

SpringBoot기반의 Restful API와 React 연결하기

주요기능

- 회원관리(아이디, 비밀번호, 이름, 주소, role, 등록일)
 - 회원가입(아이디중복체크)
 - 로그인
 - 로그아웃
- 게시판관리(글번호, 제목, 내용, 작성자, 등록일, 수정일)
 - 게시물 전체조회- 아무나 조회
 - 게시물 상세보기-인증된사용자 상세보기
 - 게시물등록-인증된 사용자 만이 등록가능
 - 게시물수정 - 인증된 사용자 + 자신이 쓴 글인 경우만
 - 게시물 삭제- 인증된 사용자 + 자신이 쓴 글인 경우만

React환경 세팅

1. 프로젝트를 생성한다.

```
npm create vite@latest step18-client-jwt
```

2. 생성된 프로젝트로 이동한다.

```
cd .\step18-client-jwt\
```

3. 필요한 lib install한다.

```
npm install react-router-dom
```

```
npm install react-axios
```

package.json파일확인

```

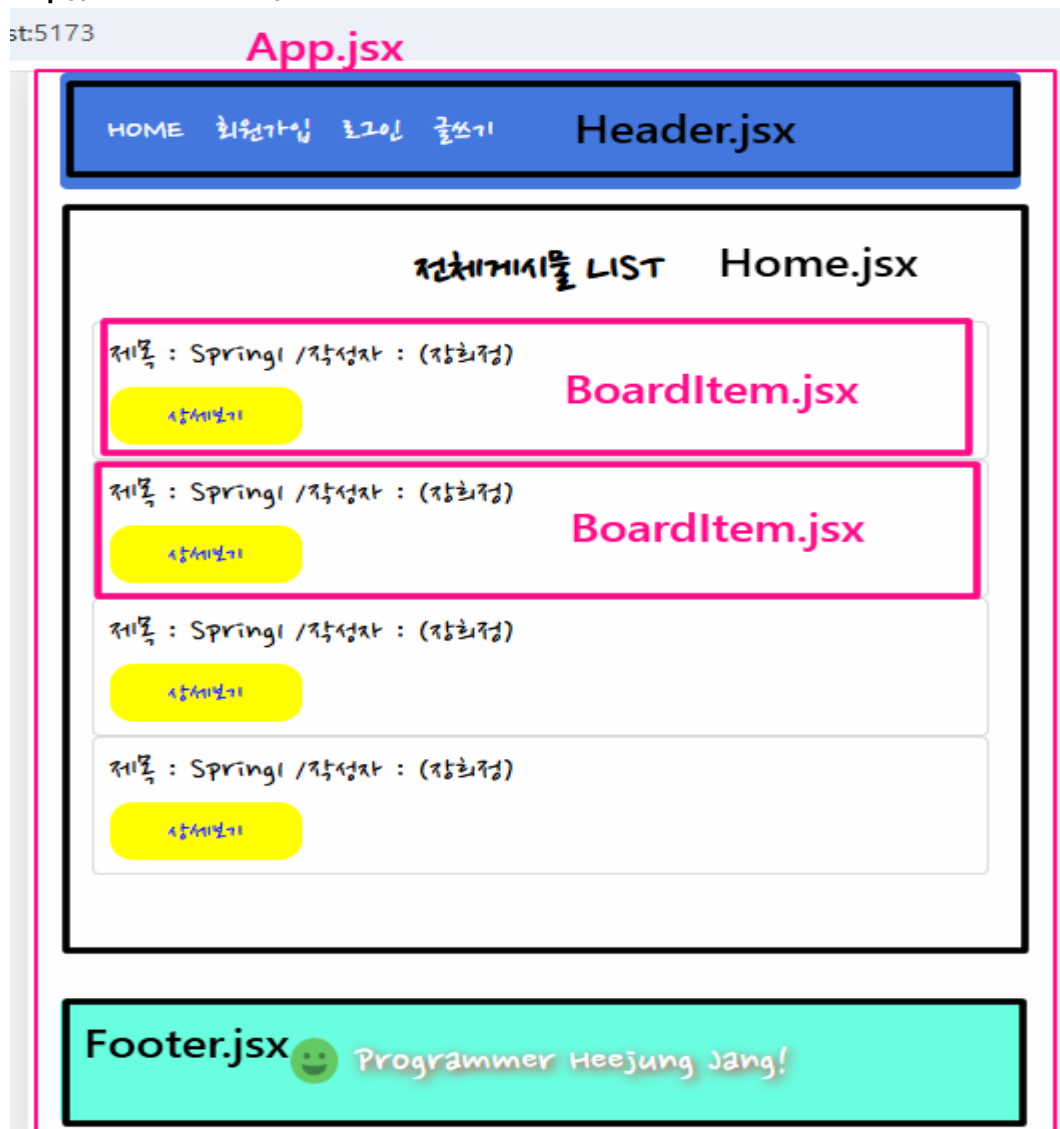
12     "dependencies": {
13         "react": "^18.3.1",
14         "react-axios": "^2.0.6",
15         "react-dom": "^18.3.1",
16         "react-router-dom": "^6.28.0"
17     },

