

ABC3012: Advanced Web Programming

Donghyun Kang (donghyun@changwon.ac.kr)

NOSLab (<https://noslab.github.io>)

Changwon National University

문서의 전형적인 구조 사례

자바 GUI 스원 기초

8 CHAPTER

323

GUI 프로그램은 이미지나 도형을 다루는 클래스와 이벤트를 처리를 위한 클래스를 사용하기 때문에, 다음 import 문이 함께 필요한 경우가 많다.

```
import java.awt.*; // 그래픽 처리를 위한 클래스들의 경로명
import java.awt.event.*; // AWT 이벤트를 사용하기 위한 경로명
import javax.swing.*; // Swing 컴포넌트 클래스들의 경로명
import javax.swing.event.*; // Swing 이벤트를 위한 경로명
```

스윙 프레임과 컨테넌트

스윙 프레임은 모든 스윙 컴포넌트들을 담는 최상위 컨테이너(Top Level Container)이다. 프레임이 출력될 때, 프레임 내에 부착된 모든 컴포넌트들이 화면에 출력된다. 컴포넌트들은 프레임 없이 독립적으로 화면에 출력될 수 없다. 프레임이 닫히면 프레임 내의 모든 컴포넌트들도 프레임과 함께 화면에서 사라진다.

스윙에서 프레임의 역할을 하는 클래스가 JFrame이다. JFrame 객체는 [그림 8-4]와 같이 구성된다. JFrame 객체는 Frame(java.awt.Frame), 메뉴바(Menu Bar), 컨테넌트(Content Pane)의 세 공간으로 구성된다. Frame은 AWT 패키지에 있는 클래스로서 JFrame이 [그림 8-2]와 같이 java.awt.Frame을 상속받기 때문에 당연히 존재하며, 모든 메뉴는 메뉴바에 부착된다.

컨테넌트패널은 메뉴를 제외한 모든 GUI 컴포넌트들을 부착하는 공간이므로, 개발자는 화면에 출력하고자 하는 GUI 컴포넌트들을 컨테넌트패널에 붙여야 한다.

[그림 8-4]의 왼쪽은 JFrame의 사용 예로서 5개의 메뉴가 메뉴바에 부착되어 있으며, 컨테넌트패널에는 add, sub, mul, div 이름의 버튼 컴포넌트들이 부착되어 있다.

JFrame

컨테넌트

[그림 8-4] 스윙 프레임(JFrame)의 구성

페이지 번호

장 제목

소제목

본문 텍스트

문단

설명 섹션

그림

용어 정리

- 시맨틱 & 시맨틱 웹
 - 시맨틱 (semantic): 의미론적인
 - **‘의미론적인 웹’**이라는 뜻으로, 현재의 인터넷과 같은 분산환경에서 정보와 자원 사이의 관계-의미 정보 (Semanteme)를 제공하는 웹
 - 즉, 문서를 구조화하여 의미 있는 내용 검색이 용이한 웹
- 웹 폼

HTML5의 문서 구조화

- 기존 HTML의 한계
 - 웹 문서 구조를 표현하는 태그 없음
 - <div> 태그나 <table> 태그로 구조화되어 보이게 작성
 - HTML 페이지의 소스를 보면 문서 구조 파악 불가능
 - 어디에 어떤 내용이 있는지??? 어떤 키워드 있는지??
 - E.g., 목차, 기사, 저작권 정보, 등.

와글메인 • 창원대학교 • 입학안내

로그인 • 회원가입

창원대학교 컴퓨터공학과

학과안내

교수소개

학사안내

학과게시판

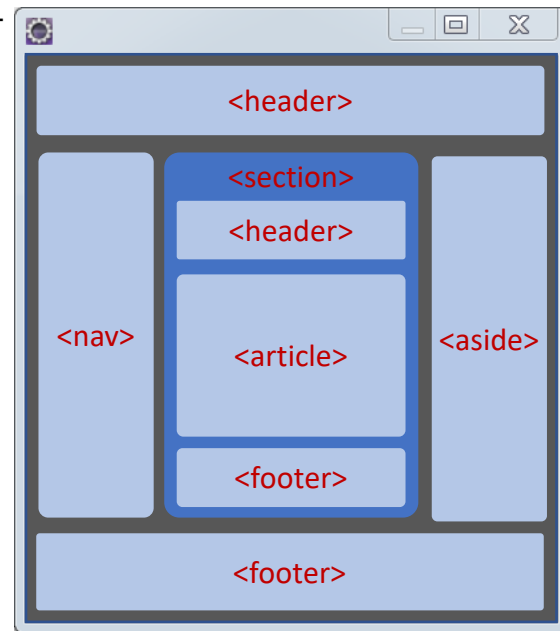
커뮤니티



교수 소개를 보고싶다면??
대학원에 대해서 알고 싶다면???

HTML5의 문서 구조화

- 문서 구조화의 이유
 - 검색 엔진이 좋아하는 웹 페이지 작성의 필요성 대두
 - 정보 탐색이 중요해진 시대
 - 사물인터넷으로 IT 장치들이 스스로 정보 검색하는 시대
 - 사용자가 만든 웹 페이지 가치 극대화 - 탐색이 쉽도록 작성
- 시맨틱 태그
 - 문서의 구조와 의미를 전달하는 태그
 - <header>, <section>, <article>, <main>, <summary>, <mark>, <time>
- 구글 검색 엔진 사례
 - 웹 페이지에서 시맨틱 태그 검색
 - 리뷰, 사람, 제품, 업체, 이벤트, 음악 등 검색 결과 제공



