리액트(React) 기초

RESTful API

김경민

HTML CSS JavaScript React JS

REST와 RESTful API

- REST(Representational State Transfer)
  - 자원을 특정 URI로 식별하고, 그 자원의 상태를 HTTP 메서드(GET, POST, PUT, DELETE 등)를 통해 변경하는 구조
- RESTful API(Representational State Transfer API)
  - REST의 원칙을 준수하는 API 또는 웹 서비스
    - GET /posts : 모든 블로그 포스트 목록 조회
    - GET /posts/1 : ID가 1인 특정 블로그 포스트 조회
    - POST /posts : 새로운 블로그 포스트 생성
    - PUT /posts/1 : ID가 1인 블로그 포스트 업데이트
    - DELETE /posts/1 : ID가 1인 블로그 포스트 삭제

supabase

정의

Supabase는 오픈소스 Firebase 대체재로, 개발자들이 백엔드 기능을 쉽고 빠르게 구축할 수 있도록 돕는 BaaS(Backend-as-a-Service) 플랫폼

PostgreSQL 데이터베이스를 기반으로 하며, 이를 통해 개발자는 백엔드 인프라 관리에 대한 부담을 줄이고 프론트엔드 개발에 집중할 수 있음

PostgreSQL 데이터베이스 기반
오픈 소스 관계형 데이터베이스 시스템을 기반

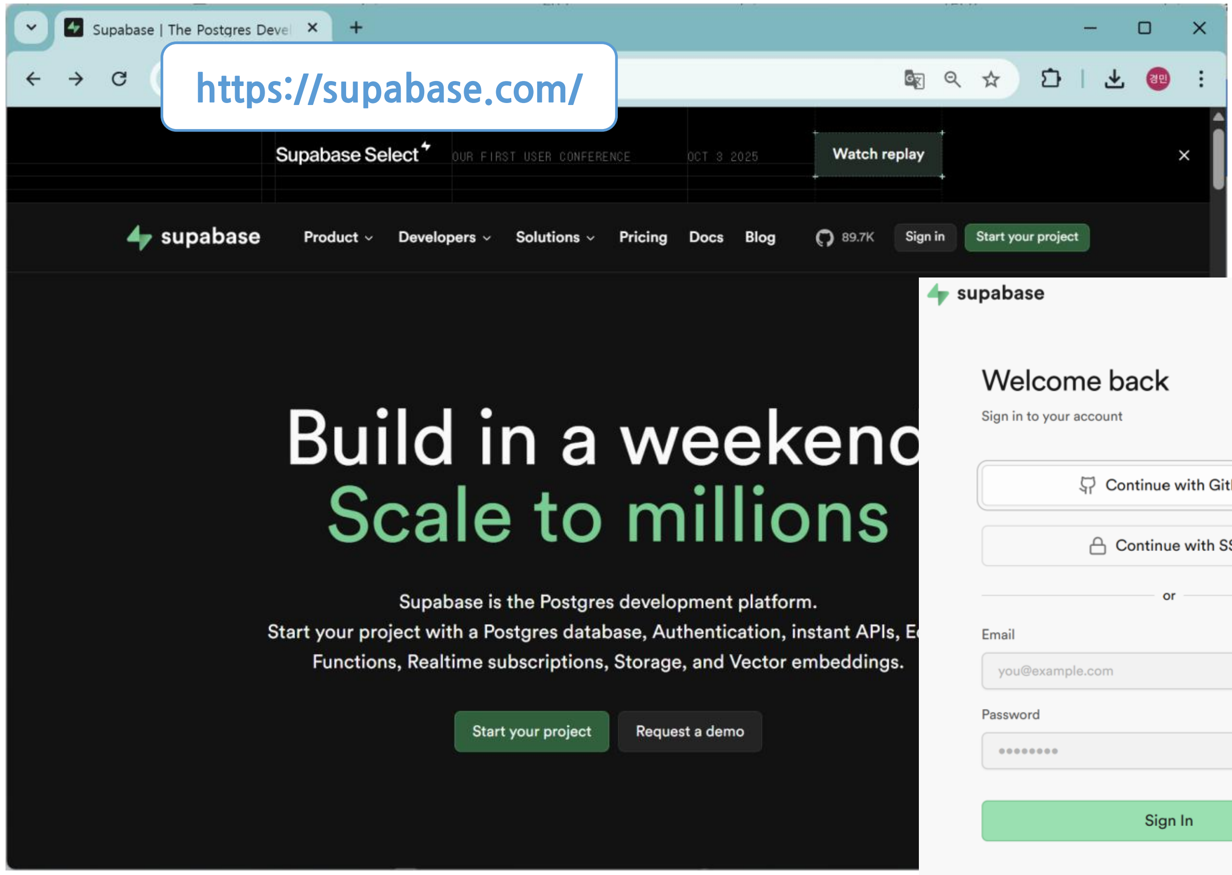
실시간 데이터베이스
데이터 변경 사항을 실시간으로 감지하고 알림을 받을 수 있음

인증(Authentication)

스토리지(Storage)
파일 업로드, 관리, 공유 기능을 쉽게 구현할 수 있음

Supabase는 개발자에게 "백엔드를 걱정하지 않아도 되는" 프론트엔드 중심 개발 환경을 제공
이를 통해 개발 속도를 높이고, 유지보수 비용을 줄일 수 있음

Supabase 프로젝트 생성



https://supabase.com/

Supabase | The Postgres Devel

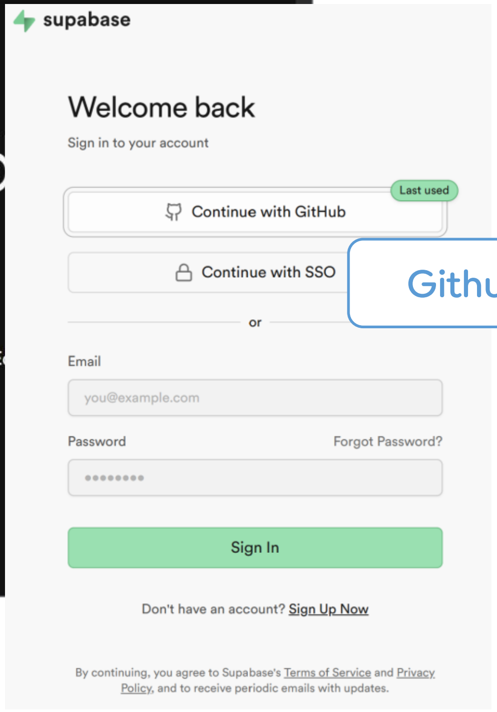
Supabase Select OUR FIRST USER CONFERENCE OCT 3 2025 Watch replay

supabase Product Developers Solutions Pricing Docs Blog 89.7K Sign in Start your project

Build in a weekend
Scale to millions

Supabase is the Postgres development platform.
Start your project with a Postgres database, Authentication, instant APIs, E
Functions, Realtime subscriptions, Storage, and Vector embeddings.

