

배포(Deploy)

Github, Netlify

ProductAllpage.js

```
App.js ProductAllpage.js ×
src > pages > ProductAllpage.js > ProductAllpage > productList.map() callback
1 import React, {useEffect, useState} from 'react';
2 import ProductBox from '../components/ProductBox';
3
4 const ProductAllpage = () => {
5   const [productList, setProductList] = useState([]);
6   const getProduct = async () =>{
7     let url = `http://localhost:3004/products`;
8     let response = await fetch(url);
9     let data = await response.json();
10    setProductList(data);
11  }
12  useEffect(()=>{getProduct()},[]);
```

Url에 있는 json파일은 로컬(내컴퓨터)내부에서 작동한다.
배포 전 myjson-server라는 제이슨 서버를 외부에서 작동하도록 해야 된다.

myJsonServer

<https://my-json-server.typicode.com/>

How to

1. Create a repository on GitHub (`<your-username>/<your-repo>`)
2. Create a `db.json` file
3. Visit <https://my-json-server.typicode.com/<your-username>/<your-repo>> to access your server

No registration. Nothing to install.

Example

You can check the following server as an example:
<https://my-json-server.typicode.com/typicode/demo>

Beta

For the moment, the project is in **beta** which means many things may change or break:

- URLs may change
- Service may be down
- Authentication may be added later
- etc...

Use it at your own risk. If you need a **reliable** solution, please **install** [JSON Server](#).

Limits

To be able to provide a free service to as many people as possible during this phase, the project comes with a few limits:

- **Changes are faked and aren't persisted** (just like [JSONPlaceholder](#))
- Requests are cached (`1 minute`)

1. GitHub에 레파지토리 생성
(`<your-username>/<your-repo>`)
2. db.json파일 만들기
3. 서버에 액세스하려면 링크변경
`https://my-json-server.typicode.com/<username>/<repo>`

1. GitHub에 레파지토리 생성

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

1

Owner *

Repository name *

song-jiae

/

kolonmallrepo

kolonmallrepo is available.

Great repository names are short and memorable. Need inspiration? How about [potential-eureka](#) ?

Description (optional)

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

You are creating a public repository in your personal account.

2

Create repository

song-jiae / kolonmallrepo

code Issues Pull requests Actions Projects Wiki Security Insights Settings

kolonmallrepo (Public)

Pin Unwatch

Set up GitHub Copilot
Use GitHub's AI pair programmer to autocomplete suggestions as you code.
[Get started with GitHub Copilot](#)

Add collaborators to this repository
Search for people using their GitHub username
[Invite collaborators](#)

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH <https://github.com/song-jiae/kolonmallrepo.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# kolonmallrepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/song-jiae/kolonmallrepo.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/song-jiae/kolonmallrepo.git
git branch -M main
git push -u origin main
```

How to

1. Create a repository on GitHub (`<your-username>/<your-repo>`)
2. Create a `db.json` file
3. Visit <https://my-json-server.typicode.com/<your-username>/<your-repo>> to access your server

No registration. Nothing to install.

```
const ProductAllpage = () => {  
  const [productList, setProductList] = useState([]);  
  let [query, setQuery] = useSearchParams();  
  const getProduct = async () => {  
    let searchQuery = query.get('q') || '';  
    let url = `http://localhost:3004/products?q=${searchQuery}`;  
    let response = await fetch(url);  
    let data = await response.json();  
    setProductList(data);  
  }  
}
```

ProductAllpage.js

예시

<https://my-json-server.typicode.com/<your-username>/<your-repo>>



레파지토리 왼쪽 상단에 있는
각자 이름, 레파지토리명 확인해서 붙여넣기

<https://my-json-server.typicode.com/song-jiae/kolonmallrepo>

ProductAllpage.js

```
5  ✓ const ProductAllpage = () => {  
6    const [productList, setProductList] = useState([]);  
7    let [query, setQuery] = useSearchParams;  
8  ✓  const getProduct = async () => {  
9    let searchQuery = query.get('q') || '';  
10   ⚡ let url = `https://my-json-server.typicode.com/song-jiae/kolonmallrepo/products?q=${searchQuery}`;  
11    let response = await fetch(url);  
12    let data = await response.json();  
13    setProductList(data);  
14  }
```

ProductDetailpage.js 페이지도 url주소 바꿔주세요

깃허브에 프로젝트 올리기


1. git init
2. git add .
3. git commit -m '첫번째 커밋'
4. git commit -m "first commit"
5. git remote add origin https://github.com/song-jiae/kolonmallrepo.git
6. git push origin main



...or create a new repository on the command line




