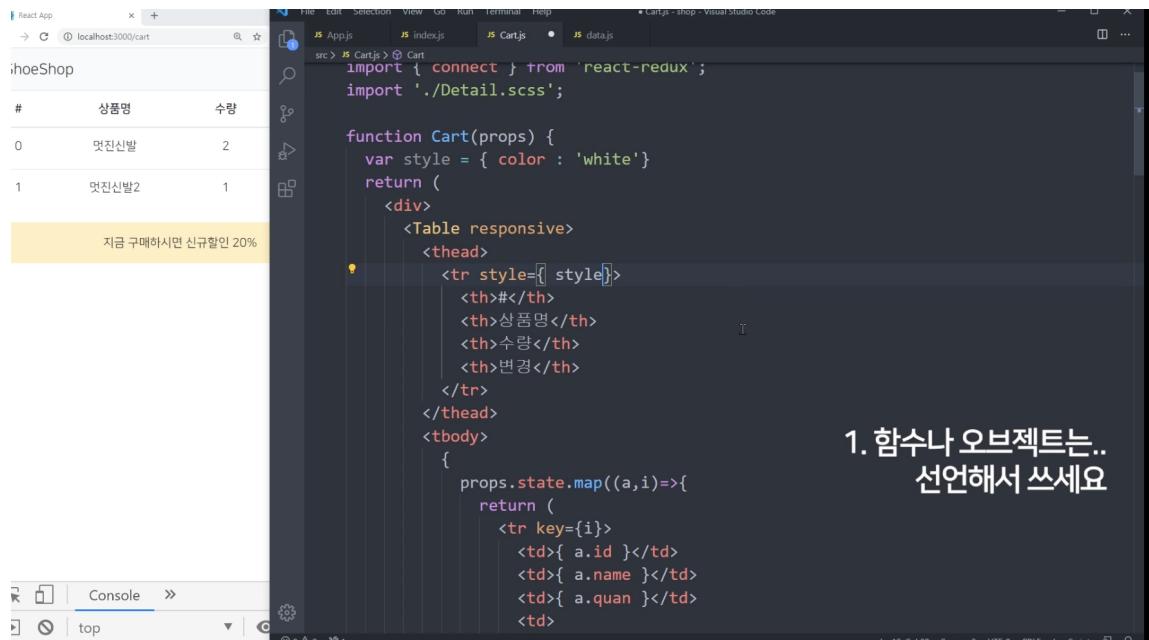


9월 14일 (화)

1. 코딩애플 리액트 강의 Part 3

- 함수나 오브젝트는 선언해서 쓰자.



```
import { connect } from 'react-redux';
import './Detail.scss';

function Cart(props) {
  var style = { color: 'white' }
  return (
    <div>
      <Table responsive>
        <thead>
          <tr style={style}>
            <th>#</th>
            <th>상품명</th>
            <th>수량</th>
            <th>변경</th>
          </tr>
        </thead>
        <tbody>
          {
            props.state.map((a,i)=>{
              return (
                <tr key={i}>
                  <td>{ a.id }</td>
                  <td>{ a.name }</td>
                  <td>{ a.quan }</td>
                  <td>
                    <button>-
                    <button>+</button>
                  </td>
                </tr>
              )
            })
          }
        </tbody>
      </Table>
    </div>
  );
}

const mapStateToProps = state => {
  return {
    state: state.cart
  }
}

const mapDispatchToProps = dispatch => {
  return {
    increment(id) {
      dispatch({ type: 'INCREMENT', id })
    },
    decrement(id) {
      dispatch({ type: 'DECREMENT', id })
    }
  }
}

export default connect(mapStateToProps, mapDispatchToProps)(Cart);
```

1. 함수나 오브젝트는..
선언해서 쓰세요

메모리 할당때문에 성능적으로 문제가 생길 수 있으므로 선언해서 사용하자.

- 레이아웃 변경을 주는 애니메이션을 줄 때는 transform 을 사용하자.

