

HTML5

목 차

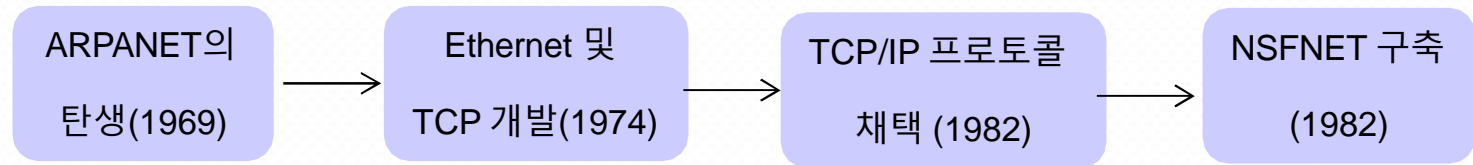
<i>Chapter 01</i>	HTML5 개요
<i>Chapter 02</i>	HTML5 문서기본
<i>Chapter 03</i>	링크와 멀티미디어 ⌚
<i>Chapter 04</i>	웹폼
<i>Chapter 05</i>	CSS 스타일 시트
<i>Chapter 06</i>	자바스크립트 기초
<i>Chapter 07</i>	DOM과 자바스크립트
<i>Chapter 08</i>	캔버스

Chapter 01

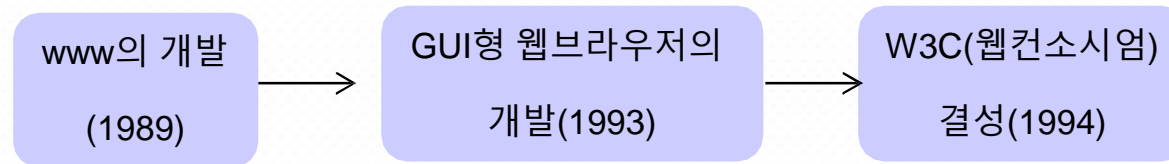
HTML5 개요

01 인터넷 역사

■ 인터넷의 발전:



■ 웹의 발전 :



■ 웹 브라우저란

- 컴퓨터 화면에서 웹 정보를 볼 수 있도록 만든 프로그램
- 상업용 브라우저
 - 웹 브라우저의 편리한 사용성으로 인터넷 사용자가 폭발적으로 증가 -> 웹 을 대중화 하는데 커다란 기여

■ 마크업 언어(Markup Language)

- 문서의 속성을 설정하기 위한 마크업을 태그의 형태로 표시
- 대표적인 마크업 언어 : HTML, XML, SHML

■ HTTP

- HTTP는 웹 상에서 파일(텍스트, 그래픽 이미지, 사운드, 비디오 그리고 기타 멀티미디어 파일)을 주고 받는 필요한 프로토콜(통신을 원하는 두 개체간에 무엇을, 어떻게, 언제 통신할 것인가를 서로 약속한 통신규약) 이다.

■ 웹 브라우저

- 웹 서버에 있는 하이퍼텍스트 정보를 사용자 화면에 보여주는 역할
- 하이퍼링크의 내비게이션을 도와주는 기능
- 텍스트와 이미지 뿐만 아니라 멀티미디어 처리 기능

■ 웹과 브라우저의 출현

- 1989년 유럽입자물리연구소의 팀 버너스-리
 - 웹(www, World Wide Web)을 처음 제안
 - 1994년 웹 컨소시엄(W3C, www Consortium)결성
- 인터넷 상 분산형 멀티미디어 하이퍼텍스트 시스템 구축
 - 서로 다른 기종의 컴퓨터에서 서로 다른 유형의 파일이나 데이터를 호환성 있게 교환하는 것이 목적

■ 웹 브라우저가 제공하는 기본적인 기능

- 웹 페이지 열기
- 최근 방문한 URL 목록 제공
- 자주 방문하는 즐겨찾기 주소록 관리
- 웹 페이지의 저장 및 인쇄
- 탭을 이용하여 여러 화면 보기
- 소스파일(HTML) 보기
- 보안 및 필터에 관련된 각종 도구

■ 인터넷 익스플로러(Internet Explorer)

- 1998년 넷스케이프 몰락이후 가장 널리 사용
- 경쟁이 거세진 이후 2007년 버전 7.0출시,
- 웹 표준 호환성이 떨어지는 편이며, 국내의 경우 맥티브 X사용 문제점

■ 모질라 파이어폭스(Mozilla Firefox)

- 비영리 단체인 모질라 재단에서 2004년 1.0버전 발표
- 모질라 재단 내에서 자유 소프트웨어 웹 브라우저 개발
- 다양한 운영체제에서 실행이 가능하며 표준 준수율이 매우 높음

■ 구글 크롬(Google Chrome)

- 2008년 출시, 속도와 안정성 및 보안성을 갖는 것을 목표로 개발
- 간단하면서도 효율적인 사용자 인터페이스 제공
- 웹 표준을 준수하는 사이트에 대한 호환성이 우수
- 모바일 안드로이드 환경과의 호환성으로 점유율 대폭 상승.

01 HTML5 소개 (I)

- HTML5란?
 - 웹 문서를 만들기 위한 기본 언어인 HTML의 최신 규격.
 - Active X와 같은 부가기능을 사용하지 않고 웹을 제작 할 수 있는 HTML의 문서 버전.
- HTML 역사

1986	SGML				
1991		HTML			
1995		HTML 2.0			
1996				CSS1	Javascript
1997		HTML 4.0			
1998	XML→			CSS2	
2000			XHTML 1.0		
2014		HTML5		CSS3	

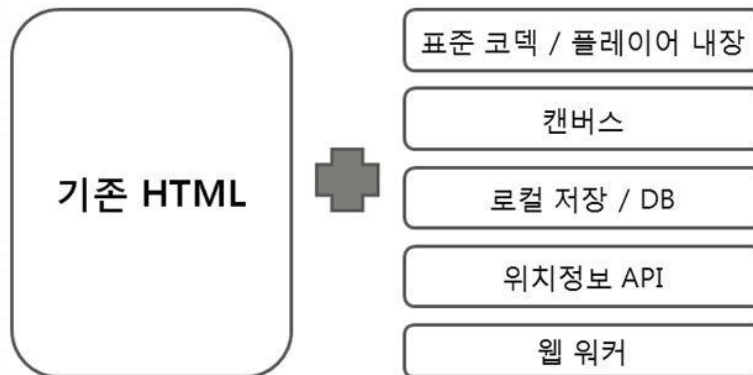
- HTML5 주요기능

CLASS	FUNCTION	COMMNET
Semantics	Web Form	보다 다양한 웹 입력 양식 제공 (예 달력, 범위, 이메일, 색상 등)
Multimedia	Video	동영상 재생을 위한 API제공
	Audio	음성 재생을 위한 API제공
Offline & Storage	Offline Web	인터넷 연결이 되지 않은 상태에서도 정상적인 기능을 지원하는 애플리케이션 API제공
	Web Database	웹 브라우저에서 DB 엔진을 이용할 수 있는 기능 제공
	Web Storage	Web APP에서 데이터를 저장 할 수 있는 기능 제공
3D Graphics & Effects	Canvas	2차원 그래픽을 그리기 및 객체에 대한 각종 효과를 주는 기능 제공
	SVG	XML기반의 2차원 벡터 그래픽을 표현하기 위한 SVG언어 지원
Performance & Integration	Web Worker	웹 애플리케이션을 위한 멀티 Thread 기능 제공
CSS3 Styling Effect	CSS3	Web APP의 다양한 스타일 및 효과를 나타내기 위한 CSS3 지원
Connectivity	Web Worker	Web App과 서버간의 양방향 통신 기능 제공

▪ HTML5와 기존 마크업의 차이점

- HTML4에서의 블록과 인라인 개념을 대체하는 콘텐츠 모델
 - 요소를 그룹화하고 각 그룹에는 이름을 붙임.
 - 각 그룹에는 해당 그룹에 소속된 요소에 어떠한 콘텐츠를 넣어야 좋은가를 규정
- 사용자와의 더 정교한 상호 작용과 사용자의 이동성을 지원하는 방향으로 표준화 작업이 진행
- 특정 기업의 독점 기술에 의존하지 않고 plug-in없이 웹을 사용.
- 표준 코덱 기반의 비디오, 오디오 재생 환경을 내장
- 그래픽 지원을 위한 캔버스 제공, 접근성 중시
- 사용자의 이동성을 위해 오프라인 캐시, 웹 파일, 웹 데이터베이스를 지원
- 위치정보, 웹 워커, 웹 소켓등을 제공

❖ HTML5와 HTML4를 비교하여 주로 변경된 부분



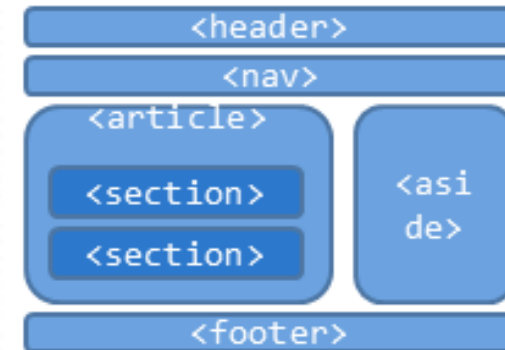
- HTML5 설계원칙

- 호환성

- 컨텐츠의 호환성 : HTML5 이전 버전으로 제작한 컨텐츠가 문제 없이 이용가능
 - 이전 브라우저와의 호환성: HTML5가 지원되지 않는 이전 버전의 브라우저에서 이용가능
 - 기능의 재사용 : 각 브라우저사들이 각자 브라우저에 구현해 놓은 서로 다른 기능들을 HTML5라는 이름 아래 공통된 사양으로 정의하여 브라우저들간의 호환성을 가짐.
 - 이용적인 호환성 : 기존에 사용하던 HTML Tag의 사용법을 그대로 사용
 - 실용성 : 웹 현장에서 필요로 하는 것들을 중점적으로 진행
 - 상호 운용성 : HTML5를 적용한 브라우저면 동일하게 동작
 - 보편적 접근성 : 컨텐츠의 소비자가 기계(Device)와 모든 사람들 이 접근 가능.
 - HTML5언어는 HTML4와 XHTML1 문법과 호환되지만, HTML4 내의 일부 SGML (Standard Generalized Markup Language)문법과는 호환되지 않는 특성을 지니고 있 다.

