

Express활용 웹페이지 구현 따라하기

- 웹 프로그래밍 응용 -

김민재(2020542005)
최지훈(2021660049)
한세윤(2020675071)
김규민(2020675008)

목 차

01 프로젝트 가져오기

- Github에서 프로젝트 클론
- 필요 모듈 설치
- Npm run dev 실행

02 백엔드 서버 설정

- 서버용 .env파일 구성
- JWT키 발급 및 삽입

03 OpenAI 연동 설정

- OpenAI API Key 생성
- Billing설정
- AI기능을 위한 초기 설정 진행

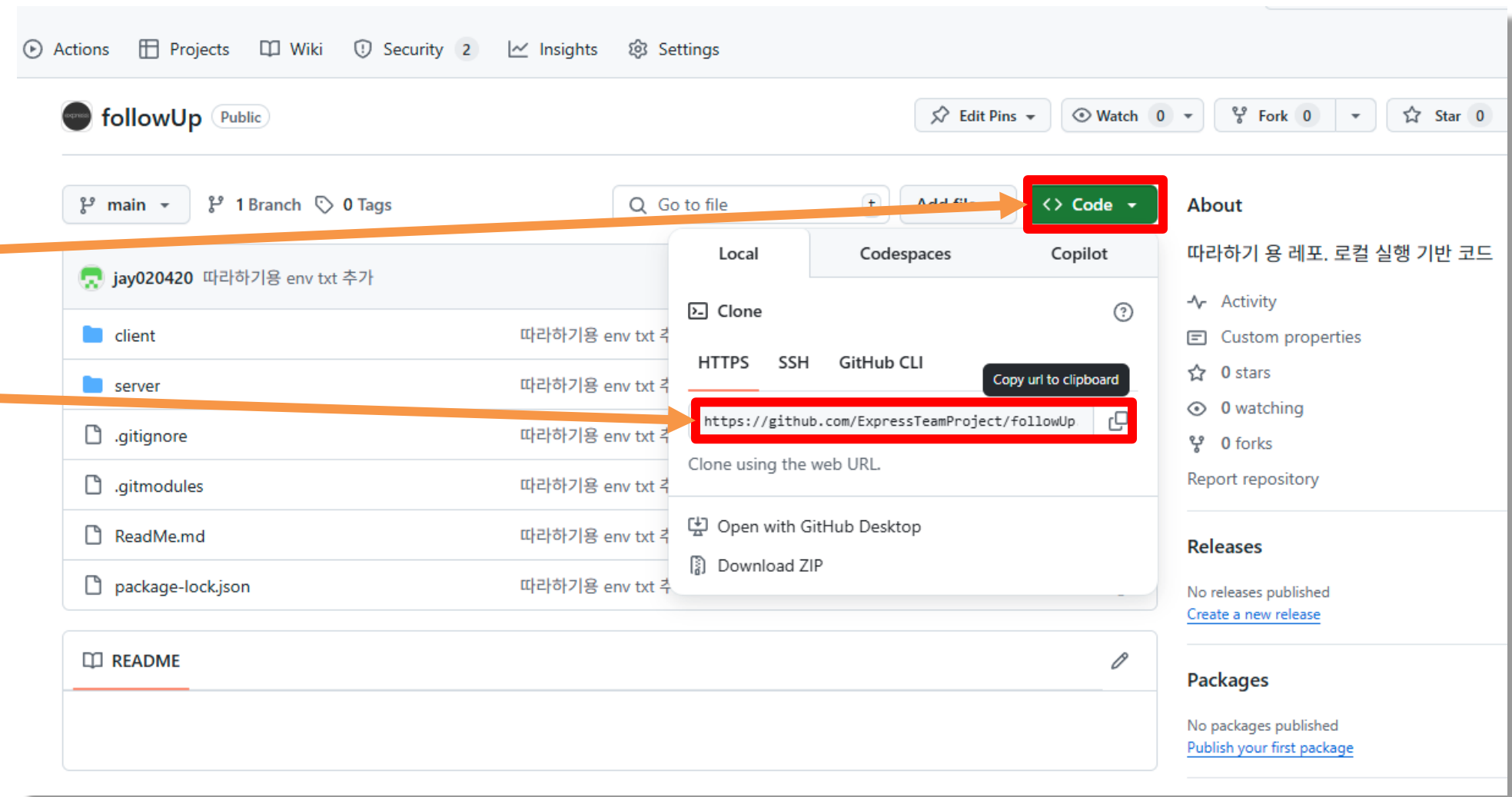
04 백엔드 서버 실행 및 테스트 확인

- 서버 실행 및 동작 확인

프로젝트 가져오기

[접속]

- 깃허브 사이트 접속
- 초록색 버튼(Code 클릭)
- URL 복사

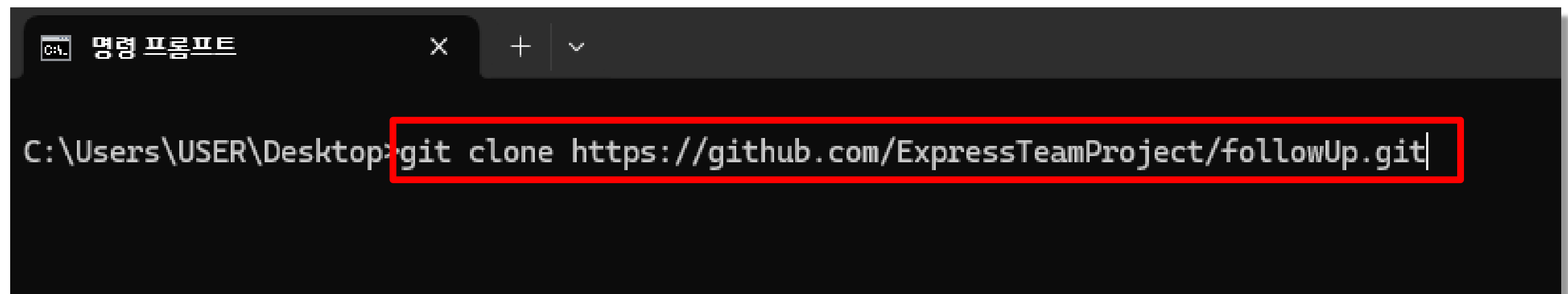
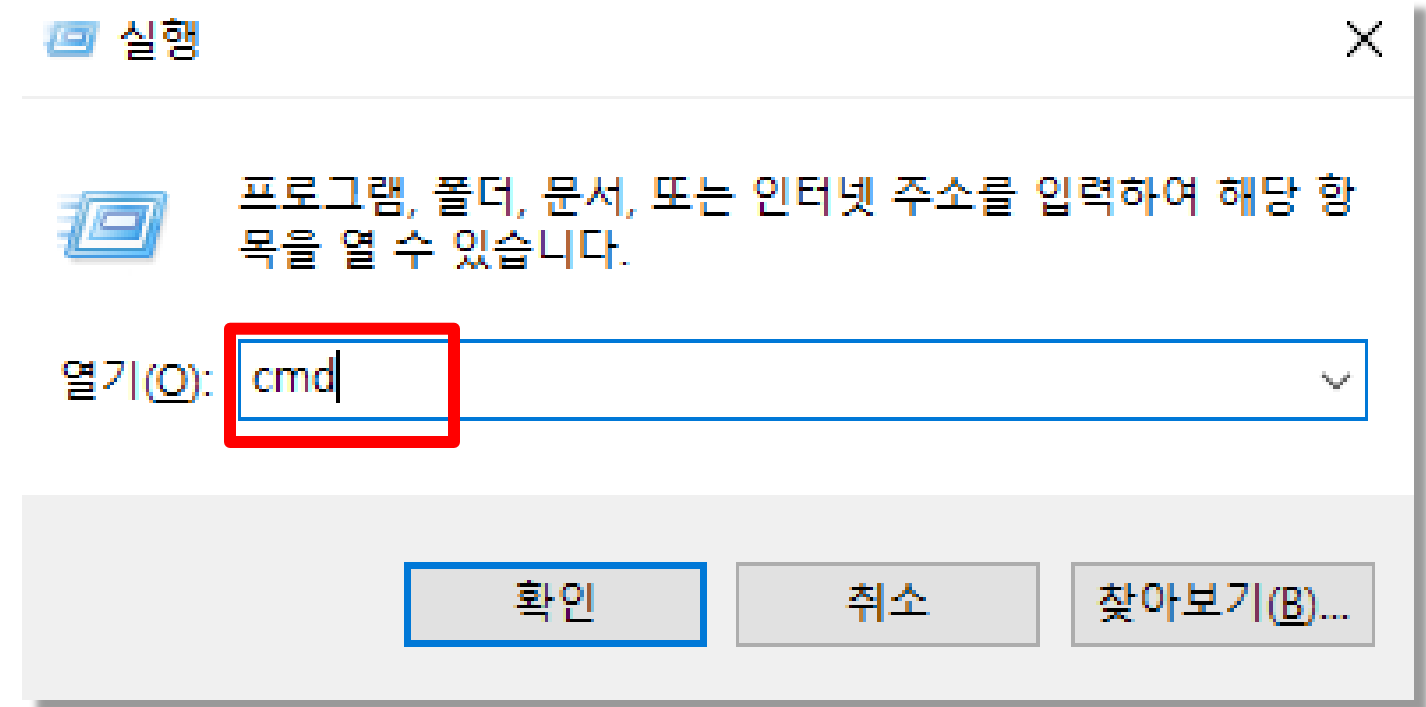


