L2-HTML

예제 실습 및 보고서 작성



학 과 : 컴퓨터공학부 컴퓨터공학전공

과 목 : 웹프로그래밍

교수님 : 한영선 교수님

학 번 : 202030494

이 름 : 윤원재

제출일 : 2023년 3월 일

목차

[1. 개요 3](#개요)

[2. 예제 2-1 웹 페이지 타이틀 달기 3](#예제1)

[3. 예제 2-2 <h1>, …, <h6> 태그로 문단 제목 달기 4](#예제2)

[4. 예제 2-3 title 속성으로 툴팁 달기 4](#예제3)

[5. 예제 2-4 <p>로 단락 나누기 5](#예제4)

[6. 예제 2-5 <hr> 태그로 수평선 긋기 6](#예제5)

[7. 예제 2-6 <br> 태그로 새로운 줄로 넘어가기 7](#예제6)

[8. 예제 2-7 특수 문자, 기호, 심볼 삽입 8](#예제7)

[9. 예제 2-8 <pre> 태그로 개발자의 포맷 그대로 출력 9](#예제8)

[10. 예제 2-9 텍스트 꾸미기 10](#예제9)

[11. 예제 2-10 <div> 블록과 <span> 인라인 11](#예제10)

[12. 예제 2-11 <img> 태그로 이미지 삽입 12](#예제11)

[13. 예제 2-12 <ol>로 라면 끓이는 순서 나열 13](#예제12)

[14. 예제 2-13 <ul>로 좋아하는 음식 나열 14](#예제13)

[15. 예제 2-14 중첩 리스트 만들기 15](#예제14)

[16. 예제 2-15 정의 리스트 만들기 16](#예제15)

[17. 예제 2-16 기본 표 만들기 17](#예제16)

[18. 예제 2-17 이미지를 가지는 표 만들기 18](#예제17)

[19. 예제 2-18 하이퍼링크 만들기 19](#예제18)

[20. 예제 2-19 링크의 target 속성 활용 20](#예제19)

[21. 예제 2-20 앵커로 이동하는 링크 만들기 21](#예제20)

[22. 예제 2-21 파일 다운로드 링크 만들기 22](#예제21)

[23. 예제 2-22 2개의 인라인 프레임을 가진 웹 페이지 23](#예제22)

[24. 예제 2-23 <iframe>으로 2개의 사이트 내장하기 24](#예제23)

[25. 예제 2-24 <video> 태그로 비디오 삽입 25](#예제24)

[26. 예제 2-25 <audio> 태그로 오디오 삽입 26](#예제25)

[27. 참고문헌 27](#참고문헌)

1. **개요**

**<!DOCTYPE html>**

**<html>**

**<head>**

문서제목, 자바스크립트 코드, CSS 스타일 정의, 메타데이터정의

**</head>**

**<body>**

문서의 본문 텍스트, 이미지, 테이블, 자바스크립트 코드, 동영상 등

**</body>**

**</html>**

HTML(Hypertext Markup Language)는 웹페이지를 구성하기 위해 사용되는 마크업 언어이다. 여러가지 태그를 사용하며, 그 중 필수적으로 사용되는 태그가 있다. <!DOCTYPE html>, <html>, <head>, <tilte>, <body>가 있으며 <!DOCTYPE html>는 HTML5 문서임을 알리는 지시어이다. <html>는 모든 html요소들을 감싸는 컨테이너이며, <head>와 <body>는 각각 문서 제목, 문서의 본문 내용을 표현하기 위한 태그이다. <title>에 대한 설명 및 추가적인 태그는 분문에서 다루겠다.

1. **예제 2-1 웹 페이지 타이틀 달기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**첫 타이틀**</title>**

**</head>**

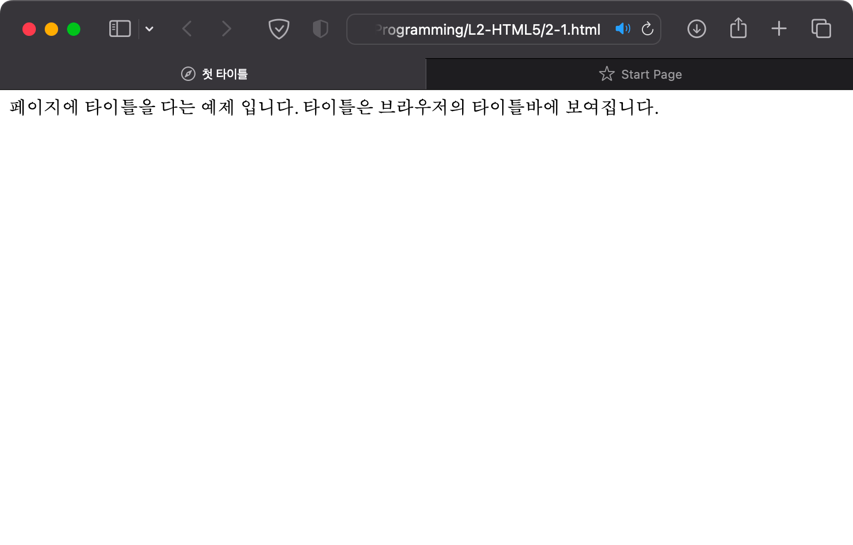
**<body>**

페이지에 타이틀을 다는 예제 입니다.

타이틀은 브라우저의 타이틀바에 보여집니다.

**</body>**

**</html>**

****

<그림 2-1> 예제 2-1 결과

<tilte>태그는 웹 브라우저의 타이틀을 지정하는 태그이다. <title>과 </title>사이에 적혀진 내용이 브라우저의 타이틀바에 보여진다. <그림 2-1>의 경우 ‘<title>첫 타이틀</title>’을 통해 ‘첫 타이틀’이 타이틀바로 보여진 것을 확인할 수 있다.

1. **예제 2-2 <h1>, …, <h6> 태그로 문단 제목 달기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**문단 제목 달기**</title>**

**</head>**

**<body>**

**<h1>**1장 홈페이지 만들기**</h1>**

**<h2>**1절 HTML 언어**</h2>**

**<h3>**1. 웹**</h3>**

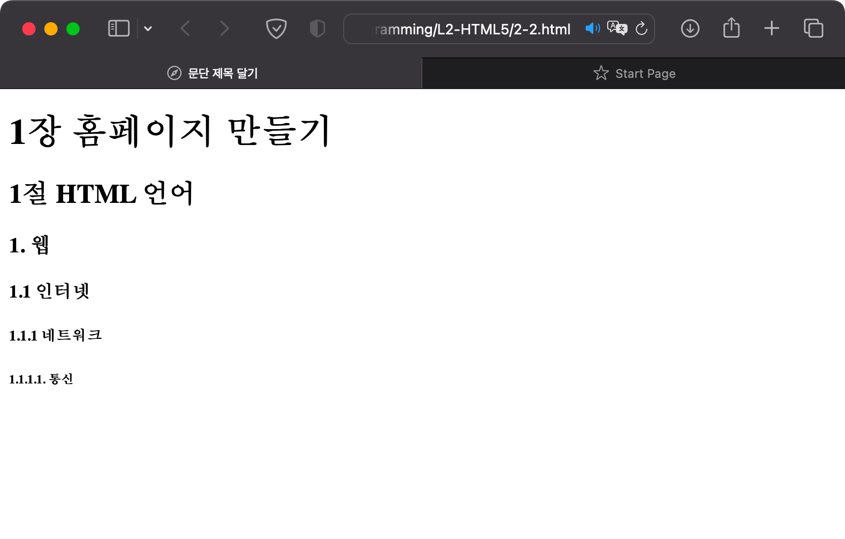
**<h4>**1.1 인터넷**</h4>**

**<h5>**1.1.1 네트워크**</h5>**

**<h6>**1.1.1.1. 통신**</h6>**

**</body>**

**</html>**

****

<그림 3-1> 예제 2-2 결과

HTML에서는 제목을 설정하기 위해 <h1>부터 <h6>까지의 태그를 사용한다. <h1>은 가장 크고 중요한 제목을, <H6>은 가장 하순위의 제목을 표현하기 위해 사용된다.

1. **예제 2-3 title 속성으로 툴팁 달기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**툴팁 달기**</title>**

**</head>**

**<body>**

**<h1 title="h1태그로 작성하였습니다.">**

1장 홈페이지

**</h1>**

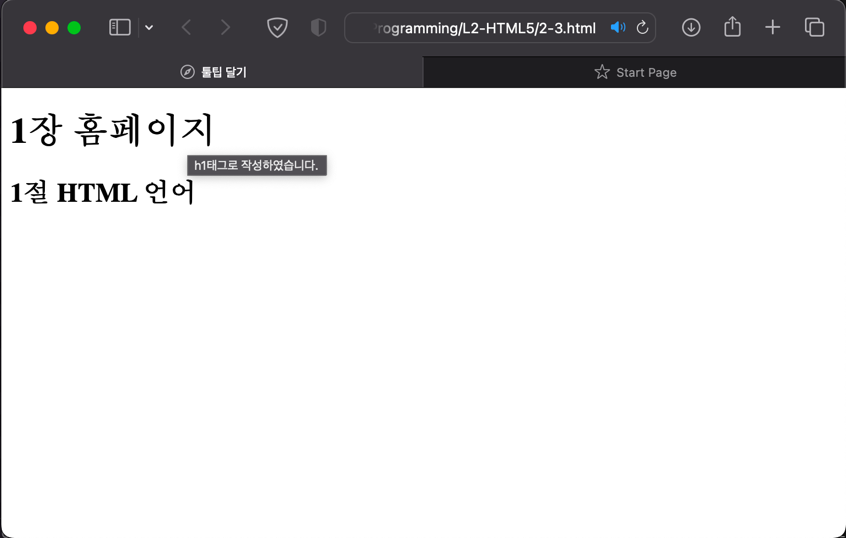
**<h2 title="h2태그로 작성하였습니다.">**

1절 HTML 언어

**</h2>**

**</body>**

**</html>**

****

<그림 4-1> 예제 2-3 결과

<h1>-<h6>태그에 ‘title=””’를 추가하면 툴팁을 달 수 있다. 툴팁이란 마우스를 글자 위로 올리면 보이는 정보로, <그림 4-1>에서 ‘<h1 title="h1태그로 작성하였습니다.">’의 내용에 따라 ‘h1태그로 작성하였습니다.’라는 툴팁을 확인할 수 있다. 뿐만 아니라 ‘1절 HTML 언어’ 에 마우스를 올려보면 ‘h2태그로 작성하였습니다.’라는 툴팁을 확인할 수 있다.

