

구글 Firebase, NodeMCU를 이용한 IoT 기반 LED 제어

Week09

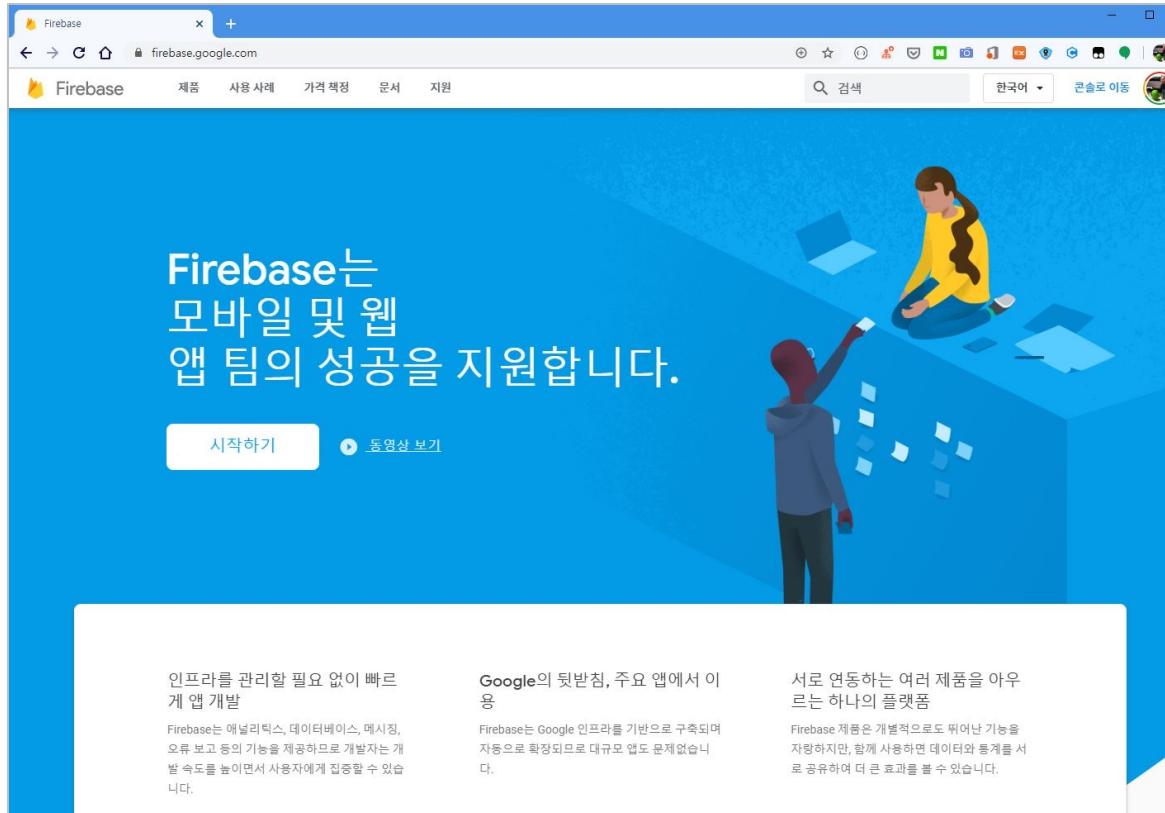
새로 배우는 내용들

- NodeMCU를 이용한 IoT 디바이스 개발
- 구글 클라우드 서비스 Firebase, Realtime Database 사용법
- Firebase IoT 웹 앱 개발(JavaScript)
- Firebase IoT 안드로이드 앱 개발



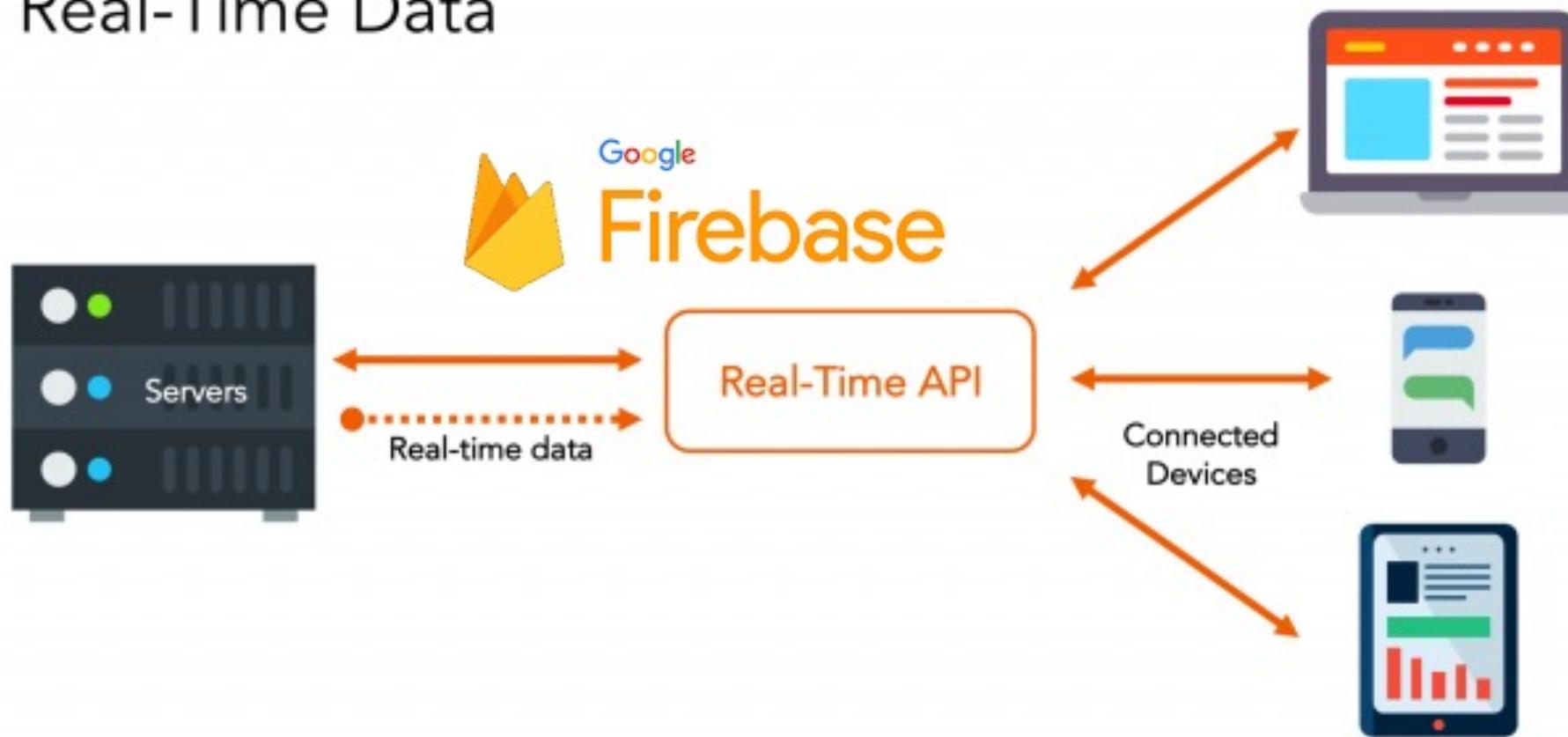
구글 Firebase

- 구글에서 제공하는 클라우드 서비스의 일종 :
 - BaaS(Backend as a Service)
- <https://firebase.google.com/>