React App → C localhost:3000/cart

ihoeShop

| # | 상품명 | 수량 |
|---|-------|----|
| 0 | 멋진신발 | 2 |
| 1 | 멋진신발2 | 1 |

지금 구매하시면 신규할인 20%

File Edit Selection View Go Run Terminal Help Cart.js - Visual Studio Code

src > Cart.js

```

import { connect } from 'react-redux';
import './Detail.scss';

function Cart(props) {
  return (
    <div>
      <Table responsive>
        <thead>
          <tr>
            <th>#</th>
            <th>상품명</th>
            <th>수량</th>
            <th>변경</th>
          </tr>
        </thead>
        <tbody>
          {
            props.state.map((a,i)=>{
              return (
                <tr key={i}>
                  <td>{ a.id }</td>
                  <td>{ a.name }</td>
                  <td>{ a.quan }</td>
                  <td>
                    <button onClick={()=>{ props.dispatch({ type : '수량증가', id : a.id }) }}>증가</button>
                    <button onClick={()=>{ props.dispatch({ type : '수량감소', id : a.id }) }}>감소</button>
                  </td>
                </tr>
              )
            )
          }
        </tbody>
      </Table>
    </div>
  );
}

export default Cart;

```

Line 6, Col 23 Spaces: 2 UTF-8 CRLF JavaScript

2. 애니메이션 막주지말고..
되도록이면 transform 그그

- 컴포넌트 import 할 때 lazy loading 하자.

React App → C localhost:3000/cart

ihoeShop

| # | 상품명 | 수량 |
|---|-------|----|
| 0 | 멋진신발 | 2 |
| 1 | 멋진신발2 | 1 |

지금 구매하시면 신규할인 20%

File Edit Selection View Go Run Terminal Help App.js - Visual Studio Code

src > App.js

```

/* eslint-disable */
import React, {useState, useContext} from 'react';
import { Navbar, Nav, NavDropdown, Button, Jumbotron } from 'react-bootstrap';
import './App.css';
import Data from './data.js';
import Detail from './Detail.js';
import axios from 'axios';
import { Link, Route, Switch, useHistory } from 'react-router-dom';

import { connect } from 'react-redux';
import Cart from './Cart.js';

export let 재고context = React.createContext();

function App() {
  let [shoes, shoes변경] = useState(Data);
  let [재고, 재고변경] = useState([10,11,12]);
  let [cart, cart변경] = useState([{ id : 0, name : '이쁜신발', quan : 1 }]);
}


```

Line 7, Col 1 (33 selected) Spaces: 2 UTF-8 LF JavaScript

App.js 방문시
Detail, Cart 다 미리 로드함
= 부담일수도

3. 컴포넌트 import 할 때

- 재렌더링을 막아주는 memo() 를 사용하자.

memo()의 단점
= 기존 props vs 바뀐 props 비교연산 후
컴포넌트 업데이트할지 말지 결정함

```

App.js          index.js      Cart.js      data.js
src > Cart.js > Parent
    j
        <Parent 이름="준박" 나이="20"></Parent>
    </div>
)
}
}

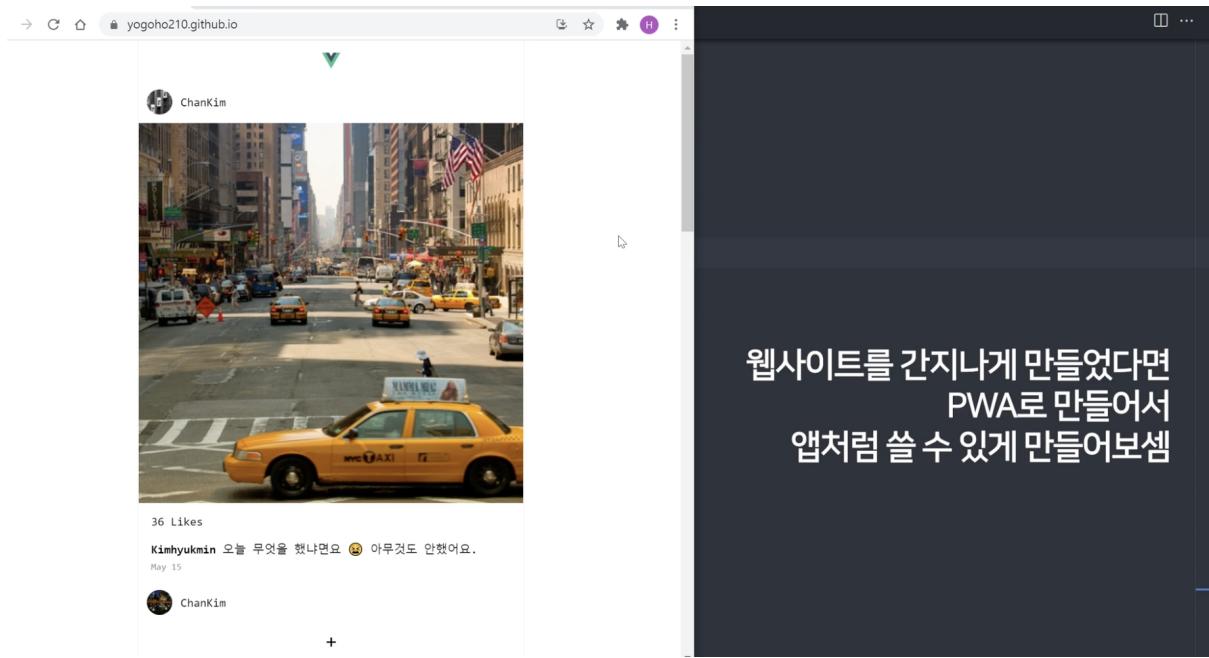
function Parent(props){
    return (
        <div>
            <Child1 이름={props.이름}></Child1>
            <Child2 나이={props.나이}></Child2>
        </div>
    )
}

function Child1(){
    useEffect(()=>{ console.log('렌더링됨1') });
    return <div>1111</div>
}

let Child2 = memo(function(){
    useEffect(()=>{ console.log('렌더링됨2') });
    return <div>2222</div>
));

