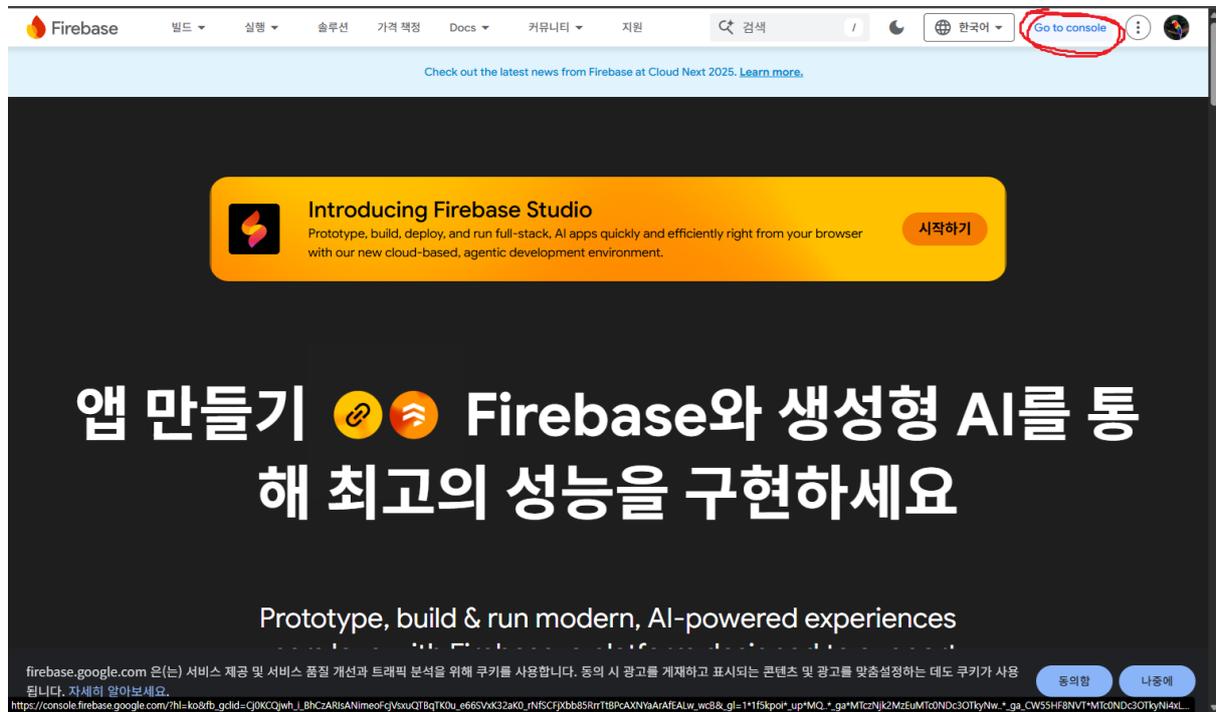
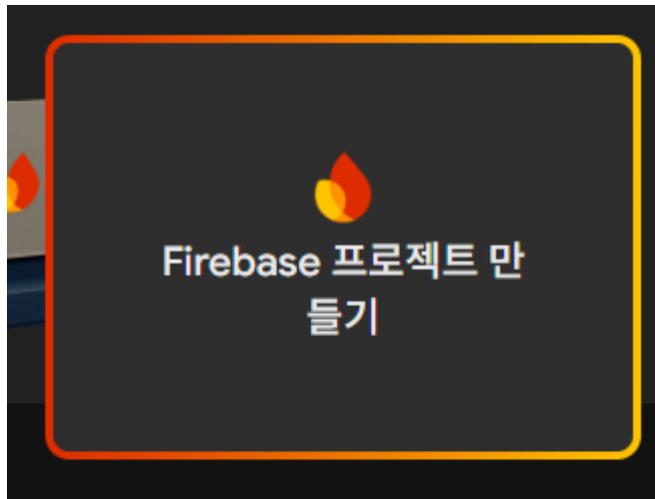


1) Firebase 콘솔로가기

Firebase 들어오면 로그인 해주면 오른쪽위에 콘솔이 있다 들어가준다.