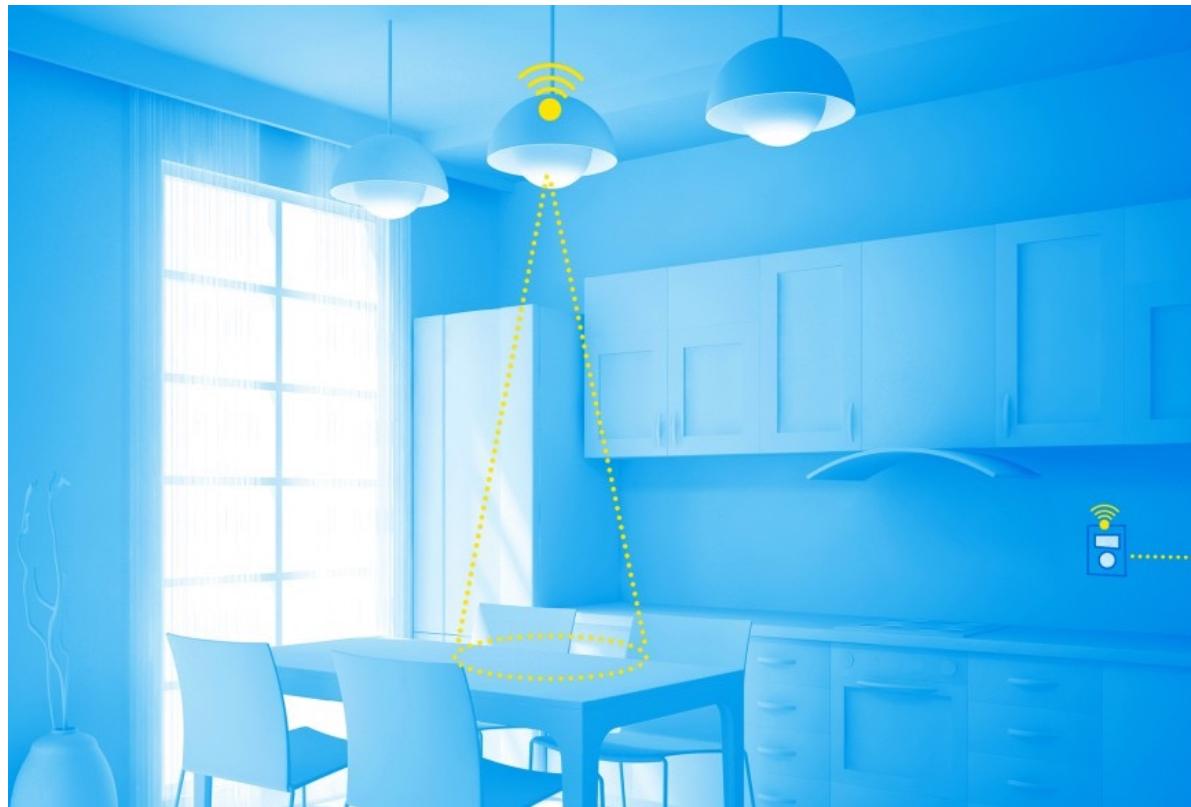
Firebase Real-Time Database : NoSQL

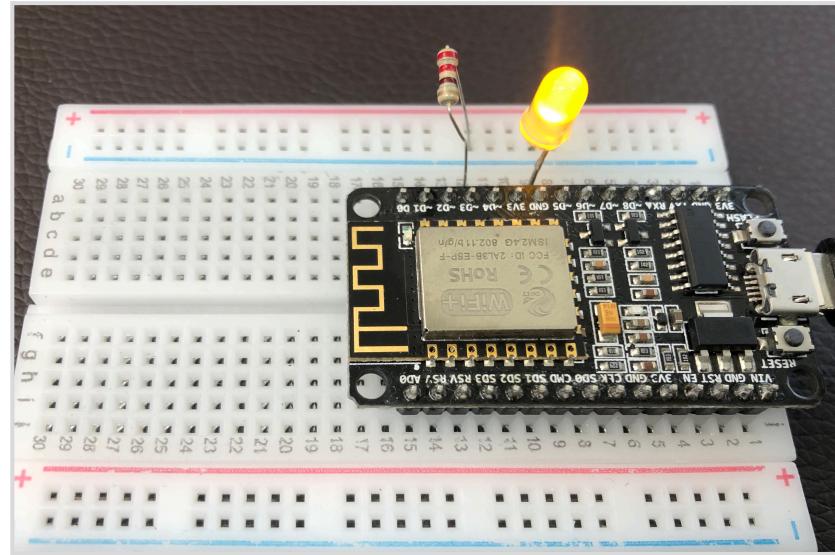
Real-Time Data



무엇을 개발하나?

- 스마트 스위치 : 원격 LED ON/OFF





< NodeMCU IoT 디바이스 >



< Firebase IoT 웹 앱 >



< Firebase IoT 안드로이드 앱 >

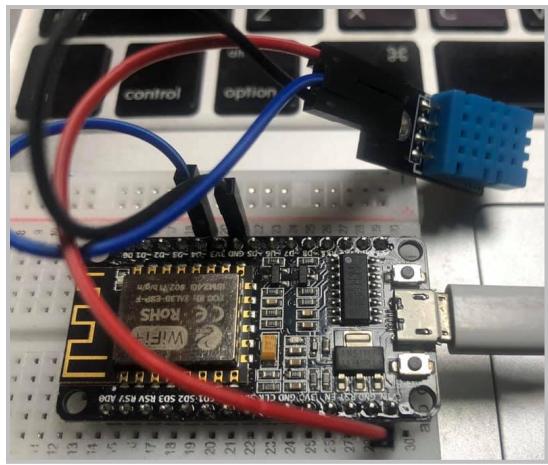


< Firebase Realtime Database >

무엇을 개발하나?