HTML5 시맨틱 태그로 구조화한 웹 페이지 사례

시맨틱 태그

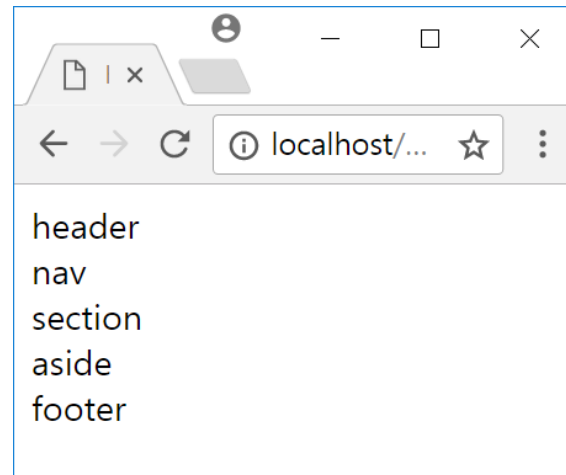
- `<header>`
 - 페이지나 섹션의 머리말 표현
 - 페이지 제목, 페이지를 소개하는 간단한 설명
- `<nav>`
 - 하이퍼링크들을 모아 놓은 특별한 섹션
 - 페이지 내 목차를 만드는 용도
- `<section>`
 - 문서의 장(chapter, section) 혹은 절을 구성하는 역할
 - 일반 문서에 여러 장이 있듯이 웹 페이지에 여러 `<section>`가
능
 - 헤딩 태그(`<h1>`~`<h6>`)를 사용하여 절 혹은 섹션의 주제 기입
- `<article>`
 - 본문과 연관 있지만, 독립적인 콘텐츠를 담는 영역
 - 보조 기사, 블로그 포스트, 댓글 등 기타 독립적인 내용
 - `<article>`에 담는 내용이 많은 경우 여러 `<section>` 둘 수 있음
- `<aside>`
 - 본문에서 약간 벗어난 노트나 팁
 - 신문, 잡지에서 주요 기사 옆 관련 기사, 삽입 어구로 표시된
논평 등
 - 페이지의 오른쪽이나 왼쪽에 주로 배치
- `<footer>`
 - 꼬리말 영역, 주로 저자나 저작권 정보

문서의 모양은 구조와 별개

- 시맨틱 태그

- <header>, <section>, <article>, <footer>, <nav> 태그
- 위치와 색, 모양이 자동으로 결정되는 것이 아님

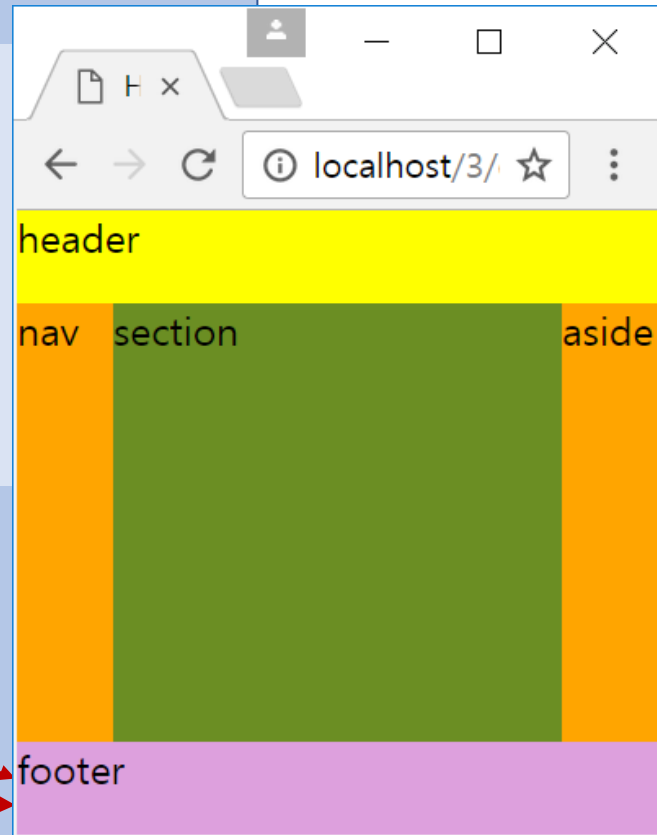
```
<!DOCTYPE html>
<html>
<head>
<title>HTML5 문서 구조 시맨틱 태그 사용</title>
</head>
<body>
  <header>header</header>
  <nav>nav</nav>
  <section>section</section>
  <aside>aside</aside>
  <footer>footer</footer>
</body>
</html>
```



문서의 모양은 구조와 별개

```
<!DOCTYPE html>
<html>
<head><title>HTML5 문서 구조 시맨틱 태그 사용</title>
<style>
  html, body { margin: 0; padding: 0; height: 100%; }
  .header { width: 100%; height: 15%;
    background: yellow; }
  .nav { width: 15%; height: 70%; float: left;
    background: orange; }
  .section { width: 70%; height: 70%; float: left;
    background: olivedrab; }
  .aside { width: 15%; height: 70%; float: left;
    background: orange; }
  .footer { width: 100%; height: 15%; clear: both;
    background: plum; }
</style>
</head>
<body>
  <header class="header">header</header>
  <nav class="nav">nav</nav>
  <section class="section">section</section>
  <aside class="aside">aside</aside>
  <footer class="footer">footer</footer>
</body>
</html>
```

CSS3 스타일 시트.
각 영역의 색과
위치 꾸미기



HTML5 문서 구조화 연습

- 문서를 구조화할 때 권장 사항
 - 웹 페이지 전체를 구조화 시맨틱 태그로 분할
 - 웹 페이지 전체 제목과 소개는 <header> 태그로 작성
 - 본문은 <section>으로 묶고,
 - 본문 내에 각 절이나 영역은 <article>로 작성
 - 링크나 메뉴들은 <nav> 태그로 작성
 - <header>, <section>, <article>, <aside> 등에는 헤딩 태그(<h1>~<h6>)를 이용하여 제목을 붙임
 - 배경 음악을 연주하는 <audio> 태그는 <header> 영역에 삽입
 - 문서의 모양(<header>, <section>, <article>, <aside>의 배치)은 CSS3 스타일 시트로 꾸미기

예제

본문
(<section>)

머리말
(<header>)

목차
(<nav>)

절
(<article>)

절
(<article>)

절
(<article>)

노트
(<aside>)


꼬리말
(<footer>)

시맨틱 태그로 구조화한 예

localhost/3/fig3-07.html

볼프강 아마데우스 모차르트

1756년 1월 27일 잘츠부르크에서 태어난 천재적인 오스트리아의 작곡가 모차르트(1756년 1월 27일 ~ 1791년 12월 5일)를 소개한다.



