

[개인 과제] 프론트엔드 기초

훈련 과 정 명	[신한투자증권] 프로디지털아카데미 7기
강 의 과 목 명	프론트엔드 개발 기초 (JavaScript)
강 사 명	신윤수
과 제 명	<p>실전 자바스크립트 종합 과제</p> <ul style="list-style-type: none"> - 웹 분석하여 데이터 수집 및 저장하기 (HTML) - 웹 분석하여 데이터 수집 및 저장하기 고급 (JSON) - Bootstrap 이용하여 UI 구성하기. - React에서 통신 후 UI에 Rendering하기 - React에서 복잡한 Component 구성하기
제 출 기 한	2026년 3월 8일 일요일 오후 23:59까지 제출

[과제 제출시 주의사항]

- 첨부파일을 다운받고 작업하십시오.
- 다운받고 homework 디렉토리 내에 각 디렉토리 (homework-react, homework-node)로 이동후 **npm install** 명령어를 입력하시면 관련 패키지가 설치 됩니다.
- 패키지 설치의 자유롭게 하셔도 되지만, **제출시에는 node_modules 디렉토리는 모두 제거**하여 주십시오.
- 과제 제출시 **homework 디렉토리(가장 상위 디렉토리)의 이름을 제출자 이름으로 변경 후 압축하여 제출**바랍니다. (예시. <신윤수/homework-react, 신윤수/homework-node>)

자바스크립트 종합 과제로서 총 **두개의 파트**로 나뉘어집니다.

1. Node.js 환경에서의 데이터 수집
2. React를 활용한 UI 실습 및 hook과 fetch 함수를 이용한 통신

배운 것을 모두 활용하시고, 배우지 않은 것을 찾아서 활용하셔도 좋습니다.

[총 세부 과제 및 과제별 배점]

총 과제의 개수는 4개입니다.

- 세부 과제의 총 개수는 7개이며, 각 세부과제별 내용과 배점은 다음과 같습니다.

- 1.1: 다음 검색(뉴스탭)에서 뉴스 데이터 수집하기 (15점)
- 1.2: 다음 주가차트에서 데이터 수집하기 (15점)
- 2.1. jsonPlaceholder에서 user 데이터 수집하여 UI 완성하기 (30점)
 - 2.1.1. UserApp UI 완성하기 (20점)
 - 2.1.2. UserApp 데이터 통신 및 Rendering (10점)
- 2.2. 수집된 데이터(stock.json, company.json) 활용하여 UI 완성하기 (40점)
 - 2.2.1. StockApp 데이터 통신하여 Rendering 하기 (15점)
 - 2.2.2. StockApp 부트스트랩 활용 Table UI Rendering 하기 (15점)
 - 2.2.3. StockApp State 활용 및 동적인 UI 응용 (10점)

1. Node.js 환경에서의 데이터 수집

1번 과제는 Node.js 환경에서 작업합니다.

1.1 다음 검색(뉴스탭)에서 뉴스 데이터 수집하기 (15점)

검색 URL: <https://search.daum.net/search?w=news&q=sk하이닉스>

D sk하이닉스



통합

뉴스

통합웹

쇼핑

지도

멜론뮤직

동영상

책



관련 sk하이닉스 주가 삼성전자 한미반도체 sk하이닉스 배당금 sk 스퀘어 삼성전기
 sk하이닉스 주가 전망 sk 이노베이션 엔비디아 sk텔레콤 sk하이닉스 성과급



• 정확도순 • 최신순 • 오래된순

옵션

전체 검색결과

뉴스검색 설정

외국인들 SK하이닉스 '찜'하더니...뭉치돈 1.2조 몰렸다 [종목+]

외인의 귀환...엔비디아 업은 SK하이닉스, 반등 시작되나외인 SK하이닉스 이달 국장서 ...

사진=게티이미지뱅크 외국인 투자자가 국내 증시에서 10개월 만에 순매수로 돌아선 가운데 5월 SK하이닉스를 1조5000억원 가까이 사들인 것으로 나타...