- <https://github.com/ExpressTeamProject/followUp>

프로젝트 가져오기

[CMD 접속]

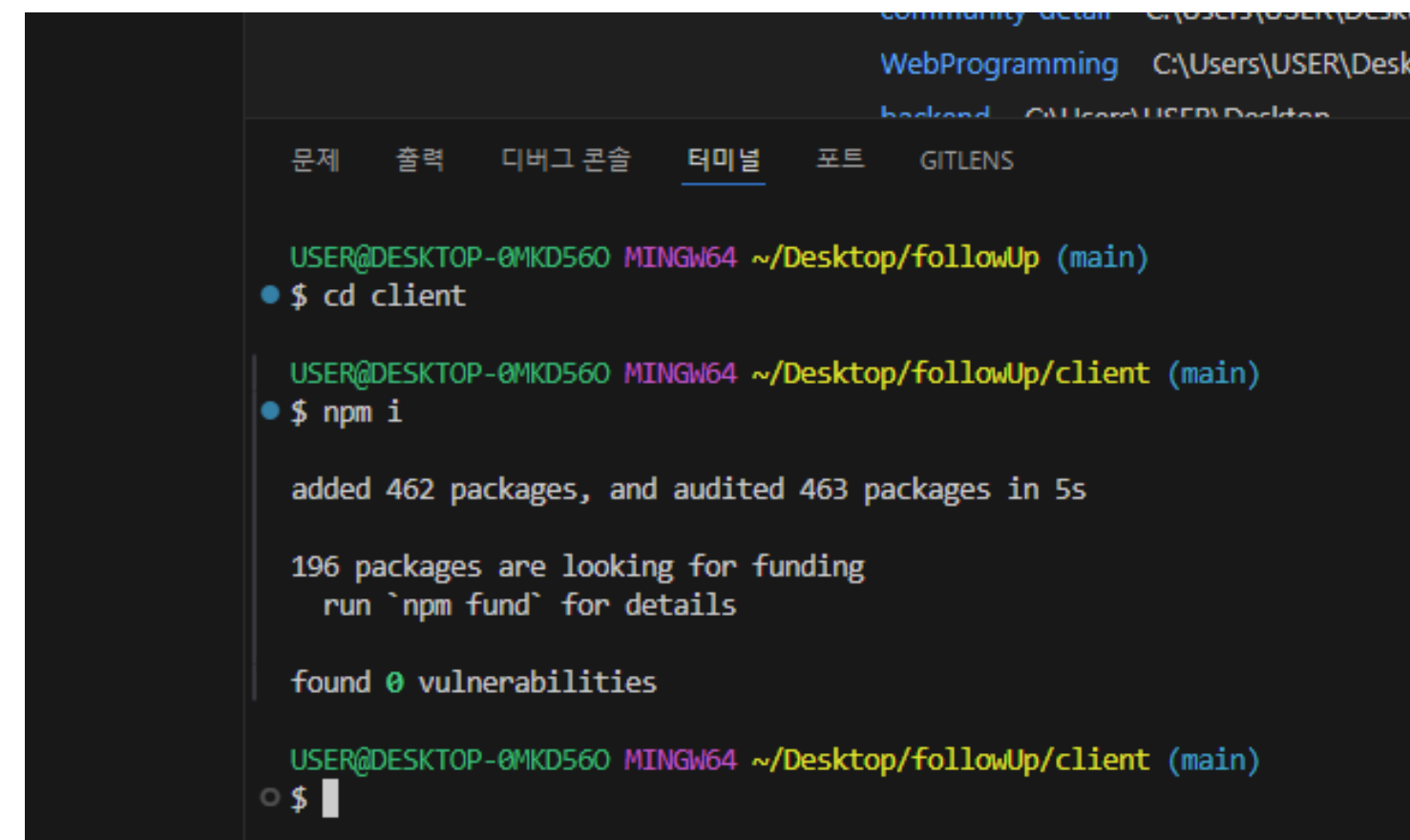
- 윈도우 +R을 누른 후 cmd 검색 및 실행
- 원하는 경로에 복사한 url 붙여넣기 -> git clone url



프로젝트 가져오기

[로컬 환경 설정]

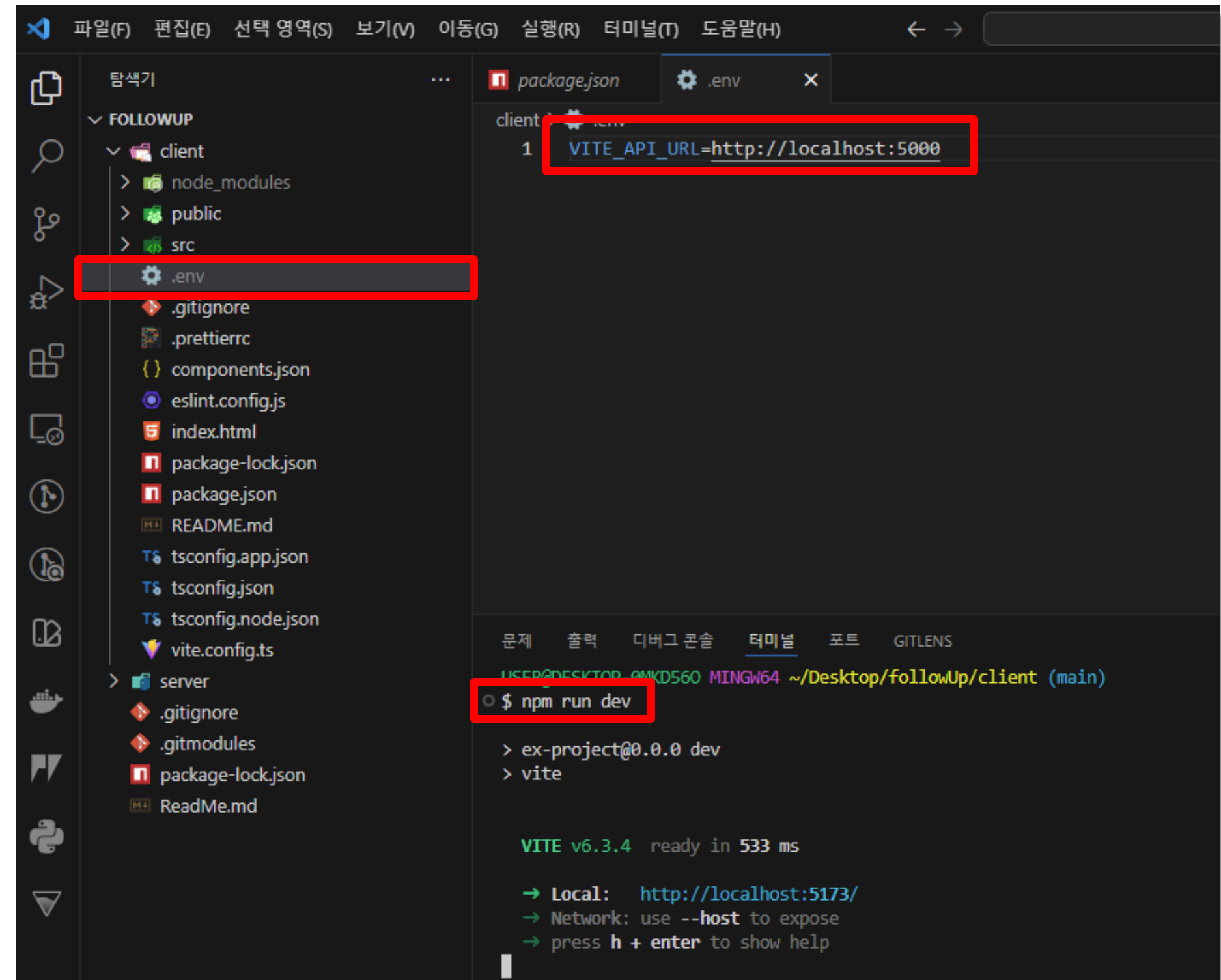
- 자신이 사용하는 IDE 열기(VSCode 등)
- [Ctrl + Shift + `]으로 터미널 열기
- VSCode를 사용 중이라면 '+'오른편의 아래 화살표 버튼을 눌러 Git Bash 클릭
- cd client입력
- npm i 혹은, npm install 입력(의존성 모듈 설치)
-> 설치 후 node_module파일이 설치 되었는지 여부 확인



