React로 Calendar app만들기

## 기본 세팅

Yarn create react-app calendar로 적당한 위치에 react-project를 만든다.

기본 세팅을 위해서 필요 없는 것들을 src폴더에서 삭제하고 App.js를 Calendar.js 로 변경 한다.

스크린샷, 앉아있는, 화면, 전화이(가) 표시된 사진

자동 생성된 설명

Calendar는 Class형 컴포넌트로 변경하고 테스트를 위해 Hello Calendar를 입력한다.

import React, { Component } from 'react'

export default class Calendar extends Component {

render() {

return (

<div>

Hello Calendar

</div>

)

}

}

export default Calendar;

index.js도 아직 붙어 있는 service worker나 index.css가 존재하기 때문에 부수적인 것들을 모두 때어내어 주자.

import React from 'react';

import ReactDOM from 'react-dom';

import Calendar from './Calendar';

ReactDOM.render(

<React.StrictMode>

<Calendar />

</React.StrictMode>,

document.getElementById('root')

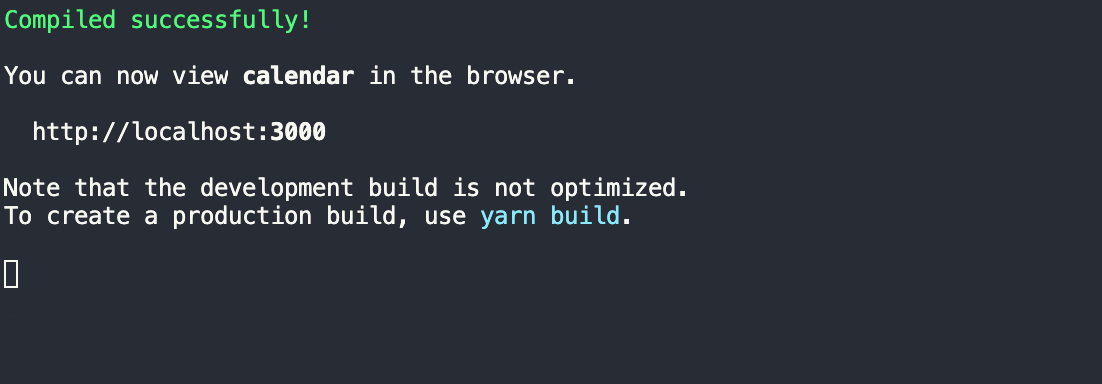
);

Git ignore 또한 public을 추가로 설정 해줌으로써 src와 package.json만 추적하도록 한다.

/.gitignore

/public

기본적인 세팅이 되었다면 yarn start를 실행하여 제대로 시작되는지 확인하도록 하자 .



스크린샷이(가) 표시된 사진

자동 생성된 설명

이제 초기 세팅은 모두 끝났으니 모두 git에 initial commit 을 한다.

## 헤더 Container 및 캘린더 Container 레이아웃 나누기

Issue발생 ! Entry point를 Calendar라고 했으나, 작성한 계획서를 보니 큰 틀의 표기는 따로 작성하지 않고, 실제 달력이 표기되어 있는 부분을 Calendar Container로 명시하였다. 이에 idex.js의 Entry point를 다시 App.js로 변경하고 App.js를 새로 만든다.

**/src/index.js**

전화이(가) 표시된 사진

자동 생성된 설명

위의 이슈를 해결하고 **/src/Header.js**를 새로 생성한다.

스크린샷, 전화, 녹색, 컴퓨터이(가) 표시된 사진

자동 생성된 설명

**App.js**의 형태를 잡기 위해 test-layout 이라는 더미 div를 만들고, 내부에 App을 차곡 차곡 구성 하도록 한다. 그리고 각 Container에 class를 부여한다.

**/src/app.js**

return (

<div className="test-layout">

<div className="RCA-app-container">

<Header />

<Calendar />

</div>

</div>

)

**/src/Header.js**

import React, { Component } from 'react'

export default class Header extends Component {

render() {

return (

<div className="RCA-header-container">

</div>

)

}

}

**/src/Calendar.js**

import React, { Component } from 'react'

export default class Calendar extends Component {

render() {

return (

<div className="RCA-calendar-container">

</div>

)

}

}

이제 대략적인 틀이 정해졌다면 layout을 위한 css작업을 실행한다.

**/src/RCA.css**

.test-layout{

width: 100vw;

height: 80vh;

border: 1px solid black;

}

*/\** this test-layout class for test it will be terminated *\*/*

.RCA-app-container{

display: flex;

flex-direction: column;

width: 100%;

height: 100%;

}

.RCA-header-container{

flex-basis: 50px;

background-color: rgb(224, 215, 202);

}

.RCA-calendar-container{

flex-grow: 1;

background-color: rgb(127, 184, 127);

}

확인을 위해 영역에 더미 색상을 주고 작업을 한다. 계획 한 디자인에서 모든 Container의 영역은 255,255,255의 화이트 칼라이지만 영역의 구분을 위해서 영역이 구분되는 상태로 두고 추후 background영역을 변경 하도록 하자 .

조류이(가) 표시된 사진

자동 생성된 설명