



웹 어플리케이션 응용


미니 블로그


# 글 보기 화면



내 Blog

홈

글 목록보기


바로 글쓰기

글 작성하기

"i love **blog**, which is my favorite **place**."


새 글

현재 시간: 오전 7:04:38




안녕하세요, 일상을 기록하는 라이프로그 블로그입니다.

안녕하세요, 소플입니다. 이번 글에서는 리액트에서 리스트를 렌더링하는 방법에 대해서 배워보겠습니다. 리스트를 렌더링하기 위해서는 자바스크립트 배열에서 제공하는 map함수를 사용합니다.



리액트의 조건부 렌더링이란?


안녕하세요, 소플입니다. 이번 글에서는 리액트의 조건부 렌더링에 대해서 배워보도록 하겠습니다. 조건부 렌더링은 말 그대로 조건에 따라서 렌더링을 다르게 한다는 의미입니다.




리액트 Hook에 대해서 배워보까요?





# 글 보기 화면



내 Blog

 홈

 글 목록보기

 바로 글쓰기


글 작성하기


"i love **blog**, which is my **favorite place**."


뒤로가기


안녕하세요, 일상을 기록하는 라이프로그 블로그입니다.

Write, Create, Describe











댓글

 실제로 개발하다보면 map함수를 진짜 많이 쓰는 것 같아요 😊


# 글 작성화면



내 Blog

 홈

 글 목록보기

 바로 글쓰기

글 작성하기

"i love **blog**, which is my **favorite place**."

뒤로가기

글 작성하기

업로드 날짜: 2024. 5. 1.

Title

Content

글 작성하기



# Render 구조

- ▼ BrowserRouter ⚠
  - ▼ Router
    - ▼ Navigation.Provider
      - ▼ Location.Provider
        - ▼ Routes
          - ▼ RenderedRoute
            - ▼ Route.Provider
              - ▼ MainPage
                - ▼ styled.div
                  - ▶ Select
                - ▼ styled.div
                  - ▼ styled.div
                    - Clock
                    - styled.div
                  - ▼ PostList
                    - ▶ styled.div

MainList.jsx

<Select/>

MainPage.jsx

<Clock/>

PostViewPage.jsx

PostWritePage.jsx

UI

<Button/>

<Story/>

<Top/>

JQUERY

# Postitem.jsx

## TitleFlex

사진 데이터 JSON에서 가져오기



Styled Component 이용

```
const TitleFlex = styled.div`
  gap: 32px;
  display: flex;
`;

const TitleText = styled.div`
  font-size: 16px;
  width: 980px;
  . . .
`;
```

```
<Wrapper onClick={onClick}>
  . . . . . <TitleFlex>
  . . . . . | . . . . . { /* <img src={` ${process.env.PUBLIC_URL}/img/${dummy.images[0].text}`} alt="" ./> */ }
  . . . . . | . . . . . { /* <img style={styles.imgsize} src={post.comments[0].i} alt="testing" ./> */ }
  . . . . . | . . . . . { /* <img src={path + ` ./img/daniil-silantev-GjmORWerlXk-unsplash.jpg`} alt="imgtest" ./> */ }
  . . . . . | . . . . . <>
  . . . . . | . . . . . <div>
  . . . . . | . . . . . <img style={styles.imgsize} src={post.comments[0].i} alt="testing" ./>
  . . . . . | . . . . . </div>
  . . . . . | . . . . . </>
  . . . . . <TitleText> . . . <h2> {post.title} </h2> <p> {post.content} </p> </TitleText>
```



