

인문사회 영역과 IT 기술융합

기초 코딩 과정

목차

- HTML/CSS 기본 사용법
- HTML/CSS 응용 – 웹 화면 구성
- 자바스크립트 기본 – 동적화면 구성

HTML/CSS 기본 사용법

HTML/CSS 기본 사용법

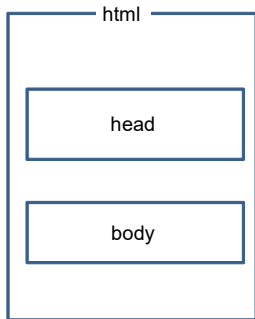
유용한 HTML5 사이트

- <http://html5test.com>
- <http://www.robertnyman.com/html5/>
- <http://html5demos.com/>
- <http://html5gallery.com/>
- <http://www.w3schools.com/>
- <http://freehtml5templates.com/>
- <http://dev.w3.org/html5/html-author/>
- <http://channy.creation.net/project/html5/html4-differences/>

html 문서의 기본 구조

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>기초코딩시작하기</title>
  </head>

  <body>
    <h1>HTML 문법과 기본 사용법</h1>
    <p>HTML 태그 사용법</p>
    <p>이미지 그리기</p>
    <p>링크 입력</p>
  </body>
</html>
```



<!DOCTYPE html> 현재 문서가 HTML5 로 작성한 웹문서라는 뜻

<html> ~ </html> 웹 문서의 시작과 끝

<head> ~ </head> 웹브라우저가 웹문서를 해석하는데 필요한 정보를 입력

<body> ~ </body> 웹브라우저 화면에 나타나는 내용

주요 태그 사용법

- <h> 태그

<h1>, <h2>, ... <h6> h1태그는 가장 큰 글씨로 h옆의 숫자가 커질수록 작은 글씨가 출력됨

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>기초코딩시작하기</title>
  </head>
  <body>
    <h1>HTML 문법과 기본 사용법</h1>
    <h3>HTML 문법과 기본 사용법</h3>
    <h5>HTML 문법과 기본 사용법</h5>
    <p>HTML 태그 사용법</p>
    <p>이미지 그리기</p>
    <p>링크 입력</p>
  </body>
</html>
```

HTML 문법과 기본 사용법

HTML 문법과 기본 사용법

HTML 문법과 기본 사용법

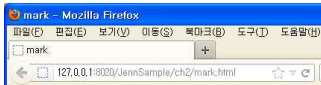
HTML 태그 사용법

주요 태그 사용법

태그	설명
<p>	텍스트 단락을 만드는 태그
 	줄바꿈
<i>	다른 텍스트와 구별되는 텍스트를 표현, 기울임체
<mark>	중요한 텍스트 마킹
	진하게 표시, 중요하지는 않지만 진하게 표시
	텍스트에서 강조하는 부분의 글자, 중요한 글자 진하게
<strike>	취소선 텍스트

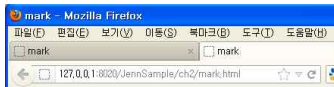
<mark>

```
<h1>HTML5</h1>
<p>
HTML5의
<mark>시만택태그</mark>
와
<mark>새로 추가된 기능들</mark>
을 살펴 봅시다
</p>
```



HTML5

HTML5의 시만택태그와 새로 추가된 기능들을 살펴 봅시다



HTML5

HTML5의 시만택태그와 새로 추가된 기능들을 살펴 봅시다

	설명	사용예
<div>	body 문서 안에서 각 영역의 세션 구분 정의	<pre><div> <h3>제목</h3> <p>단락 설명...</p> </div></pre>
<a>	링크 연결	구글로 이동
	img 태그	
	순서 없는 목록 만들기	<pre> 커피 우유 음료수 </pre>
	순서 있는 목록 만들기	<pre> 커피 우유 음료수 </pre>

실습

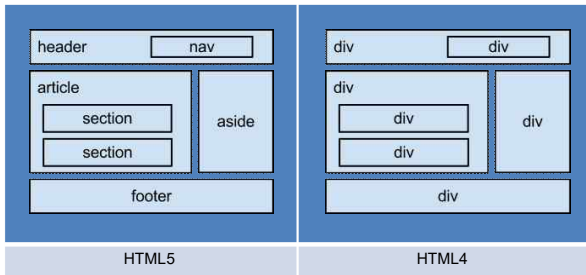
- 보라카이

11~5월은 건기로 야외 활동하기 좋지만 5월은 낮기온이 높아 매우 덥다. 건기 중에서도 1~4월의 강수량 가장 적다. 기온은 12~2월까지 상대적으로 낮은 편이며 3월부터 낮기온이 올라 5월에 가장 덥다. 우기에는 한낮에 1~2시간가량 스콜성 폭우가 쏟아지거나 밤새 비가 내린다.