```

전체화면구성

http://localhost:5173/요청 화면

st:5173



<http://localhost:5173/joinForm> 요청화면

App.jsx

HOME 회원가입 로그인 글쓰기 Header.jsx

회원가입 JoinForm.jsx

아이디 아이디중복여부

이름

비밀번호

주소

회원가입 취소

Footer.jsx 😊 Programmer Heejung Jang!

<http://localhost:5173/loginForm> 요청화면

App.jsx

HOME 회원가입 로그인 글쓰기 Header.jsx

로그인 LoginForm.jsx

아이디

비밀번호

로그인 취소

Footer.jsx 😊 Programmer Heejung Jang!

<http://localhost:5173/saveForm> 요청화면

st:5173/saveForm **App.jsx**

HOME 회원가입 로그인 글쓰기 **Header.jsx**

게시물 등록하기 **SaveForm.jsx**

title

writer

content

오늘은 어땠나요?

Footer.jsx 😊 Programmer Heejung Jang!

<http://localhost:5173/board/1> 요청화면

st:5173/board/1 **App.jsx**

HOME 회원가입 로그인 글쓰기 **Header.jsx**

상세보기 **Detail.jsx**

글번호 6

제목 React

작성자 이가현

content

React너무 좋아요

Footer.jsx 😊 Programmer Heejung Jang!