1. **예제 2-4 <p>로 단락 나누기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**단락나누기**</title>**

**</head>**

**<body>**

**<h3>**2 개의 단락 나누기**</h3>**

**<p>** HTML 문서도 본문을 여러 단락으로 나눌 수 있다.

CSS 스타일을 사용하면 단락 단위로 내여

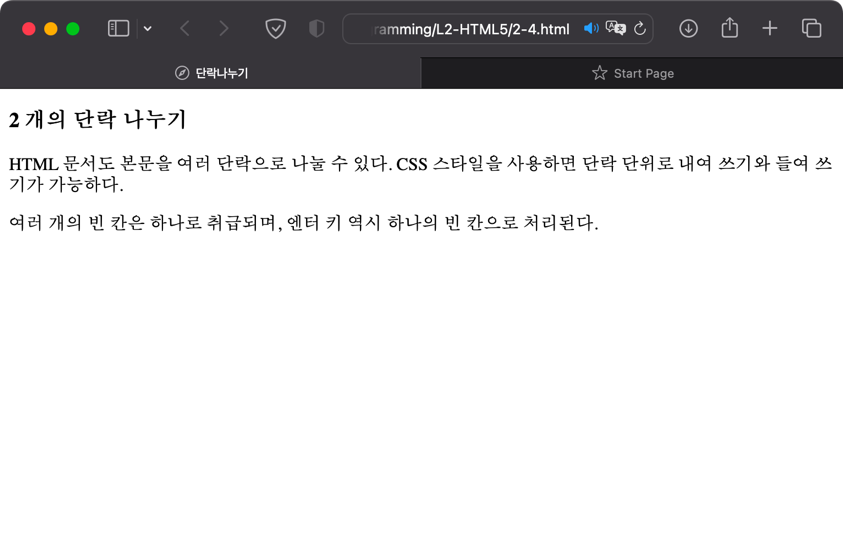
쓰기와 들여 쓰기가 가능하다. **</p>**

**<p>** 여러 개의 빈 칸은 하나로 취급되며,

엔터 키 역시 하나의 빈 칸으로 처리된다. **</p>**

**</body>**

**</html>**

****

<그림 5-1> 예제 2-4 결과

HTML에서는 엔터키를 사용해 줄 바꿈을 하여도 공백으로 인식되어 줄이 바뀌지 않는다. 따라서 <p> 태그를 사용해 단락을 나누어야 한다. <p>은 paragraph(단락)에서 명칭을 따왔으며, 위의 코드에서 엔터키를 사용하여 줄 바꿈을 하였지만 <그림 5-1>에서 보이다시피, <p>태그를 사용한 부분만 단락이 나눠진 것을 확인할 수 있다.

1. **예제 2-5 <hr> 태그로 수평선 긋기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**수평선 긋기**</title>**

**</head>**

**<body>**

**<h3>**수평선 긋기**</h3>**

**<hr>**

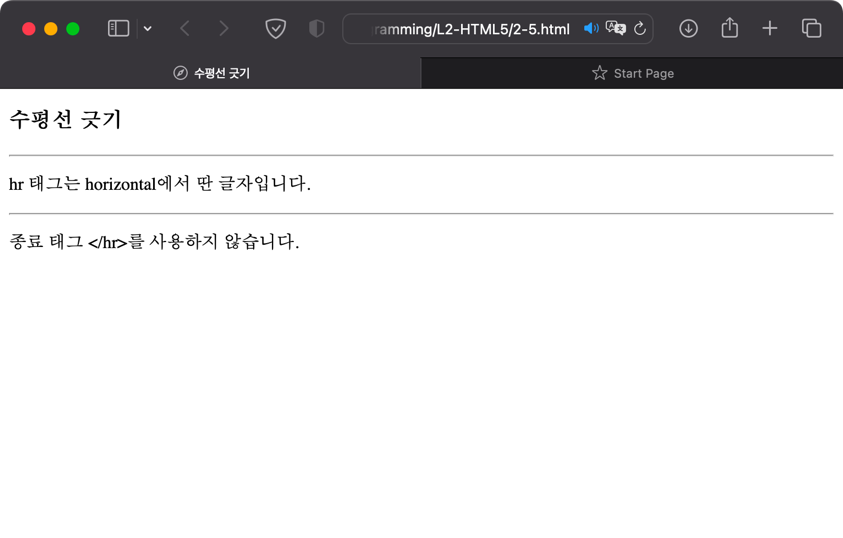
**<p>**hr 태그는 horizontal에서 딴 글자입니다.**</p>**

**<hr>**

**<p>**종료 태그 &lt;/hr&gt;를 사용하지 않습니다.**</p>**

**</body>**

**</html>**

****

<그림 6-1> 예제 2-5 결과

단락을 나눌 때, 시각적으로 줄을 그어 표현을 하여야 하는 경우가 있다. 이때 사용하는 것이 <hr>태그이다. <hr>은 horizontal(수평선)에서 명칭을 따왔으며, 수평선을 그어주는 태그이다. 이때 종료태그(</hr>)는 사용하지 않는다.

1. **예제 2-6 <br> 태그로 새로운 줄로 넘어가기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**새로운 줄 넘어가기**</title>**

**</head>**

**<body>**

**<h3>**새로운 줄 넘어가기**</h3>**

**<hr>**

&lt;br&gt; 태그로 다음 줄로 넘어갑니다.

**<br>**

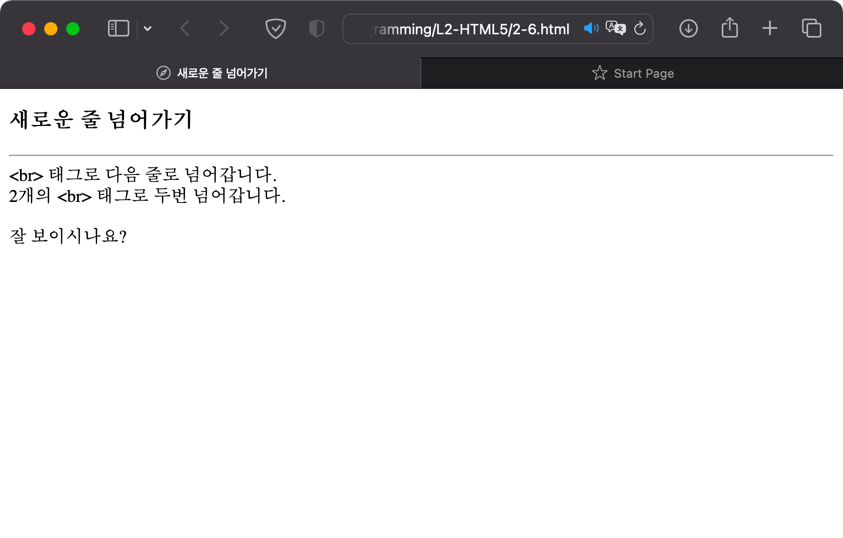
2개의 &lt;br&gt; 태그로 두번 넘어갑니다.

**<br><br>**

잘 보이시나요?

**</body>**

**</html>**

****

<그림 7-1> 예제 2-6 결과

<p>태그는 단락을 구별할 때 사용하기 때문에 <p>태그끼리 줄 바꿈이 생긴다. 하지만 단락을 구별하지 않고 줄 바꿈을 필요로 하는 경우가 생기는데, 이때 사용하는 태그가 <br>태그이다. <br>태그는 line break(줄 바꿈)에서 명칭을 따왔으며, 줄 바꿈을 할 때 사용하는 태그이다. <hr>태그와 마찬가지로 종료태그를 사용하지 않는다. <그림 7-1>에서 <br>을 연속으로 두 번 사용하여 줄 바꿈이 두 번 발생한 것을 확인할 수 있다.