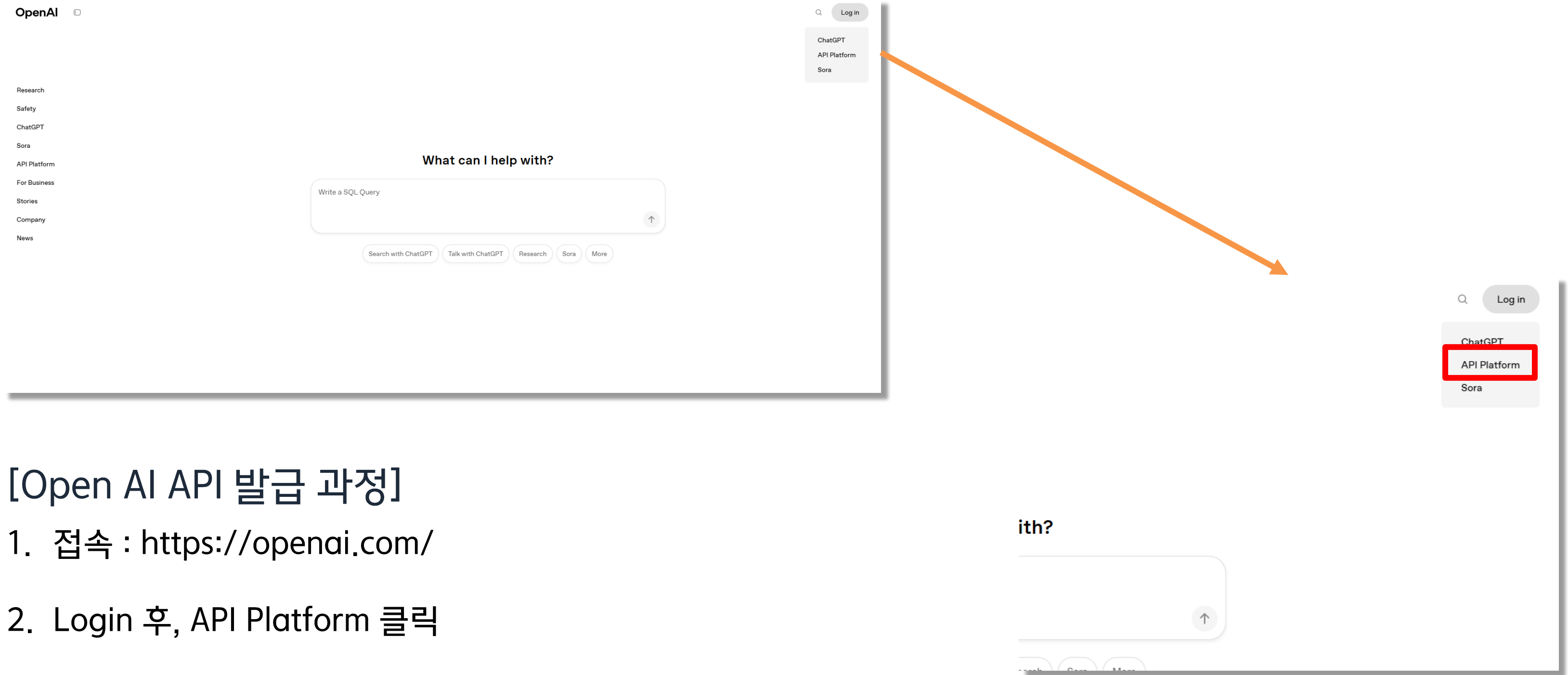
프로젝트 가져오기

[로컬 환경 설정]

- client/.env에
VITE_API_URL = <http://localhost:5000> 입력
- 터미널에서 npm run dev 입력(개발모드)
- Localhost:5173에 접속



OpenAI 연동 설정



[Open AI API 발급 과정]

1. 접속 : <https://openai.com/>
2. Login 후, API Platform 클릭

OpenAI 연동 설정

Personal / Default project

Playground Dashboard Docs API reference 김

SETTINGS

Your profile

ORGANIZATION

General

API keys

Admin keys

Members

Projects

Billing

Limits

Usage

Data controls

PROJECT

General

API keys

As an owner of this organization, you can view and manage all API keys in this organization.

Do not share your API key with others or expose it in the browser or other client-side code. To protect your account's security, OpenAI may automatically disable any API key that has leaked publicly.

View usage per API key on the [Usage page](#).

NAME	SECRET KEY	CREATED	LAST USED	PROJECT ACCESS	CREATED BY	PERMISSIONS
EXPRESS_KEY	sk-...a8IA	2025년 5월 13일	2025년 5월 19일	Default project	김 규민	All
MY_DIARY_KEY2	sk-...K74A	2024년 11월 17일	2024년 11월 18일	Default project	김 규민	All
MY_DIARY_KEY	sk-...dRgA	2024년 11월 17일	2024년 11월 17일	Default project	김 규민	All

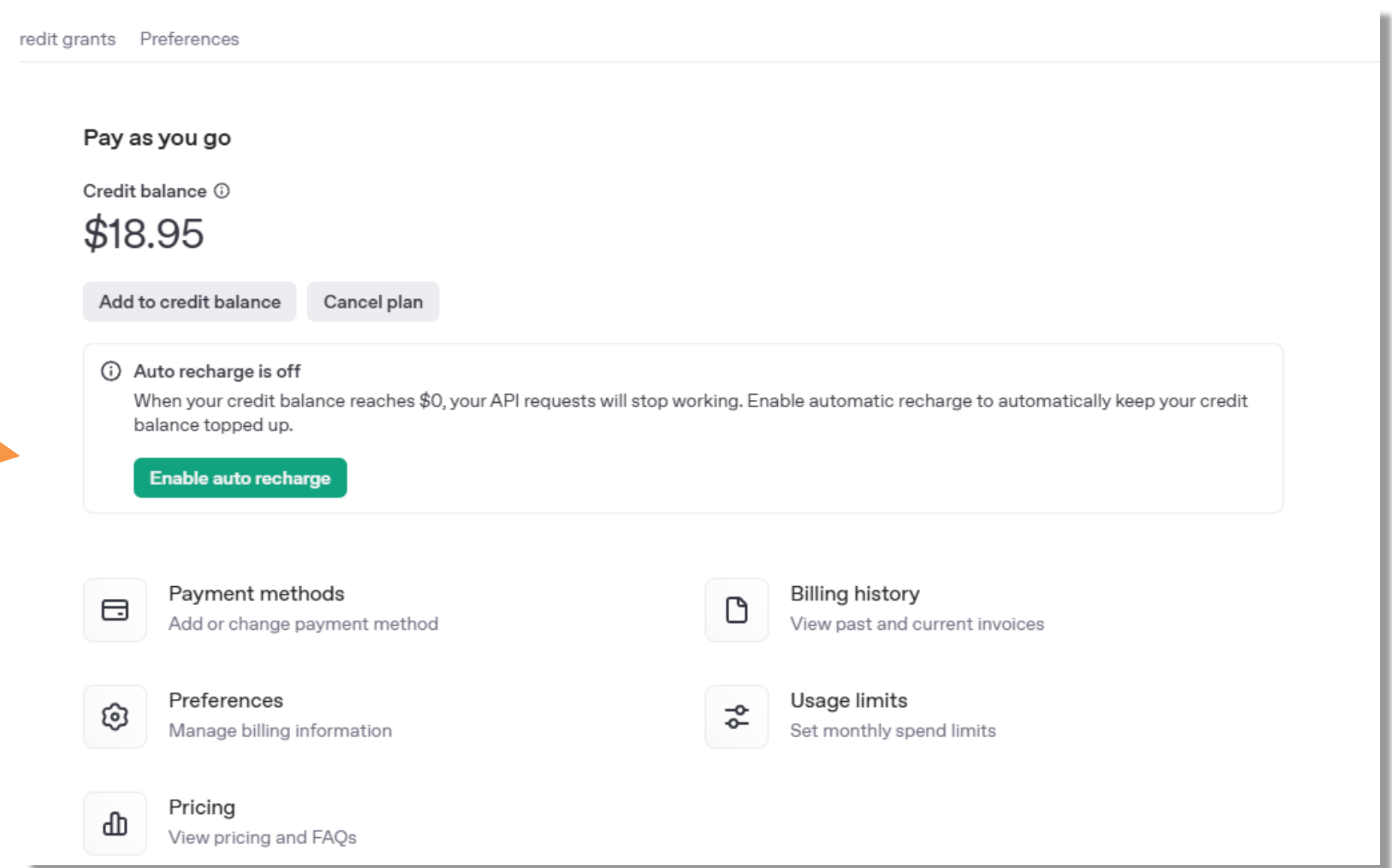
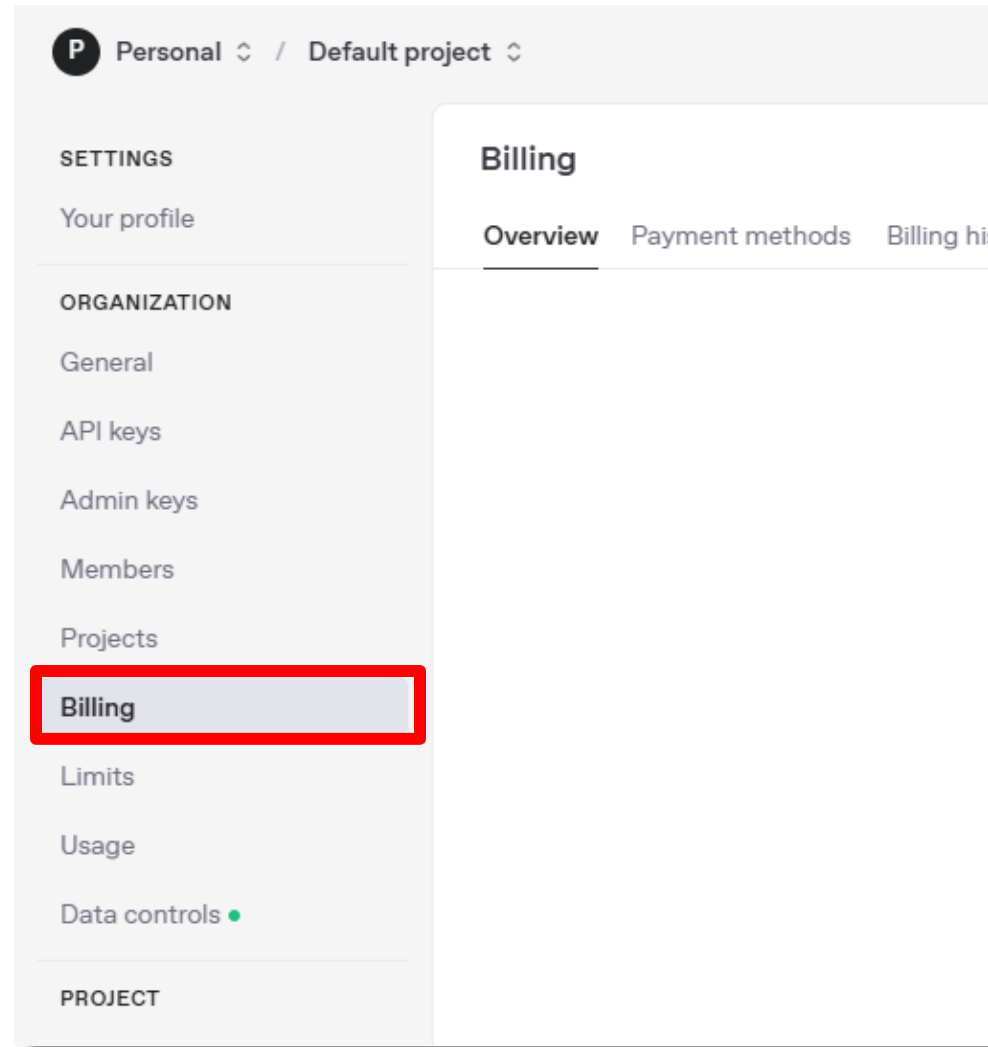
[+ Create new secret key](#)

