

LSR_과정

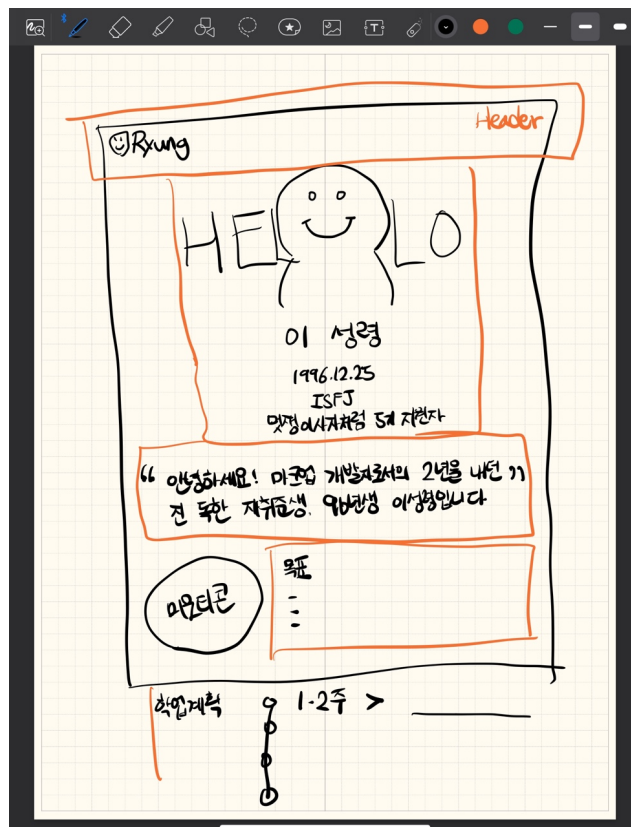
♡ 목표

- 경력자라 html/css 자신있다고 말한 거 부끄럽지 않게 만들기
- 비대면 속에서도 나를 보여줄 수 있는 내용으로 어필하기
- 깔끔하고 견고한 웹 구조 만들기

♡ 작업 과정

시작 전 구상

- UI
 - header / footer 진짜 짧게 / nav / main
 - 내용별로 section
 - 대충 초안 그려보기



- 내용
 - 자기소개 : 안녕하세요. 주니어 프론트엔드 개발자로 재도약하는 이성령입니다 :)
 - 목표
 - 18주간 학습 계획
 - 구체적인 학습계획 구상해서 내용으로 넣어보기

작업 환경

- windows chrome
- vscode
- npm gulp (+sass) 사용

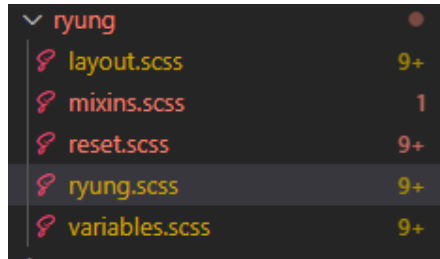
디자인 소스 참조

- 웹 폰트 : <https://fonts.google.com/> / <https://noonnu.cc/>
- 아이콘 : <https://fonts.google.com/icons> / <https://icons8.com/>
- 색 조합 참조 : <https://colorhunt.co/palette/f7fbfcd6e6f2b9d7ea769fcd>

