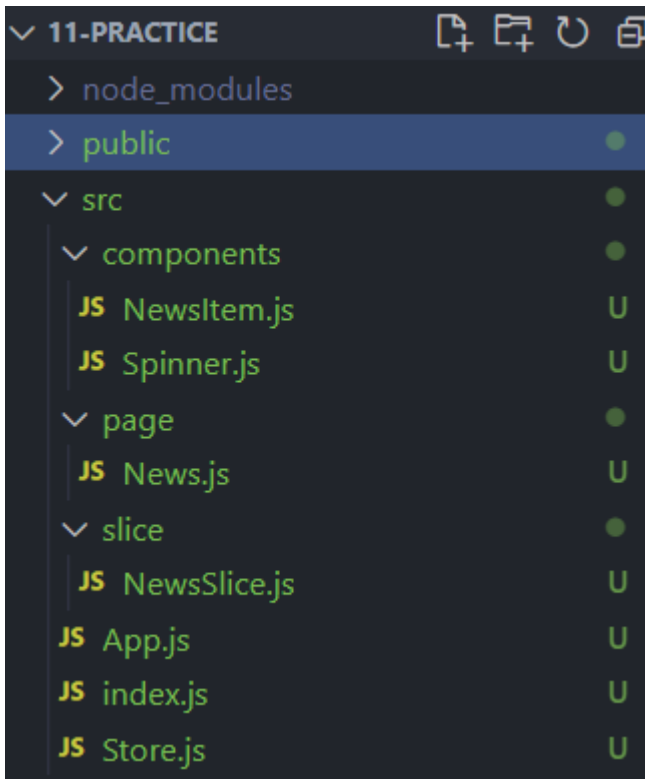







Redux 뉴스 목록 연습문제 - 박세영

DOM 구조



출력 결과

Redux 연습문제

	코스피, 미 증시 급락에 1%대 하락...시총 상위주 줄줄이 약세 코스피가 미국 증시 급락 영향으로 1% 넘게 하락하고 있다. 시가총액 상위 종목들도 대부분 약세다. 27일 코스피 지수는 오전 10시32분 현재 전 거래일보다 37.60포인트(1.41%) 내린 2630.71에 거래되고 있다. 이날 2630.58에 출발한 지수는 매일경제 Wed Apr 27 2022
	신한은행, 디지털자산 평가플랫폼 '쟁글'과 MOU 신한은행은 디지털자산 공시 및 평가 정보를 제공하는 '쟁글(Xangle)'의 운영사인 크로스앵글과 전략적 지분투자 및 사업협력 강화를 위한 업무협약을 체결했다고 26일 밝혔다. 이날 협약식은 신한은행 전월한 디지털전략그룹장과 신한캐피탈 정상훈 S금융본부장, 크로스앵글 이현우 공동대표, 크로스앵글 김준우 공동대표가 참석한 가운데 크로스앵글 본사에서 진행됐다. 크로스앵글은 국내 유일의 가상자산 평가 및 공시 플랫폼 '쟁글(Xangle)'의 운영사다. 크로스앵글은 빗... 이코노믹리뷰 Wed Apr 27 2022
	安, '인수위 1기 신도시 재건축 말 바꾸기' 논란에 "차질 없이 추진" 安, 인수위 1기 신도시 재건축 말 바꾸기 논란에 차질 없이 추진 조선비즈 Wed Apr 27 2022
	삼성전자, 휴대용 SSD 'T7 실드' 출시 삼성전자가 27일 휴대용 솔리드스테이트드라이브(SSD) 'T7 실드(Shield)'를 출시했다. 2020년 국제전자제품박람회(CES)에서 혁신상을 받은 8a... ZD넷 코리아 Wed Apr 27 2022
	'세계경제 둔화 공포' 국내증시도 급락...환율은 급등 [앵커]우크라이나 사태와 중국의 봉쇄 조치 등으로 세계 경제 성장이 둔화될 것이라는 공포감에 미국 증시가 급락한 데 이어 오늘 우리 주식시장도 크게 하락한 채 시작됐습니다. 원/달러 환율은 심리적 지지선인 1,250원을 뚫고 급상승하고 있습니다. 취재 기사를 연결해 알아보겠습니다. 박... YTN Wed Apr 27 2022

