

Case1 : Context Menu

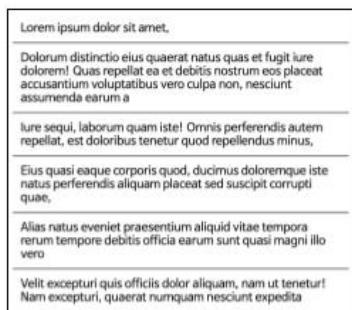
케이스 주제

Q. 버튼을 클릭하면 컨텍스트 메뉴가 나타나고, 메뉴를 선택하거나 그 외의 부분을 클릭하면 사라지는 팝오버 컴포넌트를 구현하세요

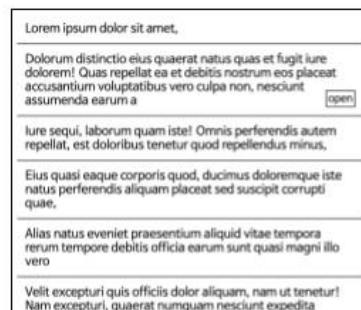
기능 요구사항

- 임의의 메뉴 목록에 대하여 각 메뉴를 클릭하면 해당 메뉴에 관련된 문구가 떠있는 형태(팝오버)로 나타난다.
- 팝오버 상태에서 해당 문구를 클릭하면 해당 메뉴가 사라진다.
- 팝오버 상태에서 다른 메뉴를 클릭하면 해당 메뉴의 팝오버가 나타난다. 이 때 기존의 팝오버는 사라진다.

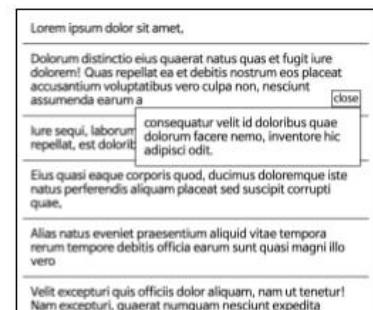
기능 작동 이미지



1) 최초상태.



2) 메뉴 하나에 마우스 오버



3) 메뉴 클릭시



4) 다른 메뉴에 마우스 오버



5) 다른 메뉴 클릭시

문제

- q1. 문제 상황에 대하여 Java Script로 동작을 구현시킬 수 있는 코드를 작성해보세요
- q2. 문제 상황에 대하여 jquery로 동작을 구현시킬 수 있는 코드를 작성해보세요
- q3. 문제 상황에 대하여 HTML(detail 태그)로 동작을 구현시킬 수 있는 코드를 작성해보세요
- q4. 문제 상황에 대하여 React로 동작을 구현시킬 수 있는 코드를 작성해보세요
- q5. 문제 상황에 대하여 React-CreatePortal 기능으로 동작을 구현시킬 수 있는 코드를 작성해보세요

주요 학습 키워드

이벤트 리스너, 이벤트 위임 구현 원리, detail 태그, React-Create Portal 등

작성해주셔야 하는 question 파일경로

q1

```
./question/q1_js/index.js  
./question/q1_js/style.css
```

q2

```
./question/q2_jquery/index.js  
./question/q2_jquery/style.css
```

q3

```
./question/q3_html-js/index.js  
./question/q3_html-js/style.css
```

q4

```
./question/q4_react.js/src/App.js
```

q5

```
./question/q5_react.js-createportal/src/App.js
```

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/q1_js

터미널

```
$ npm install  
$ npm start
```

q2

경로 ./question/q2_jquery

터미널

```
$ npm install  
$ npm start
```

q3

경로 ./question/q3_html-js

터미널

```
$ npm install  
$ npm start
```

q4

경로 ./question/q4_react.js

터미널

```
$ npm install  
$ npm start
```

q5

경로 ./question/q5_react.js-createportal

터미널

```
$ npm install  
$ npm start
```

Case2 : Scroll spy

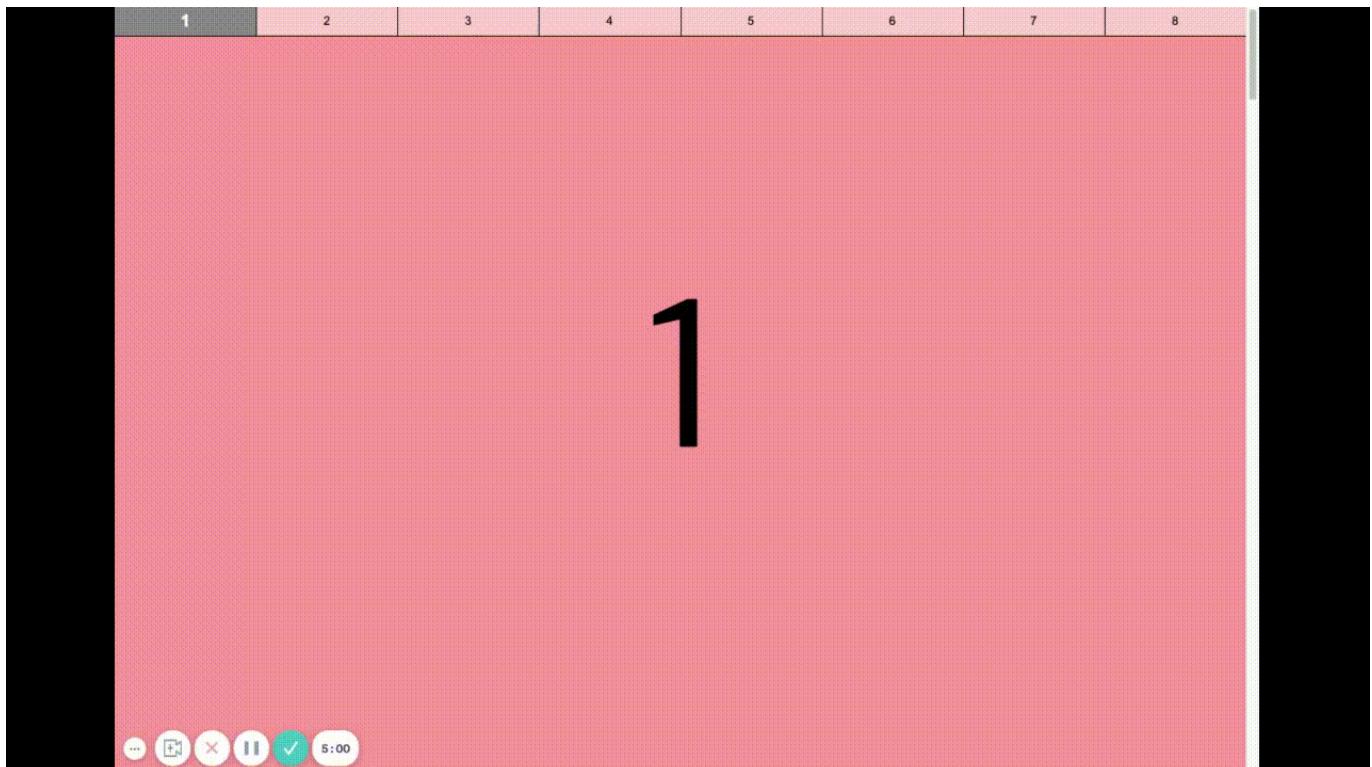
케이스 주제

Q. 스크롤스파이 구현하기

기능 요구사항

1. 최초 화면: 상단메뉴의 '1'이 활성화되어 있음.
2. 스크롤을 내리면서 나오는 숫자가 상단 메뉴에 활성화된 숫자와 일치하도록 합니다.
3. 상단 메뉴를 클릭하여, 해당 숫자가 적힌 스크롤위치로 이동할 수 있습니다.

기능 작동 이미지



실행 방법 / 풀이 방법 안내

문제 풀기 방식 :

1. 레포지토리를 clone
2. 터미널에서 각 문제 폴더 디렉토리로 이동하여 `npm install`로 의존성을 설치
3. `package.json`을 참고하여, 명시된 scripts 명령어로 개발서버 실행.
4. 코드 수정하면서 문제 해결하세요

기본 번들러로 `parcel`을 사용했습니다. - `react` 문제의 경우, `react-scripts` 사용. 문제 디렉토리에서 `npm start` 또는 `npx parcel index.html watch`로 개발서버를 실행하세요.

문제

- q1. `offsetTop`, `scrollTop`, `clientHeight`의 상관관계를 통해 해당 기능을 구현하시오.
- q2. `resize`에 무관하게 동작하게끔 기능을 구현하시오.
- q3. `resize listener`를 적용하여 동작을 구현하시오.
- q4. `throttle`과 `debounce`를 통해 해당 동작을 구현하시오.
- q5. `Intersection Observer`를 활용하여 기능을 구현하시오.
- q6. `Intersection Observer`를 활용하여 React로 해당 기능을 구현하시오.

주요 학습 키워드

스크롤 동작 감지
`offsetTop/scrollTop/clientHeight`의 상관관계
이벤트 리스너, `mousewheel`, `throttle`과 `debounce`, `IntersectionObserver`,
`useEffect`

작성해주셔야 하는 question 파일경로

q1

```
./question/q1_js_1/index.js
```

q2

```
./question/q2_js_2/index.js
```

q3

```
./question/q3_js_resize_listener/index.js
```

q4

```
./question/q4_js_throttle/util.js
```

q5

```
./question/q5_js_Intersection_Observer/index.js
```

q6

```
./question/q6_react_Intersections_Observer/src/App.js
```

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/q1_js_1`

터미널

```
$ npm install  
$ npm start
```

q2

경로 `./question/q2_js_2`

터미널

```
$ npm install  
$ npm start
```

q3

경로 `./question/q3_js_resize_listener`

터미널

```
$ npm install  
$ npm start
```

q4

경로 `./question/q4_js_throttle`

터미널

```
$ npm install  
$ npm start
```

q5

경로 `./question/q5_js_Intersection_Observer`

터미널

```
$ npm install  
$ npm start
```

q6

경로 `./question/q6_react_Interaction_Observer`

터미널

```
$ npm install  
$ npm start
```

Case3 : Infinite scroll

케이스 주제

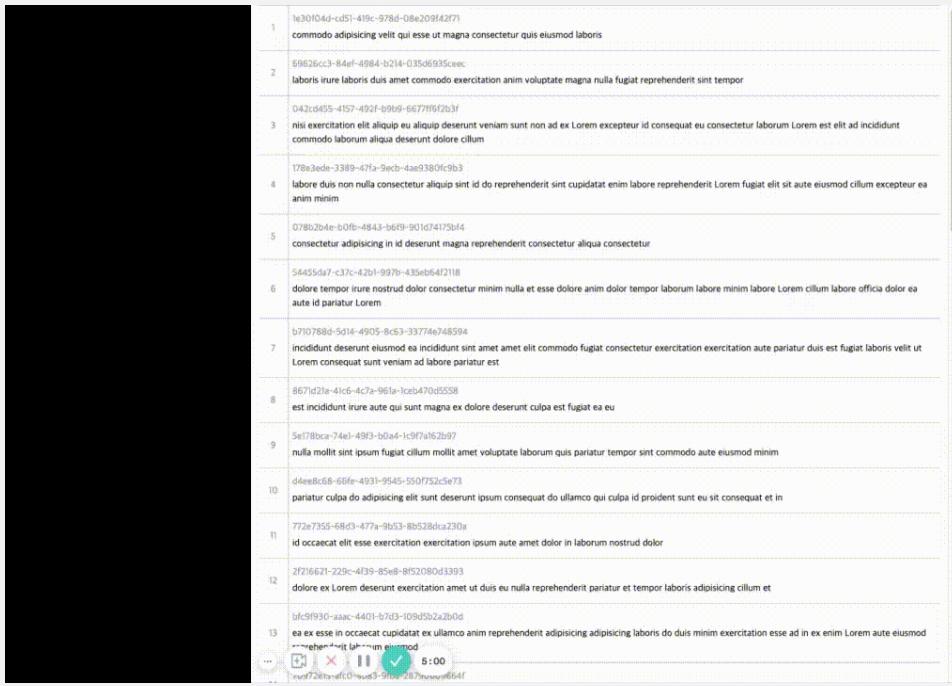
Q. 무한 스크롤 목록뷰를 구현하세요.