Start your project Request a demo



supabase

Welcome back

Sign in to your account

Last used

Continue with GitHub

Continue with SSO

or

Email

you@example.com

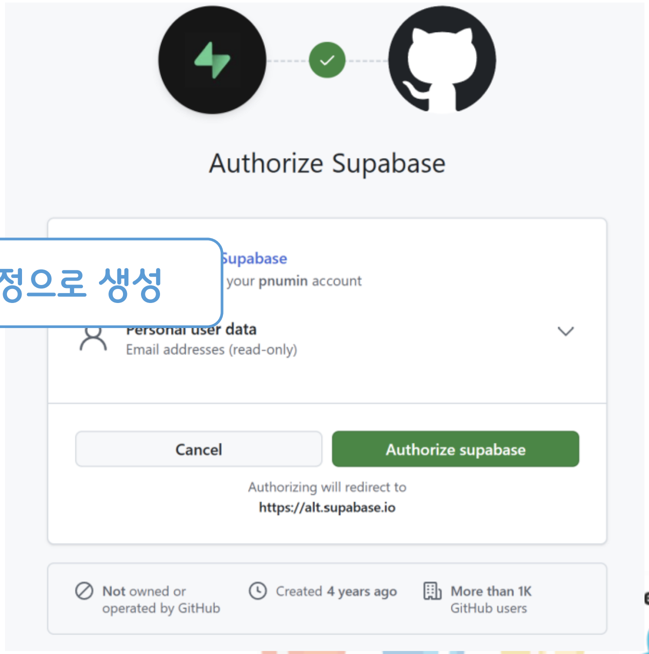
Password Forgot Password?

Sign In

Don't have an account? Sign Up Now

By continuing, you agree to Supabase's Terms of Service and Privacy Policy, and to receive periodic emails with updates.

Github 계정으로 생성



Authorize Supabase

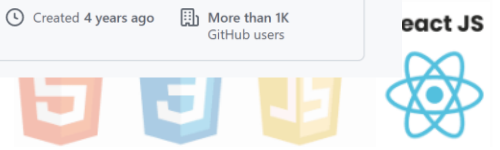
Supabase
your pnumin account

Personal user data
Email addresses (read-only)

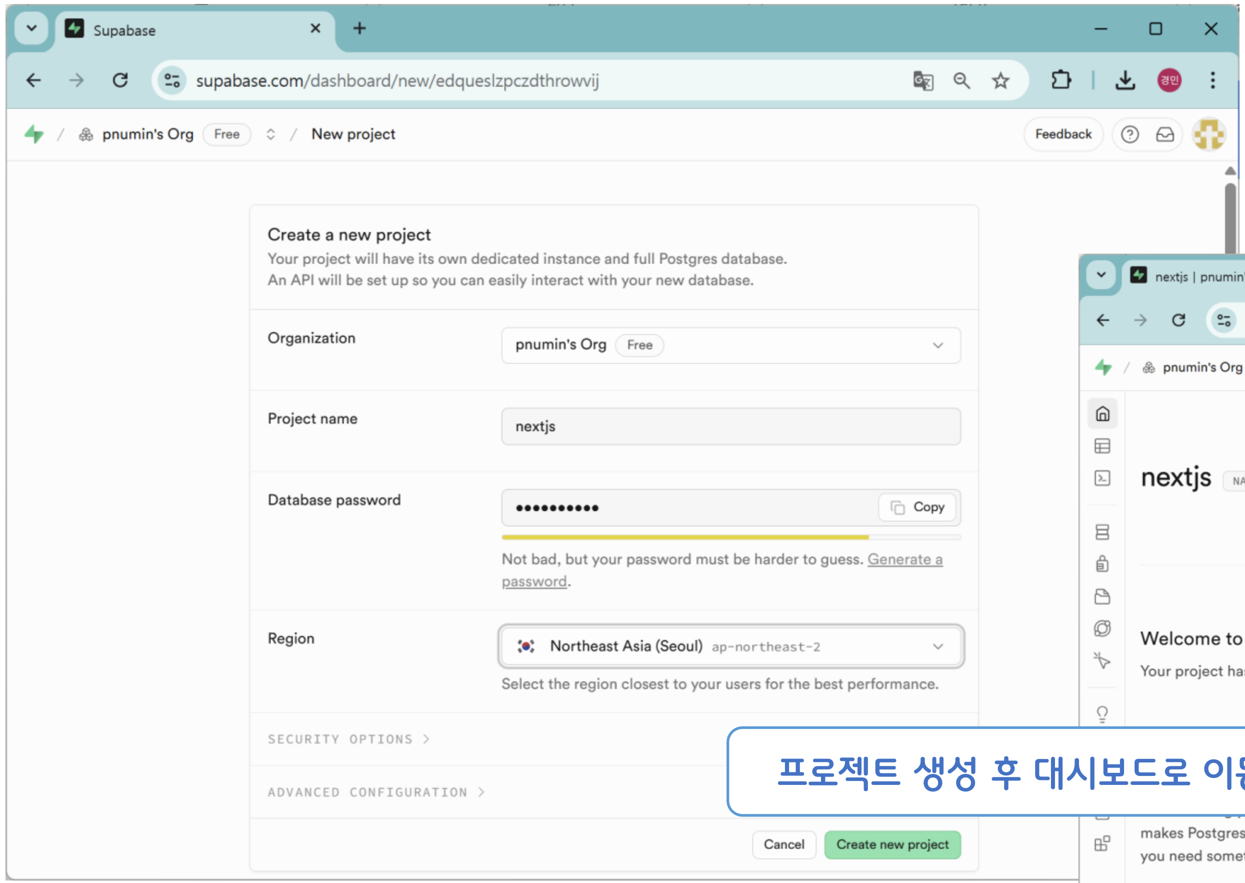
Cancel Authorize supabase

Authorizing will redirect to
https://alt.supabase.io

Not owned or operated by GitHub · Created 4 years ago · More than 1K GitHub users



Supabase 프로젝트 생성



supabase.com/dashboard/new/edqueslzpczdthrowvij

pnumin's Org Free / New project

Create a new project

Your project will have its own dedicated instance and full Postgres database.
An API will be set up so you can easily interact with your new database.

