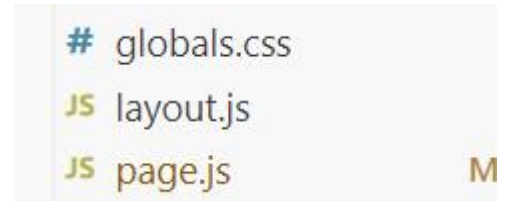
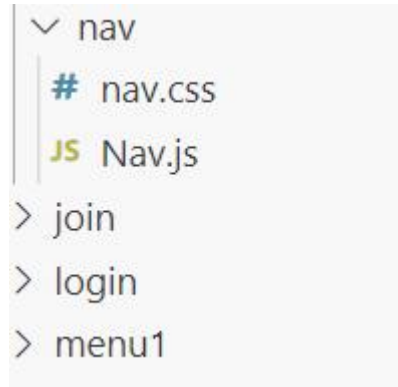
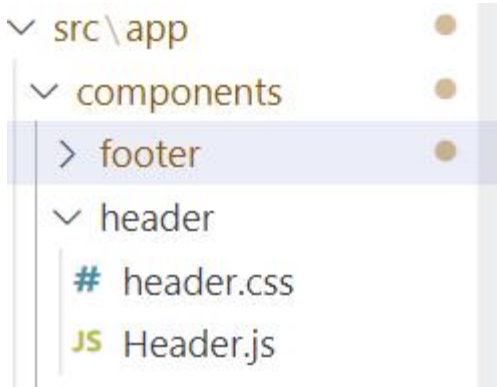
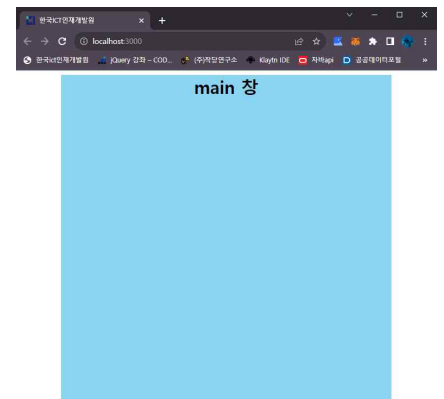
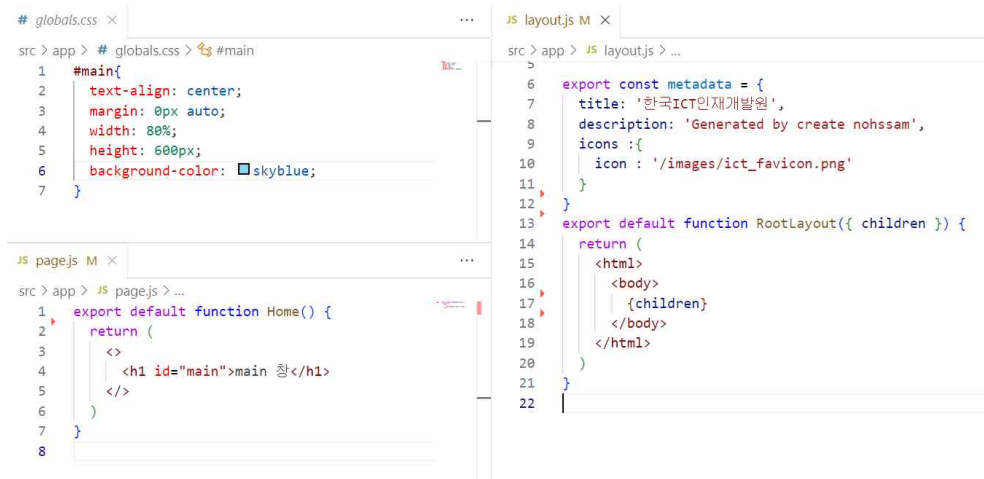


