

리액트(React) 기초

리액트 라우터를 활용한 라우팅 구현

김경민



리액트 라우터

- 브라우저의 주소를 다루기 위한 라이브러리
 - 브라우저의 URL을 감지하고 이에 따라 적절한 컴포넌트를 렌더링
- 프로젝트 생성 후 라우터 설치
 - `npm install react-router-dom`

Package.json

```
1 {
2   "name": "myapp",
3   "private": true,
4   "version": "0.0.0",
5   "type": "module",
6   "scripts": {
7     "start": "vite",
8     "dev": "vite",
9     "build": "vite build",
10    "preview": "vite preview"
11  },
12  "dependencies": {
13    "@tailwindcss/vite": "^4.0.8",
14    "react": "^19.0.0",
15    "react-dom": "^19.0.0",
16    "react-icons": "^5.5.0",
17    "react-router-dom": "^7.3.0",
18    "tailwindcss": "^4.0.8"
19  },
20  "devDependencies": {
21    "vite": "^5.0.0"
22  }
23 }
```

라우터 적용

• BrowserRouter

- React Router의 기본 구성 요소이며, URL을 관리하는 데 사용
- 애플리케이션의 최상위 레벨에 위치하며, React Router를 초기화

• Routes

- 라우팅 정보를 정의하는 컴포넌트
- 하위에 Route 컴포넌트를 렌더링하며, URL에 따라 매칭되는 Route 컴포넌트를 렌더링

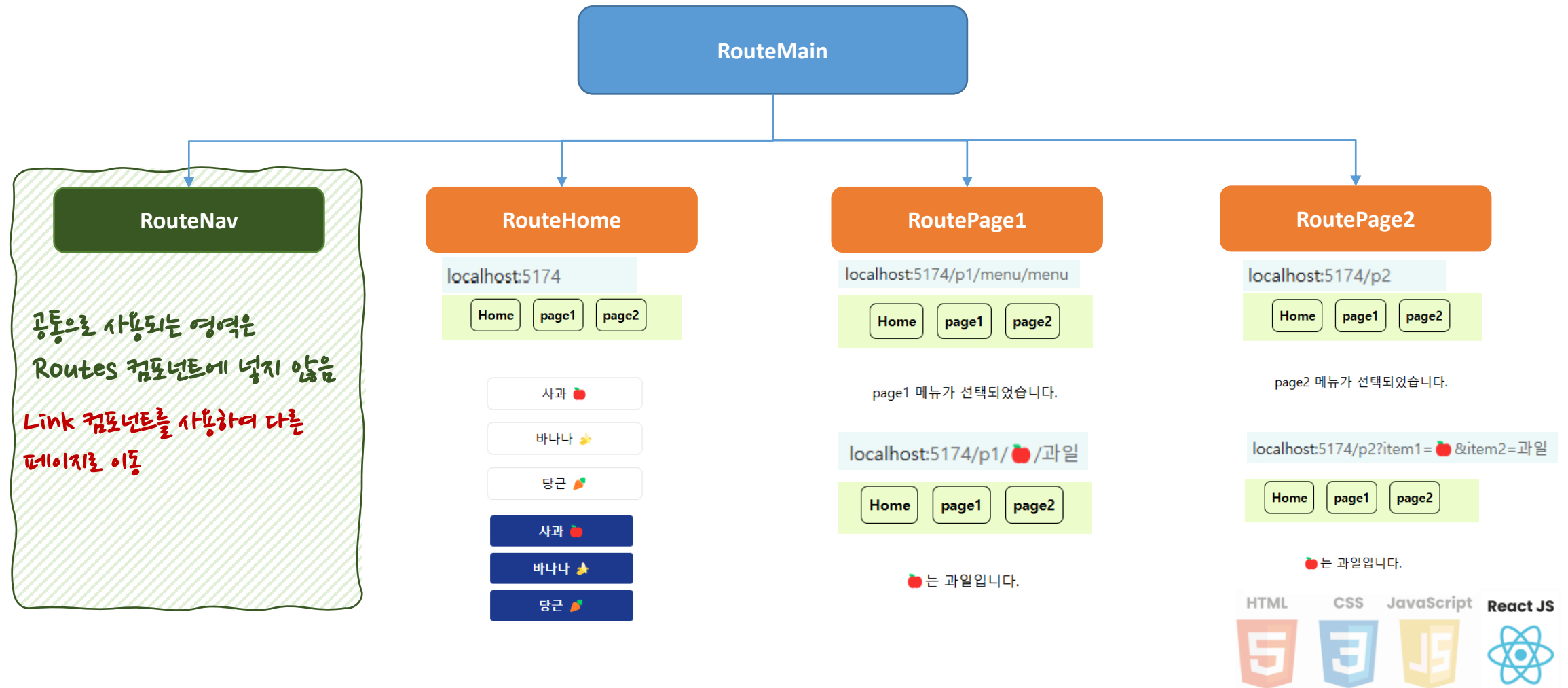
• Route

- URL과 컴포넌트를 매핑하는 컴포넌트
- path 속성과 component 속성으로, path에 맞는 URL이 요청되면 component 속성에 지정된 컴포넌트를 렌더링

```
export default function RouteMain() {  
  return (  
    <BrowserRouter>  
      <div className="w-full">  
        <RouteNav />  
        <Routes>  
          <Route path="/" element={<RouteHome />} />  
          <Route path="/p1/:item1/:item2" element={<RoutePage1 />} />  
          <Route path="/p2" element={<RoutePage2 />} />  
        </Routes>  
      </div>  
    </BrowserRouter>  
  )  
}
```



라우터 적용 예



URL 이동

• <Link>

- react-router-dom 라이브러리에서 제공하는 컴포넌트
- 페이지를 새로 고침하지 않고도 URL을 변경하여 다른 컴포넌트로 이동

```
import { Link, useNavigate } from "react-router-dom";