[Open AI API 발급 과정]

3. 우측 상단 톱니바퀴 클릭

4. 좌측 네비게이션바 API Keys 클릭 후 Create new secret key 클릭

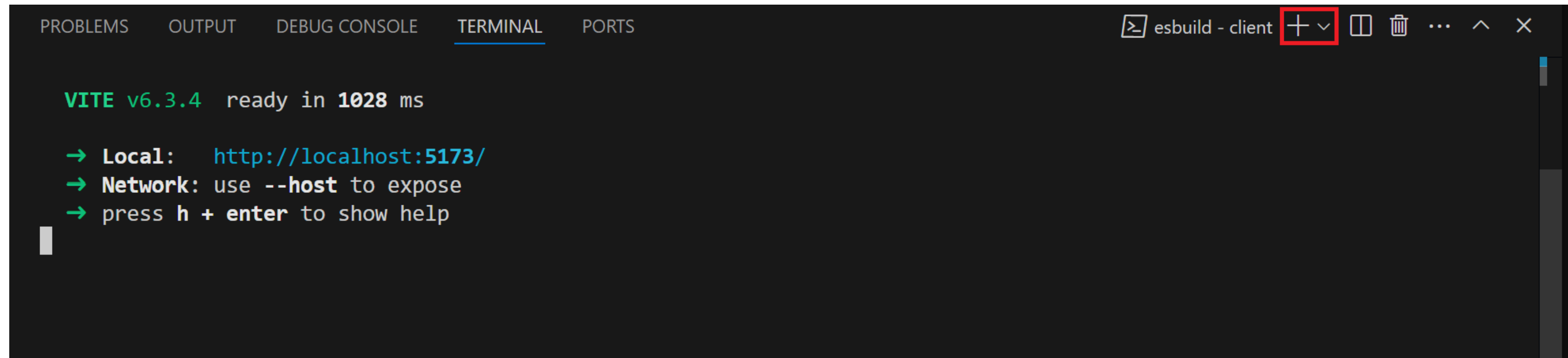
OpenAI 연동 설정



[Open AI API 발급 과정]

5. Billing 클릭 후 Pay 추가

백엔드 서버 설정



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS esbuild - client + - [ ] [X] ... ^ X
```

