	강조하여 표시
<i>	이탤릭체로 표시
<small>	작게 표시
<strong>	중요한 내용을 표시
<sub>	아래첨자로 표시
<sup>	윗첨자로 표시
<ins>	아래 밑줄을 표시
<del>	가운데를 선으로 표시
<mark>	하이라이트(Highlight) 표시
<code>	컴퓨터 코드 표시
<samp>	컴퓨터 프로그램 샘플 출력 표시
<kbd>	키보드 입력 표시
<var>	변수 표시

텍스트 관련 태그

## ❖ 텍스트

### ● 텍스트 태그 사용하기

```
<body>
  <header>
    <center>
      <h2>&lt; 텍스트 관련 태그 &gt;</h2>
    </center>
  </header>
  <section>
    <article>
      <p><b>텍스트 볼드 처리</b></p>
      <p><strong>텍스트 중요 표시</strong></p>
      <p><i>텍스트를 이탤릭체로</i></p>
      <p><em>텍스트 강조 표시</em></p>
      <p><ins>텍스트 밑줄 처리</ins></p>
      <p><del>텍스트 가운데 선 처리</del></p>
      <p>텍스트<sub>아래첨자</sub>텍스트<sup>윗첨자</sup></p>
      <p><mark>텍스트에 하이라이트 표시</mark></p>
    </article>
  </section>
</body>
```

## ❖ 하이퍼링크

### ● 웹 사이트 간 이동

- href 속성: 연결하고자 하는 웹 사이트의 URL 지정
- target 속성: 링크를 클릭했을 때 웹 사이트가 열릴 곳 지정

속성값	설 명
_blank	새로운 웹 브라우저 창에 연다.
_self	현재 웹 브라우저 창에 연다(기본).
_parent	부모 웹 브라우저 창에 연다.
_top	웹 브라우저 전체 영역에 연다.

target 속성값의 종류

## ❖ 하이퍼링크

- 타깃 속성값에 따른 결과 확인하기

```
<body>
  <header>
    <center>
      <h2>&lt; 하이퍼링크 관련 태그 &gt;</h2>
    </center>
  </header>
  <section>
    <article>
      <a href="http://www.w3.org" target="_blank">W3C 사이트(blank window)</a><p/>
      <a href="http://www.w3.org" target="_self">W3C 사이트(self window)</a><p/>
      <a href="http://www.w3.org" target="_parent">W3C 사이트(parent window)</a><p/>
      <a href="http://www.w3.org" target="_top">W3C 사이트(top window)</a><p/>
    </article>
  </section>
</body>
```

HTML5\_hrefTarget01.html

## ❖ 하이퍼링크

### ● 베이스 태그 사용하기

```
<head>
  <title>base Tag</title>
  <base href="http://www.w3.org/" />
</head>
<body>
  <header>
    <center>
      <h2>&lt; 베이스 태그 사용 &gt;</h2>
    </center>
  </header>
  <section>
    <article>
      <a href="standards/" target="_blank">W3C STANDARDS</a>
      <p/>
      <a href="Consortium/mission.html" target="_blank">W3C MISSION</a>
      <p/>
      <a href="Consortium/facts.html" target="_blank">FACTS ABOUT W3C</a>
      <p/>
      <a href="Consortium/presskit.html" target="_blank">PRESS AND ANALYSTS</a>
      <p/>
    </article>
  </section>
</body>
```

## ❖ 하이퍼링크

### ● 문서 내 특정 위치로 이동

```
<body>
  <header>
    <center>
      <h2>&lt; 책갈피 기능 &gt;</h2>
    </center>
  </header>
  <section>
    <article>
      <a href="#user">[이름]</a> &nbsp;
      <a href="#addr">[주소]</a> &nbsp;
      <a href="#tel">[전화번호]</a> &nbsp;
      <a href="#foot">[참고]</a> &nbsp;
    <p/>
  </article>
</section>

<p>정보 영역</p>
...생략
<p>정보 영역</p>
```

```
    <p><a name="user">홍길동</a></p>
    <a href="#top">[TOP]</a>
  <p>정보 영역</p>
  ...생략
  <p>정보 영역</p>

    <p><a id="addr">서울 구로구 신림동</a></p>
    <a href="#top">[TOP]</a>

  <p>정보 영역</p>
  ...생략
  <p>정보 영역</p>

    <p id="tel">02-1234-5678</p>
    <a href="#top">[TOP]</a>

  <p>정보 영역</p>
  ...생략
  <p>정보 영역</p>
</body>
```

