

기말고사 온라인 과제

결과 보고서

지도교수: 안용학 교수님

컴퓨터공학과

16010963 이선아

2020.06.28

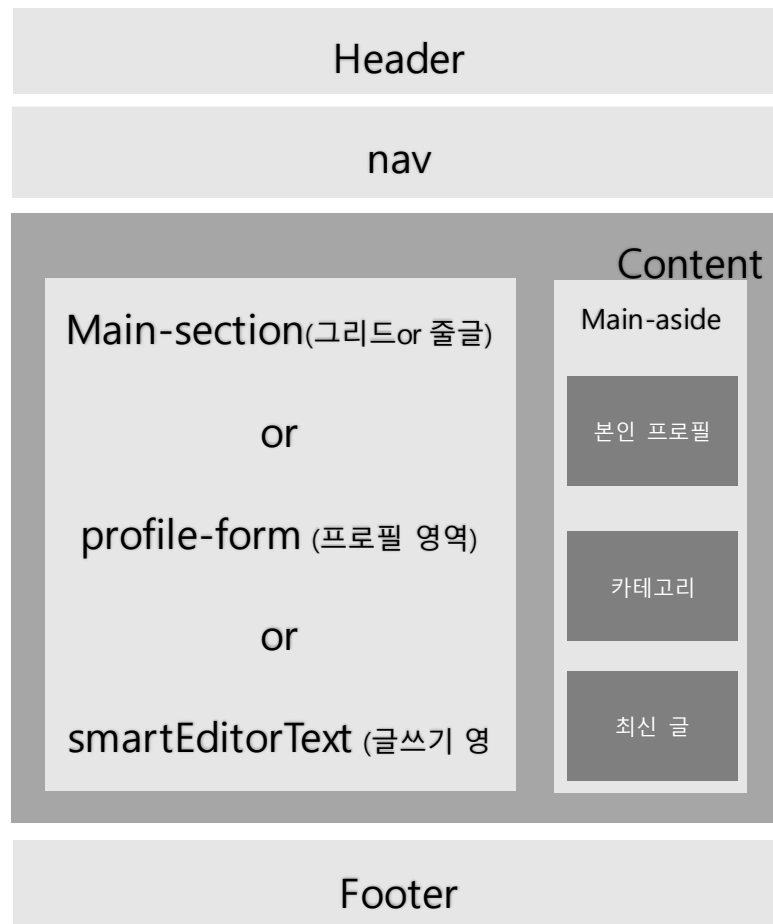
1. 웹 페이지 전체 구조도와 레이아웃

(1) 구조도



(2) 레이아웃

각 페이지는 공통으로 아래와 같은 레이아웃을 가짐(친구 추가 창 제외)



2. 메뉴(nav)

HOME	국외 드라마	국외 영화	국내 드라마
	프렌즈		목록 열기
	기묘한 이야기		

메뉴는 HOME, 국외 드라마, 국외 영화, 국내 드라마로 구성되어 있으며

HOME을 제외한 각 메뉴들은 호버시 글자색과, 배경색을 바꾸며 그 안의 inner 메뉴를 보여준다. Inner 메뉴들 또한 호버시 동일한 효과를 주었다.

HOME 항목을 클릭시 메인페이지로 이동한다.

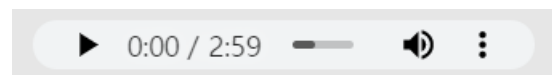
3. 페이지 공통 기능(친구 추가 창 제외)

(1) 파비콘 설정



블로그 접속 시 상단 탭에 보여지는 아이콘 기능이다. 블로그를 나타내는 이미지로 표현하여, 사이트의 성격을 나타냈다.

(2) Bgm 기능



각 페이지의 좌측 상단에 위치하는 bgm 기능이다. controls박스가 보이도록 설정했고, 노래는 반복재생 되도록 설정했다.

(3) 현재 시각 알려주는 기능

2020/06/24 16:58:07

헤더에 위치해 있으며, 해당 페이지가 로딩이 다 된후, setInterval을 이용하여 nowclock() 함수를 1초마다 호출한다. Nowclock()함수는 현재 시간을 객체로 받아와, 헤더에 html 코드를 추가한다. 시간의 형식은 moment.js를 이용하여 초 단위까지 표현했다.