시맨틱 태그

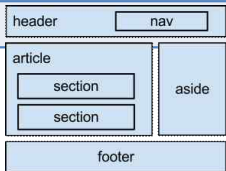
시맨틱 태그

- 태그만으로 문서를 이해할 수 있도록 의미를 갖는 태그



시맨틱태그

<header>	사이트소개, 메인메뉴, 사이트로그 등 머리말
<hgroup>	제목과 관련된 부제목을 묶어준다
<nav>	사이트의 메뉴나 링크와 같은 내비게이션 요소 <header>, <footer>, <aside> 태그 안 혹은 독립적으로 사용
<article>	블로그의 포스트, 웹사이트의 내용 등 독립적인 웹 콘텐츠 항목 다른 곳으로 배포되거나 재사용이 가능 문서 내용이 많다면 여러 개의 article 요소로 문서들 분류가능
<section>	컨텐츠를 주제로 그룹을 나눠 제목을 붙여놓은 것 사이트 소개, 뉴스, 연락처 정보 섹션 등
<aside>	본문 이외의 부가적인 내용
<footer>	작성자나 저작권 정보, 관련 문서 링크 등의 정보



스타일과 섹션 태그

- <header>
- <hgroup>
- <nav>
- <article>
- <section>
- <aside>
- <footer>

이미지 관련 태그

- <canvas>
- <figure>
- <figcaption>

실습 <figure>

```
<figure>  
    
  <figcaption>아침 일찍 일어나 바라본 바닷가 배를탈 준비를 하는  
사람들</figcaption>  
</figure>
```



<address> 태그

- 웹페이지 제작자의 연락처 또는 피드백을 위한 이메일 주소, 주소, 전화번호와 같은 정보들이 포함된다

```
<address>
  <a href="http://www.cebutravel.co.kr"
target="_blank">jennifer</a> ,
  <a href="jenndev@gmail.com">jenndev@gmail.com</a>
</address>
```

Firefox * address



정은이와 함께 머물던 세부에서의 15일
너무도 맑고 파란 바다와 하늘
모든 근심이 사라진 행복한 시간들

[jennifer , jenndev@gmail.com](mailto:jennifer_jenndev@gmail.com)

오디오/비디오 태그

- <audio>
- <source>
- <track>
- <video>

시맨틱태그를 활용한 예제

- Preexer/tagtest.html 을 적절한 코드를 넣어 그림과 같이 만든다

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
```

```
<title><!-- 1번 --></title>
```

```
<!-- 2번 -->
```

```
</head>
```

```
<body>
```

```
<header>
```

```
<h1><!-- 3번 --></h1>
```

```
</header>
```

```
<section>
```

```
<h3>보라카이 섬</h3>
```

```
<!-- 4번 -->
```

```
<b>대표적 명소 화이트비치</b>
```

```
<p>
```

```
<!-- 5번 -->
```

```
</p>
```

```
</section>
```

```
<section>
```

```
<!-- 6번 -->
```

```
</section>
```

```
<footer>
```

```
<!-- 7번 -->
```

```
</footer>
```

```
</body>
```

```
</html>
```

보라카이 섬

세계적인 휴양지로 손꼽히는 보라카이는 필리핀의 중서부 파나이 섬(Panay province) 북서쪽에 떠 있는 섬으로 마지막 남은 천국이라 불릴 만큼 대동지 않은 자연을 지닌 휴양지이다. 이곳에는 길이 7km에 달하는 **검고 넓은 화이트 비치**와 야자수 숲이 어우러진 82개의 크고 작은 독특한 매력을 지닌 비치와 있다. 보라카이에서는 자연 경관과 조화를 이룬 건축물을 짓기 위해 코코넛 나무 크기 이상의 건물을 지을 수 없으며 파도가 밀려오는 지점에서 300m 이내에도 건물을 지을 수 없다.

대표적 명소 화이트비치

디니워드 해변 *Diniwid*
발링하이 해변 *Balinghai*
푸타붕가 해변 *Puta Bunga*
푸카셸 해변 *Puka Shell*
이릴 이리간 해변 *Iril-Irgan*



아침 일찍 일어나서 떠난 온 피맛과 비를 맞으며 걷는 사람들

작성일자: 2012년 3월12일

jeeyeon@maik.com

• 2번

```
<style type="text/css">
body{
background-color: #f0f8ff
}
P {
font-family: "바탕";
font-size: 20px;
}
figcaption{
font: 15px 바탕;
color: #00f;
padding: 5px
}

mark {
background-color: #ffd700;
font-weight: bold;
}
```

4번

<p><mark>세계적인 휴양지</mark>로 손꼽히는 보라카이는 <mark>필리핀</mark>의 중서부 파나이 섬(Panay province) 북서쪽에 떠 있는 섬으로 마지막 남은 천국이라 불릴 만큼 때문지 않은 자연을 지닌 휴양지이다. 이곳에는 길이 <mark>7km</mark>에 달하는 길고 넓은 화이트 비치</mark>와 야자수 숲이 어우러진 32개의 크고 작은 독특한 매력을 지닌 비치가 있다. 보라카이에서는 자연 경관과 조화를 이룬 건축물을 짓기 위해 코코넛 나무 크기 이상의 건물을 지을 수 없으며 파도가 밀려오는 지점에서 300m 이내에도 건물을 지을 수 없다.</p>

• 7번