```

- PWA 설정해서 앱으로 발행하기



public 폴더 안에 미리 만들어져 있다.

manifest.json

```

{
  "short_name": "React App sdfsf",
  "name": "Create React App Sample",
  "icons": [
    {
      "src": "favicon.ico",
      "sizes": "64x64 32x32 24x24 16x16",
      "type": "image/x-icon"
    },
    {
      "src": "logo192.png",
      "type": "image/png",
      "sizes": "192x192"
    },
    {
      "src": "logo512.png",
      "type": "image/png",
      "sizes": "512x512"
    }
  ],
  ...
}
  
```

manifest.json 앱 설정파일



앱이름

name 은 어플의 풀네임, short_name 은 어플상에 보이는 네임

manifest.json

```

{
  "icons": [
    {
      "src": "favicon.ico",
      "sizes": "64x64 32x32 24x24 16x16",
      "type": "image/x-icon"
    },
    {
      "src": "logo192.png",
      "type": "image/png",
      "sizes": "192x192"
    },
    {
      "src": "logo512.png",
      "type": "image/png",
      "sizes": "512x512"
    }
  ],
  "start_url": ".",
  "display": "standalone",
  "theme_color": "#000000",
  ...
}
  
```

manifest.json 앱 설정파일

Android, iOS, Windows 마다
요구하는 아이콘 크기가 다름

여러 이미지를 다양하게 집어 넣어줘야 한다.

manifest.json

```
{  
    "icons": [  
        {"src": "logo192.png",  
         "type": "image/png",  
         "sizes": "192x192"},  
        {"src": "logo512.png",  
         "type": "image/png",  
         "sizes": "512x512"}],  
    "start_url": "./",  
    "display": "standalone",  
    "theme_color": "#000000",  
    "background_color": "#ffffff"  
}
```

앱 누르면 처음 뜨는 페이지 경로

manifest.json

```
{  
    "icons": [  
        {"src": "logo192.png",  
         "type": "image/png",  
         "sizes": "192x192"},  
        {"src": "logo512.png",  
         "type": "image/png",  
         "sizes": "512x512"}],  
    "start_url": ".",  
    "display": "standalone",  
    "theme_color": "#000000",  
    "background_color": "#ffffff"  
}
```

앱켜면 브라우저 상단바 제거할지 말지

manifest.json

```

{
  "name": "PWA WEBSITE",
  "short_name": "PWA",
  "start_url": ".",
  "display": "standalone",
  "background_color": "#ffffff",
  "theme_color": "#000000",
  "icons": [
    {
      "src": "logo192.png",
      "type": "image/png",
      "sizes": "192x192"
    },
    {
      "src": "logo512.png",
      "type": "image/png",
      "sizes": "512x512"
    }
  ]
}