Windows 적용 인증

components

NewsItem.js 05-stylesheet단원의 '/components/NewsItem.js'를 재사용함

Spinner.js 실습시 사용한 파일 재사용함

index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import App from './App';
import {BrowserRouter} from 'react-router-dom';

//리덕스 구성을 위한 참조
import { Provider } from 'react-redux';
import store from './Store';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <BrowserRouter>
    <Provider store={store}>
      <App />
    </Provider>
  </BrowserRouter>
);
```

App.js

```
import React from 'react';
import News from './page/News';

function App() {
  return (
    <div>
      <h1> Redux 연습문제 </h1>
      <News/>
    </div>
  );
}

export default App;
```

Store.js

```
import { configureStore } from '@reduxjs/toolkit';

import NewsSlice from './slice/NewsSlice';

const store= configureStore({
  reducer: {
    news: NewsSlice
  },
  devtools: true
});

export default store;
```

slice

NewsSlice.js

```
import {createSlice, createAsyncThunk} from '@reduxjs/toolkit';
import axios from 'axios';

//비동기 처리 함수 구현- ajax요청을 위한 함수
export const getList= createAsyncThunk('news/getList', async(payload,
{rejectWithValue})=>{
  let result= null;
  try{
    result = await axios.get('http://localhost:3001/news');
  }catch(e){
    result= rejectWithValue(e.response)
  }
  return result; //ajax 연동 결과를 리턴
});

//slice 정의 (action함수+ reducer의 개념)
//ajax의 처리결과를 관리할 상태값 정의
const newsSlice= createSlice({
  name: 'news',
  initialState:{
    data: null, //ajax처리를 통해 수신된 데이터
    loading: false, //로딩 여부
    error: null //에러 정보
  },
  extraReducers:{ // ajax연동기능을 호출할때 사용
    [getList.pending]: (state, {payload})=>{ //로딩중-기존의 상태값 복사
      //로딩만 true로 바꿈
      return {...state, loading: true}
    },
    [getList.fulfilled]: (state, {payload})=>{ //ajax완료-성공
      return{
        data: payload?.data, //payload가 ajax결과가 존재할때만 data에 집
```

```

    근함
        loading: false,
        error: null
    },
    [getList.rejected]: (state, {payload})=>{ //에러
        return{
            data: payload?.data,
            loading: false,
            error:{
                code: payload?.status ? payload.status : 500,
                message: payload?.statusText ? payload.statusText : 'Server
Error'
            }
        }
    }
}
});

//리듀서 객체 내보내기
export default newsSlice.reducer;

```

page

News.js

```

import React from "react";
import styled from "styled-components";

import Spinner from "../components/Spinner";
import NewsItem from "../components/NewsItem";

//상태값을 로드하기위한 hook과 action함수를 dispatch할 hook 참조
import { useSelector, useDispatch } from "react-redux";
//slice에 정의된 액션 함수들 참조
import { getList } from "../slice/NewsSlice";

//05-styledsheet단원의 './components/NewsList.js'를 붙여넣기함
const ListContainer = styled.ul`
    list-style: none;
    padding: 0;
    margin: 0;
    width: 100%;
    box-sizing: border-box;
    display: flex;
    flex-direction: column;
    margin-bottom: 30px;
`;

const News = () => {

```

```
//컴포넌트가 마운트될때 콘솔의 모든 내용 삭제- 출력결과가 복잡해지는것을 방지
React.useEffect(() => console.clear(), []);

//hook을 통해 slice가 관리하는 상태값 가져오기
const { data, loading, error } = useSelector((state) => state.news);

const dispatch = useDispatch();

//컴포넌트가 마운트되면 데이터 조회를 위한 액션함수를 다스패치함
React.useEffect(() => {
  dispatch(getList());
}, [dispatch]);
return (
  <div>
    <Spinner visible={loading} />
    {error ? (
      <div>
        <h1> ERROR!! {error.code}</h1>
        <hr />
        <p>{error.message}</p>
      </div>
    ) : (
      <ListContainer>
        {data && data.map((v, i) => <NewsItem key={i} item={v} />)}
      </ListContainer>
    )}
  </div>
);
};

export default News;
```