- 클라우드 기반 IoT 실시간 원격 온습도 모니터

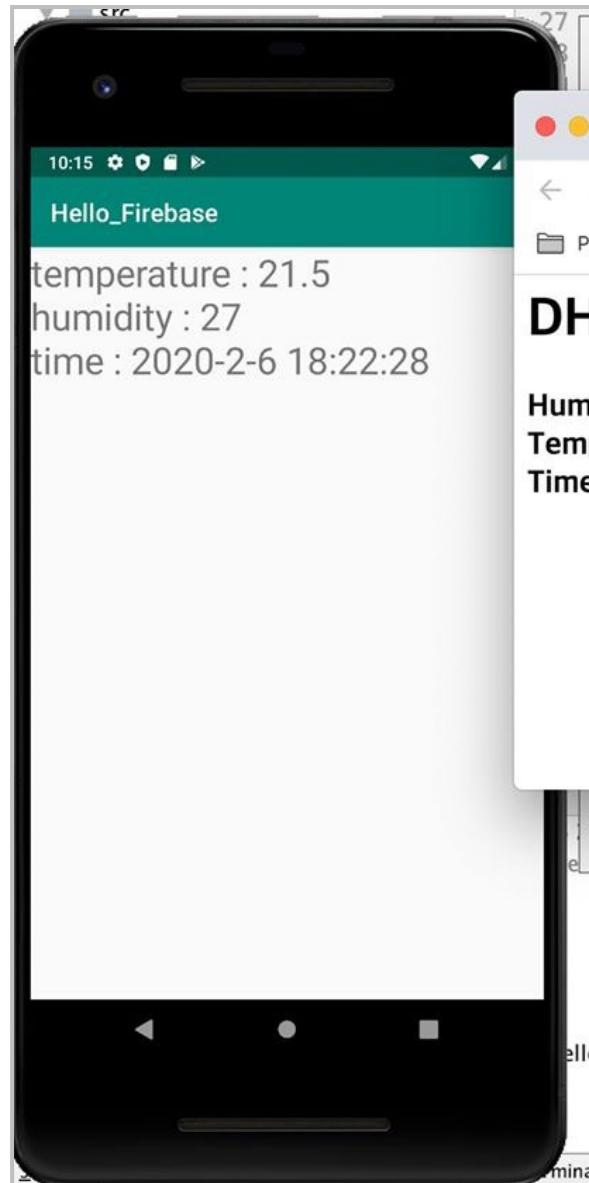




< NodeMCU/ DHT11 >



< Realtime Database >



DHT11 Logger by Firebase

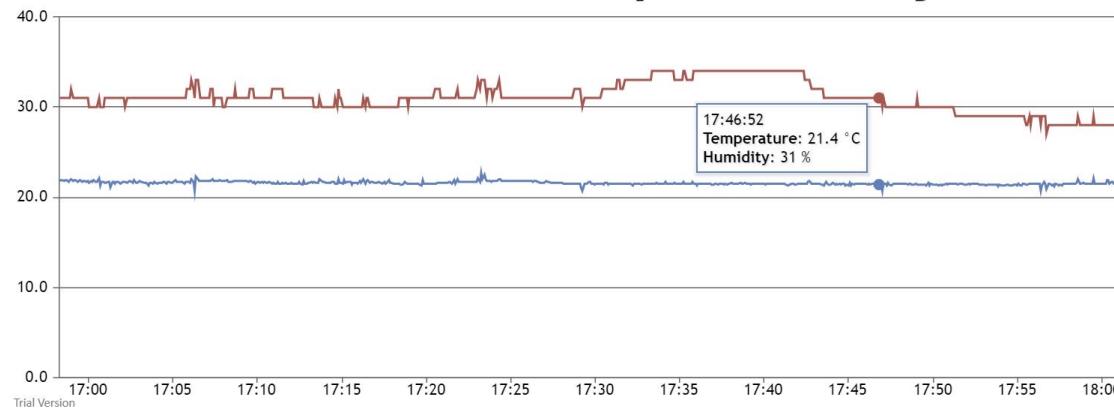
Humidity : 27 %

Temp : 21.5 C

Time : 2020-2-6 18:22:28

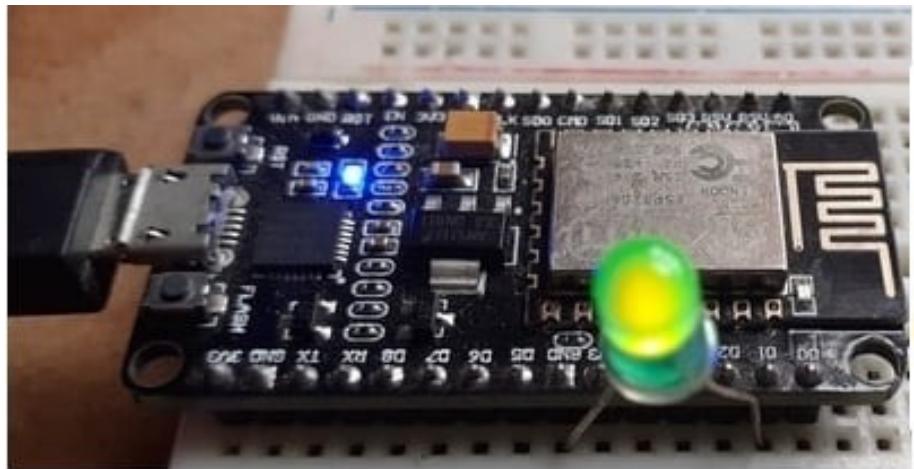
DHT11 Logger by Firebase

Temperature & Humidity



< 웹 대시보드 : CanvasJS >

< 안드로이드 앱/ 웹 앱 >



Firebase

The image displays two screenshots illustrating a real-time data connection between an Arduino and a Firebase database.