들어가면 아래사진처럼 프로젝트 만들기 버튼이 있다. 만들어 준다.



프로젝트 이름은 본인이 기억하기 쉬운 이름으로 한다 (예:test)

계속 버튼을 누르면 Firebase 프로젝트를 위한 AI 지원이 있다 허용해준다.

Firebase 프로젝트를 위한 Google 애널리틱스가 있는데 이건 거부해주고 프로젝트 만들기 버튼을 누른다.

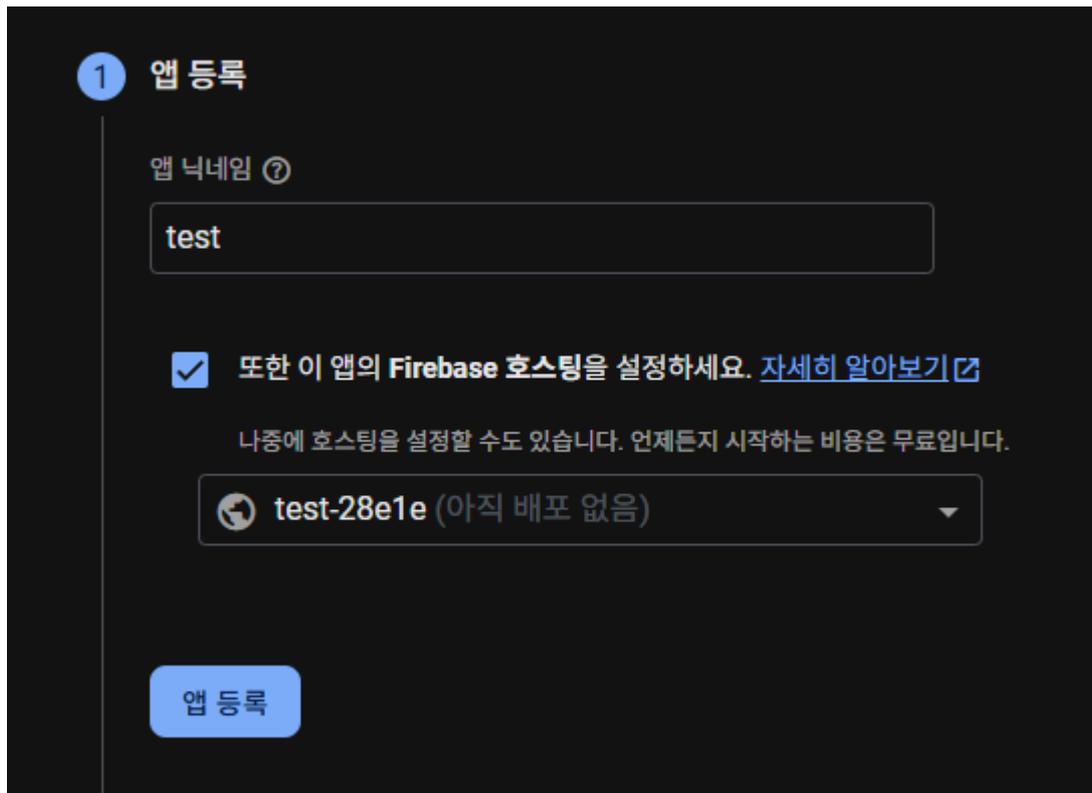
프로젝트를 만들면 자동으로 test 프로젝트에 들어와진다.

가운데에 보면 웹 만드는 버튼이 있다 눌러준다.



웹 닉네임도 프로젝트와 같은 이름으로 만들어준다.