<http://localhost:5173/updateForm/3> 요청화면

st:5173/updateForm/3

App.jsx

HOME 회원가입 로그인 글쓰기 Header.jsx

수정하기 UpdateForm.jsx

title

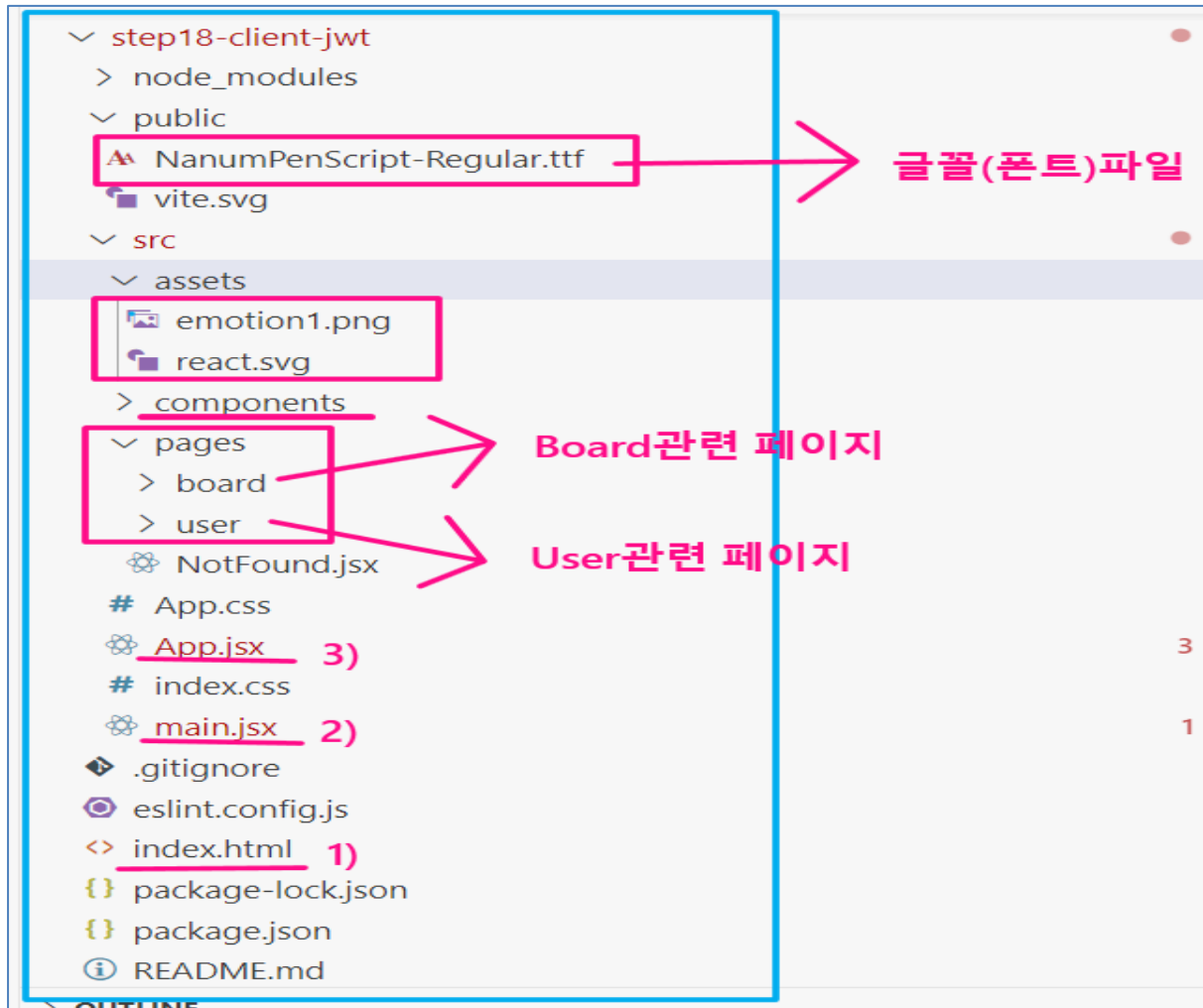
writer

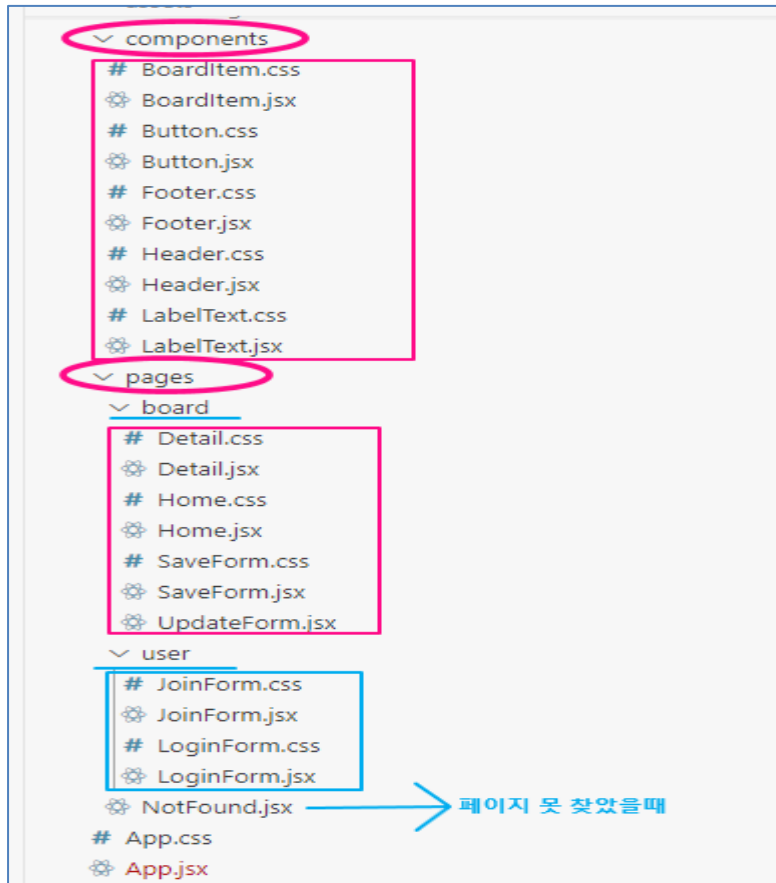
content

오늘은 어땠나요?

Footer.jsx 😊 Programmer Heejung Jang!

프로젝트 Directory구조





먼저, 각 화면을 담당하는 컴포넌트 파일들을 만들어 놓는다.

main.jsx에 BrowserRouter를 연결한다.

```

5   import {BrowserRouter} from "react-router-dom"
6
7   createRoot(document.getElementById('root')).render(
8     <BrowserRouter>
9       <App />
10    </BrowserRouter>
11  )

```

App.jsx파일에 Route를 연결한다.

```

2  import './App.css'
3  import { Route, Routes } from 'react-router-dom'
4  import Home from './pages/board/Home'
5  import SaveForm from './pages/board/SaveForm'
6  import Detail from './pages/board/Detail'
7  import UpdateForm from './pages/board/UpdateForm'
8  import LoginForm from './pages/user/LoginForm'
9  import JoinForm from './pages/user/JoinForm'
10 import NotFound from './pages/NotFound'
11 import Header from './components/Header'
12 import Footer from './components/Footer'
13
14 function App() {
15
16     return (
17         <div className="App">
18             <Header/>
19             <main className="AppManin">
20                 <Routes>
21                     <Route path="/" element={<Home/>} />
22                     <Route path="/saveForm" element={<SaveForm/>} />
23                     <Route path="/board/:id" element={<Detail/>} />
24                     <Route path="/updateForm/:id" element={<UpdateForm/>} />
25                     <Route path="/loginForm" element={<LoginForm/>} />
26                     <Route path="/joinForm" element={<JoinForm/>} />
27                     <Route path="/*" element={<NotFound/>} />
28                 </Routes>
29             </main>
30             <Footer />
31         </div>
32     )
33 }

```

Route 세팅이 완료 된 후에 요청주소로 접속하여 페이지가 잘 열리지는 확인한다.