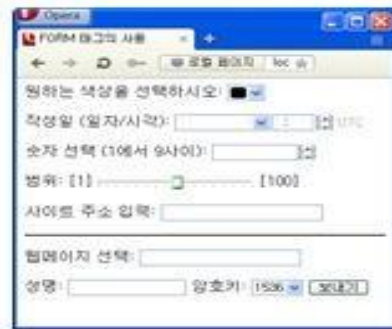
- 의미를 부여할 수 있는 구조적 마크업 요소

- 페이지 단위의 문서 구조를 표현
- <header>와 <footer> 요소
- <article> 요소와 <section> 요소
- <nav> 요소와 <aside> 요소



- 다양하고 편리한 웹 폼 입력 기능

- 편리한 사용자 인터페이스 개발을 위해 Form 기능 대폭 개선
- <input> 요소에 date,number,file 등 각종 type 속성 추가



- 웹 미디어 기능 강화

- 멀티미디어 및 그래픽 관련 기능의 추가
- <video> 요소와 <audio> 요소 : 동영상이나 비디오 스트리밍 처리
- <canvas> 요소 : 래스터 그래픽스 그리기 기능
- SVG(scalable Vector Graphics) : 벡터 그래픽스 처리
- MathML 언어 : 수식의 의미와 모양 표현

- 다양한 API 지원
 - 웹 애플리케이션 개발에 많은 도움을 줄 수 있는 다양한 API 제공
 - 자바스크립트로 구현 가능
 - <video>와 <audio> 요소를 제어하는 API
 - <canvas> 요소에 그림을 그리는 API
 - 별도 사양 API
 - 웹 소켓, 웹 스토리지, 로컬 데이터베이스, 위치정보 등의 API 지원
 - 자바스크립트로 다양하고 강력한 기능의 웹 애플리케이션 구현 가능
 - 드래그앤 드롭 API
 - 텍스트, 아이콘, 이미지, 파일 등 요소를 드래깅할 때 동작을 제어
 - 오프라인 웹 애플리케이션 API
 - 인터넷에 접속하지 못할 때 웹 애플리케이션을 사용하도록 하는 기능
 - 파일 API
 - 클라이언트 내의 로컬 파일을 읽고 쓸 수 있는 기능

Chapter 02

HTML5 문서기본

02 HTML5 기본구조

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

```
<title>HTML 기본구조</title>
```

```
</head>
```

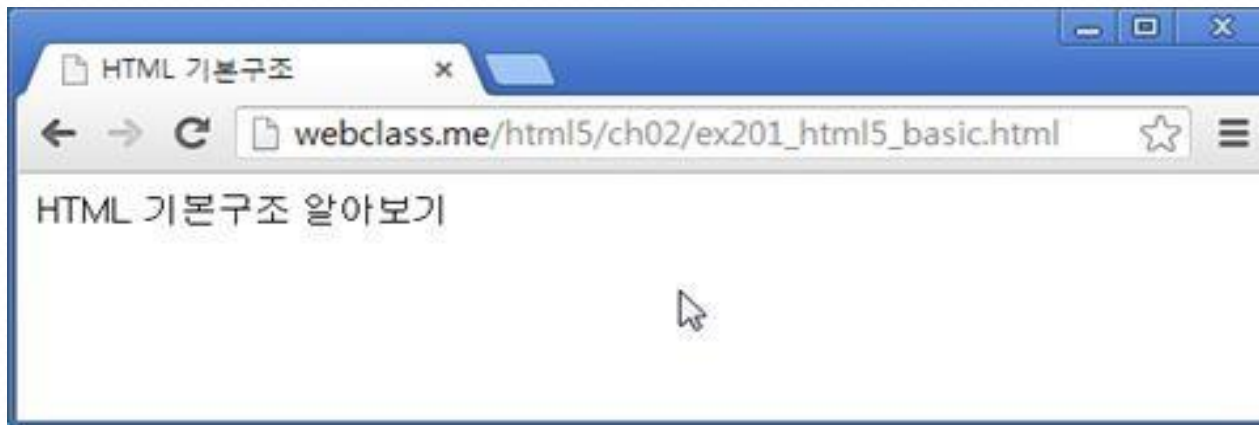
```
<body>
```

HTML 기본구조 알아보기

```
</body>
```

```
</html>
```

실행결과



▪ <!DOCTYPE>

- 문서의 종류를 알려주는 <!DOCTYPE>로 시작
 - 웹 문서에 DOCTYPE 정보가 없으면 웹 브라우저는 웹 문서에서 사용된 규약을 정확히 알 수 없기 때문에 웹 문서를 잘못 처리할 수 있음.
 - 웹 표준에서 DOCTYPE이 중요한 이유는, 개발자가 유효하다고 생각해서 만든 CSS를 적용했어도 DOCTYPE을 잘못 선택하여 비 표준 호환모드로 동작하게 되어 개발자의 의도와는 전혀 다른 결과가 나올 수 있기 때문이다.
 - 웹 표준에서 웹 페이지를 제대로 표현하기 위해서는 올바른 문서 형태를 정의해 줘야 한다. 올바른 문서 형식을 선언해 주는 것은 다양한 웹 브라우저에 따른 렌더링 차이를 최소화할 수 있기 때문에 매우 중요하다고 할 수 있다.

▪ <html> 요소

- 문서 내용을 나타낸다.
- 문서에 대한 정보를 나타내는 <head> 요소
- 문서에 내용을 나타내는 <body> 요소
 - 문서를 구성하는 여러 크기의 단락, 제목, 목록, 표를 표현하기 위한 요소 모음이 나타날 수 있다.

▪ <head> 요소

- 문서 제목을 나타내는 하나의 <title> 요소
- 문서에 대한 설명이 들어가는 시작 태그.
- 문서 제목 외에 다른 정보를 나타내기 위한 <meta> 요소
- 기타 요소 : <link> 요소(링크), <script> 요소(JAVA), <style> 요소 (CSS)

▪ <title> 요소

- 문서의 제목을 부여하는 시작 태그.

▪ <meta> 요소

- 웹 페이지에 추가 정보 전달
- 웹 브라우저의 행동을 제어.

▪ 기타요소

- <script>요소 : 자바스크립트로 기술된 웹 문서의 사용자와의 상호 작용을 위한 스크립트
- <style> 요소 : 웹 문서 표현에 사용되는 CSS(Cascade Style Sheet)
- <link> 요소: 웹 문서 외부에 기술된 자바스크립트와 CSS 파일 링크 정보

02 요소와 속성

▪ 웹 문서 : 태그로 표현되는 요소(element)로 구성

▪ 요소

- 시작을 나타내는 시작태그
- 요소의 내용 (HTML문서의 정리)
- 종료 태그 ("my_id")
- 속성(attribute) 모음



▪ 속성

- 요소에 추가정보(모양, 스타일, 크기지정, 공통 및 개별 설정)를 주기 위해서 사용
- 이름과 값의 쌍의 모음으로 구성됨.
- 요소에 공통적으로 적용되는 것과 요소마다 다르게 적용되는 것이 있음.

02 기타 문서 구성

■ 주석

- 웹 브라우저에서는 보이지 않고, 프로그램의 실행에 영향을 미치지 않고, 문서 작성자가 문서에 대한 설명 글 표기를 남기기 위해 사용

■ 특수문자

<!-- 내용 -->

- '<'문자나'>' 문자와 같은 문자는 웹 문서에서 태그를 표현하기 위해 사용

자주 사용되는 특수문자

	설명	엔티티 이름
	공백문자(스페이스 키를 누른 것) non-breaking-space를 줄여서 엔티티 명을 만들었습니다.	
<	괄호또는 ~이하의 의미로 쓰이는 기호이므로 less than을 줄여서 엔티티 명이 만들어졌습니다.	<
>	괄호또는 ~이상의 의미로 쓰이는 기호이므로 greater than을 줄여서 엔티티 명이 만들어졌습니다.	>
&	앰퍼샌드기호(ampersand)	&
"	큰 따옴표(quotation mark)	"

02 단락 제목과 줄

▪ 단락 제목

- 웹 문서에서 단락 제목을 지정할 때 <h1>~<h6>요소를 이용
-숫자가 클수록 글자가 작아진다.
 - <h1>요소는 첫 번째 단계 수준의 단락 제목
 - <h2>요소는 두 번째 단계 수준의 단락 제목
- 웹 문서의 내용 중 다른 텍스트보다 굵게 나타남

▪ 단락 : <p> 요소

- 웹 문서에서 단락을 만들기 위해 사용
- 단락과 단락 사이에 약간의 공백을 추가하여 시간적으로 두 단락이 구분되어 있음을 표시.

▪ 줄 바꿈 :
 요소

- 줄 바꿈은 단락을 구분하지 않지만, 앞 문장과 다른 문장간에 줄을 바꾸고자 할 때 사용

▪ 가로줄 : <hr> 요소

- 웹 문서에서 가로줄을 표시하여 앞뒤로 글의 주제가 바뀌었다는 것을 나타낼 때 사용

▪ 형식 유지 : <pre> 요소

- 미리 포맷 된 텍스트를 정의합니다.
- 고정 폭의 글꼴로 표시되며, 공백과 줄 바꿈에 관한 태그를 별도로 사용 하지 않아도 즉시 표현
- 특이한 형식의 텍스트 이거나 컴퓨터 코드 등을 곧바로 표현할 때 사용 가능

02 목록 나열하기

- 순서 없는 목록 : 요소 (●로 나타남)
 - 규칙과 순서가 없는 목록(unordered list)을 만들기 위해 사용.
 - 목록의 하위 항목을 나타낼 때는 요소를 사용.
- 순서 있는 목록 : 요소 (숫자로 나타남)
 - 규칙과 순서를 숫자로 나타내는 목록(ordered list)에서 사용
 - 목록의 하위 항목을 나타낼 때는 요소를 사용.
- 정의 목록 : <dl> 요소
 - 사전과 같은 정의 목록(definite list)을 만들기 위해서 사용
 - <dl> 요소: <dt> 요소와 <dd> 요소를 사용
 - <dt> 요소 : 정의할 용어의 제목(title)
 - <dd>요소 : 용어 설명(정의)

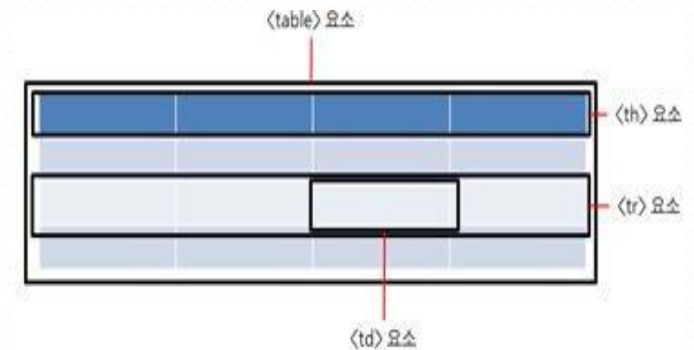
```
<ul>  
  <li>첫 번째 항목</li>  
  <li>두 번째 항목</li>  
  <li>세 번째 항목</li>  
</ul>
```

```
<ol>  
  <li>첫 번째 항목</li>  
  <li>두 번째 항목</li>  
  <li>세 번째 항목</li>  
</ol>
```

02 표 그리기(I)

■ 표의 기본 구조

- <table> 요소
 - 웹 문서에 표를 나타내고자 할 때 사용
 - <tr> 요소 : 하나의 행 (table row)
 - <td> 요소 : 표의 데이터 셀(칸), 데이터를 담고 있는 공간(table data)
 - <th> 요소 : 행 내부의 제목 셀 태그(table header)
 - <th>요소는 웹 브라우저에서 일반적으로 굵은 글씨로 중앙에 표시
- 셀 합치기
 - <td>요소인 rowspan속성과 colspan속성을 사용
 - rowspan : 세로셀 병합(행병합)
 - colspan : 가로셀 병합(열병합)
- 표의 기타 요소



thead	<ul style="list-style-type: none">• 표 머리부분으로 제목으로 구성된 row 집합• 하나 이상의 row로 구성• 표마다 하나의 thead 요소만 사용 가능
tbody	<ul style="list-style-type: none">• 일반적인 row의 집합이며 하나 이상의 row로 구성• 하나의 표에 여러 개의 tbody 사용 가능• 표의 내용 분리가 있을 때 사용
tfoot	<ul style="list-style-type: none">• 표의 꼬리말에 해당하며 표 마지막에 표시• 표마다 하나의 tfoot 요소만 사용 가능

02 표 그리기(II)