기능 요구사항

1. 최초에는 20개의 목록을 불러옵니다.
2. 스크롤을 최하단으로 이동시 ‘loading’ 상태표시가 나타나며, 이후의 20개 목록을 더 가져옵니다.
3. 로딩 완료시 ‘loading’ 표시가 사라지며, 가져온 목록이 하단에 추가됩니다. (무한 반복)

기능 작동 이미지

1. 최초 접속시



실행 방법 / 풀이 방법 안내

문제 풀기 방식 :

1. 터미널에서 각 문제 폴더 디렉토리로 이동하여 npm install로 의존성을 설치
2. package.json을 참고하여, 명시된 scripts 명령어로 개발서버 실행.
3. 코드 수정하면서 문제 해결하세요

기본 번들러로 parcel을 사용했습니다. - react 문제의 경우, react-scripts 사용. 문제 디렉토리에서 npm start 또는 npx parcel index.html watch로 개발서버를 실행하세요.

문제

- q1. Javascript - scrollTop, clientHeight, scrollHeight 등의 요소의 프로퍼티를 활용하여 해결하시오.
- q2. Javascript - 앞서 작성한 코드에 debounce를 적용해 기능을 구현하시오.
- q3. Javascript - 다음으로 IntersectionObserver를 활용해 기능을 구현하시오.
- q4. React - 마지막으로, 리액트에서 IntersectionObserver를 활용해 기능을 구현하시오.

주요 학습 키워드

스크롤 동작 감지, 이벤트 리스너, scrollTop, mousewheel, IntersectionObserver, useEffect

작성해주셔야 하는 question 파일경로

q1

./question/q1_js/index.js

q2

```
./question/q2_js_debounce_trottle/index.js
```

```
./question/q2_js_debounce_trottle/util.js
```

q3

```
./question/q3_js_Intersection_Observer/index.js
```

q4

```
./question/q4_react_Interaction_Observer/src/FetchMore.js
```

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/q1_js`

터미널

```
$ npm install  
$ npm start
```

q2

경로 `./question/q2_js_debounce_trottle`

터미널

```
$ npm install  
$ npm start
```

q3

경로 `./question/q3_js_Intersection_Observer`

터미널

```
$ npm install  
$ npm start
```

q4

경로 `./question/q4_react_Interaction_Observer`

터미널

```
$ npm install  
$ npm start
```



Case4 : Dark mode

케이스 주제

Q. 다음과 같이 토글 버튼을 클릭하여 테마(다크 모드/라이트 모드)를 설정하면 테마가 뷰에 반영되도록 구현해보자. 테마는 로컬스토리지에 저장하여 웹페이지를 리로드하거나 다시 접근했을 때 저장된 테마를 적용하도록 한다.

기능 요구사항

1. 로컬스토리지에 저장되어 있는 테마(다크 모드/라이트 모드)를 기준으로 초기 렌더링한다.
2. 로컬스토리지에 저장된 테마가 없다면 라이트 모드로 초기 렌더링한다.
3. 테마를 적용하여 렌더링할 때 기존 테마가 변경되어 깜빡거리는 현상(flash of incorrect theme, FOIT)이 발생하지 않도록 한다.
4. 토글 버튼을 클릭하면 로컬스토리지에 테마를 저장하고 저장된 테마를 기준으로 다시 렌더링한다.

기능 작동 이미지

Light / Dark Mode - Toggle Button



 Lorem ipsum dolor, sit amet consectetur adipisicing elit. Laborum optio ab porro magni
 in sunt ipsam, doloremque minima, itaque sapiente consequatur, repellat velit
 voluptatum accusantium aperiam. Nostrum sunt reprehenderit nemo!

뷰의 기본 템플릿은 다음과 같다. body 요소에 dark 클래스를 추가하면 다크 모드가 적용되고 body 요소에서 dark 클래스를 제거하면 라이트 모드가 적용된다.

Light / Dark Mode - Toggle Button



 Lorem ipsum dolor, sit amet consectetur adipisicing elit. Laborum optio ab porro magni
 in sunt ipsam, doloremque minima, itaque sapiente consequatur, repellat velit
 voluptatum accusantium aperiam. Nostrum sunt reprehenderit nemo!

The screenshot shows the browser's developer tools with the 'Elements' tab selected. The DOM tree is visible, showing the structure of the HTML document. A green toggle switch button is shown at the top left. The body element has a class attribute set to 'dark'. The 'Properties' tab is also visible, showing the styles applied to the body element, including 'font-family: Open Sans' and 'font-weight: 300;'. The 'Computed' tab shows the final styles being applied.

문제

1. JS

로컬스토리지에 저장된 테마(다크 모드/라이트 모드)를 기준으로 초기 렌더링
로컬스토리지에 저장된 테마가 없으면 라이트 모드로 초기 렌더링

2. JS

로컬스토리지에 저장된 테마가 없을 때 window.matchMedia 메서드로 사용자 OS 테마를 감지해 이를 테마에 적용
로컬스토리지에 저장된 테마가 있으면 사용자 OS 테마보다 이를 우선하여 적용

3. React 바닐라 자바스크립트로 구현한 dark mode는 body 요소에 클래스를 추가/제거하는 방식으로 동작한다. React에서도 이 방식을 사용하면 컴포넌트에서 body 요소를 조작하는 부수 효과(side effect)에 의존하게 되므로 직관적이지 않고 컴포넌트의 재사용이 어려워지며 FOIT(flash of incorrect theme)을 방지하기도 번거롭다.

요구 사항은 다음과 같다.

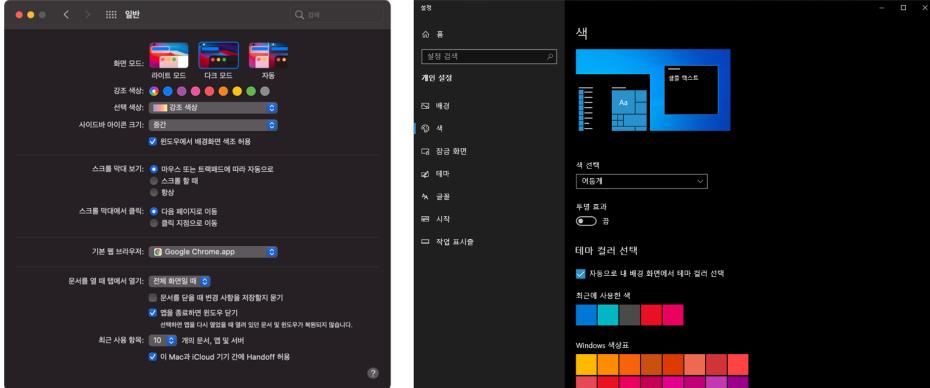
함수 컴포넌트와 훙을 사용해 구현한다.
Styled-components의 ThemeProvider를 사용하면 간단하게 테마를 전역 관리할 수 있다. 테마(dark/light)를 객체로 정의하고 Styled-components의 ThemeProvider를 사용해 테마가 필요한 컴포넌트에게 전달한다.

주요 학습 키워드

localStorage
prefers-color-scheme
window.matchMedia
styled-components: theming
useState
useEffect
Context API
useContext

참고

Windows와 macOS 등은 운영 체제 레벨에서 사용자 테마(다크 모드/라이트 모드)를 설정할 수 있다.



CSS의 prefers-color-scheme media query나 자바스크립트의 window.matchMedia 메서드를 사용하면 운영 체제 레벨에서 설정한 사용자 테마를 감지할 수 있다.

```
prefers-color-scheme: Hello darkness, my old friend
```

prefers-color-scheme media query와 window.matchMedia 메서드의 간단한 예제는 다음과 같다.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <style>
      .themed {
        display: flex;
        justify-content: center;
        align-items: center;
        width: 200px;
        height: 100px;
        background-color: rgb(250, 250, 250);
      }
      .themed::after {
        content: 'Light mode(default)';
      }

      @media (prefers-color-scheme: dark) {
        .themed {
          background-color: #000;
          color: #fff;
        }
        .themed::after {
          content: 'Dark mode detected';
        }
      }
    </style>
  </head>
```

```
<body>
  <div class="themed"></div>
  <script>
    // https://web.dev/prefers-color-scheme
    // https://caniuse.com/?search=prefers-color-scheme
    const darkModeMediaQuery = window.matchMedia('(prefers-color-scheme: dark)');
    console.log(darkModeMediaQuery);
    // MediaQueryList {media: "(prefers-color-scheme: dark)", matches: true, onchange: null}

    darkModeMediaQuery.addListener(e => {
      const darkModeOn = e.matches;
      console.log(`Dark mode is ${darkModeOn ? '🌙 on' : '☀️ off'}`);
    });
  </script>
</body>
</html>
```

작성해주셔야 하는 question 파일경로

q1

./question/1.js-1/index.js

q2

./question/2.js-2/index.js

q3

./question/3.react/src/App.js

이외 필요한 디렉토리 / 파일구조는 각자 작성하시면 됩니다.

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/1.js-1

index.html 열기

q2

경로 ./question/2.js-2

index.html 열기

q3

경로 ./question/3.react

터미널

```
$ npm install  
$ npm start
```

Case5 : Stop watch

케이스 주제

Q. 아래와 같이 작동하는 스톱워치를 구현하시오.

기능 요구사항

1. 스톱워치의 시간은 mm:ss:ms 형식(예시 '01:59:89')으로 표시한다.

구분 의미 범위 비고

mm 분 0 ~

ss 초 0 ~ 59

ms 미리초 0 ~ 99 미리초는 100분의 1초를 나타내지만 10ms 단위로 표시한다.

2. 컨트롤 버튼

스톱워치는 2개의 컨트롤 버튼을 가진다.

왼쪽 버튼: 클릭할 때마다 Start/Stop으로 토글된다.

오른쪽 버튼: 오른쪽 버튼은 아래와 같이 왼쪽 버튼에 종속적이다. 왼쪽 버튼이 Start이면 오른쪽 버튼은 Reset이고 왼쪽 버튼이 Stop이면 오른쪽 버튼은 Lap이다.

왼쪽 버튼 오른쪽 버튼

Start Reset

Stop Lap

각 버튼의 기능은 다음과 같다.

버튼 기능

Start 스톱워치를 시작한다.

Stop 스톱워치를 일시정지시킨다.

Reset 스톱워치와 랩 타임을 초기화한다. 스톱워치의 현재 시간이 '00:00:00'이면 disabled 상태이어야 한다.

Lap 랩 타임을 기록한다.

기능 작동 이미지

Stopwatch

00:00:00

Start

Reset

문제

스톱워치 경과 시간을 '00:00:00' 형식의 문자열로 변환

스톱워치 경과 시간을 렌더링

랩 타임 렌더링

Start/Stop 버튼 클릭 이벤트 핸들러

Reset/Lap 버튼 클릭 이벤트 핸들러

2. React

함수 컴포넌트와 흑을 사용해 구현한다.

스타일은 CSS, Sass, CSS Module, styled-components 중 어느 것을 사용해도 좋으나 가급적 styled-components 사용을 권장한다.

주요 학습 키워드

CSS 그리드 레이아웃

setInterval

Document.createDocumentFragment

useState

useEffect

작성해주셔야 하는 question 파일경로

q1

`./question/1.js/index.js`

q2

`./question/2.react/src/App.js`

이외 필요한 디렉토리 / 파일구조는 각자 작성하시면 됩니다.

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/1.js`

[index.html 열기](#)

q6

경로 [./question/2.react](#)

터미널