- 예) <http://localhost:5173/> - Home 컴포넌트
- 예) <http://localhost:5173/saveForm> - SaveForm 컴포넌트
- 예) <http://localhost:5173/board/1> - Detail 컴포넌트
- 예) <http://localhost:5173/updateForm/1> - UpdateForm 컴포넌트
- 예) <http://localhost:5173/loginForm> - LoginForm 컴포넌트
- 예) <http://localhost:5173/joinForm> - JoinForm 컴포넌트

-회원관리 화면 구성

1)회원 가입

user > JoinForm.jsx | JoinForm.css

2) 로그인

user > LoginForm.jsx | LoginForm.css

-게시판관리 화면구성

1) 게시물 등록

board > SaveForm.jsx | SaveForm.css

2) 전체게시물

board > Home.jsx | Home.css

3) 상세보기

board > Detail.jsx | Detail.css

4) 수정하기

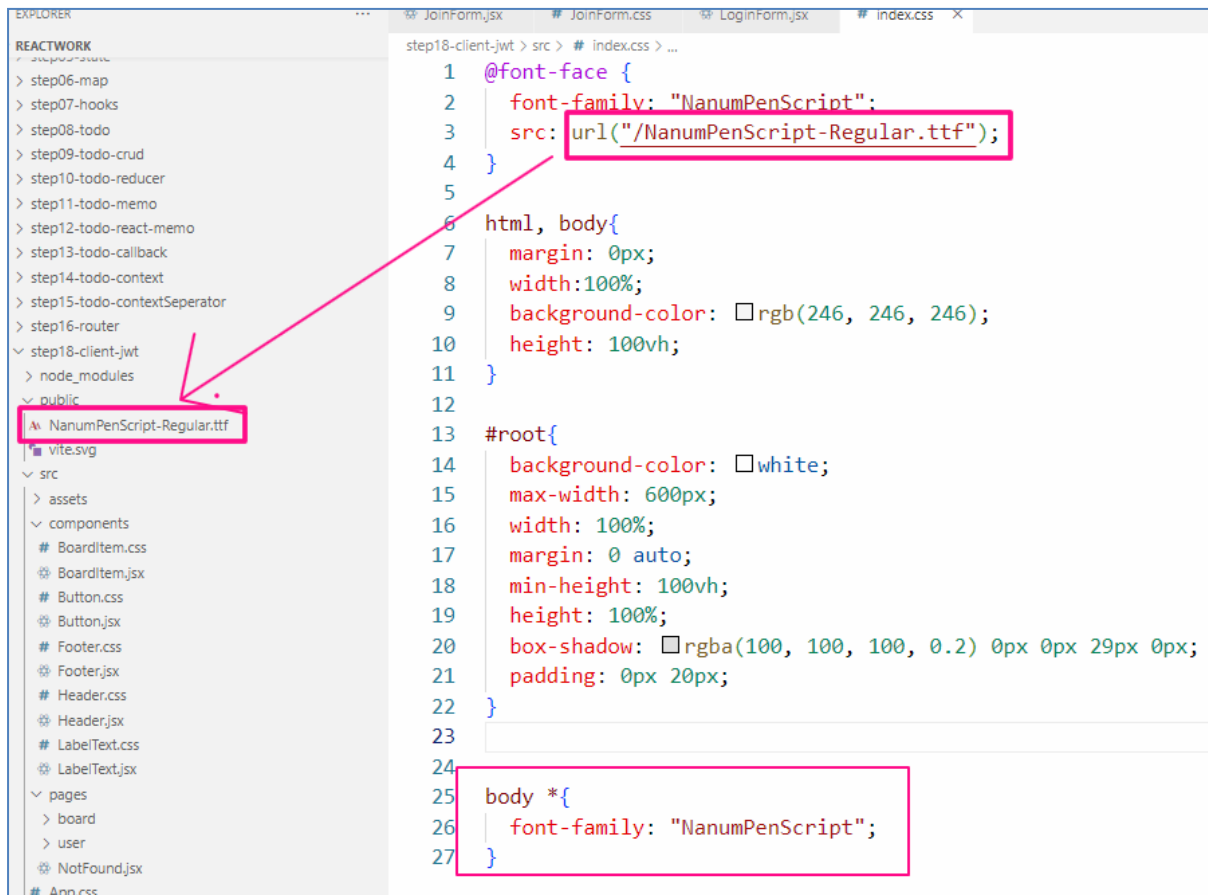
board > UpdateForm.jsx | UpdateForm.css

화면구성 만들어보자!