```

나머지 설정들은 강의 밑에 마련

service-worker.js 필요

```

import React from 'react';
import ReactDOM from 'react-dom';
import './index.css';
import App from './App';
import * as serviceWorkerRegistration from './serviceWorkerRegistration';
import reportWebVitals from './reportWebVitals';

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById('root')
);

// If you want your app to work offline and load faster, you can change
// unregister() to register() below. Note this comes with some pitfalls.
// Learn more about service workers: https://cra.link/PWA
serviceWorkerRegistration.unregister();

// If you want to start measuring performance in your app, pass a function

```

```

import App from './App';
import * as serviceWorkerRegistration from './service-worker';
import reportWebVitals from './reportWebVitals';

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById('root')
);

// If you want your app to work offline and load faster, you can change
// unregister() to register() below. Note this comes with some pitfalls.
// Learn more about service workers: https://cra.link/PWA
serviceWorkerRegistration.register();

// If you want to start measuring performance in your app, pass a function
// to log results (for example: reportWebVitals(console.log))
// or send to an analytics endpoint. Learn more: https://bit.ly/CRA-vitals
reportWebVitals();

```

만들려면 index.js에서 이렇게 바꿔셈

unregister 을 register 로 바꿔야 한다.

설정 후에 빌드하면 된다.

```

import App from './App';
import * as serviceWorkerRegistration from './service-worker';
import reportWebVitals from './reportWebVitals';

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById('root')
);

The project was built assuming it is hosted at /.
You can control this with the homepage field in your package.json.

The build folder is ready to be deployed.
You may serve it with a static server:

yarn global add serve
serve -s build

Find out more about deployment here:

https://cra.link/deployment

```

이제 npm run build 하면 ;
service-worker.js 생김

PS C:\Users\harry\Desktop\React class\my-app>

service-worker.js

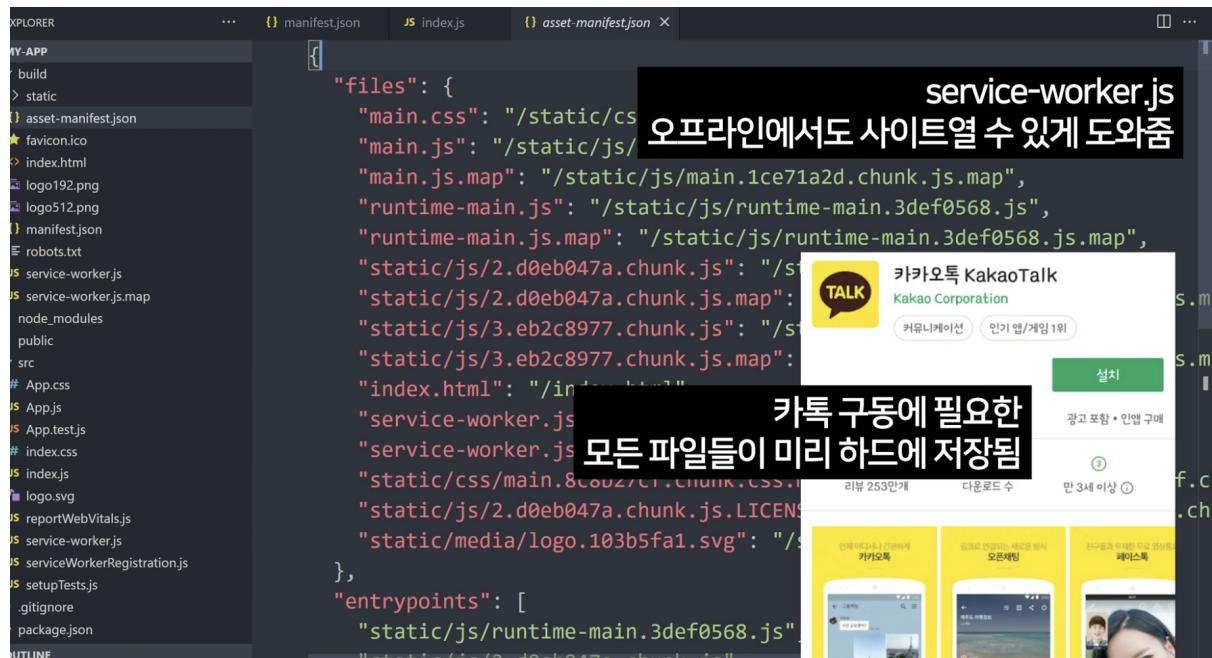
오프라인에서도 사이트 열 수 있게 도와줌

```

{
  "files": {
    "main.css": "/static/css/main.css",
    "main.js": "/static/js/main.js",
    "main.js.map": "/static/js/main.1ce71a2d.chunk.js.map",
    "runtime-main.js": "/static/js/runtime-main.3def0568.js",
    "runtime-main.js.map": "/static/js/runtime-main.3def0568.js.map",
    "static/js/2.d0eb047a.chunk.js": "/static/js/2.d0eb047a.chunk.js",
    "static/js/2.d0eb047a.chunk.js.map": "/static/js/2.d0eb047a.chunk.js.map",
    "static/js/3.eb2c8977.chunk.js": "/static/js/3.eb2c8977.chunk.js",
    "static/js/3.eb2c8977.chunk.js.map": "/static/js/3.eb2c8977.chunk.js.map",
    "index.html": "/index.html",
    "service-worker.js": "/service-worker.js",
    "service-worker.js.map": "/service-worker.js.map",
    "static/css/main.8c8b27cf.chunk.css.map": "/static/css/main.8c8b27cf.css.map",
    "static/js/2.d0eb047a.chunk.js.LICENSE.txt": "/static/js/2.d0eb047a.chunk.js.LICENSE.txt",
    "static/media/logo.103b5fa1.svg": "/static/media/logo.103b5fa1.svg"
  },
  "entrypoints": [
    "static/js/runtime-main.3def0568.js"
  ]
}