Left Screenshot (Firebase Database):

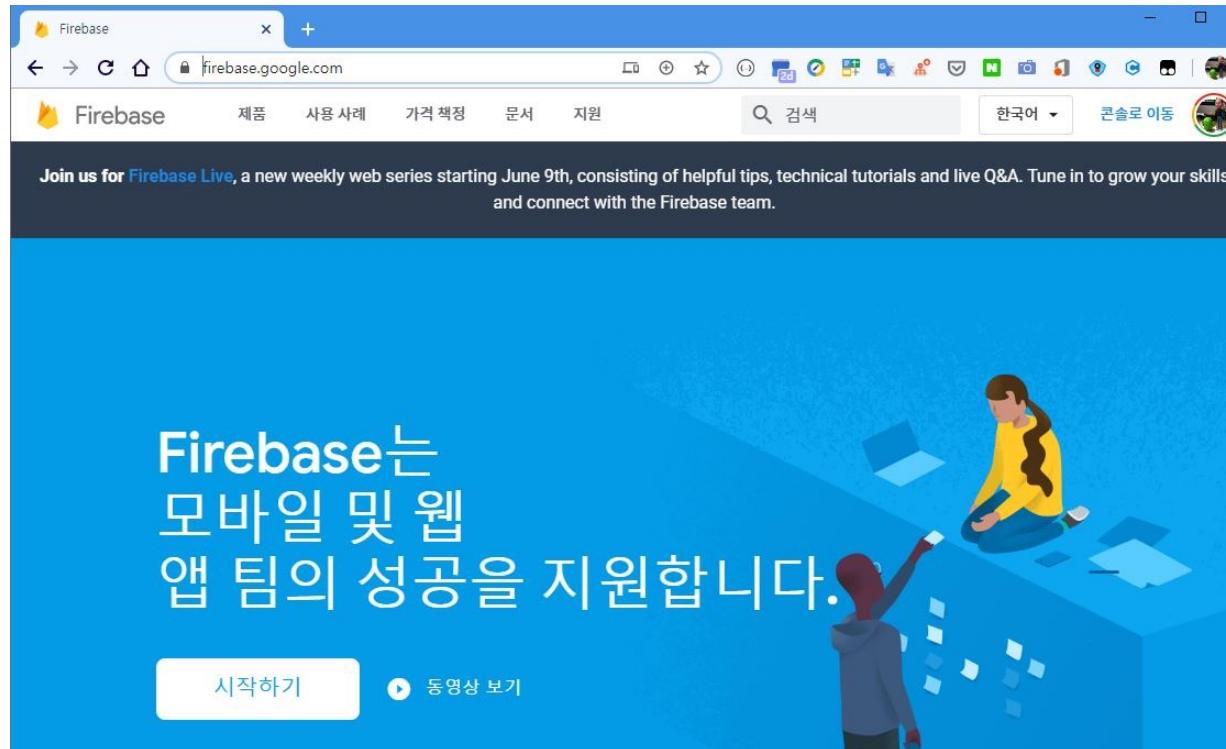
- Header: Database, Realtime Database
- Navigation: Data, Rules, Backups, Usage
- Project URL: <https://my1stproject-34e8e.firebaseio.com/>
- Database Structure:
 - my1stproject-34e8e
 - LED_STATUS: "ON"

Right Screenshot (Arduino Serial Monitor):

- Title bar: sketch_may16a
- Port: COM4
- Text area:
 - Led Turned ON
 - Led Turned ON
 - Led Turned ON
- Bottom controls: Autoscroll, Show timestamp, Newline, 9600 baud, Clear output

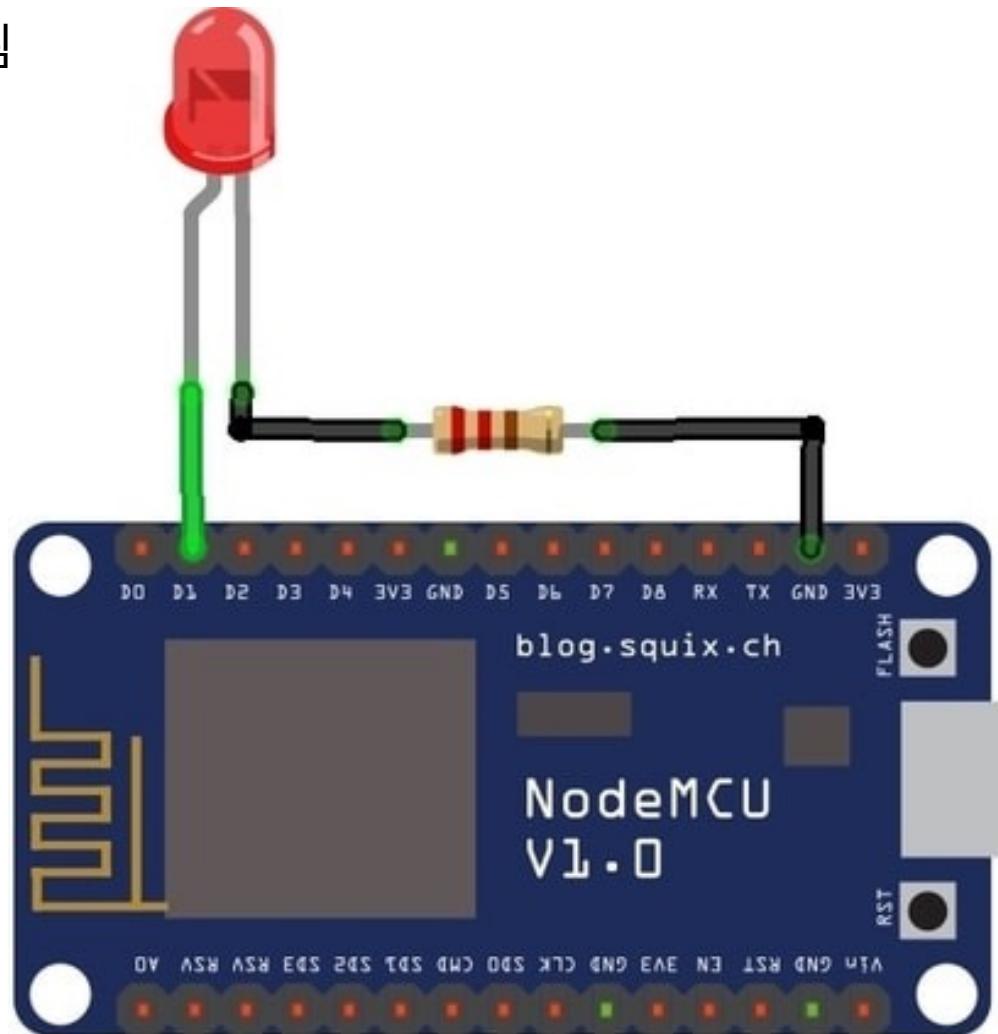
준비물

- NodeMCU ESP8266-12E 보드
- LED, 저항 220Ω, 브레드 보드
- 구글 Firebase : <https://firebase.google.com/>