```
<p>작성일자: <time datetime="2012-03-12" pubdate>2012년 3월12일</time></p>
<address>jeeyeon@mail.com</address>
```

비디오 태그 <video>

- 어트리뷰트

속성	설명
src	동영상파일 경로
poster	동영상을 로드할 수 없을 때 대체할 그림 지정
preload	백그라운드에서 동영상을 다운로드하고 재생단추를 누르면 재생되게 함
autoplay	자동재생
loop	반복 재생 횟수
controls	동영상 화면에 나타나는 컨트롤 막대 재생/일시정지, 트랙바, 볼륨조절
width	동영상 화면 너비
height	동영상 화면 높이
muted	비디오의 오디오 출력의 음소거 여부를 지정

오디오 태그<audio>

- 어트리뷰트

속성	설명
src	오디오 파일 경로
autoplay	자동재생
loop	반복 재생 횟수
controls	동영상 화면에 나타나는 컨트롤 막대 재생/일시정지, 트랙바, 볼륨조절
preload	오디오가 재생되기 전 파일을 미리 다운로드 할지 지정

멀티미디어 태그 속성

속성	설명
currentTime	현재 재생 위치(초단위)
duration	전체 재생 시간 (초단위)
paused	일시정지 여부
ended	재생 종료 여부
muted	음소거 여부
volume	볼륨값 (0.0 ~ 1.0)
error	에러 정보
networkState	네트워크 상태 (접속전, 로딩 중, 완료, 실패 등)

멀티미디어 함수와 이벤트

- 주요 함수

Load()	멀티미디어 로드
Play()	재생
Pause()	일시정지

실습 – 간단한 비디오 재생

- <video> 태그로 멀티미디어 파일을 재생한다



오디오 태그

- preexer/audiosample.html

```
<!DOCTYPE html >
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
<title>오디오 태그 샘플</title>
</head>
<body>
  
  <br><br>
  // 아래 태그를 입력하세요
</body>
</html>
```

```
<audio controls autoplay>
  <source src="media/008.ogg">
  <source src="media/008.mp3">
</audio>
```

비디오 태그

- preexer/videosample.html

```
<!DOCTYPE html >
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=EUC-KR">
<title>비디오 샘플</title>
</head>
<body>
//아래 태그를 입력하세요
</body>
</html>
```

```
<video width="480" height="320" autoplay
controls >
  <source src="media/transformers480.ogg">
</video>
```

실습 – 간단한 비디오 플레이어

- 재생과 일시 정지 버튼, 그리고 볼륨 슬라이드 바를 갖는 간단한 플레이어를 제작해 본다



심플 플레이어

- preexer/simpleplayer.html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>심플 플레이어</title>
```

```
</head>
```

```
<body>
```

```
<!-- 1번 -->
```

```
</body>
```

```
</html>
```

```
<!-- 2번 -->
```



재생

일시정지

음소거

볼륨:
140/151(초)

심플 플레이어

- 1번

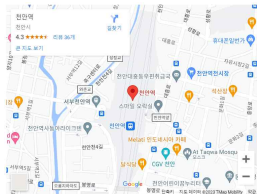
```
<video id="myvideo" controls  
      src="media/transformers480.ogg"></video>  
<br>  
<button onclick="play()">재생</button>  
<button onclick="pause()">일시정지</button>  
<button onclick="mute()">음소거</button>  
<br>  
볼륨:  
  <input id="volumecontrol" type="range" max="1"  
        step="any"  
        onchange="updateVolume()">  
<div id="time"></div>
```

- 2번

```
<script type="text/javascript">  
  var myvideo = document.getElementById("myvideo");  
  
  function play(){  
    myvideo.play();  
    myvideo.volume = volumecontrol.value;  
  }  
  function pause(){  
    myvideo.pause();  
  }  
  function mute(){  
    myvideo.muted = !myvideo.muted;  
  }  
  function updateVolume() {  
    myvideo.volume = volumecontrol.value;  
  }  
  
  myvideo.addEventListener("timeupdate",  
    function(){  
      document.getElementById("time").innerHTML =  
        Math.floor(myvideo.currentTime) + "/" +  
        Math.floor(myvideo.duration) + "(초)";  
    }  
    , false  
  );  
</script>
```

Iframe



- 현재 브라우저에 렌더링되고 있는 문서 안에 또 다른 HTML 페이지를 삽입할 수 있도록 하는 기능 제공 (Youtube, GoogleMap)



Table

	설명	사용예						
<table>	태그는 HTML 문서에서 표를 만드는 태그, 행과 열을 표현하기 위해 <tr>, <td>등의 태그와 함께 작성	<div><pre><table> <tr> <td>섹션1</td> <td>섹션2</td> </tr> <tr> <td>섹션3</td> <td>섹션4</td> </tr> <tr> <td>섹션4</td> <td>섹션5</td> </tr> </table></pre></div> <table><tr><td>섹션1</td><td>섹션2</td></tr><tr><td>섹션3</td><td>섹션4</td></tr><tr><td>섹션4</td><td>섹션5</td></tr></table>	섹션1	섹션2	섹션3	섹션4	섹션4	섹션5
섹션1	섹션2							
섹션3	섹션4							
섹션4	섹션5							
<tr>	표의 행을 나타냄							
<th>	표의 열을 나타내며(제목)							
<td>	표의 열을 나타냄							