1. **예제 2-7 특수 문자, 기호, 심볼 삽입**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**문자, 기호, 심볼 표현**</title>**

**</head>**

**<body>**

**<h3>**기호 넣기**</h3>**

**<hr>**

10 &divide; 2 = 5**<br />**

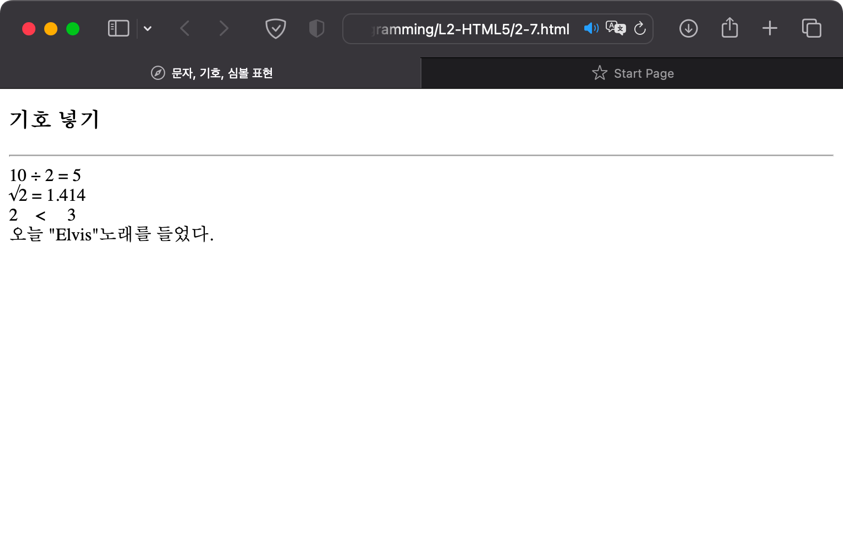
&radic;2 = 1.414**<br />**

2 &nbsp;&nbsp; &lt; &nbsp;&nbsp;&nbsp; 3**<br />**

오늘 &quot;Elvis&#34;노래를 들었다.

**</body>**

**</html>**

****

<그림 8-1> 예제 2-7 결과

HTML은 다양한 특수 문자, 기호, 심볼을 제공한다. 이런 심볼을 사용하기 위한 방법은 크게 3가지가 있다. 첫번째로 ‘&’와 ‘;’ 사이에 심볼의 이름을 적는 것이다. 해당 코드에는 ‘&divide;’, ‘&radic;’등 각 심볼의 이름을 사용하여 심볼을 표현하였고 이는 <그림 8-1>에서 확인할 수 있다. 두번째 방법으로는 ‘&#’와 ‘;’사이에 해당 심볼의 십진수 정수를 적는 것이다. 예를 들어 ‘&radic;’의 경우 ‘&#8730;’로도 동일한 결과를 얻을 수 있다. 끝으로 16진수 코드를 사용하는 방법이 있다. 16진수의 경우 ‘x00’의 형식으로 사용하므로 ’&#’와 ‘;’사이에 ‘x00’꼴의 16진수를 적으면 된다. ‘&radic;’의 경우 ‘&#x221A;’으로 표현이 가능하다.

그 외 다양한 심볼에 대해선 ‘<https://www.w3schools.com/charsets/ref_html_symbols.asp>’에서 확인이 가능하다.

1. **예제 2-8 <pre> 태그로 개발자의 포맷 그대로 출력**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**개발자의 포맷 그대로 출력**</title>**

**</head>**

**<body>**

**<h3>**개발자의 포맷 그대로 출력하기**</h3>**

**<hr>**

**<p>**

&lt;p&gt; 태그를 사용하면 여러 개의 빈칸은

하나로, 여러 줄은 한 줄에 붙여 출력됩니다.

**</p>**

**<hr>**

**<pre>**

그러나 &lt;pre&gt; 태그를 사용하면

사용자가 입력한 그대로 출력됩니다.

**</pre>**

**</body>**

**</html>**

****

<그림 9-1> 예제 2-8 결과

HTML에서는 개발 환경에서 여러 개의 빈칸, 엔터키를 통한 줄 바꿈 등을 하여도 한 개로 인식이 된다. 위의 코드에서 <p>태그 내의 내용을 <그림 9-1>과 비교해보면, <p>태그 내에서 줄 바꿈을 하여도 실행 화면에선 빈칸으로 인식되었다.

<pre>태그는 preformatted text(미리 정의된 형식)에서 명칭을 따왔으며, <pre>태그를 사용하면 개발 환경에서 작성된 내용이 그대로 출력된다. 위의 코드에서 <pre>태그 내에 있는 내용은 줄 바꿈 및 들여쓰기까지 반영되어 출력된 것을 확인할 수 있다.

1. **예제 2-9 텍스트 꾸미기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**텍스트 꾸미기**</title>**

**</head>**

**<body>**

**<h3>**텍스트 꾸미기**</h3>**

**<hr>**

**<p>**

**<b>**진하게**</b><br />**

**<strong>**중요한**</strong><br />**

**<em>**강조**</em><br />**

**<i>**이탤릭으로 강조**</i><br />**

**<b><i>**진하게 이탤릭으로 강조**</i></b><br />**

보통 문자 **<small>**한 단계 작은 문자**</small><br />**

**<del>**삭제**</del><br />**

**<ins>**추가**</ins><br />**

보통문자의 **<sup>**윗첨자**</sup><br />**

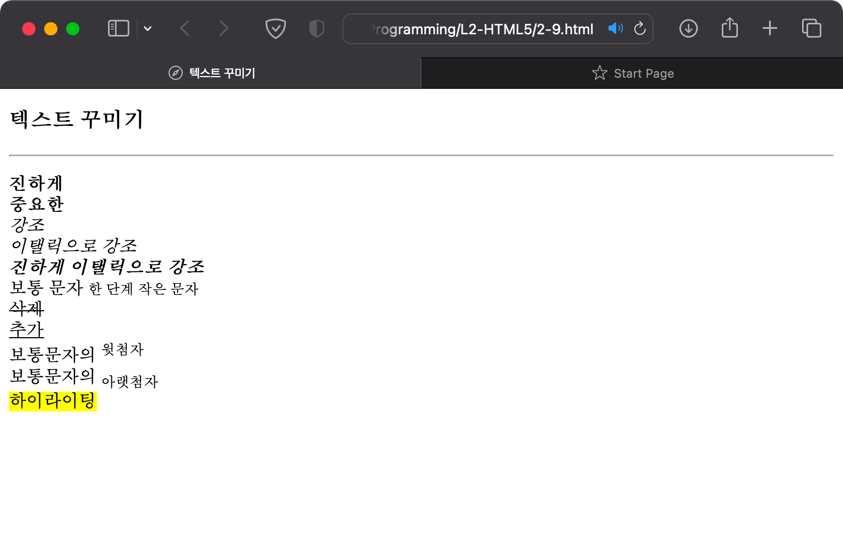
보통문자의 **<sub>**아랫첨자**</sub><br />**

**<mark>**하이라이팅**</mark><br />**

**</p>**

**</body>**

**</html>**

****

<그림 10-1> 예제 2-9 결과

HTML은 텍스트를 꾸미기 위한 여러가지 방법을 제공한다. <b>태그는 bold(굵게)에서 명칭을 따왔으며, 글자를 진하게 해준다. <b>태그는 bold(굵게)에서 명칭을 따왔으며, 글자를 진하게 해준다. <strong>태그는 이름 그대로 텍스트의 중요성을 표현할 때 사용한다. <em>태그는 emphasized text(강조된 텍스트)에서 명칭을 따왔으며, 텍스트를 강조하고자 할 때 사용한다. <i>태그는 italic(이탤릭체)에서 명칭을 따왔으며, 텍스트를 이탤릭체로 표현할 때 사용한다. <small>태그는 이름 그대로 텍스트를 한 단계 작은 문자로 표현할 때 사용한다. <del>태그는 delete(삭제)에서 명칭을 따왔으며, 텍스트 위에 취소선을 긋는 효과를 부여한다. <ins>태그는 insert(추가)에서 명칭을 따왔으며 텍스트 아래에 밑줄을 그어 추가된 텍스트라는 것을 표현할 때 사용한다. <sup>태그는 superscript(윗첨자)에서 명칭을 따왔으며, 윗첨자 텍스트를 표현할 때 사용한다. 주로 수학공식 등을 표현할 때 사용한다. <sub>태그는 subscript(아랫첨자)에서 명칭을 따왔으며, 아랫첨자 텍스트를 표현할 때 사용한다. 주로 수학공식, 화학 반응식 등을 표현할 때 사용한다. 끝으로 <mark>태그는 이름 그대로 텍스트에 형관펜으로 칠한 것처럼 마크표시를 표현할 때 사용한다.

1. **예제 2-10 <div> 블록과 <span> 인라인**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title**>&lt;div&gt;블록과 &lt;span&gt;인라인**</title>**

