

2.5 CSS3 레이아웃

레이아웃이란 웹 사이트에서 정보와 내용을 보기 편하게 한 화면에 나누고 디자인하는 것이다. 어떤 사용자는 메뉴가 오른쪽에 있는 것이 편리 할 수 있고, 또 다른 사용자는 메뉴가 위쪽에 있는 것이 편리 할 수 있다. 이렇게 성향과 목적에 따라 다양한 레이아웃이 있다. 그래서 블로그나 웹사이트를 만들때 가장 먼저 기획하는 것이 **화면 레이아웃**이다.

레이아웃을 디자인할 때는 전체 페이지의 너비를 정한 후 콘텐츠의 특징에 따라 단을 나눈다. 데스크톱 일 경우에는 모니터의 해상도를 고려하기 때문에 너비가 고정된 레이아웃을 많이 사용한다. 최근에는 모니터의 해상도가 높아짐으로써 웹페이지의 너비가 넓어지고 더 많은 다양한 정보를 한눈에 볼 수 있도록 디자인해야 한다. 해상도가 낮은 작은 화면에서는 웹페이지의 너비가 좁아져도 스크롤없이 편하게 정보를 볼 수 있도록 레이아웃을 설계해야 한다. 또한 화면의 크기가 아주 작은 모바일 사용자의 환경에 맞는 레이아웃도 설계해야 한다. 따라서 다양한 해상도에 대응할 수 있는 레이아웃이 필요하다.

2.5.1 크로스 브라우징

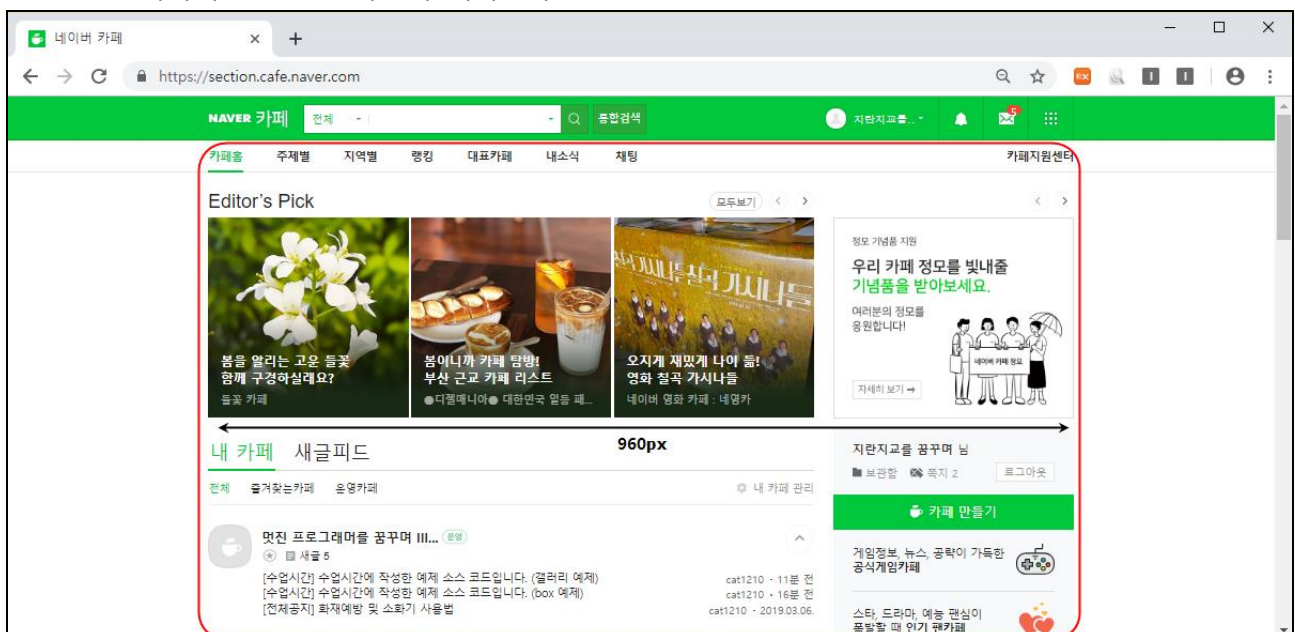
크로스 브라우징(Cross Browsing)이란 사용자의 기기나 시스템이 각각 달라도 최대한 비슷하게 표현하도록 공통 요소를 사용해서 웹 페이지를 개발하는 것이다. 웹 표준 기술을 이용해서 특정 기종(또는 특정 브라우저)에만 최적화되지 않도록 하는 것이다. 웹을 이용하는 사용자들의 OS가 다르고 글꼴이 다르며 HTML을 렌더링(Rendering) 하는 엔진이 다르기 때문에 모든 웹 브라우저에서 100% 똑같이 보이게 하는 것이 불가능하다. 또한 사용자도 비장애인, 장애인, 어린이, 노약자가 존재하며 모든 사용자가 인터넷의 정보로부터 소외되지 않도록 웹 표준 기술로 개발해야 한다.

2.5.2 레이아웃의 종류

레이아웃은 웹페이지의 영역을 Header, Footer, Menu, Content로 나누는 것이다. 레이아웃의 종류로는 고정형 레이아웃, 유동형 레이아웃, 반응형 레이아웃이 있다.

고정형 레이아웃

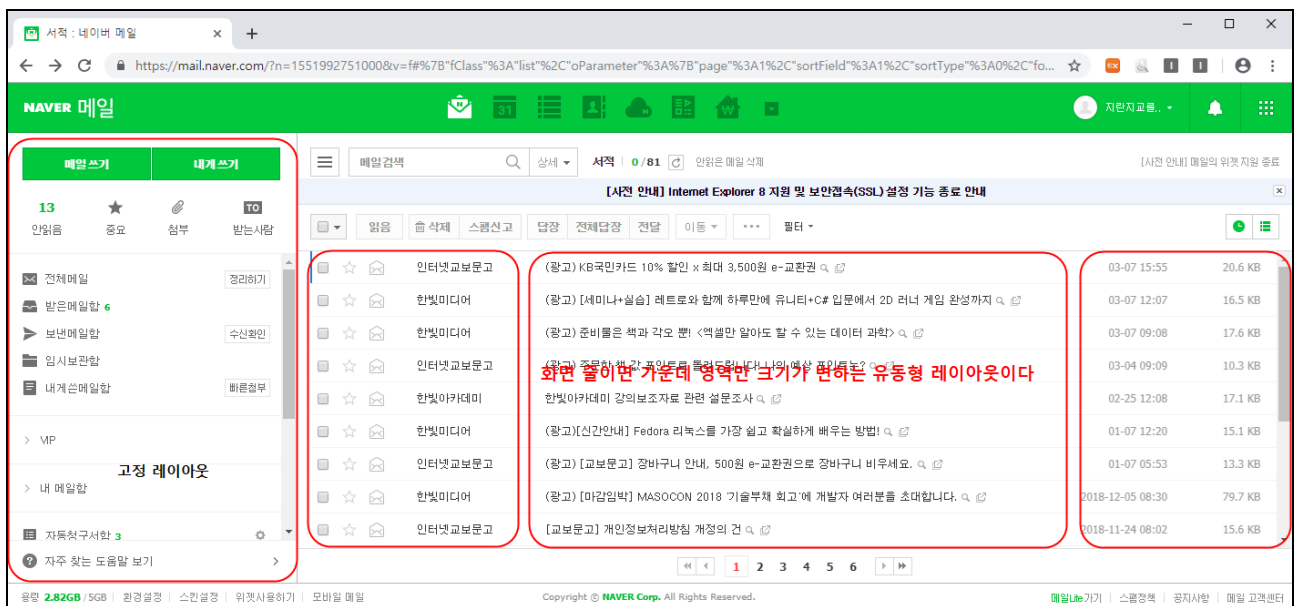
고정형 레이아웃이란 전체 페이지의 너비를 정하고 콘텐츠에 따라 단을 나누어 정렬하는 방식이다. 데스크톱 위주의 웹페이지에서 모니터 해상도를 고려해 너비가 고정된 경우에 사용한다. 고정형 레이아웃은 고정된 너비에 콘텐츠를 가운데 배치한다.



고정형 레이아웃은 텍스트의 영역을 좁게 만들어서 가독성을 좋게 한다. 또한 여백을 짜임새 있게 구성하고 콘텐츠에 포함할 내용을 표현하기 때문에 디자인하기 쉽다.

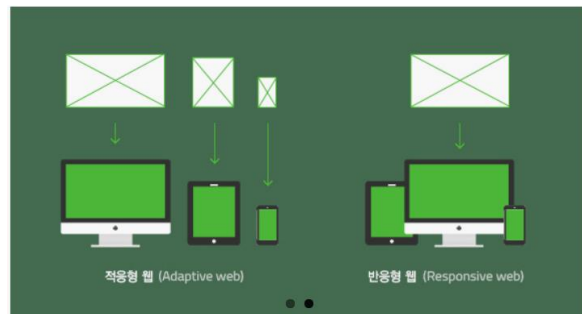
유동형 레이아웃

유동형 레이아웃은 어떤 해상도의 모니터라도 좌우 공백 없이 콘텐츠 영역이 모니터 화면을 다 채우도록 하는 방식이다. 웹페이지의 너비를 픽셀 단위로 고정된 값으로 설정하지 않고 비율로 설정하기 때문에 창 너비에 따라 콘텐츠 크기가 늘어나거나 줄어드는 레이아웃이다. 대부분 화면 해상도와 기기들이 잘 맞을 뿐만 아니라 사용자의 스크롤을 상대적으로 줄일 수 있다. 하지만 이런 레이아웃은 텍스트가 좌우로 펼쳐져 있어서 글을 읽을 때 아주 불편할 수 있다. 유동형 레이아웃은 화면을 줄였을 때 콘텐츠도 같이 주어지기 때문에 내용도 위 아래로 늘어난다. 이미지가 있는 경우 범위를 벗어나면 화면이 틀어질 수 있어서 유동형 레이아웃은 이미지가 없는 페이지에 주로 사용된다. 그래서 요즘에는 유동형 레이아웃도 일정 범위까지만 줄어 들고 설정범위 이내로 줄어들면 고정형 레이아웃으로 전환하도록 사용한다.