Table

	설명	사용예
colspan	열을 확장합니다. (좌우로)	<pre><table> <tr> <td colspan="2">2x1 셀</td> </tr> <tr> <td>1x1 셀</td> <td>1x2 셀</td> </tr> </table></pre> 
rowspan	행을 확장합니다. (상하로)	<pre><table> <tr> <td rowspan="2">1x2 셀</td> <td>1x1 셀</td> </tr> <tr> <td>1x1 셀</td> </tr> </table></pre> 

Table

영화	요일	상영시간	관람연령제한	잔여좌석이벤트	
분노의 질주	월요일	2시간	15세이상관람가	15석	○
엑시트			12세이상관람가	매진	
우리집	수요일	1시간 50분			143석
봉오동전투	금요일		73석		
광대들	토요일	1시간 58분	15세이상관람가	98석	○
만나	일요일	2시간 01분		48석	X

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, height=device-height">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
</head>
<body>

  <table border="1" width="900px" height="500px">
    <tr bgcolor="blue" span style="color:white" align="center">
      <td>영화</td>
      <td>요일</td>
      <td>상영시간</td>
      <td>관람연령제한</td>
      <td>잔여좌석이벤트</td>
    </tr>
  </table>
</body>
</html>

```

세미나 신청 폼 2

```
<tr align="center">
  <td>분노의 질주</td>
  <td>월요일</td>
  <td>2시간</td>
  <td bgcolor="green" span
style="color:white">15세이상관람가</td>
  <td>15석</td>
  <td bgcolor="purple" span
style="color:white">0</td>
</tr>
<tr align="center">
  <td>엑시트</td>
  <td bgcolor="orange" span
style="color:white">12세이상관람가</td>
  <td>매진</td>
</tr>
<tr align="center">
  <td>우리집</td>
  <td>수요일</td>
  <td>1시간 50분</td>
  <td>143석</td>
  <td bgcolor="grey" span
style="color:white">X</td>
</tr>
```

```
<tr align="center">
  <td>봉오동전투</td>
  <td>금요일</td>
  <td bgcolor="red" span
style="color:white">15세이상관람가</td>
  <td>73석</td>
</tr>
<tr align="center">
  <td>광대들</td>
  <td>토요일</td>
  <td>1시간 58분</td>
  <td>98석</td>
  <td bgcolor="purple" span
style="color:white">0</td>
</tr>
<tr align="center">
  <td>안나</td>
  <td>일요일</td>
  <td>2시간 01분</td>
  <td>48석</td>
  <td bgcolor="grey" span
style="color:white">X</td>
```

CSS(Cascading Style Sheet)

- CSS는 HTML을 보기 좋게 디자인하는 역할을 하는 언어
- CSS는 문서의 콘텐츠와 레이아웃, 글꼴 및 시각적 요소들로 표현되는 문서의 디자인을 분리하기 위한 목적으로 만들어짐
- 하나의 규칙으로 여러 HTML 요소와 HTML 문서를 제어 가능

CSS 상속

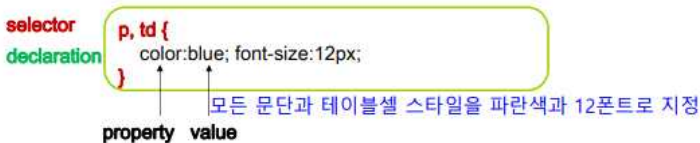
- 부모 요소로부터 상속

```
<p style="color:blue">Parent /*부모요소 파란색 스타일*/  
  <em style="font-size:12px">Child</em> /* 자식요소 파란색 12폰트 스타일*/  
</p>
```

- CSS cascading
- 이 4가지 스타일 시트가 태그에 동시에 적용될 때 스타일이 합쳐져서 적용되는 것
 1. 브라우저 디폴트
 2. .css 스타일 시트 파일
 3. <style></style> 태그
 4. style 속성에 작성된 스타일
- CSS overriding
- 동일한 CSS3 프로퍼티에 서로 다른 값 설정이 충돌 시 우선순위가 높은 스타일을 적용하는 규칙임

CSS 기본 문법

- CSS 규칙은 선택자(selector)와 선언부(declaration)로 나뉨
- 선언부는 다시 속성(property)과 속성값(value)으로 나뉨



- 선택자는 HTML tag나 부여한 class 혹은 ID가 위치
- 여러 선택자를 사용할 경우 콤마(,)로 구분
- 여러 개의 속성과 속성값이 있을때 세미콜론(;)으로 구분
- 각각의 선언은 속성과 속성값을 콜론(:)으로 구분
- CSS 주석은 `/* */`로 둘러 씌

CSS 적용 방법

- HTML 태그 내에 스타일 지정 (Inline Styles)

```
<p style="color:gray"> 이 문단의 색은 회색으로 </p>
```

- 내부 스타일 시트 (Internal Style Sheet)

```
<head>  
  <style type="text/css">  
    body { font-size:9pt;}  
  </style>  
</head>
```

- link를 이용한 외부 스타일 시트 (External Style Sheet)

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


CSS 적용 방법

- @import를 이용한 외부 스타일 시트

```
<head>  
  @import url("/css/style.css");  
</head>
```

- xml-stylesheet 처리 명령으로 외부 스타일 시트