```

비행기 모드에서 카카오톡 실행되긴한다.



하지만 웹은 비행기에서 실행할 수 없다. 이를 가능하게 해주는 것이 PWA이다.

service-worker.js

"나는 html css js 파일을 미리 하드에 저장해둘 것임"
"사이트 접속할 때 html css js 다운받지말고 하드에 있던거 쓰셈"

```

"runtime-main.js": "/static/js/runtime-main.3def0568.js",
"runtime-main.js.map": "/static/js/runtime-main.3def0568.js.map",
"static/js/2.d0eb047a.chunk.js": "/static/js/2.d0eb047a.chunk.js",
"static/js/2.d0eb047a.chunk.js.map": "/static/js/2.d0eb047a.chunk.js.m
"static/js/3.eb2c8977.chunk.js": "/static/js/3.eb2c8977.chunk.js",
"static/js/3.eb2c8977.chunk.js.map": "/static/js/3.eb2c8977.chunk.js.m
"index.html": "/index.html",
"service-worker.js": "/service-worker.js",
"service-worker.js.map": "/service-worker.js.map",
"static/css/main.8c8b27cf.chunk.css.map": "/static/css/main.8c8b27cf.c
"static/js/2.d0eb047a.chunk.js.LICENSE.txt": "/static/js/2.d0eb047a.ch
"static/media/logo.103b5fa1.svg": "/static/media/logo.103b5fa1.svg"
},
"entrypoints": [
  "static/js/runtime-main.3def0568.js",
  "static/js/runtime-main.3def0568.js.map"
]