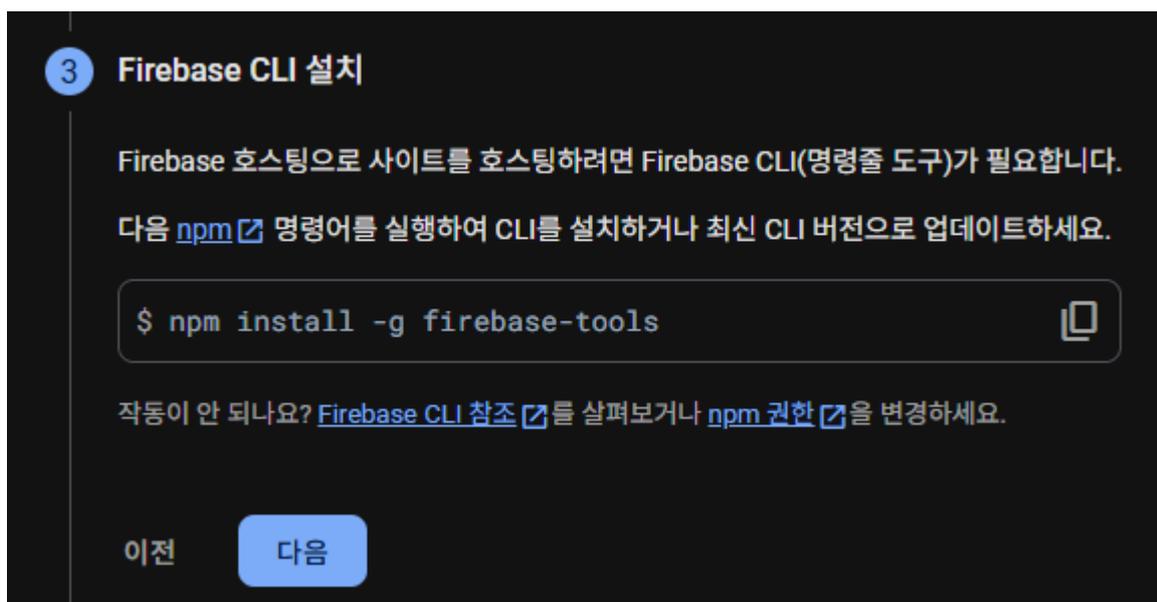
그리고 호스팅을 허용해주고 앱 등록 버튼을 누른다.



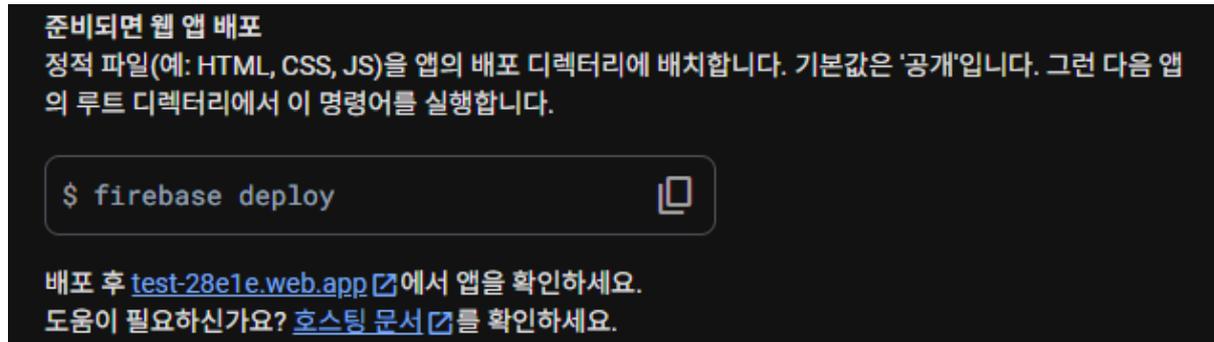
SDK에서 npm사용 하고 다음 누른다.

그리고 아래 사진보면 Firebase CLI가 있는데 이거 npm을 복사하는게 좋다

나중에 vscode에서 Firebase를 호스팅하려면 필요한 npm이다.



그리고 이걸 나중에 테스트하고 싶을 때 실행해주면 호스팅이 되면서 본인이 코드 만든 것을 적용할 수 있는 명령어이다.



이제 왼쪽에 제품 카테고리에 Authentication에 들어가준다.

들어가면 일단 시작해주고 로그인 방식에 기본제공업체에 이메일/비밀번호 버튼을 선택해준다.