(4) 검색 기능



메뉴들과 같은 nav 안에 위치해 있으며, 검색어를 입력하지 않고 엔터 혹은 검색 버튼을 클릭시 "검색어를 입력하세요."라는 알림이 뜨며, 해당 페이지 내에 존재하지 않는 검색어를 입력했을 경우에는 "검색 결과가 없습니다."라는 알림이 뜬다. Search() 함수를 구현하였으며, 이 함수는 검색어를 입력하는 input칸에서 엔터를 입력하거나 검색버튼을 클릭할 경우에 호출된다. Input 칸에서 엔터 입력시 새로고침을 방지하기 위해서 상위 태그인 form 태그에 onsubmit에 return false;를 주었다.

사용자가 입력한 검색어가 위치한 태그의 위치로 스크롤을 이동시키고, 검색어를 찾아 배경색을 바꾼다. 해당 검색어만 배경색을 바꾸기 위해 html 코드를 추가하는 방향으로 구현하였다. 사용자가 입력한 검색어가 여러 개 있을 경우 모두 다 찾아내며, 대소문자를 구분하지 않고 찾아낸다.

그리드 형식의 페이지에서는 h5태그를 기준으로, 줄글 형식의 게시글 페이지에서는 p태그를 기준으로 검색한다.

Placeholder로 Search 글자를 넣고 italic체로 css로 주고, focus시에 placeholder가 사라지도록 구현했다.

(5) 본인 프로필 추가



Main-aside에 위치하는 블로그 주인의 프로필을 나타내는 영역이다. 프로필 이미지와, 닉네임, 그 아래 아이디가 위치하며, 글쓰기 버튼 옆에 png 이미지를 삽입하였다.

프로필과 글쓰기 버튼을 클릭하면 해당 양식이 나타난다. 해당 양식은 뒤에 (7), (8)번에서 더 자세히 설명할 예정이다.

친구 추가 버튼 클릭 시 나타날 창 역시 (9)에서 설명할 예정이다.

프로필, 글쓰기, 친구 추가 버튼은 모두 hover시 커서가 pointer로 변환되며, 친구 추가 버튼은 배경색도 바뀐다.

(6) 카테고리 & 최근 글



Comedy, Romance 메뉴 접기 전

Comedy, Romance 메뉴 접기 후

Main-aside에 위치하는 카테고리 및 최근 글 목록이다. 프로필 영역의 아래에 위치한다.

각 게시물 앞에 png 이미지를 추가하여 페이지의 스타일을 명시했으며, Best, Action 등 각 메뉴들은 전체보기라는 큰 메뉴의 하위 메뉴임을 역시 png 이미지를 통해 나타냈다. 우측에 화살표 모양이 나타난 항목들은 접기가 가능하며, 접은 후에는 화살표 모양이 반대로 바뀐다. 화살표를 클릭하면 함수를 통해 아래의 게시글이 나타난 영역의 css 중 display가 none이나 block으로 변경된다. 또한 클릭 시 화살표 모양을 반대로 돌리기 위해 css에 transform을 rotate 180이나 0deg로 변경을 주어 모양을 회전시켰다.

카테고리와 최근 글 아래에 위치한 하위 항목들은 모두 hover 시에 밑줄이 그어지며 각 페이지로 이동이 가능하다.