```
<?xml-stylesheet type="text/css" href="/css/style.css" ?>
```

CSS 선택자

- tag name selector 태그 이름이 선택자로 사용

```
p, td { color:blue; font-size:12px; }
```

- class selector 선택자 이름 앞에 '.' 기호를 붙인 경우 class 속성이 같은 모든 태그에 적용(여러개 사용O)

```
.select { color: red }  
<li class="select">Hello</li>
```

- id selector 선택자 이름 앞에 '#' 기호를 붙인 경우 태그의 id 속성으로만 지정 가능(여러개 사용 X - 고유성을 가짐)

```
#content { width: 600px; float: left; background: green;}  
<div id="content">  
<h1>Content</h1>  
</div>
```

제
언

<input> 타입

<input type="text">	텍스트 입력
<input type="password">	비밀번호 입력, 화면에 • 로 표시된다
<input type="checkbox">	클릭으로 선택 해제
<input type="radio">	그룹 중 다중 선택
<input type="file">	파일 업로드
<input type="submit">	폼에 입력값 전송

HTML5 새로운 <input> 타입

<input type="email">	이메일 주소
<input type="url">	웹 사이트 주소
<input type="number">	▲/▼를 눌러서 숫자를 선택하는 스피너박스 min, max, step(간격, 기본값 1), value(초기값) 속성지정 가능
<input type="range">	슬라이드 막대로 지정된 범위내의 숫자 선택
<input type="date">	연도, 월, 날짜 선택
<input type="month">	연도와 월 선택
<input type="week">	연도와 주 선택
<input type="time">	시간 선택
<input type="datetime">	날짜와 시간을 선택 UTC 시간대 기준
<input type="datetime-local">	날짜와 시간 선택
<input type="search">	검색상자
<input type="color">	색상 선택

HTML5 새로운 속성

list	텍스트 필드 입력 시 입력 가능한 데이터 목록을 보여줌. <datalist> 로 목록을 정의하고 list속성에 datalist의 아이디를 지정 <pre><input id="pcolor" type="text" list="mycolor"> <datalist id="mycolor"> <option value="black" label="검정색"> <option value="white" label="흰색"> <option value="skyblue" label="하늘색"> </datalist></pre>
max, min, step	최대, 최소, 간격
maxLength	텍스트 필드에 입력할 수 있는 최대 입력 문자 개수
required	서버로 폼을 전송 시 필수 입력 필드를 지정
placeholder	input태그의 필드 안에 힌트 내용을 표시
autocomplete	웹브라우저가 이전에 사용자가 입력한 내용을 기억했다가 힌트로 제공
autofocus	페이지 불러오자마자 입력 필드 안에 마우스 커서 표시 firefox 3.6.3 지원하지 않음

버튼으로 폼 처리

- 폼 처리 프로그램
 - `<form id="storm" action="http://search.naver.com/search.naver" method="get">`
- Html5에서 submit 타입 버튼을 이용한 폼 처리

formaction	form태그의 action 속성을 덮어 쓴다. 폼을 처리할 프로그램의 url 지정
formmethod	form태그의 method 속성을 덮어쓴다. action에서 지정한 url로 폼을 전송하는 방법을 지정. get, post, put, delete로 지정
formenctype	폼 콘텐츠를 인코딩할 mime 타입을 지정 application/x-www-form-urlencoded, multipart/form-data, text/plain 지정
formnovalidate	폼 전송 시 필드 값 검증 여부 지정 true 면 검증 없이 전송
formtarget	폼의 타겟 속성을 덮어쓴다. _blank, _self, _parent, _top

실습 - 세미나 신청 폼

- Semina_1/register.html

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>세미나신청폼</title>
</head>
<body>
<header>
<!-- 1번 -->
</header>
<form>
<fieldset>
<!-- 2번 -->
</fieldset>
<fieldset>
<!-- 3번 -->
</fieldset>
<fieldset>
<input type="submit" formmethod="post" action="/buy" value="신청하기">
<input type="submit" formmethod="post" action="/back" formnovalidate value="취소하기">
</fieldset>
</form>
</body>
</html></html>
```

신청 상세 정보 선택	
• 세미나	<input type="text"/>
신청하는 사람	
• 이름	<input type="text" value="홍길동"/>
• 신청인원	<input type="text" value="1"/>
<input type="button" value="신청하기"/>	<input type="button" value="취소하기"/>

- 1번

```
<legend>신청 상세 정보 선택</legend>
<li>
  <label for="semina">세미나</label>
  <input id="setobj" type="text" list="object">
  <datalist id="object">
    <option value="webApp" label="웹애플리케이션">
    <option value="HTML5" label="HTML5">
    <option value="android" label="안드로이드 어플리케이션 개발">
  </datalist>
</li>
```

- 2번

```
<legend>신청하는 사람</legend>
<li>
  <label for="name">이름</label>
  <input id="name" type="text" placeholder="홍길동" required autofocus>
</li>
```

- 3번

```
<li>
  <label for="order">신청인원</label>
  <input id="num" type="number" min="1" max="3" value="1">
</li>
```