Organization	pnumin's Org Free
Project name	nextjs
Database password	●●●●●●●●●●●
Region	Northeast Asia (Seoul) ap-northeast-2

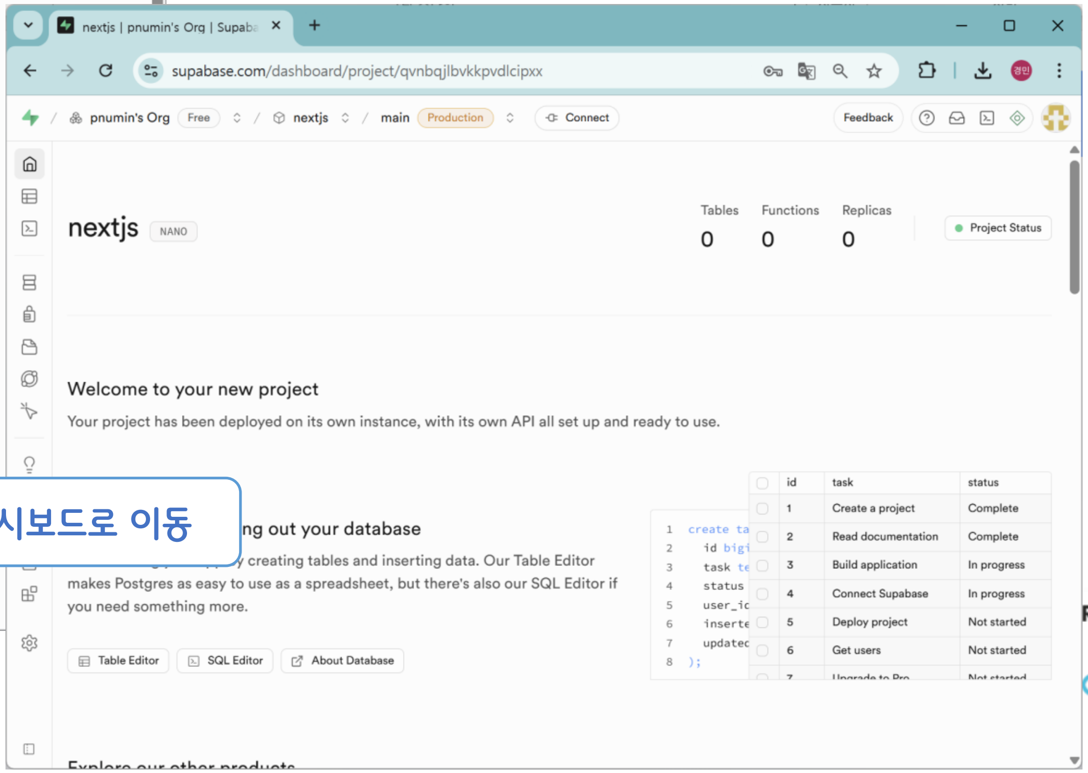
Not bad, but your password must be harder to guess. Generate a password.

Select the region closest to your users for the best performance.

SECURITY OPTIONS >

ADVANCED CONFIGURATION >

프로젝트 생성 후 대시보드로 이동



supabase.com/dashboard/project/qvnbqjlbvkkpvdlcipxx

pnumin's Org Free / nextjs / main Production / Connect

nextjs NANO

Tables	Functions	Replicas
0	0	0

Project Status

Welcome to your new project

Your project has been deployed on its own instance, with its own API all set up and ready to use.

ng out your database

y creating tables and inserting data. Our Table Editor makes Postgres as easy to use as a spreadsheet, but there's also our SQL Editor if you need something more.

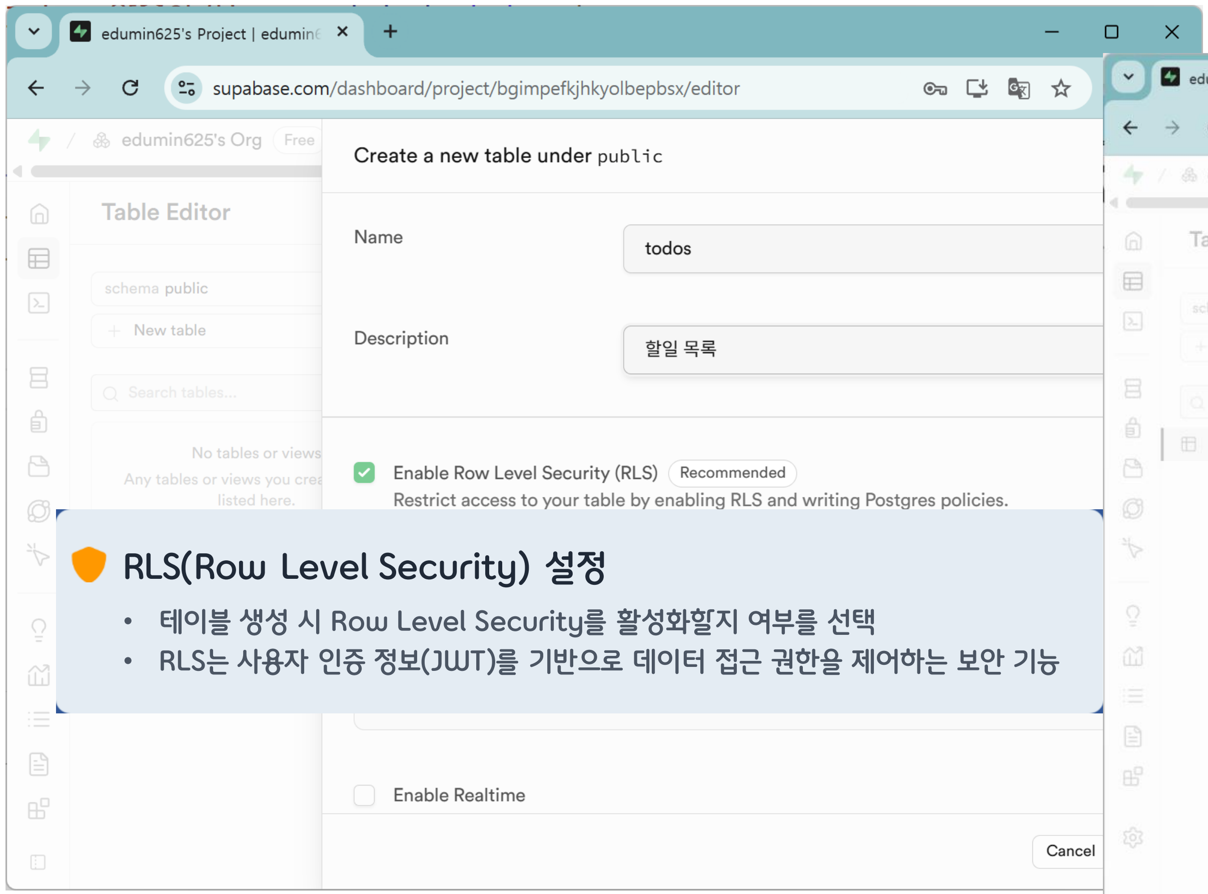
```
create ta
  id bigi
  task te
  status
  user_ic
  inserte
  updatec
);
```

id	task	status
1	Create a project	Complete
2	Read documentation	Complete
3	Build application	In progress
4	Connect Supabase	In progress
5	Deploy project	Not started
6	Get users	Not started
7	Upgrade to Pro	Not started

Explore our other products

React JS

Supabase 프로젝트 생성


edumin625's Project | edumin

supabase.com/dashboard/project/bgimpefkjhkyolbepbsx/editor

edumin625's Org Free

Table Editor

schema public

New table

Search tables...