들어가면 이메일/비밀번호와 이메일 링크가 있는데 둘 다 선택해주고 저장해준다.

이러면 Authentication은 해결됐다.

그리고 왼쪽 빌드 카테고리에 Firestore에 들어가준다.

여기서 데이터베이스 만들기를 눌러준다.

위치는 그냥 냅두고 다음 버튼을 누른다.

프로덕션 모드에서 시작을 누르고 만들기 버튼을 누른다.

만들면 위에 규칙 탭이 있는데 들어가준다.

규칙탭에 보면 규칙이 설정되어있는데 이걸 v-talk 계정에서 가져온다.

```
1 rules_version = '2';
2 service cloud.firestore {
3     match /databases/{database}/documents {
4
5         // 기타 모든 문서는 로그인한 사용자만 접근 가능
6         match /{document=**} {
7             allow read, write: if request.auth != null;
8         }
9     }
10 }
11
```

이런 규칙으로 설정되어 있으면 된다.

규칙 설정했으면 빌드에서 Realtime Database에 들어가준다.

여기서 데이터베이스 만들기 해준다.

미국으로 하고 2번째 탭에 테스트 모드에서 시작을 누르고 사용설정 버튼을 누른다.

나중에 위에 규칙 탭에 들어가서 아래 사진처럼 바꿔줘야 한다.

```
1 {
2   "rules": {
3     ".read": "now < 1777065600000", // 2026-04-25
4     ".write": "now < 1777065600000" // 2026-04-25
5   }
6 }
7
```

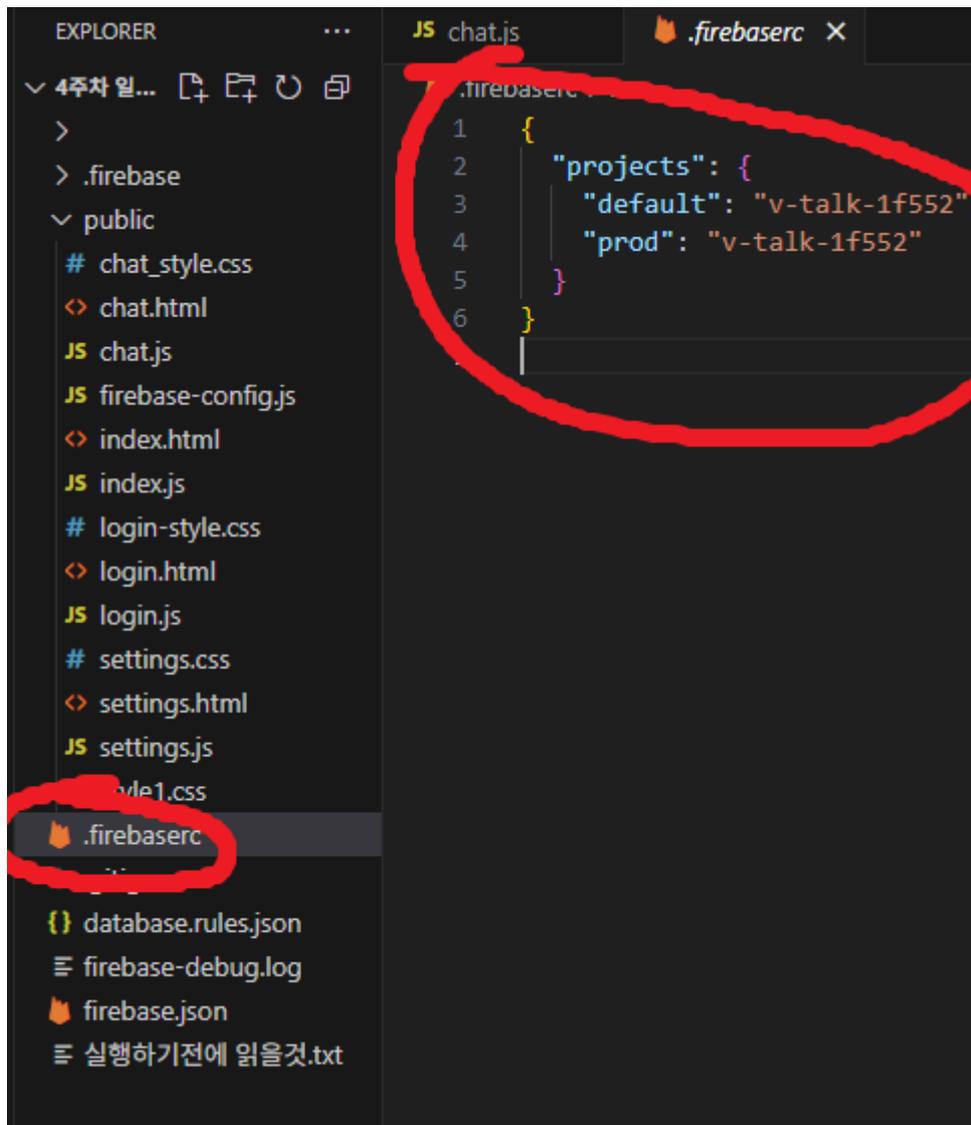
이제 왼쪽 빌드에서 Hosting을 들어가서 만들어주고 그냥 다음 버튼만 누르면 완료된다.