■ 테이블 요소

요소	설명
Align	Left center right 테이블 전체의 수평 정렬 Default :left
	예) <table align="center">
Background	Uri 테이블 전체의 배경그림으로 body 요소에서의 사용법과 동일
	예) <table background="../images/bg_hor_line.gif">
Bgcolor	테이블의 배경색
	예) <table bgcolor="silver">, <table bgcolor="#EEEECC">
Height	테이블의 높이
	예) <table height="100%">
Width	테이블의 가로 크기
	예) <table width="600">, <table width="90%">
Nowrap	테이블 전체 셀의 text 줄바꿈 방지
	예) <table nowrap>, <tr nowrap="nowrap">
cellpadding	테이블 cell border와 cell 속의 내용물과의 거리 , 기본값 1
	예) <table cellpadding="5">
cellspacing	테이블 cell간의 거리와 cell과 테이블 border와의 거리, 기본값 2
	예) <table cellspacing="5">

02 문서 특정 부분 구분

■ <div> 요소와 요소

• <div> 요소

- 블록(block) 단위로 영역을 묶음
- <div> 태그는 무조건 한 줄의 공간을 준비(table의 tr과 비슷)
- 테두리, 정렬, 문단모양 등 필요한 기능들은 style을 이용하여 지정

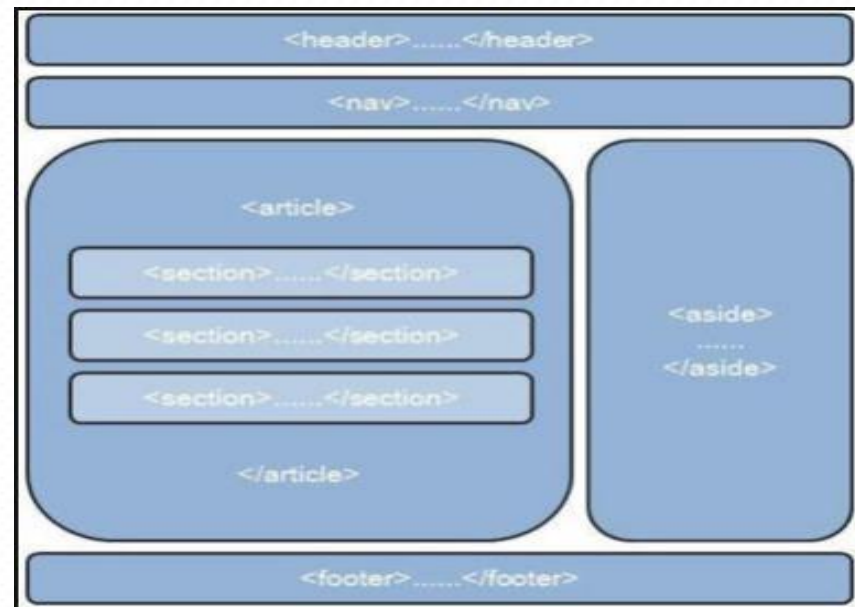
• 요소

- 인라인(inline) 단위로 영역을 묶음
- 입력하는 내용만큼 공간을 준비(table의 td와 비슷)
- 테두리, 정렬, 문단모양 등 필요한 기능들은 style을 이용하여 지정

블록(block) 요소	인라인(inline) 요소
<ul style="list-style-type: none">- 한 줄의 공간을 준비- 줄바꿈 태그를 사용하지 않아도 스스로 줄바꿈을 반영하며, 세로로 나열됨.- 예 : <div>, <h1>, <p>, , <table> 요소 등	<ul style="list-style-type: none">- 주변에 공간준비(x)- 인라인 요소는 일반적으로 줄바꿈 없이 동일한 선상에 표시되며, 가로로 나열됨- 예 : , <td>, <a>, 요소 등

02 문서 구조화

- 문서 구조화의 필요성
 - 웹 문서는 이전에 웹 브라우저를 통해 사람에게 정보를 보여주고 전달하는 요도로 사용
 - 최근에는 검색엔진이나 장애인을 위한 스크린 리더와 같이 사람이 아닌 컴퓨터가 문서를 이해해야 하는 경우가 늘어나고 있음
- 문서 구조화
 - 웹 문서 안에 있는 문장의 의미를 특정 의미를 가진 요소로 사용하여 명확하게 나타냄



Chapter 03

링크와 멀티미디어

03 문서 구조화

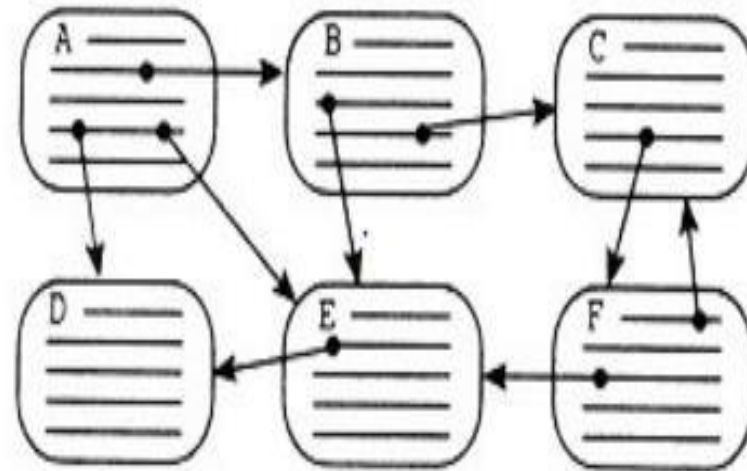
▪HTML(HyperText Markup Language)

- 웹 브라우저를 통해 보여지는 모든 문서와 사이트를 이루는 기본적인 언어

하이퍼텍스트	하이퍼미디어
<ul style="list-style-type: none">- 사용자가 연상하는 순서에 따라 원하는 정보를 얻을 수 있는 비순차적 시스템- 하이퍼링크라고도 불림	텍스트, 이미지, 음성, 영상 등 다양한 미디어를 Hyper Link로 연결해 구성한 시스템

▪ 하이퍼링크/하이퍼미디어의 구조

- 각 정보의 조각은 링크에 의해 서로 연결
- 모든 정보의 접근은 연결링크를 선택



03 노드, 링크, 앵커

■ 노드

- HTML 문서나 멀티미디어 정보를 표현하는 기본단위

■ 링크

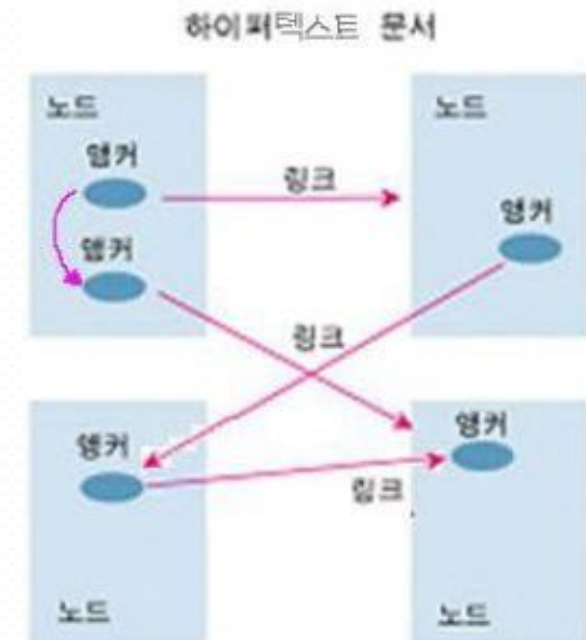
- 서로 다른 웹 페이지 사이를 이동하거나 웹 페이지 내부에서 특정한 위치로 이동할 때 사용
- 노드를 연결하여 내비게이션이 가능토록 하는 구성요소.

■ 앵커

- HTML 문서 내에서 링크의 출발점이나 도착점

■ HTML 문서에서 사용되는 링크의 종류

- 문서 내의 다른 지점으로의 링크
- 외부 URL로 연결하는 링크
- 특정 단어나 문장, 혹은 이미지에서 다른 문서로 이동하는 링크



03 <a>요소

■ <a>(anchor)요소

- 한 개 이상의 HTML 문서가 있을 때, 문서들을 서로 연결하거나 다른 다른 웹사이트의 어떤 문서를 참조하고자 할 때 사용.
- 텍스트에 다른 문서를 연결하여 하이퍼링크 즉, 브라우저에서 선택 가능한 링크로 만듦.

` 링크 텍스트 `

■ href(hyper reference)속성 : 이동하고자 하는 문서의 위치 지정

- 절대 경로 : 전체 구조내에서의 위치를 표시하는 방법

```


```

- 상대 경로 : 작업 하고 있는 파일의 현재 위치를 기준으로 경로를 지정하는 방법

```
<a href="파일명">연결할 문서</a> 또는 <a href="./파일명">연결할 문서</a>
<a href="information.html">상세안내</a> 또는 <a href="./information.html">상세안내</a>
(./는 같은 폴더를 의미)
```

- title 속성

-하이퍼링크에 대한 설명을 하고 싶을 때 사용(마우스를 가져가면 말풍선이 나타남)

03 <iframe> 요소

- <iframe> 요소 : 별도의 웹 페이지 문서를 테이블, 이미지와 같은 개체처럼 특정 위치에 삽입하는 기능 (inline frame)
- 하이퍼링크 문서와 같이 절대 경로와 상대경로 모두 사용 가능
- `<iframe src="페이지"></iframe>` 브라우저에서 지원은 하지만 html5에서는 더 이상 지원 하지 않음.

속성	기능
name	아이프레임의 이름을 지정합니다. 지정된 이름은 하이퍼링크 타겟 설정에 사용
src	아이프레임 창에 표시되는 문서 경로를 지정
width	아이프레임의 가로크기를 %와 px 크기로 지정
height	아이프레임의 세로크기를 %와 px 크기로 지정
seamless	속성을 사용하면 아이프레임이 부모 문서와 같이 지정되어 CSS, 자바스크립트 등의 영향을 받게된다.
srcdoc	HTML 태그를 직접 마크업하여 아이프레임 창에 개체로 표시되게한다.
sandbox	아이프레임 개체에 대한 제한 사항을 둘 때 사용한다. 값으로 allow-top-navigation (최상위 콘텐츠 사용 가능), allow-same-origin (도메인 내의 다른 지원 사용 가능), allow-forms (폼의 데이터 값 사용 가능), allow-scripts (스크립트 사용 가능)를 사용할 수 있다.