```

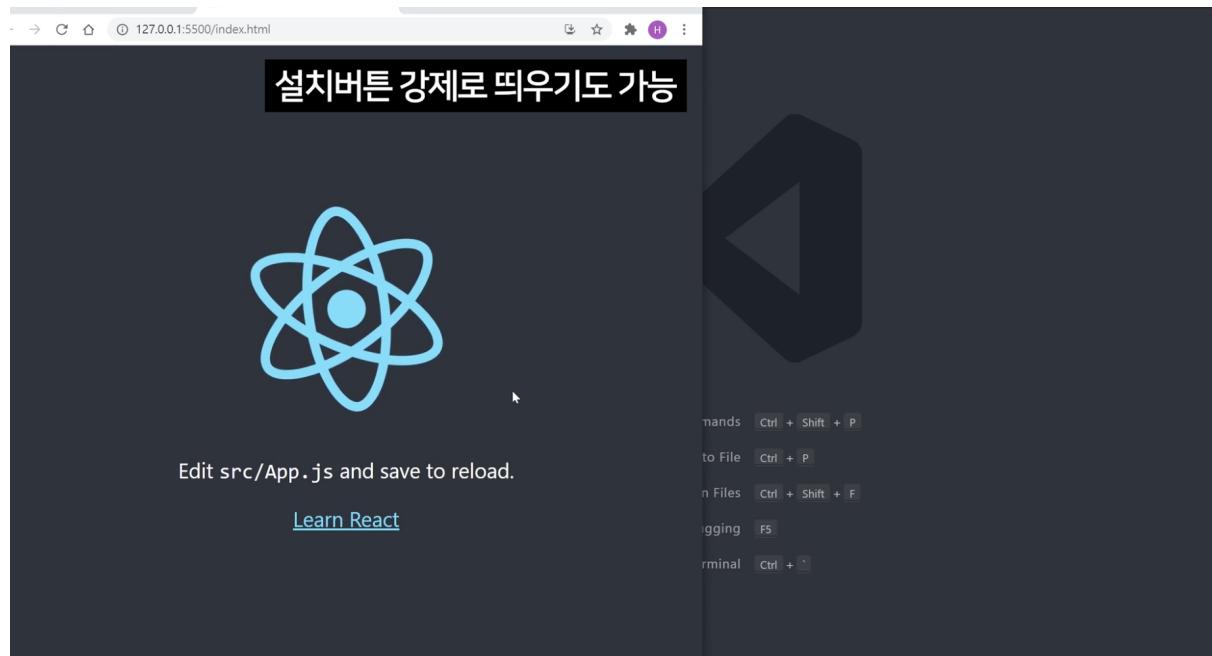
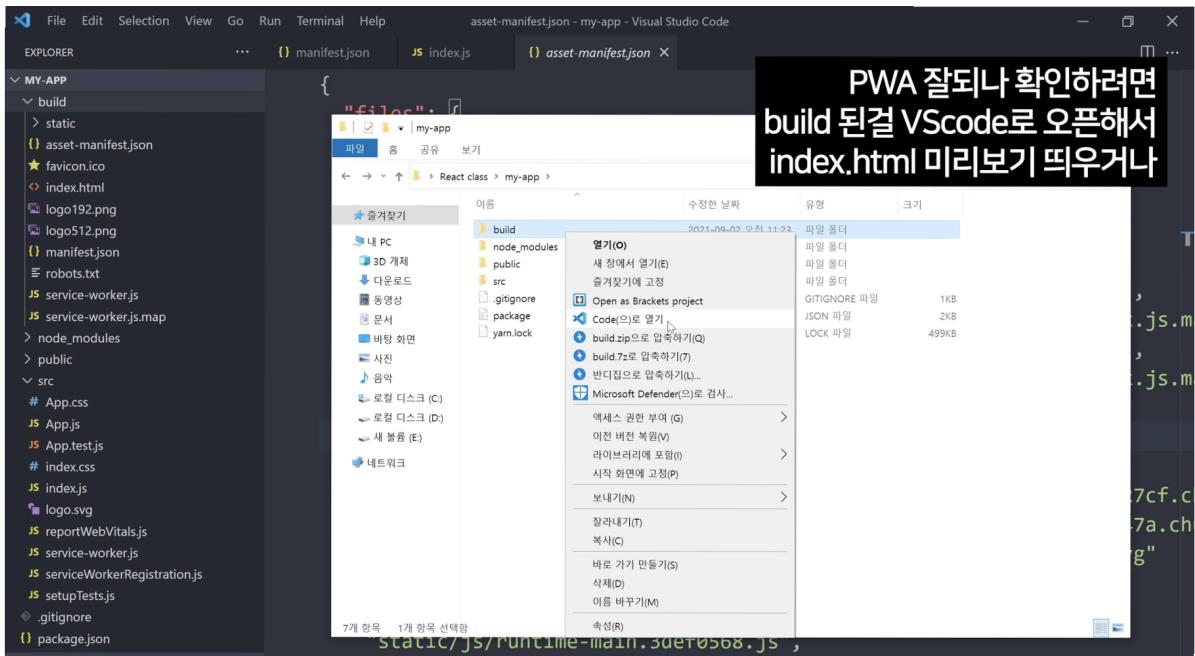
캐싱할 파일 목록

```

"files": {
  "main.css": "/static/css/main.8c8b27cf.chunk.css",
  "main.js": "/static/js/main.1ce71a2d.chunk.js",
  "main.js.map": "/static/js/main.1ce71a2d.chunk.js.map",
  "runtime-main.js": "/static/js/runtime-main.3def0568.js",
  "runtime-main.js.map": "/static/js/runtime-main.3def0568.js.map",
  "static/js/2.d0eb047a.chunk.js": "/static/js/2.d0eb047a.chunk.js",
  "static/js/2.d0eb047a.chunk.js.map": "/static/js/2.d0eb047a.chunk.js.m
  "static/js/3.eb2c8977.chunk.js": "/static/js/3.eb2c8977.chunk.js",
  "static/js/3.eb2c8977.chunk.js.map": "/static/js/3.eb2c8977.chunk.js.m
  "index.html": "/index.html",
  "service-worker.js": "/service-worker.js",
  "service-worker.js.map": "/service-worker.js.map",
  "static/css/main.8c8b27cf.chunk.css.map": "/static/css/main.8c8b27cf.c
  "static/js/2.d0eb047a.chunk.js.LICENSE.txt": "/static/js/2.d0eb047a.ch
  "static/media/logo.103b5fa1.svg": "/static/media/logo.103b5fa1.svg"
},
"entrypoints": [
  "static/js/runtime-main.3def0568.js",
  "static/js/runtime-main.3def0568.js.map"
]