## ❖ 하이퍼링크

### ● 이메일 링크

- <a> 태그의 href 속성값에 이메일 주소 작성

**<a href="mailto:이메일 주소">내용</a>**

속성값	설 명
mailto:euns_jun@naver.com	받는 사람의 이메일 주소
cc=euns_jun@naver.com	참조할 사람의 이메일 주소
bcc=euns_jun@naver.com	숨은 참조할 사람의 이메일 주소
subject=subject text	이메일 제목 (제목에 공백이 들어갈 경우 '%20' 문자 사용)
body=body text	이메일 본문
?	첫 번째 구분자
&	연속 구분자

href 속성에 사용할 수 있는 속성 값



## ❖ 하이퍼링크

### ● 이메일 링크 걸기

```
<body>
  <header>
    <center>
      <h2>&lt; 이메일 보내기 &gt;</h2>
    </center>
  </header>
  <a href="mailto:euns_jun@naver.com">받는 사람</a><br />
  <a href="mailto:euns_jun@naver.com?subject=질문 있어요">받는 사람, 제목</a><br>
  <a href="mailto:euns_jun@naver.com?
cc=abcde@gmail.com&bcc=fghij@gmail.com">받는 사람, 참조, 숨은 참조</a><br>
  <a href="mailto:euns_jun@naver.com?
cc=abcde@gmail.com&bcc=fghij@gmail.com&subject=질문 있어요&body=웹 프로그래
밍">받는 사람, 참조, 숨은 참조, 제목, 본문</a><br>
  <a href="mailto:euns_jun@naver.com?body=Line1-text%0D%0ALine2-text">받는 사
람, 본문 문단</a>
</body>
```

## ❖ 목록

### ● 무순서 목록

- 순서가 없는 목록
- <ul> 태그 사용, 각 항목은 <li> 태그로 입력
- 목록의 각 항목 앞에는 불릿(bullet)이 붙음

### ● 순서 목록

- 순서가 있는 목록
- <ol> 태그 사용, 각 항목은 <li> 태그로 입력
- 목록의 각 항목에는 기본값으로 type="1"이 적용

### ● 정의형 목록

- 각 항목을 정의하기 위한 목록
- <dl> 태그 사용
- 정의한 각 항목은 <dt>태그를, 각 항목에 대한 설명은 <dd> 태그를 사용

## ❖ 목록

구 분	무순서 목록	순서 목록	정의형 목록
결 과	<ul style="list-style-type: none"> <li>Apples</li> <li>Bananas</li> <li>Lemons</li> <li>Oranges</li> </ul>	<ol style="list-style-type: none"> <li>Apples</li> <li>Bananas</li> <li>Lemons</li> <li>Oranges</li> </ol>	Coffee - hot drink Milk - cold drink
코 드	<pre> &lt;ul&gt;   &lt;li&gt;Apples&lt;/li&gt;   &lt;li&gt;Bananas&lt;/li&gt;   &lt;li&gt;Lemons&lt;/li&gt;   &lt;li&gt;Oranges&lt;/li&gt; &lt;/ul&gt; </pre>	<pre> &lt;ol&gt;   &lt;li&gt;Apples&lt;/li&gt;   &lt;li&gt;Bananas&lt;/li&gt;   &lt;li&gt;Lemons&lt;/li&gt;   &lt;li&gt;Oranges&lt;/li&gt; &lt;/ol&gt; </pre>	<pre> &lt;dl&gt;   &lt;dt&gt;Coffee&lt;/dt&gt;   &lt;dd&gt;- hot drink&lt;/dd&gt;   &lt;dt&gt;Milk&lt;/dt&gt;   &lt;dd&gt;- cold drink&lt;/dd&gt; &lt;/dl&gt; </pre>

### 목록 태그의 종류

## ❖ 목록

무순서 목록		순서 목록	
속성	블릿 기호	속성	순서 기호
style="list-style-type:disc" style="list-style-type:circle" style="list-style-type:square" style="list-style-type:none"	○ ● ■ 기호 없음	type="1" type="A" type="a" type="I" type-"i"	1, 2, 3, . . . A, B, C, . . . a, b, c, . . . I, II, III, . . . i, ii, iii, . . .

무순서 목록과 순서 목록에 쓰이는 속성

## ❖ 목록

- 목록 출력하기

```
<body>
  <!-- 무순서 목록 -->
  <section>
    <article>
      <h2>무순서 목록</h2>
      <ul>
        <li>Coffee</li>
        <li>Tea
          <ul>
            <li>Black tea</li>
            <li>Green tea</li>
          </ul>
        </li>
        <li>Milk</li>
      </ul>
    </article>
  </section>
```

```
<!-- 순서 목록 -->
<section>
  <article>
    <h2>순서 목록</h2>
    <ol>
      <li>Apples</li>
      <li>Bananas</li>
      <li>Lemons</li>
      <li>Oranges</li>
    </ol>
  </article>
</section>
<!-- 정의형 목록 -->
<section>
  <article>
    <h2>정의형 목록</h2>
    <dl>
      <dt>첫 번째 아이템</dt>
      <dd>- HTML5</dd>
      <dt>두 번째 아이템</dt>
      <dd>- CSS3</dd>
      <dt>세 번째 아이템</dt>
      <dd>- JavaScript</dd>
    </dl>
  </article>
</section>
</body>
```