03 이미지 삽입

■ 요소

- 요소는 HTML 페이지에 있는 image를 정의
- 이미지들은 HTML 페이지에 삽입되는 것이 아니라 링크를 걸어 표현
- 다른 문서로 가기 위해 이미지에 링크를 거는 경우 <a> 요소 안에 태그를 추가

``

속성	값	설명
alt	text	이미지가 보여질 수 없을 때 이미지에 대한 대체 텍스트. 이미지 파일의 주소가 잘못되거나, 인터넷 연결이 너무 느려서 미처 이미지를 표시하지 못하는 경우 등
height	pixels	이미지의 높이를 지정
src	URL	이미지의 주소 명시
ismap	ismap(boolean 형식)	Server-side 이미지맵 정의. <map> 태그의 name 또는 id 속성과 관련.
usemap	#mapname	Client-side 이미지맵 정의 <map> 태그의 name 또는 id 속성과 관련.
width	pixels	이미지의 넓이를 지정

03 <figure>, <figcaption>

- <figure> 요소
 - 포함 된 내용에 대한 이해를 돕기 위해 그림, 다이어그램, 사진, 코드, 샘플 등 포함 할 때 정의
- <figcaption> 요소
 - <figure> 요소를 이용할 때 캡션을 나타내는데 사용
 - <figure> 요소의 첫 번째 또는 마지막에 자식 요소로 사용
 - <figure> 요소 태그에서만 사용

<figure>

<figcaption> 사진이나 삽화 등에 대한 설명</figcaption>

</figure>

03 멀티미디어

■ 지원하는 오디오/비디오 파일 형식

- MP3
 - ❖ MPEG-1의 오디오 규격으로 개발된 형식, 대중적으로 널리 사용
- Wave
 - ❖ 마이크로소프트와 IBM이 개발, 파일 용량이 큰 비압축 방식의 오디오 형식
- MPEG4
 - ❖ MPEG-4의 part14에서 규정된 비디오 파일 형식, H.264 코덱 사용
- Ogg
 - ❖ 스트리밍 방식의 멀티미디어 표현을 위한 공개소스 기반 형식
 - ❖ Vorbis, FLAC 등의 오디오 코덱, Ogg Theora 등의 비디오 코덱 사용
- WebM
 - ❖ 구글이 HTML5의 동영상에 사용하기 위해 개발

➤ 웹 브라우저에서 오디오/비디오 코덱의 지원 현황

코덱	웹브라우저				
	 크롬 (버전6 이상)	 파이어폭스 (버전3.6 이상)	 익스플로러 (버전9 이상)	 사파리 (버전5 이상)	 오페라 (버전10.6 이상)
mp3	지원	미지원	지원	지원	미지원
Wav	지원	지원	미지원	지원	지원
Ogg/Theora	지원	지원	미지원	미지원	지원
mp4 (H.264)	지원	미지원	지원	지원	미지원
WebM	지원	미지원	미지원	미지원	지원

03 오디오 사용하기<audio>

▪ <audio>

`<audio controls autoplay src="재생할 오디오 파일 이름">`

속성	설명
autoplay	파일 자동 재생
controls	제어기 표시
loop	반복 재생
preload	미리 로드 되어야 하는지 여부 지정 None:재생을 하기 전까지 오디오 파일을 다운로드 안함 Auto:페이지를 로드하고 바로 오디오 파일 다운로드 Metadata: 사용자가 재생 시키기 전까지는 오디오의 관련정보 등과 같은 메타데이터만 다운로드
src	재생할 오디오가 존재하는 경로를 지정
volume	0.0부터 1.0까지 볼륨조절

▪ <source>

- 오디오나 비디오 파일의 경우 브라우저 마다 재생할 수 있는 파일형식이 다르므로 이에 대응하기 위해 audio 태그나 video태그에 source태그를 사용하여 재생할 수 있는 여러 파일 형태를 제공
 - Type 속성
 - 파일타입을 지정 (audio/mpe)
 - 속성을 지정해 놓으면 브라우저에서는 파일을 실제 로드 하지 않고도 지정된 형식을 통해 자신이 재생할 수 있는 파일인지 판단함.
- ```
<audio controls="controls">
 <source src="" type="" />
</audio>
```

### ■ <video>

`<video controls src="비디오 파일 이름" width="폭" height="높이">`

속성	설명
controls	재생 control 표시 제어
autoplay	동영상이 로딩 되면 재생 시작
autobuffer	이 속성을 사용하게 되면 사용하기 전부터 다운로드가 진행. 사용자가 재생할 때 쯤이면 동영상이 어느 정도 다운로드 된 상태.
poster	동영상이 다운로드 중이거나 버퍼링 중에 나타낼 이미지 지정
loop	동영상 반복 재생
width	동영상의 너비
height	동영상의 높이
preload	Auto: 웹 브라우저가 페이지를 로드하고 바로 비디오 파일 다운로드 None: 재생을 시작 하기 전까지 비디오 파일을 다운로드 안함 Metadata: 사용자가 재생시키기 전까지는 비디오 관련정보의 메타데이터만 다운로드.

```
<video controls=controls poster=".jpg">
```

```
<source src="" type="video/mp4">
```

```
<p>이문장이 보인다면 HTML5의 video태그를 지원하지 않는
웹 브라우저를 사용하고 계시는 겁니다.</p>
```

```
</video>
```

# Chapter 04

## 웹 폼

## 04 <form> 요소

### ▪ <form>요소 사용

- 사용자와 애플리케이션이 상호작용을 돕는 역할
- 사용자로부터 정보를 받을 때 사용

### ▪ <form>요소의 역할

- 사용자가 입력하는 정보를 하나로 묶어서 애플리케이션에 전달할 수 있도록 다양한 입력 양식을 그룹핑하고 전송방법 설정

### ▪ <form>요소의 주요 속성

속성	값	설명
accept-charset	문자셋	form데이터 서버 전송 시 문자 인코딩 지정
action	url	form데이터를 전송할 위치 지정
enctype	text/plain multipart/form-data application/x-www-form-urlencoded	form데이터 서버 전송시의 인코딩을 지정 (단,method="post" 일 경우만 지정 가능)
method	get,post	form데이터 전송 시 사용하는 http 메소드를 지정
name	text	form의 이름 지정
target	_self, _parent, _top	form데이터 전송 후 결과를 보여줄 창 위치를 지정

## 04 GET/POST방식

### ■ 폼 <form>의 역할

- 다양한 <input>요소를 통해 입력 받은 데이터를 웹 서버 애플리케이션에 전달하여 그 실행 결과를 받는 것.

GET/POST방식  
요청(데이터 전송)  
→  
←  
응답(실행결과)



GET	POST
<ul style="list-style-type: none"> <li>• URL주소에 문자열 형태로 정보가 담겨서 전송.</li> <li>• URL 뒤에 ?에 이어서 "변수명=값, 변수가 여러 개일 경우 &amp;로 구분</li> <li>• 퍼머링크로 사용될 수 있다.(정보를 식별하는 고유한 식별자.(고유한 주소체계))</li> <li>• 속도는 POST방식에 비해 빠르고, 폼을 사용하지 않더라도 파라미터 전송 가능.</li> <li>• 전송할 수 있는 정보의 길이가 제한(1024byte).</li> <li>• 주소창에 전달하는 값이 노출되어 보안에 취약함.</li> </ul>	<ul style="list-style-type: none"> <li>■ 프로그램의 입출력 방식을 사용하여 데이터의 양에 제한이 없다.</li> <li>■ 전달하고자 하는 정보가 HTTP body에 포함되어 전달되며 사용자의 눈에 직접적으로 표시 되지 않기 때문에 보안이 필요한 경우에 많이 사용.</li> <li>■ 퍼머링크로 사용할 수 없다.</li> <li>■ GET방식에 비해 속도가 느리다.</li> </ul>

## 04 문서 특정 부분 구분

### ■ input요소 type 속성 값의 종류

type 종류	설명
text	사용자가 텍스트를 입력할 수 있는 한 줄 짜리 입력 필드 정의
button	클릭 ㉠ 가능한 버튼 정의(주로 자바스크립트를 활성화 시키기 위해서 사용)
submit	전송버튼(폼 데 ㉠ 터를 서버로 전송)
reset	리셋 버튼(폼의 모든 필드를 초기값으로 리셋)
password	비밀번호 입력 필드(필드 내의 각 글자는 다른 글자로 대체 표시됨)
checkbox	체크박스
radio	라디오 버튼
file	파일 업로드를 위한 파일 입력과 선택 필드 정의
image	㉠ 미지를 전송 버튼으로 정의
hidden	숨겨진 입력필드 정의

### ■ 텍스트 영역 필드 : <textarea>

- 여러 줄에 걸치는 텍스트 입력
- Cols 속성: 텍스트 영역의 한 줄에 해당하는 문자 수, 즉 열의 개수
- Rows 속성 : 텍스트 영역의 행의 개수

```
<textarea name="변수명" cols="열의 수" rows="행의 수">
텍스트 영역에 표시되는 문장
</textarea>
```

## 04 기타 입력 필드

- 파일 선택하기 `<input type="file">`
  - 사용자가 폼 입력에서 파일 선택
- 데이터 숨김 `<input type="hidden" name="변수명" value="값"/>`
  - 사용자가 입력하거나 선택하는 데이터가 아니라 시스템에서 특정 데이터 항목을 입력 받아 처리하고 싶은 경우
  - 화면에 아무것도 보여주지 않지만, value속성값과 name을 전달할 때
- 텍스트 라벨(텍스트를 눌러도 버튼이 눌림) `<label for="입력아이디">`
  - 텍스트 라벨과 특정 입력 필드를 연결
  - For속성과 이와 연결되는 입력 양식의 id 속성에 동일한 id의 값

라벨을 클릭 해도 됩니다. <br><br><br>

```
<label for="men"> 남</label>
```

```
<input type="radio" name="sex" id="men" value="men">
```

```
<label for="girl">여</label>
```

```
<input type="radio" name="sex" id="girl" value="girl">
```

라벨을 클릭 해도 됩니다.

남 ☐ 여 ☒

## 04 추가된 입력 형식

### ▪ input 태그의 추가된 속성들

속성	설명
autocomplete	브라우저의 자동 완성 기능 제어
autofocus	페이지가 로드하자마자 입력 필드안에 마우스 커서 표시
list	입력 가능한 값을 가지고 있는 데이터리스트 요소
max,min,step	입력 값의 범위 지정
placeholder	텍스트 필드에 입력 힌트 표시
required	필수 입력 필드

### ▪ Input 요소 이외에 추가된 입력 요소

- <output>요소 : 폼의 처리 결과
- <datalist>요소 : 입력 양식에 대한 내용을 옵션 리스트로 제공
- <keygen>요소 : 암호화 키 생성

### ▪ 유효성 검사

- 입력란에 입력 값이 형식에 맞게 입력되었는지 검사 하는 기능
- required 속성 : 오류 발생시 메시지를 표시하고 폼 전송 중단
- novalidate속성 : 유효성 검사 대상에서 제외
- 요소와 속성에는 대문자나 소문자, 혹은 이를 혼합해서 사용할 수 있다.
- 꼭 닫혀야 하는 요소 중 일부는 열린채로 있어도 유효성검사에서 오류가 발생하지 않는다 (예 p,il,dt)



## 04 서식 있는 텍스트 입력

- **이메일 주소 입력 : `<input type="email"/>`**
  - 이메일을 입력받기 위한 창
  - Multiple 속성 : 여러 개의 이메일 주소를 입력, 콤마로 구분
- **검색창 입력 : `<input type="search"/>`**
  - 검색을 위한 문자열을 입력할 수 있는 입력 필드.
  - 문자열을 입력하면 x표시가 나타남.
- **URL 주소 입력 : `<input type="url"/>`**
  - 웹 주소의 url을 직접 입력하는 경우
  - Value속성에 http://를 입력하여 초기값 설정 가능
- **날짜 입력**
  - `<input type="date"/>` : 연, 월, 일의 날짜를 입력
  - `<input type="month"/>` : 연, 월만 입력
  - `<input type="week"/>` : 연, 주 입력

rrr

확인

! 이메일 주소에 '@'를 포함해 주세요. 'rrr'에 '@'가 없습니다.

Search : HTML5

x

r

확인

! URL을 입력하세요.