반응형 레이아웃

반응형 레이아웃이란 기기에 따라 사용자 화면에 맞게 레이아웃이 변하는 것을 말한다. 최근 인터넷을 이용하는 기기는 모니터, 태블릿, 모바일 등 다양하다. 그런데 모바일 화면은 매우 작기 때문에 가독성에 더 신경 써야 한다. **반응형 레이아웃은 화면 해상도에 따라 가로 폭이나 배치를 자동으로 변경해서 가독성을 높이는 것이다.** 하나의 웹페이지가 다양한 종류의 기기 또는 창, 너비 등을 사용자 환경에 맞춰 자동으로 바꾸는 레이아웃이다. 반응형 레이아웃으로 홈페이지를 구축하면 데스크톱, 태블릿, 모바일에서 최적화된 화면을 볼 수 있다. 반응형 레이아웃은 여러 기기에 이용하는 웹페이지를 하나로 운영할 수 있다. 그러나 하나의 화면에 많은 양의 콘텐츠를 표현할 때에는 적합하지 않다.



반응형 레이아웃이 적합한 사이트는 고정적인 형태의 콘텐츠를 갖고 있는 정적인 사이트이다. 예를 들어 커뮤니티, 블로그, 회사소개 페이지 등이 있다.

2.5.3 CSS 레이아웃 속성

레이아웃에는 요소들의 출력 형태나 위치를 설정하는 속성들이 있다.

overflow

overflow 속성은 박스 모델 안의 내용이 박스의 범위를 초과할 때 어떤 형태로 출력할지를 결정한다.

[overflow의 속성]

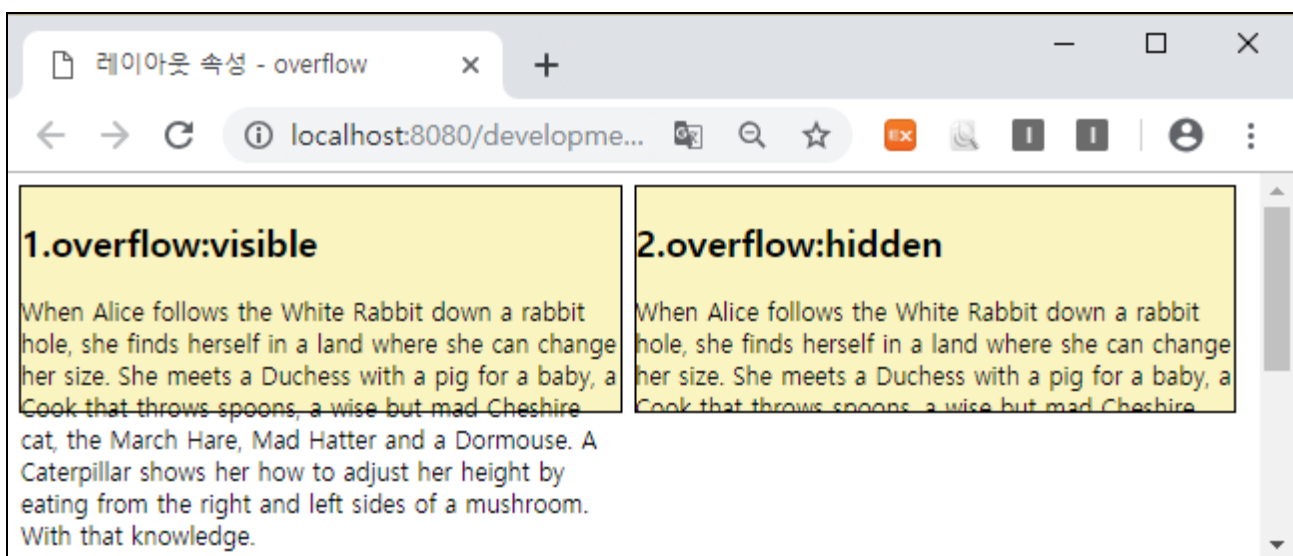
속성	속성값	설명
overflow	visible(기본값)	박스 범위를 초과하는 내용도 그대로 표현한다
	scroll	우측에 스크롤바를 이용하여 위, 아래로 이동하며 내용을 확인한다
	hidden	박스를 벗어나는 내용 모두 숨긴다.
	auto	내용이 박스를 벗어날 때에만 스크롤바가 생기도록 한다

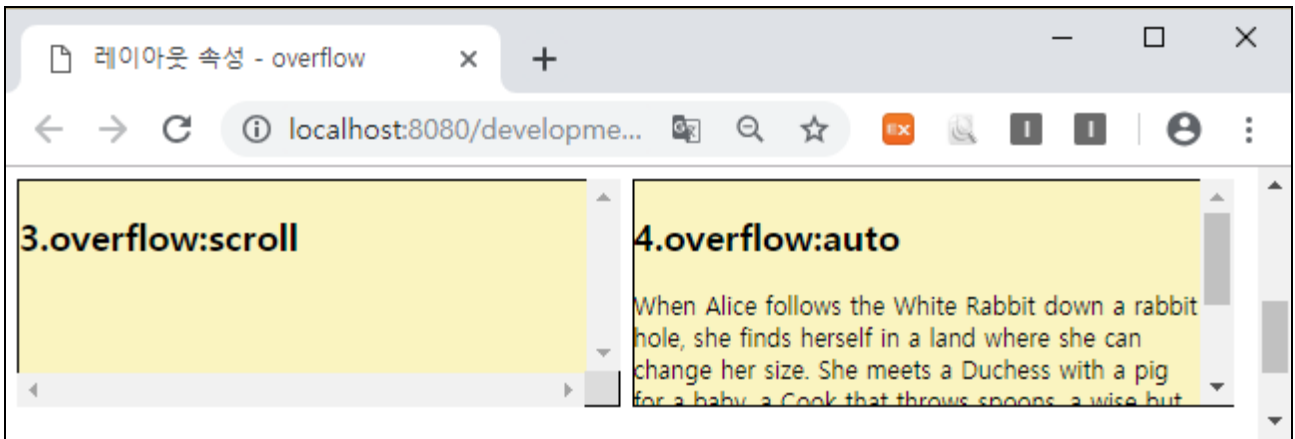
exam30_overflow.html
<pre><!DOCTYPE html> <html> <head> <meta charset="UTF-8"> <title>레이아웃 속성 - overflow</title> <style type="text/css"> #visible, #hidden, #scroll, #auto { width:400px; height:150px; background-color:#FAF4C0; border: 1px solid black; margin-bottom: 100px; margin-right:8px; float: left; } #visible{overflow: visible; } #hidden{overflow: hidden; } #scroll{overflow: scroll; } #auto{overflow: auto;} </style> </head> <body> <div id="visible"> <h2>1.overflow:visible</h2> <p>When Alice follows the White Rabbit down a rabbit hole, she finds herself in a land where she can change her size. She meets a Duchess with a pig for a baby, a Cook that throws spoons, a wise but mad Cheshire cat, the March Hare, Mad Hatter and a Dormouse. A Caterpillar shows her how</pre>

```

        to adjust her height by eating from the right and left sides of a mushroom.
        With that knowledge.</p>
    </div>
    <div id="hidden">
        <h2>2.overflow:hidden</h2>
        <p>When Alice follows the White Rabbit down a rabbit hole, she finds
        herself in a land where she can change her size. She meets a Duchess with
        a pig for a baby, a Cook that throws spoons, a wise but mad Cheshire cat,
        the March Hare, Mad Hatter and a Dormouse. A Caterpillar shows her how
        to adjust her height by eating from the right and left sides of a mushroom.
        With that knowledge</p>
    </div>
    <div id="scroll">
        <h2>3.overflow:scroll</h2>
    </div>
    <div id="auto">
        <h2>4.overflow:auto</h2>
        <p>When Alice follows the White Rabbit down a rabbit hole, she finds
        herself in a land where she can change her size. She meets a Duchess with
        a pig for a baby, a Cook that throws spoons, a wise but mad Cheshire cat,
        the March Hare, Mad Hatter and a Dormouse. A Caterpillar shows her how
        to adjust her height by eating from the right and left sides of a mushroom.
        With that knowledge</p>
    </div>
</body>
</html>

```





display와 visibility

HTML은 <div>, <h>와 같은 블록 요소와 , <a>와 같은 인라인 요소가 있다. 이 요소들은 각각 display 속성을 가지고 있으며 inline 또는 block으로 변경이 가능하다. visibility는 요소가 보이거나 안보이게 하는 속성이다. display: none과 비슷하지만 약간의 차이가 있다. display: none은 메모리에 존재하지 않기 때문에 보이지 않는다. 하지만 visibility:hidden은 메모리에 존재는 하지만 hidden으로 처리하는 것이다.

[display와 visibility의 속성]

속성	속성값	설명
display		해당 HTML 요소가 웹브라우저에 어떻게 표현할지를 결정한다. 요소의 종류를 블록 또는 인라인을 바꾸는 속성이다. 웹페이지 레이아웃을 결정하는 중요한 속성 중 하나이다.
	inline	새로운 라인에서 시작하지 않고 너비도 해당 내용만큼만 영역을 확보한다
	block	줄바꿈한 후 새로운 라인의 전체 영역(너비)를 확보한다
	inline-block	inline을 block처럼 만들어서 바로 옆에 나열한다.
	none	박스를 생성하지 않는다. 화면에 보이지 않는다
visibility		요소를 보이게 할지 숨기게 할지 선택하는 속성이다.
	visible	기본값. 요소를 보여준다.
	hidden	요소를 보여주지 않는다
	collapse	테이블 요소에서만 사용할 수 있다.