## ❖ 표

## ● &lt;table&gt; 태그

- <tr> 태그 : 행 생성
- <td> 태그 : 열 생성
- <th> 태그 : 표의 머리를 정의,  
셀 제목 글자를 강조하는 역할

```
<body>
  <section>
    <article>
      <table border="1" style="width:50%">
        <tr>
          <th>이름</th>
          <th>점수</th>
        </tr>
        <tr>
          <td>김정화</td>
          <td>90</td>
        </tr>
        <tr>
          <td>심청이</td>
          <td>80</td>
        </tr>
        <tr>
          <td>설 현</td>
          <td>70</td>
        </tr>
      </table>
    </article>
  </section>
</body>
```

HTML5\_table01.html

## ❖ 표

- 표 제목 삽입
  - <caption> 태그 사용
- 셀 병합
  - rowspan : 셀을 세로로 병합,  
속성값으로 병합하고 싶은 행의 수만큼 지정
  - colspan : 셀을 가로로 병합,  
속성값으로 병합하고 싶은 열의 수만큼 지정

## ❖ 표

- 셀 병합 하기

```
<body>
  <section>
    <article>
      <table border="1" style="width:50%">
        <caption>행 통합 테이블</caption>
        <tr>
          <th>1행</th>
          <td>1행 1열</td>
        </tr>
        <tr>
          <th rowspan="2">2행</th>
          <td>2행 2열</td>
        </tr>
        <tr>
          <td>3행 2열</td>
        </tr>
      </table>
    </article>
  </section>
<p></p>
```

```
<section>
  <article>
    <table border="1" style="width:50%">
      <caption>열 통합 테이블</caption>
      <tr>
        <th>1열</th>
        <th colspan="2">2열</th>
      </tr>
      <tr>
        <td>2행 1열</td>
        <td>2행 2열</td>
        <td>2행 3열</td>
      </tr>
    </table>
  </article>
</section>
</body>
```



## ❖ 표

- 셀 병합 하기

```
<body>
  <section>
    <article>
      <table border="1" style="width:50%">
        <tr>
          <td>가</td>
          <td>나</td>
          <td>다</td>
          <td rowspan="4">라</td>
        </tr>
        <tr>
          <td>마</td>
          <td colspan="2" rowspan="3">바</td>
        </tr>
        <tr>
          <td>사</td>
        </tr>
        <tr>
          <td>아</td>
        </tr>
        <tr>
          <td colspan="2">자</td>
          <td colspan="2">차</td>
        </tr>
        <tr>
          <td colspan="4">카</td>
        </tr>
      </table>
    </article>
  </section>
</body>
```

## ❖ 표

## ● 시맨틱 태그

- <thead> : 표 머리말(head) 부분의 그룹 태그
- <tbody> : 표 본문(body) 부분의 그룹 태그
- <tfoot> : 표 꼬리말(Footer) 부분의 그룹 태그

```
<body>
  <section>
    <article>
      <table border="1" summary="시맨틱 테이블 관련 요소 목록">
        <caption>Semantic Table Tag</caption>
        <thead>
          <tr>
            <th>태그 명칭</th>
            <th>설명</th>
            <th>사용 여부</th>
          </tr>
        </thead>
        <tfoot>
          <tr>
            <td colspan="3">참고 사이트 : http://www.w3.org/ </td>
          </tr>
        </tfoot>
```

## ❖ 표

## ● 시맨틱 태그

```
<tbody>
  <tr>
    <td>thead</td>
    <td>표 머리말(head) 부분의 그룹 태그</td>
    <td>가능</td>
  </tr>
  <tr>
    <td>tfoot</td>
    <td>표 꼬리말/footer) 부분의 그룹 태그</td>
    <td>가능</td>
  </tr>
  <tr>
    <td>tbody</td>
    <td>표 본문(body) 부분의 그룹 태그</td>
    <td>가능</td>
  </tr>
</tbody>
</table>
</article>
</section>
</body>
```

## ❖ 멀티미디어 태그

### ● 임베디드 콘텐츠 태그

- 멀티미디어 데이터를 웹 문서에 포함할 때 사용하는 태그

태 그	설 명
<img>	이미지 파일 삽입
<canvas>	그래픽 그리기
<audio>	오디오 파일 삽입
<video>	동영상 파일 삽입
<track>	미디어 요소 트랙 삽입
<source>	미디어 소스 삽입
<embed>	개체 삽입