연도-월-일

2014년 11월

일	월	화	수	목	금	토
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

연-월 입력 : ----년 --월

확인

연-월 입력 : ----, --번째 주

확인

## 04 서식 있는 텍스트 입력

### ■ 시간 입력

- `<input type="time"/>` : 시간 입력
- `<input type="datetime"/>` : 날짜와 시간 선택
- `<input type="datetime-local"/>` : 원하는 지역의 현지 시간 입력

시간 입력 예제

time (시간)  
오후 12:03

datetime (날짜-시간)  
2013-04-15 오후 12:03

datetime-local (지역시간)  
2013-04-15 오후 12:03

### ■ 색상 입력 : `<input type="color"/>`

- 색상을 선택할 수 있는 색상 메뉴 제공



### ■ 범위 입력 : `<input type="range"/>`

- 스크롤을 움직여 범위
- min, max 속성

범위 : 1  10

확인

## 04 서식 있는 텍스트 입력

### ■ 숫자 입력: `<input type="number"/>`

- 나타나는 화살표 버튼으로 숫자를 조정할 수 있는 양식
- Value 속성, 최소값(min), 최대값(max)속성, 증가/감소 간격을 지정하는 step속성

숫자 입력 :

### ■ 데이터 목록 : `<datalist>` 요소

- 데이터목록의 옵션 항목들을 제시어 목록으로 사용
- 검색어 자동완성 혹은 제시어 기능

언어 : `<input type="text" list="Language">`  
`<datalist id="Language">`  
`<option value="HTML5"/>`  
`<option value="JSP"/>`  
`<option value="Javascript"/>`  
`</datalist>`

언어 :

- HTML5
- JSP
- Javascript

## 04 문서 특정 부분 구분

- [ 예 제 ] 아래 사진의 모습이 나타나도록 만드시오.

### 상담 예약

성명 :

전화 :

이메일 :

원하는 상담사 :

예약 희망일 :

희망 시간 :  ~

# Chapter 05

## CSS 스타일 시트

## 05 CSS 와 스타일 시트

---

### ▪ 스타일 시트란?

- 웹 문서의 출력될 스타일(모양)을 다양하며 손쉽게 적용
- 콘텐츠의 내용과 디자인의 분리가 가능
- 문서 전체의 일관성을 유지할 수 있고 작업 시간 단축

### ▪ CSS (Cascading Style Sheet) 특징

- 웹 컨소시엄에서 웹 문서용으로 개발한 스타일시트 언어
- 마크업 언어로 작성된 문서(모양과 서식)를 표현하기 위한 스타일 시트 언어

### ▪ CSS3란?

- Css에 대한 최신 표준.
- 모듈 기반으로 개발(각종 브라우저나 디바이스가 필요에 따라 원하는 css 모듈만을 탑재하거나 또는 필요한 모듈만으로 빠르게 자주 업데이트하는 것을 도움)
- 포토샵 등 그래픽 디자인에 의존하던 영역(그림자 효과, 둥근 모서리 등 ) 애니메이션 효과등이 가능해지면서 더욱 화려하고 역동적인 웹 사이트를 만들 수 있게 됨.

## ■ CSS 기본 문법

- 선택자(Selector) :스타일을 넣기 원하는 HTML엘리먼트.
- 선언(Declaration) : 속성(property)과 값(value)으로 구성.
  - 속성과 값은 콜론(:)으로 구분하고, 선언은 세미콜론(;)으로 종료

Selector Declaration Declaration

```
h1 { color:red; font-size:12px; }
```

Property Value Property Value

## ■ 기본 예제

- 스타일 지정이 없는 예제

### 스타일 시트 기초 예제

스타일 지정이 없는 예제

스타일 시트 기초 예제

스타일 지정이 없는 예제

- 스타일 지정이 있는 예제

```
<style type="text/css">
h3{color:blue}
p{font-size: 12px; font-style: italic;
text-decoration: underline;}
</style>
</head>
<body>
<h3>스타일 시트 기초 예제</h3>
<p>스타일 지정이 있는 예제</p>
</body>
```

## 스타일 시트 기초 예제

### 스타일 지정이 있는 예제

## ■ 스타일시트 선언

### • 내부 스타일시트

➤ html 코드 내에서 정의하는 방식

➤ html의 <header>와 </header> 사이에 <style>태그를 이용하여

```
<head>
<style type="text/css">
p{font-size:10px}
</style>
</head>
```

### • 외부 스타일시트

➤ 외부에서 작성된 "style.css"를 불러오는 방식

➤ html의 <header>와 </header> 사이에 <link>태그를 이용하여 연결

➤ 외부 파일에 의한 css 적용은 말 그대로 확장자가 .css인 파일을 외부에 저장해 놓고 HTML 문서에 @import 방식 또는 link를 통해서 불러오는 방법이다.

➤ 외부파일에 의한 방식은 <head>와 </head> 사이에 연결 방법에 따라 다음과 같이 지정해서 css 스타일을 불러옵니다.

@import 방식

```
<style type="text/css">
 @import url(main.css);
</style>
```

link 방식

```
<link href="main.css" rel="stylesheet" type="text/css"/>
```



## ■ 인라인 방식으로 스타일 적용

- 인라인 방식으로 스타일을 적용하는 방법은 절대 권장하는 사항은 아니다.
- 5페이지 내외의 간단한 웹 페이지를 만들어야 할 때 또는 웹 페이지를 메일로 전송할 때 이외에는 사용하지 말아야 한다.
- 단점으로는 인라인 방식으로 사용하면 웹 페이지 유지 보수가 힘들다.
- 모든 요소에서 사용할 수 있는 style속성 사용
- 객체의 코드 내에 직접 사용하는 방식

### ❖ 적용방법

```
<body>
<strong style="font-size:40px; color:blue;">
인라인 스타일시트

</body>
```

- 인라인 방식은 태그에 직접 스타일을 적용하는 것이기 때문에 어떻게 적용되는지 바로 확인 할 수 있다.

### ■태그 선택자(tag selector)

- 가장 기본적으로 CSS를 작성할 때 사용하는 방식
- HTML 엘리먼트에 직접 스타일을 지정.
- 페이지 안에 있는 모든 엘리먼트에 영향을 주어 스타일 적용
- 다중 속성값은 콤마로 나열 : 순서대로 가능한 속성값 적용

예) `h1{color:blue; text-align: center;}`

`<h1>h1엘리먼트에 스타일 적용</h1>`

### ■클래스 선택자(class selector)

- 같은 태그에 다른 스타일을 적용, 혹은 여러 태그에 특정 스타일을 공통으로 적용할 때 사용
- 클래스 이름은 반드시 영문으로 시작,숫자 불가능
- 원하는 엘리먼트 정밀하게 제어 가능
- 소문자와 대문자는 구분하며 속성과 송석성 사이는 ;(세미콜론)으로 구분한다.

예) `p` 엘리먼트에 `align_center`라고 클래스명을 짓고 스타일 적용

`.align_center{text-align:center;}`

`<p class="align_center">p태그에 가운데 정렬</p>`

### ■아이디 선택자(id selector)

- 클래스와 아이디의 차이점은 클래스는 여러 개를 사용할 수 있지만 아이디는 고유성을 가지므로 한문서에 한번만 사용할 수 있다는 점이다.
- 해당 아이디로 설정된 태그에만 스타일 지정.
- 화면상에서 큰 레이아웃을 잡을 때 사용.

예) `#id1{color:blue;text-align:center;}`

### 가상클래스 선택자(pseudo class selector)

- 가상 클래스 선택자는 실제로 클래스를 부여하지 않지만 부여된 것처럼 동작하는 선택자
- 요소 이름 다음 콜론(:) 뒤에 예약어

예) 선택자: 가상클래스 선택자{속성:값;}

- 대표적인 가상클래스 선택자
  - 하이퍼 링크 - :link :visited
  - 마우스 - :active :hover :focus
  - 콘텐츠 - :before :after

link	한번도 방문하지 않은 a 엘리먼트의 링크 선택
visited	한번 이상 방문한 a엘리먼트의 링크 선택
hover	마우스 커서가 올라간 상태를 표현
active	마우스로 누르고 있는 상태를 표현
before	선택자 요소 앞에 문자 등을 넣고 꾸밀 때 사용
after	선택자 요소 뒤에 문자 등을 넣고 꾸밀 때 사용

### 그룹 선택자

- 선택자 이름만 다르고 스타일 내용을 똑같이 반복하는 경우 그룹으로 묶어 코드를 단축

```
h1{font-family:italic;}
h2{font-family:italic;} --> h1,h2{font-family:italic;}
```

- 부가적인 선택자의 종류를 다음 표와 같이 정리 할 수 있다.

선택자		CSS 적용 방법
종속 선택자	선택자에 종속된 선택자	p.className/ table.className /p#idname
그룹 선택자	⌚ 러 선택자를 복합적으로 적용 할 때	P, .classname, #idname, table (각각의 선택자를 ,(콤마)로 분리)
자식 선택자	선택자 바로 밑의 자식 선택자	p > a / ul > li (보다 큰 기호 사용 )
인접 선택자	불 ⌚ 있는 선택자와 선택자에 특 정 효과를 줄 때 사용	h1 + p (+기호 사용)
하위 선택자	선택자 내부의 모든 자식 선택자	P a/ p.calssname / p#idname (선택자와 선택자 사 ⌚ 에 빈칸)

## 05 문자와 색상 지정(I)

### ■ 폰트(Font) 속성

- font-size : 텍스트 크기 제어
  - 텍스트의 크기를 조정하는 속성으로 <font>태그의 size와 동일한 역할
  - font태그는 1~7까지 조정,CSS는 무한대로 입력 가능.
- font-family : 텍스트 글씨체 제어
  - 원하는 글씨체를 직접 입력하는 형식
  - 텍스트 글씨체를 조정하는 속성으로 <font>태그의 face과 동일한 역할
- font-color : 텍스트 색상 제어
  - 텍스트 색상을 조정하는 속성으로 <font>태그의 color와 동일한 역할
  - 색상의 고유명칭이나 색상 코드표를 입력
- font-weight : 텍스트 굵기 제어
  - 텍스트 굵기를 제어하는 속성으로 <b>,<strong>과 동일한 역할
  - 미 입력시 기본 크기
- font-style : 텍스트 스타일 제어
  - 텍스트 스타일을 제어하는 속성으로 <i>와 동일한 역할
  - 사용되는 속성 : normal ,italic

표현방식  

```
<style type="text/css">
p{font-size:12px;}
</style>
```

표현방식  

```
<style type="text/css">
p {font-family: 굴림;}
</style>
```

표현방식  

```
<style type="text/css">
p {font-color:white; or
 font-color:#FFFFFF;}
</style>
```

표현방식  

```
<style type="text/css">
p { font-weight:bold; or
 font-weight:thin }
</style>
```

표현방식  

```
<style type="text/css">
p { font-style:italic; or
 font-style:normal; }
</style>
```

### ■ 문자(Text) 속성

- text-align : 문자 정렬 제어
  - right, left, center 등 값 사용
- text-decoration : 문자에 줄긋기 제어
  - 해당 문자에 밑줄, 가운데 줄 등 줄긋기 속성
  - 속성값 : underline, overline, line-through
- text-transform : 영문 문자의 대소문자 제어
  - 입력된 문자중 영문 문자에만 적용되는 속성
  - 입력된 영문자의 대소문자 제어
  - 속성값 : uppercase, lowercase, capitalize
- text-indent : 들여쓰기 제어
  - 속성값 : px, %(퍼센트), em
  - 긴 단락의 첫째 줄을 들여쓰기 할 수 있는 속성

#### 표현방법

```
<style type="text/css">
.text-center{ text-align: center; }
.text-right { text-align: right; }
.text-left { text-align: left; }
</style>
```

#### 표현방법

```
<style type="text/css">
p {text-decoration: line-through;}
h {text-decoration: underline;}
</style>
```

#### 표현방법

```
<style type="text/css">
p {text-transform: uppercase;}
h {text-transform: lowercase;}
</style>
```

#### 표현방법

```
<style type="text/css">
p {text-indent: 20px;}
</style>
```

### ■ 색상(color)

- 10진수 표기법 : (255, 0, 0)
- 16진수 표기법 : (#ff0000)
- 색상 이름 표기법 : (red)
- rgba 표기법 : (255, 0, 0, 0.5)
- 백분율 표기법 : (100%, 0%, 0%)

### ■ 배경(background) 속성

- 배경의 속성 값들은 축약형과 일반형이 존재한다.
- ❖ 방법 1 축약형 = background: 속성값;  
color, image, repeat, attachment, position 등이 있다.