(7) 프로필 버튼 클릭시 나타나는 양식

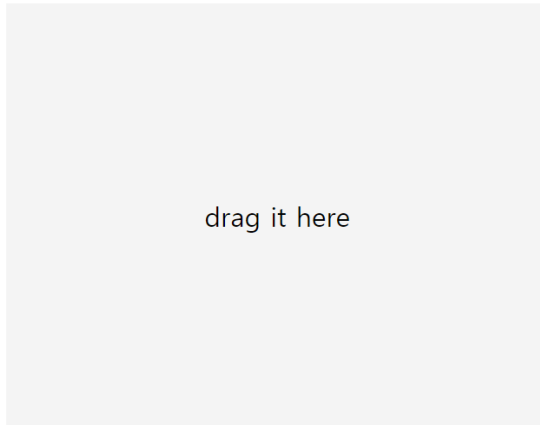
기본 정보

이름	이선아	성별	여자
별명	나서		

기본 정보

이름	이선아	성별	여자
별명	선아		

프로필 사진 변경



선택된 파일 없음



Image Upload

변경

별명 변경/ 이미지 업로드 전

프로필 사진 변경



dog5.jpg



Image Upload

변경

별명 변경/ 이미지 업로드 후

프로필 양식은 Main-aside의 왼쪽에 위치하는 프로필 변경 영역으로, 본인 프로필의 프로필 버튼을 클릭하면 나타난다. 기본 정보와 프로필 사진 변경 영역으로 구성되어 있다. 기본 정보에는 이름, 성별, 별명으로 구성되어 있으며, 별명은 변경이 가능하다. 별명 입력 칸을 누르고 입력된 값을 지우면 placeholder의 값으로 넣어놓은 “닉네임을 입력하세요”라는 값이 뜬다. 이후 새로운 별명을 입력 후 엔터 키를 누르게 되면 Main-aside에 위치한 본인 프로필 영역에서의 닉네임 값이 바뀌게 된다. 별명을 입력하지 않고 엔터 키 입력시 알림이 뜬다.

프로필 사진 변경은 본인 프로필 영역에서의 프로필 사진을 바꾸는 기능이다. 사용자가 Image Upload 버튼이나 drag & drop 기능으로 이미지를 업로드 하면 “drag it here” 문구가 나타난 영역에 이미지가 뜨게 되며 “drag it here” 안내문구는 삭제된다. 또한 아래 위치한 “선택된 파일 없음”의 문구가 사라지고 대신 업로드 된 파일의 이름이 뜨게 된다. 사용자가 하나 이상의 파일을 업로드 할 경우에는 “하나의 이미지만 업로드 해 주세요.”라는 알림이 뜨며, 업로드 파일이 이미지 파일이 아닐 경우에는 “이미지가 아닙니다.”라는 알림이 뜬다. 이미지를 업로드 한 후 옆에 위치한 변경 버튼을 누르게 되면 프로필 사진이 업로드 한 사진으로 변경된다. 만약 이미지를 업로드 하지 않고 변경 버튼을 누를 경우 “이미지를 업로드 해주세요.”라는 알림이 뜬다.

Drag & drop 기능은 이미지가 업로드 될 영역에 drop 했을 경우에만 작동하며, 이외의 영역에 drop 할 경우 해당 파일의 url로 이동하는 기본 설정을 막아두었다.

Image Upload 버튼과 변경 버튼은 hover시 배경 색을 바꾸도록 구현하였다.

아래 사진을 통해서 본인 프로필 영역의 닉네임과 프로필 사진이 바뀌는 것을 확인 할 수 있다.



(8) 글쓰기 버튼 클릭 시 나타나는 양식

The image shows a web form for creating a post. It consists of the following elements:

- 제목 (Title):** A text input field at the top.
- 내용 (Content):** A large text area for the main content.
- Toolbar:** A horizontal bar with various icons for text formatting (bold, italic, underline, etc.), font color, background color, and list creation.
- Buttons:** At the bottom, there are two buttons: '저장' (Save) and '취소' (Cancel).
- Message:** A small yellow box at the bottom of the content area says: "아래 영역을 드래그하여 입력창 크기를 조절할 수 있습니다. X" (You can adjust the size of the input area by dragging below. X).

글쓰기 양식 영역은 Main-aside의 왼쪽에 위치하며, 본인 프로필 영역에서 프로필 버튼을 클릭하면 나타난다.