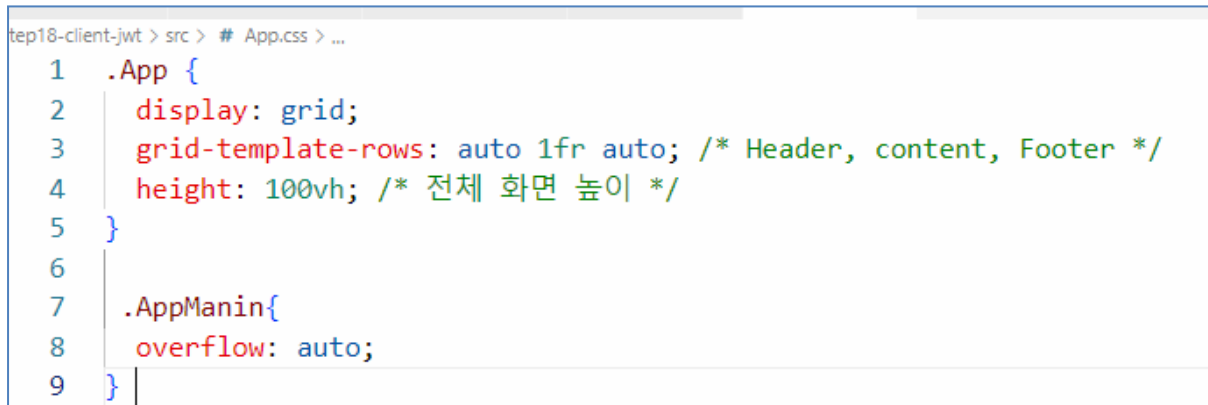
전체 페이지에 공통으로 적용 할 CSS

src > index.css | App.css

index.css파일



App.css 파일



src > components > Header.jsx | Header.css | Footer.jsx | Footer.css

Header.jsx파일

```

step18-client-jwt > src > components > Header.jsx > ...
1  import { Link } from 'react-router-dom';
2  import './Header.css';
3
4  const Header = () => {
5    return (
6      <>
7        <div className="Header">
8          <Link to="/" >HOME</Link>
9          <Link to="/joinForm">회원가입</Link>
10         <Link to="/loginForm">로그인</Link>
11         <Link to="/saveForm">글쓰기</Link>
12       </div>
13     </>
14   )
15 }
16
17 export default Header;

```

Header.css파일

```

step18-client-jwt > src > components > # Header.css > .Header
1  .Header{
2    display: flex;
3    gap: 20px;
4    justify-content: flex-start;
5    padding: 30px;
6    background-color: #43678a;
7    font-size: 25px;
8    color: white;
9    border-radius: 5px;
10 }
11
12 a{
13   text-decoration: none;
14   color: white;
15 }
16
17 a:hover{
18   text-decoration: underline;
19   color: #f08080;
20 }
21

```

실행화면



HOME 회원가입 로그인 글쓰기

Footer.jsx파일

```

step18-client-jwt > src > components > Footer.jsx > ...
1  import './Footer.css';
2  import emotion1 from '../assets/emotion1.png';
3  const Footer = ()=>{
4      return (
5          <>
6              <div className="Footer">
7                  <img src={emotion1} />
8                  <div className="Footer_div">Programmer Heejung Jang!</div>
9              </div>
10         </>
11     )
12 }
13
14 export default Footer;

```

Footer.css파일

```

step18-client-jwt > src > components > # Footer.css > ...
1  .Footer{
2      display: flex;
3      justify-content: center;
4      font-size: 30px;
5      gap: 10px;
6      padding: 30px;
7      width: auto;
8      background-color: rgba(0, 255, 204, 0.59);
9      margin-bottom: 20px;
10     border-radius: 5px;
11 }
12
13 .Footer> img{
14     width: 30px;
15 }
16
17 .Footer > .Footer_div{
18     text-shadow: 2px 2px 8px #FF0000;
19     color: white;
20 }

```

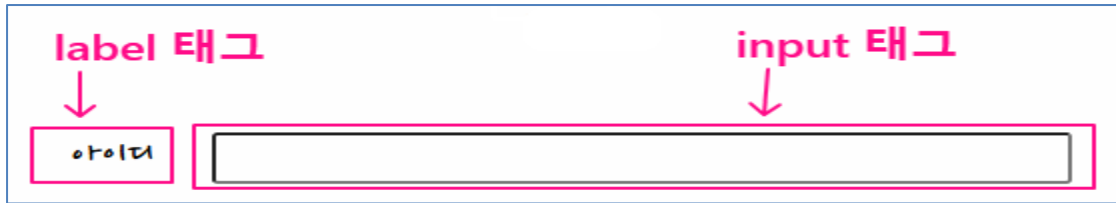
실행화면



LabelText.jsx파일 이해하기

현재 프로젝트에서 아래의 화면이 여러 컴포넌트에서 사용되고 있다. 때문에 우리는 아래 부분만

따로 제작하여(LabelText.jsx) 필요할 때 재사용 해서 사용하도록 하자..