이제 웹에서 설정해야 하는 건 다 했다.

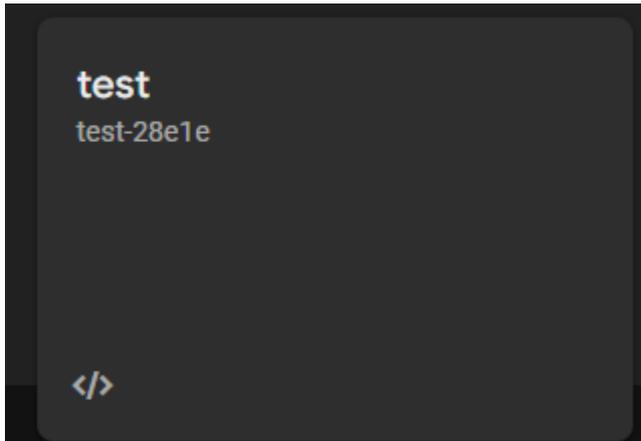
-- vscode에서 실행하는법 --

Vscode에서 일단 폴더를 가져온다.

그리고 .firebaserc에 들어가면 projects에 default와 prod를 자신을 프로젝트 이름으로 바꿔주면 된다.

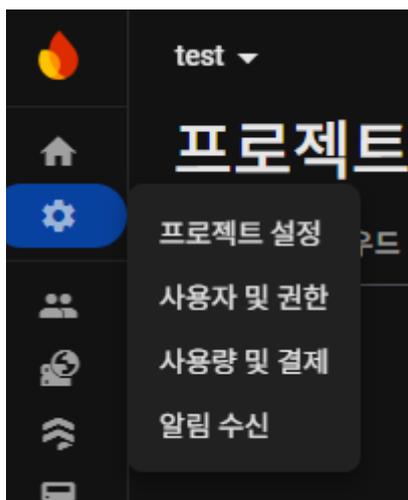


프로젝트 이름 보는법은 firebase 메인화면에 test 아래에 있는 아이디가 projects이다.



그리고 왼쪽에 firebase-config에 들어가준다.

여기서 firebase 설정 값이 있는데 이걸 firebase 웹에서 프로젝트 설정 버튼을 누르고



일반에서 내리다 보면 구성 탭이 있다 거기서 firebaseConfig를 복사해준다.

SDK 설정 및 구성

npm CDN 구성

앱의 Firebase 구성 객체에 사용할 스니펫을 가져오세요. [자세히 알아보기](#)

앱의 키 및 식별자가 포함된 Firebase 구성 객체:

```
const firebaseConfig = {
  apiKey: "AIzaSyB6JpA1hOpXWD2BYspLphkjSRqsv7YHS_0",
  authDomain: "test-28e1e.firebaseio.com",
  databaseURL: "https://test-28e1e-default-rtdb.firebaseio.com",
  projectId: "test-28e1e",
  storageBucket: "test-28e1e.firebaseio.com",
  messagingSenderId: "616182715880",
  appId: "1:616182715880:web:b3d0b27f3c04b1dfce005d"
};
```

npm과 번들러(예: webpack 또는 Rollup)를 사용 중이신가요? [모듈형 SDK](#)를 확인하세요.