한국경제 | 2025.05.31 | 다음뉴스



퓨어스토리지, SK하이닉스와 협력

퓨어스토리지가 반도체 대기업 SK하이닉스와 협력하여 차세대 QLC 플래시 스토리지 제품을 공동 개발한다고 발표했다.이 혁신적인 제품은 대용량 데이터 처리와 에너지 효율성이 필수적인 하이퍼...

테크노아 | 2025.05.31

<그림1>

D sk하이닉스



통합

뉴스

통합웹

쇼핑

지도

멜론뮤직

동영상

책



관련 sk하이닉스 주가 삼성전자 한미반도체 sk하이닉스 배당금 sk 스퀘어 삼성전기
 sk하이닉스 주가 전망 sk 이노베이션 엔비디아 sk텔레콤 sk하이닉스 성과급



• 정확도순 • 최신순 • 오래된순



기간

전체

1일

1주

1개월

6개월

1년

2025.5.31.~2025.5.31. ^

2025.5.31.

~

2025.5.31.

년	월	일
2022	3	28
2023	4	29
2024	5	30
2025	6	31

적용

유형

전체

포토

동영상

인포그래픽

자동생성기사

보도자료

언론사

전체

가나다순 ▾

직접입력 ▾

옵션유지 ☐

초기화

<그림2: 옵션 클릭 후 날짜 설정>

외국인들 SK하이닉스 '찜'하더니...뭉치돈 1.2조 몰렸다 [종목+]

외인의 귀환...엔비디아 업은 SK하이닉스, 반등 시작되나외인 SK하이닉스 이달 국장서 ...

사진=게티이미지뱅크 외국인 투자자가 국내 증시에서 10개월 만에 순매수로 돌아선 가운데 5월 SK하이닉스를 1조5000억원 가까이 사들인 것으로 나타...

한국경제 2025.05.31 다음뉴스



퓨어스토리지, SK하이닉스와 협력

퓨어스토리지가 반도체 대기업 SK하이닉스와 협력하여 차세대 QLC 플래시 스토리지 제품을 공동 개발한다고 발표했다.이 혁신적인 제품은 대용량 데이터 처리와 에너지 효율성이 필수적인 하이퍼...

테크노아 | 2025.05.31

<그림3: 해당하는 날짜에 속하는 뉴스리스트>

[과제 내용]

1. <그림1: 검색 URL: <https://search.daum.net/search?w=news&q=sk하이닉스>> 에 접속하면 나오는 페이지에서 옵션 클릭
2. <그림2>에서 볼 수 있듯이 날짜 설정을 할 수 있다.

다음 검색에서 날짜 설정을 하여 총 5종류의 날짜(아래 [목표 날짜] 참조)에 해당하는 "sk하이닉스"검색 결과 중에 제목과 뉴스 URL을 수집하여 Json 형태로 저장하는 코드를 작성하고 파일로 저장하시오.

[파일이름:news.json]

(하단 [결과예시] 확인)

- 저장데이터의 key값은 날짜범위의 종료날짜[%YYYY%MM%DD ("20250528" 등)]로 하고 value는 뉴스내용({title: "<기사제목>", "url": <기사 url>})의 배열로 한다.
- 검색결과는 1페이지만 조회하도록 한다. (뉴스의 개수 10개)

[파일저장경로]:

- <제출자이름>/homework-node/data/news.json 으로 저장

[저장할 Json Format 형태:]

```
{
  [date]: [{title: "title1...", url: "https://~~url1..."}]
}
```

예)

```
{
  "20250528": [{title: "뉴스1 제목", url: "url1링크"}, ...],
  "20250529": [{title: "뉴스1 제목", url: "url1링크"}, ...],
  "20250530": [{title: "뉴스1 제목", url: "url1링크"}, ...],
```

```
"20250531": [{title: "뉴스1 제목", url: "url1 링크"}, ...],
"20250601": [{title: "뉴스1 제목", url: "url1 링크"}, ...],
}
```

[날짜범위]

1. **20250528** 00시 00분 00초 ~ **20250528** 23시59분59초
2. **20250529** 00시 00분 00초 ~ **20250529** 23시59분59초
3. **20250530** 00시 00분 00초 ~ **20250530** 23시59분59초
4. **20250531** 00시 00분 00초 ~ **20250531** 23시59분59초
5. **20250601** 00시 00분 00초 ~ **20250601** 23시59분59초