```
echo "# kolonmallrepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/song-jiae/kolonmallrepo.git
git push -u origin main
```




업로드 된 파일 확인


 kolonmallrepo Public








 Pin  Unwatch 1

 main  1 branch  0 tags

[Go to file](#) [Add file](#) [Code](#)

 song-jiae first commit

24c859b 1 minute ago  1 commit

 public	first commit	1 minute ago
 src	first commit	1 minute ago
 .gitignore	first commit	1 minute ago
 README.md	first commit	1 minute ago
 db.json	first commit	1 minute ago
 package-lock.json	first commit	1 minute ago
 package.json	first commit	1 minute ago

Netlify(호스팅) <https://app.netlify.com/>

The screenshot shows the Netlify dashboard for a team named 'songjiae'. The interface includes a sidebar on the left with navigation links: Team overview (highlighted with a red circle and the number 1), Sites, Builds, Integrations, Domains, Members, Audit log, Billing, and Team settings. The main content area features a top banner with a 'Try Pro' offer and a 'Start Pro trial' button. Below this, a card for the 'songjiae Starter' plan shows usage statistics: 0 KB/100 GB for bandwidth, 0/300 for build minutes, and 0/1 for team members. A green square with a white 'S' represents the team's logo. The 'Sites' section is highlighted with a red circle and the number 2, showing two options: 'Import an existing project' (with an 'Import from Git' button) and 'Start from a template' (with a 'Browse templates' button). Below these, there is a section for manual deployment with a dashed box for dragging and dropping files, and a link to 'browse to upload'. An 'Upgrade' button is located at the bottom left of the sidebar.

Team overview | songjiae | Netlify

app.netlify.com/teams/songjiae/overview

Search anything...

News Support

Team overview 1

Sites

Builds

Integrations

Domains

Members

Audit log

Billing

Team settings

Try Pro free for 1 month and get background functions, password-protected sites, email support, and more. Not now Start Pro trial

songjiae Starter Current usage period (Aug 31 to Sep 30)

Bandwidth: 0 KB/100 GB

Build minutes: 0/300

Team members: 0/1

Once you have a deployed site on your team, usage will appear here.

Team members 1

Sites →

Import an existing project 2

Import from Git

Start from a template

Browse templates

...or deploy manually

Drag and drop your site output folder here

Or, [browse to upload](#)

Upgrade

🏠 Team overview

📁 Sites

🔗 Builds

🔧 Integrations

🌐 Domains

👤 Members

📋 Audit log


💰 Billing

⚙️ Team settings

1. Connect to Git provider

2. Select repository

3. Configure site and deploy

Let's  deploy your project.

🔄 Deploy with GitHub

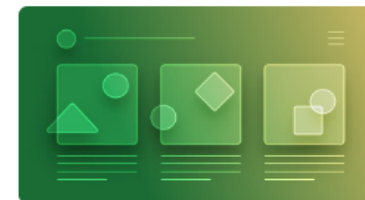
🐉 Deploy with GitLab

📁 Deploy with Bitbucket

🔄 Deploy Azure DevOps

Get started another way


[Try Netlify Drop](#) or [Choose a template](#)







Netlify by **Netlify** would like permission to:

 Verify your GitHub identity (song-jiae)

 Know which resources you can access

 Act on your behalf
[? Learn more](#)

Resources on your account

 **Email addresses** (read)
View your email addresses

[Learn more about Netlify](#)

Cancel

Authorize Netlify

Authorizing will redirect to
<https://api.netlify.com>



Install Netlify

Install on your personal account Jiae

☐ All repositories

This applies to all current *and* future repositories owned by the resource owner.
Also includes public repositories (read-only).

☒ Only select repositories

Select at least one repository.
Also includes public repositories (read-only).

Select repositories ▾

Search for a repository

with

✓

✓

Use

Net

per

✓ Read access to email addresses

3

Install

Cancel

Next: you'll be directed to the GitHub App's site to complete setup.

1. Only select repositories
2. 아까 저장한 저장소 이름찾기
3. install

1. Connect to Git provider

2. Select repository

3. Configure site and deploy

Let's deploy your project.

Review configuration for kolonmall

Deploy as song-jiae on songjiae team from main branch using npm run build command and publishing to build

Team

songjiae

Branch to deploy

main

Build settings

Specify how Netlify will build your site.

[Learn more in the docs](#)

Base directory



Git repository

kolonmall

Deploying

main

Build command

npm run build

Published to

build

Build settings

Specify how Netlify will build your site.

[Learn more in the docs](#)

Base directory

The directory where Netlify installs dependencies and runs your build command.

Build command

CI=false npm run build

Examples: `jeekyll build`, `gulp build`, `make all`

Publish directory

build

Examples: `_site`, `dist`, `public`

Functions directory

Examples: `my_functions`

Add environment variables

Deploy kolonmall


thunderous-sopapillas-c9d820


- <https://thunderous-sopapillas-c9d820.netlify.app>


Deploys from [GitHub](#).

Created at 8:15 AM.



 Site configuration

 Domain settings

 Favorite site

Set up your site

1

Your site is deployed ✓

Try a test build and deploy, directly from your Git repository or a folder.

2

[Set up a custom domain](#) →

Buy a new domain or set up a domain you already own.

3

Secure your site with HTTPS

Your site is secured automatically with a Let's Encrypt certificate.

Production deploys >

Production: main@HEAD Published

8:15 AM: No deploy message >

Deploy Previews



songjiae

thunderous-sopapillas-c9d820

Search anything...

News

- Site overview
- Site configuration
- Deploys
- Functions
- Edge Functions
- Integrations
- Analytics
- Domain management**
- Forms

Domain management

Use your own domain for your Netlify site for free.

Production domains

Your site is always accessible at a netlify.app subdomain based on the site name. Custom domains allow visitors to access your site at your own domains.

thunderous-sopapillas-c9d820.netlify.app
Netlify subdomain

[Learn more about custom domains in the docs](#)

Add a domain

Options

Edit site name

Automatic deploy subdomains

This feature is available on paid plans

Netlify can automatically create URLs for Deploy Previews and branch deploys based on your chosen custom domain. Your deploys will always be available at *--sitename.netlify.app as well. Your subdomain, or its parent domain, must be managed by [Netlify DNS](#).

Deploy Previews: deploy-preview-123--thunderous-sopapillas-c9d820.netlify.app

Branch Deploys: branch-name--thunderous-sopapillas-c9d820.netlify.app
 Set up Branch Deploys in [Build & Deploy settings](#)

[Learn more about automatic deploy subdomains in the docs](#)

Upgrade to unlock

Upgrade