임베디드 콘텐츠 태그

## ❖ 이미지

파일 형식	설 명	확장자
GIF	<ul style="list-style-type: none"><li>• 256개의 색상만 지원하여다른 파일에 비해 용량이 작다.</li><li>따라서 아이콘이나 단색 계역의 배경을 만드는 데 적합</li><li>• 손실이 발생하지 않는 무손실 압축 방식을 사용한다.</li><li>• 투명 이미지와 움직이는 애니메이션 이미지를 만들 수 있으며, 간단한 배너 광고나 불릿 제작이 가능하다.</li></ul>	.gif
JPG	<ul style="list-style-type: none"><li>• 작은 용량으로 사진과 같이 섬세한 색상과 명암을 표현하기 때문에 웹에서 가장 많이 사용되는 형식이다.</li><li>• 1,600만 개의 색상을 사용하며 압축률이 높다.</li><li>손실 압축 방식이라 압축 과정에서 약간의 손실이 발생한다.</li><li>• GIF와 같은 투명 이미지나 움직이는 이미지는 만들 수 없다.</li></ul>	.jpg
PNG	<ul style="list-style-type: none"><li>• 무손실 압축 방식을 사용하기 때문에 높은 품질의 이미지를 처리할 수 있고 GIF와 같은 투명 이미지도 사용할 수 있어 최근 사용빈도가 높아지고 있다.</li><li>• 이미지가 복잡할 경우 파일의 크기가 커지는 단점이 있다.</li></ul>	.png

## 이미지 파일의 특징

## ❖ 이미지

### ● <img> 태그

- 웹 문서에 이미지 삽입 시 사용

```
  
  

```

### ● <img> 태그 속성

- **src** : 이미지 파일이 저장된 경로 지정
- **alt** : 이미지를 웹 브라우저에서 표시하지 못했을 경우 표시되는 대체 텍스트 지정
- **width/height** : 이미지의 가로, 세로 길이를 픽셀 단위로 지정  
(% 단위를 사용하면 웹 브라우저의 크기에 따라 이미지 크기가 조절되도록 지정할 수 있음)
- **style** : 이미지의 스타일(크기, 위치 등)을 픽셀 단위로 지정
- **border** : 이미지 경계선의 두께를 픽셀 단위로 지정

## ❖ 이미지

- 이미지 삽입하기

```
<body>
  <h3>이미지 기본 표현</h3>
  
  <h3>이미지에 설명 추가 & 오른쪽 정렬</h3>
  
  <h3>이미지의 크기를 픽셀 단위로 조정</h3>
  
  <h3>이미지의 크기를 % 단위로 조정</h3>
  
</body>
```

HTML5\_img01.html

## ❖ 이미지

- 이미지 링크 걸기

```
<body>
  <h3>이미지를 클릭하세요.</h3>
  <a href="http://www.w3.org" target="_blank">
    
  </a>
  <a href="http://cafe.naver.com/go2web" target="_blank">
    
  </a>
  <a href="http://en.wikipedia.org/wiki/Isaac_Newton" target="_blank">
    <img src="newton.gif" alt="newton site" title="뉴턴 홈페이지로 이동" border="2"
```

HTML5\_img02.html



## ❖ 이미지

- 이미지에 제목 붙이기

```
<body>
  <h3>이미지에 제목 추가하기</h3>
  <figure>
    
    <figcaption>[그림1] HTML5</figcaption>
  </figure>
  <figure>
    <figcaption>[그림2] Google</figcaption>
    
  </figure>
</body>
```

HTML5\_img03.html

## ❖ 오디오

파일 형식	설 명	확장자
MP3	MPEG-1 Layer 3'를 줄여서 부르는 말로 1988년 독일의 브라운호퍼연구소에서 처음 개발하였다. 오디오 CD와 맞먹는 높은 음질을 제공하지만 데이터 크기는 1/10에 불과해 현재가장 많이 사용되고 있다.	.mp3
OGG	mp3의 대안으로 새롭게 부상하고 있는 오디오 파일 형식으로 Ogg Vorbis라는 이름에서 온 말이다. 디지털 오디오 포맷의 특허권에 반대하여 보다 좋은 음질을 가진 음악 파일을 표방한다. 오픈소스를 지향하고 있어 수정과 배포가 자유롭다.	.ogg .oga
WAV	개인용 PC에서 오디오를 재생하기 위한 IBM과 Microsoft의 표준 오디오 파일 형식이다.	.wav

## 오디오 파일의 종류

## ❖ 오디오

### ● <audio> 태그

- 웹 페이지에 오디오 파일을 삽입하는 태그

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

### ● <audio> 태그 속성

- **src** : 재생할 오디오 파일이 저장된 경로 표시
- **controls** : 재생 제어기 표시
- **loop** : 반복 재생
- **autoplay** : 웹 문서가 열림과 동시에 자동으로 재생
- **muted** : 음소거 상태로 시작
- **preload** : 오디오 파일 사용 여부와 상관없이 미리 다운로드함