두 번째 프로젝트

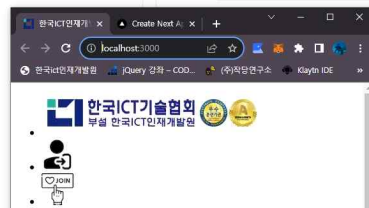
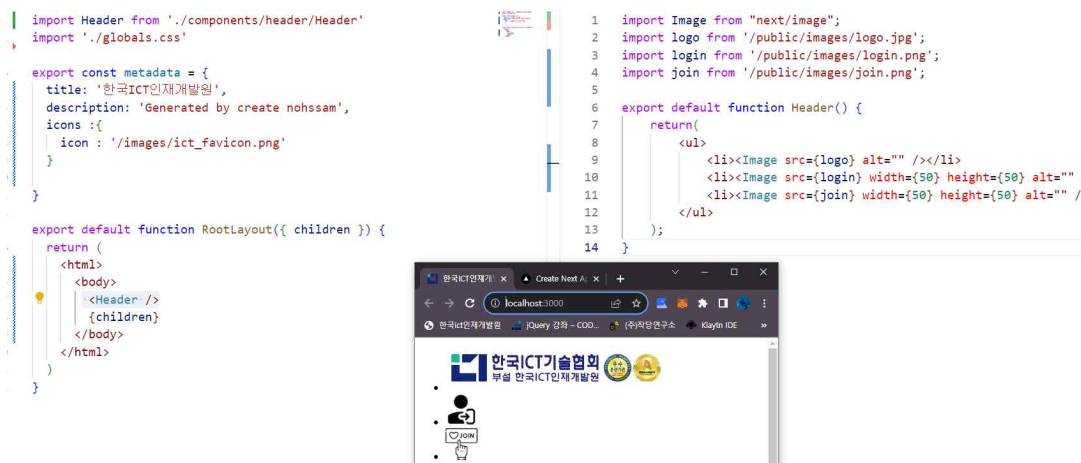
Component 들의 위치와 CSS 위치