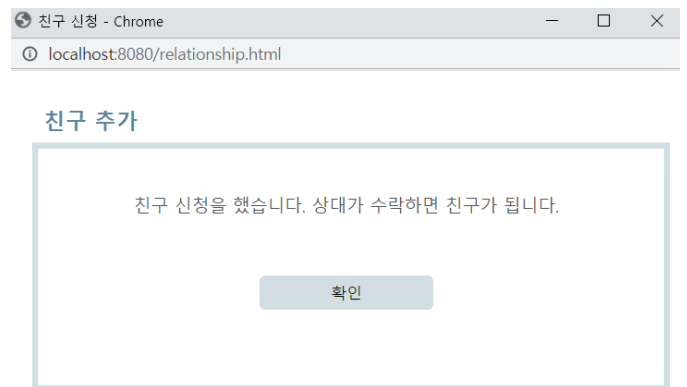
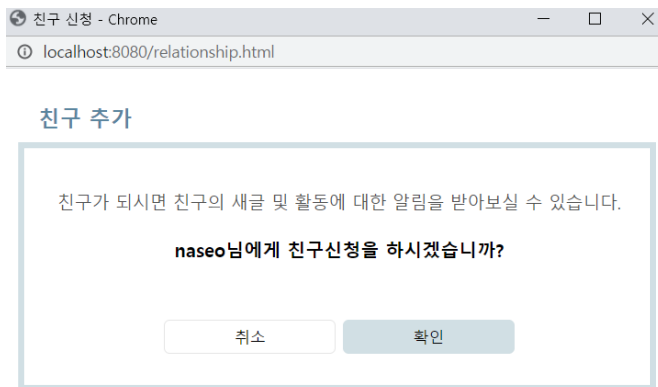
table 태그로 제목과 내용을 입력할 공간을 구성하고, input태그로 저장과 취소 버튼을 만들었다. 실제로 글이 등록되는 기능은 구현 하지 못했지만, naver의 smartEditor를 이용하여 글쓰기 양식을 구성했다. 실제 위에 나타난 기능들은 모두 적용이 가능하다.

저장 버튼과 취소 버튼은 hover시 배경 색을 바꾸고 마우스 cursor모양을 pointer로 바꾸도록 구현하였다.

저장 버튼 클릭 시 alert을 통해 "저장되었습니다"라는 알림이 뜨며,

취소 버튼 클릭 시 confirm을 통해 "작성된 글이 저장되지 않았습니다. 양식을 벗어나시겠습니까?" 라는 알림이 뜨며, 확인 버튼 클릭 시 글쓰기 양식이 사라지며, 원래 게시글의 Main-section이 나타나게 된다. 취소 버튼 클릭시에는 해당 글쓰기 양식이 그대로 유지된다.

(9) 친구 추가 버튼 클릭 시 나타날 창



친구 버튼 클릭 시 onclick을 통해 open_page()함수가 호출되고, window.open을 이용하여 왼쪽에 위치한 창을 나타나게한다.

왼쪽 창에는 안내문구와 함께 친구 신청의 여부를 묻는 문구가 나타나며, 아래에는 취소, 확인 두개의 버튼이 존재한다. 확인 버튼은 따로 배경색을 주었다.

왼쪽 창의 취소버튼 클릭 시, onclick의 self.close()를 통해 해당 창을 종료하며, 확인 버튼 클릭시 박스 쳐진 영역이 삭제되고 오른쪽 창처럼 변하게 된다.

오른쪽 창에서의 확인 버튼 클릭시 취소버튼과 같은 방법으로 해당 창이 종료된다.

(10) 좋아요 기능

*그리드 내 좋아요 버튼



*게시글 내 좋아요 버튼



좋아요 버튼의 하트 모양은 가운데에 사각형을 회전시켜 마름모 모양으로 만든 후 css로 before, after 기능을 이용하여 위에 둥근 모양을 추가시켜 만들었다.

그리드 내의 좋아요 버튼을 클릭하면 like 함수가 호출되며 해당 index(그리드 순서)값이 인자로 넘어간다. 따라서 해당 그리드의 like버튼에만 변화가 나타난다.

게시글 내에는 좋아요 버튼이 하나만 있기 때문에 따로 인덱스 값을 넘기지 않고, 클릭 시 버튼의 배경색과 글씨색을 변경하는 변화를 주었다.

좋아요 버튼이 눌린 상태에서 한번 더 누르게 되면 원래 상태로 돌아가게된다.