**</head>**

**<body>**

**<h3>**부경대학교**</h3>**

**<hr>**

**<div style="background-color: skyblue; padding: 20px;">**

부경대학교(Pukyong National University)는 부산광역시 남구에 있는

**<span style="color: red;">**대한민국의 국립대학**</span>**이다.

영문 약칭으로는

**<span style="color: red">**PKNU**</span>이다.**

**</div>**

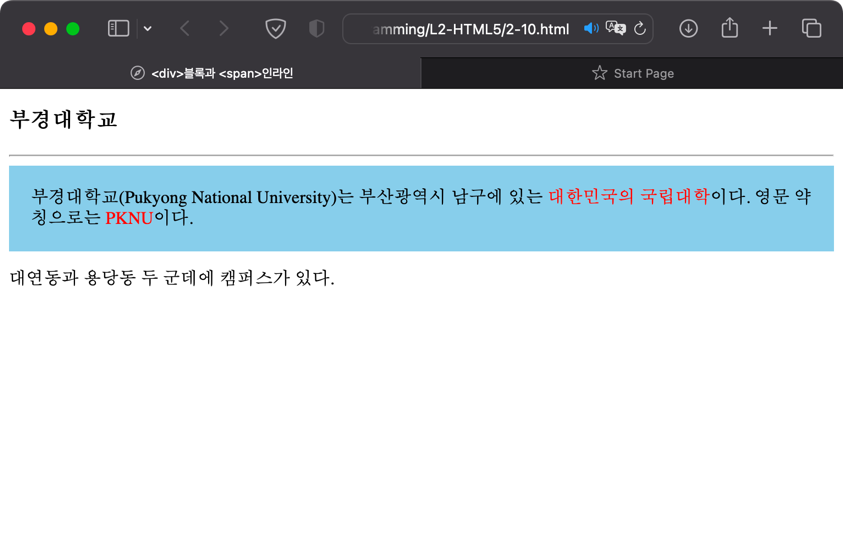
**<p>**

대연동과 용당동 두 군데에 캠퍼스가 있다**.**

**</p>**

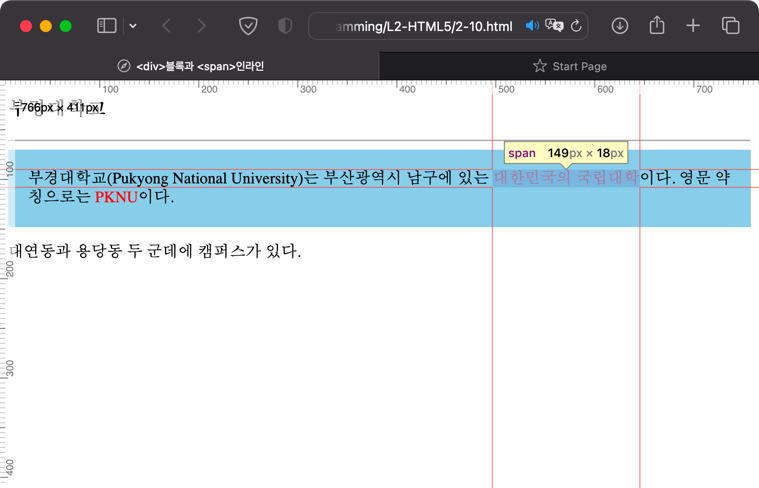
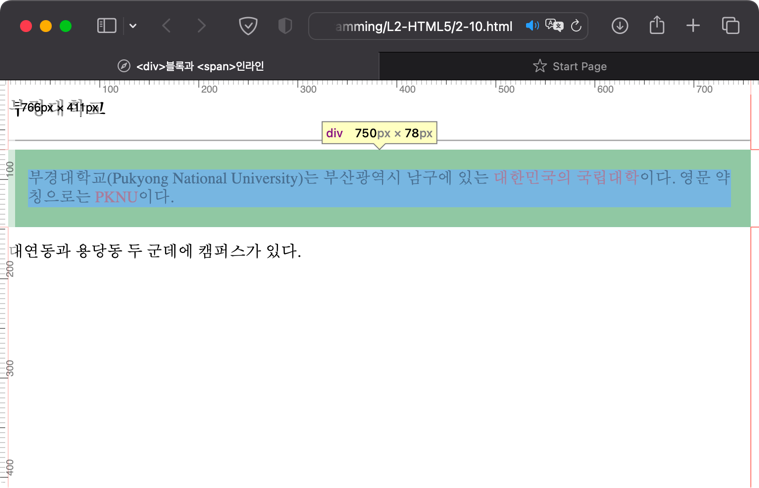
**</body>**

**</html>**

****

<그림 11-1> 예제 2-10 결과

<div>태그는 division(영역)에서 명칭을 따왔으며, 특정 구역을 정의할 때 사용한다. <그림 11-1>처럼 해당 영역의 배경색을 바꿀 수 있고 안쪽 여백을 설정할 수도 있다. <span>태그는 span(폭)에서 명칭을 따왔으며, 특정 텍스트나 요소를 묶을 때 사용한다. <그림 11-1>에서 ‘대한민국의 국립대학’ 및 ‘PKNU’가 <div>영역 내에서 하나의 인라인 타입으로 묶여 ‘color: red’를 통해 빨갛게 보이는 것을 확인할 수 있다. <그림 11-2>에서 Page Source를 통해 정의된 구역을 살펴보면 <div>태그는 블록 타입으로 묶여있는 반면, <span>태그는 인라인 요소인 텍스트만 구분되어 있는 모습을 볼 수 있다.

****

<그림 11-2> <div>태그와 <span>태그 비교

1. **예제 2-11 <img> 태그로 이미지 삽입**

**<!DOCTYPE <!DOCTYPE html>**

**<html>**

**<head>**

**<title>**이미지 삽입**</title>**

**</head>**

**<body>**

**<h3>**이미지 삽입**</h3>**

**<hr>**

**<p>**엘비스 프레슬리의 사진입니다.**</p>**

**<img src="Elvis.jpeg" width="150" height="200" alt="Elvis"/>**

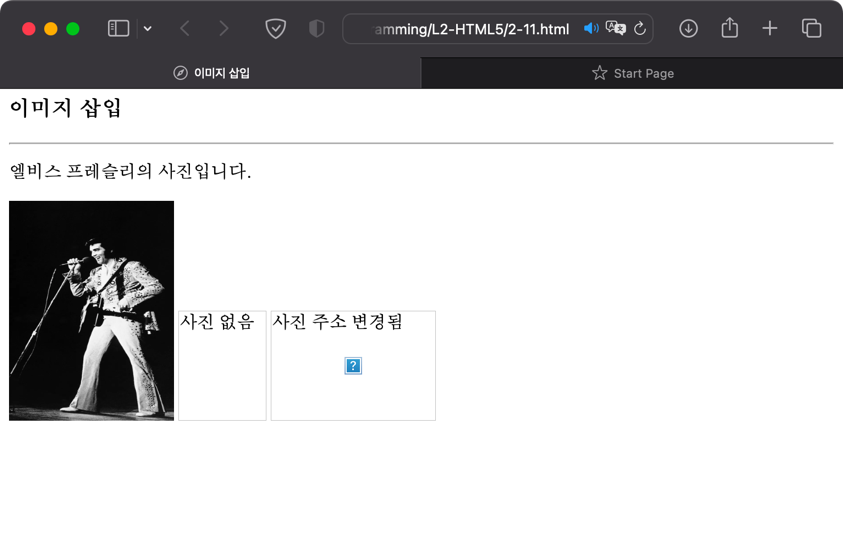
**<img src="Elvis2.jpeg" width="80" height="100" alt="사진 없음" />**

**<img src="http://img.naver.net/nmms\_224940510.gif" width="150"**

**height="100" alt="사진 주소 변경됨" />**

**</body>**

**</html>**

****

<그림 12-1> 예제 2-12 결과

<img>태그는 HTML문서에 이미지(image)를 삽입할 때 사용하는 태그이다. 다만, HTML문서에 직접 삽입되는 것이 아니라 작성된 링크를 통해 이미지가 표현되는 것이다. 따라서 <img>태그를 사용하기 위해선 ‘src’속성을 사용하여 이미지의 위치를 명시해야 한다. 또한 ‘alt’속성을 사용하여 이미지 표현에 실패한 경우에 띄울 메시지를 명시해야 한다. <그림 12-1>에서 ‘Elcis.jpeg’는 성공적으로 불러왔지만, ‘Elvis2/jpeg’는 경로에 그림 파일이 없어서 ‘alt’속성으로 명시해둔 ‘사진 없음’ 메시지가 나타남을 확인할 수 있다. ‘src’속성으로 웹페이지의 사진을 불러 올 수도 있는데 ‘http://img.naver.net/nmms\_ 224940510.gif’에 사진이 없으므로 역시 ‘alt’속성으로 명시해둔 ‘사진 주소 변경됨’ 메시지가 나타남을 확인할 수 있다. 그림의 크기는 ‘width’속성와 ‘heigit’속성으로 조절이 가능하다.

1. **예제 2-12 <ol>로 라면 끓이는 순서 나열**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**라면을 끓이는 순서**</title>**

**</head>**

**<body>**

**<h3>**라면을 끓이는 순서**</h3>**

**<hr>**

**<ol type="A">**

**<li>**물을 끓인다.**</li>**

**<li>**라면과 스프를 넣는다.</**li>**

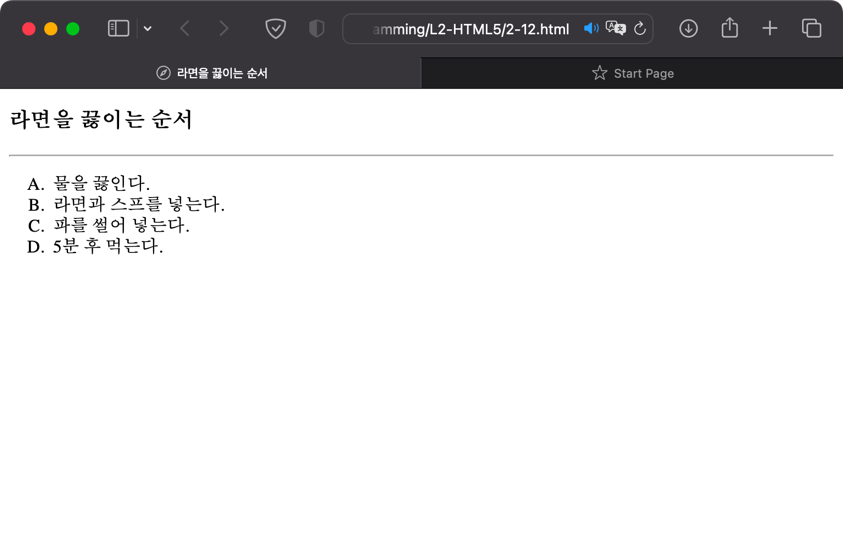
**<li>**파를 썰어 넣는다.</**li>**

**<li>**5분 후 먹는다.</**li>**

**</ol>**

**</body>**

**</html>**

****

<그림 13-1> 예제 2-12 결과

<ol>태그는 ordered list(순서 목록)에서 명칭을 따왔으며, 이를 사용하면 텍스트의 리스트를 ‘type’속성으로 지정해둔 형식에 맞게 목록으로 표현할 수 있다. 이때 <li>태그를 함께 사용하며 <그림 13-1>에서 ‘type’속성을 ‘A’로 지정했기에 A, B, C.. 순으로 목록이 표현됨을 확인할 수 있다. ‘type’속성을 ‘1’로 하면 1, 2, 3.. 순으로 목록이 표현된다.

