

Next.js 2021

- *<https://slides.com/woongjae/nextjs2021>*
- *<https://github.com/xid-mark/my-blog-contents>*
- *<https://github.com/xid-mark/my-blog>*



Mark Lee

이웅재

Lead Software Engineer @ProtoPie

Microsoft MVP

TypeScript Korea User Group Organizer

Marktube (Youtube)

1. Sanity 로 Blog 모델 만들기

1-1) *What is Sanity?*

1-2) *Sanity Project 만들고 Deploy 하기*

1-3) *Schema 만들기 (1) - author*

1-4) *Schema 만들기 (2) - post*

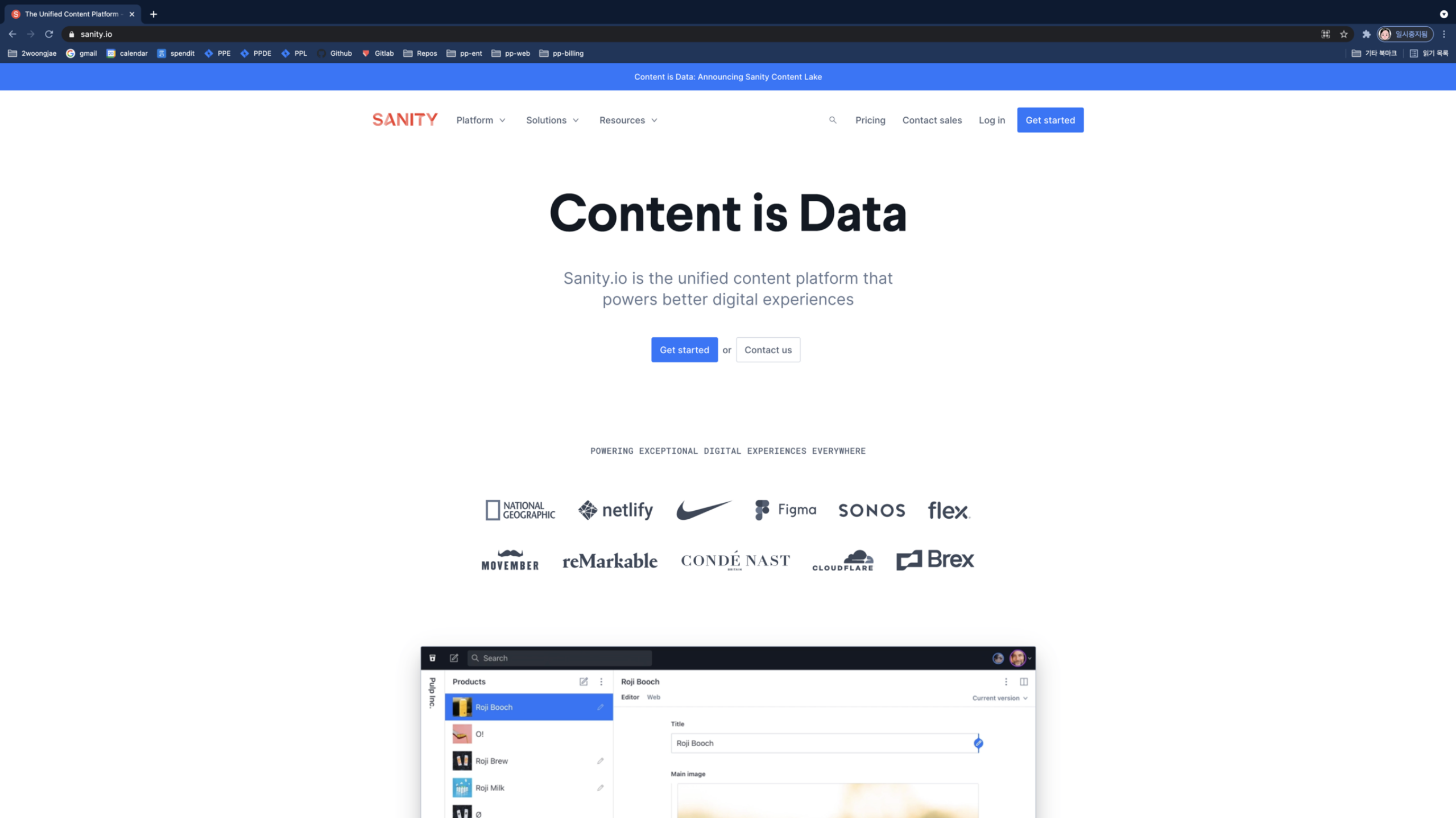
1-5) *Schema 만들기 (3) - home*

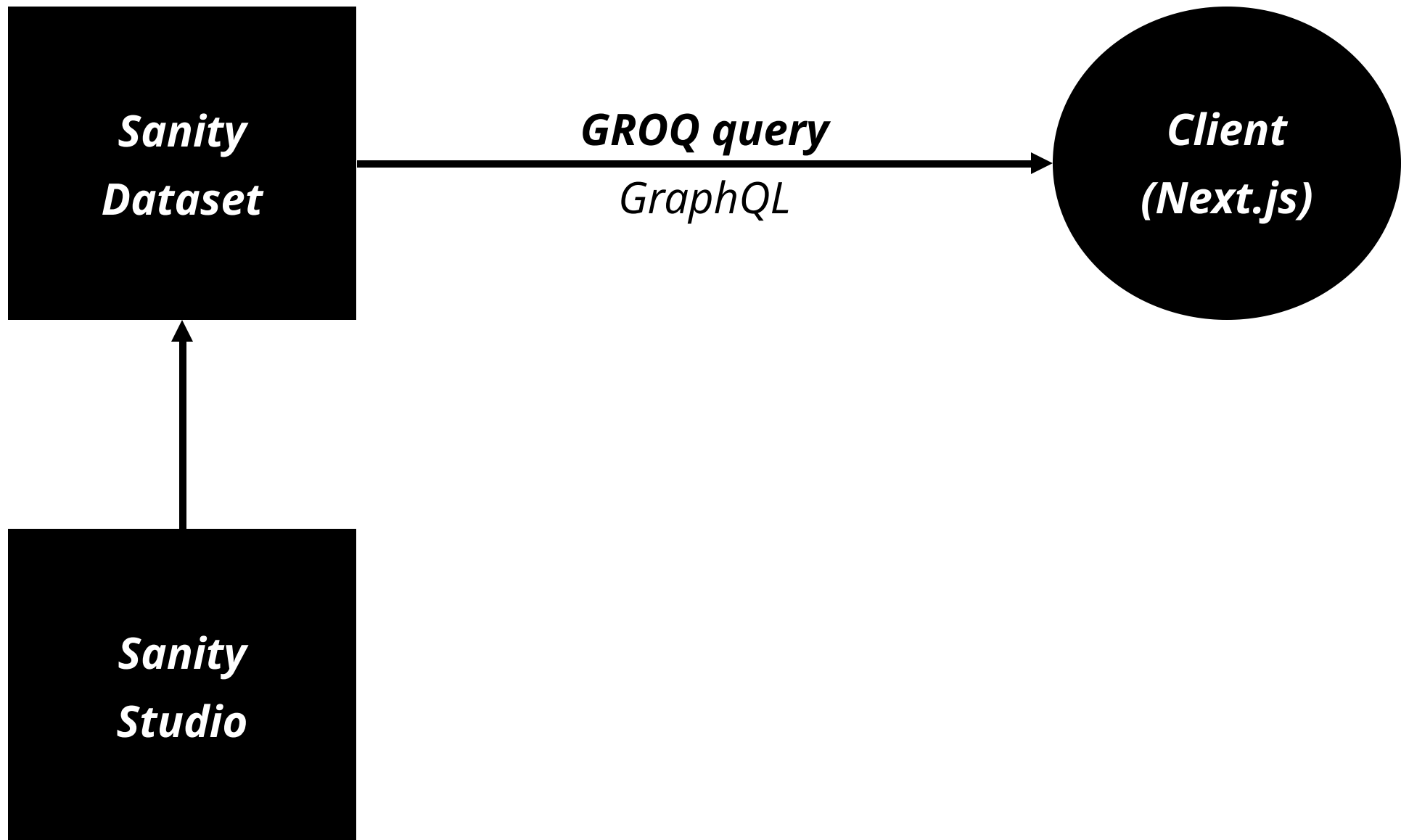
1-6) *Schema 만들기 (4) - blockContent*

1-7) *Studio 의 input 창 커스터마이징*

1-8) *query 사용하기*

What is Sanity?





