

Redux Dashboard

타이타닉 탑승객 명단 데이터에 대한 Dashboard 예제

Web3 2nd 프로젝트 대비

패키지 설치

```
$ yarn add @reduxjs/toolkit dayjs react-loader-spinner react-redux react-router-dom redux-devtools-extension styled-components styled-components-breakpoints styled-reset chart.js react-chartjs-2 react-countup
```

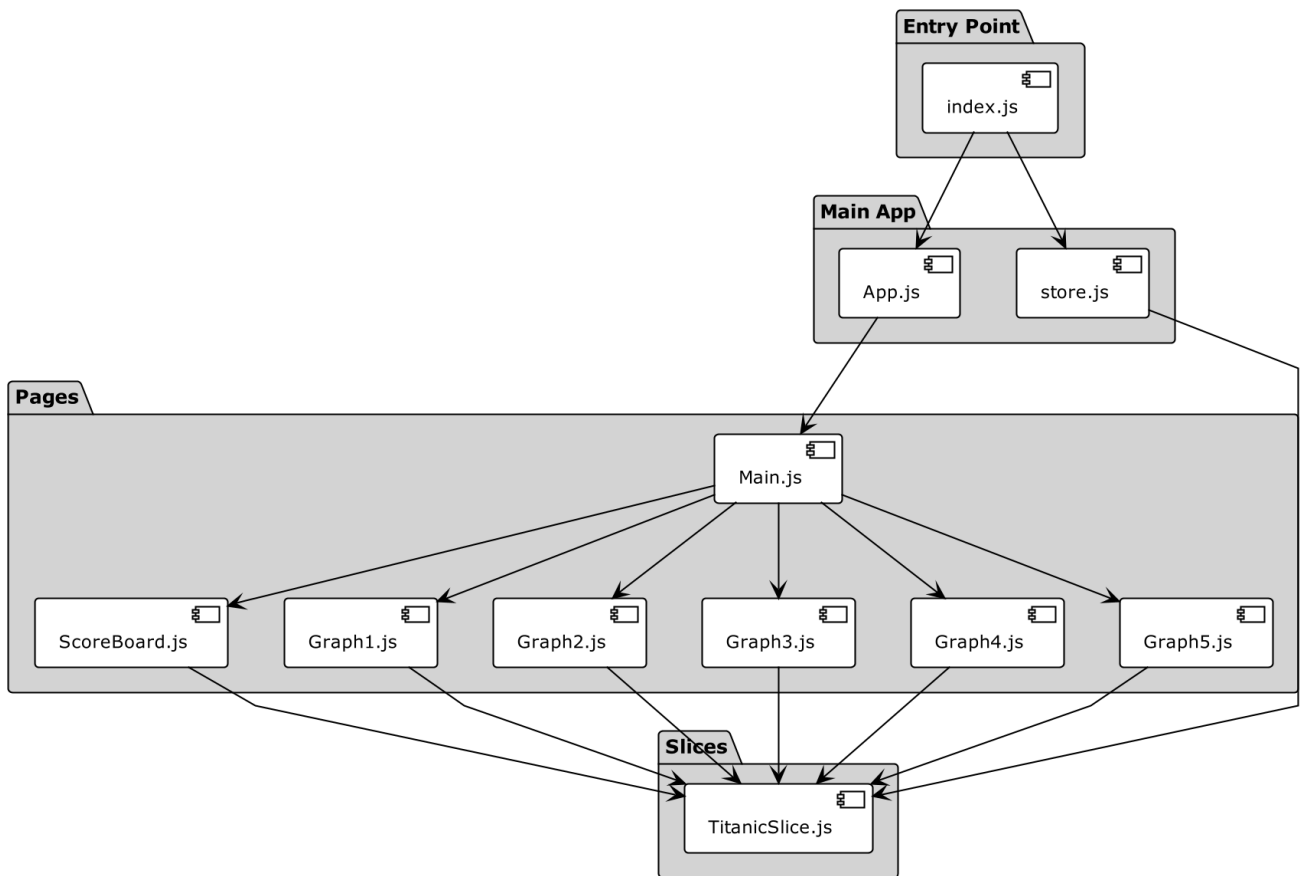
chart.js 관련 수업 내용 이외의 추가 학습은 다음의 포스팅을 참고하세요.

<https://blog.hossam.kr/posts/React-Chartjs-플러그인/>

파일 구조

```
src
├── App.js
├── index.js
├── store.js
├── components                <-- 이 폴더의 파일들은 기존 실습 파일 복사
│   ├── ErrorView.js
│   ├── GlobalStyles.js
│   ├── MediaQuery.js
│   ├── Spinner.js
│   └── Styled.js
├── helpers                  <-- 이 폴더의 파일들은 기존 실습 파일 복사
│   ├── FetchHelper.js
│   └── ReduxHelper.js
├── pages
│   ├── Graph1.js
│   ├── Graph2.js
│   ├── Graph3.js
│   ├── Graph4.js
│   ├── Graph5.js
│   ├── GraphBoard.js
│   ├── Main.js
│   └── ScoreBoard.js
└── slices
    └── TitanicSlice.js
```

컴포넌트 구조



컴포넌트 다이어그램 생성 프롬프트 (VSCode Copilot 사용)

React 프로젝트의 컴포넌트 import 계층관계 다이어그램을 "간결하게" 생성해줘.

****요구사항:****

- 출력 파일: component-diagram.puml
- 다이어그램 타입: Component diagram with packages
- 해상도: 200dpi
- 내가 작성한 소스코드에 대해서만 분석할 것.
- 각 컴포넌트에 포함되는 구성요소는 표현할 필요 없음
- 컴포넌트간의 관계만 표현할 것
- 연결선은 곡선이 아닌 직선으로 표현할 것

****포함할 파일 유형:****

- React 컴포넌트 (.js, .jsx)
- 스타일 컴포넌트 (styled-components)
- 유틸리티/헬퍼 파일\

****그룹화 기준:****

- Entry Point (index.js 등)
- Main App (App.js 등)
- Pages (페이지 컴포넌트들)
- Components (재사용 컴포넌트들)
- Utils/Helpers (유틸리티 함수들)

****제외할 파일:****

- node_modules
- 테스트 파일 (*.test.js, *.spec.js)
- 설정 파일
- 재사용 컴포넌트 및 유틸리티 헬퍼 (components/*.js, helpers/*.js)

pum1 파일 생성 후에 구문 에러가 없는지 검증하고 에러가 있다면 수정할 것.