```
VITE v6.3.4 ready in 1028 ms
```

```
→ Local: http://localhost:5173/
```

```
→ Network: use --host to expose
```

```
→ press h + enter to show help
```

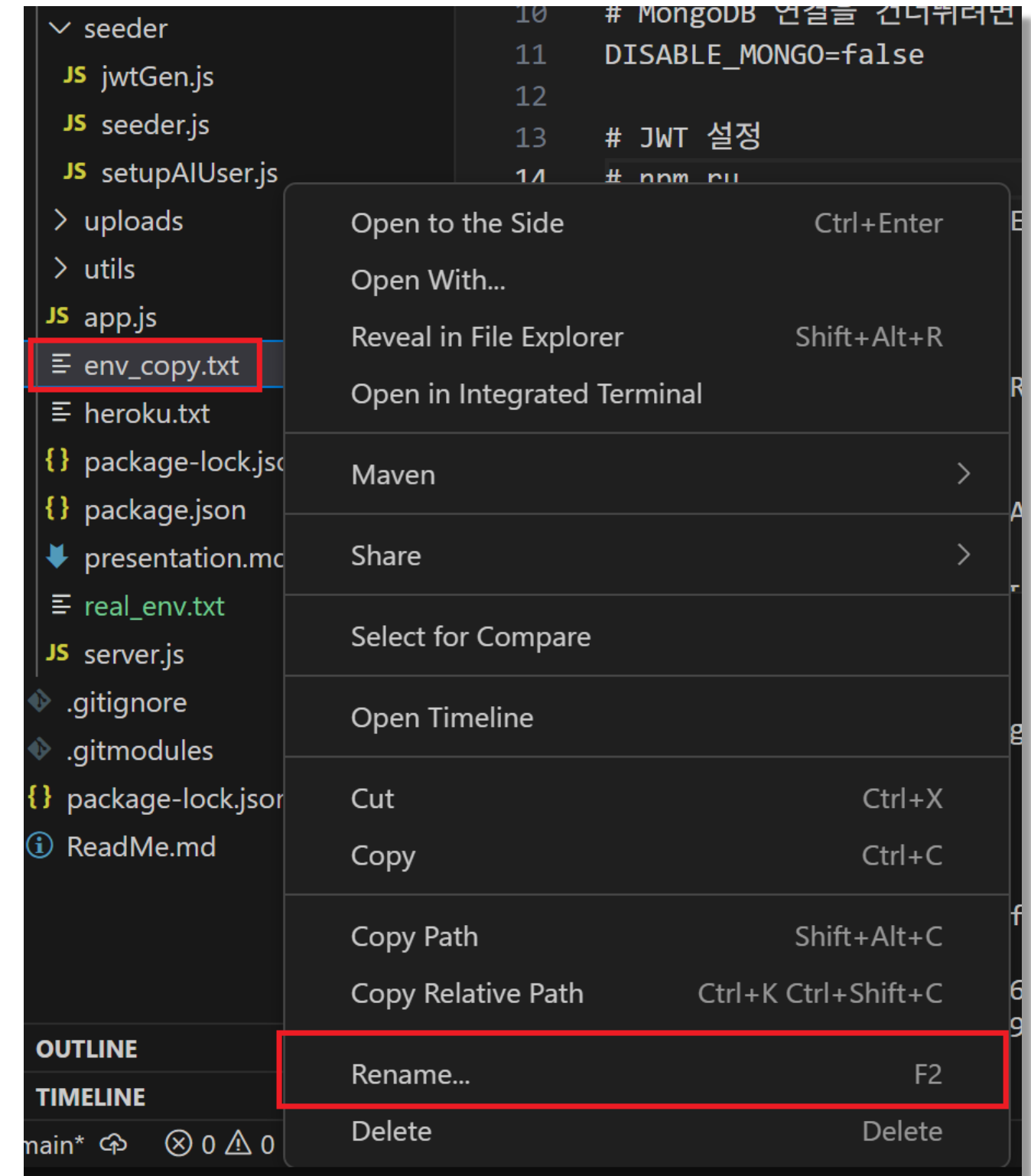
[새로운 터미널 추가]

- 기존 프론트는 계속 실행 중일 수 있도록 터미널을 추가 생성

백엔드 서버 설정

[파일명 수정]

- Server/env_copy.txt를 우클릭 하여
Rename -> '.env' 파일 명으로 수정



백엔드 서버 설정

[MongoDB Atlas URI 생성]

- Atlas에 계성 생성 및 등록
- 계정 생성시에 입력하는 'User_name'과 'Password'는 숙지 및 메모
- <https://www.mongodb.com/ko-kr/cloud/atlas/register>

MongoDB.

MongoDB Atlas

로그인

무료로 Atlas 기능 사용해보기

구글 계정으로 등록

이름*

성*

회사

이메일*

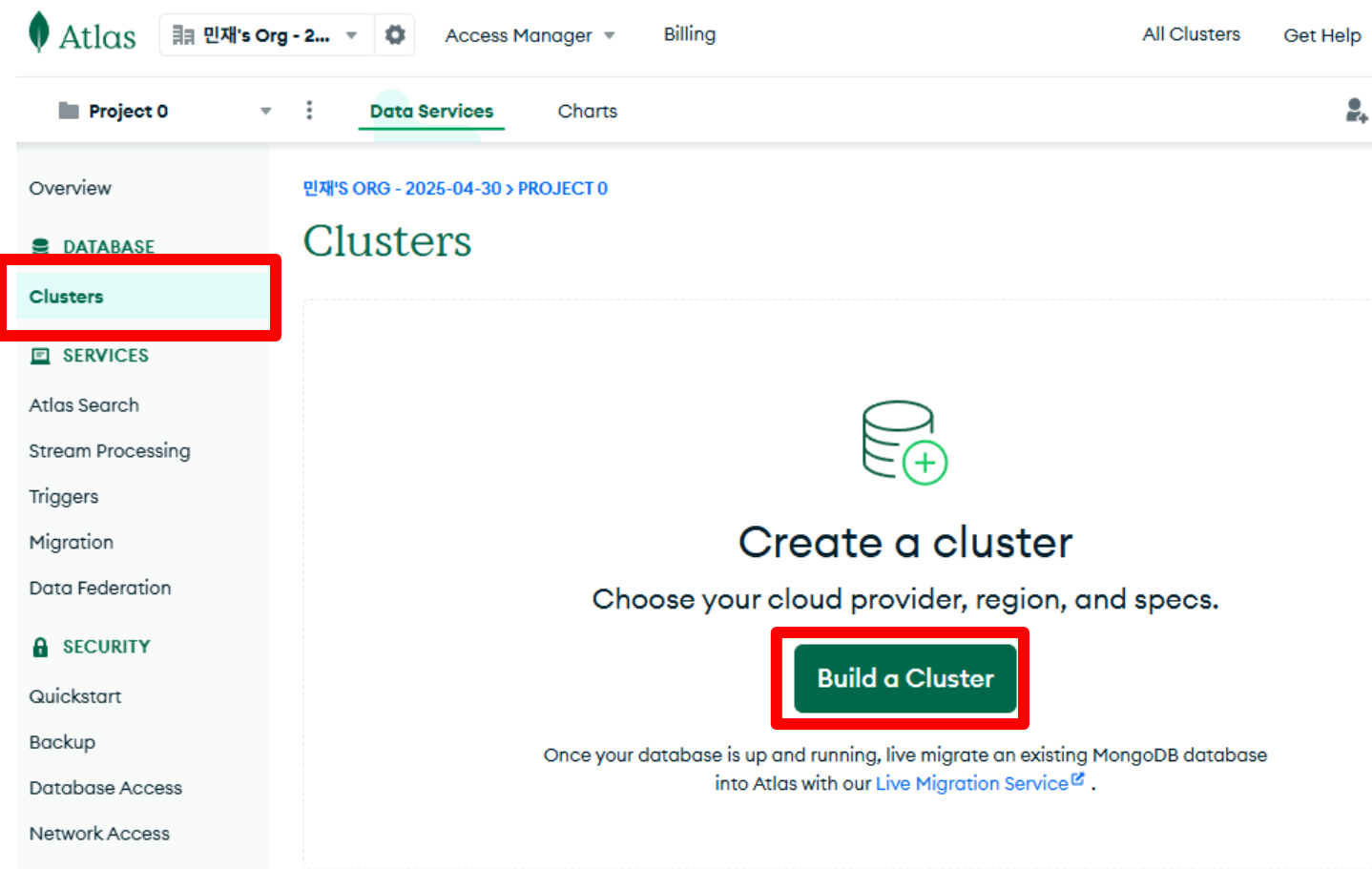
비밀번호* ☐

✓ 코드로 데이터 작업
MongoDB의 문서들을 해당 프로그래밍 언어의 오브젝트로 직접 매핑하십시오. 해당 앱이 일정 기간이 경과한 경우 스키마를 수정하십시오.

✓ 관리가 아닌 개발에 집중하십시오.
상시보안부터 시점복구(PITR, Point-In-Time-Recovery)까지 확장 기능에 필요한 인프라 운영을 MongoDB Atlas에 맡기십시오.

✓ 사용자 데이터 독립성을 단순화하십시오.
싱글 API 및 최소한의 데이터 전송으로 풀텍스트 검색, 실시간 분석, 다양한 시각화 및 기타 기능용 애플리케이션 데이터를 활용하십시오.

백엔드 서버 설정



Deploy your cluster

Use a template below or set up advanced configuration options. You can also edit these configuration options once the cluster is created.

Cluster Type	Price	Description	STORAGE	RAM	vCPU
M10	\$0.10/hour	Dedicated cluster for development environments and low-traffic applications.	10 GB	2 GB	2 vCPUs
Flex	From \$0.011/hour Up to \$30/month	For application development and testing, with on-demand burst capacity for unpredictable traffic.	5 GB	Shared	Shared
Free		For learning and exploring MongoDB in a cloud environment.	512 MB	Shared	Shared

✓ **Free forever!** Your free cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

Configurations

Name
You cannot change the name once the cluster is created.

Provider

Quick setup

- ☒ Automate security setup ⓘ
- ☒ Preload sample dataset ⓘ

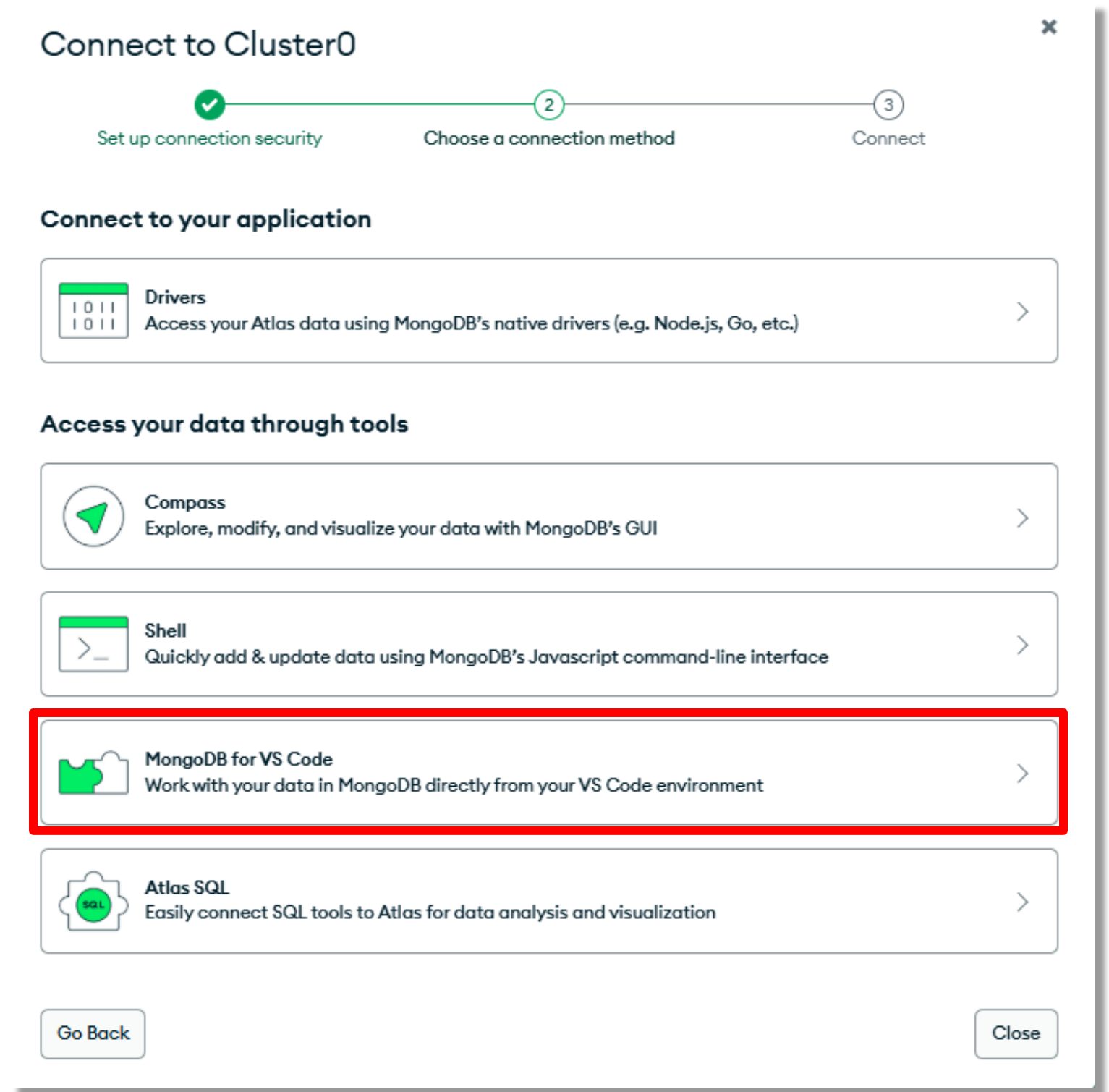
[MongoDB Atlas URI 생성]

- 좌측 메뉴의 Cluster > Build a Cluster 클릭
- 요금제 : free / Name : (자유롭게 입력) / Provider : aws 로 설정
- Create Deployment 클릭

백엔드 서버 설정

[MongoDB Atlas URI 생성]

- Conect > MongoDB for VS Code 클릭



백엔드 서버 설정

[MongoDB Atlas URI 생성]

- VSCode에서 MongoDB를 연결하기 위한 문자열 확인 및 복사
- Done 클릭

Connect to Cluster0



Connecting with MongoDB for VS Code

1. Install MongoDB for VS Code.

In [VS Code](#), open "Extensions" in the left navigation and search for "MongoDB for VS Code." Select the extension and click install.

2. In VS Code, open the Command Palette.

Click on "View" and open "Command Palette."

Search "MongoDB: Connect" on the Command Palette and click on "Connect with Connection String."

