# 캡스톤디자인Ⅱ

컴퓨터공학과

학번

이름

20197120

이병규

목 차

• 1학기 캡스톤디자인

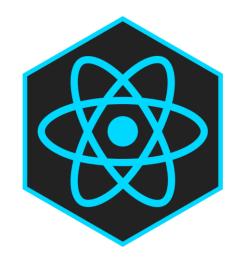
• 2학기 캡스톤디자인 계획

1학기 캡스톤디자인◀

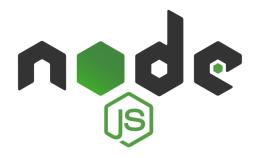
React

MongoDB

Node.js







2학기 캡스톤디자인 계획

로그인

1학기 캡스톤디자인◀

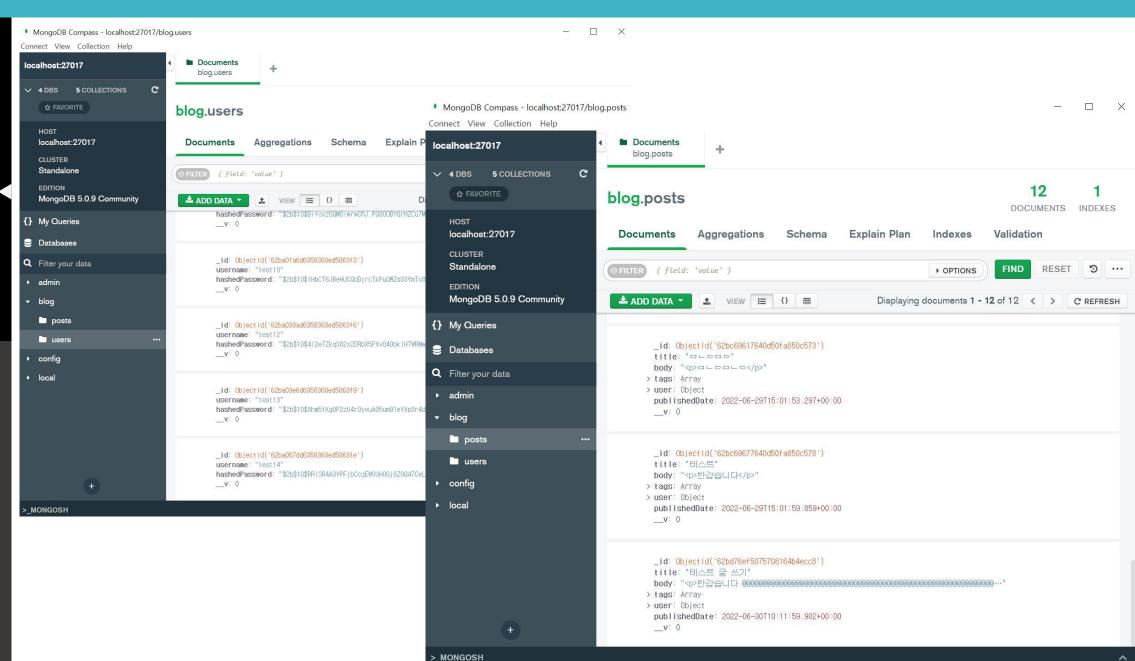
× + ₩ 이병규의 블로그 □ ☆ ★ □ ○  $\leftarrow$   $\rightarrow$  C (i) localhost:4000 이병규의 블로그 🚱 글 작성하기 - 이병규의 블로그 🗙 🕇 ← → C ① localhost:4000/write 테스트 글 쓰기 test13 · 2022. 6. 30. #테스트 #글쓰기 제목을 입력하세요 반갑습니다 내용을 작성하세요... 테스트 test13 · 2022. 6. 30. #大ヨ #大ヨ大ヨ大ヨ #a 반갑습니다 