[결과 예시 - news.json]

```
1 {
2   "20250528": [
3     {
4       "title": "SK하이닉스, 엔비디아와 HBM4 협상 마무리 단계... 'HBM 1위' 자리 굳힌다",
5       "url": "http://v.daum.net/v/20250528060103484"
6     },
7   > { ...
10  },
11  > { ...
14  },
15  > { ...
18  },
19  > { ...
22  },
23  > { ...
26  },
27  > { ...
30  },
31  > { ...
34  },
35  > { ...
38  },
39  > { ...
42  }
43 ],
44 "20250529": [
45   {
46     "title": "엔비디아 또 어닝 서프라이즈...SK하이닉스 HBM 수혜 지속",
47     "url": "http://v.daum.net/v/20250529163518835"
48   },
49   {
50     "title": "젠슨 황 \"AI 인프라 수요 강해\"...SK하이닉스 기대감도 'UP'",
51     "url": "http://v.daum.net/v/20250529150256962"
52   },
53 ]
54 }
```

1.2 다음 주가차트에서 데이터 수집하기 (15점)

sk하이닉스 코드: 000660

daum.net 주가차트URL: <https://finance.daum.net/quotes/A000660#chart>

분석차트 ☒ 수정주가 적용

빅차트 >

SK하이닉스

257,000

25.06.20

▲ 11,000 +4.47%

시가: 249,000

고가: 257,000

저가: 247,500

거래량: 3,548,259

24.08.19 시: 199,900 고: 199,900 저: 192,400 종: 193,900 ▼ -5,800 -2.90% 거래량: 3,925,283

종목명 or 코드

추가

일봉

주봉

월봉

분봉 ▾

차트종류 ▾

지표설정 ▾



<그림1>

[과제 내용]

- daum.net 주가차트URL에 접속하고 스크롤을 조금 내리면 <그림1> 과 같은 페이지를 확인할 수 있다.
- 현재 <그림1>은 수정주가 적용이 체크되어 있는 상태이고, sk하이닉스에 대한 일봉에 대한 내역을 볼 수 있는 상태이다.
- 2025-06-06 00:00:00까지의 60일치의 일봉(2025-03-10~2025-06-05 까지의 데이터가 들어있음.) [총 60개의 데이터] 를 수집하여 Array<StockData>형태의 json포맷으로 저장하는 코드를 작성하고

실제로 저장하여라.

- (60개 이상의 데이터를 가져와도 됩니다. 2025-03-10~2025-06-05. 데이터만 포함시켜주세요.)

[파일이름: data/stock.json] (하단 결과예시 확인)

1. StockData에는 반드시 [date, tradePrice, openingPrice, highPrice, lowPrice, candleAccTradePrice] key는 들어야 한다.

2. (1)내용에서 각 key에 따른 의미는 다음과 같습니다..

(date: 종가 날짜, tradePrice: 종가, openingPrice: 시가, highPrice: 고가, lowPrice: 저가, candleAccTradePrice: 거래대금)

[결과예시 - stock.json]

```

1  [
2    {
3      "symbolCode": "KR7000660001",
4      "date": "2025-03-10",
5      "candleTime": "2025-03-10 00:00:00",
6      "tradePrice": 187900,
7      "openingPrice": 187500,
8      "highPrice": 191500,
9      "lowPrice": 187300,
10     "change": "FALL",
11     "changePrice": -4500,
12     "changeRate": -0.0233887734,
13     "candleAccTradeVolume": 2408989,
14     "candleAccTradePrice": 454523032900,
15     "timestamp": 1741532400000
16   },
17   { ...
31   },
32   { ...
46   },
47   {
48     "symbolCode": "KR7000660001",
49     "date": "2025-03-13",
50     "candleTime": "2025-03-13 00:00:00",
51     "tradePrice": 199700,
52     "openingPrice": 202500,
53     "highPrice": 204500,
54     "lowPrice": 195500,
55     "change": "RISE",
56     "changePrice": 800,
57     "changeRate": 0.0040221217,
58     "candleAccTradeVolume": 3860466,
59     "candleAccTradePrice": 772070452100,
60     "timestamp": 1741791600000
61   },

```

위 빨간 박스에 해당하는 key는 필수, 그 외는 자유입니다.(넣으셔도 되고 빼도 됨)

2. React 응용

본 과제는 React를 자유롭게 응용 하시면 됩니다. 컴포넌트 분리는 하셔도 되고 하지 않으셔도 됩니다. 그에 따른 점수에 대한 영향은 없습니다. 요구사항만 충족시켜주시면 됩니다.