import TailButton from "../UI/TailButton";
export default function RouteHome() {
  const navigate = useNavigate();

  return (
    <div className="w-full flex flex-col justify-center items-center">
      <ul className="w-50 flex flex-col justify-center items-center">
        <li className="w-full">
          <Link to="/p1/🍎/과일" ...
            hover:bg-lime-800 hover:text-white"> 사과 🍎 </Link>
        </li>
        <li className="w-full">
          <Link to="/p1/🍌/과일" ...
            hover:bg-lime-800 hover:text-white"> 바나나 🍌 </Link>
        </li>
        <li className="w-full">
          <Link to="/p1/🥕/채소" ...
            hover:bg-lime-800 hover:text-white"> 당근 🥕 </Link>
        </li>
      </ul>
    </div>
  );
}
```

• useNavigate Hook

- react-router-dom 라이브러리에서 제공하는 React 훅
- 이벤트 핸들러나 조건에 따라 URL을 변경하고 싶을 때 사용

```
import { Link, useNavigate } from "react-router-dom";
import TailButton from "../UI/TailButton";
export default function RouteHome() {
  const navigate = useNavigate();

  return (
    <div className="w-full flex flex-col justify-center items-center">
      <ul className="w-50 flex flex-col justify-center items-center">
        <li className="w-50 grid-cols-1 gap-4 p">
          <TailButton caption = "사과 🍎"
            onClick = {() => navigate("/p2?item1=🍎&item2=과일")}
            color = "blue" />
          <TailButton caption = "바나나 🍌"
            onClick = {() => navigate("/p2?item1=🍌&item2=과일")}
            color = "blue" />
          <TailButton caption = "당근 🥕"
            onClick = {() => navigate("/p2?item1=🥕&item2=채소")}
            color = "blue" />
        </li>
      </ul>
    </div>
  );
}
```