2학기 캡스톤디자인 계획

태그 태그를 입력하세요!! 추가 #테스트 취소

1학기 캡스톤디자인◀

2학기 캡스톤디자인 계획



1학기 캡스톤디자인◀

2학기 캡스톤디자인 계획

```
파일(F) 편집(E) 선택 영역(S) 보기(V)
                                 이동(G) 실행(R) 터미널(T) 도움말(H) main.js - blog-backend - Visual Studio C... 🔲 🔲 🗍 🔘
      탐색기
    ∨ BLOG-BACKEND
                     src > 👺 main.js > .
                                         require('dotenv').config();
      > node modules
      ∨ ks src
                                         import Koa from 'ko 📢 파일(F) 편집(E) 선택 영역(S) 보기(V) 이동(G) 실행(R) 터미널(T) 도움말(H) App.js - blog-frontend - Visual Studio C... 🔲 🔲 🔐 🕒
      v 🚞 api
                                          import Router from
                                                                                                                                                                                      II ...
                                                                      탐색기
                                                                                                   App.is

✓ ■ auth
                                         import bodyParser f
           auth.ctrl.js
                                                                                      『ここで 日 Src > ∰ App.js > ...
                                          import mongoose from

✓ BLOG-FRONTEND

          index.is
                                                                                                           import './App.css';
                                         import serve from '
                                                                     > build
        ∨ 📭 models
B
                                                                      > node modules
                                                                                                           import { Route, Routes } from 'react-router-dom';
                                          import path from 'p
           post.js
                                                                      > e public
                                          import send from 'k 🧐
                                                                                                           import PostListPage from './pages/PostListPage';
           e user.js
                                                                      ∨ ks src
                                                                                                           import LoginPage from './pages/LoginPage';

✓ ■ posts

                                         import api from './
                                                                      components
                                                                                                           import RegisterPage from './pages/RegisterPage';
           mindex.js
                                          import jwtMiddlewar
                                                                       > 🛅 auth
           posts.ctrl.is
                                                                                                           import WritePage from './pages/WritePage';
                                                                       > 🔳 base
          index.is
                                                                                                           import PostPage from './pages/PostPage';
                                                                       > la common
                                         // 비구조화 할당을
       ∨ 🛅 lib
                                                                                                           import { Helmet } from 'react-helmet-async';
                                                                       > 🏬 post
          checkLoggedIn.js
                                         const { PORT, MONGO
         jwtMiddleware.js
                                                                       > posts
                                                                                                           const App = () \Rightarrow \{
         ereateFakeData.js
                                                                       > write
                                         mongoose
         index.js
                                                                                                              return (
                                                                      .connect(MONGO UR
         e main.js
                                                                       > 🛅 auth
                                           .then(() => {