(11) 전체 글 목록

목록 열기



목록 닫기		
글 제목	조회수	작성일
프렌즈	12	2020.06.22
킹덤	22	2020.06.20
기묘한 이야기	23	2020.06.19

1 2 3 4

전체보기 페이지를 제외한 모든 페이지의 Main-section의 가장 상단에는 전체 글 목록이 위치한다. 기본으로 보이지 않게 설정 한 후, 목록 열기를 클릭하면 글 목록이 보이도록 설정 했다. 목록 열기 글씨 또한 목록 닫기로 변경된다.

Table 태그로 목록을 작성했으며 글 제목, 조회수, 작성일로 구성되어있다. 글 제목 클릭 시 해당 게시글로 이동이 가능하다.

한 목록 페이지에 3개의 글 목록들이 나열되며 총 10개의 글이 존재 해, 목록 페이지 수는 4개로 구성된다.

이에 관한 js는 내부 스크립트로 작성하였다.

한 목록 페이지에 3개의 글 목록들을 나열할 것이기 때문에, 목록 테이블의 하위 tr요소를 배열로 받아와 3개로 구성된 새로운 배열로 만들었다.

아래에 위치한 목록 페이지 숫자는 js를 통해 해당 div에 html 코드를 추가하는 방식으로 진행했다. 목록 페이지 숫자를 클릭하면 해당 숫자만 크기가 커지는 css를 주었고, 나머지 숫자들은 그대로 유지하게 만들었다.

각 페이지 숫자마다 나열되는 글이 달라야 하기 때문에 새로운 배열을 만들어 보여주는 방식으로 구현했다. 만약 페이지 숫자 1을 클릭한다면 해당 a태그의 rel 속성 값인 0(페이지 숫자-1)을 받아와 시작 목록으로 설정하고, 마지막 목록은 시작목록에 +3(한 목록 페이지에 나열될 목록의 개수)으로 설정하여 배열을 만들었다.

목록 열기/닫기와 각 게시물의 제목, 아래의 페이지 숫자들은 hover시 밑줄이 그어지도록 구현했다.

(12) Footer

Created By Naseo

해당 글자 클릭 시 블로그 메인 화면으로 이동한다.

^TOP

항상 같은 위치에 고정 값으로 위치한 top 버튼이다. svg 이미지와 같이 나타냈으며, hover시 커서 모양을 포인터로 바꾼다. 클릭 시 현재 페이지의 최상단으로 이동하도록 js를 이용하여 구현하였다.

4. 각 페이지 추가 기능

(1) Best 페이지(메인 페이지)

- Section 영역



메인 페이지는 게시물중 Best 항목 6개를 뽑아 한 페이지에 그리드 형식으로 나열한 레이아웃의 페이지이다.

각 그리드에는 각 게시물의 대표 이미지와, 댓글 수, 작성된 날짜, 좋아요 버튼으로 구성되어있다. 3가지 항목들은 position을 absolute로 주어 고정된 위치로 구현했다.

그리드에 관한 css는 내부 스타일시트로 작성하였다.

그리드 클릭시 goReplace함수를 호출하여, location.replace를 통해 연결된 각 게시물로 이동이 가능하다.

이미지는 div를 설정하여 div크기보다 이미지의 크기가 클경우 보이지 않도록 설정했다.

각 게시글의 제목은 hover시에 밑줄이 그어지도록 설정했다.

(2) 전체보기 페이지

· Section 영역



전체보기 페이지는 전체 게시물을 그리드 형식으로 2페이지에 걸쳐 세로로 나열한 레이아웃이다.

한 화면에 총 5개의 그리드가 보이고 아래의 페이지 숫자를 통해 뒷 페이지로 넘어갈수 있다.

각 그리드는 Best 페이지의 그리드의 구성과 같고, 위치와 크기만 수정했다.