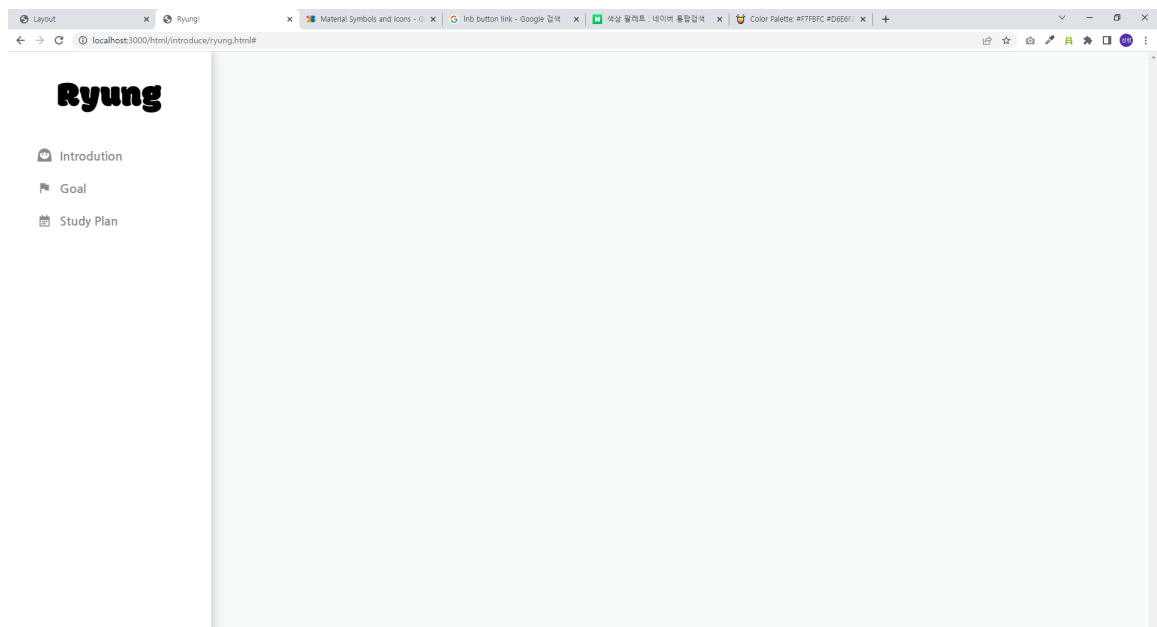


페이지 제작

- 저장소 구성
 - 단일 페이지, 단일 스타일시트로 구성하고자 했습니다
 - LSR.html
 - css/LSR.css
 - 개발 환경에서는 용도 별 scss 파일로 제작하고 조합했습니다



- images/
- reset.scss 와 variables.scss 구성
 - 참조할 색상표와 사용이 예상가는 태그들의 reset을 일부 작성하고 시작했습니다
 - `margin:0 padding:0` 지정을 우선으로 했습니다
- aside + gnb 구현



- 타이틀과 네비게이션 완성
- aside는 스크롤과 상관없이 항상 화면에 가득차 보일 수 있도록 `position: fixed` 와 `height: 100vh` 를 이용했습니다.
- gnb 안에 a태그와 id를 이용해서 각 섹션별로 이동 가능하도록 했습니다

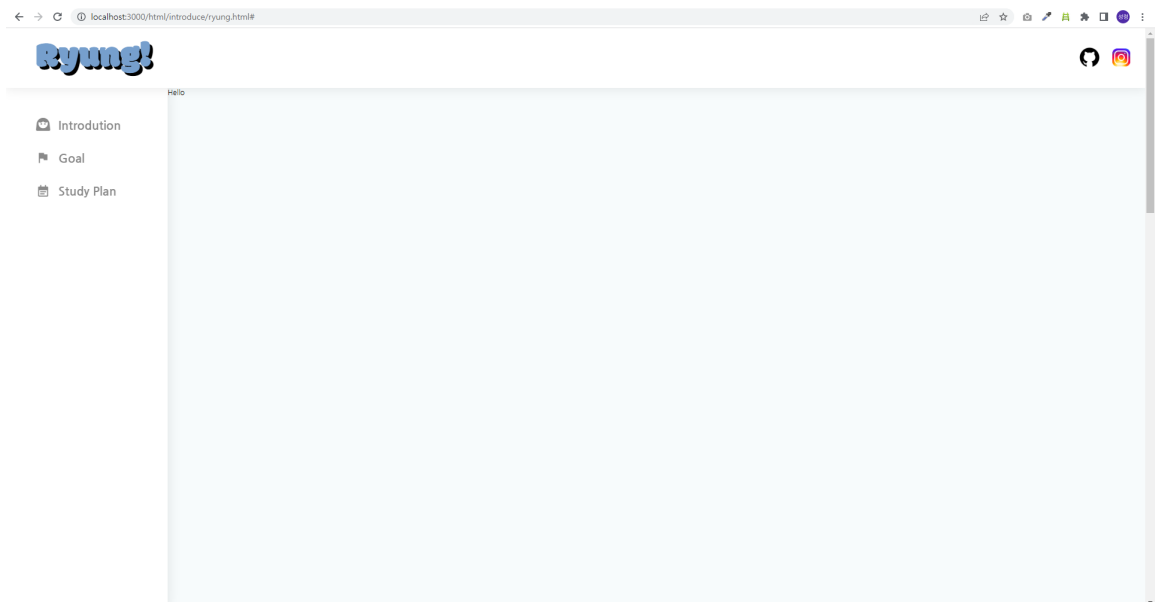
```
<nav class="gnb">
  <h2 class="blind">메뉴 보기</h2>
  <ul>
    <li>
      <a href="#introduction" class="gnb_menu">
        <span class="gnb_menu_ico material-symbols-outlined">face_3</span>
        Introduction
      </a>
    </li>
```

```

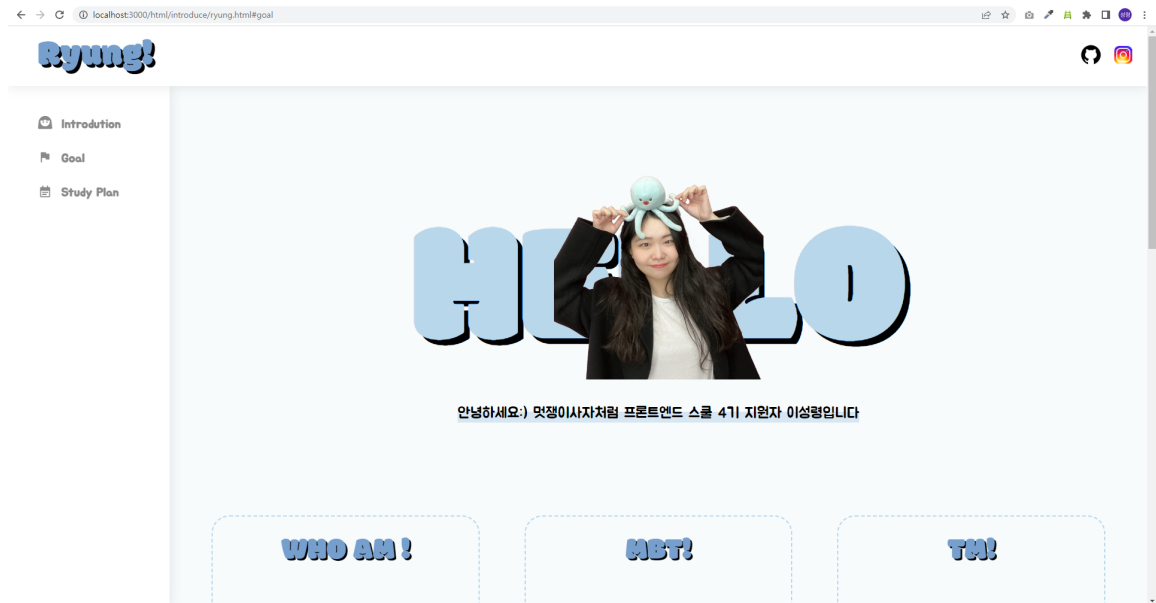
        <li>
            <a href="#goal" class="gnb_menu">
                <span class="gnb_menu_ico material-symbols-outlined">flag</span>
                Goal
            </a>
        </li>
        <li>
            <a href="#studyplan" class="gnb_menu">
                <span class="gnb_menu_ico material-symbols-outlined">event_note</span>
                Study Plan
            </a>
        </li>
    </ul>
</nav>