        til .env
                                                                       > 📴 common
                                                                                                                   <Helmet>
                                             console.log('Co
        eslintrc.ison
                                                                       > 📭 post
                                                                                                                     <title>이병규의 블로그</title>
        prettierrc .
                                             터미널
                                                                       > posts
                                                                                                                   </Helmet>
        isconfig.json
                                                                       > m write
                                   varn run v1.22.18
                                                                                                                   <Routes>
        package.json
                                   $ nodemon --watch src/ -r esm src/i
                                                                      V 🛅 lib
                                                                                                                     <Route path="/" element={<PostListPage />} />
        yarn.lock
                                   [nodemon] 2.0.16
                                                                                                      17
                                                                       > 🚞 api
                                   nodemon] to restart at any time,
                                                                                                                     <Route path="/login" element={<LoginPage />} />
                                   [nodemon] watching path(s): src\**
                                                                       > 📑 styles
                                       mon] watching extensions: js,
                                                                                                                     <Route path="/register" element={<RegisterPage />} />
                                   [nodemon] starting `node -r esm sr
                                                                         createRequestSaga.js
                                                                                                                     <Route path="/write" element={<WritePage />} />
                                                                                                      20
                                   Listening to port 4000
                                                                      > modules
    > 개요
                                   Connected to MongoDB
                                                                                                                     <Route path="/@:username">
    > 타임라인
                                                                      > leg pages
                                                                                                                       <Route index element={<PostListPage />} />

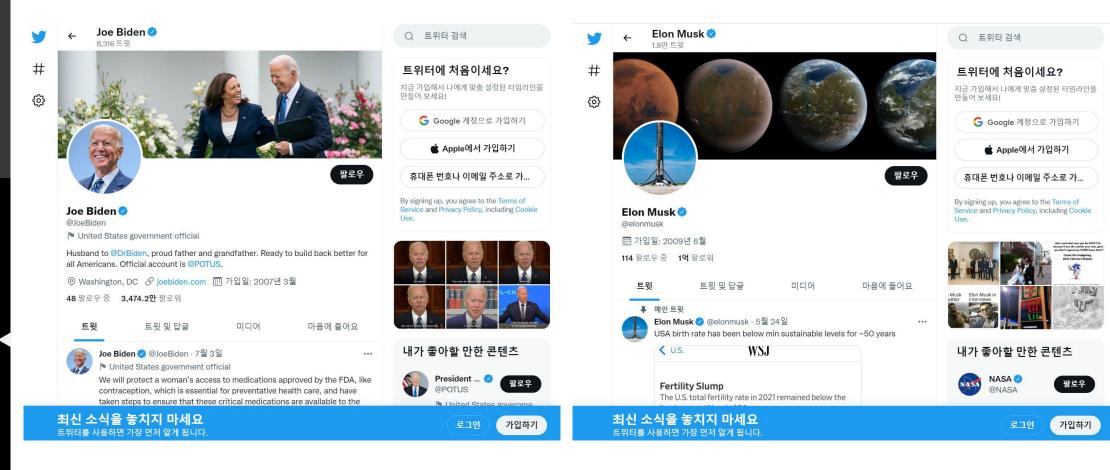
∃ App.css

                                                                                                                       <Route path=":postId" element={<PostPage />} />
                                                                         App.js
                                                                         App.test.js
                                                                                                                     </Route>
                                                                         index.css
                                                                                                                   </Routes>
                                                                         index.js
                                                                   > 개요
                                                              £63
                                                                    > 타임라인
                                                               ⊗ 0 ▲ 0
                                                                                                                             줄 1, 열 1 공백: 2 UTF-8 LF {} JavaScript React ♀ Go Live ✓ Prettier 尽 ♀
```