작업디렉토리: **homework-react**

현재 프로젝트는 react-bootstrap이 설치되어있는 환경이다.

- src/components/Root.jsx 에는 UserApp과 StockApp이 각각 Rendering되고 있다.
- public/data 디렉토리에는 각각 **stock.json**과 **company.json**이 있다.

다음 조건들에 맞게 UserApp과 StockApp을 완성하시오.

** 사전지식 (react개발 서버를 실행한 후 <npm run dev 후>)

- 해당 url(예:localhost:5173)에 public에 위치한 경로를 입력하면 public에 위치한 static file을 반환
⇒ localhost:5173/data/company.json에 HTTP Request시 company.json 내용을 반환
⇒ localhost:5173/data/stock.json에 HTTP Request시 stock.json 내용을 반환

2.2.1. UserApp UI완성하기 (20점)

작업 Component: src/components/UserApp.jsx

- react-bootstrap List Group Component:
<https://react-bootstrap.netlify.app/docs/components/list-group#actionable-items>

[과제 내용]

1. react-bootstrap ListGroup 컴포넌트를 사용하여 다음 [결과 예시]에 해당하는 UI를 완성하시오.

- 해당하는 데이터는 임의로 하셔도 무방합니다.
(ListGroup과 ListGroup.Item 만 사용하시면 됩니다. 세부 스타일링은 과제점수에 반영 되지 않습니다.)

[결과 예시]

Users

0
1
2
3
4
5
6
7
8
9
10

ETF: [SOL 200Top10](#)

2.1.2. UserApp 데이터 통신 및 Rendering (10점)

작업 Component: src/components/UserApp.jsx

- 통신할 user URL: <https://jsonplaceholder.typicode.com/users>

[과제 내용]

- 위 링크(<https://jsonplaceholder.typicode.com/users>)에 요청을 보내면, 총 10명의 user에 대한 data를 불러온다.
위 링크에 HTTP요청을 하고 state에 저장한 후,
각각 <ListGroup.Item> 컴포넌트에서 다음형태로 Rendering하시오 { **user.name** }
⇒ 데이터에서 **[user.name]** 속성이 필요
⇒ **[결과 예시]** 참조
- 컴포넌트가 마운트 되었을 때 단 한 번만 API가 호출되도록 작성하여 주세요.

[결과 예시]

Users
Leanne Graham
Ervin Howell
Clementine Bauch
Patricia Lebsack
Chelsey Dietrich
Mrs. Dennis Schulist
Kurtis Weissnat
Nicholas Runolfsdottir V
Glenna Reichert
Clementina DuBuque

ETF: [SQL 200Top10](#)

2.2.1. StockApp 데이터 통신하여 Rendering하기 (15점)

작업 Component: src/components/StockApp.jsx

[과제 내용]

- **/data/comapny.json** 에 요청을 보내어 회사이름과 종목코드를 받아오고 해당 내용을 Rendering하시오. (렌더링 스타일 형식은 자유이며 채점에 들어가지 않습니다. <단 “회사이름”과 “종목코드”는 반드시 구분하여 렌더링 하세요.> 예시에서는 Bootstrap List 사용.)
- 컴포넌트가 마운트 되었을 때 단 한 번만 API가 호출되도록 작성하여 주세요.
- 결과는 아래 **[결과 예시]** 확인

[결과 예시]

Users

ETF: [SOL 200Top10](#)

SK하이닉스	000660
NAVER	035420
HD현대중공업	329180
현대차	005380
KB금융	105560
기아	000270
LG에너지솔루션	373220
삼성전자	005930
삼성바이오로직스	207940
셀트리온	068270

2.2.2. StockApp 부트스트랩 활용 Table UI Rendering 하기 (15점)

작업 Component: src/components/StockApp.jsx

- **react-bootstrap table Component:**
<https://react-bootstrap.netlify.app/docs/components/table>

[과제 내용]

- react-bootstrap table을 사용하여 다음 <그림1>에 해당하는 UI를 완성하시오. 부트스트랩 Table Component에 다음 속성에 해당하는 props만 true로 설정하시고 나머지 스타일링은 자유입니다.