```

asset-manifest.json 파일은 캐싱할 파일의 목록을 볼 수 있다. 즉 오프라인에서도 이용이 가능한○ 파일들을 보여준다.



개발자도구 - Application에서 PWA 설정 확인 가능

Edit src/App.js and save to reload.

Learn React

Elements Console Sources Network Performance Application ⌂ 1 ⚙ ⋮ X

Application

- Manifest
- Service Workers**
- Storage

Service Workers

Offline Update on reload Bypass for network

http://127.0.0.1:5500/

Source [service-worker.js](#)

Received 2021. 9. 2. 오전 11:28:06

Status ● #1018 activated and is running [stop](#)

Clients <http://127.0.0.1:5500/index.html> [focus](#)

[Network requests](#) [Update](#) [Unregister](#)

Storage

- Local Storage
- Session Storage
- IndexedDB
- Web SQL
- Cookies
- Trust Tokens

개발자도구 - Application에서 PWA 설정 확인 가능

Edit src/App.js and save to reload.

Learn React

Elements Console Sources Network Performance Application ⌂ 1 ⚙ ⋮ X

Cache

| # | Name | Respon... | Content... | Content... | Time C... | Vary H... |
|---|-------------------------------------|-----------|------------|------------|-----------|-----------|
| 0 | /index.html?_WB_REVISION_=1dcd0... | basic | text/ht... | 4,535 | 2021. ... | Origin |
| 1 | /static/css/main.8c8b27cf.chunk.css | basic | text/cs... | 804 | 2021. ... | Origin |
| 2 | /static/js/2.d0eb047a.chunk.js | basic | appli... | 131,266 | 2021. ... | Origin |
| 3 | /static/js/3.eb2c8977.chunk.js | basic | appli... | 3,549 | 2021. ... | Origin |
| 4 | /static/js/main.1ce71a2d.chunk.js | basic | appli... | 2,537 | 2021. ... | Origin |
| 5 | /static/js/runtime-main.3def0568.js | basic | appli... | 2,360 | 2021. ... | Origin |
| 6 | /static/media/logo.103b5fa1.svg | basic | image... | 4,186 | 2021. ... | Origin |

Background Services

- Background Fetch
- Background Sync
- Notifications

Headers Preview

Total entries: 7

cache 페이지로 들어가면 어떤 파일을 캐싱하는지 볼 수 있다. 즉 하드에서 불러와서 사용하는 것.