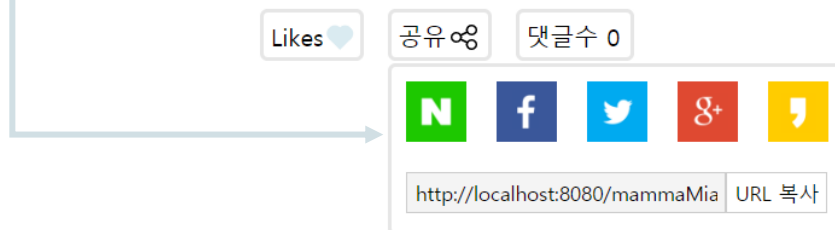
그리드를 클릭하면 각 페이지로 이동하며, like 버튼 클릭시에 event.cancelBubble=true를 통해 해당 게시물의 url로의 이동을 막고 like버튼 변화효과가 나타나도록 구현하였다. 이는 Best 페이지에도 똑같이 적용된다.

(3) 게시물 페이지

- Article 영역



- 공유 버튼



공유버튼을 클릭하면 5개의 sns 로고 버튼들과 해당 게시물의 url을 복사 할 수 있는 div가 나온다. 각 sns 이미지들은 onclick속성으로 각 함수로 이동하게 되고, 각 함수들은 각 sns 공유 창을 새 창으로 띄운다. 해당 js는 내부 스크립트로 작성하였다.

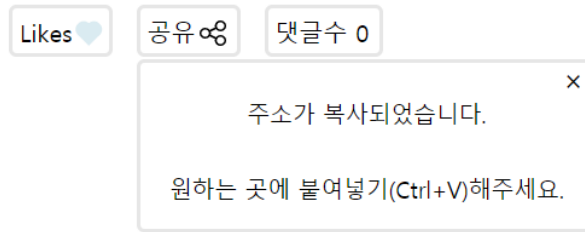
아래의 url 영역은 해당 게시물의 url이 나타날 input text 부분과 input button으로 구현된 url복사 버튼으로 구성되었다. 공유버튼을 클릭하여 해당 sns 박스가 나타나면 getURL함수를 호출 후, location.href를 통해 자동으로 해당 게시글의 url이 나타난다.

위의 그림과 같은 상태에서 공유 버튼 클릭시 sns버튼과 url이 담긴 sns 박스가 사라진다.

각 게시물 페이지에는 해당 게시물에 관한 내용이 article 태그 안에 담겨있다. Box-shadow로 css를 주어 떠 있는 느낌을 주었다.

그 아래로 like, 공유, 댓글 수 버튼이 존재하고, 각 버튼은 border-radius와 border를 이용하여 경계를 표현했고 cursor를 pointer로 설정하였다.

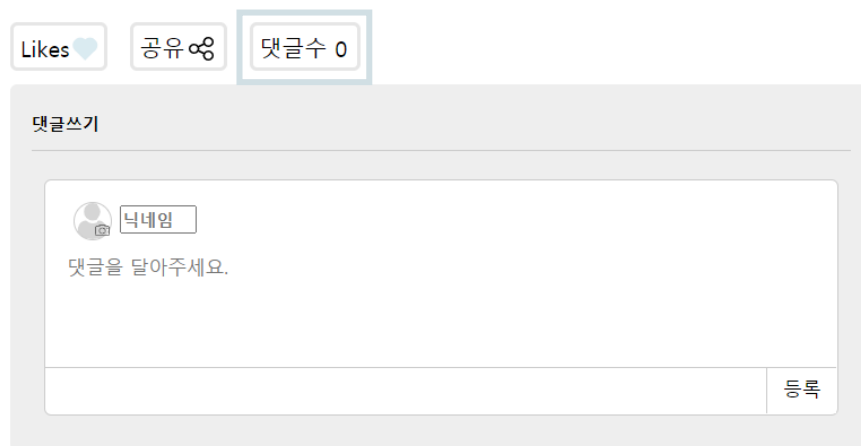
공유버튼은 svg이미지와 높이를 맞추기 위해 position을 relative로 주고, svg이미지의 position을 absolute로 주었다.



url 복사 버튼 클릭시 나타날 안내문구가 담긴 div이다.

오른쪽 상단에 위치한 x버튼은 `onclick="this.parentElement.style.display='none'"`을 통해 div가 닫히도록 구현했다. Position은 absolute로 설정했다.