No tables or views

Any tables or views you crea

listed here.

Create a new table under `public`

Name: todos

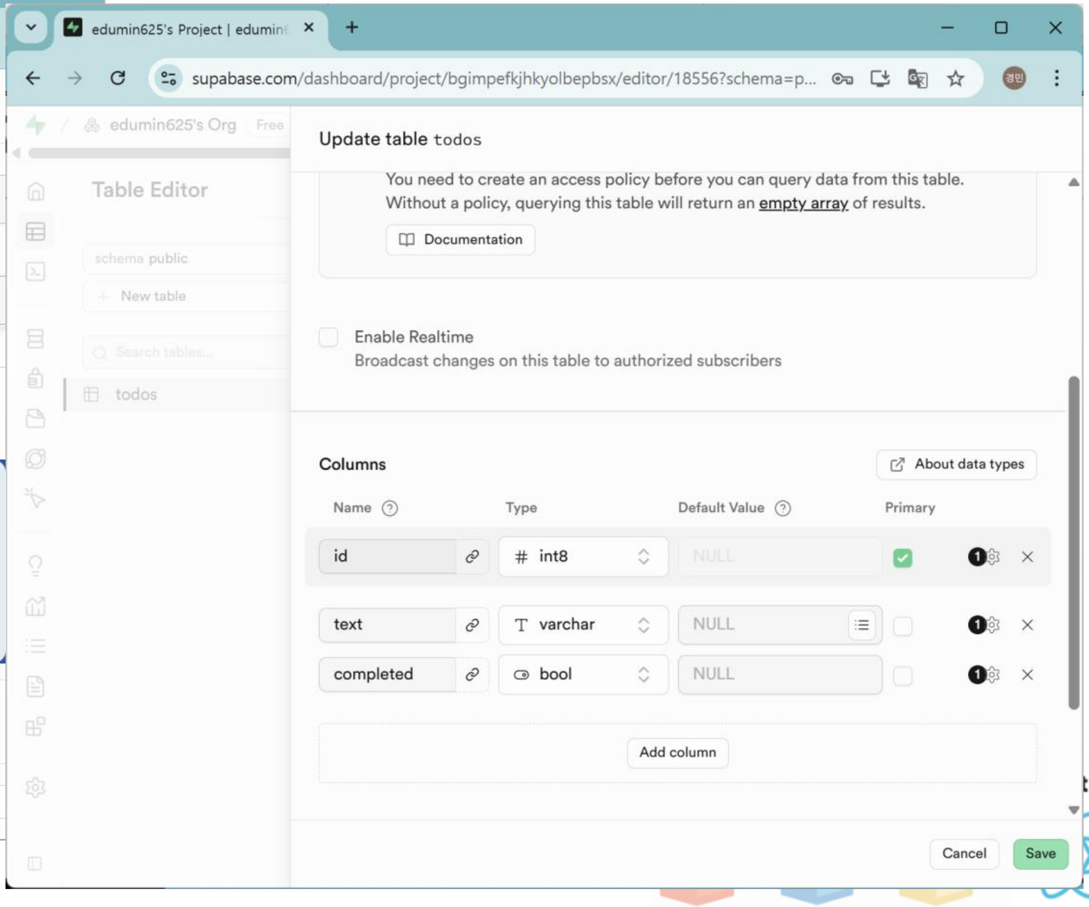
Description: 할일 목록

Enable Row Level Security (RLS) Recommended

Restrict access to your table by enabling RLS and writing Postgres policies.

Enable Realtime

Cancel


edumin625's Project | edumin

supabase.com/dashboard/project/bgimpefkjhkyolbepbsx/editor/18556?schema=p...

edumin625's Org Free

Table Editor

schema public

New table

Search tables...

todos

Update table `todos`

You need to create an access policy before you can query data from this table.
Without a policy, querying this table will return an empty array of results.

Documentation

Enable Realtime

Broadcast changes on this table to authorized subscribers

Columns

About data types

Name	Type	Default Value	Primary
id	int8	NULL	✓
text	varchar	NULL	
completed	bool	NULL	

Add column

Cancel Save

RLS(Row Level Security) 설정

- 테이블 생성 시 Row Level Security를 활성화할지 여부를 선택
- RLS는 사용자 인증 정보(JWT)를 기반으로 데이터 접근 권한을 제어하는 보안 기능

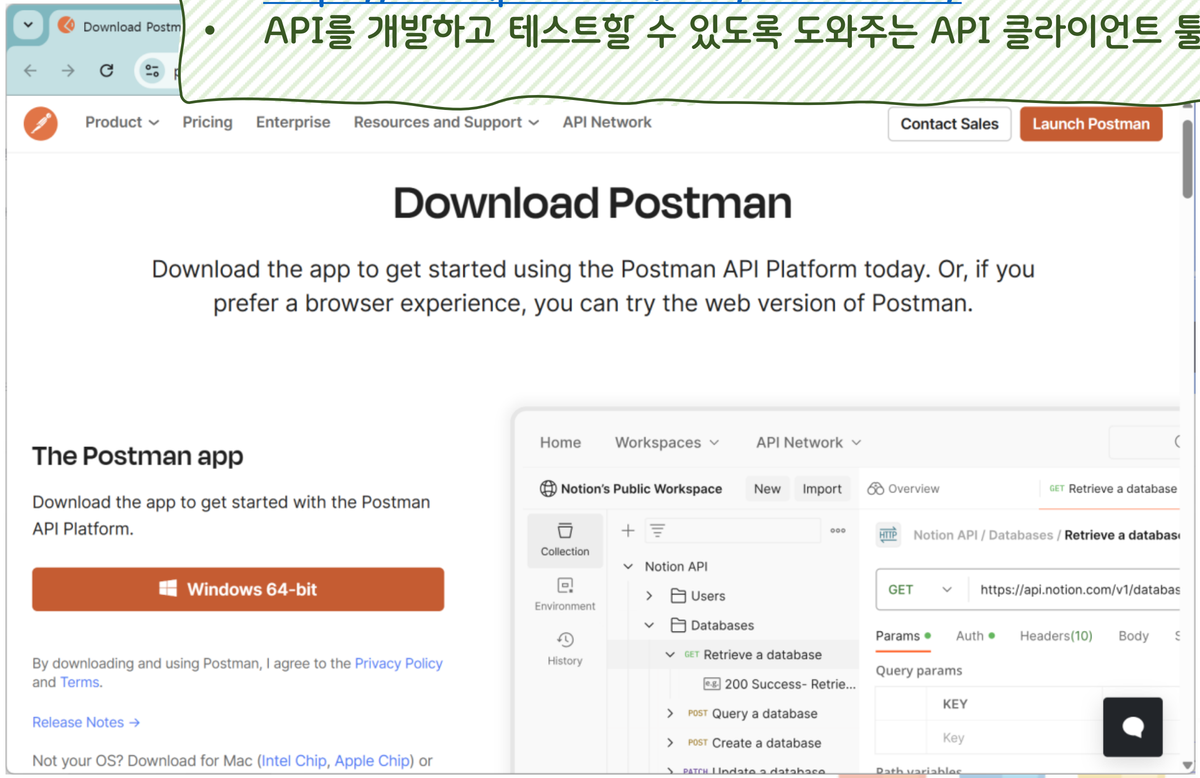
RESTful API

RESTful API

REST(Representational State Transfer)의 원칙을 엄격하게 따르며, 웹에서 자원을 URI로 식별하고 HTTP 메서드(GET, POST, PUT, DELETE 등)를 통해 자원에 대해 CRUD 작업을 수행하는 방식

Postman 설치

- https://www.postman.com/downloads/
- API를 개발하고 테스트할 수 있도록 도와주는 API 클라이언트 툴



Download Postman

Download the app to get started using the Postman API Platform today. Or, if you prefer a browser experience, you can try the web version of Postman.