```
$ npm install  
$ npm start
```

Case6 : Tabs

케이스 주제

Q. 아래와 같이 작동하는 탭 메뉴를 구현하시오.

기능 요구사항

요구 사항은 아래와 같다.

1. 비동기 함수인 fetchTabsData 함수를 사용해 탭 메뉴 정보를 담고 있는 배열을 전달받아 탭 메뉴를 생성한다.
2. fetchTabsData 함수는 프로미스를 반환하며 이 프로미스가 fulfilled 상태가 될 때까지는 1초 소요된다. 프로미스가 fulfilled 상태가 될 때까지 스피너(.spinner 요소)를 표시한다.
3. 탭 정보를 담고 있는 배열의 length는 가변적이다.

React

1. 함수 컴포넌트와 흑을 사용해 구현한다.
2. 스타일은 CSS, Sass, CSS Module, styled-components 중 어느 것을 사용해도 좋으나 가급적 styled-components 사용을 권장한다.

실행 방법 / 문제 풀이 방법

q1. Javascript - index.html 실행 q2. React

```
$ yarn install $ yarn start
```

주요 학습 키워드

```
비동기 처리와 Promise  
React hook을 사용해 데이터 패칭하기
```

작성해주셔야 하는 question 파일경로

q1

```
./question/q1_js/src/app.js
```

q2

```
./question/q2_react/src/App.js
```

이외 필요한 디렉토리 / 파일들은 각자 작성하시면 됩니다.

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/q1_js`

index.html 열기

q2

경로 `./question/q2_react`

터미널

```
$ yarn install  
$ yarn start
```

Case7 : Analog clock

케이스 주제

Q. 아래와 같이 동작하는 아날로그 시계를 구현해보십시오.

기능 요구사항

JavaScript

1. 시계의 시침(.hand.hour 요소), 분침(.hand.minute 요소), 초침(.hand.second 요소)을 1초 간격으로 회전시켜 현재 시간을 표시하시오.
2. 뷰의 기본 템플릿을 그대로 사용해도 좋으나, 더 나은 방법이 있다면 변경해도 좋습니다.

React

3. 함수 컴포넌트와 흑을 사용해 구현하시오.
4. 스타일은 CSS, Sass, CSS Module, styled-components 중 어느 것을 사용해도 좋으나 가급적 styled-components 사용을 권장한다.

기능 작동 이미지

Analog clock



문제

1. JS : Javascript로 해당 기능을 구현하시오.
2. React : React로 해당 기능을 구현하시오.

styled-components를 사용하여 뷰를 구현하시오.
useRef와 휙을 사용하여 1번과 동일한 동작을 구현하시오.

주요 학습 키워드

CSS 변수(CSS 커스텀 프로퍼티)
styled-components
useRef
useEffect

작성해주셔야 하는 question 파일경로

q1

```
./question/1.js/index.js
```

q2

```
./question/2.react-styled-component/src/App.js
```

이외 필요한 디렉토리 / 파일 구조는 각자 작성하시면 됩니다.

q3

```
./question/3.react-useRef/src/components/AnalogClock.js
```

출제자 강사님 코드 기반으로, 해당 경로에서 요구한 문제사항을 해결해주세요 이외 필요한 디렉토리 / 파일 구조는 각자 작성하시면 됩니다.

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/1.js`

index.html 열기

q2

경로 `./question/2.react-styled-component`

터미널

```
$ npm install  
$ npm start
```

q3

경로 `./question/3.react-useRef`

터미널

```
$ npm install  
$ npm start
```

Case8 : Carousel

케이스 주제

Q. 아래와 같이 작동하는 캐러셀을 구현하시오.

캐러셀(Carousel)은 컨텐츠를 슬라이드 형태로 순환하며 표시하는 UI를 말한다. 캐러셀은 비생산적인 디자인 패턴이라는 주장이 있기도 하지만 사용자가 스크롤을 내리지 않은 상태에서도 많은 정보를 노출할 수 있는 장점이 있어 많은 웹사이트에서 사용하고 있다.

기능 요구사항

요구 사항은 아래와 같다.

1. 무한 루핑 기능을 지원한다.
2. 슬라이딩 애니메이션을 지원한다.
3. 각 슬라이드의 width/height는 가변적이다. 단, 모든 슬라이드의 width/height는 동일하다.
4. 슬라이드 이동 버튼을 연타해도 이상없이 동작해야 한다.
5. 캐러셀 슬라이더를 표시할 HTML 요소와 슬라이드 이미지의 url로 구성된 배열을 전달하면 동적으로 캐러셀 슬라이더를 생성한다.

캐러셀 슬라이더 알고리즘

정상적으로 한칸씩 이동하는 경우

.carousel 요소의 위치를 고정시키고 넘치는 자식 요소를 감추고
.carousel-slides 요소 .carousel 요소의 width만큼 좌우로 이동시킨다.

마지막 슬라이더를 클론하여 추가

```
.carousel-slides {  
    --currentSlide: 1;  
    --duration: 0;  
    display: flex;  
    transition: transform calc(var(--duration) * 1ms) ease-out;  
    transform: translate3D(calc(var(--currentSlide) * -100%), 0, 0);  
}  
  
.carousel-slides {  
    --currentSlide: 2;  
    --duration: 500;  
    display: flex;  
    transition: transform calc(var(--duration) * 1ms) ease-out;  
    transform: translate3D(calc(var(--currentSlide) * -100%), 0, 0);  
}
```

첫번째 슬라이더를 클론하여 추가

슬라이더의 선두 또는 가장 뒤에서 이동하는 경우

.carousel 요소의 위치를 고정시키고 넘치는 자식 요소를 감추고
.carousel-slides 요소 .carousel 요소의 width만큼 좌우로 이동시킨다.

마지막 슬라이더를 클론하여 추가

```
.carousel-slides {  
    --currentSlide: 1;  
    --duration: 0;  
    display: flex;  
    transition: transform calc(var(--duration) * 1ms) ease-out;  
    transform: translate3D(calc(var(--currentSlide) * -100%), 0, 0);  
}  
  
.carousel-slides {  
    --currentSlide: 1;  
    --duration: 0;  
    display: flex;  
    transition: transform calc(var(--duration) * 1ms) ease-out;  
    transform: translate3D(calc(var(--currentSlide) * -100%), 0, 0);  
}
```

첫번째 슬라이더를 클론하여 추가

(예: 데미션 없이) 슬라이더 3로 이동

```
.carousel-slides {  
    --currentSlide: 3;  
    --duration: 0;  
    display: flex;  
    transition: transform calc(var(--duration) * 1ms) ease-out;  
    transform: translate3D(calc(var(--currentSlide) * -100%), 0, 0);  
}
```

React

- 함수 컴포넌트와 흑을 사용해 구현한다.
- 스타일은 CSS, Sass, CSS Module, styled-components 중 어느 것을 사용해도 좋으나 가급적 styled-components 사용을 권장한다.

주요 학습 키워드

CSS 변수(CSS 커스텀 프로퍼티)
HTMLElement.offsetWidth
HTMLElement:transitionend event
styled-components

useState

작성해주셔야 하는 question 파일경로

q1_js

./question/q1_js/src/app.js

q2_react

./question/q2_react/src/App.js

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/q1_js

index.html 열기

q2

경로 ./question/q2_react

터미널

```
$ yarn install  
$ yarn start
```

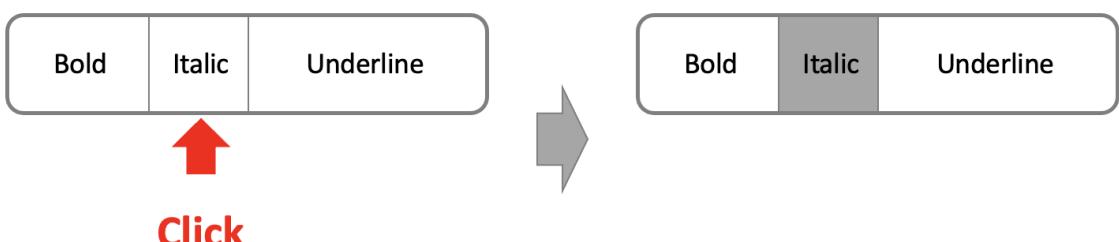
Case9 : Toggle button

케이스 주제

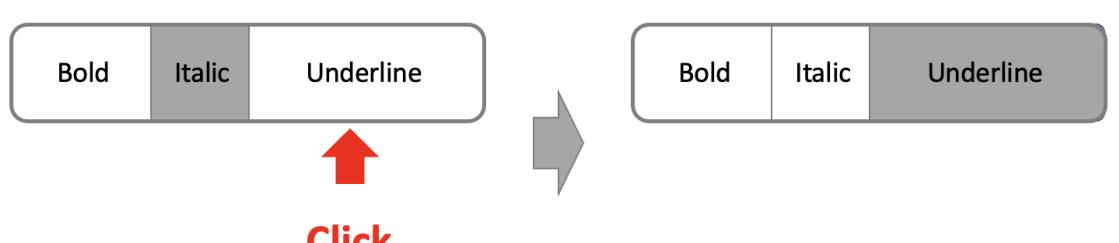
Q. on, off 기능 (toggle 기능)을 가지는 버튼 리스트 컴포넌트를 구현하십시오.

기능 요구사항

- 각 버튼이 선택이 되었는지 안되었는지를 확인할 수 있다.



- 여러개의 버튼 중 하나의 버튼만 toggle 할 수 있다.



기능 작동 이미지



문제

- q1. 데이터에 따른 버튼리스트를 가로로 출력하시오.
- q2. 한개의 버튼만이 on이 될 수 있도록 하시오.
- q3. 선택된 버튼의 index를 application으로 전달 하시오.

주요 학습 키워드

버튼 스타일 적용
컴포넌트와 어플리케이션과의 통신

작성해주셔야 하는 question 파일경로

q1

.src/question/toggle-button/index.js

line : 33

q2

.src/question/toggle-button/index.js

line : 44

q2

.src/question/toggle-button/index.js

line : 47

실행 방법 및 의존성 모듈 설치

경로 ./

(root directory)

터미널