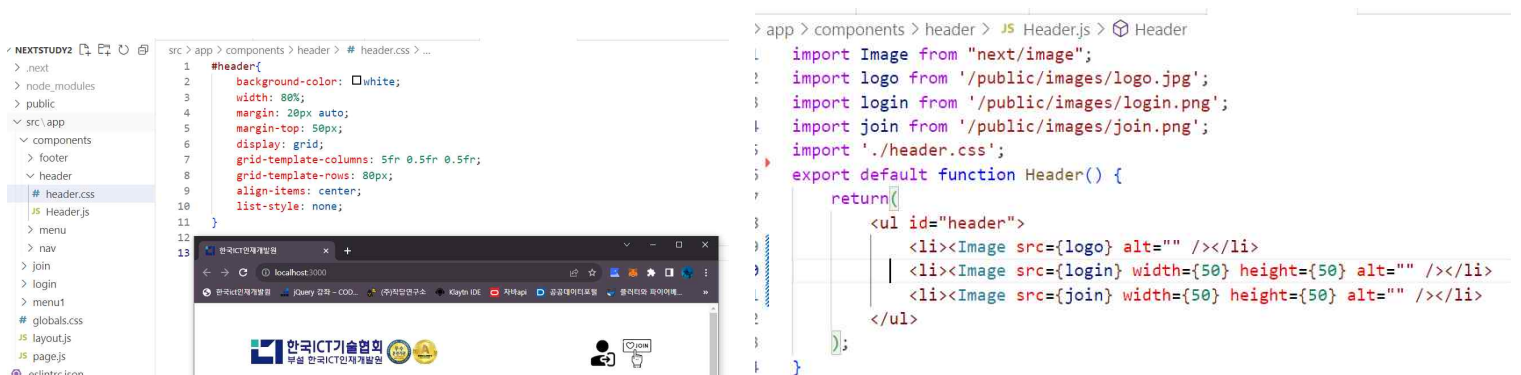
Main 부분



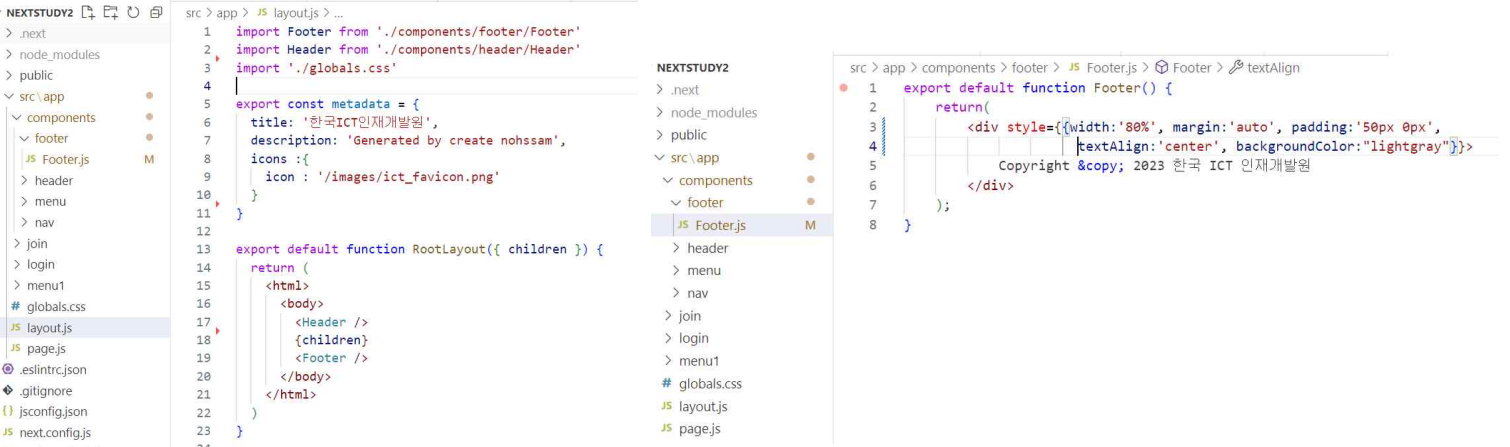
Header 부분



CSS 적용



footer 부분



Nav 부분

<pre>/* 전체영역에서 여백을 없애줌 */ * { margin: 0; padding: 0; } /* ul li태그에 리스트 스타일을 없앴 */ ul li{ list-style: none; } /* a태그에 텍스트 밑줄을 없애고 색상을 #333 */ a { text-decoration: none; color:white; } /* 글자크기를 16px 맑은 고딕 굵게하고 width넓이 700, 높이 50만큼 배경색 은 #ccc, 글자색은 검정색, 라인높이50px menu박스 가운데정렬, 글자가운데 정렬 */ #menu { font:bold 16px "malgun gothic"; width:80%; height:50px; background: #4369f1; color:white; line-height: 50px; margin:0 auto; text-align: center; } /* menu태그 자식의 ul의 자식 li를 왼쪽정렬과 넓이 140설정 */ #menu > ul > li { float:left; width:20%; position:relative; } #menu > ul > li > ul { width:100%; display:none; position: absolute; font-size:14px; background: skyblue; } #menu > ul > li:hover > ul { display:block; } #menu > ul > li > ul > li:hover { background: rgb(88, 114, 185); transition: ease 1s; }</pre>	<pre>import Link from 'next/link'; import './nav.css'; export default function Nav() { return(<div id="menu"> MENU1 <Link href="/menu1/1">SUB_MENU</Link> <Link href="/menu1/2">SUB_MENU2</Link> <Link href="/menu1/3">SUB_MENU3</Link> MENU2 SUB_MENU SUB_MENU2 SUB_MENU3 MENU3 SUB_MENU SUB_MENU2 SUB_MENU3 MENU4 SUB_MENU SUB_MENU2 SUB_MENU3 MENU5 SUB_MENU SUB_MENU2 SUB_MENU3 </div>); }</pre>
---	---

MENU1

MENU2

MENU3



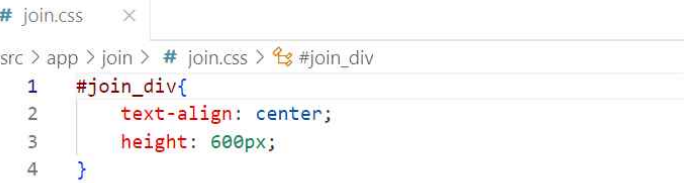

MENU4

MENU5

main 창

Header Link 처리

	<pre>import Image from "next/image"; import logo from '/public/images/logo.jpg'; import login from '/public/images/login.png'; import join from '/public/images/join.png'; import './header.css'; import Link from "next/link"; export default function Header() { return(<ul id="header"> <Link href="/"> <Image src={logo} alt="" /></Link> <Link href="/login"><Image src={login} width={50} height={50} alt="" /></Link> <Link href="/join"><Image src={join} width={50} height={50} alt="" /></Link>); }</pre>
--	--