3. Connect to your MongoDB deployment.

Paste your connection string into the Command Palette.

mongodb+srv://<db_username>:<db_password>@cluster0.i9psmch.mongodb.net/

Replace [<db_password>](#) with the password for the [<db_username>](#) user. Ensure any options are [URL encoded](#).
You can edit your database user password in [Database Access](#).

4. Click "Create New Playground" in MongoDB for VS Code to get started.

[Learn more about Playgrounds](#)

RESOURCES

[Connect to MongoDB through VSCode](#)

[Explore your data with playgrounds](#)

[Access your Database Users](#)

[Troubleshoot Connections](#)

Go Back

Done

백엔드 서버 설정



Add New Database User

Create a database user to grant an application or user access to databases and collections in your clusters in this Atlas project. Granular access control can be configured with default privileges or custom roles. You can grant access to an Atlas project or organization using the corresponding [Access Manager](#).

Authentication Method

Four authentication methods are shown as buttons: Password (highlighted with a red box), Certificate, AWS IAM, and Federated Auth (MongoDB 7.0 and up).

MongoDB uses [SCRAM](#) as its default authentication method.

Password Authentication

The Password Authentication section contains two input fields: 'Enter username, e.g. new-user_31' and 'Enter password' (with a 'SHOW' toggle). Below these fields are two buttons: 'Autogenerate Secure Password' and 'Copy'.

User Description

Add an optional description to your user.

A text input field for the user description.

Database User Privileges

Configure role based access control by assigning database user a mix of one built-in role, multiple custom roles, and multiple specific privileges. A user will gain access to all actions within the roles assigned to them, not just the actions those roles share in common. **You must choose at least one role or privilege.** [Learn more about roles.](#)

The 'Built-in Role' section shows a selection box with '0 SELECTED' and an upward arrow. Below it is a button labeled 'Add Built In Role'.

[MongoDB Database Access설정]

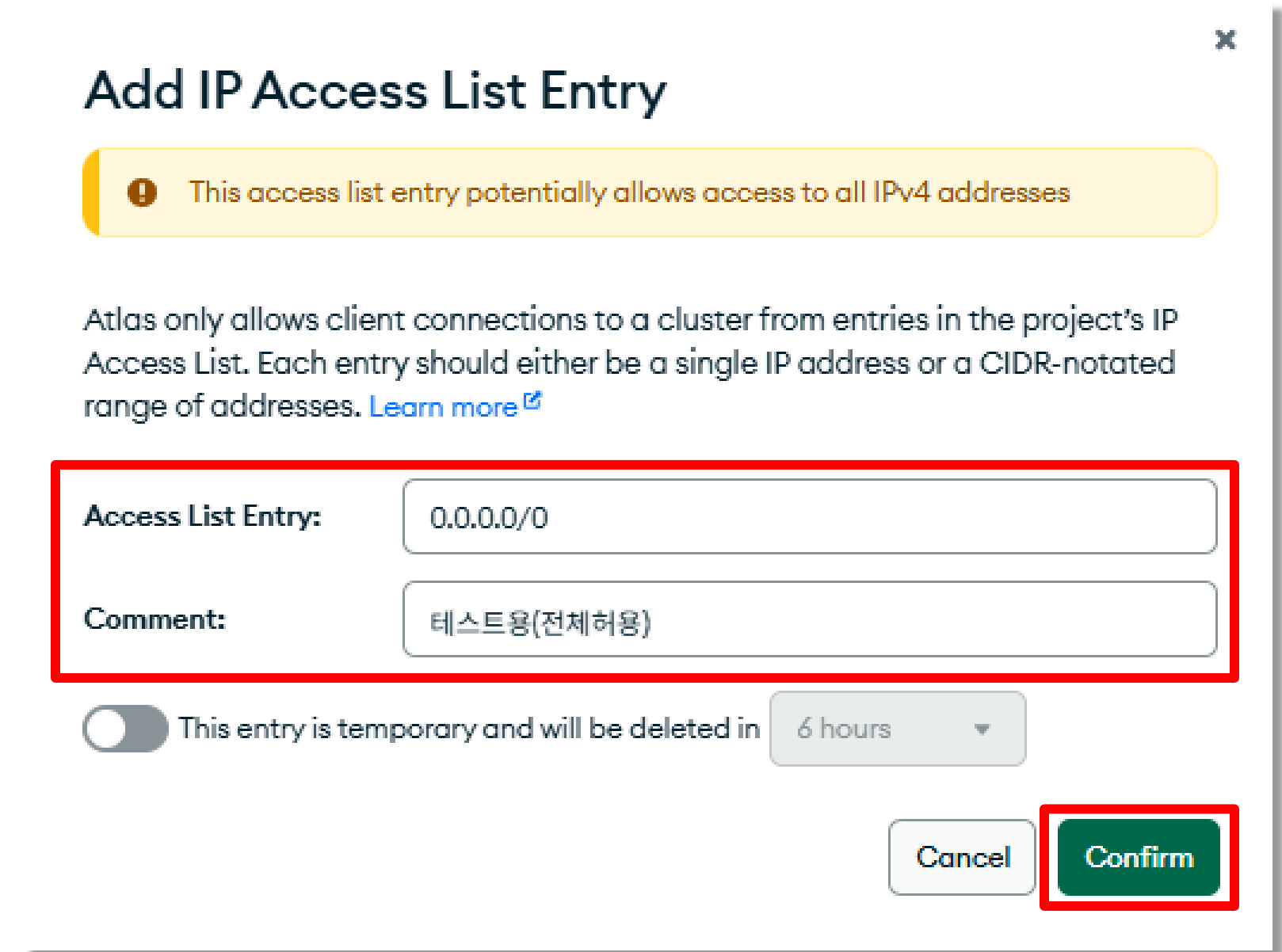
- 좌측 메뉴의 Database Access 클릭
- ADD NEW DATABASE USER 클릭
- Password Authentication 입력(사용자명, 패스워드)
- Built-in Role 설정
Add Built in Role > Select Role > Atlas admin 선택

백엔드 서버 설정



[MongoDB Network Access 설정]

- 좌측 메뉴의 Network Access 클릭
- ADD IP ADDRESS 클릭
- Access List Entry : 0.0.0.0/0
Comment : (자유롭게 입력)
- Confirm 클릭



백엔드 서버 설정