1. **예제 2-13 <ul>로 좋아하는 음식 나열**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**내가 좋아하는 음식**</title>**

**</head>**

**<body>**

**<h3>**내가 좋아하는 음식**</h3>**

**<hr>**

**<ul>**

**<li>**감자탕**</li>**

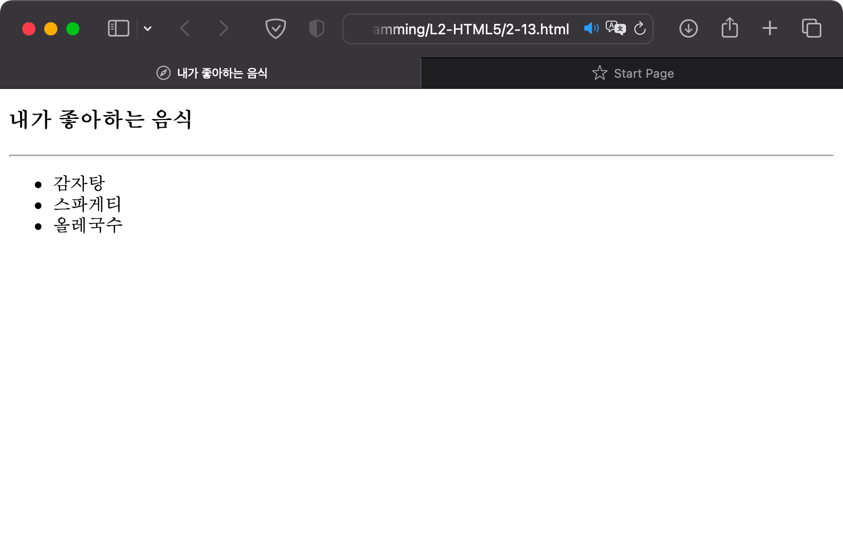
**<li>**스파게티**</li>**

**<li>**올레국수**</li>**

**</ul>**

**</body>**

**</html>**

****

<그림 14-1> 예제 2-13 결과

<ul>태그는 unordered list(비순서 목록)에서 명칭을 따왔으며, 이를 사용하면 텍스트의 목록을 순서표기 없이 표현할 수 있다. 사용법은 ‘type’속성 지정을 제외하곤 <ol>태그와 동일하며, <그림 14-1>에서 <ul>태그를 사용하여 ‘•’로 목록이 표현됨을 확인할 수 있다.

1. **예제 2-14 중첩 리스트 만들기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**중첩 리스트 만들기**</title>**

**</head>**

**<body>**

**<h3>**내가 사는 이유**</h3>**

**<hr>**

**<ul>**

**<li>**내가 좋아하는 음식 많아요

**<ul>**

**<Li>**감자탕**</Li>**

**<li>**제주고기국수**</li>**

**</ul>**

**</li>**

**<li>**라면 먹기 좋아해요

**<ol type="1">**

**<li>**물을 끓인다.</**li>**

**<li>**라면과 스프를 넣는다.</**li>**

**<li>**파를 썰어 넣는다.**</li>**

**</ol>**

**</li>**

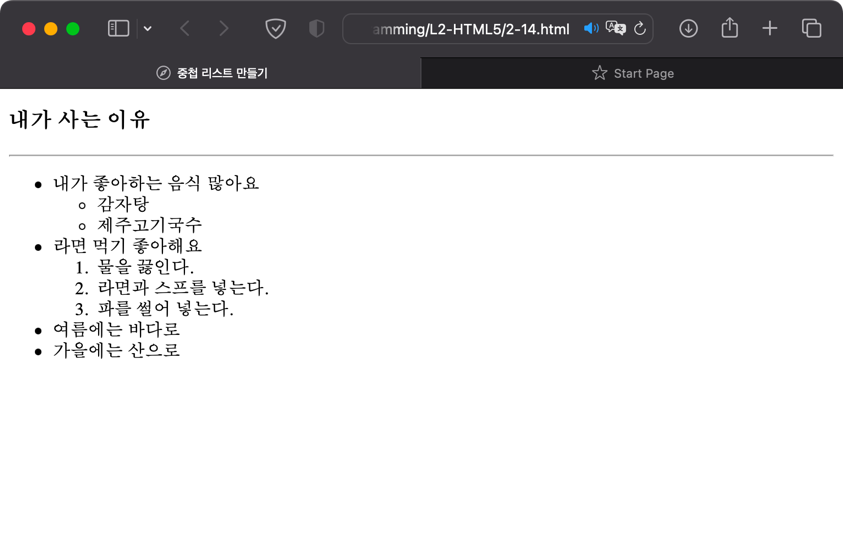
**<li>**여름에는 바다로**</li>**

**<li>**가을에는 산으로**</li>**

**</ul>**

**</body>**

**</html>**

****

<그림 15-1> 예제 2-14 결과

‘예제 2-14’는 <ul>태그와 <ol>태그를 중첩하여 사용하는 예제이다. 여기서 확인할 수 있는 사실은 <ul>태그는 ‘type’속성을 지정하지 않는 대신, 하위 목록을 표현할 때, ‘•’, ‘○’, ‘■’순으로 이어진다는 것이다. ‘■’이후로는 계속하여 ‘■’가 생성된다.

1. **예제 2-15 정의 리스트 만들기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**정의 리스트**</title>**

**</head>**

**<body>**

**<h3>**웹 브라우저 종류**</h3>**

**<hr>**

**<dl>**

**<dt><strong>Internet Explorer</strong></dt>**

**<dd>**마이크로소프트에서 만든 브라우저로 현재 국내 시장에서 가장 많이 사용**</dd>**

**<dt><strong>Firefox</strong></dt>**

**<dd>**Mozilla 재단에서 오픈 소스로 만든 것으로 W3C의 웹 표준을 선도**</dd>**

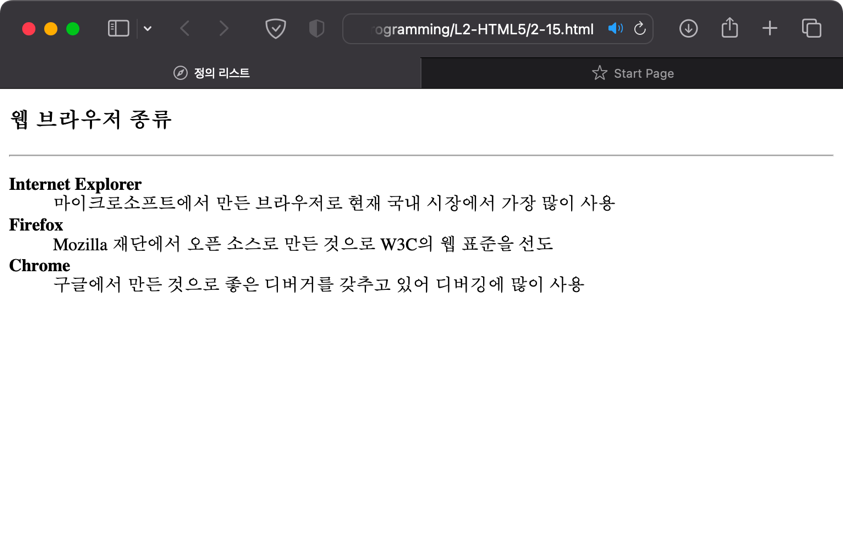
**<dt><strong>Chrome</strong></dt>**

**<dd>**구글에서 만든 것으로 좋은 디버거를 갖추고 있어 디버깅에 많이 사용**</dd>**

**</dl>**

**</body>**

**</html>**

****

<그림 16-1> 예제 2-15 결과

<dl>태그는 description list(설명 목록)에서 명칭을 따왔으며, 용어에 대한 설명을 할 때 주로 사용한다. <dl>태그의 하위 태그로는 <dt>태그와 <dd>태그가 있으며, 각각 description tern(설명 용어), description description(설명)에서 명칭을 따왔다. <dl>태그로 목록을 만들고 <dt>태그로 용어를 표현한 다음 <dd>로 용어에 대한 내용을 표현한다. <그림 16-1>을 살펴보면 <dl>태그로 리스트를 만들고 <dt>태그와 <strong>태그를 사용하여 ‘Internet Explorer’를 표현하고 <dd>태그를 통해 ‘마이크로소프트에서~가장 많이 사용’을 표현한다. 이때, <dd>태그를 사용하면 자동으로 들여쓰기가 적용된다.