## ❖ 오디오

### ● <source> 태그

- 여러 형식의 오디오 파일을 동시에 제공할 때 사용
- 재생에 적합한 파일이 없다면 마지막 문장 출력

```
<audio controls>  
  <source src="audio_file.mp3" type="audio/mpeg">  
  <source src="audio_file.ogg" type="audio/ogg">  
  <source src="audio_file.wav" type="audio/wav">  
  오디오 파일을 재생할 수 없습니다.  
</audio>
```

웹 브라우저	MP3	OGG	WAV
인터넷 익스플로러	가능	불가능	불가능
크롬	가능	가능	가능
파이어폭스	가능	가능	가능
사파리	가능	불가능	가능
오페라	가능	가능	가능

임베디드 콘텐츠 태그

## ❖ 오디오

- 오디오 삽입하기

```
<body>
  <h3>오디오 자동 재생</h3>
  <audio src="eagle.mp3" controls loop autoplay></audio>
  <h3>오디오 수동 재생</h3>
  <audio src="bear.mp3" controls></audio>
  <h3>오디오 다중 재생</h3>
  <audio controls>
    <source src="chicken.mp3" type="audio/mpeg">
    <source src="chicken.ogg" type="audio/ogg">
    <source src="chicken.wav" type="audio/wav">
    오디오 파일을 재생할 수 없습니다.
  </audio>
</body>
```

HTML5\_audio01.html

## ❖ 비디오

### ● <video> 태그 사용

- 비디오 파일을 웹 문서에 삽입할 때 사용
- <source> 태그로 여러 가지의 비디오 파일을 동시에 제공

### ● <video> 태그 속성

- **src** : 재생할 비디오 파일이 저장된 경로 표시
- **controls** : 재생 제어기 표시
- **width, height** : 비디오의 가로/세로 크기 설정
- **loop** : 반복 재생
- **autoplay** : 웹 문서가 열림과 동시에 자동 재생
- **muted** : 음소거 상태로 시작
- **preload** : 비디오 파일을 사용 여부와 상관없이 미리 다운로드함
- **poster** : 비디오 파일이 재생되기 전이나  
다운로드되지 않고 있는 경우 표시될 이미지의 URL을 표시

## ❖ 비디오

### ● 비디오 삽입하기

```
<body>
  <h3>비디오 자동 재생</h3>
  <video src="bear.mp4" controls loop autoplay width="300" height="250"></video>
  <h3>비디오 수동 재생</h3>
  <video src="bear.mp4" controls width="300" height="250"></video>
  <h3>비디오 다중 재생-비디오 파일이 현재 폴더에 있는 경우</h3>
  <video controls autoplay width="300" height="250">
    <source src="small.mp4" type="video/mpeg">
    <source src="small.ogv" type="video/ogg">
    <source src="small.webm" type="video/webm">
    비디오 파일을 재생할 수 없습니다.
  </video>
  <h3>비디오 다중 재생-비디오 파일이 웹 사이트에 있는 경우</h3>
  <video controls autoplay width="300" height="250" poster="wait.jpg">
    <source src="http://media.w3.org/2010/05/sintel/trailer.mp4"
      type='video/mp4; codecs="avc1, mp4a"'>
    <source src="http://media.w3.org/2010/05/sintel/trailer.ogv"
      type='video/ogg; codecs="theora, vorbis"'>
    비디오 파일을 재생할 수 없습니다.
  </video>
</body>
```

## ❖ 개체 삽입

### ● 개체 삽입 태그

- 이미지, 오디오, 비디오, 태그로 지원하지 않는 파일을 웹 문서에 포함하기 위해 사용
- <object> 태그와 <embed> 태그가 있음

### ● <embed> 태그 사용 방법

```
<embed src="object.swf">  
<embed src="/web/myvideo.mov" width="340" height="140" type="video/quicktime">
```

### ● <object> 태그 사용 방법

```
<object data="image.bmp"></object>  
<object data="object.avi" width="340" height="140"></object>
```



## ❖ 개체 삽입

### ● 개체 삽입하기

```
<body>
  <h1>embed 태그 사용</h1>
  <h3>이미지 파일 삽입</h3>
  <embed src="html.bmp" width="150" height="170">
  <h3>플래시 파일 삽입</h3>
  <embed src="object.swf">
  <h3>웹 문서 삽입</h3>
  <embed src="http://www.hanbit.co.kr" width="500" height="170">

  <h1>object 태그 사용</h1>
  <h3>이미지 파일 삽입</h3>
  <object data="html.bmp"></object>
  <h3>플래시 파일 삽입</h3>
  <object type="application/x-shockwave-flash" data="object.swf"
    width="250" height="200"></object>
  <h3>PDF 파일 삽입</h3>
  <object type="application/pdf" data="pdflogo.pdf"></object>
</body>
```