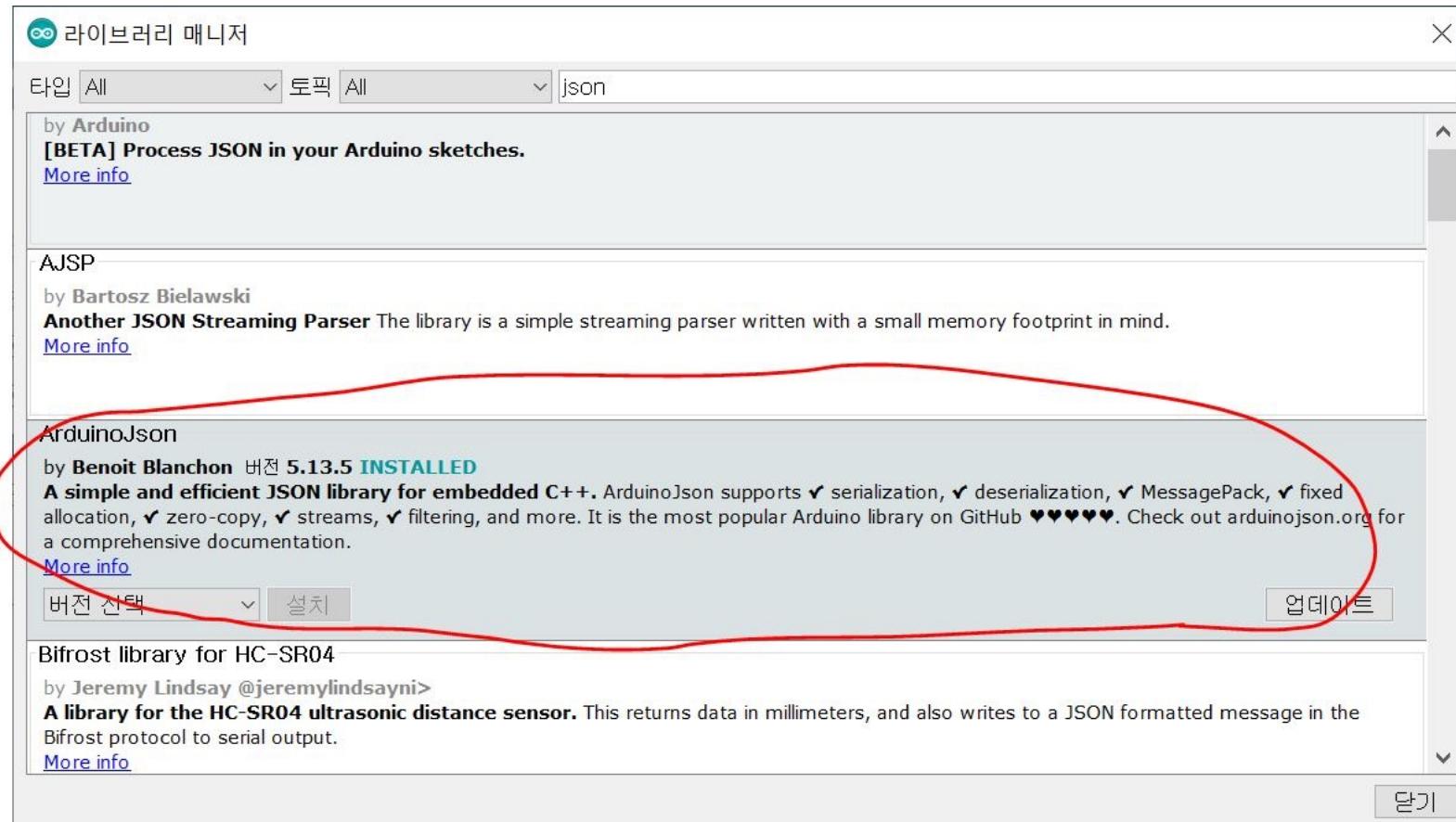
회로 구성

- 브레드 보드를 사용해도 됨



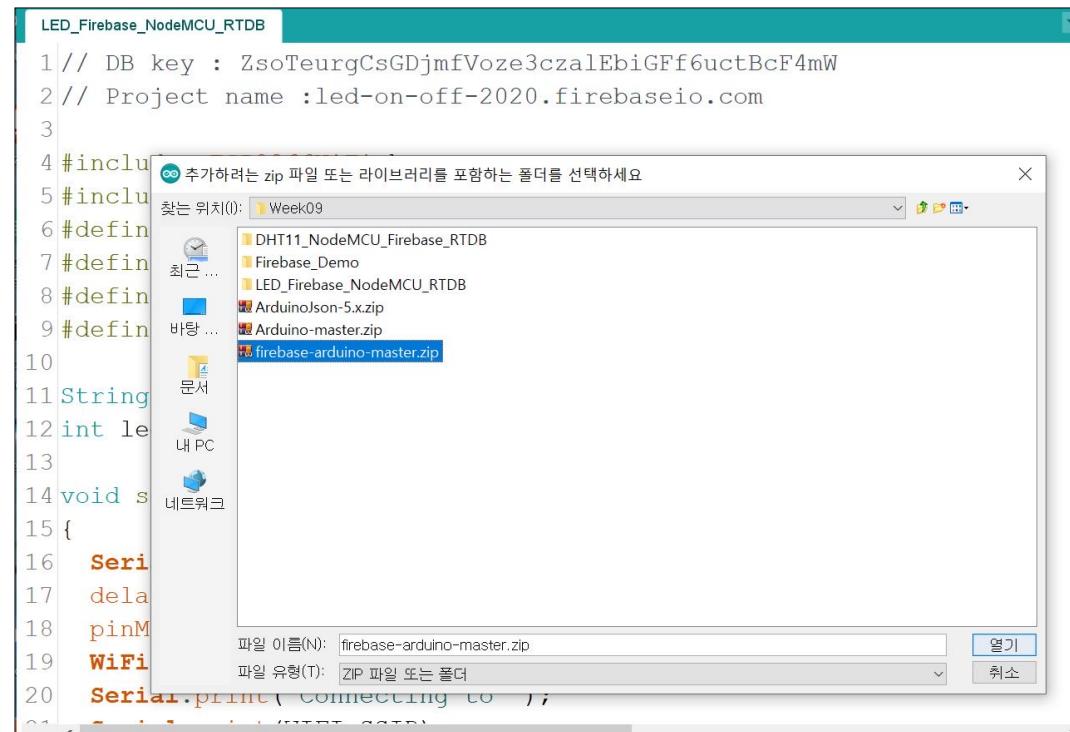
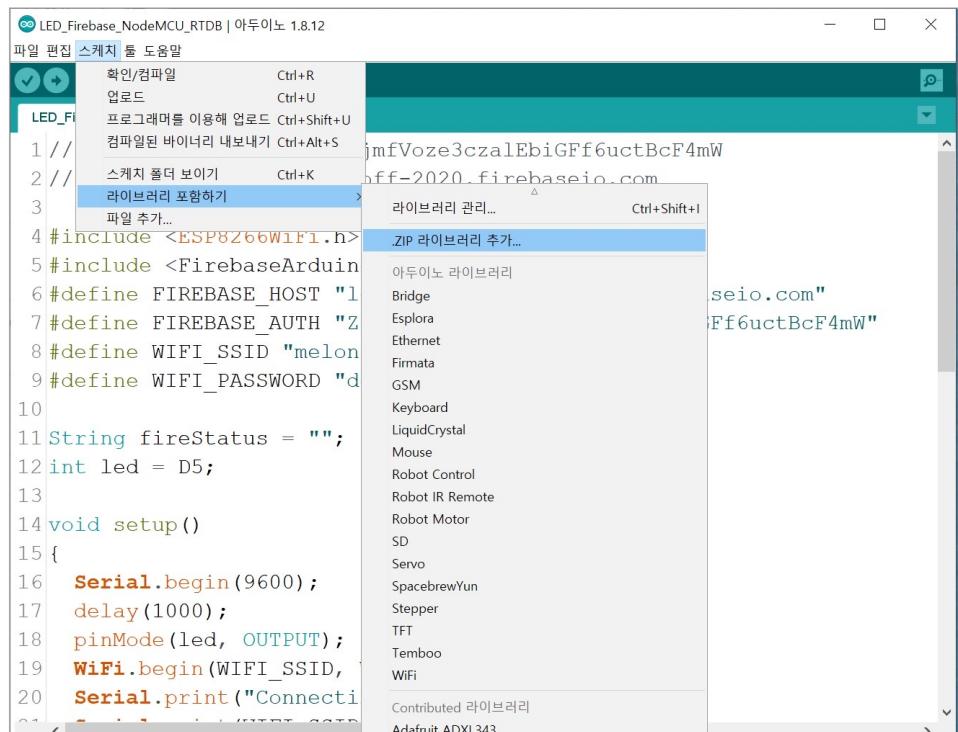
ArdunioJSON 라이브러리 설치