# MainList.jsx

## JSX 구조

<Story/> 컴포넌트 (사진 컴포넌트) 조합

<Button/> 컴포넌트 조합

```
function Select(props) {
  ... const navigate = useNavigate();

  ... return (
    ...   <Wrapper>
    ...     <Story></Story>
    ...     <div>
    ...       <h2 style={styles.title}>내 Blog</h2>

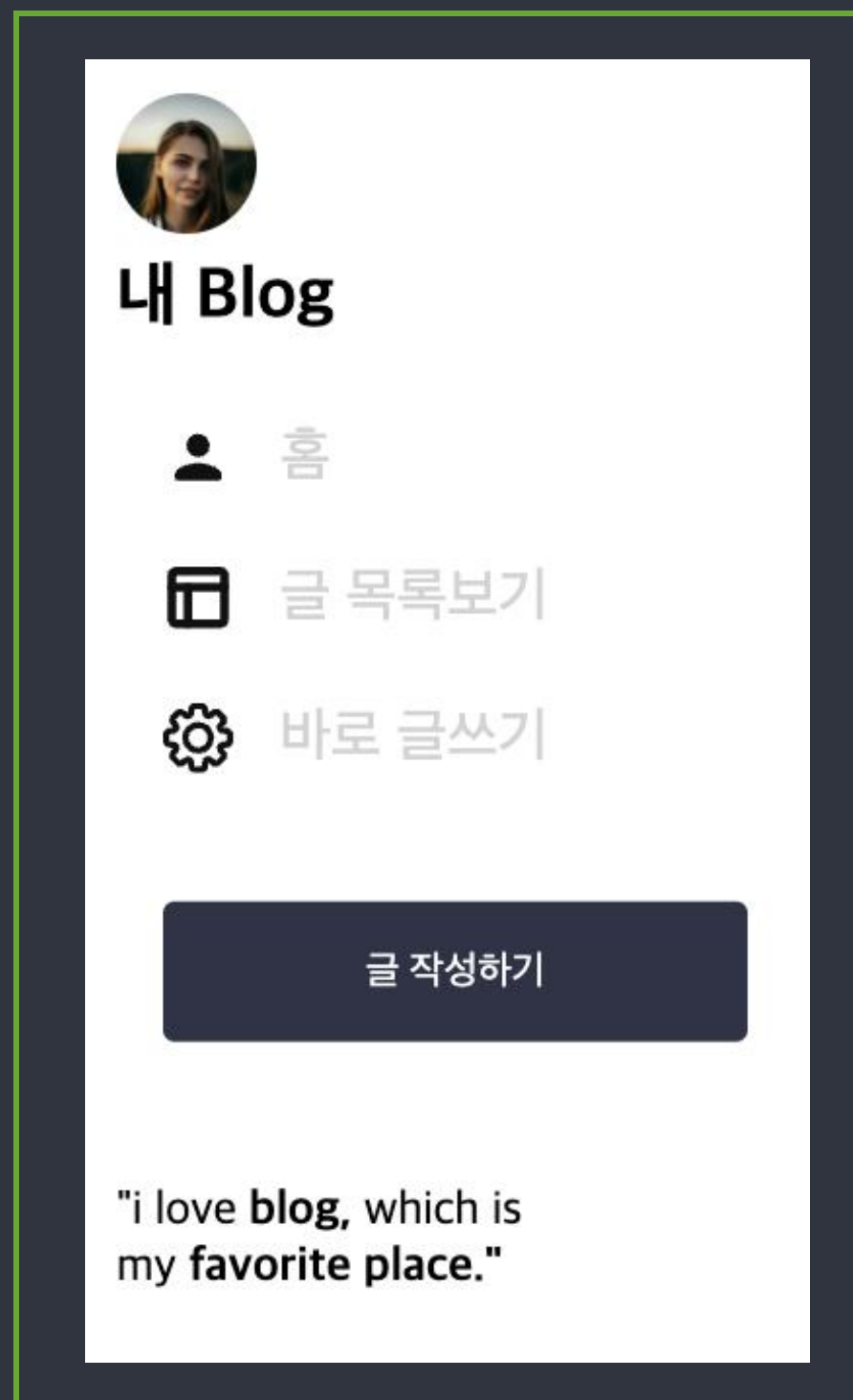
    ...       <div style={styles.flex}>
    ...         
    ...         <P> <p>홈</p></P>
    ...       </div>
    ...       <div style={styles.flex} onClick={() => { navigate('/') }}>
    ...         
    ...         <P><p>글 목록보기</p></P>
    ...       </div>
    ...       <div style={styles.flex} onClick={() => { navigate('/write') }}>
    ...         
    ...         <P><p>바로 글쓰기</p></P>
    ...       </div>
    ...     </div>
    ...     <div style={styles.button}>
    ...       <Button title="글 작성하기" onClick={() => { navigate('/write') }}></Button>
    ...     </div>

    ...     <p style={styles.title}>"i love <strong>blog,</strong> which is <br /> my <strong>favorite place."</strong></p>
    ...   </Wrapper>
    ... )
}
```

# MainList.jsx

## Select 컴포넌트

사이드바



```
function Select(props) {
  ... const navigate = useNavigate();

  ... return (
    ... <Wrapper>
    ...   <Story></Story>
    ...   <div>
    ...     <h2 style={styles.title}>내 Blog</h2>

    ...     <div style={styles.flex}>
    ...       
    ...       <P> 홈<p></p></P>
    ...     </div>

    ...     <div style={styles.flex} onClick={() => { navigate('/') }}>
    ...       
    ...       <P><p>글 목록보기</p></P>
    ...     </div>

    ...     <div style={styles.flex} onClick={() => { navigate('/write') }}>
    ...       
    ...       <P><p>바로 글쓰기</p></P>
    ...     </div>
    ...   </div>

    ...   <div style={styles.button}>
    ...     <Button title="글 작성하기" onClick={() => { navigate('/write') }}></Button>
    ...   </div>

    ...   <p style={styles.title}>"i love <strong>blog,</strong> which is <br /> my <strong>favorite place."</strong></p>
    ... </Wrapper>
    ... )
}
```



# MainPage.jsx

## JSX 구조

실시간 시계

```
function Clock() {  
  ... const [time, setTime] = useState(new Date());  
  
  ... useEffect(() => {  
    ... const interval = setInterval(() => {  
      ... setTime(new Date());  
      ... }, 1000);  
  }  
}
```

Key 고유의 키 값 부여, 중복 키 해결

```
... const navigate = useNavigate();  
... // const element = <h2>현재 시간: {new Date().toLocaleTimeString()}</h2>  
  
... return (  
  ... <Wrapper>  
    ... <Select />  
    ... <Container>  
      ... <Enter>  
        ... <h2>새 글</h2> <Clock />  
      ... </Enter>  
      ... <Line />  
      ... <PostList posts={data} onClickItem={(p) => { navigate('/post/' + p.id +  
        ... 'database') }}></PostList>  
    ... </Container>  
  ... </Wrapper>  
);
```