클릭이벤트가 발생할 때 url 조작



URL 파라미터

• 주소의 경로에 값을 넣어 전송

- URL 파라미터로 사용할 파라미터를 :뒤에 작성
- 여러 개의 파라미터를 사용할 경우 /:i1/:i2 형식으로 사용

```
<Routes>
  <Route path="/" element={<RouteHome />} />
  <Route path="/p1/:item1/:item2" element={<RoutePage1 />} />
  <Route path="/p2" element={<RoutePage2 />} />
</Routes>
```

```
<li className="w-full">
  <Link to="/p1/🍎/과일"
        className="w-full h-full flex justify-center items-center
        border border-gray-200 p-2 m-2 rounded-lg
        hover:bg-lime-800 hover:text-white"> 사과 🍎 </Link>
</li>
```

localhost:5174/p1/🍎/과일

Home page1 page2

🍎는 과일입니다.

```
import { useParams } from
```

```
export default function RoutePage1() {
```

```
  const { item1, item2 } = useParams();
```

```
  return (
```

```
    <div>
```

```
      { item1 == "menu" ? "page1 메뉴가 선택되었습니다."
        : ` ${item1}는 ${item2}입니다.`
```

```
    }</div>
```

```
  )
```

```
}
```

현재 URL 경로에 포함된 동적
파라미터들을 객체 형태로
반환함으로 구조 분해 할당으로
필요한 파라미터들을 사용



URL 쿼리 스트링

• useLocation()

- 현재 URL에 대한 위치 정보를 반환
 - pathname : URL의 경로
 - search : 쿼리 문자열
 - hash : 해시
 - 상태 : state

• useSearchParams()

- React Router v6에서 제공하는 hook으로, URL의 쿼리 스트링을 쉽게 다루기 위한 기능
- useSearchParams hook은 두 가지 값을 반환
 - searchParams: 현재 URL의 쿼리 스트링을 읽을 수 있는 객체.
 - setSearchParams: 쿼리 스트링을 업데이트 하는 함수
- 쿼리 스트링 읽기
 - searchParams.get('key'): 특정 쿼리 파라미터 값

```
import { useLocation, useSearchParams } from "react-router-dom"
export default function RoutePage2() {
  const loc = useLocation() ;
  console.log("loc =", loc) ;
```

현재의 URL에 대한 정보를 가져오는 데 사용

```
const [sParams] = useSearchParams() ;
console.log("sParams =", sParams) ;
```

```
const queryList = [...sParams] ;
console.log("queryList =", queryList)
```

URLSearchParams 객체를 배열로 변환

```
return (
  <div>
    { queryList.length == 0 ? "page2 메뉴가 선택되었습니다."
      : `${queryList[0][1]}는 ${sParams.get("item2")}입니다.`
    }
  </div>
)
```



해결문제 - 데이터 수집

· 기상청_단기예보 ((구)_동네예보) 조회서비스

기본정보

데이터명	기상청
서비스유형	RES
신청유형	개발
활용기간	202

서비스정보

참고문서	기상청
데이터포맷	JSC
End Point	http

항목명	샘플데이터	설명
ServiceKey	8qw7g%2FC%2BM	공공데이터포털에서 받은 인증키
pageNo	1 샘플데이터	페이지번호
numOfRows	1000	한 페이지 결과 수
dataType	json	요청자료형식(XML/JSON) Default: XML
base_date	20210628	'21년 6월 28일발표
base_time	0500	05시 발표
nx	55	예보지점의 X 좌표값
ny	127	예보지점의 Y 좌표값

4

예보버전조회

단기예보정보조회서비스 각각의 오퍼레이션(초단기실황, 초단기예보, 단기예보)들의 수정된 예보버전을 파악하기 위해 예보버전을 조회하는 기능

	일일 트래픽	미리보기
발표일자, 발표시각, 예보지점의 조회 조건으로 발표일자, 발표시각, 예보지점 정보를 조회하는 기능	10000	확인
해 발표일자, 발표시점 Y 좌표의 조회 조건으로 발표일자, 발표시점 Y 좌표의 정보를 조회하는 기능	10000	확인
해 발표일자, 발표시점 X 좌표의 조회 조건으로 발표일자, 발표시점 X 좌표, 예보지점 Y 좌표의 정보를 조회하는 기능	10000	확인
단기예보정보조회서비스 각각의 오퍼레이션(초단기실황, 초단기예보, 단기예보)들의 수정된 예보버전을 파악하기 위해 예보버전을 조회하는 기능	10000	확인