[블록과 인라인 종류]

태그	설명
블록 요소(block)	<div>, <h>, <p>, , , <form> 등
인라인 요소(inline)	, , <a>, <label> 등

exam31_display.html
<pre><!DOCTYPE html> <html> <head> <meta charset="UTF-8"></pre>

```

<title>display - 블록과 인라인</title>
<style type="text/css">
    span{
        width: 100px; height: 100px;
        display: none;
        /* display: inline; */
        /* display: block; */
        /* display: inline-block; */
    }
    #html {
        background-color:#FFE400;
    }
    #css {
        background-color:#B5B2FF;
        /* visibility: hidden; */
    }
    #js {
        background-color:#D9418C;
    }
</style>
</head>
<body>
    웹 표준은 <span id="html">HTML</span>구조,
    <span id="css">CSS</span>디자인,
    <span id="js">JavaScript</span>동작이 있습니다.
</body>
</html>

```



[display: inline;]

웹 표준은 HTML구조, CSS디자인, JavaScript동작이 있습니다.

[display: block;]

웹 표준은

HTML

구조,

CSS

디자인,

JavaScript

동작이 있습니다.

[display: inline-block;]

웹 표준은 HTML 구조, CSS 디자인, JavaScript 동작이 있습니다.

position과 z-index

position은 CSS에서 특정 요소의 위치를 지정하는 속성이다. z-index를 이용해서 요소를 앞쪽으로 배치하거나 뒤쪽으로 배치할 수 있다.

[position과 z-index의 속성]

속성	속성값	설명
position	레이아웃을 배치하거나 요소의 위치를 정할 때 사용하는 속성이다.	
	static	무조건 요소를 순서대로 배치한다
	relative	직전 요소의 상대 위치를 배치한다
	absolute	문서(또는 상위 요소) 내에서 절대 위치에 배치한다
	fixed	현재 보이는 브라우저 화면 내에 절대(고정) 위치에 배치한다.
z-index	어느 요소가 앞에 나올지 배치 순서를 정하는 순서이다. position 속성을 적용한 요소에서만 동작한다.	
	auto number	값이 높을수록 앞쪽에 배치하고 값이 작을수록 뒤쪽에 배치한다

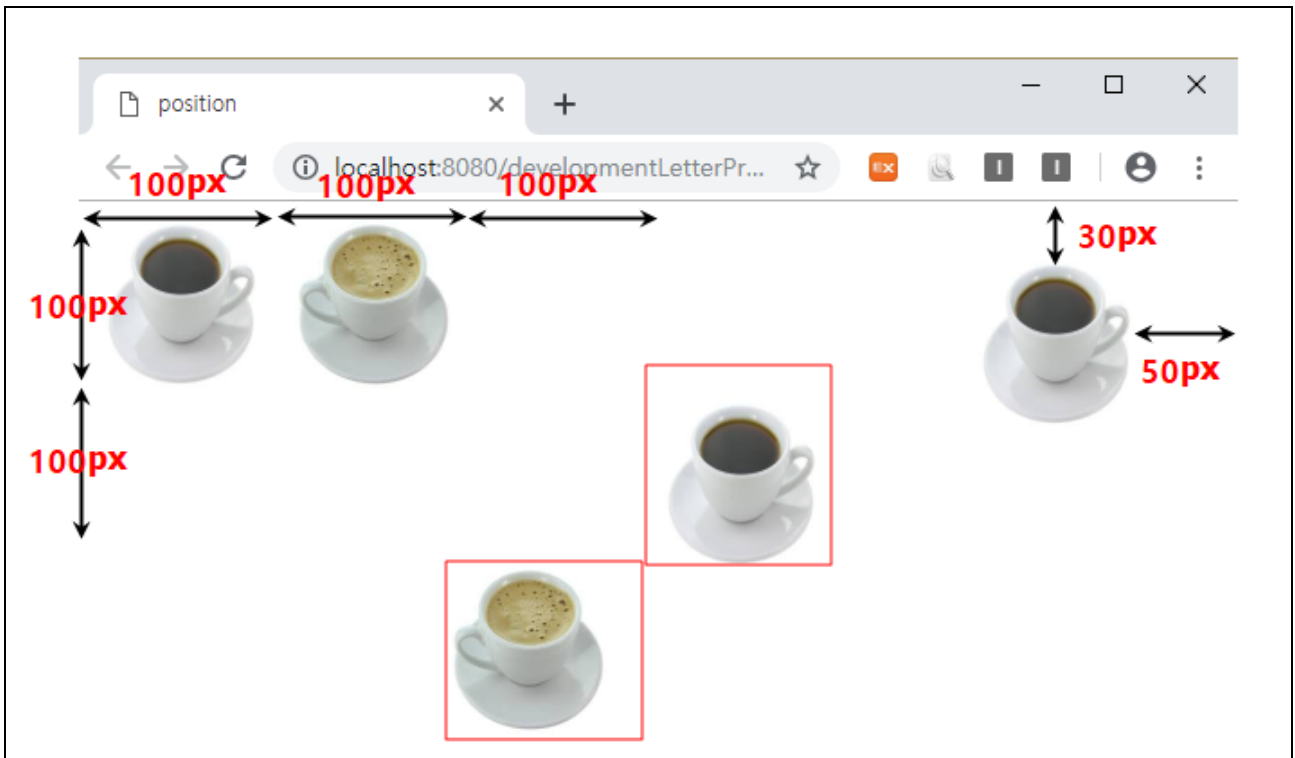
exam32_positionindex.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>position과 z-index 속성</title>
    <style type="text/css">
      .box{
        width:100px; height:100px;
        position:absolute;
      }
      .box:nth-child(1) {
        background-color:red;
        left:10px; top:10px;
        z-index:1;
      }
      .box:nth-child(2) {
        background-color:green;
        left:50px; top:50px;
        z-index:2;
      }
      .box:nth-child(3) {
        background-color:blue;
        left:90px; top:90px;
        z-index:3;
      }
    </style>
  </head>
  <body>
    <div>
      <div class="box"></div>
      <div class="box"></div>
      <div class="box"></div>
    </div>
  </body>
</html>
```




exam33_position.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>position</title>
    <style type="text/css">
      .pos1{position: static;}
      .pos2{position: static; top:100px; left: 150px;}
      .pos3{position: relative; top:100px; left: 100px;}
      .pos4{position: absolute; top:200px; left: 200px;}
      .pos5{position: fixed; top:30px; right: 50px;}
    </style>
  </head>
  <body>
    
    
    
    
    
  </body>
</html>
```



float와 clear

float 속성은 본문 중간에 이미지 넣고자 하거나 문서의 위치를 설정하고자 할 때 사용한다. clear는 float를 적용해 놓은 요소를 다시 원래의 요소로 적용하고자 할 때 사용한다

[float와 clear의 속성]

속성	속성값	설명
float		본문 중간에 이미지 넣고자 하거나 문서의 위치를 설정하고자 할 때 사용한다. 요소를 오른쪽 또는 왼쪽에 배치한다
	none	플로팅 박스 적용없이 순서대로 배치한다
	left	플로팅 박스를 왼쪽 경계에 배치하며 주변 콘텐츠가 오른쪽에 나타난다
	right	플로팅 박스를 오른쪽 경계에 배치하며 주변 콘텐츠가 왼쪽에 나타난다
clear		요소의 옆을 취소한다.
	left	왼쪽 요소를 취소한다.
	right	오른쪽 요소를 취소한다
	both	모든 float(left, right)을 취소한다.