LabelText.jsx파일

```

1 import { useState } from "react";
2 import "../LabelText.css"
3
4 const LabelText = ({text, name, value:initialValue})=>{
5   const [value, setValue] = useState(initialValue || "");
6   return (
7     <div className="LabelText" >
8       <label className="label">{text}</label>
9       <input className={` ${value==="" ? 'input' : 'input_disabled'} `}
10         type={` ${name==="pwd" ? "password" : "text"} `} name={name} value={value}
11         onChange={(e) => setValue(e.target.value)} />
12     </div>
13   )
14 }
15
16 export default LabelText;

```

text : label에 들어갈 글씨
 name : input요소가 password인지 text인지 전달
 value : 초기값
 value의 값이 없으면 className=(input) 있으면 className=(input_disabled) 설정
 type="password" or type="text"

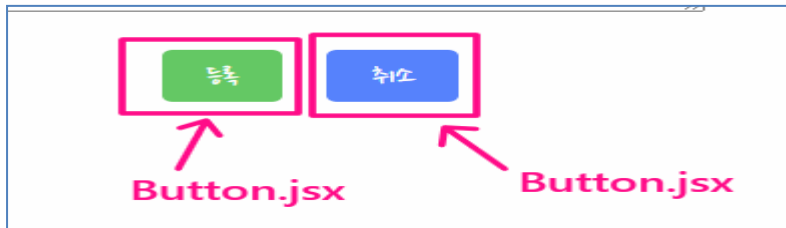
LabelText.css

```

1 .LabelText{
2   display: flex;
3   gap: 10px;
4   font-size:25px;
5   padding: 15px 50px;
6 }
7
8 .LabelText .label{
9   min-width: 60px;
10 }
11
12
13 .LabelText .input{
14   flex: 1;
15   border-radius: 3px;
16   font-size:1em;
17 }
18
19 .LabelText .input_disabled{
20   flex: 1;
21   border:none;
22   font-size:1em;
23 }
24

```

Button.jsx파일 이해 하기



전체 프로젝트에서 button이 여러 곳에서 쓰여지고 있고 text와 css와 다르기 때문에 컴포넌트로 만들어 놓고 재 사용하자.

```
step18-client-jwt > src > components > Button.jsx > [default]
1  import './Button.css';
2
3  const Button = ({text, type, onClick})=>{
4    return (
5      <button type={type} className={`Button Button_${type}`}
6        onClick={onClick}>{text}</button>
7    )
8  }
9
10 export default Button;
11
```

Button.css파일

```
step18-client-jwt > src > components > # Button.css > ...
1  .Button{
2    background-color: #rgb(236, 236, 236);
3    cursor: pointer;
4    border: none;
5    border-radius: 5px;
6    padding: 10px 20px;
7    font-size: 18px;
8    white-space: nowrap; /*이버튼안에 있는 text가 화면이 줄어들어도 줄바꿈되지않도록 한다.*/
9  }
10
11 .Button_button{
12   background-color: #rgb(100, 201, 100);
13   color: #white;
14 }
15
16 .Button_reset{
17   background-color: #rgb(86, 131, 253);
18   color: #white;
19 }
20
```

1)회원 가입

user > JoinForm.jsx | JoinForm.css

```
518-client-jwt > src > pages > user > JoinForm.jsx > JoinForm
1  import Button from "../../components/Button";
2
3  import LabelText from "../../components/LabelText";
4  import "../JoinForm.css";
5  const JoinForm = ()=>{
6      return (
7          <>
8              <h1 className="h1">회원가입</h1>
9
10             <div className="idDuplicate">
11                 <LabelText text={"아이디"} name={"id"}/>
12                 <div className="idText">아이디중복여부</div>
13             </div>
14
15             <LabelText text={"이름"} name={"name"}/>
16             <LabelText text={"비밀번호"} name={"pwd"}/>
17             <LabelText text={"주소"} name={"address"}/>
18
19             <div className="divBtn">
20                 <Button text={"회원가입"} type={"button"} />
21                 <Button text={"취소"} type={"reset"} />
22             </div>
23         </>
24     )
25 }
26
27 export default JoinForm;
```