속성	설명
background-color	배경색 지정
background-image	배경 이미지를 지정해 주는 것으로 ("URL")로 지정해 주거나 (URL) 또는 ('URL')로 지정 가능합니다
background-repeat	반복 요소를 말하는데, repeat , no-repeat, repeat-x, repeat-y로 나눌 수 있습니다. repeat는 요소를 무한 반복하는 것이며, no-repeat는 반복 금지, repeat-x는 X축으로 반복, repeat-y는 Y축으로 이미지를 반복하는 것을 말합니다. repeat 요소를 적절하게 사용하는 것만으로도 사이트를 매우 풍부하게 꾸밀 수 있습니다.

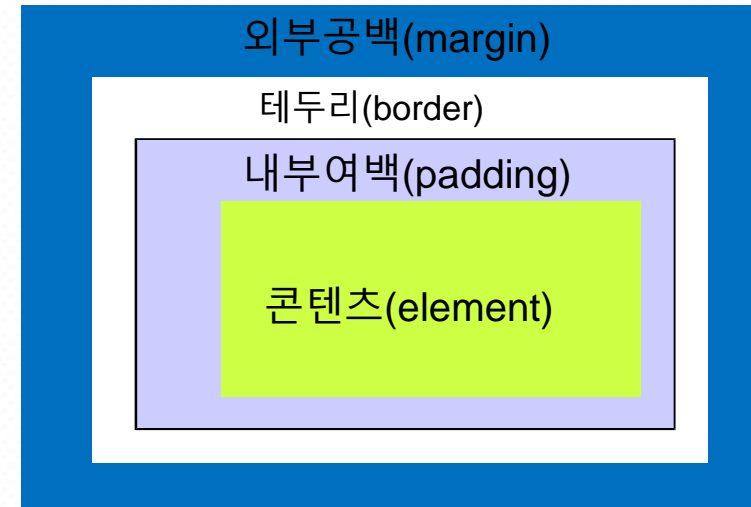
- 배경(background) 속성

속성	설명
background-position	백그라운드 이미지의 위치를 정해 줄 수 있는데, X Y 축을 기준으로 top, center, bottom, left, right 로 지정할 수 있습니다. 보통 2가지 요소를 복합적으로 사용하거나, 하나만 사용하거나 합니다.
background-attachment	scroll과 fixed로 나눌 수 있는데, 이게 정말 재미있습니다. 예를 들어 웹 문서가 아주 길 경우 백그라운드 이미지가 스크롤을 따라서 같이 움직이면, scroll을 사용하고, 백그라운드 이미지를 고정할 경우에는 fixed를 사용하게 됩니다. 즉 이 말은 백그라운드 이미지의 높이가 100픽셀인데 한 페이지가 너무 길어 1000픽셀 정도의 높이를 가질 경우, scroll을 해주면 백그라운드 이미지도 따라서 움직이지만, fixed 해 버리면, 백그라운드 이미지는 스크롤을 할수록 없어져 버립니다.
background	배경 속성 한번에 모두 지정
background-size	배경 이미지 크기 조절
background-clip	배경 적용 범위 조절
background-origin	배경 이미지 위치 기준 조절



### •박스 모델 공간 구성

- 박스모델이란?
  - 요소가 차지하는 공간의 개념
  - 모든 박스는 선택 가능한 패딩, 테두리, 마진이 있는 콘텐츠 영역으로 구성



### ▪박스 모델 공간 속성

- margin(외부 공백)이란?
  - 선택 가능한 요소로 테두리를 둘러싸며, 빈 공간의 역할
  - margin-top, margin-bottom, margin-right, margin-left
  - margin : 10px 20px 30px 40px -> 4방향(상, 우, 하, 좌 순)-시계방향
  - margin : 10px 20px -> 2방향 (상하 / 우좌)
- padding(내부 여백)이란?
  - 콘텐츠와 박스 사이에 여백을 주는 역할, 선택사항이 아니므로 반드시 사용할 필요는 없다.
  - padding-top, padding-bottom, padding-right, padding-left

# 05 border

## • border(테두리)란?

- 콘텐츠 주위를 둘러싼 선이며, 콘텐츠와 다른 요소를 시각적으로 분리하는 역할
- CSS로 레이아웃을 디자인할 때 border는 가장 많이 사용되는 요소 중 하나이다.
- border의 속성도 margin과 padding 값을 입력할 때와 마찬가지로 단축형과 일반형 두 가지 방법이 있다.

### ▪ 방법 1 일반형

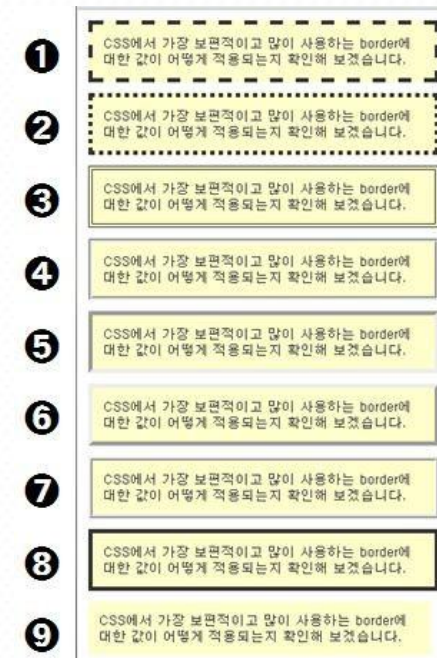
-border-width: 속성값; -> 두께를 지정합니다.

-border-style: 속성값; -> dashed(1),dotted(2),  
double(3),groove(4),  
inset(5), outset(6),  
ridge(7),solid(8), none(9)

-border-color: 속성값; -> 여기는 색상값을 지정합니다.

### ▪ 방법 2 단축형

-border: 두께 스타일 컬러; -> 속성을 부여하는 순서는 상관없다.



## ■ 글머리 기호(list)

- list속성 (list-style-type)
  - 리스트를 표시 할 경우 앞에 생기는 기호

목록	속성값	설명
순서 있는 목록	1.decimal ii.lower-roman III.upper-roman d.lower-alpha E.upper-alpha	정수 기호 로마 소문자 기호 로마 대문자 기호 알파벳 소문자 기호 알파벳 대문자 기호
순서 없는 목록	●disc ○circle ■square none	검은 동그라미 기호 빈 동그라미 기호 검은 네모 기호 마커 제거 기호
이미지 목록	Image:url	사용자 지정 이미지 기호

- 글머리 기호 위치(list-style-position)
  - outside : 목록의 바깥쪽에 위치
  - inside : 목록의 안쪽에 위치

위치 : list-style-inside

• 안쪽  
• 안쪽  
• 안쪽

위치 : list-style-outside

• 바깥쪽  
• 바깥쪽  
• 바깥쪽

## 05 레이아웃 설정(I)

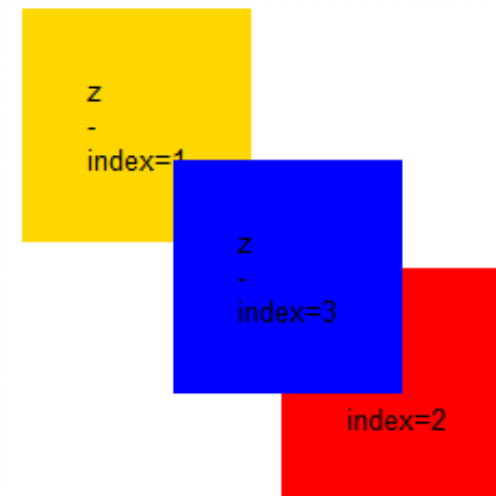
### ■ 요소 위치 정하기

- 요소의 위치는 **top, bottom, right, left** 속성으로 지정
- 위치 설정 속성(**position**)

속성값	설명
정적 위치 설정(static)	정상적인 흐름에 따른 배치.(top ,bottom, right, left속성값이 적용 되지않는다)
상대 위치 설정(relative)	상위 요소에 대한 상대 배치.
고정 위치 설정(fixed)	웹 브라우저가 지정한 위치에 고정되는 배치.
절대 위치 설정(absolute)	상위 요소에 대한 절대 배치.

### ■ 콘텐츠 순서 지정하기

- z-index속성을 사용하며, position 속성과 함께 사용
- 숫자가 높을 수록 앞에 배치되며 기본값은 0(auto)
- 적용 대상은 static값이 아닌 요소들



### ■ 플로팅 박스 위치 정하기

- 플로팅 박스와 주변 콘텐츠와 배치 방법은 float 속성을 사용하여 지정.

속성값	설명
inherit	기본 값이며 상위 태그의 상태를 상속
left	플로팅 박스는 왼쪽, 콘텐츠는 오른쪽, clear속성이 있으면 달라짐.
right	플로팅 박스는 오른쪽, 콘텐츠는 왼쪽
none	플로팅 박스 적용 없이 순서대로 배치

### ■ 콘텐츠 박스 크기 조정하기

- 콘텐츠 공간의 크기를 임의로 조정(가로,세로,최소 세로 및 가로, 최대 세로 및 가로)

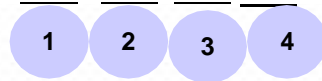
속성값	설명
width	가로 크기 지정
height	세로 크기 지정
min-width, min-height	최소 가로 및 최소 세로 크기 지정
max-width, max-height	최대 가로 및 최대 세로 크기 지정

## 05 다양한 효과 설정

### ■ 둥근 모서리

- 박스, 테두리의 모서리를 둥글게 설정

`border-radius : 5px 5px 5px 5px;`



속성	설명
<code>border-radius</code>	테두리의 모서리 둥글기 지정
<code>border-top-left-radius</code> <code>border-top-right-radius</code> <code>border-bottom-right-radius</code> <code>border-bottom-left-radius</code>	박스의 각 모서리에서 반경지정

### ■ 박스 그림자

- 박스에 그림자 효과를 설정
- `box-shadow : <hoffset> <voffset> <blur> <color> <inset/outset>`

`box-shadow : 3px 3px 3px #ff0000 inset/outset`  
가로위치 세로위치 번지는 정도 색상 Box안,밖 그림자 설정

## ■ 전환 효과(transition)

- 객체 모양의 변화는 객체의 CSS 속성 값의 변화

속성 구분	속성값	설명
property	all(기본값), none	전환효과(transition)가 적용될 속성 지정
duration	시간값(5s)	전환효과 지속 시간

## ■ transform

- Transform 속성을 이용하여 이동(translate), 크기 변화(scale), 회전(rotate), 기울기(skew)등의 효과를 줄 수 있다.