1770년대 초상화

목차

- [생애](#)
- [죽음](#)
- [음악](#)

생애

모차르트는 1756년 1월 27일 잘츠부르크에서 태어나서, 궁정 음악가였던 아버지에게 피아노와 바이올린을 배웠고, 다섯살 때 이미 작곡을 하기 시작했으며, 1764년에서 1765년 사이에 바흐로부터 처음으로 교향곡을 작곡하는 법을 배웠는데 이것이 모차르트가 수많은 교향곡을 남기는 계기가 되었다. 모차르트는 빈에서 1784년에 14세인 베토벤을 만나 베토벤을 교육시키는데 전념하기도 했다.

죽음

모차르트는 1791년 12월 5일 오전 0시 55분경에 갑자기 병으로 죽었으며 모차르트가 완성하지 못한 작품 레퀴엠은 프란츠 크사버 쥐스마이어(Franz Xaver Süssmayr)가 완성시켰다.

음악

오페라, 교향곡, 행진곡, 관현악용 무곡, 피아노 협주곡, 바이올린 협주곡, 교회용 성악곡, 칸타타, 미사곡 등 다양한 장르를 아우르며 600 여곡을 작곡하여 후대에 남겼다.

모차르트의 죽음에 얽힌 전설

모차르트의 장례식 날 비가 오고 천둥이 쳤다고 하나 New Groove에 따르면 사실은 구름 한 점 없는 쾌청한 날이었다고 한다.

2017년 10월 7일 작성, 위키피디어 참고

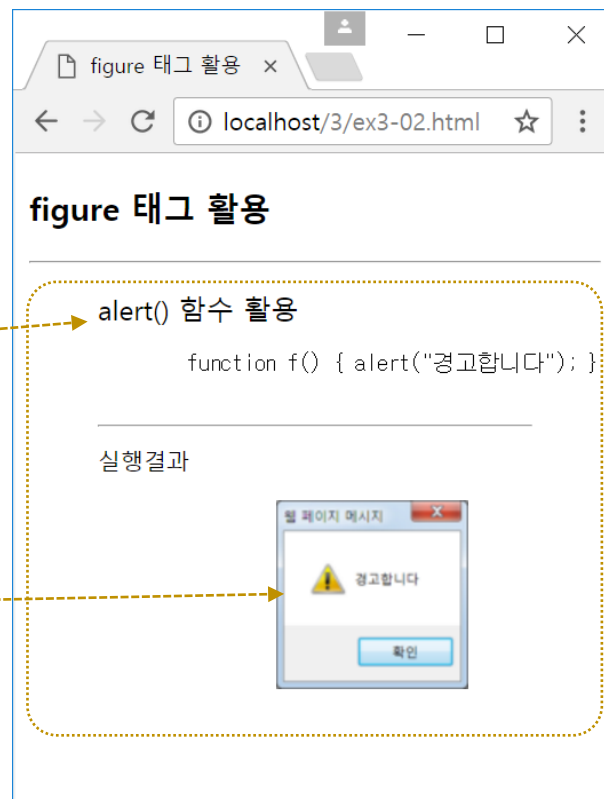
시맨틱 태그들

- 시맨틱 블록 태그
 - `<figure>`
 - 책이나 보고서 등 본문에 삽입하는 사진, 차트, 삽화, 소스 코드 등
 - `<details>`와 `<summary>`
 - `<details>`는 상세 정보를 담는 시맨틱 블록 태그
 - `<summary>` 태그는 `<details>`로 구성되는 블록의 제목 표현

예제 <figure>

- 'alert() 함수에 대한 설명'과 '실행 결과'를 하나의 그림으로 블록화 함

```
<!DOCTYPE html>
<html>
<head><title>figure 태그 활용</title>
</head>
<body>
<h3>figure 태그 활용</h3>
<hr>
<figure id="1-1">
  <figcaption>alert() 함수 활용</figcaption>
  <pre>
    <code>function f() { alert("경고합니다");}</code>
  </pre>
  <hr>
  <small>실행결과</small>
  <pre>
    
  </pre>
</figure>
</body>
</html>
```

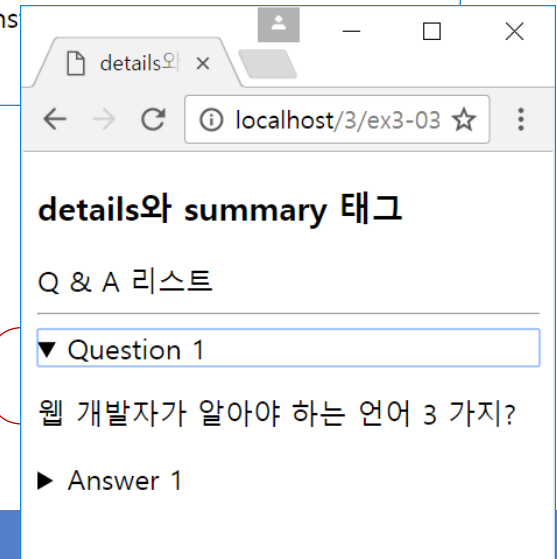
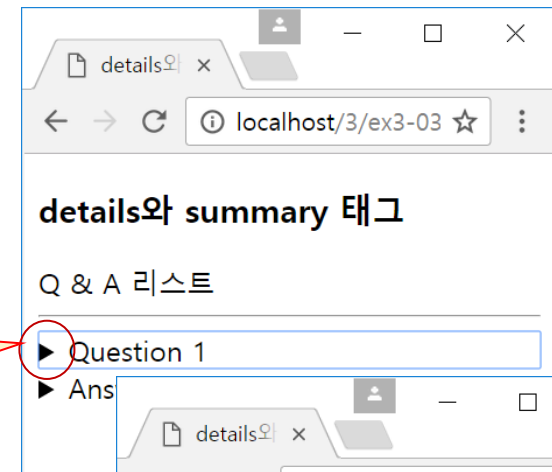