- 툴 -> 라이브러리 관리 -> “ArdunioJSON” by Benoit Blanchon
- 반드시 ArduinoJson-5x.zip 설치(**5.13.5**)



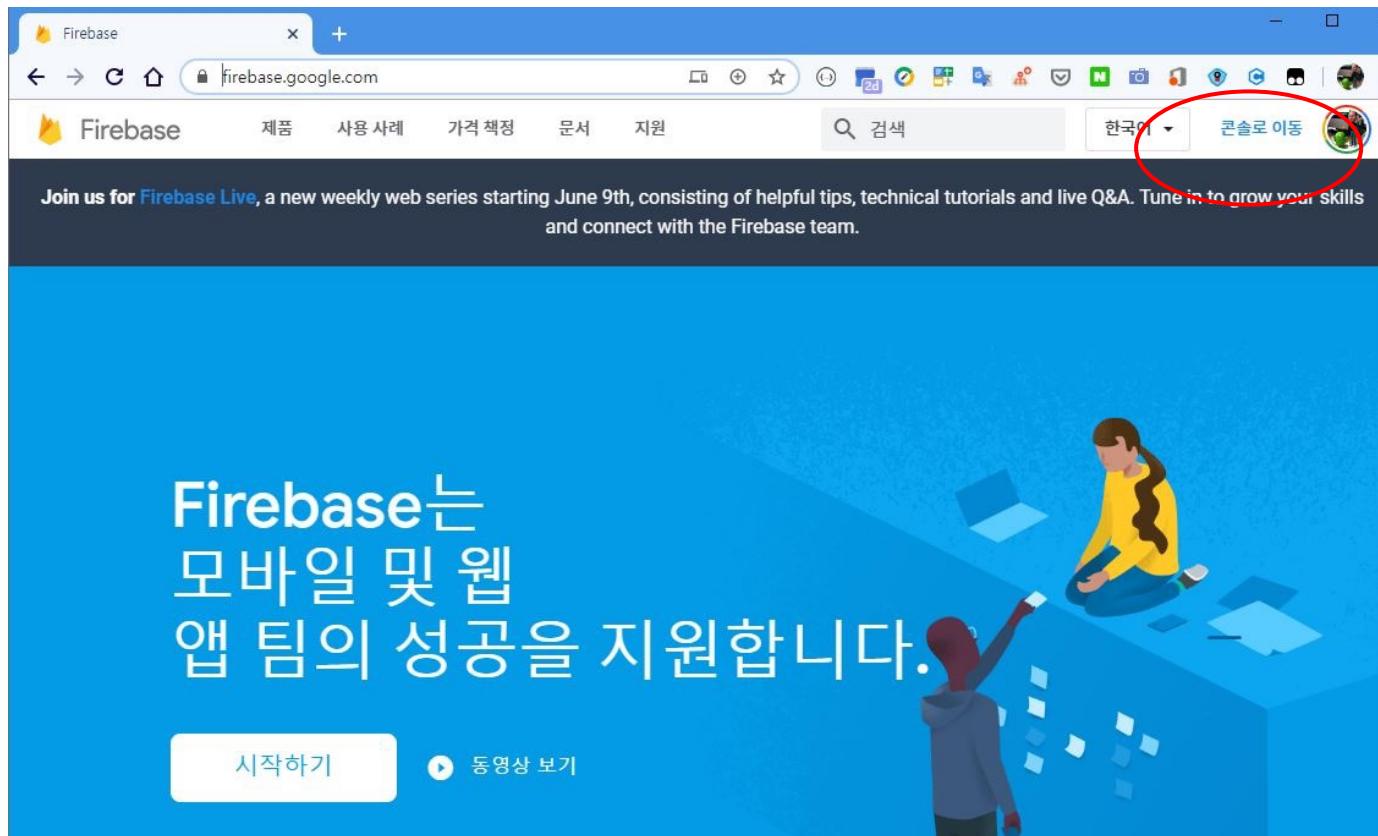
구글 Firebase-arduino 라이브러리 설치

- Firebase-arduino 라이브러리 다운로드, zip 파일 설치
- 다운로드 : <https://github.com/FirebaseExtended/firebase-arduino>
 - Firebase-arduino-master.zip
- 스케치 -> 라이브러리 포함하기 -> .ZIP 라이브러리 추가
- 툴 -> 보드:"NodeMCU1.0(ESP12E Moudle)" -> 보드매니저 -> esp8266 버전 2.77 다운그레이드