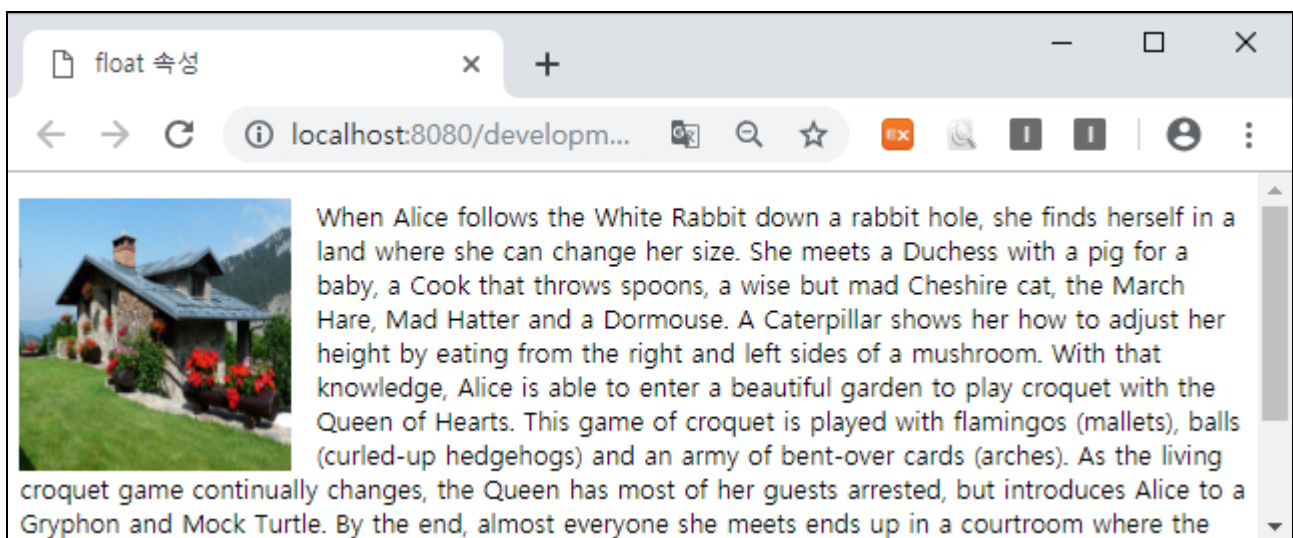
exam34_float.html
<pre> <!DOCTYPE html> <html> <head> <meta charset="UTF-8"> <title>float 속성 </title> <style type="text/css"> </pre>

```

        img {
            float: left;
            width:170px; height:170px;
            margin-right:15px;
        }
    </style>
</head>
<body>
    <p>
        
        When Alice follows the White Rabbit down a rabbit hole, she finds herself in a land
        where she can change her size. She meets a Duchess with a pig for a baby, a Cook
        that throws spoons, a wise but mad Cheshire cat, the March Hare, Mad Hatter and a
        Dormouse. A Caterpillar shows her how to adjust her height by eating from the right
        and left sides of a mushroom. With that knowledge, Alice is able to enter a beautiful
        garden to play croquet with the Queen of Hearts. This game of croquet is played
        with flamingos (mallets), balls (curled-up hedgehogs) and an army of bent-over cards
        (arches). As the living croquet game continually changes, the Queen has most of
        her guests arrested, but introduces Alice to a Gryphon and Mock Turtle.

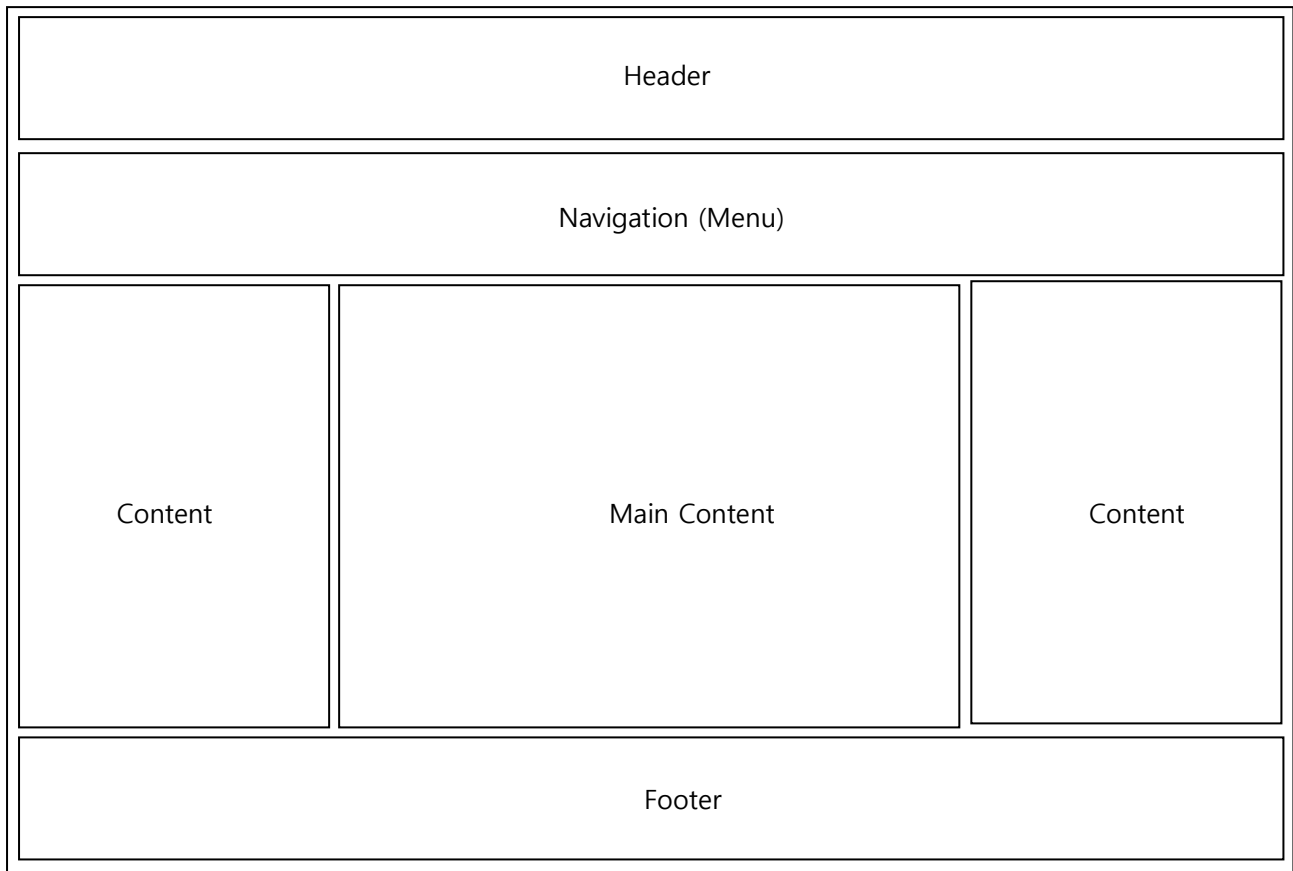
        By the end, almost everyone she meets ends up in a courtroom where the King of
        Hearts wants a verdict before the evidence, jurors pay attention to all the wrong
        information, witnesses are threatened with punishment and Alice grows tall enough
        not to be intimidated by a pack of cards, which is what the court and soldiers are.
        Suddenly Alice wakes up and realizes that her entire adventure was an amazing
        dream.</p>
    </body>
</html>

```



레이아웃 실습

레이아웃 구성요소들을 규칙적으로 배치하면서도 주목성, 가독성, 창조성, 명쾌성, 조형성, 등을 요구한다. 따라서 시각적인 효과를 높이기 위해 질서 있는 레이아웃을 만들어야 한다.



모바일 브라우저는 화면이 작기 때문에 보통 1단으로 레이아웃을 처리하며, 태블릿과 노트북은 2단으로 레이아웃을 처리한다. 데스크톱의 화면만 유일하게 3단으로 처리한다.



먼저 데스크톱에서 주로 사용하는 고정형 레이아웃으로 페이지를 구성해 보자.

```
fixed_width_layout.html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>고정형 레이아웃</title>
    <link href="https://fonts.googleapis.com/css?family=Single+Day&display=swap"
```

```

rel="stylesheet">
    <style type="text/css">
        #container {
            width: 960px;
            margin: 0px auto;
            padding: 20px;
            border: 1px solid #5D5D5D; }
        header {
            font-family: 'Single Day', cursive;
            padding: 20px; margin-bottom: 20px;
            border: 1px solid #5D5D5D; }
        section {
            width: 600px; padding: 20px;
            margin-bottom: 20px;
            float: left;
            border: 1px solid #5D5D5D; }
        aside {
            width: 260px; height:230px; padding: 20px;
            margin-bottom: 20px;
            float: right;
            border: 1px solid #5D5D5D; }
        aside a{text-decoration: none;}
        footer {
            clear: both;
            padding: 20px; border: 1px solid #5D5D5D; }

    </style>
</head>
<body>
    <div id="container">
        <header>
            <h1>동화 속 이야기</h1>
        </header>
        <section>
            <article>
                <h2>내용</h2>
                <p>When Alice follows the White Rabbit down a rabbit
                hole, she finds herself in a land where she can change her
                size. </p>
                <p>She meets a Duchess with a pig for a baby, a Cook
                that throws spoons, a wise but mad Cheshire cat, the
                March Hare, Mad Hatter and a Dormouse. A Caterpillar

```

```

                                shows her how to adjust her height by eating from the
                                right and left sides of a mushroom. With that
                                knowledge,</p>
                                </article>
</section>
<aside>
    <h2>도서 선택</h2>
    <ol>
        <li><a href="#">Alice's Adventures in Wonderland</a> </li>
        <li><a href="#">The Adventures of Tom Sawyer</a> </li>
        <li><a href="#">Anne of Green Gables</a> </li>
        <li><a href="#">Les Miserables</a> </li>
    </ol>
</aside>
<footer>
    <p> Copyright &copy; 2018 0000, Inc. All rights reserved.
    Contact webmaster for more information. 02-000-0000 </p>
</footer>
</div>
</body>
</html>

```

동화 속 이야기

내용

When Alice follows the White Rabbit down a rabbit hole, she finds herself in a land where she can change her size.

She meets a Duchess with a pig for a baby, a Cook that throws spoons, a wise but mad Cheshire cat, the March Hare, Mad Hatter and a Dormouse. A Caterpillar shows her how to adjust her height by eating from the right and left sides of a mushroom. With that knowledge,

도서 선택

1. [Alice's Adventures in Wonderland](#)
2. [The Adventures of Tom Sawyer](#)
3. [Anne of Green Gables](#)
4. [Les Miserables](#)

Copyright © 2018 mrhi, Inc. All rights reserved. Contact webmaster for more information. 02-441-6006

뷰포트(viewport)를 이용한 반응형 레이아웃으로 변경하여 보자.

반응형 웹 (responsive web)이란 하나의 웹사이트로 데스크탑 PC, 스마트폰, 태블릿 PC 등 접속하는 디스플레이의 종류에 따라 화면의 크기가 자동으로 변하도록 만든 웹 페이지를 말한다.

미디어 쿼리는 CSS3부터 지원이 되는 CSS기술로 미디어 타입, 화면 크기 등을 기준으로 다른 스타일 시트를 적용할 수 있도록 해준다. 이를 이용해서 화면 크기가 변할때 스타일을 바꿔도록 해서 반응형 웹을 구현할 수 있다. 데스크톱을 기반으로 설계한 웹페이지를 모바일에서 보면 내용이 작게 보이게 된다. 이것은 작은 화면에서 전체 페이지를 표시해야 하기 때문에 화면 비율이 조정된 것이다. 뷰포트를 설정하면 모바일 기기에서 웹페이지의 너비나 화면 비율을 설정할 수 있다. 뷰포트 없이 화면을 설계하면 해상도와 상관없이 무조건 기본 사이즈인 크기로 설정된다. 뷰포트는 레이아웃의 가로값을 가지고 사용 기기의 화면 크기와 비교하고 여백없이 꽉 찬 화면으로 만들어 준다.