```
$ npm install
$ npm run dev
```



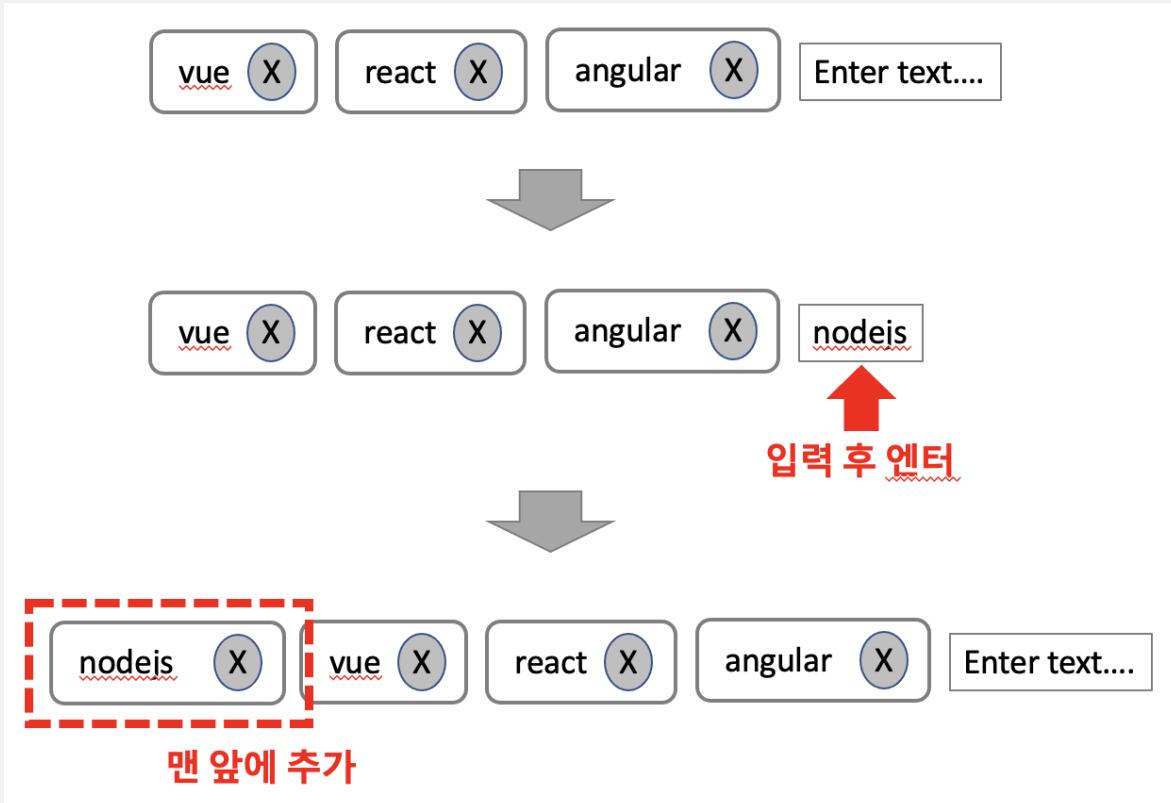
Case10 : Chips UI

케이스 주제

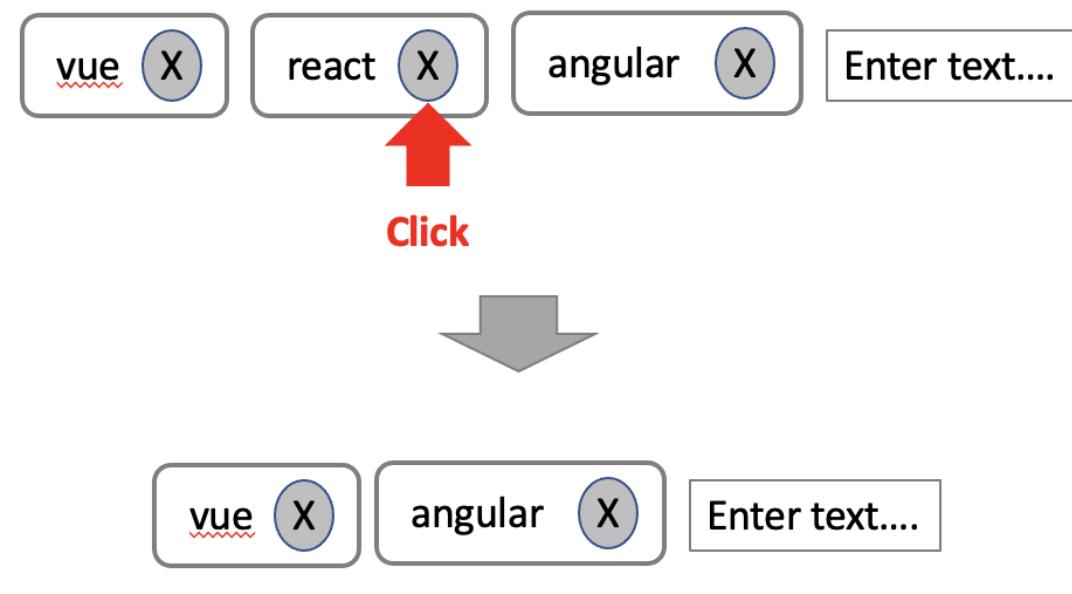
Q. 임의 문자열을 추가 삭제할 수 있는 (SNS 태그 입력) 컴포넌트를 구현하십시오.

기능 요구사항

1. 텍스트 입력 후 엔터를 치면 리스트의 맨 앞에 추가가 된다.



2. 리스트 중에 아이템을 삭제 할 수 있다.



기능 작동 이미지



문제

- q1. 데이터에 따른 문자열 리스트와 입력 폼을 함께 출력하시오.
- q2. 입력된 문자열은 키보드 이벤트 엔터를 통해 리스트의 앞단에 추가 하시오.
- q3. 입력된 문자열을 삭제할 수 있도록 하시오.
- q4. 입력된 문자열에 대한 데이터를 가져올 수 있도록 하시오.

주요 학습 키워드

동적으로 추가되는 element

template 분리
element.insertAdjacentElement 함수 활용

작성해주셔야 하는 question 파일경로

q1

.src/question/index.js

line : 43

q2

.src/question/index.js

line : 54

q3

.src/question/index.js

line : 72

q4

.src/question/index.js

line : 76

실행 방법 및 의존성 모듈 설치

경로 ./

(root directory)

터미널

```
$ npm install  
$ npm run dev
```

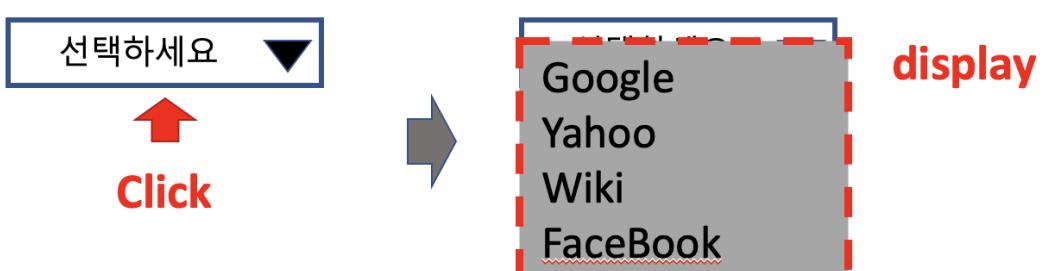
Case11 : Dropdown menu

케이스 주제

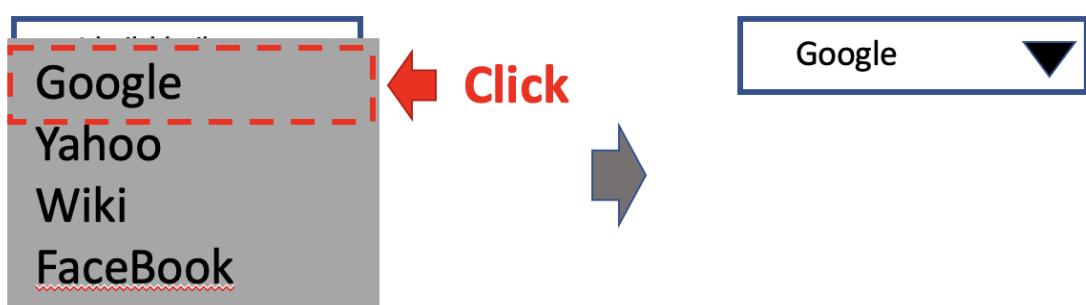
Q. 라벨 클릭 시 선택 가능한 list가 출력되고, dropdown item 선택 시 선택된 데이터의 라벨을 출력해주는 dropdown item을 만드시오.

기능 요구사항

- 라벨 클릭 시 선택 가능한 list가 출력된다.

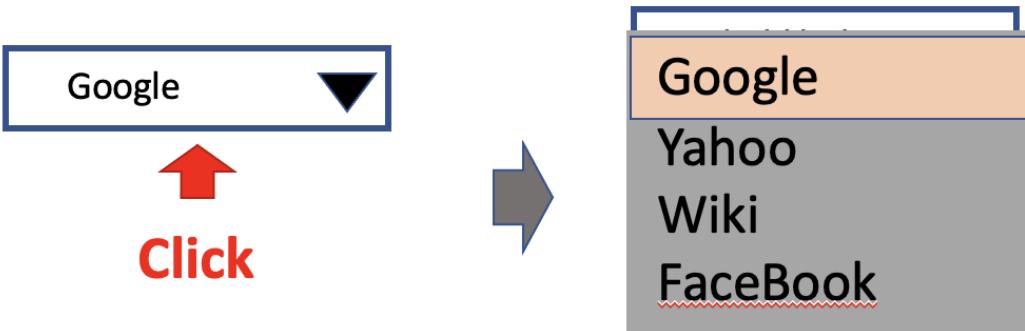


- dropdown item을 선택 시 선택된 데이터를 라벨에 출력한다.



- dropdown item을 선택 시 선택된 데이터를 외부로 전달할 수 있도록 한다.

4. dropdown item을 선택 시 선택된 데이터에 선택 여부를 표시한다.



5. dropdown 외에 영역 클릭 시 아무런 이벤트 없이 리스트를 닫는다.



6. 디자인 템플릿을 변경할 수 있도록 템플릿을 분리한다.

기능 작동 이미지

A screenshot of a dropdown menu interface. The menu is titled "선택하세요." with a dropdown arrow. The list contains the following items:

- User: Kenneth, favorites: Google
- User: John, favorites: Google
- User: Daniel, favorites: Yahoo
- User: Peter, favorites: FaceBook
- User: Brian, favorites: Wiki

문제

- q1. label position에 dropdown list 영역을 출력하시오.
q2. backdrop 영역 클릭 시의 이벤트를 처리하시오.
q3. label 영역 클릭 시의 이벤트를 처리하시오.

q4. 함수(dispatchEvent)를 참조하여 id, label 값을 인자로 넘겨 이벤트를 발생시키시오.

주요 학습 키워드

click event target element 설정에 대한 고민
selector.getBoundingClientRect() 함수
arrow function의 활용
Template literals 을 활용한 html template 관리

작성해주셔야 하는 question 파일경로

q1

`./src/question/dropdown/index.js`

line : 80

q2

`./src/question/dropdown/index.js`

line : 95

q3

`./src/question/dropdown/index.js`

line : 101

q4

`./src/question/dropdown/index.js`

line : 118

실행 방법 및 의존성 모듈 설치

경로 ./

(root directory)

터미널

