




React Mini blog

2024 년

미니 블로그

■ 주요 기능

- 글목록 보기
- 글 보기 (상세) 기능
- 댓글 보기 기능
- 글작성 기능
- 댓글작성 기능

 **Prettier - Code formatter** 499ms
Code formatter using prettier
Prettier

default format

사용자 작업 영역

- 텍스트 편집기 (1)
- 기능 (1)
 - Notebook (1)
- 확장 (4)
 - ESLint (1)

Editor: Default Formatter
다른 모든 포맷터 설정보다 우선하는 기본 포맷터를 정의합니다. 포맷터를

Prettier - Code formatter

format on save

사용자 작업 영역

- 텍스트 편집기 (4)
 - 서식 (2)
 - 파일 (2)
- 기능 (1)
 - Notebook (1)
- 확장 (1)
 - ESLint (1)

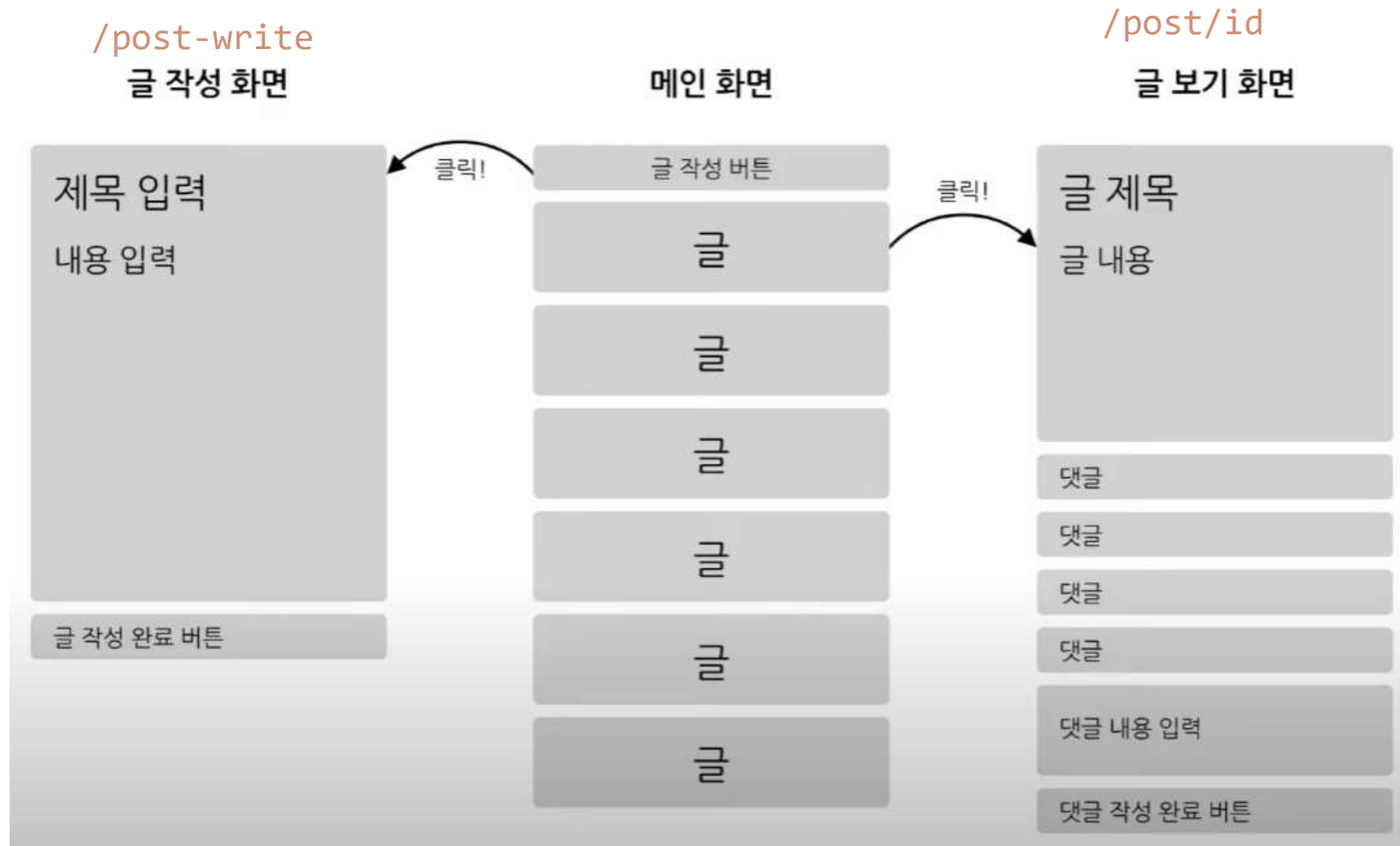
Editor: Format On Save Mode
저장 시 포맷이 전체 파일을 포맷하는지 또는 수정 내용만 포맷하는지를 제어합니다. [Editor: Format On Save](#)