뷰포트는 <head></head>안에서 정의한다

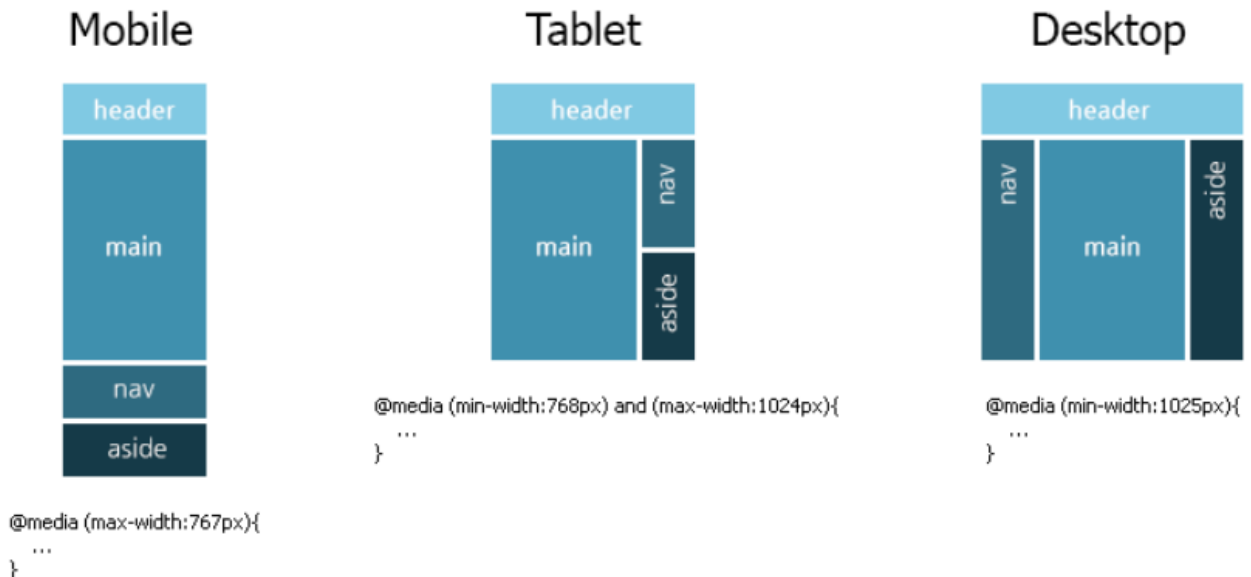
```
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0, user-scalable=no" />
```

[뷰포트의 속성]

속성	설명
width	뷰포트의 가로(너비)이다
height	뷰포트의 세로(높이)이다 기본값은 너비 속성값과 기기이 비율에 따라 계산한다
initial-scale	뷰포트의 초기 화면의 크기 값이다 설정 범위 1~10.0이다.(기본값: 1)
minimum-scale	뷰포트의 최소 비율값이다 설정 범위 0~10.0(기본값: 0.25)
maximum-scale	뷰포트의 최대 비율값이다 설정 범위 0~10.0(기본값: 1.6)
user-scalable	사용자가 확대 또는 축소를 설정하는 속성이다. (기본값: yes)

viewport는 화면에 보이는 영역을 제어하는 기술이다. 속성값은 device-width, device-height가 있다. device-width는 기기의 가로 너비 픽셀값이다. 예를 들어, width= device-width는 기기가 사용하는 가로 너비만큼 적용한다는 의미 (브라우저 너비를 장치 너비에 맞추어 표시) 이다. device-height는 기기의 세로 높이 픽셀값이다. initial-scale는 초기비율(보이는 영역과 웹 페이지를 맞춤)이다. user-scalable= "no" 는 사용자가 화면축소를 하지 못하도록 설정하는 것이다.

주의할 점은 콘텐츠보다 작게 width와 height를 설정할 경우 width와 height는 무시된다. 또한 뷰포트에서 initial-scale을 설정하지 않고 width와 height를 쓰면 전체 화면으로 표시된다. initial-scale에서 width와 height를 설정하지 않으면 기본값으로 자동 설정된다. 그리고 설정하는 방법은 **@media**는 매체 유형에 맞게 스타일 시트를 지원한다. 미디어 쿼리를 사용하면 콘텐츠를 변경하지 않고도 기기에 맞는 형태로 웹페이지의 스타일을 변경할 수 있다.



CSS3 미디어 쿼리는 화면 크기에 따라서 각기 다른 CSS를 적용할 수 있도록 제공한다.

아래의 미디어 쿼리를 사용한 간단한 예제를 살펴보면, 데스크톱 브라우저의 창 크기를 조절해서 뷰 포트의 크기가 달라질 때 마다 서로 다른 내용과 스타일이 화면에 표시되게 작성한 것을 볼 수 있다.

```
/* All - 모든 해상도에서 해석하는 코드(미디어 쿼리 적용 안 함) */
...
/* Mobile - 767px 이하 해상도에서 해석하는 코드 */
@media (max-width:767px){ ... }
/* Tablet & Desktop - 768px 이상 해상도에서 해석하는 코드 */
@media (min-width:768px){ ... }
/* Tablet - 768px ~ 1024px 해상도에서 해석하는 코드 */
@media (min-width:768px) and (max-width:1024px){ ... }
/* Desktop - 1025px 이상 해상도에서 해석하는 코드 */
@media (min-width:1025px){ ... }
```



항상 웹 문서의 최소 너비와 최대 너비 값을 설정한다.

예를 들어, 모든 단말(모바일, 태블릿, 데스크톱)의 해상도를 커버해야 한다면 320~1280 정도의 최솟값과 최댓값을 설정한다.

태블릿과 데스크톱 기준으로 페이지를 작성한다면 너비는 768~1280 정도를 설정한다.

모바일과 태블릿 기준으로 페이지를 작성한다면 너비는 320~1024 정도를 설정한다.

최소값을 설정하지 않으면 웹 문서의 폭은 이론적으로 0px 이 될 수도 있다.

최대값을 설정하지 않으면 웹 문서의 폭은 수천 픽셀(또는 무한대)이 될 수도 있다.

이렇게 최소값과 최대값을 제한하는 이유는 우리의 예측이 언제든지 빗나갈 수 있다는 것(브라우저의 창 너비를 0px에 가깝게 줄이거나 듀얼 모니터를 사용하는 경우)을 가정하기 때문이다.

viewport.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>반응형 레이아웃 적용</title>
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <!-- 브라우저의 호환성 보기 모드를 막고, 해당 브라우저에서 지원하는 가장
    최신 버전의 방식으로 HTML 보여주도록 설정.-->
    <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-
scale=1.0, minimum-scale=1.0, user-scalable=no"/>
    <!--viewport : 화면에 보이는 영역을 제어하는 기술. width는 device-width로 설정.
    initial-scale는 초기비율-->
    <link href="https://fonts.googleapis.com/css?family=Single+Day&display=swap"
    rel="stylesheet">
    <style type="text/css">
      #container {
        width: 960px;
        margin: 0px auto;
        padding: 20px;
        border: 1px solid #5D5D5D;
      }
      header {
        font-family: 'Single Day', cursive;
        padding: 20px; margin-bottom: 20px;
        border: 1px solid #5D5D5D;
      }
      section {
        width: 600px; padding: 20px;
        margin-bottom: 20px;
        float: left;
        border: 1px solid #5D5D5D;
      }
      aside {
        width: 260px; height:230px; padding: 20px;
        margin-bottom: 20px;
        float: right;
        border: 1px solid #5D5D5D;
      }
      aside a{text-decoration: none;}
      footer {
```