```
$ npm install
$ npm run dev
```



Case12 : Auto Complete 2

케이스 주제

Q. 검색어를 입력하는 동시에 DB에 저장되어 있는 유관 검색어를 실시간으로 출력하는 기능을 구현하십시오.

기능 요구사항

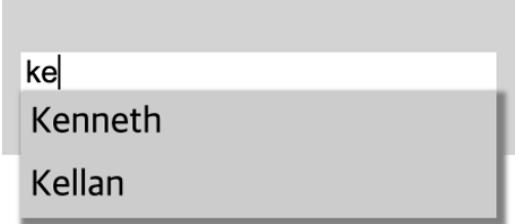
1. 키보드 이벤트를 자연시간(debounce 기능)을 통해 request 횟수를 줄인다.



2. 검색 필드를 벗어나면 출력된 리스트는 사라진다.



기능 작동 이미지



<텍스트 입력 중>

<텍스트 입력 후>

문제

- q1. 검색결과 리스트를 text input 하단에 출력하시오.
- q2. debounce 기능을 구현하시오.
- q3. debounce 기능을 통해 request 호출을 최소화 하시오.
- q4. 검색 필드 외에 다른 곳을 클릭할 때 출력된 리스트를 보이지 않도록 하시오.

주요 학습 키워드

- input event의 활용도
- selector.getBoundingClientRect() 함수
- 지연시간을 적용하여 마지막 이벤트만 발생시키는 기능 (debounce)

작성해주셔야 하는 question 파일경로

q1

./src/question/auto-complete/index.js

line : 90

q2

./src/question/auto-complete/util/index.js

q3

```
./src/question/auto-complete/index.js
```

line : 104

q4

```
./src/question/auto-complete/index.js
```

line : 115

실행 방법 및 의존성 모듈 설치

경로 ./

(root directory)

터미널

```
$ npm install  
$ npm run dev
```

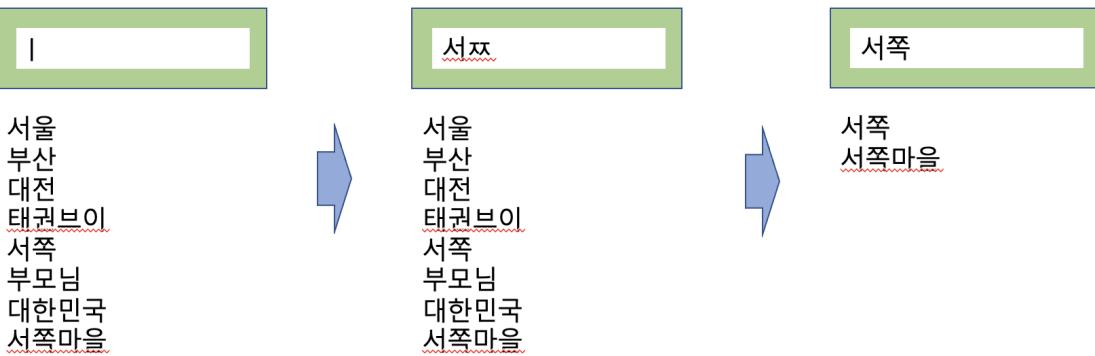
Case13 : Instant Search

케이스 주제

Q. 검색어를 입력하는 동시에 엔터키, 검색 버튼 누를 필요 없이 결과값을 즉시 반영할 수 있는 순간 검색 기능을 구현하십시오.

기능 요구사항

1. 키보드 이벤트를 지연시간(debounce 기능)을 통해 request 횟수를 줄인다.



2. 디자인 템플릿을 변경할 수 있도록 템플릿을 분리한다.

기능 작동 이미지

please enter keyword

서울
부산
대전
태권브이
서쪽
부모님
대한민국
서쪽마을

<텍스트 입력 전>

서

서울
서쪽
서쪽마을

<텍스트 입력 후>

문제

- q1. configuration을 참고하여 input element를 생성하시오. (기능과 디자인을 분리하기 위한 방법, src->question->instant-search->index.js)
- q2. debounce 기능을 구현하시오. (src->question->instant-search->util->index.js)
- q3. debounce 기능을 통해 가져온 데이터를 외부로 전달한다. (src->question->instant-search->index.js)
- q4. Promise를 사용하여 검색 키워드에 맞는 데이터를 가져와 리스트를 출력하시오. (src->question->instant-search->mock->word->index.js / src->index.js)

주요 학습 키워드

지연시간을 적용하여 마지막 이벤트만 발생시키는 기능 (debounce)
request api를 대신하기 위한 Promise 활용 (test를 위한 mock data 활용)
디자인 부분과 기능을 분리하기 위한 방법

작성해주셔야 하는 question 파일경로

q1

./src/question/instant-search/index.js

line : 43

q2

```
./src/question/instant-search/util/index.js
```

q3

```
./src/question/instant-search/index.js
```

line : 53

q4

```
./src/question/instant-search/mock/word/index.js
```

line : 30

```
./src/index.js
```

line : 35

실행 방법 및 의존성 모듈 설치

경로 ./

(root directory)

터미널

```
$ npm install  
$ npm run dev
```

Case14 : Stop watch

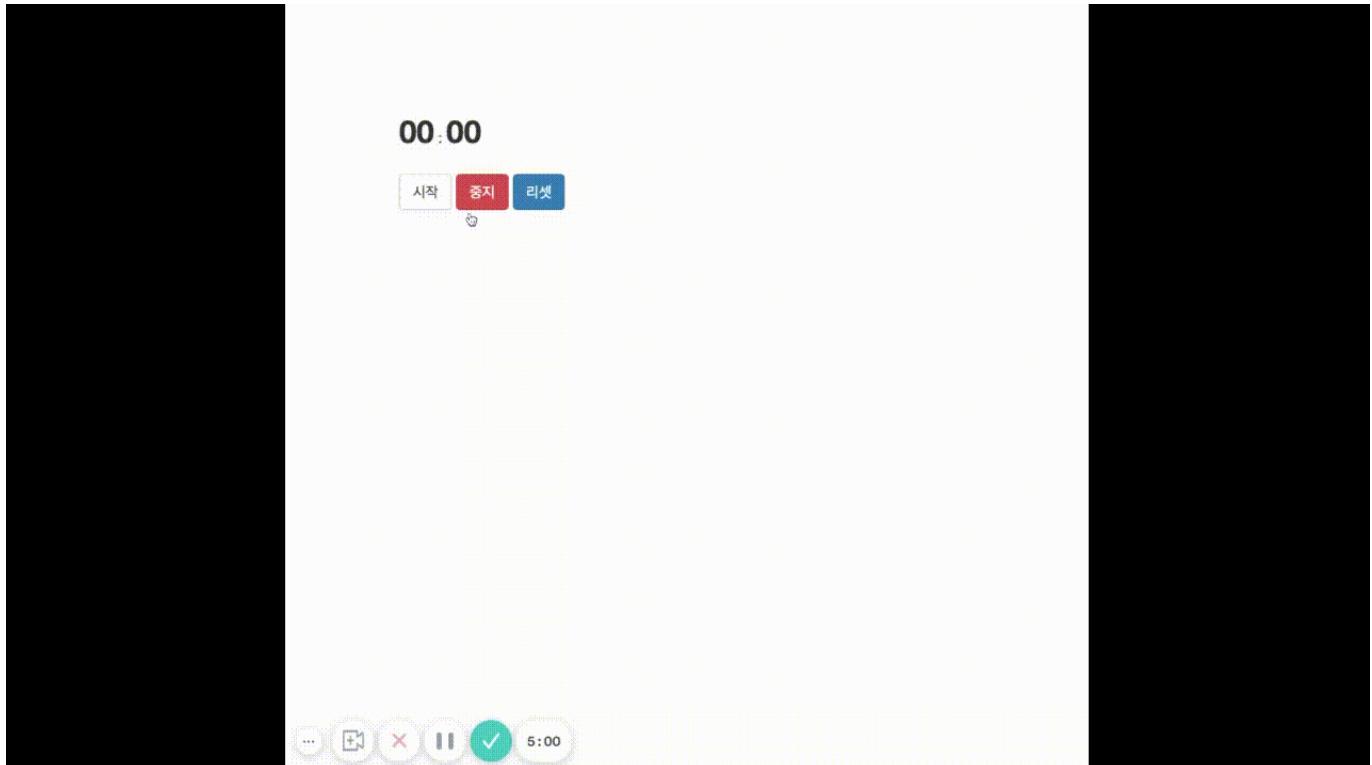
케이스 주제

Q. 타이머(스톱워치) 구현

기능 요구사항

1. 타이머 뷰를 구현하고 시작 / 중지 / 리셋에 따라 타이머가 동작하도록 한다.

기능 작동 이미지



문제

q1. Javascript - 주어진 템플릿(index.html)
)의 요소에 페이지에 맞는 요소를 표현하는 문제

1. isPending이 false 일때만 setTimeout을 재귀 호출하는 것을 만드세요.
2. reset 버튼을 누를 시 화면과 시간을 00:00:00으로 초기화되게 만드세요.
3. 타이머를 시작하는 부분을 만드세요.

q2. Jquery - Javascript로 구현한 기능을 동일하게 Jquery로도 구현해보는 문제
q3. React - 주어진 코드를 이해하고, 상태를 조작하는 이벤트핸들러와 라이프사이클 흐름을 활용하는 문제

1. setInterval 또는 setTimeout으로 반복시키는 부분을 완성해주세요.
2. 60초가 될때를 감지하는 부분을 완성한다.

주요 학습 키워드

데이터 모델 -> 뷰 방향의 단방향 데이터 바인딩 구현

작성해주셔야 하는 question 파일경로

q1

`./question/q1_js/index.js`

q2

`./question/q2_jquery/index.js`

q3

`./question/q3_react/src/App.js`

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/q1_js`

index.html 열기

q2

경로 `./question/q2_jquery`

index.html 열기

q3

경로 `./question/q3_react`

터미널

```
$ npm install  
$ npm start
```

Case15 : Modal window

케이스 주제

Q. 모달 팝업 구현

기능 요구사항

1. 팝업을 엽니다를 누르면 팝업이 뜬다.
2. x 버튼을 누르면 팝업이 닫힘
3. 팝업 이외 부분을 누르면 닫힘
4. 팝업 안을 눌렀을 때 닫히면 안됨

기능 작동 이미지

팝업을 엽니다.



문제

q1. JavaScript - 주어진 템플릿(q_index.html)
)의 요소에 페이지에 맞는 요소를 표현하는 문제 q2. jQuery

1. JavaScript로 구현한 기능을 동일하게 jQuery로도 구현해보는 문제
2. <https://jquerymodal.com/> 문서를 참조해서
3. 자바스크립트 소스 없이 완성하세요

q3. React - 주어진 코드를 이해하고, 상태를 조작하는 툴을 이용해서 완성

풀이 참고

q_index.html안에 body-blackout이 모달 바탕화면이고, Modal 보다 z-index 가 낮다. body-blackout 위로 Modal 이 뛴다.

팝업 띄우기 클릭시 is--visible 클래스를 활성화

주요 학습 키워드

CSS의 z-index를 사용하여 팝업 / 배경 순서 위치 지정하기

CSS의 pointer-events를 사용하여 popup이 없을 때, 이벤트가 발생하지 않도록 설정하기

classList를 사용하여 버튼에 click event 발생 시, 팝업 보여주고 닫아주기

작성해주셔야 하는 question 파일경로

q1

`./question/q1_js/q1_index.js`

q2

`./question/q2_jquery/q2_index.html`

<https://jquerymodal.com/> 문서를 참조해서 자바스크립트 소스 없이 CDN을 html파일에 추가하여 완성하세요

q3

`./question/q3_react/src/App.js`

`./question/q3_react/src/BodyBlackout.js`

`./question/q3_react/src/Modal.js`

출제자 강사님 코드 기반으로, 해당 경로에서 요구한 문제사항을 해결해주세요

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/q1_js`

index.html 열기

q2

경로 `./question/q2_jquery`

index.html 열기

q3

경로 `./question/q3_react`

터미널