The Postman app

Download the app to get started with the Postman API Platform.

Windows 64-bit

By downloading and using Postman, I agree to the Privacy Policy and Terms.

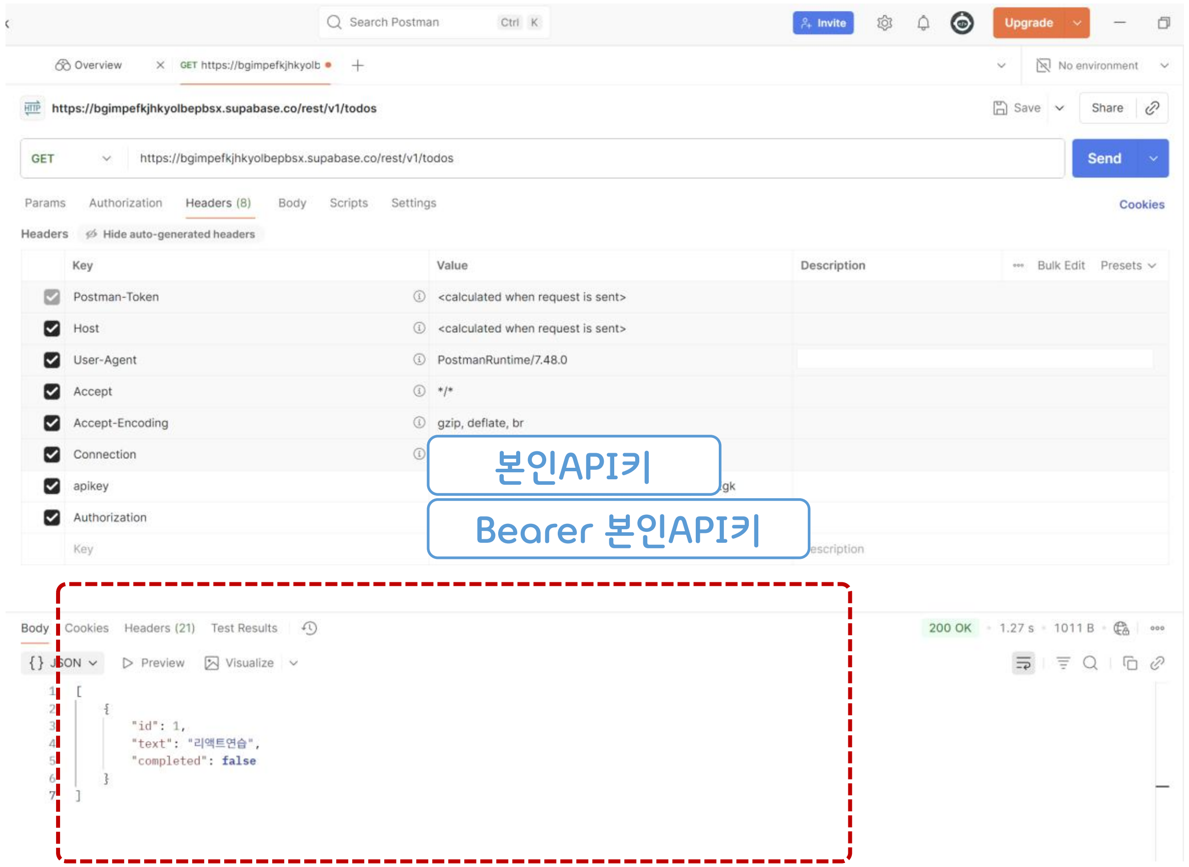
Release Notes →

Not your OS? Download for Mac (Intel Chip, Apple Chip) or

Supabase RESTful API

```
https://<project-ref>.supabase.co/rest/v1/<table_name>
```

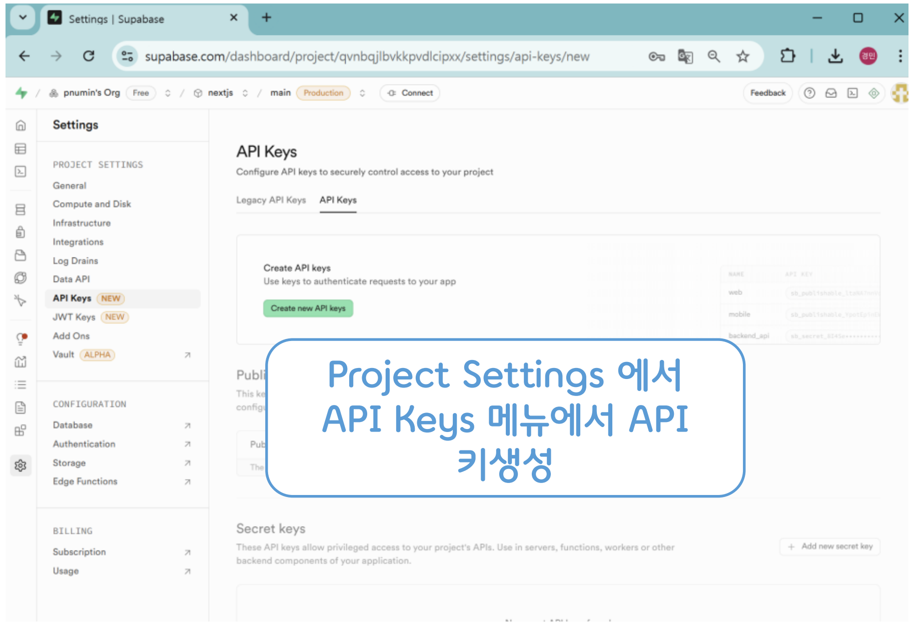
Supabase는 데이터베이스 스키마를 자동으로
탐색해 즉시 사용 가능한 RESTful API를 자동 생성