-

<true로 설정할 props>

- striped
- bordered
- hover

[결과 예시]

Users

ETF: [SOL 200Top10](#)

SK하이닉스	000660
NAVER	035420
HD현대중공업	329180
현대차	005380
KB금융	105560
기아	000270
LG에너지솔루션	373220
삼성전자	005930
삼성바이오로직스	207940
셀트리온	068270

주식:

날짜	종가	시가	고가	저가	거래대금
----	----	----	----	----	------

2.2.3. StockApp State를 활용 및 동적인 UI 응용 (10점)

[파일 설명]

- /data/stock.json: 종목코드를 key로 하고, 일일 주식 시세 데이터 배열을 value로 하는 객체

[과제 내용]

- /data/stock.json 에 요청을 보내어 종목코드별 일일 주식 시세 데이터를 받아와서 state에 setState 하세요.
- 2.1.1에서 작성한 리스트에서 종목을 클릭할 경우 우측에 시세 데이터 렌더링
 - 시세 데이터 렌더링시 현재 선택된 주식 이름을 표시하여 주세요. (결과 예시에서 우상단 "주식:NAVER")
 - 만약 선택된 주식이 없을 경우엔, 평가하지 않습니다.
 - 아래 테이블에 주식 시세 데이터를 모두 렌더링 하세요.
 - 데이터에서 "change"의 값이 "FALL" 이면 종가의 색상을 "blue"로, "RISE"이면 종가의 색상을 "red"로 해주세요.
 - "change"의 값이 "EVEN"인 경우 색상은 고려하지 않습니다. (어떤 색상도 가능)
- 결과는 아래 [결과 예시] 확인

[결과 예시]

Users	ETF: SOL 200Top10		주식: NAVER					
	SK하이닉스	000660	날짜	종가	시가	고가	저가	거래대금
	NAVER	035420	2025-03-10	215000	211500	217000	210500	115570159000
	HD현대중공업	329180	2025-03-11	212500	205500	214500	205000	123261000500
	현대차	005380	2025-03-12	214000	212500	215500	211000	86574114500
	KB금융	105560	2025-03-13	216000	217500	219500	213000	133437758000
	기아	000270	2025-03-14	208000	215500	216500	208000	107821859000
	LG에너지솔루션	373220	2025-03-17	211000	210000	211000	207500	100333785000
	삼성전자	005930	2025-03-18	209000	211000	214000	208000	139895420750
	삼성바이오로직스	207940	2025-03-19	208000	208000	211500	207500	144488306250
	셀트리온	068270	2025-03-20	208000	210000	210000	207500	122998460750
			2025-03-21	209500	209500	210500	206000	138783623000
			2025-03-24	207000	209500	210500	206000	73558846750
			2025-03-25	207500	208500	209000	206500	75818845750
			2025-03-26	204000	207500	208500	203000	148746915000
			2025-03-27	197800	201000	203000	196500	297296749500
			2025-03-28	194700	197600	197700	193900	135939077150
			2025-03-31	191000	191400	192100	189200	127432011050
			2025-04-01	196700	191600	198000	191600	94912626650
			2025-04-02	195600	197600	197800	194200	64564997000
			2025-04-03	196600	199000	199000	199000	113535510000

다시 한 번 강조

[과제 제출시 주의사항]

- 첨부파일을 다운받고 작업하십시오.
- 다운받고 homework디렉토리 내에 각 디렉토리 (homework-react, homework-node)로 이동후 npm install 명령어를 입력하시면 관련 패키지가 설치 됩니다.
- 패키지 설치의 자유롭게 하셔도 되지만, 제출시에는 node_modules 디렉토리는 모두 제거하여 주십시오.
- 과제 제출시 homework 디렉토리(가장 상위 디렉토리)의 이름을 제출자 이름으로 변경 후 압축하여

제출바랍니다. (예시. <신윤수/homework-react, 신윤수/homework-node>)