```
1 VITE_API_URL = http://localhost:5000
2 MONGO_URI=mongodb+srv://<db_username>:<db_password>@cluster0.i9psmch.mongodb.net/
3
```

[VSCode URL 연결]

- .env 파일에 MONGO_URI 환경변수 설정
-> MONGO_URI=mongodb+srv://<db_username>:<db_password>@cluster0.i9psmch.mongodb.net/
(<db_username> : atlas 계정 생성 시 입력했던 사용자 이름으로 변경)
(<db_password> : atlas 계정 생성 시 입력했던 사용자 비밀번호로 변경)

백엔드 서버 설정

```
PS F:\Last_web> cd server
PS F:\Last_web\server> npm run gen-jwt

> express_server@1.0.0 gen-jwt
> node seeder/jwtGen.js

JWT_SECRET=73fe0ebe50b2a1271983fc1c0a31a343937e32f1afc31eb508ab65b65d714ca90fe16a4d5b1af1866ac92680c69f189af8d76d7
3cd2c1ea6a3d8a06cbe12c634
REFRESH_TOKEN_SECRET=14b7eecfa967bf02caecfa1868ac94795f3721a6e6a6c6a5e67c301467234dfa87997f6f04766729e19bdc2612651
d2c26a4a0670430b9161d73679580a29ddd
```

[npm 설치]

- cd server -> npm install -> npm run gen-jwt 실행
- package.json에서 어떤 명령이 실행되는지 알 수 있음

```
"scripts": {
  "start": "node server.js",
  "dev": "nodemon server.js",
  "seed": "node seeder/seeder.js",
  "setup-ai": "node seeder/setupAIUser.js",
  "gen-jwt": "node seeder/jwtGen.js"
},
```

백엔드 서버 설정

[npm 설치]

- 콘솔의 JWT_SECRET={서명키 값} 을 복사하여 .env의 15열에 붙여넣기
- REFRESH_TOKEN_SECRET={서명키 값} 을 복사하여 .env의 19열에 붙여넣기

```

15 JWT_SECRET=YOUR_JWT_SECRET
16 JWT_EXPIRE=30d
17 JWT_COOKIE_EXPIRE=30
18
19 REFRESH_TOKEN_SECRET=YOUR_REFRESH_TOKEN_SECRET
20 REFRESH_TOKEN_EXPIRE=7d
21
22 OPENAI_API_KEY=YOUR_OPENAI_API_KEY
23
24 # 게시글 조회 시 자동으로 AI 응답 생성 (true/false)
25 AUTO_GENERATE_AI_RESPONSE=false
26 # 게시글 AI 답변 댓글 생성 (true/false)
27 AUTO_GENERATE_AI_COMMENT=true
28
29 # 클라이언트 URL
30 CLIENT_URL=https://localhost:5173

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS F:\Last_web> cd server

PS F:\Last_web\server> npm run gen-jwt

> express_server@1.0.0 gen-jwt

> node seeder/jwtGen.js

```

JWT_SECRET=73fe0ebe50b2a1271983fc1c0a31a343937e32f1afc31eb508ab65b65d714ca90fe16a4d5b1af1866ac92680c69f189af8d76d7
3cd2c1ea6a3d8a06cbe12c634
REFRESH_TOKEN_SECRET=14b7eecfa967bf02caecfa1868ac94795f3721a6e6a6c6a5e67c301467234dfa87997f6f04766729e19bdc2612651
d2c26a4a0670430b9161d73679580a29ddd

```

백엔드 서버 실행 및 테스트 확인

[시드 데이터 삽입 & AI유저 생성]

- Npm run seed로 작동 확인용 시드 데이터 삽입

```
PS F:\Last_web\server> npm run seed
```

- Npm run setup-ai로 질문 응답용 AI유저 생성 및 권한 부여

```
PS F:\Last_web\server> npm run setup-ai
```

백엔드 서버 실행 및 테스트 확인

```
PS F:\Last_web\server> npm run dev

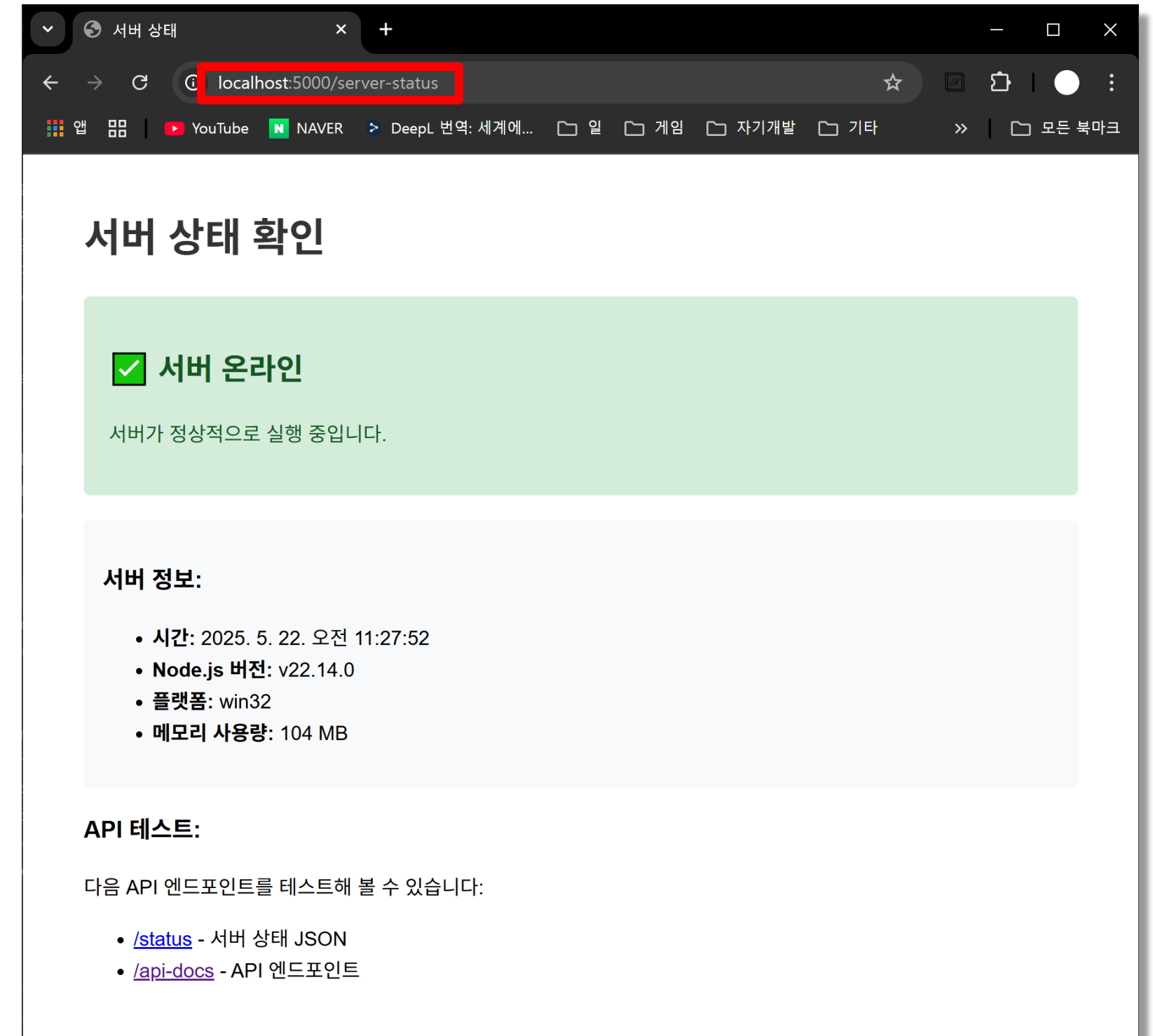
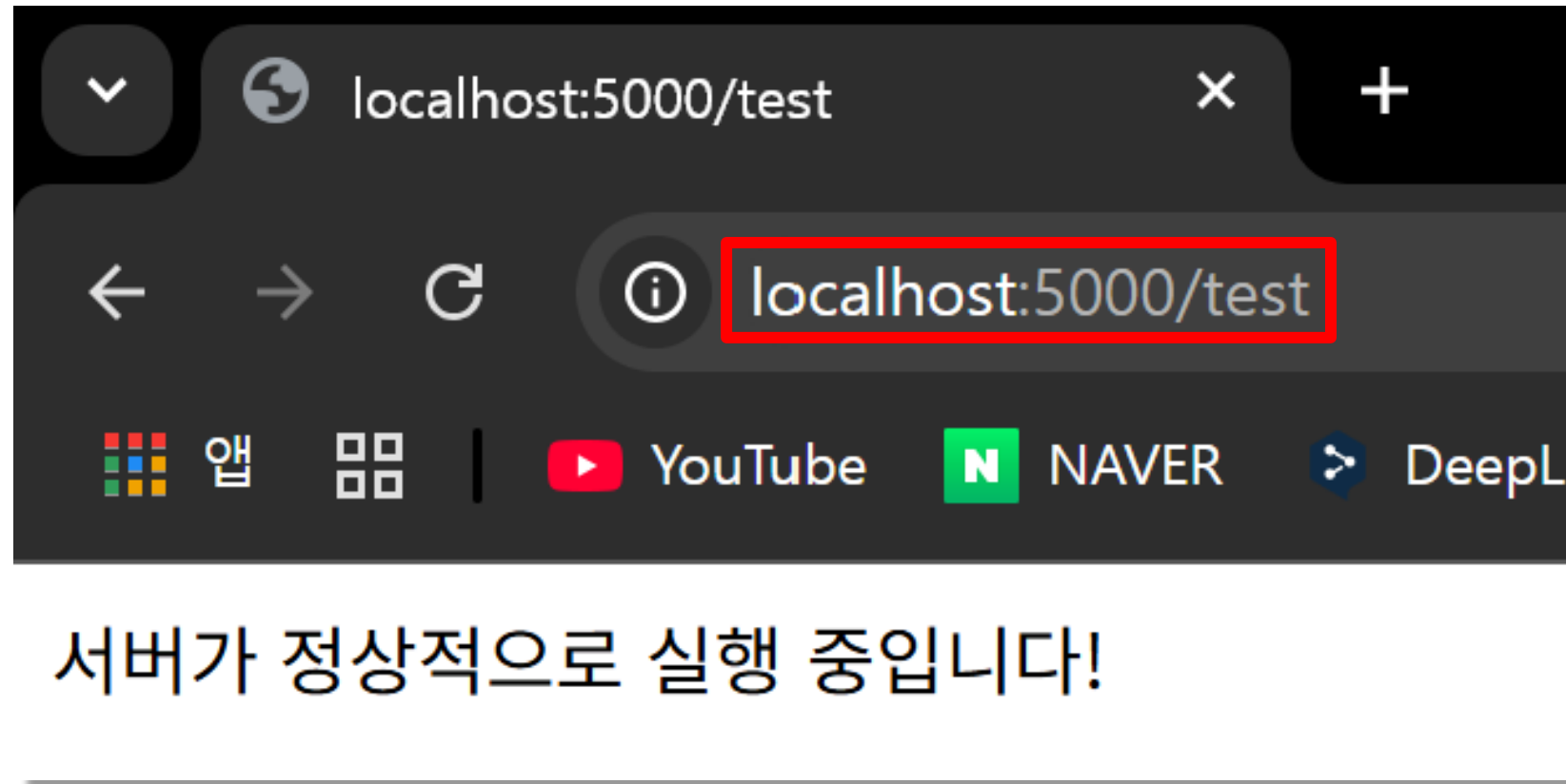
> express_server@1.0.0 dev
> nodemon server.js

[nodemon] 3.1.10
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node server.js`
MongoDB에 연결되었습니다
서버가 5000 포트에서 실행 중입니다 (MongoDB 활성화됨)
```