예제 <details> & <summary>

- 이 예제는 <details> 태그를 이용하여 Q&A 리스트를 출력함

```
<!DOCTYPE html>
<html>
<head>
<title>details와 summary 태그</title>
</head>
<body>
<h3>details와 summary 태그</h3>
Q & A 리스트
<hr>
<details>
  <summary>Question 1</summary>
  <p>웹 개발자가 알아야 하는 언어 3 가지?</p>
</details>
<details>
  <summary>Answer 1</summary>
  <p>HTML5, CSS, Javascript</p>
</details>
</body>
</html>
```

핸들을 클릭하면
아래와 같이 상세
정보가 펼쳐짐



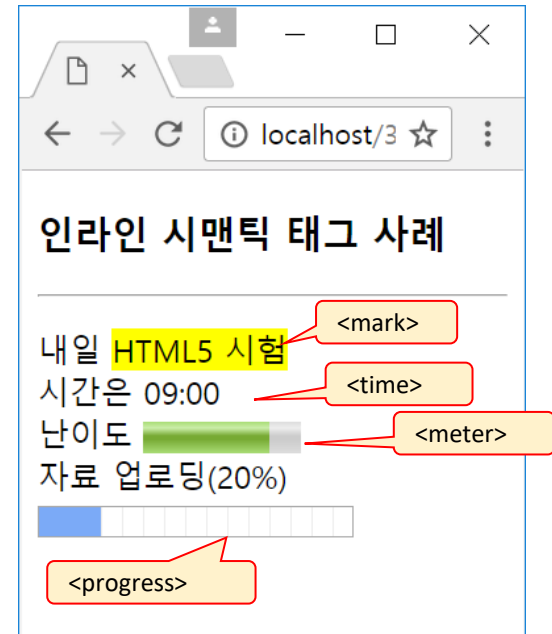
시맨틱 태그들

- 시맨틱 인라인 태그
 - <mark>
 - 중요한 텍스트임을 표시
 - <time>
 - 텍스트의 내용이 시간임을 표시
 - <meter>
 - 주어진 범위나 %의 데이터 량 표시
 - <progress>
 - 작업의 진행 정도 표시

예제 (인라인 태그)

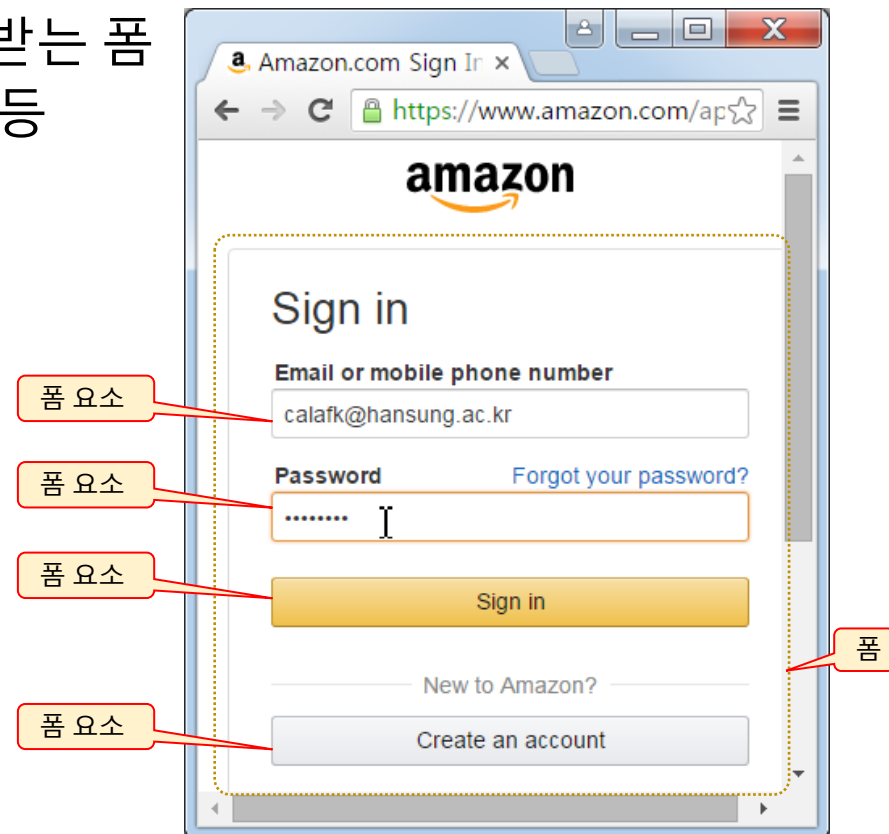
- 4가지 인라인 시맨틱 태그의 사용 사례

```
<!DOCTYPE html>
<html>
<head>
<title>인라인 시맨틱 태그</title>
</head>
<body>
<h3>인라인 시맨틱 태그 사례</h3>
<hr>
<p>내일 <mark>HTML5 시험</mark><br>
시간은 <time>09:00</time><br>
난이도 <meter value="0.8">80%</meter><br>
자료 업로딩(20%)
  <progress value="2" max="10"></progress><br>
</p>
</body>
</html>
```