Sanity Project 만들고 Deploy 하기

Start with the CLI

Install the Sanity CLI to create a local Sanity project.



```
npm install @sanity/cli -g  
sanity init
```

[Read the CLI docs](#)


```
npm i @sanity/cli -g
```

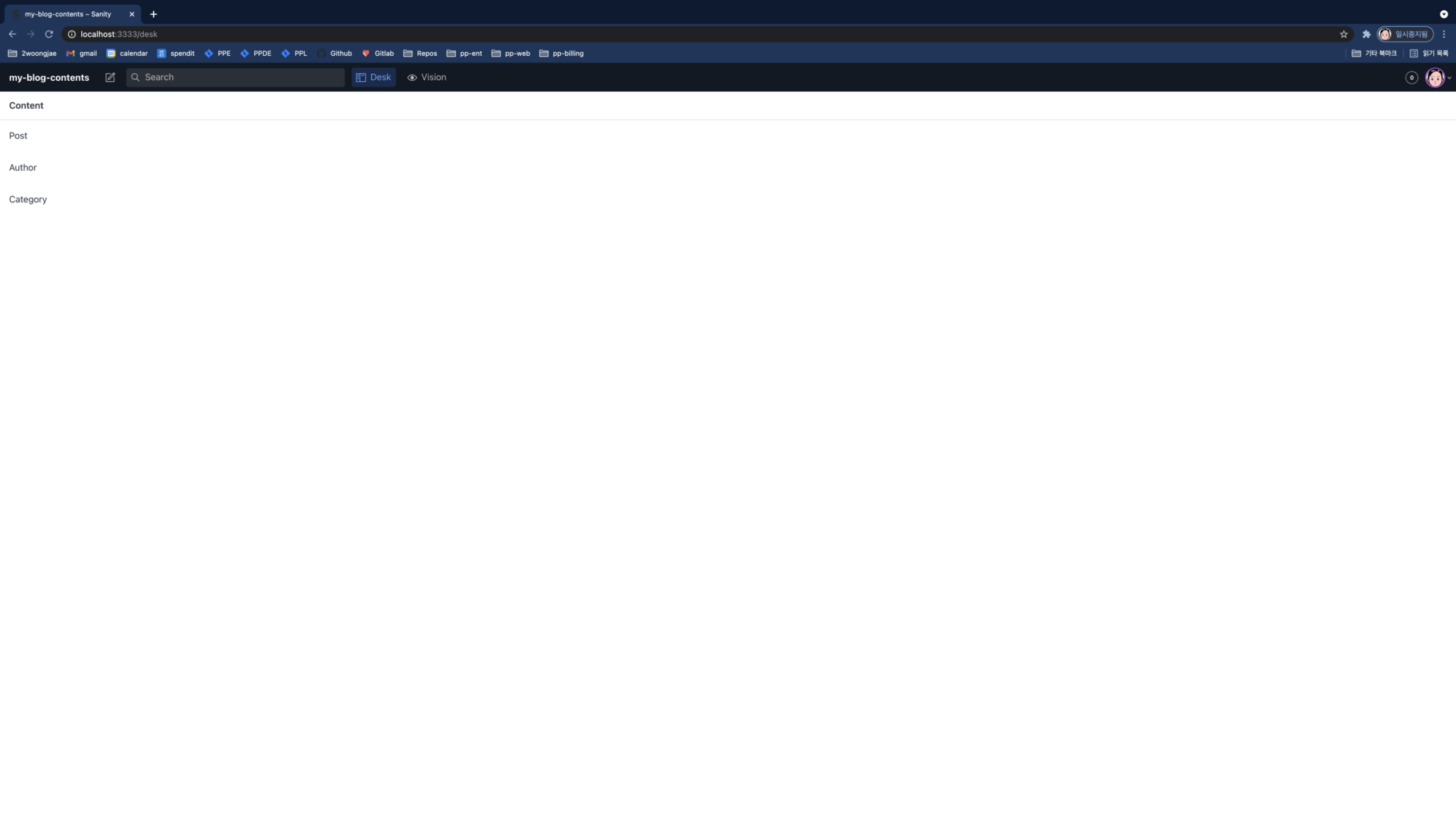
```
sanity login
```

```
mkdir my-blog-content
```