1. **예제 2-16 기본 표 만들기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**기본 테이블 만들기**</title>**

**</head>**

**<body>**

**<h3>**기본 구조를 가진 표**</h3>**

**<hr>**

**<table border="1">**

**<caption>**1학기 성적**</caption>**

**<thead>**

**<tr><th>**이름**</th><th>**HTML**</th><th>**CSS**</th></tr>**

**</thead>**

**<tfoot>**

**<tr><th>**합**</th><th>**255**</th><th>**230**</th></tr>**

**</tfoot>**

**<tbody>**

**<tr><td>**홍길동**</td><td>**80**</td><td>**70**</td></tr>**

**<tr><td>**아무개**</td><td>**95**</td><td>**99**</td></tr>**

**<tr><td>**화이팅**</td><td>**40**</td><td>**61**</td></tr>**

**</tbody>**

**</table>**

**</body>**

**</html>**

****

<그림 17-1> 예제 2-16 결과

<table>태그는 이름 그대로 table(표)을 표현하는 태그이다. 하위 태그로 <tr>, <th>, <td>를 포함하고 있으며 각각 table row(표 열), table header cell(표 헤더 셀), teble data cell(표 내용 셀)을 의미한다. <thead>태그와 <tbody>, <tfoot>태그를 사용할 수도 있는데, <thead>태그는 표 내에서 가장 상단에 위치하여 해당 열의 목록을 지정할 때 사용한다. <tbody>는 표 내부 내용을 표현할 때 사용하며, <tfoot>은 표의 가장 하단 행을 지정할 때 사용한다. <그림 17-1>에서 ‘이름’, ‘HTML’, ‘CSS’는 <thead>태그에 속하며 <tr>태그로 각각의 열이 지정되고 <th>태그로 표 헤더 셀로 표현되었다. ‘합’, ‘255’, ‘230’은 <tfoot >태그에 속하며 <tr>로 열이 지정되고 <th>태그로 표 헤더 셀로 표현되었다. <tbody>태그 내부의 내용들은 <tr>태그로 열이 지정되고 <td>태그로 내용 셀로 표현되었다. 예제 2-16에서 <caption>태그도 사용되었는데 이는 테이블이나 사진 등에 캡션을 달 때 사용하는 태그이다. <그림 17-1>에서 ‘1학기 성적’이 <caption>태그를 사용하여 표 상단 캡션으로 표현된 것을 확인할 수 있다. 끝으로 <table>태그에 ‘border’속성을 사용하여 표 레이아웃 테두리를 지정할 수 있다. 아무것도 작성하지 않거나 0으로 지정하면 테두리가 없고 ‘1’로 지정되면 테두리가 사용된다. 다만, 이는 HTML5에서는 지원하지 않는 방식이다.

1. **예제 2-17 이미지를 가지는 표 만들기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**표에 이미지 삽입**</title>**

**</head>**

**<body>**

**<h3>**표에 이미지 삽입**</h3>**

**<hr>**

**<table>**

**<caption>**좋아하는 과일**</caption>**

**<tbody>**

**<tr>**

**<td><img src="media/apple.jpg" height="100"></td>**

**<td><img src="media/banana.jpg" height="100"></td>**

**<td><img src="media/mango.jpg" height="100"></td>**

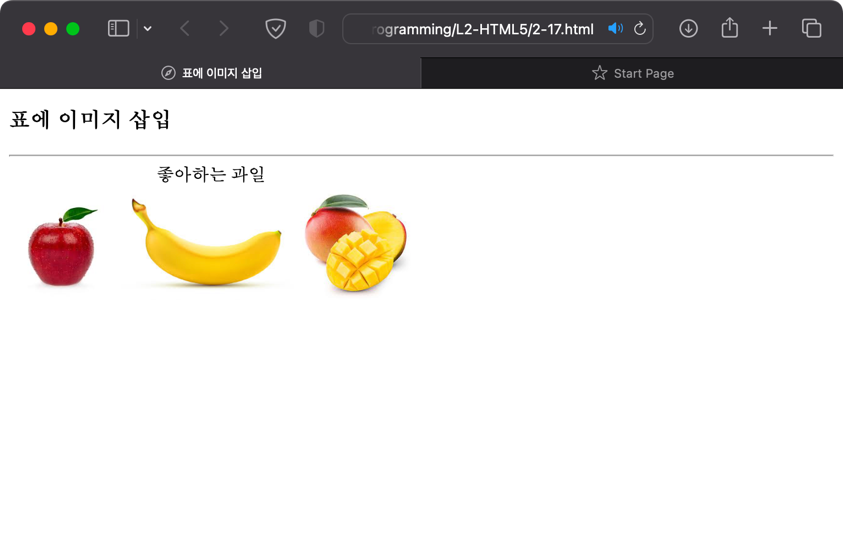
**</tr>**

**</tbody>**

**</table>**

**</body>**

**</html>**

****

<그림 18-1> 예제 2-17 결과

‘예제 2-11’과 ‘예제 2-16’에서의 내용을 바탕으로 이미지를 표로 표현하는 예제이다. <td>태그 내에 <img>태그를 사용하여 이미지 또한 표 내용으로 표현할 수 있다. 이후 사용법은 앞과 동일하다.

1. **예제 2-18 하이퍼링크 만들기**

**<!DOCTYPE html>**

**<html>**

**<head><title>**링크 만들기**</title></head>**

**<body>**

**<h3>**링크 만들기**</h3>**

**<hr>**

포털 사이트

**<ul>**

**<li><a href="http://www.naver.com">**네이버**</a></li>**

**<li><a href="http://www.daum.com">**다음**</a></li>**

**</ul>**

미디어 사이트

**<ul>**

**<li><a href="http://www.etnews.com"><img src="media/iconetnews.png"**

**height=40 alt="전자신문"></a></li>**

**<li><a href="http://www.cnn.com"><img src="media/iconcnn.png"**

**height=40 alt="CNN"></a></li>**

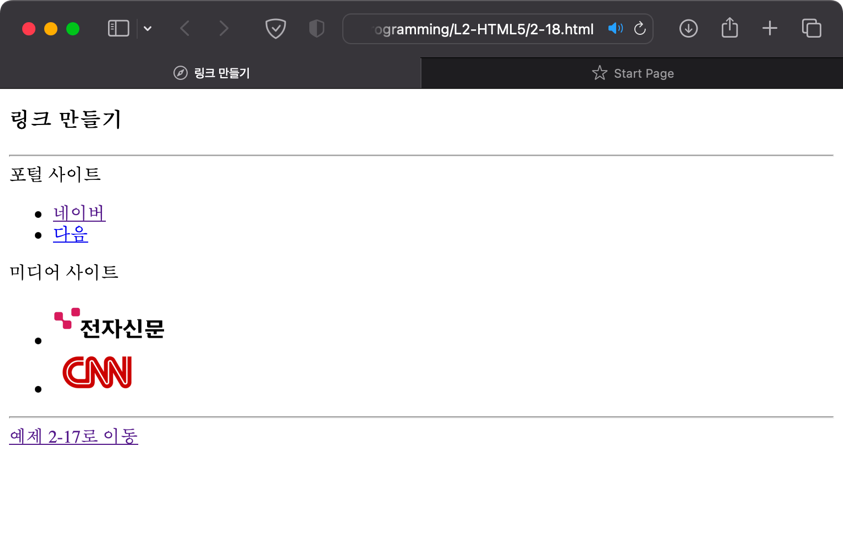
**</ul>**

**<hr>**

**<a href="2-17.html">**예제 2-17로 이동**</a>**

**</body>**

**</html>**

****

<그림 19-1> 예제 2-18 결과

<a>태그는 하이퍼링크를 달기 위해 사용한다. Hyperlink reference를 의미하는 ‘href’속성을 통해 링크의 목적지를 지정할 수 있으며 <a>태그와 </a>태그 사이에서 표현된 글이나 그림을 클릭하여 ‘href’속성으로 지정해둔 링크로 이동할 수 있다. <그림 19-1>에서 ‘포털사이트’의 경우 ‘네이버, ‘다음’의 글로 하이퍼링크가 걸려있고 ‘미디어 사이트’의 경우 각각의 그림들로 하이퍼링크가 걸려있는 모습을 확인할 수 있다. 뿐만 아니라 HTML문서와 같은 폴더에 위치한 또다른 HTML문서로도 이동이 가능한데, ‘예제 2-17로 이동하기’의 경우, 같은 폴더 내에 있는 ‘2-17.html’문서로 이동할 수 있다.

1. **예제 2-19 링크의 target 속성 활용**

**<!DOCTYPE html>**

**<html>**

**<head><title>**링크의 target 속성 활용**</title></head>**

**<body>**

**<h3>**링크의 target 속성 활용**</h3>**

**<hr>**

**<ul>**

**<li><a href="http://www.w3.org" target="\_blank">**W3C(새 윈도우, \_blank)**</a>**

**<li><a href="http://www.etnews.com" target="\_self">**전자신문(현재 윈도우, \_self)**</a>**

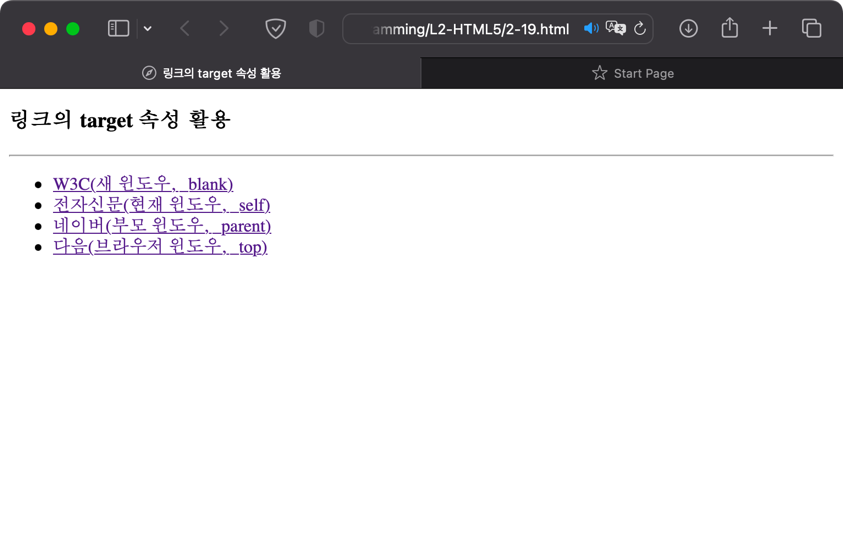
**<li><a href="http://www.naver.com" target="\_parent">**네이버(부모 윈도우, \_parent)**</a>**