웹 폼

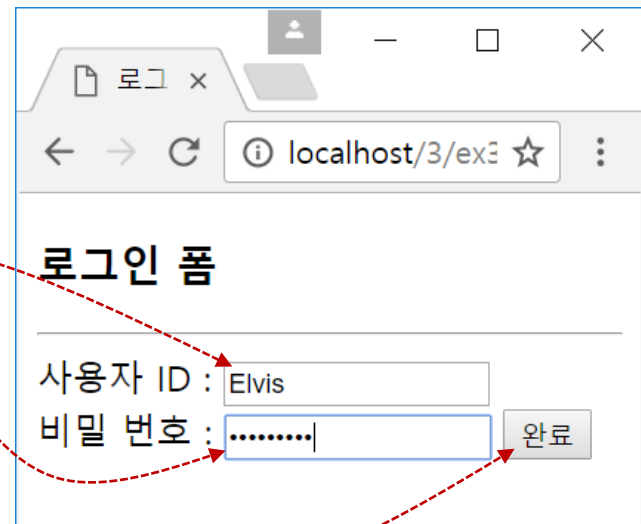
- 웹 폼
 - 웹 페이지에서 사용자 입력을 받는 폼
 - 로그인, 등록, 검색, 예약, 쇼핑 등
- 폼 요소
 - 폼을 만드는 다양한 태그
 - `<input>`, `<textarea>`, `<select>` 등



로그인 폼

- 예제

```
<!DOCTYPE html>
<html>
<head><title>로그인 폼</title></head>
<body>
<h3>로그인 폼</h3>
<hr>
<form name="fo" method="get">
사용자 ID : <input type="text" size="15" value=""><br>
비밀 번호 : <input type="password" size="15" value="">
               <input type="submit" value="완료">
</form>
</body>
</html>
```



폼 작성

- 폼 태그 <form> 태그로 둘러싸는 모양

```
<form name="fo" method="get">  
...  
</form>
```

- name 속성
 - 폼의 이름 지정
- action 속성
 - 폼 데이터를 처리할 웹 서버 응용프로그램의 이름
 - submit 버튼이 눌리면 브라우저는 action에 지정된 웹 서버 응용프로그램 실행 요청
 - 웹 서버 응용프로그램은 Java, JSP, PHP, C/C++ 등 다양한 언어로 작성
- method 속성
 - 폼 데이터를 웹 서버로 전송하는 형식
 - 대표적인 전송 방식 : GET, POST

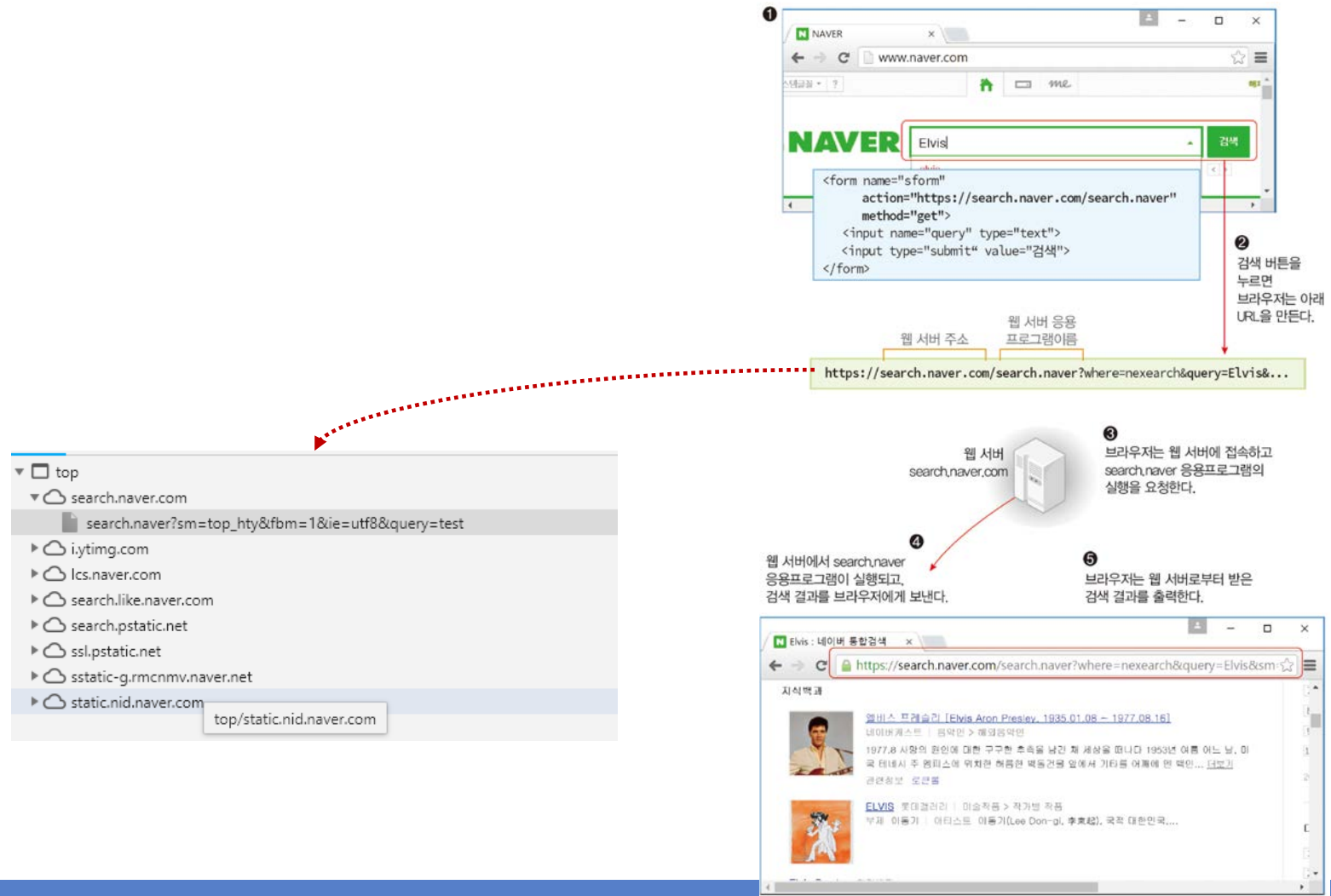
<form> 태그

```
<form action="웹 서버 응용 프로그램의 URL"
      enctype="데이터의 인코딩 타입"
      method="GET|POST"
      name="폼 이름"
      target="윈도우 이름">
  ... 여러 개의 폼 요소 작성
```

```
</form>
```