GET https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos

	Key	Value	Description
✓	Postman-Token	<calculated when request is sent>	
✓	Host	<calculated when request is sent>	
✓	User-Agent	PostmanRuntime/7.48.0	
✓	Accept	*/*	
✓	Accept-Encoding	gzip, deflate, br	
✓	Connection		
✓	apikey	본인API키	
✓	Authorization	Bearer 본인API키	

```
[
    {
        "id": 1,
        "text": "리액트연습",
        "completed": false
    }
]
```



API Keys

Configure API keys to securely control access to your project

Create API keys

Use keys to authenticate requests to your app

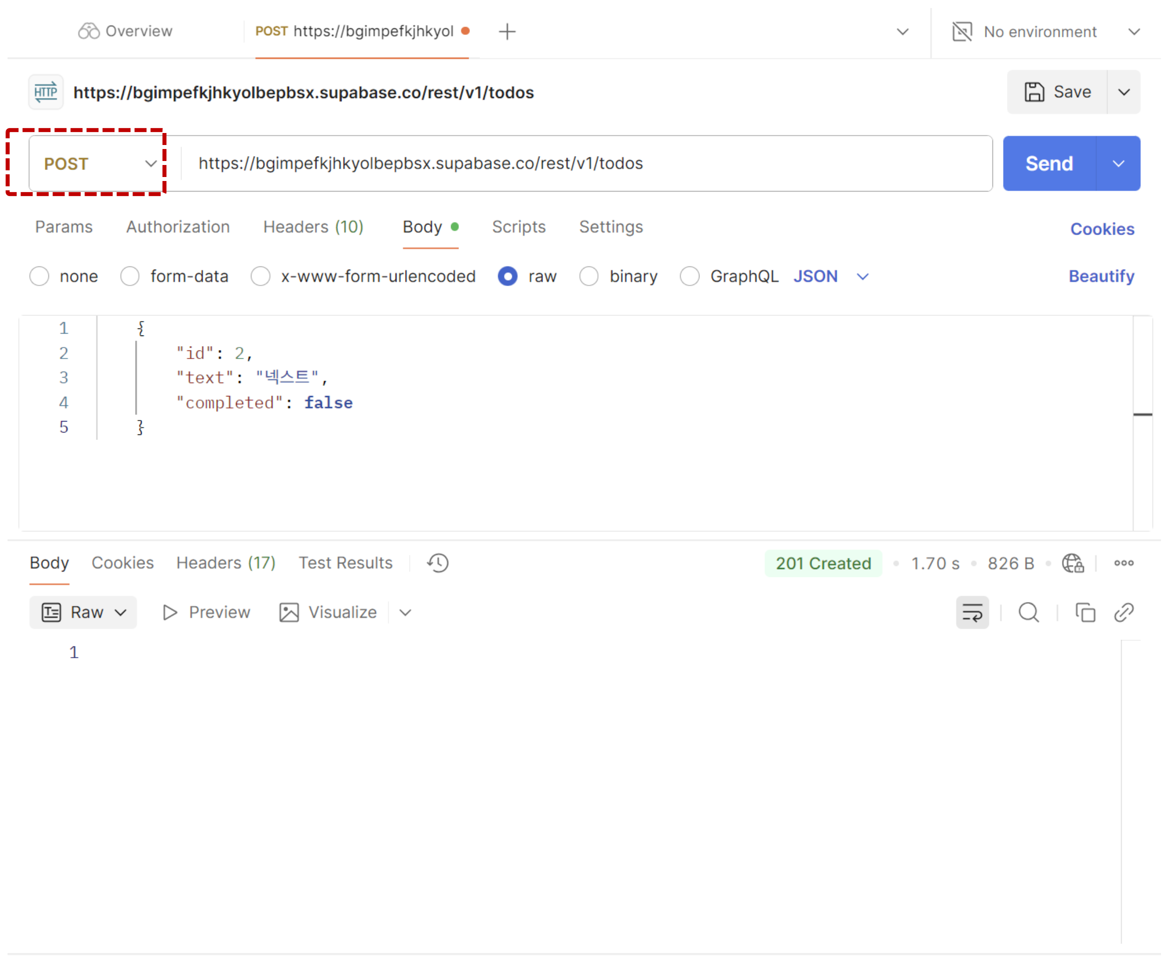
Secret keys

These API keys allow privileged access to your project's APIs. Use in servers, functions, workers or other backend components of your application.

Project Settings 에서
API Keys 메뉴에서 API
키생성

HTML CSS JavaScript React JS

Supabase RESTful API CRUD



https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos

POST https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos

none | form-data | x-www-form-urlencoded | [✓ raw] | binary | GraphQL | JSON

```
{
    "id": 2,
    "text": "넥스트",
    "completed": false
}
```

201 Created • 1.70 s • 826 B

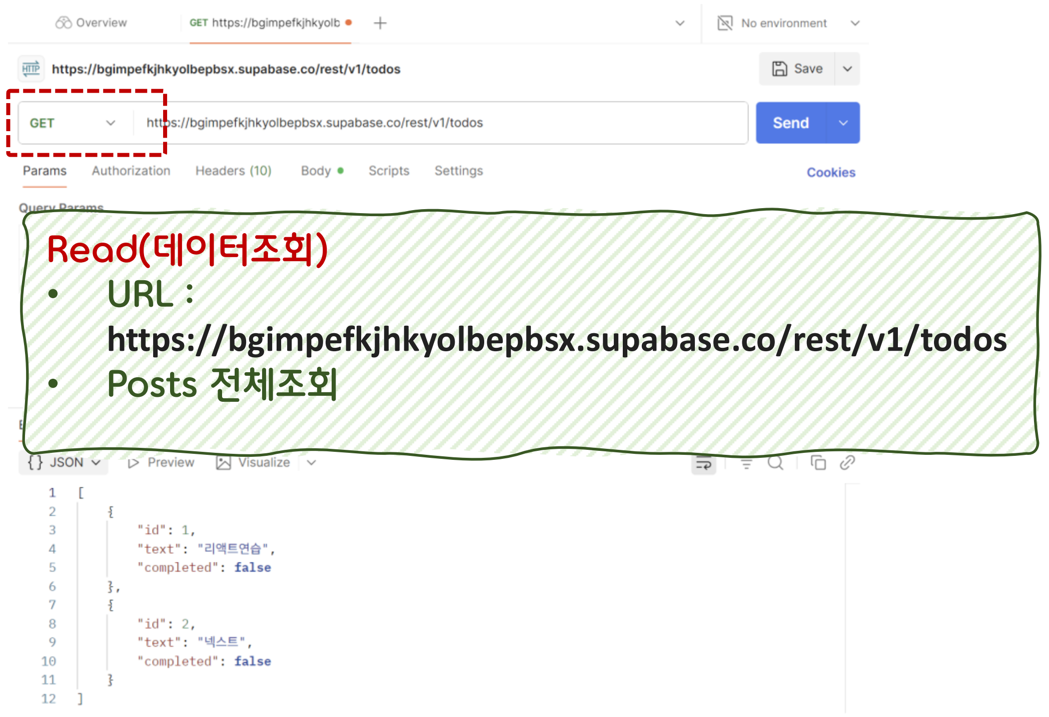
```
1
```

Create(데이터 생성)

