

# 상세 페이지 구현

## 1 콘텐츠 다루기

상세 페이지

컨텐츠(이미지 / 글 / 영상...)

내용과 배치가 중요

## 사람에게 보이는 부분

- 이미지
- 글

## 봇(bot)에게 보이는 부분

- robots.txt
- meta data
- 시멘틱 HTML

먼저,  
bot을 위해서

robots.txt 와 sitemap 설정을 해주세요

next-sitemap

(<https://www.npmjs.com/package/next-sitemap>)

```
yarn add -D next-sitemap
```

next-sitemap.config.js

파일 생성

```
module.exports = {  
  siteUrl: [블로그 url],  
  generateRobotsTxt: true,  
}
```

package.json

script 추가

‘postbuild’: ‘next-sitemap’

## SEO

(<https://developers.facebook.com/docs/sharing/webmasters/>)

(<https://reactjs.org/>)



og:title

og:type

og:url

og:description

og:image

## 컨텐츠 분류

소개글 / 카테고리 / 상세 설명 / 후기 / 이벤트 / 기타 등등

컨텐츠를 어떻게 저장하고,  
어떻게 보여줄 것인가?

mysql 에 저장해볼까?

Notion API를 활용해서 md 컨텐츠를 관리하면 좋을까?

그냥 줄글로 저장하게 하는게 좋을까?

admin 톨이 필요하려나?

draft.js

(<https://draftjs.org/>)

slate.js

(<https://docs.slatejs.org/>)

tiptap

(<https://tiptap.dev/>)

draft.js 기반 react-draft-wysiwyg을 써보자

```
yarn add draft-js react-draft-wysiwyg
```

```
yarn add -D @types/draft-js @types/react-draft-wysiwyg
```

\* wysiwyg(위즈윅) what you see is what you get

Editor를 만들자

작성시에도 사용하고 / 보여줄때도 사용할 수 있도록

products 테이블에 contents를 추가하자 type은 TEXT로

```
ALTER TABLE products ADD COLUMN contents TEXT NULL;
```

## 정리

### 컨텐츠 다루기

상세 페이지

이미지 / 글 / 영상 ...

대상

봇 / 사용자

다양한 서식 담기

Editor 활용 draft.js

DB에 직접 추가

보여질때는 read / 수정하고는 update