#### SNS 웹 개발

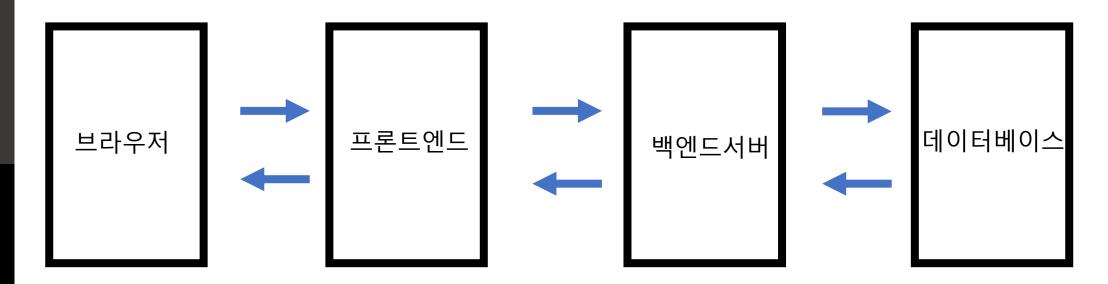
1학기 캡스톤디자인

2학기 캡스톤디자인◀ 계획



### Server Side Rendering (전통 방식)

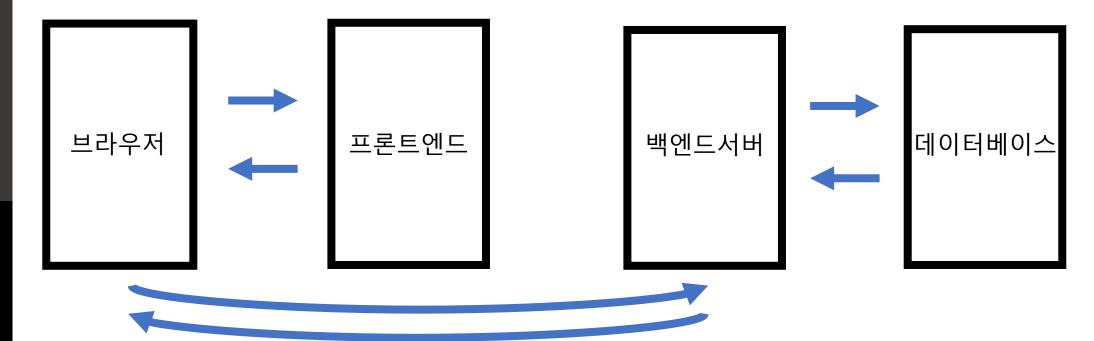
1학기 캡스톤디자인



2학기 캡스톤디자인◀ 계획

### **Client Side Rendering (React)**

1학기 캡스톤디자인



2학기 캡스톤디자인◀ 계획

• 단점

모든 JS 파일을 로드하고 사용자는 웹을 본다. 이때까지 사용자는 많은 대기시간을 가진다.

1학기 캡스톤디자인

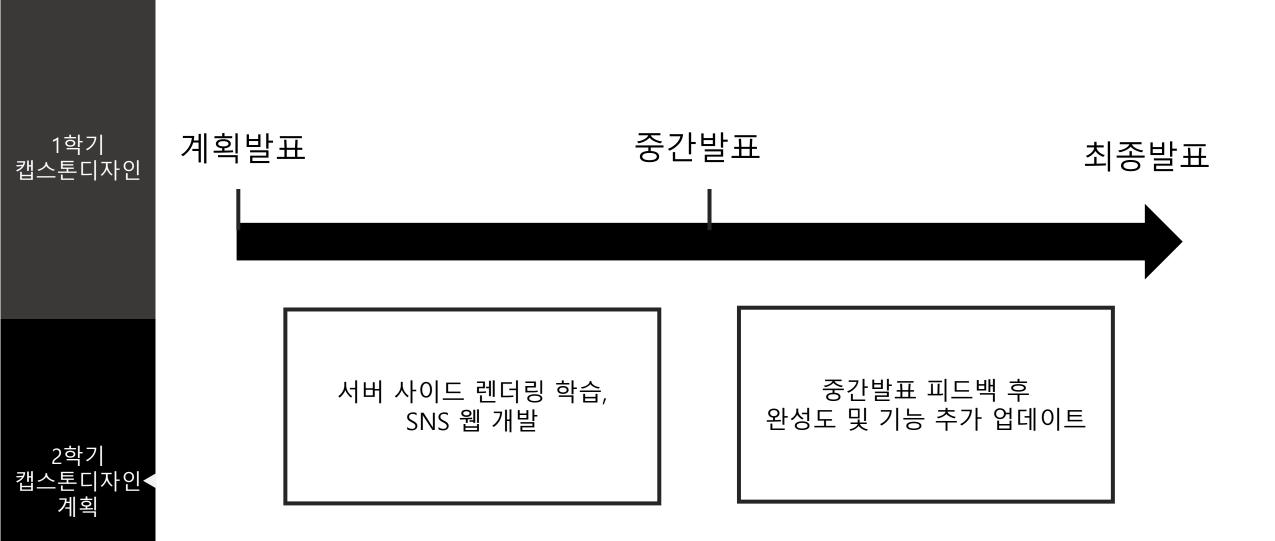


Next.js

첫 방문 했을 때 Server Side Rendering으로 로드하고

다른 페이지로 이동할 때는 Client Side Rendering 방식

2학기 캡스톤디자인◀ 계획



## 감사합니다