- URL : https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos
- Method : POST
- Body : raw + json

```
{
"id": 2,
"text": "넥스트",
"completed": false
}
```

Supabase RESTful API CRUD



Overview | GET https://bgimpefkjhkyolb

No environment

https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos

Save

GET https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos Send

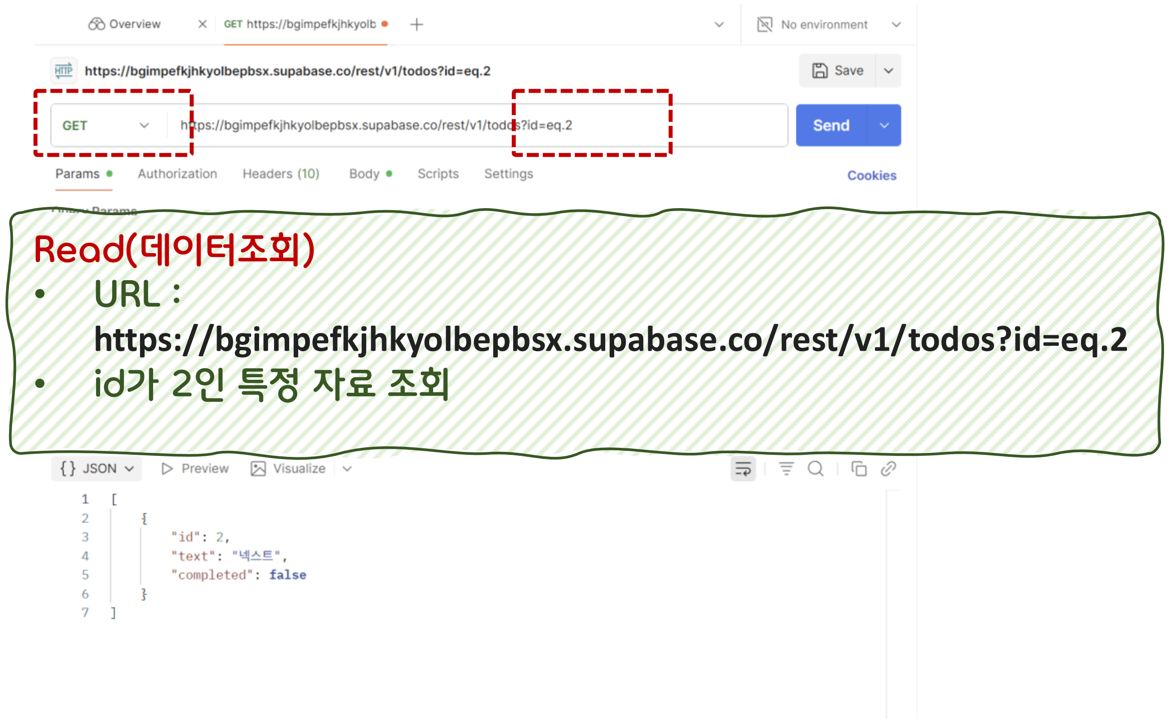
Params | Authorization | Headers (10) | Body | Scripts | Settings | Cookies

Query Params

Read(데이터조회)
- URL : https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos
- Posts 전체조회

{} JSON | Preview | Visualize

```
[
    {
        "id": 1,
        "text": "리액트연습",
        "completed": false
    },
    {
        "id": 2,
        "text": "넥스트",
        "completed": false
    }
]
```



Overview | GET https://bgimpefkjhkyolb

No environment

https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos?id=eq.2

Save

GET https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos?id=eq.2 Send

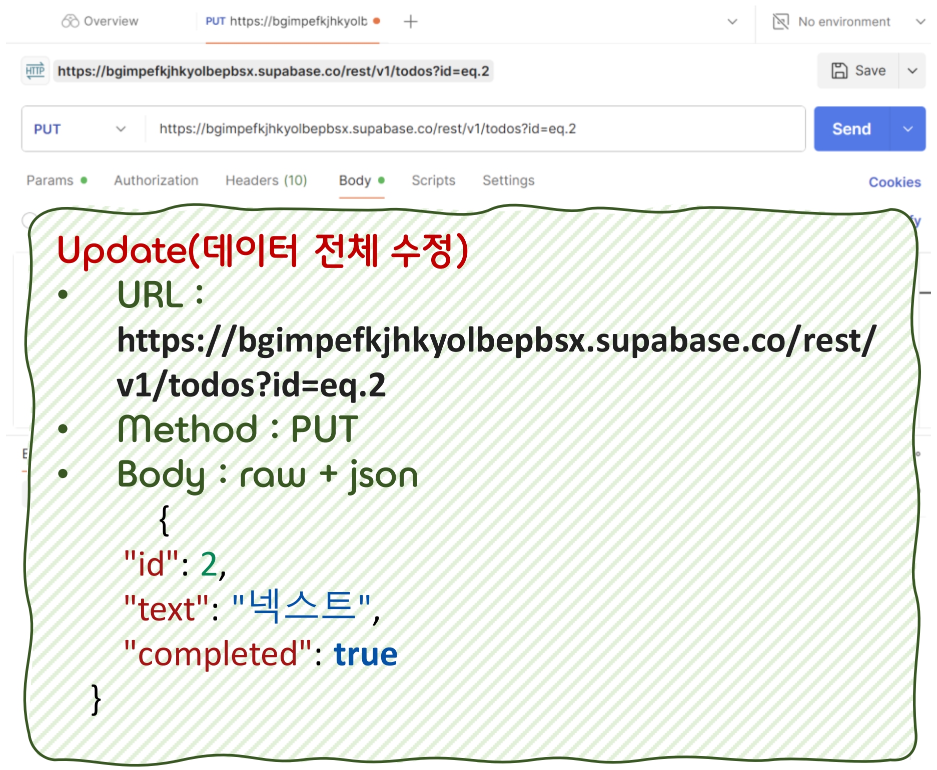
Params | Authorization | Headers (10) | Body | Scripts | Settings | Cookies

Read(데이터조회)
- URL : https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos?id=eq.2
- id가 2인 특정 자료 조회

{} JSON | Preview | Visualize

```
[
    {
        "id": 2,
        "text": "넥스트",
        "completed": false
    }
]
```