실습 - 세미나 신청 폼2

2023년 세미나 신청

교육기관 : (주)모블에듀

신청마감 : 2023년 11월 17일 오후 6시

신청 상세 정보 선택

- 세미나
- 일정선택 연도-월-일
- 과정선택

신청하는 사람

- 이름
- 메일주소
- 전화번호

- 주소

- 신청인원

신청하기

취소하기

세미나 신청 폼 2

```
<!DOCTYPE HTML>
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<title>세미나신청폼</title>
```

```
</head>
```

```
<body>
```

```
<header>
```

```
<!-- 1번 : 세미나 신청 헤드 -->
```

```
</header>
```

```
<form>
```

```
<fieldset>
```

```
<!-- 2번 : 신청상세정보 -->
```

```
</fieldset>
```

```
<fieldset>
```

```
<!-- 3번 : 신청하는 사람 -->
```

```
</fieldset>
```

```
<fieldset>
```

```
<!-- 4번 : 신청인원 -->
```

```
</fieldset>
```

```
<fieldset>
```

```
<input type="submit" formmethod="post" action="/buy" value="신청하기">
```

```
<input type="submit" formmethod="post" action="/back" formnovalidate value="취소하기">
```

```
</fieldset>
```

```
</form>
```

```
</body>
```

```
</html>
```

2023년 세미나 신청

교육기관 : (주)모블예듀

신청마감 : 2023년 11월 17일 오후 6시

신청 상세 정보 선택

- 세미나
- 일정선택 연도-월-일
- 과정선택

신청하는 사람

- 이름
- 메일주소
- 전화번호
- 주소

• 신청인원

신청하기

취소하기

세미나 신청 폼 2

• 2번 일정 선택

```
<legend>신청 상세 정보 선택</legend>
<li>
  <label for="semina">세미나</label>
  <input id="setobj" type="text" list="object">
  <datalist id="object">
    <option value="webApp" label="웹애플리케이션">
    <option value="HTML5" label="HTML5">
    <option value="android" label="안드로이드 어플리케이션">
  </datalist>
</li>
<li>
  <label for="setdate">일정선택</label>
  <input id="setdate" type="date">
</li>
<li>
  <label for="settime">과정선택</label>
  <input id="settime" type="text" list="settime">
  <datalist id="settime">
    <option value="주간오전" label="주간오전">
    <option value="주간오후" label="주간오후">
    <option value="주말종일" label="주말종일">
  </datalist>
</li>
```

• 3번 신청하는 사람

```
<legend>신청하는 사람</legend>
<li>
  <label for="name">이름</label>
  <input id="name" type="text" placeholder="홍길동"
    required autofocus>
</li>
<li>
  <label for="email">메일주소</label>
  <input id="email" type="email"
    placeholder="kdhong@email.com" required>
</li>
<li>
  <label for="phone">전화번호</label>
  <input id="phone" type="text"
    placeholder="02-1234-5678" required>
</li>
<li>
  <label for="address">주소</label>
  <textarea id="address" name="address"
    rows="5" required>
</li>
```

Canvas - image

- drawImage

- 캔버스에 이미지, 캔버스 또는 비디오를 그리는 메소드로 그림의 일부를 그리거나 확대, 축소하는 기능

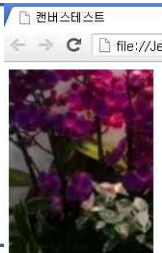
파라미터	용도
img	사용할 이미지, 캔버스, 비디오 요소를 지정
sx	클리핑 시작점의 x 좌표
sy	클리핑 시작점의 y좌표
swidth	클리핑된 이미지의 폭
sheight	클리핑된 이미지의 높이
x	이미지를 위치할 캔버스에서의 x 좌표
y	이미지를 위치할 캔버스에서의 y좌표
width	이미지의 폭으로 확대, 축소에도 사용함
height	이미지의 높이로 확대, 축소에도 사용함

Canvas - image

- 이미지 그리기 : `drawImage(image, x, y)`
 - `image` 객체를 `x, y` 위치에 그린다
- 확대, 축소 : `drawImage(image, x, y, width, height)`
 - `x, y`의 위치에 `width, height` 폭과 높이로 그린다
- 자르기 : `drawImage(image, sx, sy, sWidth, sHeight, dx, dy, dWidth, dHeight)`
 - 원본 이미지의 왼쪽 상단을 기준으로 `sx, sy` 위치에서 `sWidth, sHeight`의 폭과 높이로 잘라 `dx, dy`의 위치에 `dWidth, dHeight` 크기로 그린다

Canvas – image 클리핑

```
function draw_m(){  
var c=document.getElementById("m");  
var ctx=c.getContext("2d");  
var img=new Image();  
img.onload=function(){  
ctx.drawImage(img, 150, 100, 150, 200, 0, 0, 150, 200);  
}  
img.src = "../image/flw.jpg";  
}
```



Canvas – 이미지 클립

- Canvas/canvas_1.html

- 바디와 캔버스를 만든다 (1번)

```
<body onload="draw()">
<canvas id="c" width="500" height="500"></canvas>
</body>
```