JoinForm.css 파일

```

step18-client-jwt > src > pages > user > # JoinForm.css > ...
1
2 .idDuplicate{
3     display: flex;
4     justify-content: flex-start;
5 }
6
7
8 .idDuplicate .idText{
9     flex: 1;
10    padding-top: 20px ;
11 }
12 .divBtn{
13     display: flex;
14     gap: 20px;
15     justify-content: center;
16 }
17
18 .h1{
19     text-align: center;
20     font-size: 30px;
21     padding: 20px;
22 }
23
24 .idText{
25     color: red;
26     font-weight: bold;
27     font-size: 1.3em;
28 }

```

실행화면

회원가입

아이디

아이디 중복 여부

이름

비밀번호

주소

회원가입
취소

2) 로그인

user > LoginForm.jsx | LoginForm.css


```

step18-client-jwt > src > pages > user > LoginForm.jsx > LoginForm
1  import Button from "../../components/Button";
2  import LabelText from "../../components/LabelText";
3
4  import "./LoginForm.css";
5  const LoginForm = () => {
6      return (
7          <div className="LoginForm">
8              <h1 className="h1">로그인</h1>
9              <LabelText text={"아이디"} name={"id"} />
10             <LabelText text={"비밀번호"} name={"pwd"} />
11
12             <div className="divBtn">
13                 <Button text={"로그인"} type={"button"} />
14                 <Button text={"취소"} type={"reset"} />
15             </div>
16
17             </div>
18         )
19     }
20
21     export default LoginForm;

```

LoginForm.css

```

step18-client-jwt > src > pages > user > # LoginForm.css > LoginForm
1  .LoginForm{
2
3  }
4
5  .LoginForm .h1{
6      text-align: center;
7      font-size: 30px;
8      padding: 20px;
9  }

```

실행화면

로그인

아이디

비밀번호

1) 게시물 등록

board > SaveForm.jsx | SaveForm.css

SaveForm.jsx파일

```
p18-client-jwt > src > pages > board > SaveForm.jsx > SaveForm
1  import Button from "../../components/Button";
2  import LabelText from "../../components/LabelText";
3  import "../SaveForm.css";
4  const SaveForm = () => {
5    return (
6      <>
7        <h1 className="h1"> 게시물 등록하기 </h1>
8
9        <LabelText text={"title"} name={"title"} />
10       <LabelText text={"writer"} name={"writer"} />
11
12       <div className="divContent">
13         <div className="divContent_label">content</div>
14         <textarea className="divContent_textarea" placeholder="오늘은 어땠나요?" name="content" />
15       </div>
16
17       <div className="divBtn">
18         <Button text={"등록"} type={"button"} />
19         <Button text={"취소"} type={"reset"} />
20       </div>
21     </>
22   );
23 }
24
25 export default SaveForm;
```

SaveForm.css파일

```
1
2 .idDuplicate{
3     display: flex;
4     justify-content: flex-start;
5 }
6
7 .idDuplicate .idText{
8     flex: 1;
9     padding-top: 20px ;
10 }
11 .divBtn{
12     display: flex;
13     gap: 20px;
14     justify-content: center;
15 }
16
17 .h1{
18     text-align: center;
19     font-size: 30px;
20     padding: 20px;
21 }
22
23
24 .divContent{
25     display: flex;
26     gap: 10px;
27     margin-bottom: 30px;
28 }
29
30 .divContent .divContent_label{
31     font-size: 25px ;
32     margin-left: 50px;
33 }
34
35 .divContent .divContent_textarea{
36     padding: 20px;
37     width: 60%;
38     min-height: 200px;
39     resize: vertical;
40     box-sizing: border-box;
41     font-size: 25px ;
42 }
```

실행화면

게시물 등록하기

title

writer

content

오늘은 어땠나요?

등록

취소

2) 전체게시물

components > BoardItem.jsx | BoardItem.css

board > Home.jsx | Home.css

BoardItem.jsx

```

step18-client-jwt > src > components > BoardItem.jsx > BoardItem
1  import { Link } from "react-router-dom";
2  import "./BoardItem.css";
3  const BoardItem = ({title, writer}) => {
4    return (
5      <div className="BoardItem">
6        <div className="title_writer">제목 : {title} /작성자 : ({writer})</div>
7        <div className="detailLink">
8          <Link to={"/board/1"}>상세보기 </Link>
9        </div>
10     </div>
11   )
12 }
13
14 export default BoardItem;