```

        clear: both;
        padding: 20px; border: 1px solid #5D5D5D;
    }

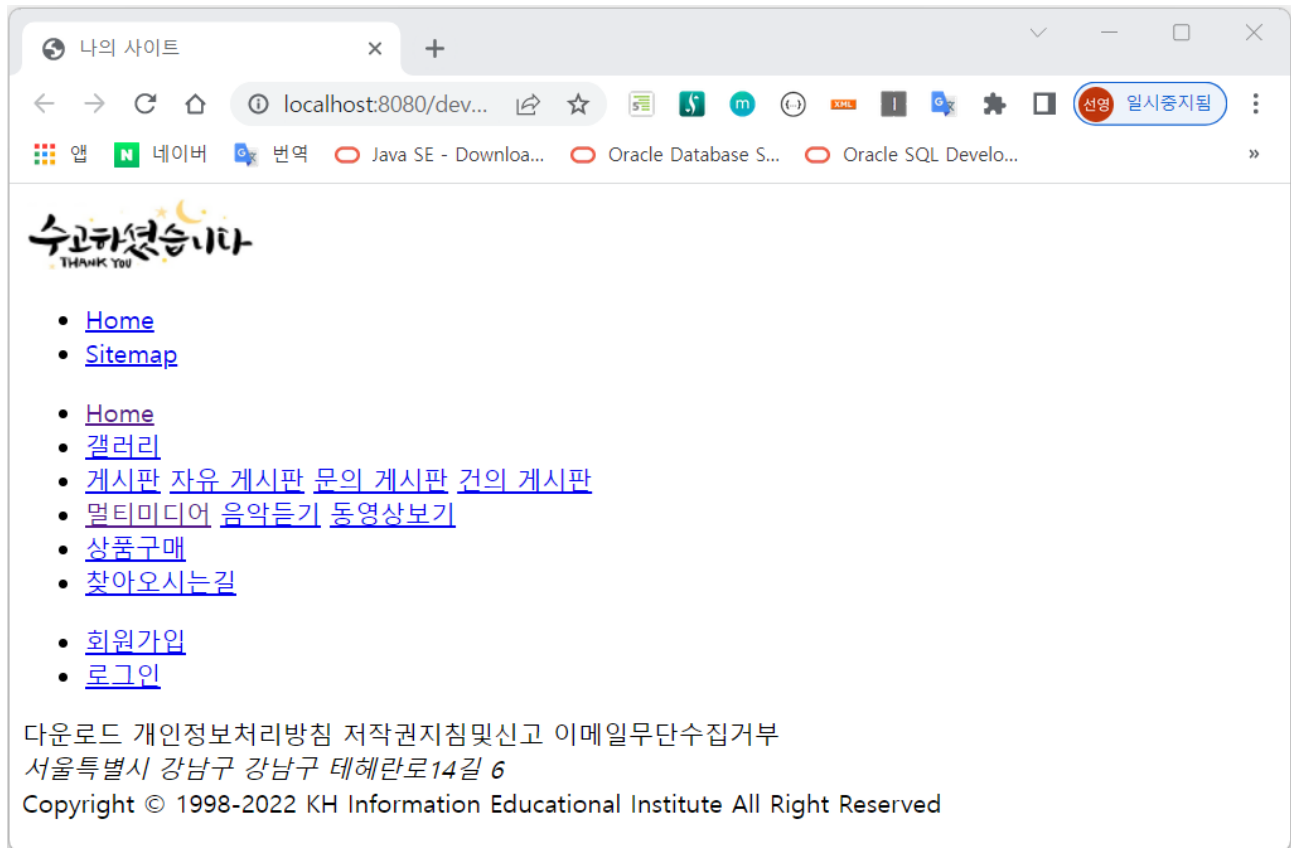
    @media ( max-width: 550px ) {
        #container {
            width: auto;
        }
        section {
            float: none;
            clear: both;
            width: auto;
        }
        aside {
            float: none;
            clear: both;
            width: auto;
        }
    }
</style>
</head>
<body>
    <div id="container">
        <header>
            <h1>동화&nbsp;속 이야기</h1>
        </header>
        <section>
            <article>
                <h2>내용</h2>
                <p>When Alice follows the White Rabbit down a rabbit
                hole, she finds herself in a land where she can change her
                size. </p>
                <p>She meets a Duchess with a pig for a baby, a Cook
                that throws spoons, a wise but mad Cheshire cat, the
                March Hare, Mad Hatter and a Dormouse. A Caterpillar
                shows her how to adjust her height by eating from the
                right and left sides of a mushroom. With that
                knowledge,</p>
            </article>
        </section>
        <aside>

```

```
<h2>도서 선택</h2>
<ol>
  <li><a href="#">Alice's Adventures in Wonderland</a></li>
  <li><a href="#">The Adventures of Tom Sawyer</a></li>
  <li><a href="#">Anne of Green Gables</a></li>
  <li><a href="#">Les Miserables</a></li>
</ol>
</aside>
<footer>
  <p> Copyright &copy; 2018 mrhi, Inc. All rights reserved.
  Contact webmaster for more information. 02-441-6006 </p>
</footer>
</div>
</body>
</html>
```

2.6 실습

[HTML 문서 작성 후 실행 화면 – CSS 적용 전]



```
index.html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>나의 사이트</title>
    <!-- <link rel="stylesheet" type="text/css" href="css/main.css"> -->
  </head>
  <body>
    <!-- header ----- -->
    <header>
      <div class="container">
        <div class="logo">  </div>
        <ul class="sidemenu">
          <li> <a href="#">Home</a> </li>
          <li> <a href="#">Sitemap</a> </li>
        </ul>
      </div>
    </header>
  </body>
</html>
```

```

</header>
<!-- //header -->

<!-- nav ----- -->
<nav>
    <div class="container">
        <ul class="leftMenu">
            <li class="active"><a href="index.html">Home</a></li>
            <li><a href="html/galary.html">갤러리</a></li>
            <li class="dropBox">
                <a href="html/board.html">게시판</a>
                <span class="dropdown">
                    <span><a href="#">자유 게시판</a></span>
                    <span><a href="#">문의 게시판</a></span>
                    <span><a href="#">건의 게시판</a></span>
                </span>
            </li>
            <li class="dropBox">
                <a href="">멀티미디어</a>
                <span class="dropdown">
                    <span><a href="#">음악듣기</a></span>
                    <span><a href="#">동영상보기</a></span>
                </span>
            </li>
            <li class="coming-soon"><a href="#">상품구매</a></li>
            <li><a href="#">찾아오시는길</a></li>
        </ul>
        <ul class="rightMenu">
            <li><a href="html/join.html">회원가입</a></li>
            <li class="active"><a href="html/login.html">로그인</a></li>
        </ul>
    </div>
</nav>
<!-- //nav -->

<!-- section ----- -->
<section id="section1">
    <div class="container">
        <div class="leftBox">
            <div class="graph1">

```

```

                                <!-- 그래프1 추가 -->
                                </div>
                                <div class="graph2">
                                    <!-- 그래프2 추가 -->
                                </div>
                            </div>
                            <div class="rightBox">
                                <div class="imgBox">
                                    <!-- 이미지 추가 -->
                                </div>
                                <div class="table">
                                    <!-- 테이블 추가 -->
                                </div>
                            </div>
                        </div>
                    </section>
                    <div class="clear"> </div>
                    <!-- //section -->

                    <!-- footer ----- -->
                    <footer>
                        <div class="top">
                            <div class="container">
                                <span>다운로드</span>
                                <span>개인정보처리방침</span>
                                <span>저작권지침및신고</span>
                                <span>이메일무단수집거부</span>
                            </div>
                        </div>
                        <div class="container">
                            <address>
                                서울특별시 강남구 강남구 테헤란로14길 6
                            </address>
                            <div class="copyright">
                                Copyright © 1998-2022 KH Information Educational Institute
                                All Right Reserved
                            </div>
                        </div>
                    </footer>
                    <!-- //footer -->
                </body>

```

</html>

[css 적용 결과 화면]

css/main.css

```
/* 기본 설정 */
*{
    padding: 0;
    margin: 0 auto;
    box-sizing: border-box;
    list-style: none;
}
.container{
    width: 1080px;
    height: 100%;
    padding: 0 20px;
}
li{
    float: left;
}
a{
    width: 100%;
    height: 100%;
    color: inherit;
    text-decoration: none;
    display: inline-block;
}
body{
    min-height: 100vh;
    position: relative;
}
.clear{
    clear: both;
}
/* //기본 설정 */

/* header */
header{
    width: 100%;
    height: 75px;
    line-height: 75px;
```

```

        color: #00bd39;
    }
    header .logo{
        float: left;
        font-size: 24px;
        font-weight: 700;
    }

    header .sidemenu{
        float: right;
    }
    header .sidemenu > li{
        width: 100px;
        text-align: center;
        cursor: pointer;
        font-weight: 700;
    }
    header .sidemenu > li:hover{
        color: green;
    }
    /* //header */

    /* nav */
    nav{
        width: 100%;
        height: 60px;
        border-bottom: 1px solid #cbcbcb;
    }
    nav .leftMenu{
        float: left;
        height: 100%;
        font-size: 19px;
        font-weight: bold;
    }
    nav .leftMenu > li{
        width: auto; height: 100%;
        padding: 0 20px;
        line-height: 60px;
        text-align: center;
        cursor: pointer;
        position: relative;
    }

```



```

}
nav .leftMenu > li:hover > .dropmenu{
    visibility: visible;
}
nav .leftMenu > li > .dropmenu{
    display: inline-block;
    width: 150px;
    clear: both;
    position: absolute;
    top: 59px;
    left: 0;
    text-align: left;
    box-shadow: 0px 2px 2px rgba(0, 0, 0, 0.4);
    visibility: hidden;
}
nav .leftMenu > li > .dropmenu > span{
    width: 100%;
    display: inline-block;
    padding-left: 20px;
    background: #00bd39;
    color: white;
    font-weight: 400;
}
nav .leftMenu > li > .dropmenu > span:hover{
    font-weight: 700;
}
nav .leftMenu > li:hover{
    color: #00bd39;
    font-weight: 700;
    border-bottom: 3px solid #00bd39;
}
nav .leftMenu > li.active{
    color: #00bd39;
    font-weight: 700;
    border-bottom: 3px solid #00bd39;
}

nav .rightMenu{
    float: right;
}
nav .rightMenu > li{

```

```

width: 100px; height: 36px;
line-height: 32px;
margin-top: 12px; margin-left: 10px;
padding: 0 10px;
text-align: center;
color: #00bd39;
font-weight: 700;
border: 2px solid #00bd39;
cursor: pointer;
border-radius: 50px;
}
nav .rightMenu > li:hover, nav .rightMenu > li.active{
color: white;
background: #00bd39;
}
/* //nav */

/* #section1 */
section{
width: 100%;
height: 650px;
padding: 30px 0;
background: #fafafa;
}