변환 함수	설명
translate(x,y) translateX(x),translateY(y)	x,y 좌표값으로 위치 이동 X방향,Y방향으로 이동
scale(x,y) scaleX(x),scaleY(y)	개체의 크기 변환 X방향,Y 방향으로 크기변환
rotate(각도) rotateX(x),rotateY(y)	회전각을 입력하여 회전 정면에 서 입체적으로 개체를 회전
skew() skewX(x),skewY(y)	개체카메라 앵글변환,기울임 변환 X방향,Y방향으로 기울임 변환

# Chapter 06

## 자바스크립트 기초



## ■ 자바스크립트란 ?

- 클라이언트, 웹프로그래밍 언어이며 스크립트언어(객체지향)이며, 웹페이지에서 존재하는 모든 동적인 부분을 표현 할 수 있는 언어이다. 스크립트언어이다 보니 인터프리터방식으로 실행이 되며, 프로그램개발이 좋고 단순한 프로그램들을 만들기 좋고, 반복적인 작업을 수행한다.

## • 자바스크립트와 자바 비교

자바스크립트	자바
웹브라우저에서 바로 자바스크립트 코드를 해석하고 바로 실행(인터프리터기반 언어)	자바 프로그램 컴파일 후 변환된 오브젝트 코드를 자바 가상 머신에서 실행하는 방식(컴파일 기반 언어)
객체기반 언어	객체지향언어
HTML파일내 포함되어 기술됨	HTML페이지내의 일정 공간에 로딩되어 별도수행
동적 바인딩(실행중 object reference를 체크)	정적 바인딩(컴파일중 object reference체크)
따로 변수의 타입이 없고 타입 검사도 느슨함	변수 선언이 필요하며 타입 검사가 엄격함

## 06 자바스크립트 기본 문법

### ▪ 웹문서 내장 방식

```
<script type="text/javascript"> ← 자바스크립트의 시작
<!--
```

자바스크립트 소스

← 자바스크립트 코드들이 위치하는 곳

```
//-->
</script>
```

← 자바스크립트의 종료

### ▪ 외부 파일 참조 방식

```
<script type="text/javascript" src="test.js"> ← src 속성값으로 파일의 경로 지정.
```

```
</script>
```

```
<script type="text/javascript" src="http://"> ← src속성값으로 자바스크립트 파일이 위치한 URL경로 지정
```

```
</script>
```

## ■ 기본 변수 타입

변수타입	값
Number	정수, 실수 등 숫자 값
String	작은 따옴표와 큰 따옴표 사이에 들어가는 문자들을 의미
Boolean	참(true)과 거짓(false)으로 표현해주는 연산자
Undefined	변수를 선언하고 아무런 값을 할당하지 않았을 때의 값
null	아무 값도 없는 의미.

## ■ 변수의 선언 `var 변수명`

`var 변수명 = 값`

## ■ 변수 선언 시 주의사항

- 변수는 반드시 알파벳 문자나 문자열로 시작, 한글 사용불가
- 변수는 영문자의 대·소문자 구분
- 자바스크립트의 키워드(예약어)는 변수로 사용 불가

abstract	boolean	break	byte	case	catch	switch
default	false	function	in	native	protected	this
super	throws	void	char	do	Double	throw
else	extends	final	finally	float	for	var
goto	if	implement	import	new	null	while
package	private	public	return	short	static	with

## ❖ 산술 연산자

- 연산자는 하나 또는 두 개의 값(변수)를 취해 연산을 수행하고 결과를 만들어낸다.
- 자바스크립트에서 사칙 연산을 수행할 때 사용되는 산술연산자이다.
- 더하기(+)는 문자열을 더하기 연산가능하지만 다른 연산자는 할 수 없다

+	더하기
-	빼기
*	곱하기
/	나누기
%	나머지

## ❖ 증감 연산자

## \* 전위 연산자

피연산자의 앞쪽에 위치한다. 더하거나 빼서 값을 보여준다

## \* 후위 연산자

피연산자의 뒤쪽에 위치한다. 일단 현재의 값을 보여준 다음 증가, 감소 시켜준다.

++a	기존의 변수의 값에 1을 더한다(전위)
--a	기존의 변수의 값에 1을 뺀다(전위)
a++	기존의 변수의 값에 1을 더한다(후위)
a--	기존의 변수의 값에 1을 뺀다(후위)

## ❖ 비교 연산자

- 두 피연산자를 비교하여 true 또는 false를 반환한다
- 숫자 뿐만 아니라 문자도 비교 가능하고, 두 데이터의 비교가 불가능 할 경우 항상 false가 반환된다.

>=	크거나 같다.
<=	작거나 같다.
>	~보다 크다
<	~보다 작다.
==	동치(Equal)연산자, 두 값이 동일한 경우 true를 리턴한다. 두 입력 값의 타입이 같지 않은 경우 형변환하여 비교한다
!=	NOT Equal연산자. 두 개의 입력값이 같지 않은 경우 true를 리턴한다, 타입이 다른 경우 형변환을 수행한다

## ■ 화면 출력

- 자바스크립트 내에서 HTML 문서는 Document라는 객체로 모델링 되어 있다.
- document객체는 write()라는 메소드를 사용하여 HTML 문서내 콘텐츠를 삽입할 수 있다.

## ■ 대화상자 화면 출력(alert())

- alert() 명령은 사용자에게 메시지나 경고사항을 나타내기 위해 사용
- alert()의 반환 값은 없다.

## ■ 키보드로 확인 입력 (confirm())

- confirm()명령은 사용자에게 선택할 수 있는 선택창이 나타남
- 선택을 하기 전까지 대기하며 확인 버튼을 누르면 true, 취소 버튼을 누르면 false를 리턴

## ■ 키보드로 문자열 입력(prompt())

- prompt() 명령을 이용하여 사용자로부터 키보드를 통해 문자열을 입력 받을 수 있다.
- 선택창과 입력창이 나타나며 확인 버튼을 누르면 문자열, 취소를 누르면 null값 리턴

- 조건문 : 주어진 조건에 따라 다른 결과가 실행되도록 해주는 형식

- ❖ If문

- 조건을 설정한 후 조건에 만족하면 그에 따른 명령문을 실행하게 하는 구문
- 조건에 만족하면 명령문이 실행되고, 조건에 만족하지 않으면 else구문의 명령문 실행
- 실행될 문장이 한 줄인 경우 {}를 생략할 수 있다.

```
if(조건)
{명령문1}
else
{명령문2}
```

- ❖ switch문

- 하나의 비교대상을 여러 개의 값과 비교
- C/C++/자바 언어와 달리 정수형 외 다른 타입도 사용 가능.

```
switch(변수)
{
 case value1: 문장1; break;
 case value2: 문장2; break;
 case value3: 문장3; break;
}
```



## ■ 반복문

- 계속해서 반복되는 작업을 간단하게 만들어 주는 실행문
- 입력되는 값을 조건과 비교하여 조건에 만족했을 때 실행문이 반복되고, 만족하지 않을 시 빠져 나오는 형식

### ❖ for문

- 변수의 초기 값을 설정한 후, 증감문을 증가/감소 시켜 초기 값이 조건식의 값이 될 때까지 문장을 반복하여 실행 해주는 반복문

### ❖ for(초기값; 조건식; 증감문) {조건식의 값이 true일 동안 반복할 실행문장}

- 조건문을 판별하여 조건식의 값이 true일 동안 반복해서 실행되는 문장
- 실행될 문장이 한 줄인 경우 {}를 생략할 수 있다.

### ❖ while(조건식) {조건식의 값이 true일 동안 반복될 실행문장}

- 문장을 실행한 후 다음 조건을 확인해 주는 반복문
- 실행될 문장이 한 줄인 경우 {}를 생략할 수 있다.

```
do
{조건식의 값이 true일 동안 반복할 실행문장}
while(조건식)
```

## ■ 함수

- 프로그램 내에서 특정 작업을 수행하기 위해 만들어진 명령어들의 모임
- 함수를 정의하기 위해서는 function 구문 사용
- 자바스크립트의 함수는 보통 HTML 파일의 head 태그영역 안에 작성, 함수를 호출하기 전 정의된 함수들에 대한 구분 분석이 먼저 이루어져야 하기 때문.

```
function 함수이름(매개변수...)
{함수가 호출되었을 때 실행될 문장}
```

## ■ 지역변수와 전역변수

- 함수 내에서 변수에 var를 붙이면 이 변수는 이 함수 안에서만 적용이 되고, 이 함수를 벗어나면 정의되지 않은 변수로 인식(**지역변수**)
- var를 붙이지 않은 변수는 전역변수라고 하며 문서 전체에 변수의 내용이 정의(**전역변수**)

```
function()
{
 var 변수; <-지역변수
 변수; <-전역변수
}
```

## ❖ 내장함수의 종류

- **alert()**  
-alert(문자열)의 형태로 호출, 확인 버튼이 있는 경고,알림 대화상자를 화면에 출력
- **confirm()**  
-confirm(문자열)의 형태로 사용되며 문자열을 대화상자에 출력,[확인]과 [취소] 버튼이 있는 선택창  
-[확인]버튼을 누르면 true리턴, [취소]버튼을 누르면 false리턴
- **prompt()**  
-입력 상자를 화면에 출력해서 사용자의 입력을 받을 목적으로 사용  
-입력받은 값을 반환하므로 [확인]버튼은 입력값을 리턴, [취소]버튼은 null값 리턴
- **eval()**  
-특정 문자열을 객체로 변환해서 리턴하는 함수  
-form에서 멤버에 접근하기 위한 목적으로 많이 사용
- **isNaN()**  
-주어진 데이터가 숫자인지를 판단하며 숫자이면 false, 숫자가 아니면 true리턴
- **escape()**  
-ASCII문자는 그대로 리턴하고 그 외에 문자는 유니코드 값을 리턴
- **parseFloat() parseInt()**  
-문자열을 정수와 실수로 변경해주는 함수

## ■ 내장객체

- 내장 객체는 자바스크립트에서 미리 정의되어 있는 내장된 객체를 말함
- 객체를 정의할 필요가 없고 new라는 예약어를 이용하여 객체를 생성해서 사용.
- 내장 객체의 종류로는 Date, String, Math, Screen, Array등이 있다.
- Date 객체
  - Date객체는 컴퓨터에서 제공되는 날짜와 시간을 알아내거나 설정하기 위한 객체.

기본형식  
객체 이름 =new Date()

메소드	기능 설명
getFullYear(), getMonth(),getDate(),getDay(), getHours(), getMinutes(), getSeconds()	컴퓨터의 시계가 제공하는 현재 시간을 구하는 메소드(연도,월,일,요일,시간,분,초값 리턴)
setFullYear(), setMonth(), setDate(), setDay(), setHours(), setMinutes(), setSeconds()	컴퓨터의 시계를 설정하기 위한 메소드(연도, 월, 일, 요일, 시간, 초값 설정)
getTime()	1970년 1월 1일 이후 현재까지의 시간을 천분의 1초 단위 알아낸다.

## ■ Math 객체

- 수학적 계산이나 난수 등을 얻기 위해 자바스크립트에서 기본적으로 제공, 별도의 선언이나 생성 없이 사용
- Math 속성

속성	설명
E	자연 로그 밑인 e에 해당하는 상수 값
LN2	2의 자연로그 값
LN10	10의 자연로그 값
LOG2E	2를 밑으로 하는 e의 로그, LN2의 역수
LOG10E	10을 밑으로 하는 e의 로그, LN10의 역수
PI	원주율(3.14...)
SQRT1_2	½의 루트 값
SQRT2	2의 루트 값

## • Math 메소드

메소드	기능 설명
cos(), sin(), tan()	삼각함수 코사인, 사인, 탄젠트 함수 제공
acos(), asin(), atan()	코사인, 사인, 탄젠트 역함수 제공
ceil(), floor(), round()	올림, 내림, 반올림 값 리턴
max(), min(), abs()	입력 인자 값들 중 최대, 최소, 절대값 리턴
pow(), sqrt()	X의 제곱 값 반환, x의 루트 값 반환
log(), exp()	X의 로그반환, 밑이 E인 자연 지수에서 E의 x승의 값 반환

## ▪ Array 객체

- 배열을 위한 데이터를 따로 제공하지 않고 Array라는 내장객체를 이용하여 배열 생성
- Array 객체는 배열의 크기를 알려주는 length 속성이 있다.