```
$ npm install  
$ npm start
```

Case16 : Progress Bar

케이스 주제

Q. Progressbar 구현

기능 요구사항

- 총 4번 스텝 이동을 할수 있고 숫자는 조절 가능하다.
- 이전으로 하면 왼쪽으로(25%), 다음으로 오른쪽(25%)
- 다 채우거나 다 사라지면 클릭해도 작동안한다.

기능 작동 이미지



[이전](#) [다음](#)

실행 방법 / 풀이 방법 안내

jQuery 와 Vanilia.js 는 span 안에 width 값을 조절한다.
css 로는 애니메이션 효과를 주지않고, setInterval 함수를 사용한다.

q1. Javascript - q1_index.html 실행

q2. JQuery - q2_index.html 실행

q3. React

css를 활용한다. (src/ProgressBar.js 에 transition 참고)

다음이나 이전을 연속클릭해도 함께 동작하지 않게 한다.

문제

q1. Javascript - 주어진 템플릿(q_index.html)으로 캐러셀 구현

q2. Jquery - Javascript로 구현한 기능을 동일하게 Jquery로도 구현해보는 문제

q3. React - 주어진 코드를 이해하고, 상태를 조작하는 툥을 이용해서 완성

풀이 참고

jQuery animate 를 활용한다.

작성해주셔야 하는 question 파일경로

q1

```
./question/q1_js/q1_index.js
```

q2

```
./question/q2_jquery/q2_index.js
```

q3

```
./question/q3_react/src/App.js
```

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/s1_js

index.html 열기

q2

경로 ./question/s2_jquery

index.html 열기

q3

경로 ./question/s3_react

터미널

```
$ yarn install  
$ yarn start
```

Case17 : Carousel

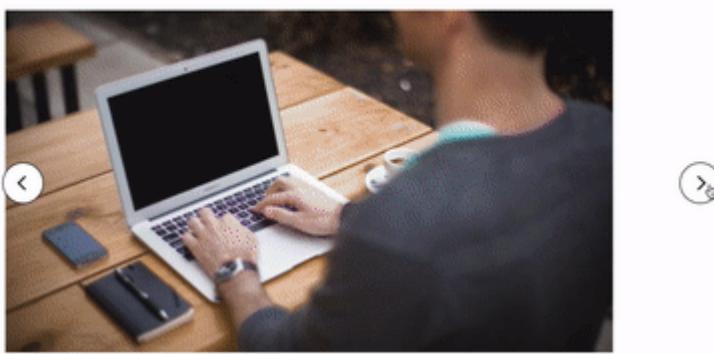
케이스 주제

Q. 캐러셀(이미지슬라이더) 구현

기능 요구사항

1. 오른쪽, 왼쪽 화살표로 슬라이더 컨트롤
2. 슬라이더 갯수는 3개 이상
3. 슬라이더가 이동하는 동안 화살표 작동안됨
4. 슬라이더 자연시간은 0.5초
5. 첫 로딩시 현재위치는 0번째 슬라이더

기능 작동 이미지



The screenshot shows the browser's developer tools with the 'Elements' tab selected. The DOM tree displays a structure for a carousel, including a wrapper and multiple items. The 'Styles' panel on the right shows the computed styles for the active item, which includes a background image and some margin settings.

```
<!DOCTYPE html>
<html lang="en">
  <head></head>
  <body> == $0
    <div class="carousel-wrapper">
      <div class="carousel">
        
        
        
        
        
      </div>
    </div>
  </body>
</html>
```

Styles Computed >
element.style {
}
body user agent stylesheet
y {
 display: block;
 margin: 8px;
}

문제

q1. javaScript - 주어진 템플릿(`q_index.html`

)으로 캐러셀 구현 q2. jQuery - Javascript로 구현한 기능을 동일하게 Jquery로도 구현해보는 문제 q3.

React - 주어진 코드를 이해하고, 상태를 조작하는 툥을 이용해서 완성

풀이 참고

1. 슬라이더 갯수가 5개라면 첫로딩시 아래 클래스 활성화

0번째에 active

1번째에 next

4번째 prev

2. css 설명

.carousel_item 의 transition 딜레이 조절

transform: translateX(-100%) x축 왼쪽이동

.carousel 의 preserve-3d 효과

주요 학습 키워드

prototype을 사용하여 메소드와 속성을 정의한 뒤, new 연산자를 사용해 함수로 호출하여 기능을 구현

각각의 슬라이드 이동은 translateX의 값으로 슬라이드의 위치를 조절

반복문을 사용하여 슬라이드 위치에 맞게 className을 지정

useState, useRef, useEffect hook 사용

작성해주셔야 하는 question 파일경로

q1

`./question/q1_js/q1_index.js`

q2

```
./question/q2_jquery/q2_index.js
```

q3

```
./question/q3_react/src/Carousel.js
```

출제자 강사님 코드 기반으로, 해당 경로에서 요구한 문제사항을 해결해주세요

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/q1_js

index.html 열기

q2

경로 ./question/q2_jquery

index.html 열기

q3

경로 ./question/q3_react

터미널

```
$ npm install  
$ npm start
```

Case18 : Pagination

케이스 주제

Q. 페이지네이션 구현

기능 요구사항

1. 주어진 댓글 데이터 22개에 대해 페이지를 나눠 댓글 요소를 뷰에 표현한다.
2. 페이지 당 표현할 수 있는 댓글 요소 수를 설정할 수 있다.
3. 현재 페이지를 알 수 있고, 다음 페이지로 넘어갈 수 있는 뷰에 기능을 구현한다.

기능 작동 이미지



abc_1

2020-05-01

UI 테스트는 어떻게 진행하나요



abc_2

2020-05-02

막히면 대답은 빨리 해주나요



abc_3

2020-05-03

코드에러가 발생했는데 어딘지 모르겠어요



abc_4

2020-05-04

Javascript 기초가 부족한데 좋은 가이드 있나요



abc_5

2020-05-05



TypeScript는 어떤점에서 좋나요

- 1
- 2
- 3
- 4
- 5

실행 방법 / 풀이 방법 안내

문제 풀기 방식 :

1. 레포지토리를 clone
2. question 디렉토리에 package.json이 없는 경우, index.html을 열거나, live-server extension으로 열기. 이외 경우 아래 안내 참고.
3. 터미널에서 각 문제 폴더 디렉토리로 이동하여 npm install로 의존성을 설치
4. package.json을 참고하여, 명시된 scripts 명령어로 개발서버 실행.
5. 코드 수정하면서 문제 해결하세요

기본 번들러로 `parcel`을 사용했습니다. - `react` 문제의 경우, `react-scripts` 사용. 문제 디렉토리에서 `npm start` 또는 `npx parcel index.html watch`로 개발서버를 실행하세요.

문제

q1. JavaScript로 해당 기능을 구현하세요. (* `index.js` 파일 확인)

페이지 번호를 동적으로 생성했을 때, 이벤트를 할당하기
페이지네이션 페이지번호를 주었을 때, 나오는 게시물수 컨트롤하기
활성화 된 버튼 처리하기

q2. jQuery 해당 기능을 구현하세요. (* `index.js` 파일 확인)

jQuery 문서로드 처리하기
jQuery로 동적으로 엘리먼트 생성시, 이벤트를 할당하는 방법

q3. React로 해당 기능을 구현하세요. (* `comment.js` 파일 확인)

각 클릭 시 react로 상태를 어떻게 처리하는지
redux toolkit을 사용

주요 학습 키워드

페이지 설정에 따른 댓글 데이터 가공 표현

작성해주셔야 하는 question 파일경로

q1

`./question/q1_js/index.js`

q2

`./question/q2_jquery/index.js`

q3

```
./question/q3_react_redux/src/containers/CommentListContainer.js
```

```
./question/q3_react_redux/src/containers/PageListContainer.js
```

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/q1_js`

index.html 열기

q2

경로 `./question/q2_jquery`

index.html 열기

q3

경로 `./question/q3_react_redux`

터미널

```
$ npm install  
$ npm start
```

Case19 : Scroll top

케이스 주제

Q. 스크롤을 내렸을 경우 상단에 고정된 내비게이션바의 배경과 폰트 색상이 변경되는 기능을 구현하세요.

기능 요구사항

- 스크롤을 내릴 경우 내비게이션 배경 색상과 폰트 색상이 변경됩니다.
- 스크롤을 다시 올릴 경우 변경된 상태를 유지하다가 더 이상 올릴 수 없을 때(최상단에 스크롤이 위치할 때) 이전 상태로 돌아가게 됩니다.
- 스크롤을 다시 올릴 경우 곧바로 배경/폰트 색상을 이전 상태로 변경하는 방식으로도 구현해봅니다.

기능 작동 이미지

<스크롤 전>



<스크롤 후>



문제

q1. javaScript - 스크롤을 다시 올릴 경우 변경된 상태를 유지하다가 더 이상 올릴 수 없을 때(최상단에 스크롤이 위치할 때) 이전 상태로 변경

1. 현재 스크롤 위치를 가져온다
2. 스크롤 위치를 바탕으로 active 클래스를 추가하거나 제거한다

q2. javaScript - 스크롤을 다시 올릴 경우 곧바로 배경/폰트 색상을 이전 상태로 변경

1. 현재 스크롤 위치를 가져온다.
2. oldvalue, 스크롤의 위치와 연산 작업을 하여 active 클래스를 추가하거나 제거한다.
3. oldvalue를 스크롤 위치로 변경한다.

q3. javaScript - 자바스크립트에서 제공하는 마우스 휠 이벤트 동작 감지 기능을 사용해서 구현

1. 마우스 휠 이벤트 동작을 감지하여 동작시킨다
2. *참고 : 자바스크립트에서는 마우스 휠 방향을 알 수 있는 mousewheel, wheel, DOMMouseScroll 이벤트를 제공합니다.

q4. jQuery - 스크롤을 다시 올릴 경우 변경된 상태를 유지하다가 더 이상 올릴 수 없을 때(최상단에 스크롤이 위치할 때) 이전 상태로 변경

1. 현재 스크롤 위치를 가져온다.
2. 스크롤 위치를 바탕으로 active 클래스를 추가하거나 제거한다.

q5. jQuery - 스크롤을 다시 올릴 경우 곧바로 배경/폰트 색상을 이전 상태로 변경

주요 학습 키워드

크로스 브라우징을 고려하여 스크롤 위치를 가져오는 방법을 학습하게 됩니다.

작성해주셔야 하는 question 파일경로

q1

`./Question/q1_js_1/main.js`

q2

`./Question/q2_js_2/main.js`

q3

`./Question/q3_js_3/main.js`

q4

`./Question/q4_jq_1/main.js`

q5

`./Question/q5_jq_2/main.js`

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/q1_js_1`

index.html 열기

q2

경로 `./question/q2_js_2`

index.html 열기

q3

경로 `./question/q3_js_3`

index.html 열기

q4

경로 `./question/q4_jq_1`

index.html 열기

q5

경로 `./question/q5_jq_2`

index.html 열기

Case20 : List

케이스 주제

Q. 서버 통신으로 가져온 Post 정보를 화면에 리스트 방식으로 적용하세요.

기능 요구사항

GET 방식으로 Post 정보를 가져옵니다.
가져온 정보를 순차적으로 화면에 적용시킵니다.

기능 작동 이미지

<적용 전>

(빈 화면)

<적용 후>

sunt aut facere repellat provident occaecati excepturi optio reprehenderit
quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto

qui est esse
est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui neque nisi nulla

ea molestias quasi exercitationem repellat qui ipsa sit aut
et iusto sed quo iure voluptatem occaecati omnis eligendi aut ad voluptatem doloribus vel accusantium quis pariatur molestiae porro eius odio et labore et velit aut

eum et est occaecati
ullam et saepe reiciendis voluptatem adipisci sit amet autem assumenda provident rerum culpa quis hic commodi nesciunt rem tenetur doloremque ipsam iure quis sunt voluptatem rerum illo velit

문제

q1. javaScript