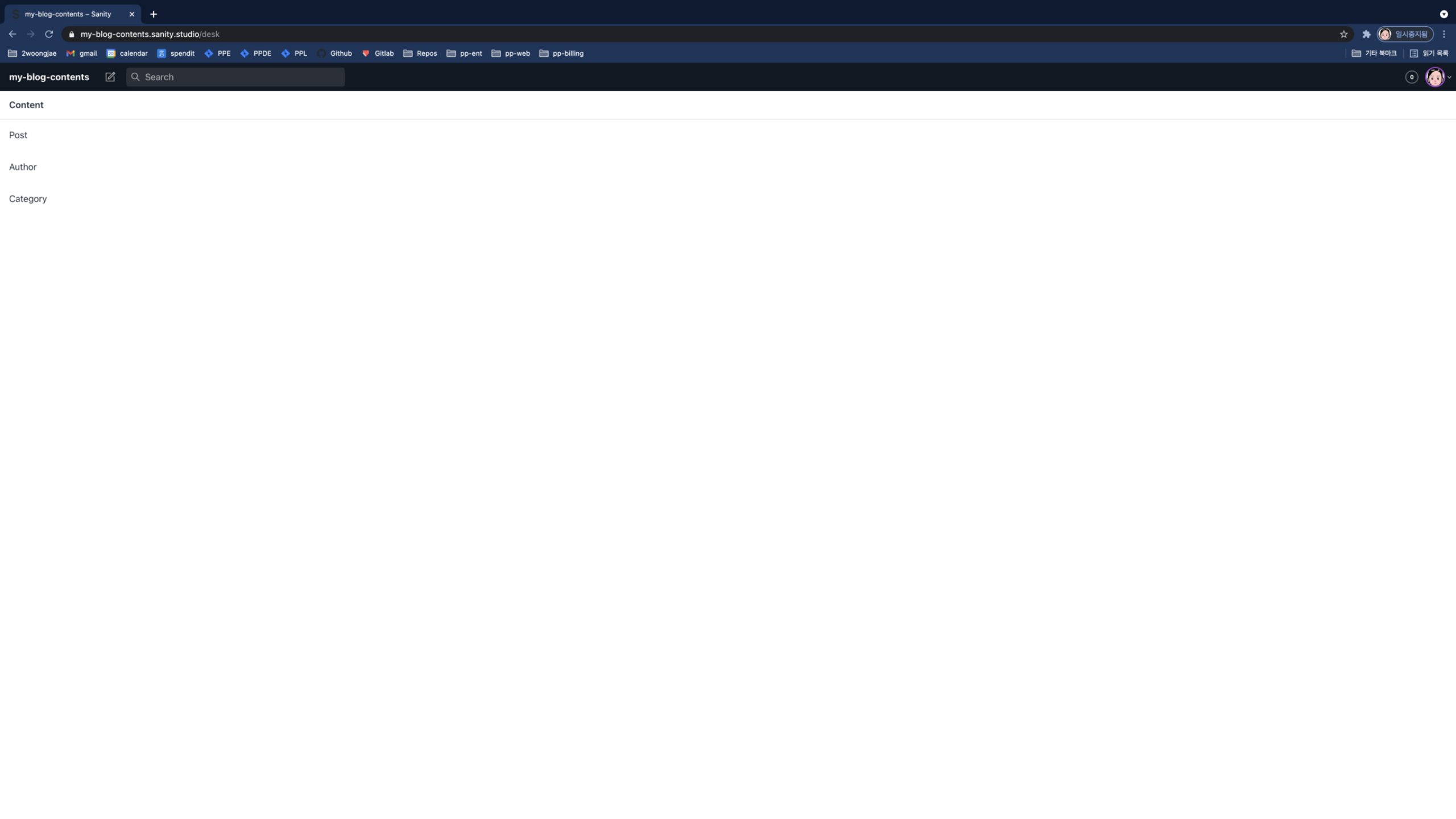
```
cd my-blog-content
```

```
sanity init
```

sanity start



sanity deploy



Schema 만들기 (1) - author

```
// schemas/author.js
```


```
export default {
  name: "author",
  title: "Author",
  type: "document",
  fields: [
    {
      name: "name",
      title: "Name",
      type: "string",
      validation: (Rule) => Rule.required(),
    },
    {
      name: "role",
      title: "Role",
      type: "string",
    },
    {
      name: "image",
      title: "Image",
      type: "image",
      options: {
        hotspot: true,
      },
      validation: (Rule) => Rule.required(),
    },
  ],
  preview: {
    select: {
      title: "name",
      media: "image",
    },
  },
};
```

my-blog-contents - Sanity

localhost:3333/desk/author;c49b118c-a048-464d-b5c8-fefc78cf8734%2Ctemplate%3Dauthor