```

- gnb에 `opacity` 와 `transform` 을 이용해 hover 애니메이션을 삽입했습니다
- 타이틀 부분을 nav 에서 분리해 header 안으로 재구성



- 시멘틱태그가 `header` - `aside` - `main` 순으로 정렬되게 재구성
 - 앞에서 aside 안으로 타이틀을 넣고나니 header의 역할과 위치가 애매모호해져서 UI를 변경하게 되었습니다
- header 우측에 심심함을 덜어줄 sns 영역 추가
- `section#introduction`



◦ 인트로 영역

- `flex` 를 이용해 좌우 중앙 배치했습니다
- 짧은 인사 멘트에 하이라이트 효과를 넣기 위해 `background: linear-gradient` 를 이용했습니다



◦ 세가지 구성요소

- `flex` 를 이용해 3등분으로 정렬했습니다
- Who am I : q&a 형태임을 이용해 `dl` `dt` `dd` 이용
- MBTI : 각 특징을 해시태그 형식으로 짧게 나열하기 위해 가볍게 `span` 이용
- TMI : 항목이므로 `ul` `li` 이용
- 가상 선택자를 이용하여 가벼운 댓글모양, 체크모양을 만들었습니다

- **section#goal** 구성



멋사의 일원이 되어 이루고자 하는 것은?

**완성도 높은 주니어 프론트엔드 개발자가 되어
몸 담았던 업계에서 재도약 하는 것!**

제가 멋쟁이 사자처럼에서 교육을 받고 싶은 이유는 혼자 공부하면서 반복되는 의문인 '이걸 지금 이렇게 공부하는게 맞는건가?'에 대한 조연이 간절하기 때문입니다. 의욕과는 다르게 혼자 공부하면서 중구난방 불규칙하게 뒤는 방식에서 벗어나 제대로 된 아픔을 받고 싶습니다. 그리고 혼자라면 쉽게 경험할 수 없는 팀 프로젝트의 기회를 놓치고 싶지 않습니다.