- draw() 함수를 만든다 (2번)

```
<script type="application/javascript">
function draw(){
var canvas = document.getElementById('c');
var cntx = canvas.getContext('2d');
var img = new Image();
img.onload = function(){
cntx.drawImage(img, 10, 10);
}
img.src = '../image/flw.jpg';

cntx.beginPath();
cntx.arc(200, 110, 100, 0, Math.PI*2, true);
cntx.clip();
}
</script>
```



응용 - 폴로라이드 사진 배경 넣기

• Canvas/canvas_2.html

- 사진으로 이용할 이미지와 폴로라이드 프레임으로 쓸 이미지를 hidden속성으로 추가

```
<canvas id="gallery" width="500" height="500"></canvas>
<div style="visibility:hidden;">


</div>
```

- draw() 함수에서 프레임에 맞도록 사진이미지를 조절하고 프레임을 그린다 (2번)

```
<script type="application/javascript">
function draw(){
var canvas = document.getElementById('gallery');
var img_flower = document.getElementById('source');
var img_frame=document.getElementById('frame');

var cntx = canvas.getContext('2d');
cntx.drawImage(img_frame, 30, 30);
cntx.drawImage(img_flower, 100, 100, 500, 375, 168, 65, 158, 160);}
</script>
```



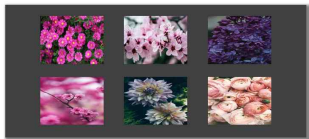
- Canvas/canvas_3.html

- <body> 태그의 안에 태그 삽입
- 태그 안에 태그를 6개 넣는다

```
<li> </li>  
<li> </li>  
<li> </li>  
<li> </li>  
<li> </li>  
<li> </li>
```

- small 클래스이 스타일을 만든다

```
#gallery ul li .small{  
  width:120px;  
  height:90px;  
}
```



응용 - 앨범 만들기 2

- 이전에 만든 앨범의 썸네일에 마우스를 가져다 대면 큰 화면으로 사진을 보는 기능
 - 각 `` 태그 밑에 아래와 같이 `` 태그를 추가
 - 클래스명을 `pic`로 한다

```
<li>  
  
  
</li>
```

- 마우스를 가져 갔을 때의 속성을 `visible`로 바꾸고
스타일을 지정한다

```
#gallery ul li: hover .pic{  
width:320px;  
height:240px;  
opacity: 1;  
visibility: visible;  
float:right;  
position:absolute;  
}
```



자바스크립트

기본 문법

- **자료형**
 - 숫자, 문자열, 불리언, 함수, 객체, 정의되지 않은 자료형
- **문자열**
 - 작은 따옴표 또는 큰 따옴표를 사용
- **일치연산자**
 - `===` 양변의 자료형과 값이 일치
 - `!==` 양변의 자료형과 값이 불일치
- **삼항연산자**
 - 조건문을 간단하게 대체하는 연산자
 - `(불리언 표현식)?(참일 때 실행하는 문장):(거짓일 때 실행하는 문장)`
 - `(number > 0) ? alert("참입니다") : alert("거짓입니다")`
- **짧은 조건문**
 - `(불리언 표현식) || (불리언표현식이 거짓일 때 실행할 문장)`
 - `(불리언 표현식) && (불리언 표현식이 참일 때 실행할 문장)`
- **for in 반복문**
 - 배열이나 객체를 쉽게 다룰 수 있는 반복문

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

```
for(var i in array){}
```

함수

- 선언적 함수

- 함수의 이름을 구체적으로 명시해 함수를 선언

```
function sumAll() {  
  alert(typeof (arguments) + ': ' + arguments.length);  
}
```

- 익명함수

- 함수의 형태를 가지고 있으나 이름이 없고 변수에 넣어 사용

```
var FunA = function () {  
  var output = prompt('이름을 입력하세요', '숫자');  
  alert(output);  
};
```

- 매개변수

- 함수를 선언할 때 매개 변수 보다 많거나 적은 매개변수의 사용을 허용

함수

- 가변인자 함수
 - 매개변수의 개수가 변할 수 있는 함수
- 내부함수
 - 함수 내부에 만들어진 함수

```
function FuncA(a, b) {  
  function FuncB(x) {  
    return x * x;  
  }  
  return Math.sqrt(FuncB(a) + FuncB(b));  
}
```

- 함수를 매개변수로 사용
 - 매개변수로 함수를 전달할 수 있다

```
function FuncA(paramFunc) {  
  for (var i = 0; i < 3; i++) {  
    paramFunc();  
  }  
}  
FuncA(function () {  
  alert('함수를 매개변수로');  
});
```

- Eval

- 전달된 문자열을 자바스크립트 코드로 실행하는 자바스크립트의 내장함수

```
var script = "";  
script += 'var a = "hello" ;';  
script += 'alert(a);';  
eval(script);  
alert(a);
```


속성의 추가와 삭제

```
var report = {};  
report.국어 = 90;  
report.수학 = 95;  
report.영어 = 90;  
report.체육 = 80;  
report.print = function(){  
  var output = "";  
  for( var key in this){  
    if(key !== 'print'){  
      output += key + " : " + this[key] + "\n";  
    }  
  }  
  return output;  
}  
alert(report.print());  
delete(report.체육);  
alert(report.print());
```