/* footer */
footer{
width: 100%;
height: 115px;
border-top: 1px solid #cbcbcb;
background: white;
font-size: 18px;
}
footer > .container{
height: auto;
}
footer .top{
width: 100%;
height: 40px;
border-bottom: 1px solid #cbcbcb;
margin-bottom: 15px;
}

```

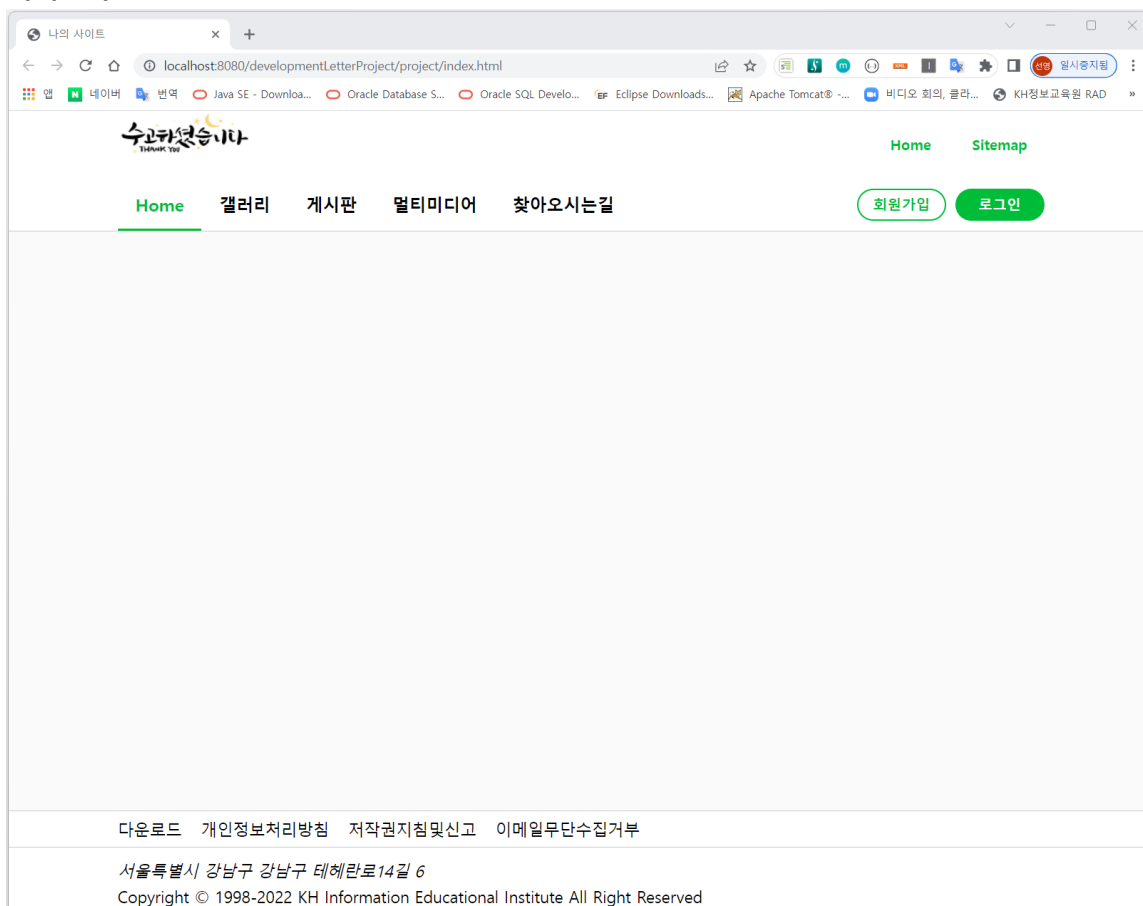
```

}
footer .top span{
    height: 100%;
    line-height: 40px;
    padding-right: 15px;
    cursor: pointer;
}
footer .top span:hover{
    font-weight: 700;
}
footer .copyright{
    margin-top: 5px;
}
/* //footer */

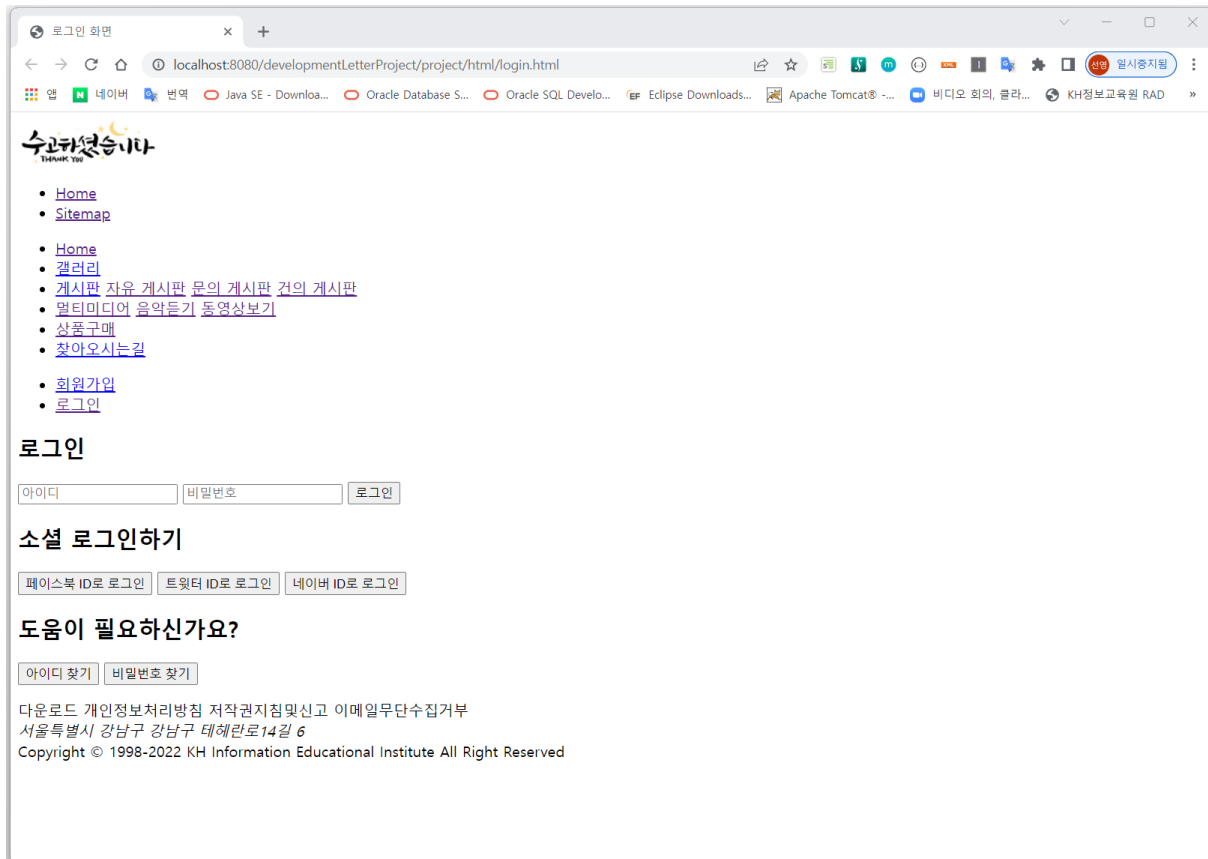
li.coming-soon {
    display: none; /* 상품구매는 잠시 보이지 않게 설정 */
}

```

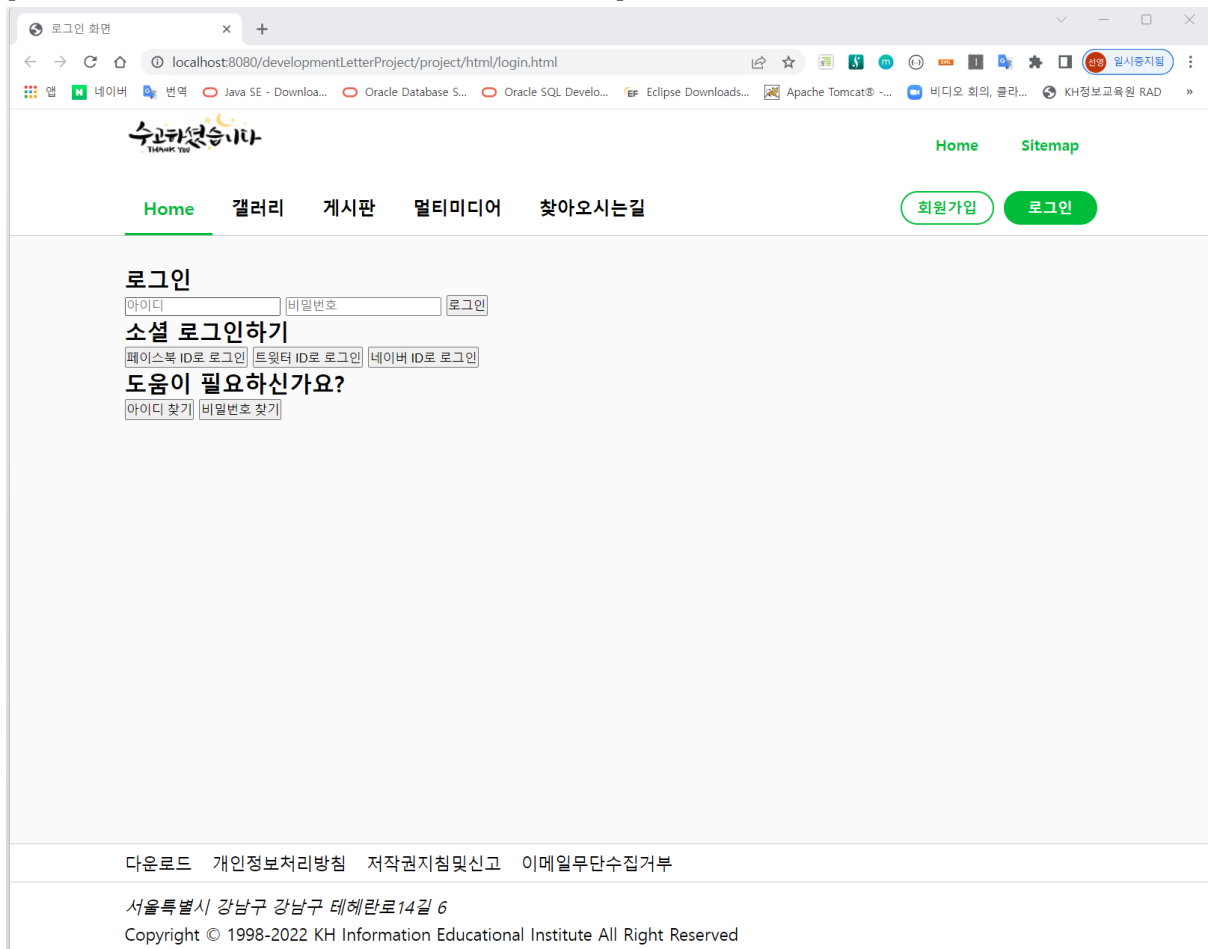
html 문서에서 <!--<link rel="stylesheet" type="text/css" href="css/main.css"> -->이 부분에 주석을 해제한다.



[스타일시트를 전혀 적용하지 않았을 때 결과 화면]



[main.css 스타일시트만 적용하였을 때 결과 화면]



login.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>로그인 화면</title>
    <link rel="stylesheet" type="text/css" href="../css/main.css">
    <!-- <link rel="stylesheet" type="text/css" href="../css/login.css"> -->
  </head>
  <body>
    <!-- header ----- -->
    <header>
      <div class="container">
        <div class="logo">  </div>
        <ul class="sidemenu">
          <li> <a href="../index.html">Home</a> </li>
          <li> <a href="#">Sitemap</a> </li>
        </ul>
      </div>
    </header>
    <!-- //header -->

    <!-- nav ----- -->
    <nav>
      <div class="container">
        <ul class="leftMenu">
          <li class="active"> <a href="../index.html">Home</a> </li>

          <li> <a href="gallery.html">갤러리</a> </li>
          <li class="dropBox">
            <a href="board.html">게시판</a>
            <span class="dropdownmenu">
              <span> <a href="#">자유 게시판</a> </span>
              <span> <a href="#">문의 게시판</a> </span>
              <span> <a href="#">건의 게시판</a> </span>
            </span>
          </li>
          <li class="dropBox">
            <a href="">멀티미디어</a>
            <span class="dropdownmenu">
              <span> <a href="#">음악듣기</a> </span>
            </span>
          </li>
        </ul>
      </div>
    </nav>
  </body>
</html>
```

```

        <span><a href="#">동영상보기</a></span>
    </span>

    </li>
    <li class="coming-soon"><a href="#">상품구매</a></li>
    <li><a href="location.html">찾아오시는길</a></li>
</ul>
<ul class="rightMenu">
    <li><a href="join.html">회원가입</a></li>
    <li class="active"><a href="login.html">로그인</a></li>
</ul>
</div>
</nav>
<!-- //nav -->

<!-- section ----- -->
<section id="section2">
    <div class="container">
        <div class="loginBox">
            <div>
                <div class="box">
                    <div class="logo">
                        <h1>로그인</h1>
                    </div>
                    <form action="/login" method="post">
                        <input type="text" name="uid" placeholder="아이디">
                        <input type="password" name="pw" placeholder="비밀번호">
                        <button type="submit">로그인</button>
                    </form>
                </div>
                <div class="box">
                    <div class="social logo">
                        <h1>소셜 로그인하기</h1>
                        <button class="facebook">페이스북 ID로 로그인</button>
                        <button class="twitter">트위터 ID로 로그인</button>
                        <button class="naver">네이버 ID로 로그인</button>
                    </div>
                </div>
                <div class="clear"></div>
                <div class="alert">
                    <h1>도움이 필요하신가요?</h1>
                    <p>