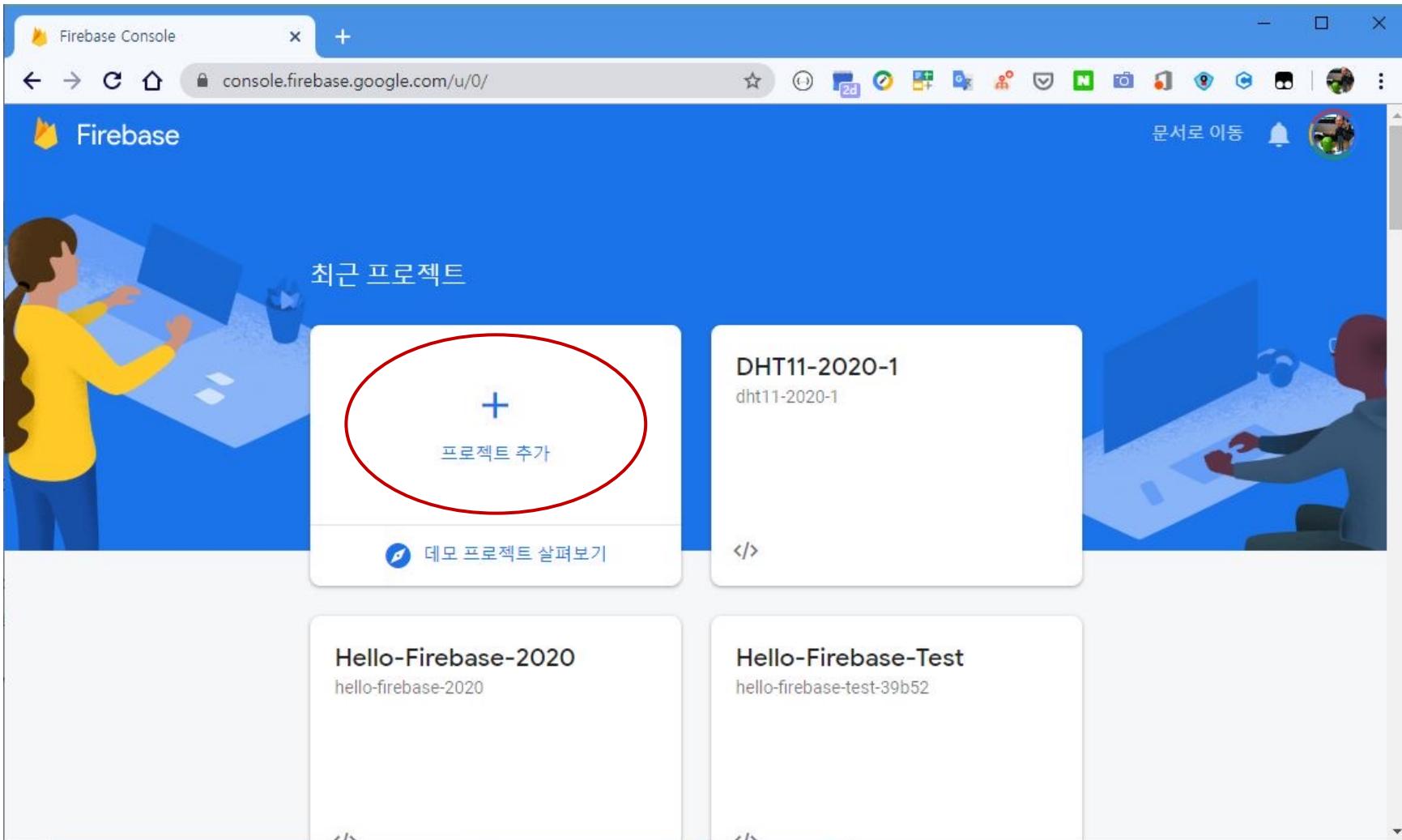


구글 Firebase 설정

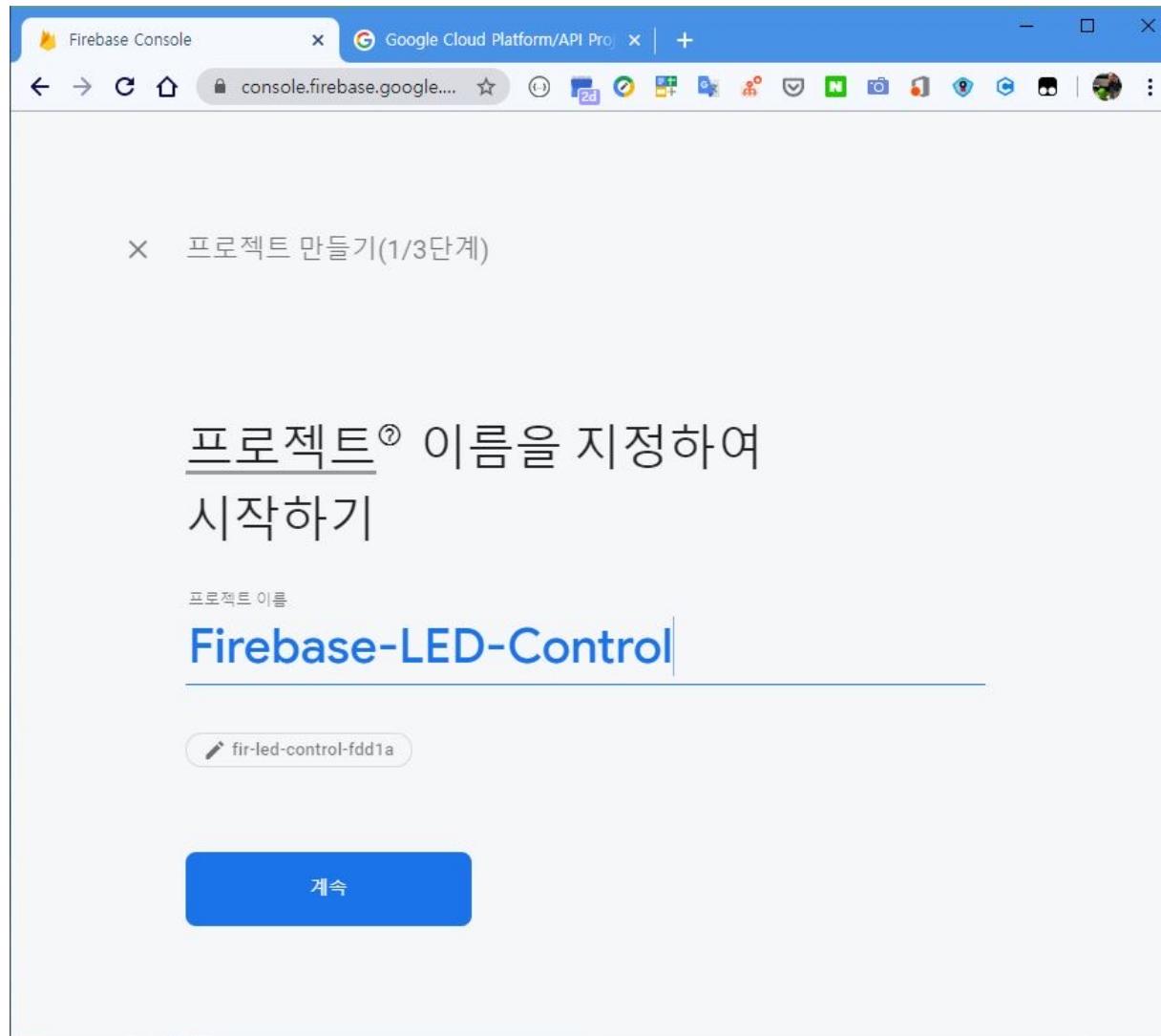
- 구글 계정으로 접속
- <https://firebase.google.com> -> 콘솔로 이동