- action : 폼 데이터를 처리할 웹 서버 응용프로그램 이름 URL
- enctype : 폼 데이터를 웹 서버로 전송할 때 암호화 방식 지정
- method : 폼 데이터를 웹 서버에 전송하는 방식
- target : 웹 서버 응용프로그램으로부터 전송받은 데이터(HTML 문서나 이미지)를 출력할 윈도우 이름

웹 폼 데이터 전송 순서



폼 요소의 종류

폼 요소	설명
<code><input type="text"></code>	한 줄 텍스트 입력 창
<code><input type="password"></code>	암호 입력을 위한 한 줄 텍스트 입력 창
<code><input type="button"></code>	단순 버튼
<code><input type="submit"></code>	웹 서버로 폼 데이터를 전송시키는 버튼
<code><input type="reset"></code>	입력된 폼 데이터를 초기화시키는 버튼
<code><input type="image"></code>	이미지 버튼
<code><input type="checkbox radio"></code>	체크박스과 라디오버튼
<code><select></code>	드롭다운 리스트를 가진 콤보박스
<code><input type="month week date time datetime-local"></code>	년, 월, 일, 시간 등의 시간 정보 입력 창
<code><input type="number range"></code>	스핀 버튼과 슬라이드바로 편리한 숫자 입력 창
<code><input type="color"></code>	색 입력을 쉽게 하는 컬러 다이얼로그
<code><input type="email url tel search"></code>	이메일, URL, 전화번호, 검색키워드 등 형식 검사 기능을 가진 텍스트 입력 창
<code><input type="file"></code>	로컬 컴퓨터의 파일을 선택하는 폼 요소
<code><button type="button reset submit"></code>	단순 버튼, reset, submit 버튼
<code><textarea></code>	여러 줄의 텍스트 입력 창

텍스트 입력

- `<input type="text">`
 - 한 줄 짜리 입력 창
- `<input type="password">`
 - 암호 입력 창
 - 사용자 입력 문자 대신 '*' 등 다른 글자로 출력
- `<textarea>`
 - 여러 줄 입력 창

```
<input type="text|password"
      name="요소이름"
      maxlength="문자개수"
      size="문자개수"
      value="초기텍스트">
```

- `maxlength` : 입력할 수 있는 문자의 최대 개수
- `size` : 입력 창의 크기. 단위는 문자 개수

```
<textarea cols="열 개수"
          rows="행 개수"
          name="요소이름"
          wrap="OFF|HARD|SOFT">
  초기 출력될 텍스트
</textarea>
```

- `cols`, `rows` : 텍스트 입력창의 크기로 가로세로 문자 수 (영어를 기준으로)
- `wrap` : 자동 줄바꿈 처리 지정

예제 (텍스트 입력)

```
<!DOCTYPE html>
<html>
<head><title>텍스트 입력 폼</title></head>
<body>
<h3>자기 소개서 작성</h3>
<hr>
<form>
  이름 : <input type="text" value=""><br>
  암호 : <input type="password" value="" maxlength="4"><br>
  자소서 : <textarea cols="20" rows="5">
            이곳에 자기소개서 작성
          </textarea>
</form>
</body>
</html>
```

자기 소개서 작성

이름 : Elvis

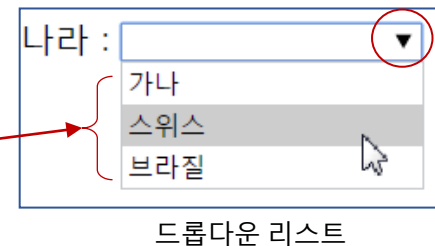
암호 :

자소서 : 저는 엘비스입니다. 저를 채용해주시면, 최고의 회사를 만들겠습니다.

데이터 목록을 가진 텍스트 입력 창, <datalist>

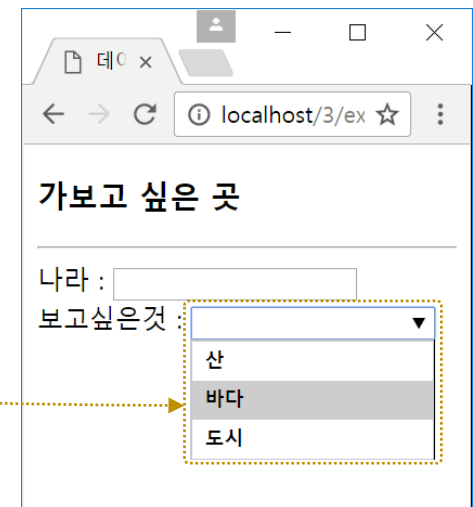
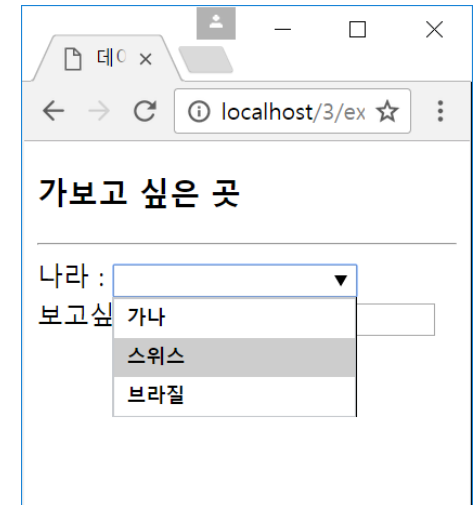
- <datalist>
 - 목록 리스트를 작성하는 태그
 - <option> 태그로 항목 하나 표현
 - <input type="text">에 입력 가능한 데이터 목록 제고