HTML5\_embed01.html

## ❖ 입력 양식 요소

- 웹 양식

- 웹 문서에 어떤 데이터를 입력하고 그 결과값을 가져오기 위해 사용

- 폼 태그

- 웹 양식을 만드는 데 사용하는 태그

```
<form name="입력 폼 이름" action="웹 프로그램 페이지" method="전달 방식" >  
  <input type="폼 모양과 기능" name="입력 폼 변수" value="전달 값" >  
</form>
```

- **action** : 사용자가 입력한 데이터를 받아 처리하기 위한  
웹 프로그램(ASP, PHP, JSP... 등)의 페이지 지정
- **method** : 웹 서버와 클라이언트 간의 통신 방법 지정(GET 방식, POST 방식)
- **type** : 폼의 모양과 기능 결정
- **name** : 폼의 이름 결정

## ❖ method 속성에서 사용하는 방식

### ● GET 방식

- URL 뒤에 파라미터를 붙여서 데이터를 전달하는 방식
- 사용자가 보내는 데이터는 이름과 값이 결합된 문자열 형태로 전달, 각 이름과 값의 쌍은 ‘&’ 기호로 구분
- 서버로 보낼 수 있는 최대 글자수는 2,048자
- URL을 보면 어떤 데이터를 전송하고자 하는지 알 수 있기 때문에 보안에 취약

```
<form name="INPUT1" action="page.jsp" method="get">  
  <input type="text" name="user">  
  <input type="text" name="dept">  
  <input type="submit" value="전송">  
  <input type="reset" value="다시작성">  
</form>
```

http://localhost/book/page.jsp?user=Hong&dept=Computer

## ❖ method 속성에서 사용하는 방식

### ● POST 방식

- HTTP Request 헤더에 파라미터를 붙여서 데이터를 전송하는 방식
- 서버로 보낼 수 있는 글자수 제한 없음
- GET 방식과 비교하여 보안상 우위에 있음

```
<form name="INPUT1" action="page.jsp" method="post">  
  <input type="text" name="user">  
  <input type="text" name="dept">  
  <input type="submit" value="전송">  
  <input type="reset" value="다시작성">  
</form>
```

http://localhost/book/page.jsp

## ❖ method 속성에서 사용하는 방식

- GET 방식으로 데이터 전송하기

```
<!DOCTYPE html>
<html>
<head>
  <title>Get 방식</title>
</head>
<body>
  <h2>GET 방식으로 데이터 전송</h2>
  <form action="01_getdata.jsp" method="get">
    <p>이름 : <input type="text" name="name"></p>
    <p>전공 : <input type="text" name="major"></p>
    <p></p>
    <input type="submit" value="전송">
    <input type="reset" value="다시작성">
  </form>
</body>
</html>
```

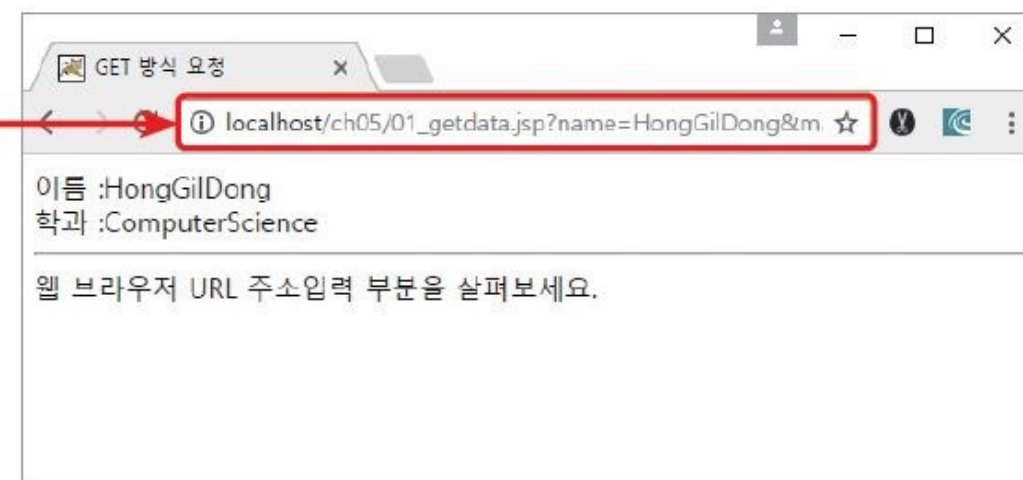
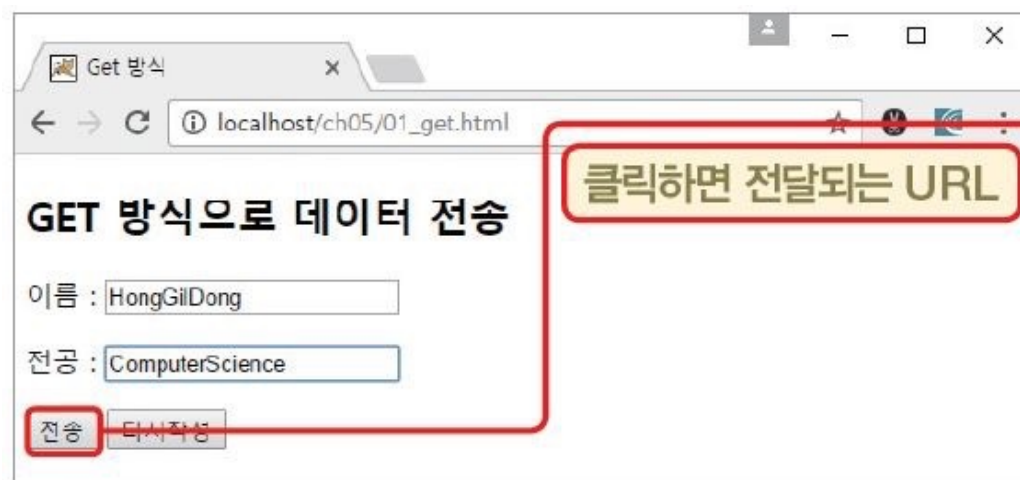
HTML5\_get01.html