· 댓글 버튼



댓글 수 버튼 클릭 시 나올 댓글 입력 div이다. 위쪽에는 댓글쓰기 칸이라고 명시되어 있고 아래에는 댓글을 직접 입력 할 수 있는 칸으로 구성되어있다.

사용자가 댓글 작성시 프로필 이미지를 설정 할 수 있도록 input file 태그로 구현하였다. Input file 태그는 보이지 않도록 설정하고, 대신 태그와 연결된 label 태그를 통하여 프로필 사진을 선택 할 수 있다. label에는 기본 프로필 이미지와 함께 카메라 svg이미지로 프로필 사진 선택 기능이 있음을 나타냈다.

사용자가 프로필 이미지를 선택하면 label태그의 onchange를 통하여 함수가 호출되어 이미지 파일인 경우에 기본 프로필 이미지에서 선택 된 이미지로 변경되고, 카메라 사진은 사라지게 된다. 이미지 파일이 아닐 경우에는 알림이 뜨게된다. 프로필 사진을 선택 하지 않았을 경우에 댓글 작성 시 기본 이미지로 등록된다.

프로필 사진 옆에는 닉네임 입력 칸이 있다. Placeholder로 닉네임 칸임을 명시 했고, focus시에는 placeholder가 사라지도록 구현했다. 이는 아래 댓글칸의 "댓글을 달아주세요"에도 똑같이 적용된다. 또한, 닉네임을 입력하지 않았을때 알림창이 뜬다. 이도 댓글칸에 똑같이 적용된다.

댓글 칸은 댓글 내용을 입력 할 수 있는 텍스트 칸과 등록버튼으로 구성되어있다. 텍스트 칸은 textarea로 구현하였는데 크기 조절을 막기위해 resize를 none으로 주었다. 등록버튼과 가로 정렬이 정확히 맞지 않아 텍스트 칸의 border-bottom으로 경계를 표현하고, z-index값을 1로 주어 등록 버튼보다 위에 나타나도록 구현했다.

```

<div class="user-comment-group"><!-- 댓글 입력시 입력할 댓글 양식; 보이지 않게 설정 되어 있음-->
  <div class="user-comment">
    <div class="user">
      <label for="photo" class="userImg"></label>
      <span class="userId"></span>
    </div>
    <p class="text"></p>
    <span class="date"></span>
  </div>
</div>

```

댓글 작성 시 등록 될 댓글 형식 html 코드

댓글 등록시 나타나게 될 형식은 위의 사진과 같이 html 코드에 작성은 되어있지만 보이지 않도록 설정 되어 있다.

등록 버튼은 cursor를 pointer로 변경하고 클릭 시 작성한 댓글 내용이 나타나게 한다.

등록 버튼 클릭 시 사용자 프로필 이미지, 아이디, 댓글 내용, 날짜가 들어갈 영역의 class 이름을 뒤에 숫자를 붙여 변경한다. 예를 들면 프로필 이미지의 경우 userImg1, userImg2 이런식으로 변경해 다른 사용자와 겹치지 않도록 한다. 사용자가 입력한 닉네임과 댓글 내용을 가져와, 형식에 맞는 div위치에 값을 넣는다. date칸에는 시계 기능과 똑같이 구현하였지만 형식을 분까지 표현했다.



댓글 작성 후의 화면

사용자가 입력한 내용을 출력 한 후에는 사용자의 프로필 이미지와 닉네임, 댓글 내용 텍스트 칸을 처음과 같이 초기화한다.

사용자를 겹치지 않기 위해 userImg1, userImg2 이렇게 클래스 명을 변경할 때 사용했던 카운트 변수를 이용하여 댓글 수 세는 것에도 이용해 위 댓글수 버튼의 숫자도 변경되도록 구현했다.



공유 버튼과 댓글 수 버튼을 둘 다 눌렀을 경우

공유버튼과 댓글수 버튼을 클릭하여 동시에 두개의 영역이 보이게 되는 경우에는 sns 박스가 댓글 쓰기 영역보다 더 위에 위치하도록 z-index값을 주었다.