```

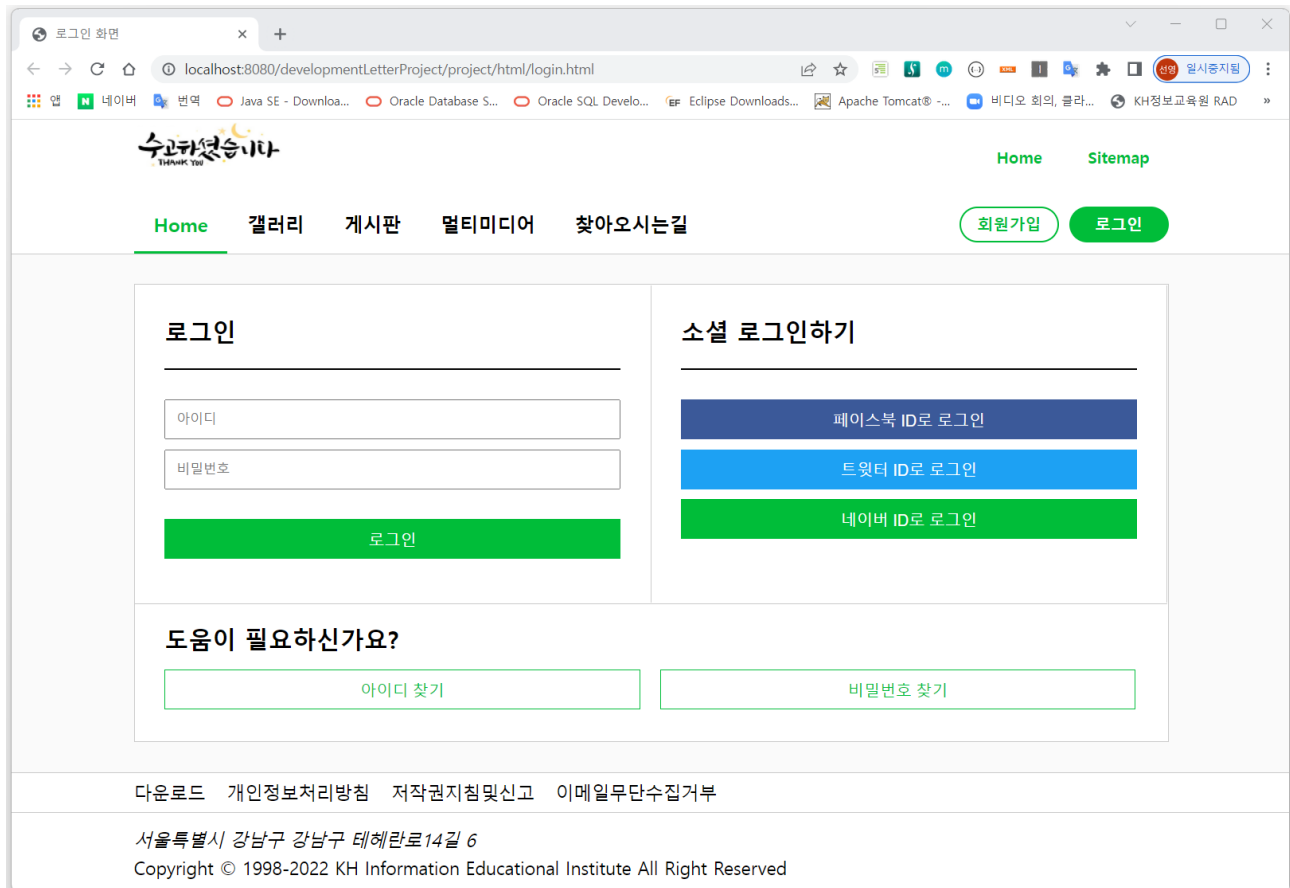
```

        <button>아이디 찾기</button>
        <button>비밀번호 찾기</button>
    </p>
</div>
</div>
</div>
</div>
</section>
<div class="clear"></div>
<!-- //section -->

<!-- footer ----->
<footer>
    <div class="top">
        <div class="container">
            <span>다운로드</span>
            <span>개인정보처리방침</span>
            <span>저작권지침및신고</span>
            <span>이메일무단수집거부</span>
        </div>
    </div>
    <div class="container">
        <address>
            서울특별시 강남구 강남구 테헤란로14길 6
        </address>
        <div class="copyright">
            Copyright © 1998-2022 KH Information Educational Institute
            All Right Reserved
        </div>
    </div>
</footer>
<!-- //footer -->
</body>
</html>

```

[스타일시트 적용 후 결과 화면]



css/login.css

```
@charset "UTF-8";

/* #section2 */
#section2{
    height: 520px;
}
#section2 > .container > .loginBox{
    width: 100%;
    height: 100%;
    background: white;
    border: 1px solid #cbcbcb;
}
#section2 .loginBox > div{
    width: 100%;
    height: 320px;
    border-bottom: 1px solid #cbcbcb;
}
}
```



```

#section2 .loginBox .box{
    width: 50%;
    height: 100%;
    float: left;
    padding: 30px 30px;
    border-right: 1px solid #cbcbcb;
}
#section2 .loginBox .box .logo{
    height: 55px;
    border-bottom: 2px solid black;
}
#section2 .loginBox .box form{
    margin-top: 20px;
}
#section2 .loginBox .box input{
    width: 100%;
    height: 40px;
    margin-top: 10px;
    padding-left: 10px;
}
#section2 .loginBox button{
    width: 100%;
    height: 40px;
    background: #00bd39;
    color: white;

    border: 0;
    font-size: 16px;
    outline: 0;
    cursor: pointer;
}
#section2 .loginBox .box form > button{
    margin-top: 30px;
}
#section2 .social{
    margin-top: 0px;
}
#section2 .social > h1{
    margin-bottom: 20px;
}
#section2 .social > button{

```

```

float: left;
margin-top: 10px;
}
#section2 .social > button.facebook{
background: #3b5999;
margin-top: 33px;
}
#section2 .social > button.twitter{
background: #1da1f3;
}
#section2 .social > button.naver{
background: #00bd39;
}

#section2 .alert{
padding: 20px 30px;
}
#section2 .alert button{
width: 478px;
margin-top: 15px;
float: left;
background: white;
color: #00bd39;
border: 1px solid #00bd39;
}
#section2 .alert button:nth-child(1){
margin-right: 10px;
}
#section2 .alert button:last-child{
margin-left: 10px;
}
#section2 .alert button:hover{
color: white;
background: #00bd39;
}

```

html 문서에서 <link rel="stylesheet" type="text/css" href="../css/login.css"> -->
이 부분에 주석을 해제한다.