멋사의 일원이 되어, 18주간의 짧고도 긴 시간동안 배우는 것들을 모두 내 것으로 만들어서 기초도 탄탄하고 실무에 밀고 던져질 수 있을 만한, 경쟁력 있는 프론트엔드 신입이 되고 싶습니다. 그러기 위해서 매일매일 충분한 연습과 복습을 진행하고, 온전히 나의 시간을 태우면서 이 과정을 들을 것 입니다.

- **flex** 를 이용해 상하 중앙이 되도록 정렬했습니다
- **img** 를 좌측에 위치시키고, 텍스트 영역이 가변으로 움직이도록 했습니다

- **section#studyPlan** 구성

교육 커리큘럼과 이전 기수 회고 글을 참고하면서 주차별로 학습 일정을 세웠습니다.

0주차 (11/21 - 11/27)

[개인] 개발 블로그 재 정비
[개인] git 재 정비
[개인] Vanilla JS 토이 프로젝트 무작정 따라해보기 : 링크

1 - 5주차 (11/28 - 1/1)

HTML/CSS 커리큘럼 충실하게 듣기
[개인] 모던 자바스크립트 튜토리얼 완독하기 : 링크
[개인] Vanilla JS 토이 프로젝트 무작정 따라해보기 : 링크

6 - 11주차 (1/2 - 2/12)

JavaScript 커리큘럼 충실하게 듣기
[개인] CS 지식 간간히 채우기 : 링크
[개인] React 기초 강의 무작정 따라해보기 : 링크

12 - 14주차 (2/13 - 3/5)

React 커리큘럼 충실하게 듣기
[개인] JavaScript 공부 내용 회고하기

15 - 18주차 (3/6 - 3/29)

팀 프로젝트에 몰입하기

꾸준하고 성실하게 학습 참여하기

HTML, CSS 교육 동안은 정규 강의 시간 외에 JS, React 틈틈히 연습하기

팀 프로젝트 돌입 전에 충분한 역량 키우기

공부는 체력싸움이다! 운동 꾸준히 하기

가상시간, 취침시간 규칙적으로 맞추기

18주 동안 공부에만 몰입하기

- **grid** 를 이용해 2*2 배열로 정렬했습니다
- 리스트 앞에 **border** 와 가상선택자를 활용해서 간단한 포인트를 주었습니다

- 전체 검토

- ✓ 접근성 확인
- ✓ class 네이밍, depth 확인
- ✓ 불필요한 style 사용 내역 없는 지 확인
- ✓ 크로스 브라우징

✓ window-chrome

✓ window-microsoft-edge

♡ 회고

- 목표의 80% 구현했다고 생각합니다
- 시간 분배에 어려움이 있었습니다
 - 하루를 온전하게 사용할 수 있는 날은 수요일과 목요일 이틀. 그것도 영상과제와 함께 해야하다보니 시간의 압박이 있었습니다
 - 개인 공부는 잠시 미뤄두고 과제의 우선순위를 1순위로 올려 이틀이라도 온전하게 몰입했습니다
 - 결론적으로 목표했던 반응형은 만들지 못했는데, 페이지에 `min-width` 를 설정하여 깨지지 않도록 방어했습니다
- 디자인 구성이 아쉽습니다
 - 디자인에는 소질이 없다보니 처음 페이지를 구상해야하는 단계에 놓였을 때 펜을 잡고 어쩔 줄 몰라했던 것 같습니다 ㅎㅎ;
 - 일단은 필수적으로 들어가야하는 내용들이 있으니 이를 항목화하고 나열하는 구성으로 레이아웃부터 짰 뒤, 컴포넌트마다 하나씩 상상하고 바로 구현하는 식으로 작업했습니다
 - 노력은 했는데 결과물을 보니 텅 비어 아쉽습니다
- `grid` 사용 시도에 만족합니다
 - 사실 `grid` 는 ie 크로스 브라우징 문제로 실무동안 거의 사용하지 않던 속성인데, 이번 기회로 속성도 다시 공부하고 직접 적용해보았습니다
- 단조로운 애니메이션 효과를 보며 한계를 느낍니다
 - 인터랙션이 화려한 페이지에 대한 작업 경험이 적어, 상대적으로 적은 시간 내에 구현하기 어려워하는데 이번 과제를 통해 그 한계를 다시 한 번 느꼈습니다
 - 여유가 된다면 복잡한 애니메이션도 배워보고 싶습니다
- 시간이 추가로 주어진다면 `media-query` 반응형 작업을 적용하고 싶습니다
 - 사실 MWeb 한 벌, PC 한 벌로 따로 따로 작업하는 것이 손에 익어 미디어쿼리 대응에 손이 느린 편인데 이번 과제에서도 역시 마지막에 적용할 여유가 남지 않았습니다
 - 추가 시간이 주어진다면 저는 반응형 추가 작업을 가장 우선으로 진행할 것입니다