Nav Link 처리

```
import Manu1 from '@app/components/menu/Manu1'
import Manu2 from '@app/components/menu/Manu2'
import Manu3 from '@app/components/menu/Manu3'
import './menu1.css'
// Layout은 반드시 인자(props)를 받아야 한다.
export default function Layout(props) {
  return(
    <>
      {props.params.id === '1' ? <Manu1 /> : props.params.id === '2'? <Manu2 /> : <Manu3 />}
    </>
  )
}
```

The screenshot displays a code editor with three files open:

- Nav.js** (src > app > components > nav > JS Nav.js > ...):

```
1 import Link from 'next/link';
2 import './nav.css';
3 export default function Nav() {
4   return(
5     <div id="menu">
6       <ul>
7         <li>MENU1
8           <ul>
9             <li><Link href="/menu1/1">SUB_MENU</li>
10            <li><Link href="/menu1/2">SUB_MENU2</li>
11            <li><Link href="/menu1/3">SUB_MENU3</li>
12          </ul>
13        </li>
14        <li><a href="#">MENU2</a>
15          <ul>
16            <li><a href="#">SUB_MENU</a></li>
17            <li><a href="#">SUB_MENU2</a></li>
18            <li><a href="#">SUB_MENU3</a></li>
19          </ul>
20        </li>
21        <li><a href="#">MENU3</a>
22          <ul>
23            <li><a href="#">SUB_MENU</a></li>
24            <li><a href="#">SUB_MENU2</a></li>
25            <li><a href="#">SUB_MENU3</a></li>
26          </ul>
27        </li>
28      </ul>
29    </div>
30  )
31}
```
- Manu1.js** (src > app > components > menu > JS Manu1.js > Manu1):

```
1 export default function Manu1() {
2   return(
3     <h2> 메뉴 1 - 1 </h2>
4   );
5 }
```
- Manu2.js** (src > app > components > menu > JS Manu2.js > Manu2):

```
1 export default function Manu2() {
2   return(
3     <h2> 메뉴 1 - 2 </h2>
4   );
5 }
```
- Manu3.js** (src > app > components > menu > JS Manu3.js > Manu3):

```
1 export default function Manu3() {
2   return(
3     <h2> 메뉴 1 - 3 </h2>
4   );
5 }
```

===== AXIOS =====

Axios는 브라우저, Node.js를 위한 Promise API를 활용하는 HTTP 비동기 통신 라이브러리이다.

쉽게 말해서 백엔드랑 프론트엔드랑 통신을 쉽게하기 위해 Ajax와 더불어 사용한다.
이미 자바스크립트에는 fetch api가 있지만,
프레임워크에서 ajax를 구현할땐 axios를 쓰는 편이라고 보면 된다.

예제)
먼저 화장품API를 사용할것 이므로 Google에서 Makeup API로 검색해보자!
그리고 나타나는 내용들 중 [Makeup API :makeup-api.herokuapp.com]선택하자!

Example URL이있고 이것을 브라우저에서 확인할수있다.
요청에 따른 결과가 잘 받아지는것을 확인하려면 POSTMAN같은 Tool을활용하면된다
Example URL
http://makeup-api.herokuapp.com/api/v1/products.json?brand=maybelline

메인 page.js에서 내용 추가

```
// useEffect() 함수는 React component가 렌더링 될 때마다
// 특정 작업(Sied effect)을 실행할 수 있도록 하는 리액트 Hook입니다.
// 여기서 Side effect는 component가 렌더링 된 이후에
// 비동기로 처리되어야 하는 부수적인 효과들을 뜻합니다.

import { useEffect } from "react"
import axios from "axios"

export default function Home() {
  const API_URL = "/api/v1/products.json?brand=maybelline";
  const getData = () => {
    axios.get(
      // 정보를 얻기 위한 서버 주소
      API_URL
    )
    // 성공 후 일처리를 하고자 할때 then() 안 일처리 작성
    // then(data)는 받은 정보를 의미한다.
  ).then(data)
}

// 렌더링 될때마다 실행
// useEffect(() => { });

// 최초 한번만 실행할때
useEffect(() => {
  getData();
},[]);

return (
  <>
  <h1 id="main">main 창</h1>
  </>
)
```

