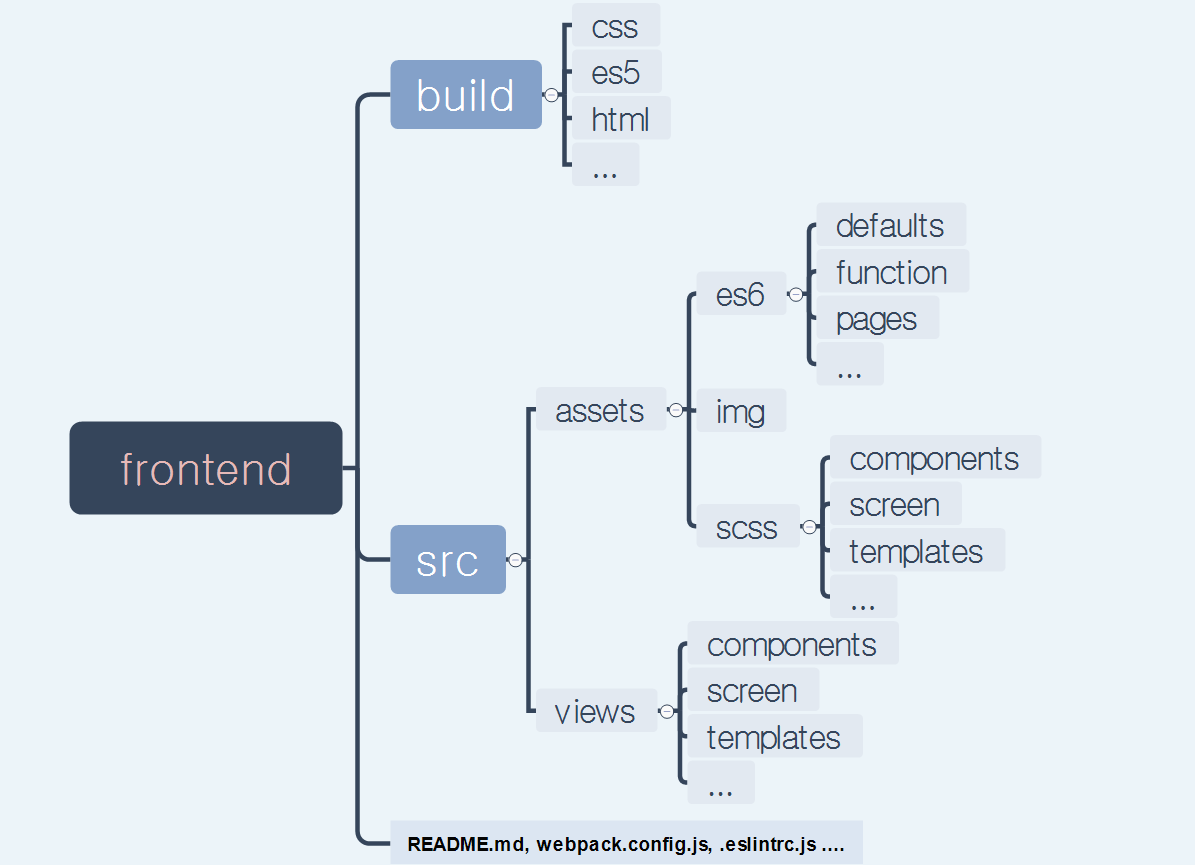
**커스텀 웹팩 설명서** update 200920

개발 디렉터리는 다음과 같이 구성된다.



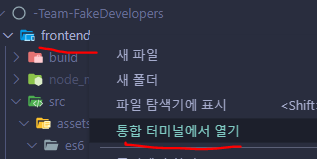
build/ 는 서버에 배포될 파일들이 들어있는 디렉터리이며, webpack dev server의 베이스 폴더가 된다. 웹팩 사이클을 통과한 파일들이 번들링, 빌드되어 나오는 곳이다.

src/ 는 개발 작업이 진행되는 디렉터리이다. ext에 따라 나뉜 디렉터리는 용도에 따라 다시한번 세분화된다.

frontend/ 는 루트 디렉터리를 명시한다.

웹팩을 사용하여 작업 시작하기

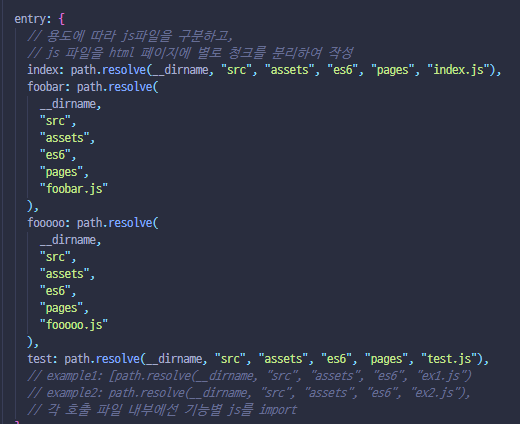
1. 프로젝트 디렉터리를 clone 한다.
2. vscode git bash 터미널로 프로젝트 최상위 폴더 아래 있는 frontend/ 에서 터미널을 연다



1. 터미널에 npm install 을 입력하여 packge.json에 명시된 모듈들을 전부 설치한다. (npm 패키지 사용을 위해 node 사전 설치 필요)
2. npm run build 명령을 실행하여 웹팩 사이클을 수행한다.
3. 작업이 끝나면 npm start 명령을 실행하여 webpack-dev-server를 킨다
4. src/ 안에서 작업을 시작한다.

🕭 주의사항 – **새로운 페이지를 만들때마다,** 조금 복잡한 작업을 수행해줘야 한다.

1. src/views/screen/ 안에 새로운 pug 파일을 만든다.
2. 같은 이름의 js 파일을 src/assets/es6/pages/ 안에 다른 js 파일을 참고하여 기본 틀 양식을 지켜 만든다.
3.  에서  이 배열에 파일의 이름을 문자열로 추가해준다.
4. entry에서 위에서 추가한 js 파일의 ‘’[파일명]: [절대경로]” 를 입력해준다.



1. **build 폴더 전체 혹은 build 폴더 안의 html 폴더를 통째로 삭제한다.**
2. npm run build 명령을 수행한다.
3. 파일이 잘 생성됐는지 확인한다.