Supabase RESTful API CRUD

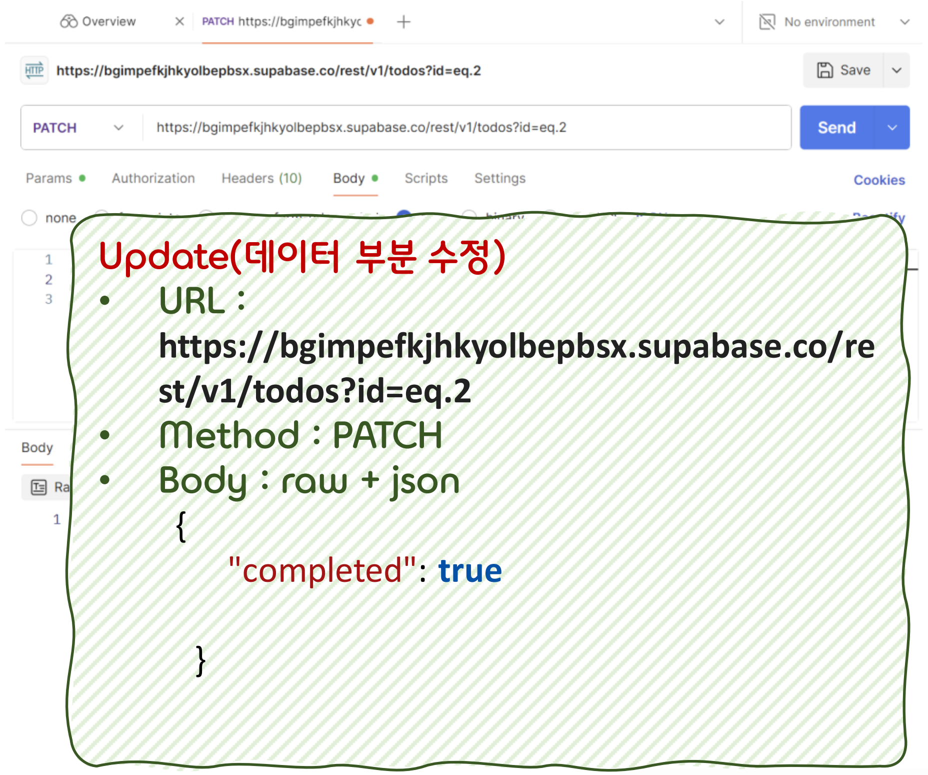


PUT https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos?id=eq.2

Update(데이터 전체 수정)

- URL : **https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos?id=eq.2**
- Method : PUT
- Body : raw + json

```
{
"id": 2,
"text": "넥스트",
"completed": true
}
```



PATCH https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos?id=eq.2

Update(데이터 부분 수정)

- URL : **https://bgimpefkjhkyolbepbsx.supabase.co/rest/v1/todos?id=eq.2**
- Method : PATCH
- Body : raw + json

```
{
    "completed": true

}
```

HTML CSS JavaScript React JS

Supabase Fetch 데이터 가져오기

```
const getTodos = async () => {
  const resp = await fetch(`${supabaseUrl}/rest/v1/todos?select=*&order=id.desc`, {
    method: 'GET',
    headers: {
      'apikey': supabaseKey,
      'Authorization': `Bearer ${supabaseKey}`,
    }
  });

  if (resp.ok) {
    const data = await resp.json();
    setTodos(data);
  } else {
    console.error('Error fetching todos:', resp.statusText);
    setTodos([]);
  }
}
```

HTML CSS JavaScript React JS

Supabase Fetch 데이터 저장하기

```
const handleAdd = async () => {
  if ( inRef.current.value == "") {
    alert("값을 입력해 주세요.");
    inRef.current.focus() ;
    return
  }

  const response = await fetch(`${supabaseUrl}/rest/v1/todos`, {
    method: 'POST',
    headers: {
      'apikey': supabaseKey,
      'Authorization': `Bearer ${supabaseKey}`,
      'Content-Type': 'application/json'
    },
    body: JSON.stringify({ text: inRef.current.value, completed: false })
  });

  if (response.ok) {
    getTodos();
    inRef.current.value = "" ;
    inRef.current.focus() ;
  } else {
    console.error('Error adding todo:', response.statusText);
  }
}
```

HTML CSS JavaScript React JS

Supabase Fetch 데이터 수정하기

```
 const handleToggle = async () => {
 const response = await fetch(`${supabaseUrl}/rest/v1/todos?id=eq.${todo.id}`, {
   method: 'PATCH',
   headers: {
     'apikey': supabaseKey,
     'Authorization': `Bearer ${supabaseKey}`,
     'Content-Type': 'application/json'
   },
   body: JSON.stringify({ completed: !todo.completed })
 });

 if (response.ok) {
   getTodos();
 } else {
   console.error('Error toggling todo:', response.statusText);
 }
}
```

HTML CSS JavaScript React JS

Supabase Fetch 데이터 삭제하기

```
const handleDelete = async () => {
  const response = await fetch(`${supabaseUrl}/rest/v1/todos?id=eq.${todo.id}`, {
    method: 'DELETE',
    headers: {
      'apikey': supabaseKey,
      'Authorization': `Bearer ${supabaseKey}`
    }
  });

  if (response.ok) {
    getTodos();
  } else {
    console.error('Error deleting todo:', response.statusText);
  }
}
```

Supabase 라이브러리 설치

npm install @supabase/supabase-js

- 라이브러리를 사용하여 Supabase와 통신하도록 설정

1. 환경변수 추가

```
VITE_SUPABASE_URL = “https://<project-ref>.supabase.co”
VITE_SUPABASE_KEY = 본인API키
```

2. src/supabase/client.js 만들기

```
import { createClient } from '@supabase/supabase-js’

export const supabase = createClient(
  import.meta.env.VITE_SUPABASE_URL,
  import.meta.env.VITE_SUPABASE_KEY
);
```

HTML CSS JavaScript React JS

Supabase 라이브러리로 가져오기

```
import { supabase } from "../supabase/client";

const getTodos = async () => {
    const { data, error } = await supabase
      .from('todos')
      .select('*')
      .order('id', { ascending: false });

    if (error) {
      console.error('Error fetching todos:', error);
    } else {
      setTodos(data);
    }
  }
```

HTML CSS JavaScript React JS

Supabase 라이브러리로 저장하기

```
import { supabase } from "../supabase/client";

const handleAdd = async () => {
…

    const { data, error } = await supabase
      .from('todos')
      .insert([
        { text: inRef.current.value, completed: false },
      ]);

    if (error) {
      console.error('Error adding todo:', error);
    } else {
      getTodos();
      inRef.current.value = "" ;
      inRef.current.focus() ;
    }
  }
```

HTML CSS JavaScript React JS

Supabase 라이브러리로 수정하기

```
const handleToggle = async () => {
  const { error } = await supabase
    .from('todos')
    .update({ completed: !todo.completed })
    .eq('id', todo.id);

  if (error) {
    console.error('Error toggling todo:', error);
  } else {
    getTodos();
  }
}
```

HTML CSS JavaScript React JS

Supabase 라이브러리로 삭제하기

```
const handleDelete = async () => {
  const { error } = await supabase
    .from('todos')
    .delete()
    .eq('id', todo.id);

  if (error) {
    console.error('Error deleting todo:', error);
  } else {
    getTodos();
  }
}
```

HTML CSS JavaScript React JS