```

BoardItem.css파일

```

step18-client-jwt > src > components > # BoardItem.css > .BoardItem .title_writer
1  .BoardItem{
2      border: 1px solid gainsboro;
3      border-radius: 5px;
4      padding: 10px;
5  }
6
7  .BoardItem .title_writer{
8      font-size: 25px;
9  }
10
11 .BoardItem .detailLink{
12     margin-top: 10px;
13     padding: 10px;
14     font-size: 20px;
15     border-radius: 15px;
16     background-color: #FFFF00;
17     width: 100px;
18     text-align: center;
19 }
20
21 .BoardItem .detailLink > a{
22     color: blue;
23 }

```

Home.jsx파일

```

step18-client-jwt > src > pages > board > Home.jsx > default
1  import BoardItem from "../../components/BoardItem";
2  import "./Home.css";
3
4  const Home =()=>{
5      return (
6          <div className="Home">
7              <h1>전체게시물 LIST</h1>
8              <BoardItem title={"Spring1"} writer={"장희정"}/>
9              <BoardItem title={"Spring1"} writer={"장희정"}/>
10             <BoardItem title={"Spring1"} writer={"장희정"}/>
11             <BoardItem title={"Spring1"} writer={"장희정"}/>
12         </div>
13     )
14 }
15
16 export default Home;

```

Home.css 파일

```
step18-client-jwt > src > pages > board > # Home.css > .Home h1
1  .Home{
2    | padding: 20px;
3  }
4
5  .Home h1{
6    | text-align: center;
7  }
8
```

실행화면



3) 상세보기

board > Detail.jsx | Detail.css

Details.jsx파일

```

1 import { useNavigate } from "react-router-dom";
2 import Button from "../../components/Button";
3 import LabelText from "../../components/LabelText";
4 import "../Detail.css";
5
6 const Detail = () => {
7   const nav = useNavigate();
8
9   return (
10     <div className="Detail">
11       <h1>상세보기</h1>
12
13       <LabelText text={"글번호"} value={"6"}/>
14       <LabelText text={"제목"} value={"React"}/>
15       <LabelText text={"작성자"} value={"이가현"}/>
16
17       <div className="divContent">
18         <div className="divContent_label">content</div>
19         <textarea className="divContent_textarea" placeholder="오늘은 어땠나요?" name="content" disabled value={"React너무 좋아요"}></textarea>
20       </div>
21
22       <div className="divBtn">
23         <Button text={"수정"} type={"button"} onClick={() => nav("/updateForm/3")} />
24         <Button text={"삭제"} type={"button"} />
25       </div>
26     </div>
27   )
28 }
29
30 export default Detail;

```

Details.css 파일

```

1 .Detail h1{
2   text-align: center;
3 }
4
5 .Detail > input{
6   border: none;
7 }

```

실행화면

상세보기

글번호	6
제목	React
작성자	이가현
content	<div style="border: 1px solid #ccc; padding: 10px; min-height: 80px;"> React너무 좋아요 </div>

수정
삭제

4) 수정하기

board > UpdateForm.jsx | UpdateForm.css

UpdateForm.jsx파일

```
step18-client-jwt > src > pages > board > UpdateForm.jsx > UpdateForm
1 import Button from "../../components/Button";
2 import LabelText from "../../components/LabelText";
3
4 const UpdateForm = () => {
5   return (
6     <>
7       <h1 className="h1"> 수정하기 </h1>
8       <form>
9
10        <LabelText text={"title"} name={"title"} />
11
12        <LabelText text={"writer"} name={"writer"} />
13
14        <div className="divContent">
15          <div className="divContent_label">content</div>
16          <textarea className="divContent_textarea" placeholder="오늘은 어땠나요?" name="content" /></textarea>
17        </div>
18
19        <div className="divBtn">
20          <Button text={"수정"} type={"button"} />
21          <Button text={"취소"} type={"reset"} />
22        </div>
23      </form>
24    </>
25  );
26 }
27
28 export default UpdateForm;
```

UpdateForm.css파일


```

2  .idDuplicate{
3      display: flex;
4      justify-content: flex-start;
5  }
6
7  .idDuplicate .idText{
8      flex: 1;
9      padding-top: 20px ;
10 }
11 .divBtn{
12     display: flex;
13     gap: 20px;
14     justify-content: center;
15 }
16
17 .h1{
18     text-align: center;
19     font-size: 30px;
20     padding: 20px;
21 }
22
23 .divContent{
24     display: flex;
25     gap: 10px;
26     margin-bottom: 30px;
27 }
28
29 .divContent .divContent_label{
30     font-size: 25px ;
31     margin-left: 50px;
32 }
33
34 .divContent .divContent_textarea{
35     padding: 20px;
36     width: 60%;
37     min-height: 200px;
38     resize: vertical;
39     box-sizing: border-box;
40     font-size: 25px ;
41 }

```

실행화면

수정하기

title

writer

cotent

오늘은 어땠나요?

수정

취소