## ❖ method 속성에서 사용하는 방식

HTML5\_getdata01.jsp

```
<%@page language="java" contentType="text/html; charset=EUC-KR" pageEncoding="EUC-KR"%>

<html>
<head>
    <title>GET 방식 요청</title>
</head>
<body>
<!-- JSP 문법 작성 -->
<%
String strName=request.getParameter("name");
String strMajor=request.getParameter("major");
out.println("이름 :" + strName + "<br/>");
out.println("학과 :" + strMajor + "<hr/>");
%>
웹 브라우저 URL 주소 입력 부분을 살펴보세요.
</body>
```



## ❖ method 속성에서 사용하는 방식

- POST 방식으로 데이터 전송하기

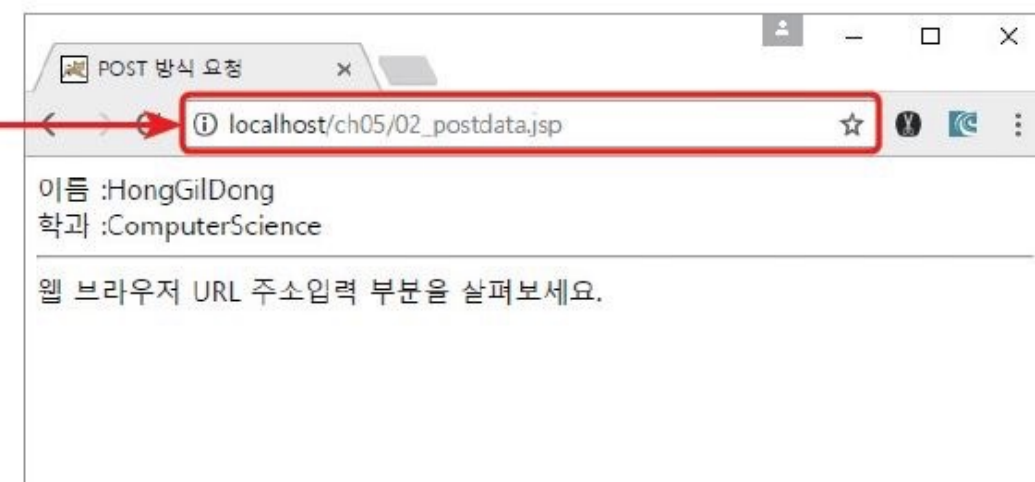
```
<!DOCTYPE html>
<html>
<head>
  <title>POST 방식</title>
</head>
<body>
  <h2>POST 방식으로 데이터 전송</h2>
  <form action="02_postdata.jsp" method="POST">
    <p>이름 : <input type="text" name="name"></p>
    <p>전공 : <input type="text" name="major"></p>
    <p></p>
    <input type="submit" value="전송">
    <input type="reset" value="다시작성">
  </form>
</body>
</html>
```

## ❖ method 속성에서 사용하는 방식

HTML5\_postdata01.jsp

```
<%@page language="java" contentType="text/html; charset=EUC-KR" pageEncoding="EUC-KR"%>

<html>
<head>
    <title>POST 방식 요청</title>
</head>
<body>
<!-- JSP 문법 작성 -->
<%
String strName=request.getParameter("name");
String strMajor=request.getParameter("major");
out.println("이름 :" + strName + "<br/>");
out.println("학과 :" + strMajor + "<hr/>");
%>
웹 브라우저 URL 주소 입력 부분을 살펴보세요.
</body>
```