웹용 Firebase 자세히 알아보기: [시작하기](#), [Web SDK API 참조](#), [샘플](#)

복사해주고 그대로 붙여넣기만 해주면 된다.

```
// Firebase 설정 값
const firebaseConfig = {
  apiKey: "AIzaSyAS0jfYQ8sbbM3E8zzR7m35zpJvklQ8gB0",
  authDomain: "v-talk-1f552.firebaseio.com",
  databaseURL: "https://v-talk-1f552-default-rtdb.firebaseio.com",
  projectId: "v-talk-1f552",
  storageBucket: "v-talk-1f552.firebaseio.com",
  messagingSenderId: "563843573392",
  appId: "1:563843573392:web:d6c5473a3e00bc18d55a15"
};
```

그리고 chat.js에 가면 firebaseConfig가 또 있는데 이것도 링크에 맞게 바꿔주면 된다.

```
// Firebase 설정
const firebaseConfig = {
  apiKey: "AIzaSyASOjfYQ8sbbM3E8zzR7m35zpJvklQ8gB0",
  authDomain: "v-talk-1f552.firebaseio.com",
  databaseURL: "https://v-talk-1f552-default-rtdb.firebaseio.com"
};
/*
```

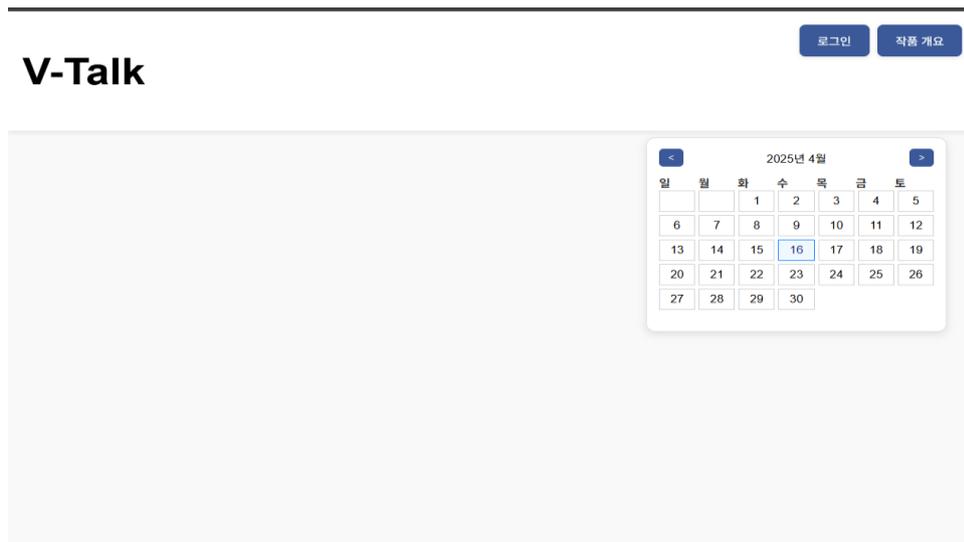
터미널에서 `npm i -g firebase-tools`를 쳐서 다운해준다.

그리고 `firebase deploy` 치면 지금까지 코딩한 것이 호스팅이 되면서 아래 사진처럼 Hosting URL 링크가 뜨는데 이게 본인의 사이트이다.

호스팅 할 때 마다 이 사이트로 가야하기에 저장하는 것이 좋다.

```
Project Console: https://console.firebase.google.com/project/v-talk-1f552/overview
Hosting URL: https://v-talk-1f552.web.app
```

호스팅 된 사이트로 가주면 잘 실행된 것을 볼 수 있다.



이러면 끝