나라 : <input type="text" list="countries">
 <datalist id="countries">
 <option value="가나">
 <option value="스위스">
 <option value="브라질">
 </datalist>



예제

```
<!DOCTYPE html>
<html>
<head><title>데이터 목록을 가진 텍스트 입력</title></head>
<body>
<h3>가보고 싶은 곳</h3>
<hr>
<form>
나라 : <input type="text" list="countries"><br>
      <datalist id="countries">
        <option value="가나">
        <option value="스위스">
        <option value="브라질">
      </datalist>
보고싶은것 : <input type="text" list="what"><br>
             <datalist id="what">
               <option value="산">
               <option value="바다">
               <option value="도시">
             </datalist>
</form>
</body>
</html>
```



텍스트/이미지 버튼 만들기

- `<input type="button|reset|submit|image" value="버튼의 문자열">`
- `<button type="button|reset|submit">버튼의 문자열</button>`

```
<input type="button|reset|submit|image"
      name="버튼 이름"
      value="문자열"
      src="이미지 URL">
```

- value : 버튼에 출력되는 문자열
- src : type="image"인 경우에만 필요한 것으로 이미지의 URL

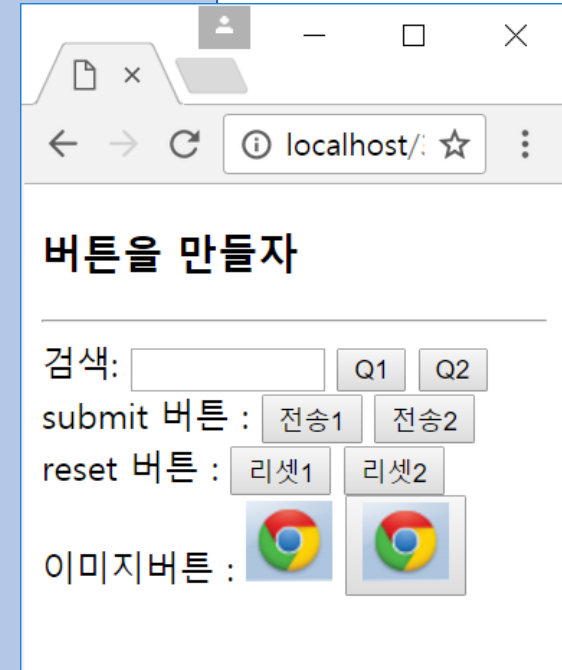
```
<button type="button|reset|submit"
      name="버튼 이름"
      value="문자열">
```

버튼 문자열이나 이미지

```
</button>
```

예제

```
<!DOCTYPE html>
<html>
<head><title>버튼이 있는 입력 폼</title></head>
<body>
<h3>버튼을 만들자</h3>
<hr>
<form>
  검색: <input type="text" size="10" value="">
        <input type="button" value="Q1">
        <button type="button">Q2</button><br>
  submit 버튼 : <input type="submit" value="전송1">
                  <button type="submit">전송2</button><br>
  reset 버튼 : <input type="reset" value="리셋1">
                 <button type="reset">리셋2</button><br>
  이미지버튼 : <input type="image" src="media/button.png"
                 alt="이미지 버튼">
                <button type="button">
                  
                </button>
</form>
</body>
</html>
```



선택형 입력 : 체크박스과 라디오버튼

- `<input type="checkbox">`
 - 체크박스 만들기
- `<input type="radio">`
 - 라디오 버튼 만들기
 - name 속성 값이 같은 라디오버튼들이 하나의 그룹 형성

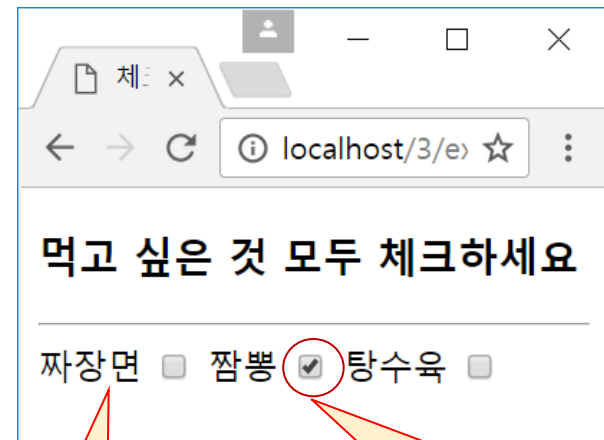
```
<input type="checkbox|radio"  
      name="요소 이름"  
      value="요소 값"  
      checked>
```

- value : 폼 요소가 선택된 상태일 때, 웹서버에 전송되는 값
- checked : 이 속성이 있으면, 초기에 선택 상태로 출력

예제 (체크 박스)

```
<!DOCTYPE html>
<html>
<head><title>체크 박스 만들기</title></head>
<body>
<h3>먹고 싶은 것 모두 체크하세요</h3>
<hr>
<form>
  짜장면 <input type="checkbox" value="1">
  짬뽕 <input type="checkbox" value="2" checked>
  탕수육 <input type="checkbox" value="3">
</form>
</body>
</html>
```