해결문제 - 데이터 수집

• 기상청_단기예보 ((구)_동네예보) 조회서비스

2 초단기예보조회

초단기예보정보를 조회하기 위해 발표일자, 발표 시각, 예보지점 X 좌표, 예보지점 Y 좌표의 조회 조건으로 자료구분코드, 예보값, 발표일자, 발표시각, 예보지점 X 좌표, 예보지점 Y 좌표의 정보를 조회하는 기능

10000

확인

요청변수(Request Parameter)

항목명	샘플데이터	
ServiceKey	8qw7g%2FC%2BMGd2iRc	공공데이터포털에서 받은
pageNo	1	페이지번호
numOfRows	1000	한 페이지 결과 수
dataType	json	요청자료형식(XML/JSON)
base_date	20210628	'21년 6월 28일 발표
base_time	0630	06시30분 발표(30분 단위)
nx	55	예보지점 X 좌표값
ny	127	예보지점 Y 좌표값

```
▼"item": [  
  ▼{  
    "baseDate": "20241011",  
    "baseTime": "0630",  
    "category": "LGT",  
    "fcstDate": "20241011",  
    "fcstTime": "0700",  
    "fcstValue": "0",  
    "nx": 55,  
    "ny": 127  
  },  
]
```

3 단기예보조회

단기예보 정보를 조회하기 위해 발표일자, 발표시각, 예보지점 X좌표, 예보지점 Y 좌표의 조회 조건으로 발표일자, 발표시각, 자료구분문자, 예보 값, 예보일자, 예보시각, 예보지점 X 좌표, 예보지점 Y 좌표의 정보를 조회하는 기능

10000

확인

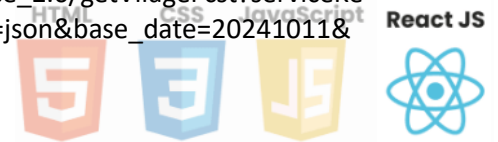
요청변수(Request Parameter)

항목명	샘플데이터	
ServiceKey	8qw7g%2FC%2BM	공공데이터포털에서 받은 인증키
pageNo	1	페이지번호
numOfRows	1000	한 페이지 결과 수
dataType	json	요청자료형식(XML/JSON) Default: X
base_date	20210628	'21년 6월 28일발표
base_time	0500	05시 발표
nx	55	예보지점의 X 좌표값
ny	127	예보지점의 Y 좌표값

```
▼"item": [  
  ▼{  
    "baseDate": "20241011",  
    "baseTime": "0500",  
    "category": "TMP",  
    "fcstDate": "20241011",  
    "fcstTime": "0600",  
    "fcstValue": "13",  
    "nx": 55,  
    "ny": 127  
  },  
]
```

https://apis.data.go.kr/1360000/VilageFcstInfoService_2.0/getUltraSrtFcst?serviceKey=apikey&pageNo=1&numOfRows=1000&dataType=json&base_date=20241011&base_time=0630&nx=55&ny=127

https://apis.data.go.kr/1360000/VilageFcstInfoService_2.0/getVilageFcst?serviceKey=apikey&pageNo=1&numOfRows=1000&dataType=json&base_date=20241011&base_time=0500&nx=55&ny=127



해결문제

일기예보 선택



--시도선택-- 

초단기예보

단기예보

부산광역시 단기예보 (2025.03.23)