DOM

- Document Object Model
- HTML 문서의 모든 요소에 접근
- 자바스크립트로 HTML의 요소나 요소의 속성 또는 스타일을 모두 변경 가능
- 요소획득
 - getElementById()

```
function draw_m(){  
var canvas = document.getElementById('m');  
var cntx = canvas.getContext('2d');  
cntx.fillStyle="magenta";  
cntx.fillRect(0, 0, 100, 10);  
}
```

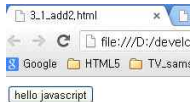
- 태그나 클래스 이름을 이용

```
var x=document.getElementById("main");  
var y=x.getElementsByTagName("p");
```

HTML 요소의 추가

```
<!DOCTYPE html>
<html>
<head>
</head>
<body>
<button>hello javascript</button>

</body>
</html>
```



HTML 요소의 추가

- **createElement()**

- document 객체의 메서드로 주어진 이름으로 요소를 생성한다.

- **appendChild()**

- 자식노드로서 노드에 노드를 생성한다.

- **createTextNode()**

- document 객체의 메서드로 텍스트노드를 생성

```
<head>
<script type="text/javascript">
function appendNode(){
var btn=document.createElement('button')
document.body.appendChild(btn);
var text = document.createTextNode('hello javascript');
btn.appendChild(text);
}
</script>
</head>
<body onload='appendNode()>
</body>
```

HTML 요소의 추가

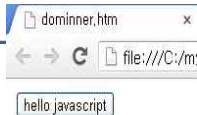
- **innerHTML**

- HTMLElement의 속성으로 요소의 내부 HTML 속성을 설정하거나 반환

```
<script type="text/javascript">
window.onload = function drawText(){
document.body.innerHTML = "hello javascript";
}
</script>
```

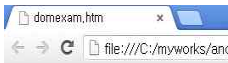
- 텍스트가 아닌 요소도 아래와 같이 추가 가능

```
document.body.innerHTML = '<button>hello javascript</button>'
```



예제 - 객체 만들기

- h1 요소를 만들고 텍스트 노드를 만들어 **hello javascript** 문자열을 웹페이지에 보여준다



hello javascript

요소의 속성

- `setAttribute()`

입력창 변경하기

날짜를 입력하세요

입력창 변경하기

날짜를 입력하세요

연도-월-일

<< < 2013년 2월 > >>

일	월	화	수	목	금	토
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	1	2
3	4	5	6	7	8	9

오늘 삭제

```
<body>
입력창 변경하기<button onclick='changeType()>change</button>
<p>날짜를 입력하세요</p>
<input id='in_date'>
</body>
```

```
<script type="text/javascript">
function changeType(){
var input = document.getElementById('in_date');①
input.setAttribute('type','date'); ②
}
</script>
```

요소의 속성

- **getAttribute()**

```
<head>
<script type="text/javascript">
function getWH(){
var img = document.getElementById('mying');
var out = img.getAttribute('width');
out += ', ' + img.getAttribute('height');
document.getElementById('output').innerHTML=out;
}
</script>
</head>
<body>
<img id = 'mying' src='image/boracay2.jpg' width='300' height='200'>
</body>
```


예제 - 속성 확인 및 설정

- 문서에 나타난 이미지의 폭을 확인하고 그 값을
- 문서에 `img` 태그와, 변경할 `width`를 입력할 `input`, 폭을 설정할 버튼, 폭을 확인할 버튼과 확인한 폭을 표시할 요소를 아래 그림과 같이 구성



스타일 변경과 요소의 제거

- 스타일 변경하기

- DOM은 자바스크립트로 HTML5 요소의 변경이 가능

```
document.getElementById(id).style.property=new style
```

예제 – 요소의 변경

- `document.getElementById('id1').style.color='red';`
- 그림과 같이 텍스트 색상을 바꿔본다

Hello World

Button

Hello World

Button

요소 제거하기

- `elementNode.removeChild(node)`

```
<!DOCTYPE html>
<html>
<head>
<script>
function removeHeader () {
var remove = document.getElementById('header');
document.body.removeChild(remove);
};
</script>
</head>

<body>
<h1 id="header">Hello World</h1>
<button onclick="removeHeader()">remove</button>
</body>
</html>
```

요소 제거하기

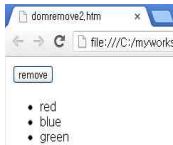
- 특정 아이디의 요소 제거

```
<body>
<button onclick="removeHeader()">remove</button>
<ul id="color">
<li id="red">red</li>
<li id="blue">blue</li>
<li id="green">green</li>
</ul>
</body>
```

```
function removeHeader () {
var red = document.getElementById('red');
red.parentNode.removeChild(red);
};
```

예제 - 리스트 아이템 삭제

- 빨강, 노랑, 파랑 텍스트를 리스트로 구현하고 버튼을 눌러 빨강 아이템을 삭제해보자



이벤트

```
<body>
<script>
window.onload = function () {
var header = document.getElementById('h1_text'); ①
header.onclick = function (event) {              ②
    alert('click');                               ③
};
};
</script>

<h1 id="h1_text">hello world</h1>
</body>
```

- 이벤트는 **click** 이고, **onclick**는 이벤트 속성이다. 그리고 이벤트 속성에 연결된 함수 **function(event){ alert('click');}**가 이벤트 핸들러다.