- ```
1. fetch 함수를 사용하여 해당 기능을 구현하세요.
2. Fetch Refactoring : 콜백함수를 사용하여 통신 영역, 템플릿 영역을 분할하여 작성하시오
```

## q2. javaScript

- ```
1. axios 라이브러리를 사용하여 해당 기능을 구현하세요.  
2. Axios Refactoring
```

q3. javaScript - async/await와 콜백 함수를 사용하여 해당 기능을 구현하세요.

q4. jQuery로 해당 기능을 구현하세요.

주요 학습 키워드

```
Fetch, Axios, Ajax 기본 사용법과 함께 async / await를 사용한 비동기 방식과 함께 역할별로 코드를 리팩토링 하는 방법을  
익히게 됩니다.
```

작성해주셔야 하는 question 파일경로

q1

```
./question/1. js-fetch/main.js
```

q2

```
./question/2. js-axios/main.js
```

q3

```
./question/3. js-async-await/main.js
```

q4

```
./question/4. jq/main.js
```

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/1. js-fetch

index.html 열기

q2

경로 ./question/2. js-axios

index.html 열기

q3

경로 ./question/3. js-async-await

index.html 열기

q4

경로 ./question/4. jq

index.html 열기

Case 21 : Modal Window 2

케이스 주제

Q. 이미지를 클릭하면 그 이미지 정보를 담고 있는 모달창이 등장하고, 레이어를 클릭하면 모달창이 사라지는 기능을 만드세요.

기능 요구사항

Unsplash의 List photo API를 사용해서 이미지 정보를 리스트로 화면에 띄우고,
특정 이미지를 클릭했을 때, 그 이미지에 대한 설명이 기재된 모달창을 띄우세요.
이미지에 대한 설명은 해당 이미지의 데이터 속성 값으로 리스트 안에 미리 넣어두는 형태로 처리하시오.
모달창을 제외한 외부 여백 공간 클릭 시 모달창이 사라지도록 하세요.

기능 작동 이미지



문제

JavaScript로 해당 기능을 구현하시오.

주요 학습 키워드

객체 형태로 데이터 속성값이 정보를 저장하여 이를 가져오는 방법을 학습합니다.
레이어 팝업을 처리하는 방법을 익히게 됩니다.

작성해주셔야 하는 question 파일경로

q1

```
./question/1.js/main.js
```

실행 방법

q1

경로 `./question/1.js`

index.html 실행

Case 22 : Scroll Indicator

케이스 주제

Q. 요구사항 : 스크롤 시 현재 남의 컨텐츠의 분량을 화면에 표기해 주세요. (e.g
<https://sports.v.daum.net/v/20210123112225915>)

기능 요구사항

스크롤을 내리면 상단에 현재 스크롤이 어느 정도 내려갔는지를 나타내는 상태 표시바를 만드시오.
스크롤이 끝까지 내려가면 Indicator도 끝까지 이동하고, 스크롤을 다시 상단으로 올리면 Indicator도 다시 뒤로 돌아가게 만드시오.

기능 작동 이미지

Hello World

It was seven o'clock of a very warm evening in the Seonee hills when Father Wolf woke up from his day's rest, scratched himself, yawned, and spread out his paws one after the other to get rid of the sleepy feeling in their tips. Mother Wolf lay with her big gray nose dropped across her four tumbling, squealing cubs, and the moon shone into the mouth of the cave where they all lived. "Augh!" said Father Wolf. "It is time to hunt again." He was going to spring down hill when a little shadow with a bushy tail crossed the threshold and whined: "Good luck go with you, O Chief of the Wolves. And good luck and strong white teeth go with noble children that they may never forget the hungry in this world."

It was the jackal--Tabaqui, the Dish-licker--and the wolves of India despise Tabaqui because he runs about making mischief, and telling tales, and eating rags and pieces of leather from the village rubbish-heaps. But they are afraid of him too, because Tabaqui, more than anyone else in the jungle, is apt to go mad, and then he forgets that he was ever afraid of anyone, and runs through the forest biting everything in his way. Even the tiger runs and hides when little Tabaqui goes mad, for madness is the most disgraceful thing that can overtake a wild creature. We call it hydrophobia, but they call it devanee--the madness--and run.

"Enter, then, and look," said Father Wolf stiffly, "but there is no food here."

"For a wolf, no," said Tabaqui, "but for so mean a person as myself a dry bone is a good feast. Who are we, the Gidur-log [the jackal people], to pick and choose?" He scuttled to the back of the cave, where he found the bone of a buck with some meat on it, and sat cracking the end merrily.

"All thanks for this good meal," he said, licking his lips. "How beautiful are the noble children! How large are their eyes! And so young too! Indeed, indeed, I might have remembered that the children of kings are men from the beginning."

Now, Tabaqui knew as well as anyone else that there is nothing so unlucky as to compliment children to their faces. It pleased him to see Mother and

문제

JavaScript로 해당 기능을 구현하시오.

- scrollBar의 width값을 변경하는 방식으로 indicator를 적용하시오.
- 스크롤 시 translateX를 사용하여 scrollBar 위치를 변경하시오.

주요 학습 키워드

q1. 크로스 브라우징을 고려하여 현재 문서의 높이를 가져오기
q2. width 또는 translateX를 사용하여 남은 컨텐츠를 표기하는 방법 학습하기

작성해주셔야 하는 question 파일경로

q1

```
./question/1.js-width/main.js
```

q2

```
./question/2.js-translateX/main.js
```

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/1.js-width

index.html 실행

q2

경로 ./question/2.js-translateX

index.html 실행

Case 23 : Loading

케이스 주제

Q. 요구사항 : 회원가입, 로그인, 게시물 작성 등 서버에 데이터를 보낼 경우 전송중인 상태를 화면에 표기합니다.

기능 요구사항

서버에 데이터를 보내는 경우, 데이터를 전송중인 상태를 나타내기 위해 화면에 스피너 UI를 띄우세요.
통신이 완료되었을 시, 완료된 상태를 나타내는 Alert 창을 띄우세요.

기능 작동 이미지

Title

234

Body

234

User Id

Submit

문제

JavaScript로 해당 기능을 구현하시오.

주요 학습 키워드

각 역할별로 콜백함수를 생성하여 코드를 관리하는 방법을 익히고, 통신 성공 / 실패를 효율적으로 관리하는 방법을 익하게 됩니다.

작성해주셔야 하는 question 파일경로

q1

`./question/1.js/main.js`

실행 방법 및 의존성 모듈 설치

q1

경로 `./question/1.js`

index.html 열기

Case24 : Search bar

기능 요구사항

Q. 검색 리스트를 드랍다운 메뉴로 표기하고, 메뉴를 선택하면 선택한 이미지를 화면에 표기합니다.

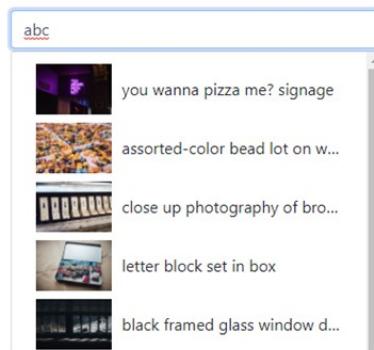
1. 이미지 API : <https://unsplash.com/documentation#search-photos>
2. debounce : <https://lodash.com/docs/4.17.15#debounce>

기능 작동 이미지

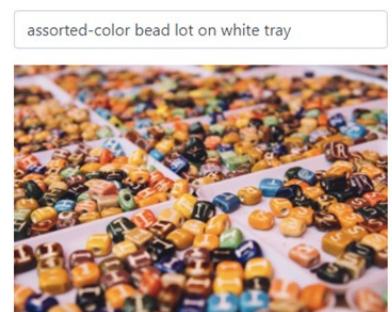
1. 검색 전



2. 검색어 입력 시



3. 특정 검색어 선택 시



문제

q1. 문제 상황에 대하여 JavaScript로 동작을 구현시킬 수 있는 코드를 작성해보세요

1. 검색 이미지 리스트 가져오기
2. 하나의 이미지 정보만 가져오기
3. 검색 드랍다운 표기
4. 검색 드랍다운 리스트 표기
5. 선택한 이미지 정보 표기

q2. 문제 상황에 대하여 jQuery로 동작을 구현시킬 수 있는 코드를 작성해보세요

1. JavaScript로 짠 코드를 jQuery로 전환
2. Data 속성 활용

주요 학습 키워드

검색어 입력에 따른 API 호출을 제어하는 방법을 학습하게 됩니다.

axios 라이브러리와 async, await를 이용한 API호출

lodash 라이브러리의 debounce()를 사용하여 불필요한 호출 줄이기

반복문을 통해 createElement, innerHTML, appendChild를 사용하여 리스트 표기

jQuery의 data- 속성 활용

작성해주셔야 하는 question 파일경로

q1

./question/1.js/main.js

q2

./question/2.jq/main.js

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/1.js

index.html 열기

q2

경로 ./question/2.jq

터미널