## ❖ 텍스트/비밀번호 입력 양식

- 텍스트 입력 양식
  - 기본적인 텍스트를 입력할 때 사용

```
<body>
  <h2>문자 입력 양식</h2>
  <form>
    <p>이름 : <input type="text" name="name" size="10"></p>
    <p>학번 : <input type="text" name="name" size="10" value=""></p>
    <p>학과 : <input type="text" name="school" value="컴퓨터과학과"></p>
    <p></p>
    <input type="submit" value="전송">
    <input type="reset" value="다시작성">
  </form>
</body>
```

HTML5\_text01.html

## ❖ 텍스트/비밀번호 입력 양식

- 비밀번호 입력 양식

- 사용자가 입력한 문자를 보이지 않게 ‘.’으로 처리

```
<body>
  <h2>회원 인증 입력 양식</h2>
  <form>
    <p>ID : <input type="text" name="ID" size="15" required></p>
    <p>PW : <input type="password" name="psw" size="15" placeholder="비밀번호"
required></p>
    <p></p>
    <input type="submit" value="전송">
    <input type="reset" value="다시작성">
  </form>
```

HTML5\_password01.html

## ❖ 텍스트 공간 입력/필드셋 양식

- 텍스트 공간 입력 양식
  - 텍스트를 여러 줄에 걸쳐 자유롭게 입력 가능

```
<body>
  <h2>텍스트 공간 입력 양식</h2>
  <form>
    <textarea rows="5" cols="50">텍스트를 작성하는 공간입니다.</textarea>
    <p></p>
    <input type="submit" value="전송">
    <input type="reset" value="다시작성">
  </form>
</body>
```

HTML5\_textarea01.html

## ❖ 텍스트 공간 입력/필드셋 양식

- <fieldset> 태그

- 입력 폼이 여러 개 있을 때, 경계선을 그려서 하나의 그룹으로 만들 때 사용

```
<body>
  <form>
    <fieldset>
      <legend>개인 정보 입력</legend>
      <p>이름 : <input type="text" name="name"></p>
      <p>학교 : <input type="text" name="school"></p>
      <input type="submit" value="제출">
      <input type="reset" value="다시작성">
    </fieldset>
  </form>
</body>
```

HTML5\_filedset01.html

## ❖ 라디오/체크박스/버튼 양식

- 라디오 입력 양식
  - 여러 항목 중 하나만 선택할 때 사용

```
<body>
  <form>
    <h3>당신의 성별은 무엇입니까?</h3>
    <input type="radio" name="sex" value="male" checked>남자
    <input type="radio" name="sex" value="female">여자
    <p></p>
    <h3>당신은 몇 학년입니까?</h3>
    <input type="radio" name="year" value="1" checked>1학년
    <input type="radio" name="year" value="2">2학년
    <input type="radio" name="year" value="3">3학년
    <input type="radio" name="year" value="4">4학년
    <p></p>
    <input type="submit" value="제출">
    <input type="reset" value="다시작성">
  </form>
</body>
```

## ❖ 라디오/체크박스/버튼 양식

- 체크박스 양식
  - 동시에 여러 항목을 선택할 때 사용

```
<body>
  <form>
    <h3>현재 관심을 가지고 있는 학습 주제는 무엇입니까?</h3>
    <input type="checkbox" name="subject" value="HTML5" checked>HTML5 <br>
    <input type="checkbox" name="subject" value="CSS3">CSS3 <br>
    <input type="checkbox" name="subject" value="Javascript">Javascript <br>
    <input type="checkbox" name="subject" value="Jquery">Jquery
    <p></p>
    <input type="submit" value="제출">
    <input type="reset" value="다시작성">
  </form>
</body>
```

HTML5\_checkbox01.html

## ❖ 라디오/체크박스/버튼 양식

- 버튼 양식

- <button> 태그 혹은 <input> 태그를 사용하여 정의

```
<body>
  <form>
    <h3>Button 태그 사용</h3>
    <button type="button" onclick="alert('클릭-1 사용')">클릭-1</button>
    <h3>Input 태그 사용</h3>
    <input type="button" onclick="alert('클릭-2 사용')" value="클릭-2">
    <h3>Image 버튼 사용</h3>
    <button type="button" onclick="alert('클릭-3 사용')"></button>
  </form>
</body>
```

HTML5\_button01.html

## ❖ 선택 목록 양식

- 선택 목록
  - 펼침목록에서 한 가지만 선택할 수 있도록 지원하는 양식

```
<body>
  <h3>관심 있는 학습 주제 한 가지를 선택하세요.</h3>
  <form>
    <select name="subjects">
      <option value="1" selected>HTML5</option>
      <option value="2">CSS3</option>
      <option value="3">Javascript</option>
      <option value="4">Jquery</option>
    </select>
    <p></p>
    <input type="submit" value="제출">
    <input type="reset" value="다시작성">
  </form>
</body>
```

HTML5\_button01.html