[백엔드 서버 실행]

- npm run dev 또는, npm run start로 백엔드 서버 실행
- + npm run dev : nodemon을 사용하여 서버 실행 중 코드 수정 및 저장 시 서버 자동 재시작
- + nodemon : 파일 변경을 감지하여 서버를 자동으로 재시작해주는 도구

백엔드 서버 실행 및 테스트 확인



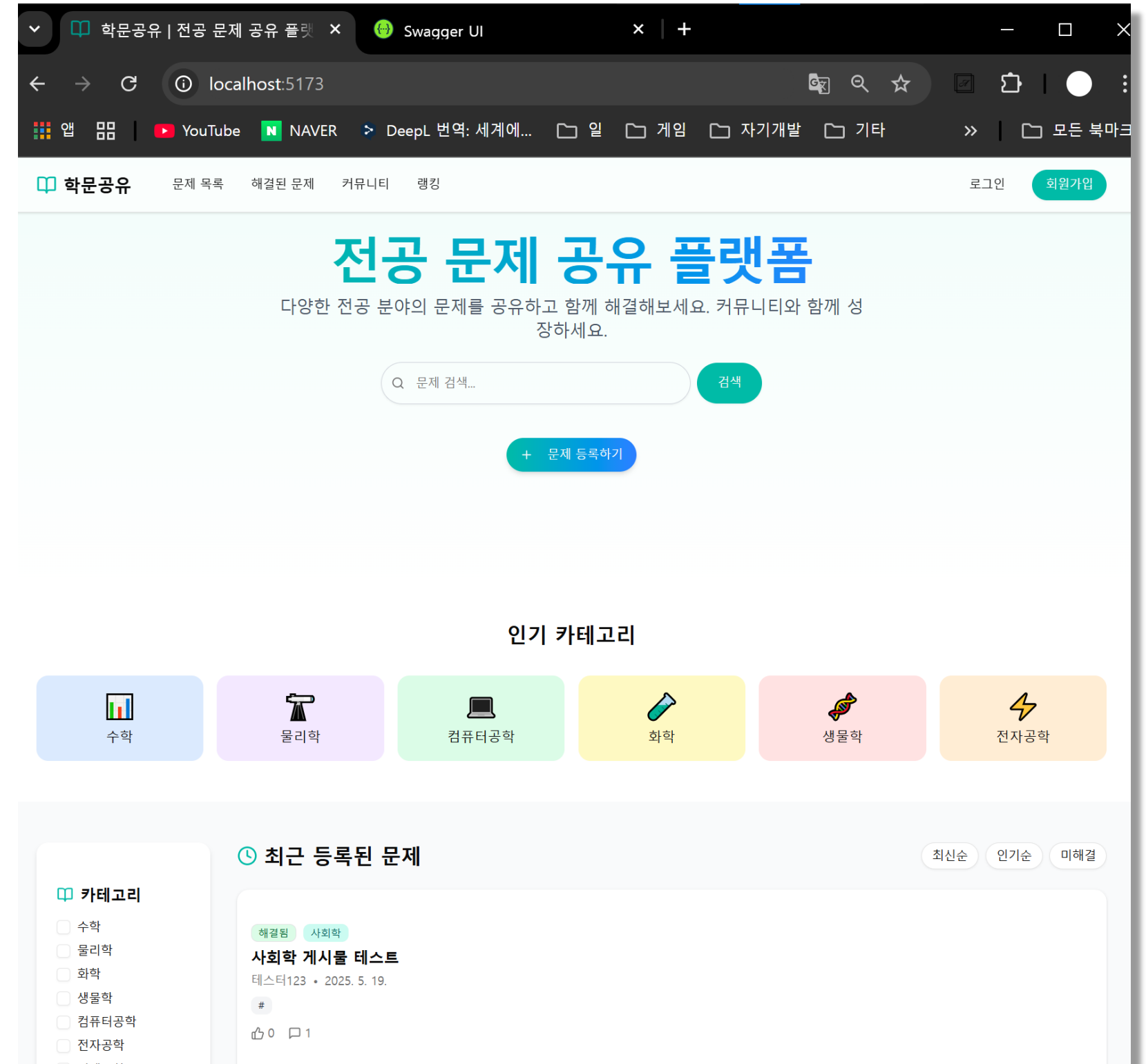
[백엔드 서버 실행]

- 브라우저 주소창에 localhost:5000/test 입력
또는, localhost:5000/server-status로 실행 중 확인 가능

백엔드 서버 실행 및 테스트 확인

[페이지 실행 확인]

- 정상적으로 실행 중인 페이지 확인



Q & A

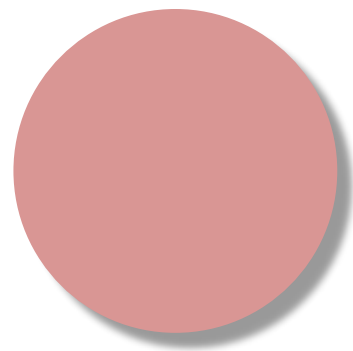
— 프로젝트 레포지토리로 이동하기 —

|

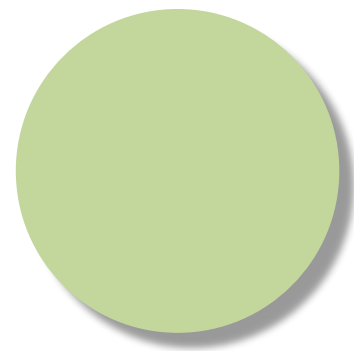
|

|

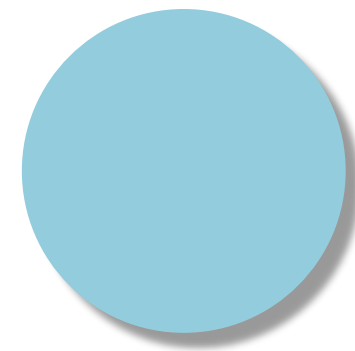
- 슬라이드로 이동 -



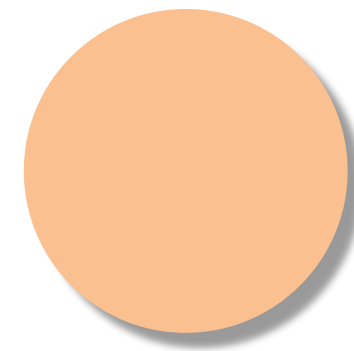
프로젝트
가져오기



백엔드
서버 설정



OpenAI
연동 설정



백엔드 서버 실행 및
테스트 확인