캡션



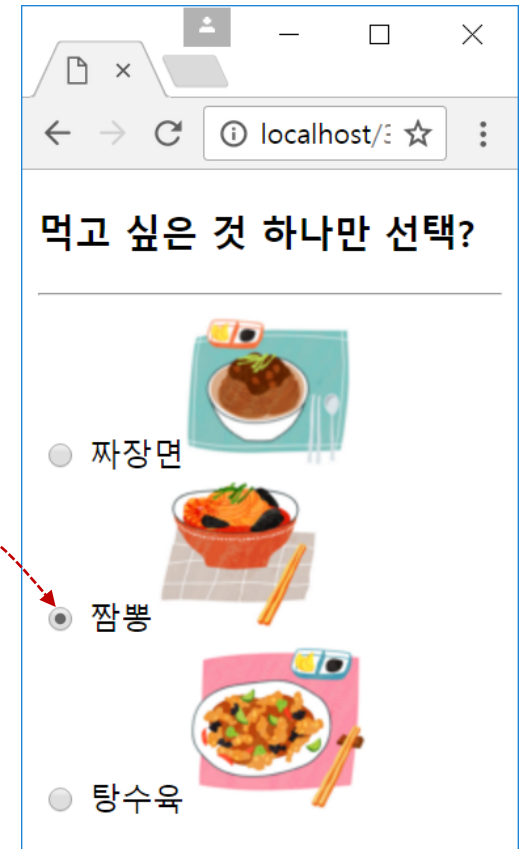
캡션

checked 속성이 사용되면
초기에 체크된 상태로 출력

예제 (라디오 박스)

```
<!DOCTYPE html>
<html>
<head><title>라디오버튼 만들기</title></head>
<body>
<h3>먹고 싶은 것 하나만 선택?</h3>
<hr>
<form>
  <input type="radio" name="china" value="1">
    짜장면<br>
  <input type="radio" name="china" value="2" checked>
    짬뽕<br>
  <input type="radio" name="china" value="3">
    탕수육
</form>
</body>
</html>
```

같은 name을 가진
라디오버튼 중 하나만
선택 가능



선택형 입력 : 콤보 박스

- `<select>`
 - 드롭다운 리스트에 목록 출력, 목록을 선택하는 입력 방식
 - `<option>` 태그로 항목 하나 표현

```
<select name="요소 이름"
        size="개수"
        multiple>
```

여러 개의 `<option>...</option>` 태그로 선택 항목 만들기

```
</select>
```

- `size` : 콤보박스 창 크기(보일 수 있는 최대 항목 개수). 디폴트는 1
- `multiple` : 이 속성이 있으면 다수 항목 선택 가능

```
<option value="옵션 값"
        selected>
```

항목 문자열

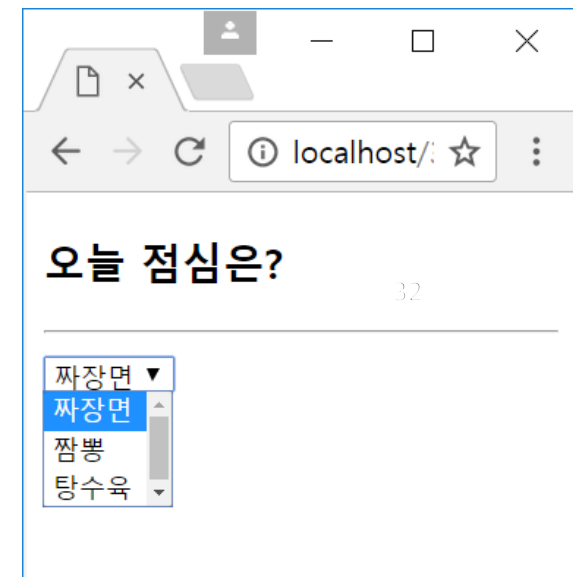
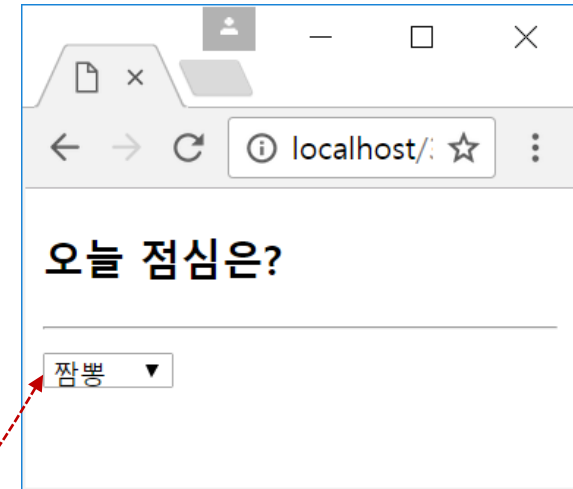
```
</option>
```

- `value` : 이 항목이 선택되었을 때 웹 서버에 전송되는 값
- `selected` : 이 속성이 있으면 초기에 선택 상태로 출력

예제

```
<!DOCTYPE html>
<html>
<head><title>콤보박스 만들기</title></head>
<body>
<h3>먹고 싶은 음식은?</h3>
<hr>
<form>
  <select name="china">
    <option value="1">짜장면</option>
    <option value="2" selected>짬뽕</option>
    <option value="3">탕수육</option>
  </select>
</form>
</body>
</html>
```

selected 속성이
사용되면 선택 상태



Q&A

과제 1

- 다음과 같이 출력되는 HTML5 페이지를 작성하시오
 - <summary> and <details> 태그
 - <mark> 태그
 - <time> 태그
 - <progress> 태그

라디오 박스 실습

웹프로그래밍특론의 교수는 누구인가요?

☐ 이우선 ☐ 우용태 ☐ 정성환 ☐ 이수현 ☐ 차정원 ☐ 정성욱 ☐ 안동혁 ☐ 강동현

▼ 강동현

이메일: donghyun@changwon.ac.kr

과목명

웹프로그래밍특론

학습 진도(40/100)



과제 2

- 다음과 같이 출력되는 HTML5 페이지를 작성하시오

- `<button>` 태그

버튼을 만들자

1	2	3
4	5	6
7	8	0

과제 3

- 다음과 같이 출력되는 HTML5 페이지를 작성하시오
 - 검색 방법: P127 참고

```
<form name="sform"
  action="https://search.naver.com/search.naver"
  method="get">
  <input name="query" type="text">
  <input type="submit" value="검색">
</form>
```

나만의 검색 폼

검색할 단어를 입력하세요 (최대 길이는 10 문자입니다).

네이버 검색 : 장원대학교

검색

클릭시