**<li><a href="http://www.daum.net" target="\_top">**다음(브라우저 윈도우, \_top**)</a>**

**</ul>**

**</body>**

**</html>**

****

<그림 20-1> 예제 2-19 결과

하이퍼링크의 ‘target’속성을 사용하는 예제이다. ‘target’속성에는 ‘\_blank’, ‘\_self’, ‘\_parent’, ‘\_top’이 있으며 ‘\_blank’는 새 창으로 하이퍼링크를 연다. ‘\_self’는 현재 보고 있는 창에서 하이퍼링크를 연다. ‘\_parent’는 부모 프레임에서 창을 연다. 예를 들어, 현재 보고 있는 ‘2-19.html’페이지를 창 A를 통해서 열었다면 ‘\_parent’속성으로 되어있는 하이퍼링크는 창 A에서 열리게 된다. 끝으로 ‘\_top’은 최상위 프레임에서 창을 연다. 창 A를 통해 창 B를 열고, 창 B를 통해 ‘2-19.html’페이지를 연 다음 ‘\_top’속성으로 되어있는 하이퍼링크를 열면 창 A에서 열리게 된다.

1. **예제 2-20 앵커로 이동하는 링크 만들기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**링크 만들기**</title>**

**</head>**

**<body>**

**<h3>**Elvis Presley**</h3>**

**<hr>**

**<ul>**

**<li><a href="#love">**Love me tender**</a></li>**

**<li><a href="#can">**Can't help falling in love**</a></li>**

**<li><a href="#it">**It's now or never**</a></li>**

**</ul>**

**<h3 id="love">**love me tender**</h3>**

Love me tender, love me sweet, Never let me go**<br>**

You have made my life complete**<br>**

And I love you so**<br>**

**<h3 id="can">**Can't help falling in love**</h3>**

Love me tender, love me sweet, Never let me go**<br>**

You have made my life complete, And I love you so**<br>**

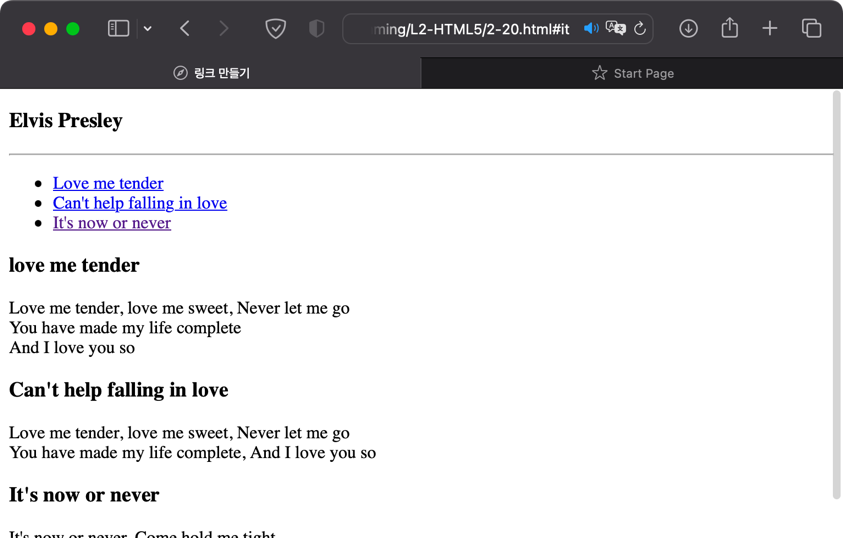
**<h3 id="it">**It's now or never**</h3>**

It's now or never, Come hold me tight**<br>**

Kiss me my darling, Be mine tonight**<br>**

**</body>**

**</html>**

****

<그림 21-1> 예제 2-20 결과

<a>태그를 사용한 하이퍼링크는 주소를 통하여 새로운 창을 열 수도 있지만 해당 문서의 특정 위치로 이동하는 경우에도 사용한다. 이동하고자 하는 태그에 ‘id’속성을 사용하여 이름을 지정한 다음, ‘href’속성에서 해당 이름 앞에 ‘#’을 붙이면 지정한 이름이 있는 태그로 이동한다. <그림 21-1>에서 하이퍼링크로 지정되어 있는 ‘Love me tender’를 클릭하면 ‘id’속성으로 지정한 ‘#love’를 통해 ‘<h3 id="love">’로 스크롤이 이동한다.

1. **예제 2-21 파일 다운로드 링크 만들기**

**<!DOCTYPE html>**

**<html>**

**<head><title>**파일 다운로드**</title></head>**

**<body>**

**<h3>**파일 다운로드**</h3>**

**<hr>**

**<ul>**

**<li><a href="media/Elvis.jpg" download>**엘비스 이미지 다운로드**</a>**

**<li><a href="media/chapter9.pdf" download>**chapter9.pdf 다운로드 **</a>**

**<li><a href="media/EmbraceableYou.mp3" download>**mp3 다운로드**</a>**

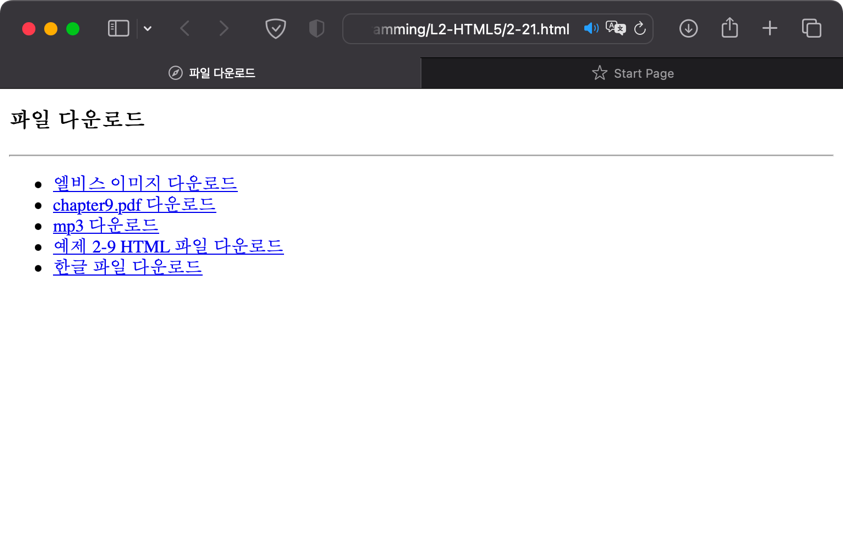
**<li><a href="2-9.html" download>**예제 2-9 HTML 파일 다운로드**</a>**

**<li><a href="media/test.hwp" download>**한글 파일 다운로드**</a>**

**</ul>**

**</body>**

**</html>**

****

<그림 22-1> 예제 2-21 결과

<a>태그를 사용하여 파일을 다운로드하는 링크를 만들 수 있다. ‘download’속성을 사용하면 ‘href’속성으로 지정해둔 위치의 파일을 다운로드 하는 하이퍼링크를 만들 수 있다. ‘download=”*filename*”’으로 지정하면 *filename*으로 지정한 파일을 다운 받을 수 있고, 만약 이를 생략할 경우 ‘href’속성으로 지정해둔 위치의 파일을 다운로드한다.

1. **예제 2-22 2개의 인라인 프레임을 가진 웹 페이지**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**첫번째 iframe**</title>**

**</head>**

**<body>**

**<h4>**첫번째 iframe**</h4>**

**</body>**

**</html>**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**두번째 iframe**</title>**

**</head>**

**<body>**

**<h4>**두번째 iframe**</h4>**

**</body>**

**</html>**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**iframe을 가지는 웹 페이지**</title>**

**</head>**

**<body>**

**<h3>**2 개의 &lt;iframe&gt;을 가집니다.**</h3>**

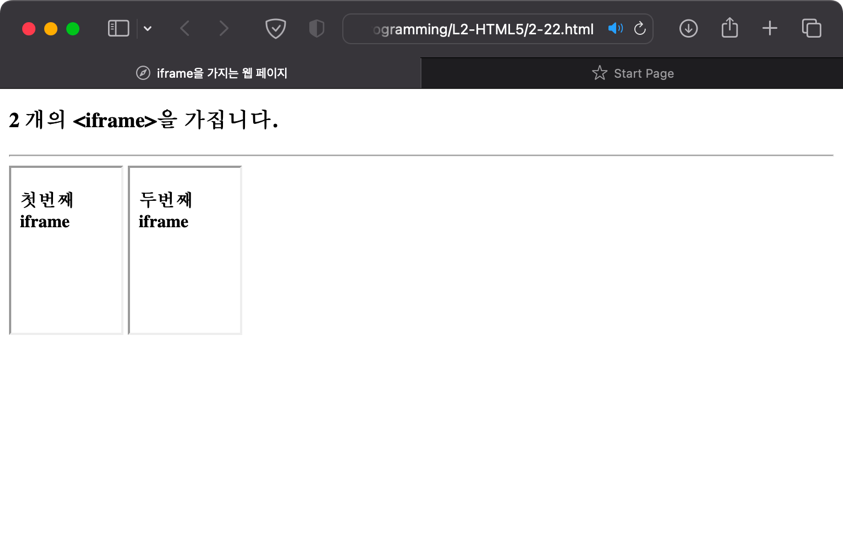
**<hr>**

**<iframe src="iframe1.html" width="100" height="150"></iframe>**

**<iframe src="iframe2.html" width="100" height="150"></iframe>**

**</body>**

**</html>**

****

<그림 23-1> 예제 2-22 결과

<iframe>태그는 inline frame(인라인 프레임)에서 명칭을 따왔으며, 현재 문서에서 다른 문서를 포함할 때 사용한다. ‘src’속성을 통해 포함하고자 하는 문서를 지정할 수 있다. <그림 23-1>에서 <iframe>태그를 사용하여 각각 ‘iframe1.html’, ‘iframe2.html’ 문서를 인라인 프레임으로 불러온 것을 확인할 수 있다. 해당 프레임은 ‘width’속성과 ‘height’속성으로 크기를 조절할 수 있다.