형식

```
배열명 = new Array() //배열의 크기를 지정하지 않는 방법
배열명 = new Array(1) //배열의 크기를 미리 지정하는 방법
배열명 = new Array(배열1, 배열2) //배열요소를 모두 열거해 주는 방법
```

## ❖ Array 메소드

join	배열의 값을 묶어 하나의 문자열로 만드는 메소드. 매개 변수에 의해 주어지는 문자 열이 분리자의 역할을 하며 분리자 생략 시 ","가 자동으로 사용
reverse()	배열의 순서를 역순으로 바꾸어 주는 메소드
sort()	비교 함수를 지정 하면 비교함수에서 지정한 대로 정렬이 되고, 지정하지 않을 시 알파벳 순으로 정렬하는 메소드.
slice()	배열의 일부를 선택하여 새로운 배열을 생성하여 주는 메소드
concat()	배열의 뒤에 요소를 붙혀 배열의 내용을 추가하는 메소드

# Chapter 07

## DOM과 자바스크립트

## 07 DOM 정의 및 문서 계층 구조

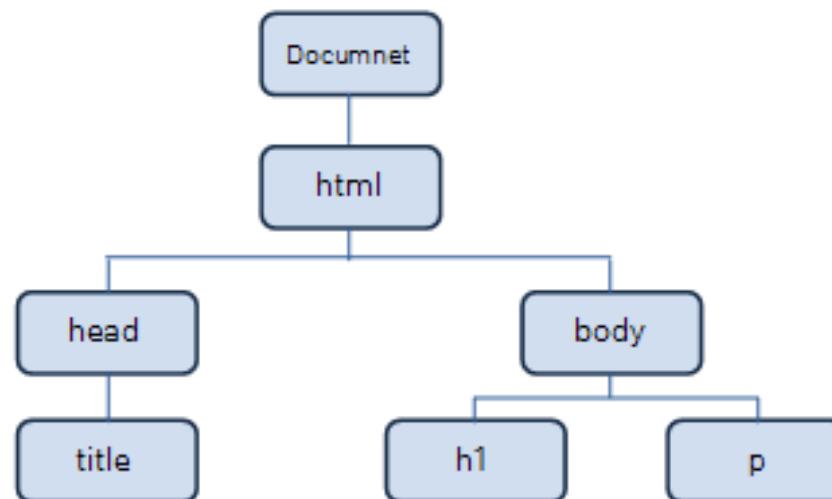
### ■ DOM이란?

- DOM(Document Object Model)은 웹 문서를 자바스크립트 입장에서 구조적 객체의 형태로 바라보고 처리하는 모델.
- 트리 형태의 구조를 가진 텍스트 문서와 각종 구성요소를 일정한 규칙에 의해 각각의 객체로 만들고 요소간의 상관관계 정보를 차가하여 문서 전체를 모형화 시키는 규격.
- 문서의 모든 구성원이 각각 인식되며 이러한 객체를 Node라고 한다.

### ■ 문서의 계층구조

- 트리 구조 DOM

```
<!DOCTYPE html>
<html>
<head>
 <meta charset="utf-8">
 <title>domtree</title>
</head>
<body>
 <h1>test</h1>
 <p>test</p>
</body>
</html>
```





## ■ 문서 객체 생성

- 문서 객체는 기본적으로 텍스트 노드를 갖는 문서와 텍스트 노드를 갖지 않는 문서 객체로 구분

### 1. 텍스트 노드를 갖는 문서 생성

```
<script type="text/javascript">
window.onload =function(){
var textNode = document.createTextNode('DOM');//텍스트 노드 생성
</script>
```

### 2. 텍스트 노드를 갖지 않는 문서 생성

```
<script type="text/javascript">
window.onload =function(){
var textNode = document.createElement('img');//요소 노드 생성
</script>
```

### 3. 화면에 문서 객체 출력

```
<script type="text/javascript">
window.onload =function(){
var textNode = document.createTextNode('DOM');//텍스트 노드 생성
document.body.appendChild(textNode);//객체 노드를 연결
</script>
```

- 웹 페이지에 이미 존재하는 HTML 태그를 자바 스크립트로 가져오는 방법  
`getElementById(id)` = 태그의 id와 일치하는 문서 객체를 가져온다.
- 메소드

메소드	설명
<code>getElementById("id")</code>	태그의 id와 일치하는 문서 객체를 가져온다.
<code>getElementByName("name")</code>	태그의 name과 일치하는 문서 객체를 배열로 가져온다.
<code>getElementsByTagName("tagname")</code>	Tagname과 일치하는 문서 객체를 배열로 가져온다.

#### 사용방법

```
<script type="text/javascript">
window.onload = function()
{
var value = document.getElementById("value");
};
```

- 문서 객체의 style 속성 사용.

#### 사용방법

```
<script type="text/javascript">
window.onload = function()
{
var stylechange = document.getElementById("stylechange");
stylechange.style.border="2px Solid Black";
stylechange.style.color="green";
stylechange.style.fontFamily="Verdana";
};
```

## ■ 노드(Node)

- DOM의 기본 원자는 Node이며, Node는 HTML문서의 트리를 구성하는 단위이다.
- Document, Element, text, 주석 까지도 모두 Node로부터 파생된다.

### ❖ 노드 타입

타입	설명
Element	요소 노드이며, 태그라고 생각하면 된다.
Attr	태그의 속성이다.
Text	태그에 포함된 문자열이다.
Comment	주석이다
Document	전체 문서이다.
DocumentType	문서의 종류이다.

- nodeName
  - 노드의 이름이며 엘리먼트 노드나 속성노드는 태그나 속성의 이름과 같아 직관적이지만 텍스트 노드나 주석 노드는 별도의 이름이 없으므로 #text, #comment식으로 상수 문자열이 리턴 된다.
- nodeValue
  - 노드의 실제 값, 속성노드, 텍스트 노드, 주석노드만 실제 값이 있으며 엘리먼트 노드나 문서 노드는 실제 값이 없으므로 null이 리턴 된다.

## ■ 노드 속성

속성	설명
firstChild	첫 번째 자식 노드를 찾는다.
lastChild	마지막 자식 노드를 찾는다.
parentNode	부모 노드를 구한다.
previousSibling	이 전 형제 노드를 구한다.
nextSibling	다음 형제 노드를 구한다.
ownerDocument	노드가 속한 문서의 루트 노드를 구한다.
childNodes	모든 자식 노드를 구한다

## ■ 노드 메소드

메소드	설명
appendChild()	자식노드를 마지막에 추가한다.
insertBefore()	자식노드를 중간에 삽입한다.
removeChild()	자식노드를 제거한다.
replaceChild()	자식노드를 교체한다.
cloneNode()	노드를 복사한다.
hasChildNodes	자식노드가 있는지 조사한다.

## Window 객체

- 브라우저 기반 자바스크립트의 최상위 객체.
- Var 키워드로 선언한 일반 변수도 모두 window 객체의 속성.

### ❖ window 메소드

메소드	설명
open(UIURL,name,features,replace)	윈도우 창을 연다.
close()	현재 윈도우를 닫는다.
focus()	현재 윈도우에 초점을 준다.
blur()	현재 윈도우에 초점을 제거한다.
print()	윈도우의 내용을 인쇄하거나 인쇄 대화상자가 나타난다.

## Navigator 객체

- 현재 사용 하고 있는 웹 브라우저의 종류와 버전 등 관련 정보 알아낼 때 사용.

속성	설명
appName	브라우저의 코드명
appVersion	브라우저의 이름
platform	브라우저의 버전
userAgent	사용중인 운영체제의 시스템 환경
	브라우저의 전체적인 정보

### ■ 이벤트(Event)

- 사용자가 웹 브라우저를 사용하는 중 마우스, 키보드 입력 등을 하게 되는데 이를 이벤트라 한다
- 이벤트 처리를 하기 위해서는 이벤트정의, 이벤트 핸들러 작성, 이벤트 등록을 알아야 한다.

```
<script tpye="text/javascript">
function test(){}
</script>
<input tye="button" onClick="test();" />
```

← 이벤트 핸들러 함수

↑  
이벤트 타입

↑  
이벤트 등록

## 07 이벤트 (II)

- 이벤트의 종류
  - 마우스 이벤트

이벤트 속성	설명
onclick	HTML문서 내의 요소를 마우스로 클릭 했을 때 발생.
ondblclick	HTML문서 내의 요소를 마우스로 더블클릭 했을 때 발생.
onmousedown	HTML문서 내의 요소 위에서 누를 때 발생.
onmousemove	HTML문서 내의 요소 위에서 이동 시킬 때 발생
onmouseover	마우스 커서가 요소 위에 위치할 경우 발생하며 1회만 발생.
onmouseout	마우스 커서가 해당 요소 위를 벗어날 경우 발생.
onmouseup	HTML문서 내의 요소 위에서 버튼을 떼는 경우 발생 (onmousedown과 반대)

- 키보드 이벤트

이벤트 속성	설명
Onkeydown	키보드를 누르고 있는 경우 발생.
Onkeypress	키보드를 누른 후 떼 때 발생.(마우스클릭과 유사)
Onkeyup	키보드를 누른 후 떼 때 발생.(mouseup과 유사)

## 07 이벤트 (III)

### ■ 프레임 이벤트

이벤트 속성	설명
onabort	이미지가 완전히 로드 되기 전 정지 되었을 때 발생.
onload	문서,객체,프레임 등 로드가 완료 되었을 때 발생.
onunload	윈도우 창, 프레임으로부터 문서가 제거되었을 경우 발생.
onresize	웹 브라우저 창의 크기를 변경할 때 마다 이벤트 발생.
onscroll	웹 브라우저 화면을 스크롤 할 때 발생.
onerror	이미지를 로드 할 때 정상적으로 이루어지지 않을 경우 발생.

이벤트 속성	설명
submit	폼 이 실행될 때 발생.
reset	폼 이 리셋 될 때 발생.
focus	요소에 초점 이 되었을 때 발생.
blur	요소에 초점 이 없어 질 때 발생(focus와 반대)
change	폼 요소 콘텐츠의 내용이 변경되었을 때 발생.



### ■ 고전 이벤트 모델

- 고전 이벤트 모델은 예전부터 사용해 왔던 전통적인 방법
- 모든 웹 브라우저가 지원하므로 호환성이 좋다.

```
<body>
<h3 id="header">고전 이벤트 모델 클릭</h3>
<script>
 var header=document.getElementById("header");
 header.onclick=function(){
 alert("고전 이벤트 모델을 클릭했습니다.");
 }
</script>
</body>
```

### ■ 인라인 이벤트 모델

- 인라인 이벤트 모델은 간편하지만, 코드가 섞이게 되어 구조적으로 바람직하지 못하다.

```
<body>
<h1 onclick="alert('클릭했습니다.');">클릭하세요</h1>
</body>
```

- ❖ 아래와 같은 방법으로 함수로 분리 하는 것이 구조상 더 좋다.

```
<body>
<h3 id="header">클릭하세요</h3>
<script>
function test_click()
{alert("클릭했습니다."); header.onclick=null;}
</script>
</body>
```