file

Editor: Format On Save

☒ 저장 시 파일의 형식을 지정합니다. 포맷터를 사용할 수 있어야 하며 편집기를 종료하면 안 됩니다. [Files:](#) 일은 명시적으로 저장될 때만 형식이 지정됩니다.

미니 블로그



미니 블로그

■ 필요 패키지

- react-router-dom
- styled-components
- npm install --save react-router-dom styled-components
- npm install --save react@18.3.1 react-dom@18.3.1

```
D:\Wkdt2024\Workspace\react\myblog>npm list
myblog@0.1.0 D:\Wkdt2024\Workspace\react\myblog
+-- cra-template@1.2.0
+-- myblog@0.1.0 -> .
+-- react-dom@18.3.1
+-- react-router-dom@7.0.2
+-- react-scripts@5.0.1
+-- react@18.3.1
+-- styled-components@6.1.13
```

미니 블로그

- 글 목록 보기 기능 (리스트 형태)
 - PostList, PostListItem
- 글 보기 기능
 - Post
- 댓글 보기 기능
 - CommentList, CommentListItem
- 글 작성 기능
 - PostWrite
- 댓글 작성 기능
 - CommentWrite

미니 블로그

■ 폴더 구성하기

- src

- component

- list : 리스트와 관련된 컴포넌트들을 모아놓은 폴더
 - page : 페이지 컴포넌트들을 모아놓은 폴더
 - ui : ui 컴포넌트들을 모아놓은 폴더

미니 블로그

■ UI 컴포넌트 및 List컴포넌트 구성하기

1. ui 컴포넌트
 - 버튼 컴포넌트
 - text input 컴포넌트
2. List 컴포넌트
 - PostListItem 컴포넌트
 - PostList 컴포넌트
 - CommentListItem 컴포넌트
 - CommentList 컴포넌트
3. 더미 데이터 작성
4. Page 컴포넌트
 - MainPage 컴포넌트
 - PostWritePage 컴포넌트
 - postViewPage 컴포넌트
5. 각 페이지별 경로 구성하기

미니 블로그

- react-router-dom 활용
 - App.js
 - index.js
- 페이지 이동 `useNavigate()`

미니 블로그

■ 빌드와 배포

- npm run build
- serve 패키지 설치
- npm install -g serve

미니 블로그

- Rich Text-Editor(리치 텍스트 에디터)와 WisWig(위즈윅) 에디터
- 리치 텍스트 에디터
 - 사용자는 문서의 스타일, 글꼴, 크기 등을 선택하거나, 강조, 들여쓰기, 링크 등의 일반적인 기능을 수행
 - 사용자가 작성한 문서는 HTML 코드로 저장된다.
- 위지윅 에디터
 - "What You See Is What You Get"의 약자로, 사용자가 작성한 문서가 실제로 출력될 모습과 비슷하게 표시
 - `npm install react-quill`

```

src > JS App.js > ...
1  import logo from './logo.svg';
2  import './App.css';
3  import Editor from './Editor';
4
5  function App() {
6    return (
7      <div className="App">
8        <Editor theme="snow"></Editor>
9      </div>
10   );
11 }
12
13 export default App;
14

```

```

src > JS Editor.js > ...
1  import ReactQuill from "react-quill";
2  import "react-quill/dist/quill.snow.css";
3
4  const Editor = () => {
5    return (
6      <div>
7        <ReactQuill theme="snow" />
8      </div>
9    );
10 };
11
12 export default Editor;
13

```

게시판 작성

- 기능 정의
- 화면 정의
- 개발 환경설정
 - 폴더 구조
 - 필요한 패키지.. 설치..
- 코드작성
- 배포