2woongjae, gmail, calendar, spendit, PPE, PPDE, PPL, Github, Gitlab, Repos, pp-ent, pp-web, pp-billing

my-blog-contents, Search, Desk, Vision

Content	Author	Mark Lee
Post	<div> Mark Lee</div>	
Author		
Category		


Name

Mark Lee

Role

Software Engineer @ ProtoPie

Image



Upload

Select

Edit details

Remove

Published 8m ago

Publish

Schema 만들기 (2) - post

Schema 만들기 (3) - home

Schema 만들기 (4) - blockContent

Studio 의 *input* 창 커스터마이징

query 사용하기

```
*[_type == 'home'][0]{ 'mainPostUrl': mainPost -> slug.current}
```

```
*[_type == 'post']{
  title,
  subtitle,
  createdAt,
  'content': content[]{
    ...,
    ...select(_type == 'imageGallery' => {'images': images[]{..., 'url': asset -> url}})
  },
  'slug': slug.current,
  'thumbnail': {
    'alt': thumbnail.alt,
    'imageUrl': thumbnail.asset -> url
  },
  'author': author -> {
    name,
    role,
    'image': image.asset -> url
  },
  'tag': tag -> {
    title,
    'slug': slug.current
  }
}
```

2. Next.js 로 Blog 만들기

2-1) *What is Next.js?*

2-2) *Next.js 프로젝트 만들기*

2-3) *라우팅 설정하기*

2-4) *Sanity 연결하고 데이터 가져오기*

2-5) *스타일 작업 (1) - Blog Home*

2-6) *스타일 작업 (2) - Post*

2-7) *next.config.js*

2-8) *Next.js 배포 이해하기*

What is Next.js?

[Showcase](#)[Docs](#)[Blog](#)[Analytics](#)[Commerce](#)[Enterprise](#)[Learn](#)

NEW Next.js Conf – A special event to improve how you develop for the web. [Claim Ticket →](#)

The React Framework for Production

Next.js gives you the best developer experience with all the features you need for production: hybrid static & server rendering, TypeScript support, smart bundling, route pre-fetching, and more. No config needed.

[Start Learning](#)[Documentation](#)

License: MIT [GitHub](#)

Why Next.js

The world's leading companies use and love Next.js

Image Optimization

<Image> and Automatic Image Optimization with instant builds.

[Documentation →](#)

Internationalization

Built-in Domain & Subdomain Routing and Automatic Language detection.

[Documentation →](#)

Next.js Analytics

A true lighthouse score based on real visitor data & page-by-page insights

[Documentation →](#)

Zero Config

Automatic compilation and bundling.

Hybrid: SSG and SSR

Pre-render pages at build time (SSG) or

Incremental Static Regeneration

Next.js 프로젝트 만들기

- **Manual Setup**
- **create-next-app**

라우팅 설정하기

- `pages/index.js` → `/`
- `pages/blog/index.js` → `/blog`

- `pages/blog/first-post.js`
→ `/blog/first-post`
- `pages/dashboard/settings/username.js`
→ `/dashboard/settings/username`

- `pages/blog/[slug].js`
→ `/blog/:slug (/blog/hello-world)`
- `pages/[username]/settings.js`
→ `/:username/settings (/foo/settings)`
- `pages/post/[...all].js`
→ `/post/* (/post/2020/id/title)`

Sanity 연결하고 데이터 가져오기

- **getStaticProps** (Static Generation)
Fetch data at **build time**.
- **getStaticPaths** (Static Generation)
Specify **dynamic routes** to pre-render pages based on data.
- **getServerSideProps** (Server-side Rendering)
Fetch data on **each request**.

스타일 작업 (1) - Blog Home

스타일 작업 (2) - Post

next.config.js

Next.js 배포 이해하기