1. **예제 2-23 <iframe>으로 2개의 사이트 내장하기**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<meta charset="utf-8" />**

**<title>**iframe을 이용한 사이트**</title>**

**</head>**

**<body>**

**<h3>**2 개의 사이트입니다.**</h3>**

**<hr>**

**<iframe src="https://www.gnu.org/" width="200" height="300">**

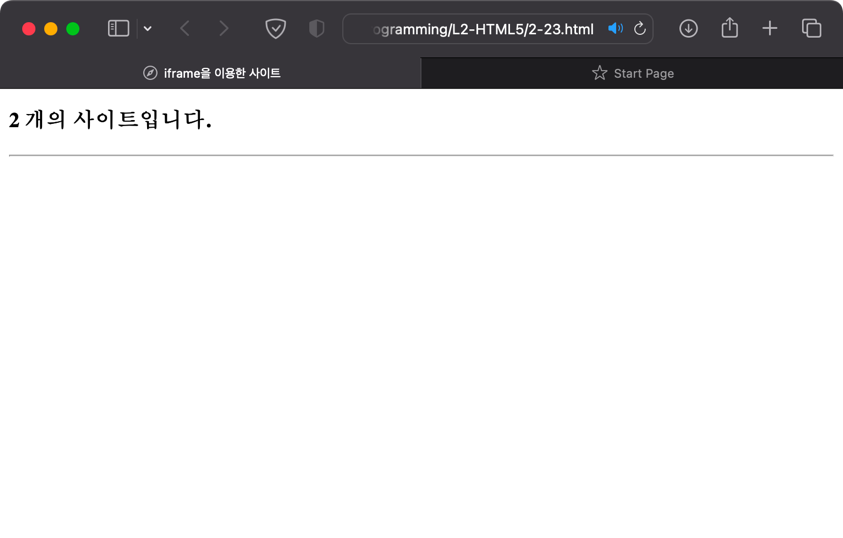
**</iframe>**

**<iframe src="https://en.wikipedia.org/" width="200" height="300">**

**</iframe>**

**</body>**

**</html>**

****

<그림 24-1> 예제 2-23 결과

<iframe>태그를 사용하면 현재 위치에 있는 HTML문서 뿐만 아니라 외부 웹사이트를 인라인 프레임으로 표현할 수 있다. ‘src’속성에서 해당 웹사이트의 주소를 지정하면 그 웹사이트가 인라인 프레임으로 표현된다. 다만 <iframe>태그를 막아 둔 웹사이트가 많기에 주의해야한다. <그림 24-1>에서 확인할 수 있듯, ‘https://www.gnu.org/’와 ‘https://en.wikipedia.org/’의 주소가 수업 때는 잘 표현되었으나 지금 현재는 <iframe>태그에 대해 막혀서 표현되지 않은 모습을 확인할 수 있다.

1. **예제 2-24 <video> 태그로 비디오 삽입**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**비디오 삽입**</title>**

**</head>**

**<body>**

**<h3>**비디오 삽입**</h3>**

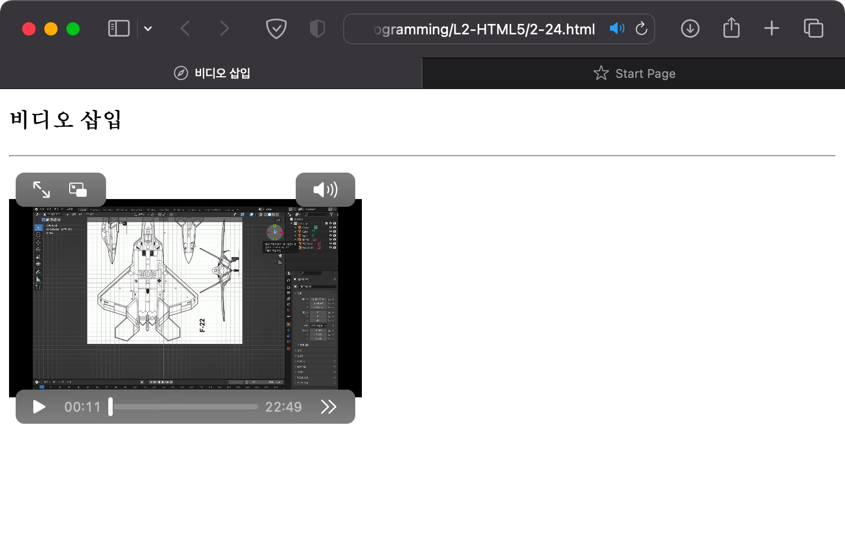
**<hr>**

**<video src="media/video.mp4" width="320" height="240" controls autoplay>**

브라우저가 video 태그를 지원하지 않습니다.**</video>**

**</body>**

**</html>**

****

<그림 25-1> 예제 2-24 결과

<video>태그는 <img>태그와 사용법이 동일하다. 차이점이라면 <img>태그는 이미지를 표현하는 태그이고, <video>태그는 비디오를 표현하는 태그이다. 또한, <video>태그는 여러가지 속성을 가지는데 그중 ‘controls’속성은 비디오의 플레이 버튼 등과 같은 제어기를 표현하기 위해 사용한다. ‘autoplay’ 속성은 해당 페이지가 열리면 비디오가 자동으로 재생됨을 명시한다. <그림 25-1>에서 ‘media’ 폴더 내에 있는 ‘video.mp4’가 페이지에 표현되는 것을 확인할 수 있다. 명시된 속성에 따라 ‘video.mp4’를 제어하는 제어기가 표시되며 창을 열었을 때 비디오가 자동으로 재생되는 것을 확인하였다. 만약 브라우저가 <video>태그를 지원하지 않는다면 지정해둔 텍스트인 ‘브라우저가 video 태그를 지원하지 않습니다.’ 문구가 뜬다.

1. **예제 2-25 <audio> 태그로 오디오 삽입**

**<!DOCTYPE html>**

**<html>**

**<head>**

**<title>**오디오 삽입**</title>**

**</head>**

**<body>**

**<h3>**오디오 삽입**</h3>**

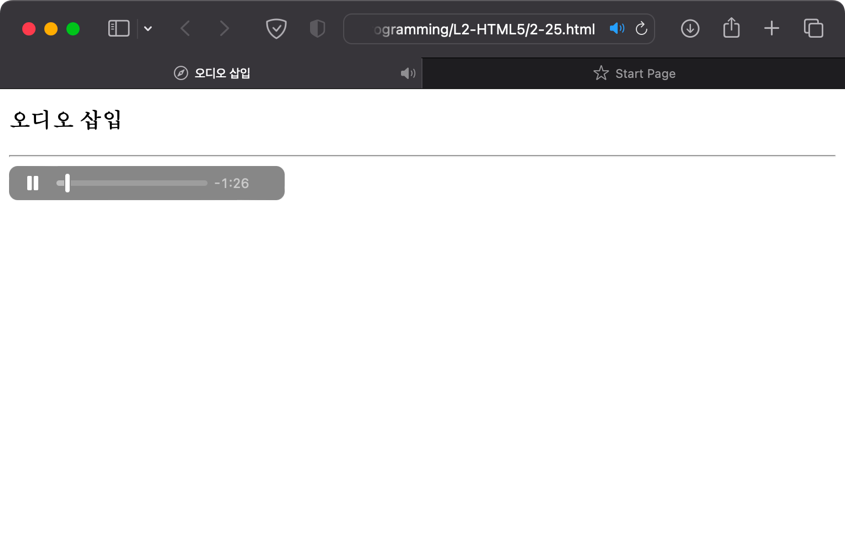
**<hr>**

**<audio src="media/audio.mp3" controls autoplay loop>**

브라우저가 audio 태그를 지원하지 않습니다.</**audio>**

**</body>**

**</html>**

****

<그림 26-1> 예제 2-25 결과

<audio>태그는 오디오를 표현하는 태그로 <video>태그와 사용법이 동일하다. 겹치는 부분은 생략하고 추가적인 속성을 설명하자면, ‘loop’속성은 해당 오디오의 재생이 끝나면 또다시 재생하겠다는 것을 의미한다. 만약 브라우저가 <audio>태그를 지원하지 않는다면 지정해둔 텍스트인 ‘브라우저가 audio 태그를 지원하지 않습니다.’ 문구가 뜬다. <그림 26-1>에서 ‘media’ 폴더 내에 있는 ‘audio.mp3’가 페이지에 표현되는 것을 확인할 수 있다.

1. **참고문헌**

* W3schools, (n.d.), HTML Tutorial, <https://www.w3schools.com/html/default.asp>
* TCPSCHOOL, (n.d.), “HTML 태그 레퍼런스”, <http://www.tcpschool.com/html-tags/intro>