Q. 특정 파일들은 캐싱안되게?

```
{
  "files": {
    "main.css": "/static/css/main.8c8b2",
    "main.js": "/static/js/main.1ce71a2d.chunk.js",
    "main.js.map": "/static/js/main.1ce71a2d.chunk.js.map",
    "runtime-main.js": "/static/js/runtime-main.3def0568.js",
    "runtime-main.js.map": "/static/js/runtime-main.3def0568.js.map",
    "static/js/2.d0eb047a.chunk.js": "/static/js/2.d0eb047a.chunk.js",
    "static/js/2.d0eb047a.chunk.js.map": "/static/js/2.d0eb047a.chunk.js.map",
    "static/js/3.eb2c8977.chunk.js": "/static/js/3.eb2c8977.chunk.js",
    "static/js/3.eb2c8977.chunk.js.map": "/static/js/3.eb2c8977.chunk.js.map",
    "index.html": "/index.html",
    "service-worker.js": "/service-worker.js",
    "service-worker.js.map": "/service-worker.js.map",
    "static/css/main.8c8b27cf.chunk.css.map": "/static/css/main.8c8b27cf.chunk.css.map",
    "static/js/2.d0eb047a.chunk.js.LICENSE.txt": "/static/js/2.d0eb047a.chunk.js.LICENSE.txt",
    "static/media/logo.103b5fa1.svg": "/static/media/logo.103b5fa1.svg"
  },
  "entrypoints": [
    "static/js/runtime-main.3def0568.js",
  ]
}
```

Q. 특정 파일들은 캐싱안되게?
node_modules/react-scripts/config/webpack.config.js

```
// @remove-on-eject-begin
/*
 * Cop
 *
 * This source code is licensed under the MIT License found in the
 * LICENSE file in the root directory of this source tree.
 */
// @remove-on-eject-end
'use strict';

const fs = require('fs');
const path = require('path');
const webpack = require('webpack');
const resolve = require('resolve');
const PnpWebpackPlugin = require('pnp-webpack-plugin');
const HtmlWebpackPlugin = require('html-webpack-plugin');
const CaseSensitivePathsPlugin = require('case-sensitive-paths-webpack-plugin');
const InlineChunkHtmlPlugin = require('react-dev-utils/InlineChunkHtmlPlugin');
const TerserPlugin = require('terser-webpack-plugin');
const MiniCssExtractPlugin = require('mini-css-extract-plugin');
```

위 경로로 들어가서

```
// Generate a service worker sc | injectma
// the HTML & assets that are part of the webnach build
isEnvProduction &&
  fs.existsSync(swSrc) &&
  new WorkboxWebpackPlugin.InjectManifest({
    swSrc,
    dontCacheBustURLsMatching: /[0-9a-f]{8}\./,
    exclude: [/\index.html/, /\.map$/, /asset-manifest\.json$/, /LICENSE/],
    // Bump up the default maximum size (2mb) that's precached,
    // to make Lazy-Loading failure scenarios less likely.
    // See https://github.com/cra-template/pwa/issues/13#issuecomment-1024
    maximumFileSizeToCacheInBytes: 5 * 1024 * 1024,
  }),
  // TypeScript type checking
useTypeScript &&
  new ForkTsCheckerWebpackPlugin({
    typescript: resolve.sync('typescript', {
      basedir: paths.appNodeModules,
    }),
    async: isEnvDevelopment,
```

```
// Generate a service worker sc | injectma
// the HTML & assets that are part of the webnach build
isEnvProduction &&
  fs.existsSync(swSrc) &&
  new WorkboxWebpackPlugin.InjectManifest({
    swSrc,
    dontCacheBustURLsMatching: /[0-9a-f]{8}\./,
    exclude: [/\index.html/, /\.map$/, /asset-manifest\.json$/, /LICENSE/],
    // Bump up the default maximum size (2mb) that's precached,
    // to make Lazy-Loading failure scenarios less likely.
    // See https://github.com/cra-template/pwa/issues/13#issuecomment-1024
    maximumFileSizeToCacheInBytes: 5 * 1024 * 1024,
  }),
  // TypeScript type checking
useTypeScript &&
  new ForkTsCheckerWebpackPlugin({
    typescript: resolve.sync('typescript', {
      basedir: paths.appNodeModules,
    }),
    async: isEnvDevelopment,
```

exclude 부분을 변경해주면 된다. 정규표현식 사용보통은 건드릴 필요 없다.