



## 3회차 Challenge Mission - 노진현

### 1. swiper 모듈을 이용한 터치 슬라이드 구현하기

#### 설치

```
npm install swiper --save
```

#### 적용

```
<div id={id} className="row_posters">
  <Swiper slidesPerView={`${isLargeRow ? "7" : "6"}`} spaceBetween={20}>
    {movies.map((movie) => (
      <SwiperSlide key={movie.id}>
        <img
          className={`row_poster ${isLargeRow && "row_posterLarge"}`}
          src={`https://image.tmdb.org/t/p/original${
            isLargeRow ? movie.poster_path : movie.backdrop_path
         }`}
          alt={movie.name}
          onClick={() => handleClick(movie)}
        />
      </SwiperSlide>
    ))}
  </Swiper>
</div>
```

조금 더 자연스러운 디자인을 위해서는 몇 가지 요소를 수정해 줄 필요가 있을 것 같다.

### 2. GitHub Pages 를 이용해서 리액트 애플리케이션 배포하기

## Git Repository 생성 및 로컬과 연결

레포지토리 네임은 url 경로가 된다. netflix-clone이라는 이름으로 생성

→ <http://nohow.github.io/netflix-clone> 이런식으로 페이지가 생성될 예정

## GitHub Pages 모듈 설치

```
npm install gh-pages
```

## package.json에 homepage 추가

```
"homepage": "http://nohow.github.io/netflix-clone"
```

homepage에 입력한 주소 경로가 PUBLIC\_URL이 된다.

## package.json scripts 수정

```
"scripts": {  
  "start": "react-scripts start",  
  "build": "react-scripts build",  
  "test": "react-scripts test",  
  "eject": "react-scripts eject",  
  "predeploy": "npm run build", //추가  
  "deploy": "gh-pages -d build" //추가  
},
```

**react router dom 을 사용했는데, basename을 추가해야 제대로 라우팅 할 수 있다.**

basename은 root URL로 위에서 추가하나 homepage 주소와 동일하게 설정해야한다.

모든 Route의 route URL설정을 위해서 최상위인 `BrowserRouter` 에 속성으로 부여한다.

```
<BrowserRouter basename={process.env.PUBLIC_URL}>  
  <App />
```

```
</BrowserRouter>,
```

이렇게 설정해주어야 root URL을 `/` 가 아닌 내가 설정한 `netflix-clone` 으로 인식할 수 있다.

## npm run build & npm run deploy

`npm run build` 를 통해서 build 폴더 & 파일을 생성하고,

`npm run deploy` 를 통해서 전개해준다.

---

## 오류. 빈화면 출력

- 제대로 설정했다고 생각했는데, 빈화면이 표시되었다.
- Router 주소 문제라고 생각하고 주소를 수정해보거나 `.env` 파일을 생성해서 `PUBLIC_URL` 을 설정하는 등 시도를 해보았는데 계속해서 빈화면이 표시되었다.

## 해결

- github에서 gh-pages 모듈을 추가할 때 자동으로 gh-pages branch가 생성되고 GitHub Pages Source Branch로 gh-pages가 세팅되어있었다.
- 이를 간과하고 프로젝트 파일들을 모두 main에 push했기 때문에 빈화면이 출력되는 것이 당연했다.
- gh-pages branch에 프로젝트 파일을 push하거나 source branch를 main 으로 수정하는 것으로 해결가능했다.

## GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

✓ Your site is published at <https://nohow.github.io/netflix-clone/>

### Source

Your GitHub Pages site is currently being built from the main branch. [Learn more.](#)

🔗 Branch: main ▾

📁 / (root) ▾

Save

### Theme Chooser

Select a theme to publish your site with a Jekyll theme. [Learn more.](#)

Choose a theme

### Custom domain

Custom domains allow you to serve your site from a domain other than nohow.github.io. [Learn more.](#)

Save

Remove

### ☒ Enforce HTTPS

— Required for your site because you are using the default domain (nohow.github.io)

HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site. When HTTPS is enforced, your site will only be served over HTTPS. [Learn more.](#)



### • Publish privately to people with read access to this repository

Try risk-free for 30 days using a GitHub Enterprise organization, or [learn more about changing the visibility of your GitHub Pages site.](#) ✕