```
$ npm install  
$ npm start
```

Case25 : Combo Box

케이스 주제

드롭다운 리스트는 단순히 클릭하면 리스트가 나오는 형태뿐 아니라, input 태그로 값을 입력하고 input 우측의 버튼을 눌러 값 목록을 드롭다운 리스트로 볼 수 있는 형태로도 만들 수 있습니다. 이를 콤보 박스라고 합니다.

Q. 아래와 같은 스펙을 가진 콤보 박스를 만들어보세요.

기능 요구사항

하나의 input tag를 만든다.

이 input에 타이핑을 해서 값을 입력할 수 있다.

타이핑을 할때 엔터키를 치면 입력한 값을 드롭다운 리스트에 추가한다.

드롭다운 리스트는 안보이다가 input tag 우측의 버튼을 누르면 나온다.

(옵션) 엔터키를 쳐서 값을 드롭다운 리스트에 추가할때 입력이 되었다는 표시를 하는 notification을 만든다.

드롭다운 리스트의 가로 크기는 input + button의 가로이고 세로 크기는 임의의 숫자로 고정이다.

이때 드롭다운 리스트에 height 크기를 넘어갈 정도로 값이 차면 스크롤을 할 수 있도록 한다.

기능 작동 이미지

문제

q1. Javascript - 각각의 이벤트 리스트에 코드를 작성하여 위 기능이 동작되는 콤보 박스를 완성하시오.

주요 학습 키워드

CSS animation을 사용하여 항목이 추가 되었을 때, classList를 add하여 알림메세지 보여주기(CustomEvent 사용)
keyup 이벤트를 사용하여 입력한 항목을 배열에 추가 후, innerHTML을 사용하여 DOM에 동적으로 추가하기
setTimeout을 사용하여 일정시간 후 알림메세지를 자동으로 사라지게 하여 UX개선

작성해주셔야 하는 question 파일경로

q1

`./question/index.js`

실행 방법

q1

경로 `./question`
`index.html` 열기

Case26 : Radio Box

케이스 주제

라디오 박스는 여러 값들이 주어지고 그 중에서 단 하나를 선택할 수 있는 UI입니다.

기본적으로 type이 radio인 input tag를 사용해 만들며, 브라우저가 type이 radio인 input들 중 name 속성이 같은 것들은 하나만 선택되도록 강제하고 그에 따라 렌더링 합니다.

이 라디오 박스는 이렇게 input tag를 사용해도 되지만 다른 tag를 사용해 커스텀하게 작성할 수도 있습니다. 다만 이 경우엔 하나만 선택되고 그에 따라 렌더링이 될 수 있도록 추가적인 작업을 JavaScript로 해주어야 합니다.

Q. 아래와 같은 스펙을 가진 라디오 박스를 만들어보세요.

기능 요구사항

하나의 form tag를 만든다.

이 form tag 안에 연락 수단과 배송 수단을 선택할 수 있는 라디오 박스 UI를 최소 두개를 만든다.

form을 제출하면(type이 submit인 버튼을 누르면) 모든 수단이 선택되었는지 검사한다.

항목 하나라도 선택하지 않았으면 모든 항목에 대한 선택을 해야한다는 정보성 vUI를 만들어 보여준다.

검사를 무사히 통과했으면 현재 form이 최종적으로 어떤 데이터를 보내려는지 form UI 하단에 보여준다.

기능 작동 이미지

연락 수단을 선택해주세요!

Email Phone Email

배송 방법을 선택해주세요!

Ship Plane Car

확인

문제

q1. Javascript로 위 기능을 구현하시오.

주요 학습 키워드

submit event가 발생하면 formData를 이용해 value를 가져오고 배열로 만들어주기
Object.keys를 사용하여 폼을 모두 작성했는지 체크하기
입력한 정보를 innerHTML을 사용하여 DOM에 동적으로 추가하기

작성해주셔야 하는 question 파일경로

q1

`./question/index.js`

실행 방법

q1

경로 `./question/index.html` 열기

Case27 : Check box

케이스 주제

Q. 아래와 같이 작동하는 체크 박스를 구현하시오.

체크 박스는 여러 값들이 주어지고 그 중에서 여러 값을 선택할 수 있는 UI입니다. 기본적으로 type이 checkbox인 input tag를 사용해 만들며 이 체크 박스는 이렇게 input tag를 사용해도 되지만 다른 tag를 사용해 커스텀하게 작성할 수도 있습니다. 여러개를 선택할 수 있고 그에 따라 여러 체크 박스들이 제대로 렌더링이 될 수 있도록 추가적인 작업을 JavaScript로 해주어야 합니다.

기능 요구사항

요구 사항은 아래와 같다.

1. 하나의 form tag를 만든다.
2. 이 form tag 안에 좋아하는 음식과 관심사를 여러개 선택할 수 있는 체크 박스 UI를 최소 두개를 만든다.
3. form을 제출하면(type이 submit인 버튼을 누르면) 좋아하는 음식과 관심사가 각각 최소 하나씩 선택되었는지 검사한다. 최소 하나씩 선택하지 않았으면 선택을 해야한다는 정보성 UI를 만들어 보여준다.
4. 검사를 무사히 통과했으면 현재 form이 최종적으로 어떤 데이터를 보내려는지 form UI 하단에 보여준다.

첫 화면

좋아하는 음식

Pizza Steak Noodle

관심사

Cooking Bike Swimming Piano Book

아무것도 선택하지 않고 submit 했을때

좋아하는 음식

Pizza Steak Noodle

관심사

Cooking Bike Swimming Piano Book

확인

폼을 모두 작성해주세요.

항목 중 하나만 선택했을때

좋아하는 음식

Pizza Steak Noodle

관심사

Cooking Bike Swimming Piano Book

확인

폼을 모두 작성해주세요.

각 항목당 최소 하나 이상 선택했을때

좋아하는 음식

Pizza Steak Noodle

관심사

Cooking Bike Swimming Piano Book

확인

좋아하는 음식 : steak
관심사 : Bike, swimming

주요 학습 키워드

```
submit event가 발생하면 formData를 이용해 value를 가져오고 배열로 만들어주기  
filter 함수를 사용하여 length 체크하기  
입력한 정보를 innerHTML을 사용하여 DOM에 동적으로 추가하기
```

작성해주세요 하는 question 파일경로

./question/app.js

실행 방법 및 의존성 모듈 설치

경로 ./question

터미널

```
$ yarn install  
$ yarn start
```

Case28 : Instant search

케이스 주제

Q. 아래와 같이 작동하는 즉시 검색 기능을 구현하시오.

기능 요구사항

요구 사항은 아래와 같다.

1. 하나의 input tag를 만든다.
2. 이 input에 키보드 타이핑이 될때 현재 검색어 기준으로 필요한 API를 호출한다. 이때 API 호출은 즉각 실행해서 결과를 보여준다.
3. 검색어 API가 진행 중일때 input tag 우측에 loading 중임을 표시한다.
4. 검색어 API Response가 도착하면 그 내용을 input tag 아래에 리스트로 보여준다.

첫화면



검색어를 입력하고 로딩 중일때



검색어 API의 Response가 도착했을때

a

Abyssinian
Aegean
American Bobtail
American Curl
Polydactyl

해당하는 검색어가 없을때

abcd|

해당하는 검색어가 없습니다..!

주요 학습 키워드

fetch와 async, await를 이용한 API호출
slice, map 함수를 사용하여 특정조건에 맞는 검색어를 특정갯수만큼 노출 시키는 기능 구현
Element 속성 innerHTML, style을 사용하여 DOM에 보여주기

작성해주셔야 하는 question 파일경로

./question/src/app.js

실행 방법 및 의존성 모듈 설치

경로 ./question

터미널

```
$ yarn install  
$ yarn start
```

Case29 : Auto Complete 1

케이스 주제

검색어 자동완성 기능 만들기

(현업에서 서비스를 만들때 화장품을 검색하거나 영상을 검색할때 등에 사용할 검색 자동완성 기능을 요구하는 경우가 많이 있습니다. 사용자는 이 기능으로 내가 검색한 내용과 관련된 다른 검색어들을 알 수 있게 되어 사용자에게 정보 편의성이 주어지고 서비스 입장에서는 사용자로부터 더 많은 검색을 이끌어낼 수 있습니다.)
Q. 아래와 같은 스펙을 가진 검색 자동완성 기능을 만들어보세요.

기능 요구사항

1. 하나의 input tag를 만든다.
2. 이 input에 키보드 타이핑이 될때 현재 검색어 기준으로 관련 검색어 API를 호출한다.

이때 API 호출은 즉각 실행하길 바라는 경우
타이핑을 멈추고 0.5초 등의 시간이 지나고 요청하는 경우가 있다.

3. 검색어 API가 진행 중일때 input tag 우측에 loading 중임을 표시한다.
4. 검색어 API Response가 도착하면 그 내용을 input tag 아래에 리스트로 보여준다.

기능 작동 이미지



Loading...

문제

q1. Javascript

가장 마지막 타이핑이 일어나고 0.5초 뒤에 API Request를 실행하도록 하는 debounce logic을 작성하시오.

q2. RxJS를 이용해 스트림 구조로 동일한 기능을 작성하시오

주요 학습 키워드

fetch와 async, await를 이용한 API호출

지연시간을 적용하여 마지막 이벤트만 발생시키는 debounce를 이용해, 마지막으로 타이핑을 한 순간에 API를 호출 할 수 있는 기능 구현

slice, map 함수를 사용하여 특정조건에 맞는 검색어를 특정갯수만큼 노출 시키는 기능 구현

작성해주셔야 하는 question 파일경로

q1

`./question/1. Vanilla JavaScript/src/app.js`

q2

`./question/2. RxJS/src/app.js`

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/1. Vanilla JavaScript

터미널

```
$ npm install  
$ npm start
```

q2

경로 ./question/2. RxJS

터미널

```
$ npm install  
$ npm start
```

./solution/2.other

풀이 실행 시

q1

경로 ./solution/2.other/1. Vanilla JavaScript

index.html 열기

q2

경로 ./solution/2.other/2. RxJS

터미널

```
$ npm install  
$ npm run dev
```

Case30 : Rating UI

케이스 주제

별점을 평가할 수 있는 Rating UI를 구현하세요.

Q. 아래와 같은 스펙을 가진 Rating UI를 구현하세요.

기능 요구사항

1. 0 ~ N 개의 별을 표시합니다. (0.5점 단위)
2. 각 별은 비어있거나(Empty), 반만 차있거나(Half), 꽉 차있을 수 있습니다(Full). (3가지 각 이미지 파일은 assets 폴더 안에 있습니다.)
3. 별 위로 마우스오버하면 왼쪽부터 마우스 위치까지의 채워진 별로 표시합니다.
4. 클릭시 점수 상태를 업데이트합니다.
5. X 버튼을 클릭하면 별점을 0점으로 초기화합니다.

별의 갯수가 N값만 조정하면 쉽게 변경되도록 작성해보세요.

기능 작동 이미지



문제

q1. N점 만점의 Rating UI를 JavaScript(ES6+)로 구현해보세요.

q2. N점 만점의 Rating UI를 React Component로 구현해보세요.

주요 학습 키워드

DOM Access, DOM Events(mousemove, mouseleave, click, etc.), MouseEvent, Mouse Position Calculation, useState, State Management

작성해주셔야 하는 question 파일경로

q1

./question/q1/index.js

q2

./question/q2/src/components/StarRate.jsx

출제자 강사님 코드 기반으로, 해당 경로에서 요구한 문제사항을 해결해주세요

실행 방법 및 의존성 모듈 설치

q1

경로 ./question/q1

터미널

```
$ npm install  
$ npm start
```

빌드가 완료되면 localhost:3001로 접속합니다.

q2

경로 ./question/q2

터미널

```
$ npm i  
$ npm run start
```

빌드가 완료되면 localhost:3002로 접속합니다.