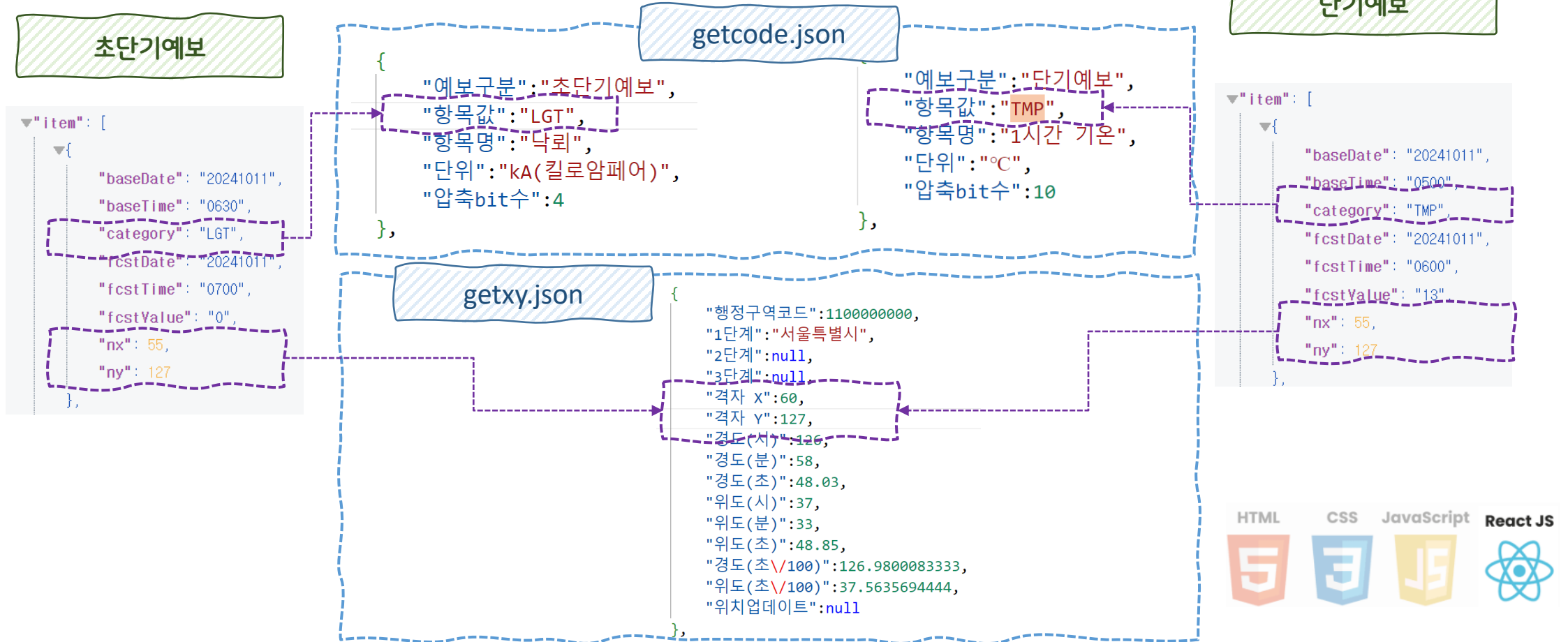
하늘상태(SKY) 

항목명	예측일자	예측시간	예측값
하늘상태(SKY)	2025.03.23	06:00	맑음(☀)
하늘상태(SKY)	2025.03.23	07:00	맑음(☀)
하늘상태(SKY)	2025.03.23	08:00	맑음(☀)
하늘상태(SKY)	2025.03.23	09:00	맑음(☀)
하늘상태(SKY)	2025.03.23	10:00	맑음(☀)
하늘상태(SKY)	2025.03.23	11:00	맑음(☀)
하늘상태(SKY)	2025.03.23	12:00	맑음(☀)
하늘상태(SKY)	2025.03.23	13:00	맑음(☀)
하늘상태(SKY)	2025.03.23	14:00	맑음(☀)
하늘상태(SKY)	2025.03.23	15:00	맑음(☀)
하늘상태(SKY)	2025.03.23	16:00	맑음(☀)
하늘상태(SKY)	2025.03.23	17:00	맑음(☀)
하늘상태(SKY)	2025.03.23	18:00	구름많음(☁)
하늘상태(SKY)	2025.03.23	19:00	구름많음(☁)
하늘상태(SKY)	2025.03.23	20:00	구름많음(☁)



해결문제 - 데이터 수집

• 기상청_단기예보 ((구)_동네예보) 조회서비스



해결문제