프로젝트 추가



프로젝트 이름 지정



프로젝트 만들기

Firebase Console x Google Cloud Platform/API Pro x +

console.firebaseio.google....

×

프로젝트 만들기(2/3단계)

Firebase 프로젝트를 위한 Google 애널리틱스

무제한 무료 분석 솔루션인 Google 애널리틱스를 사용하면 Firebase Crashlytics, 클라우드 메시징, 인앱 메시지, 원격 구성, A/B 테스팅, 예측, Cloud Functions에서 타겟팅, 보고 등을 이용할 수 있습니다.

Google 애널리틱스를 통해 다음 기능을 이용할 수 있습니다.

- A/B 테스팅 ⓘ
- 장애가 발생하지 않은 사용자 ⓘ
- Firebase 제품 전반에서 사용자 세분화 및 타겟팅 ⓘ
- 이벤트 기반 Cloud Functions 트리거 ⓘ
- 제한 없는 무료 보고 ⓘ
- 사용자 행동 예측 ⓘ

이 프로젝트에서 Google 애널리틱스 사용 설정 권장사항

이전

계속

Firebase Console x Google Cloud Platform/API Pro x +

console.firebaseio.google....

×

프로젝트 만들기(3/3단계)

Google 애널리틱스 구성

Google 애널리틱스 계정 선택 또는 만들기 ⓘ

Default Account for Firebase

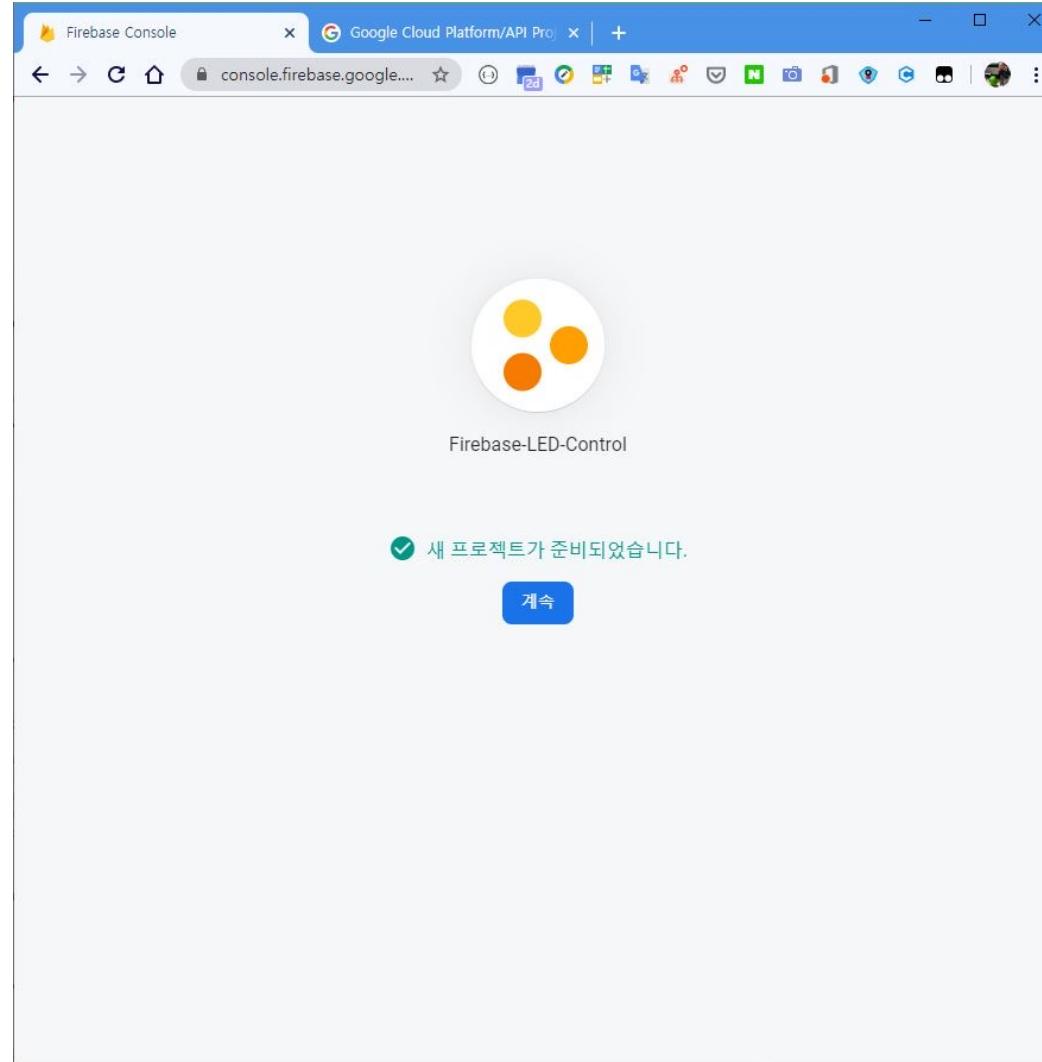
이 계정에서 자동으로 새 속성 만들기

프로젝트를 만들면 선택한 Google 애널리틱스 계정에 새 Google 애널리틱스 속성이 생성되고 Firebase 프로젝트에 연결됩니다. 이 연결을 통해 제품 간에 데이터 흐름이 활성화됩니다. Google 애널리틱스 속성에서 Firebase로 내보낸 데이터에는 Firebase 서비스 약관이 적용되지만 Google 애널리틱스로 가져온 Firebase 데이터에는 Google 애널리틱스 서비스 약관이 적용됩니다. [자세히 알아보기](#)

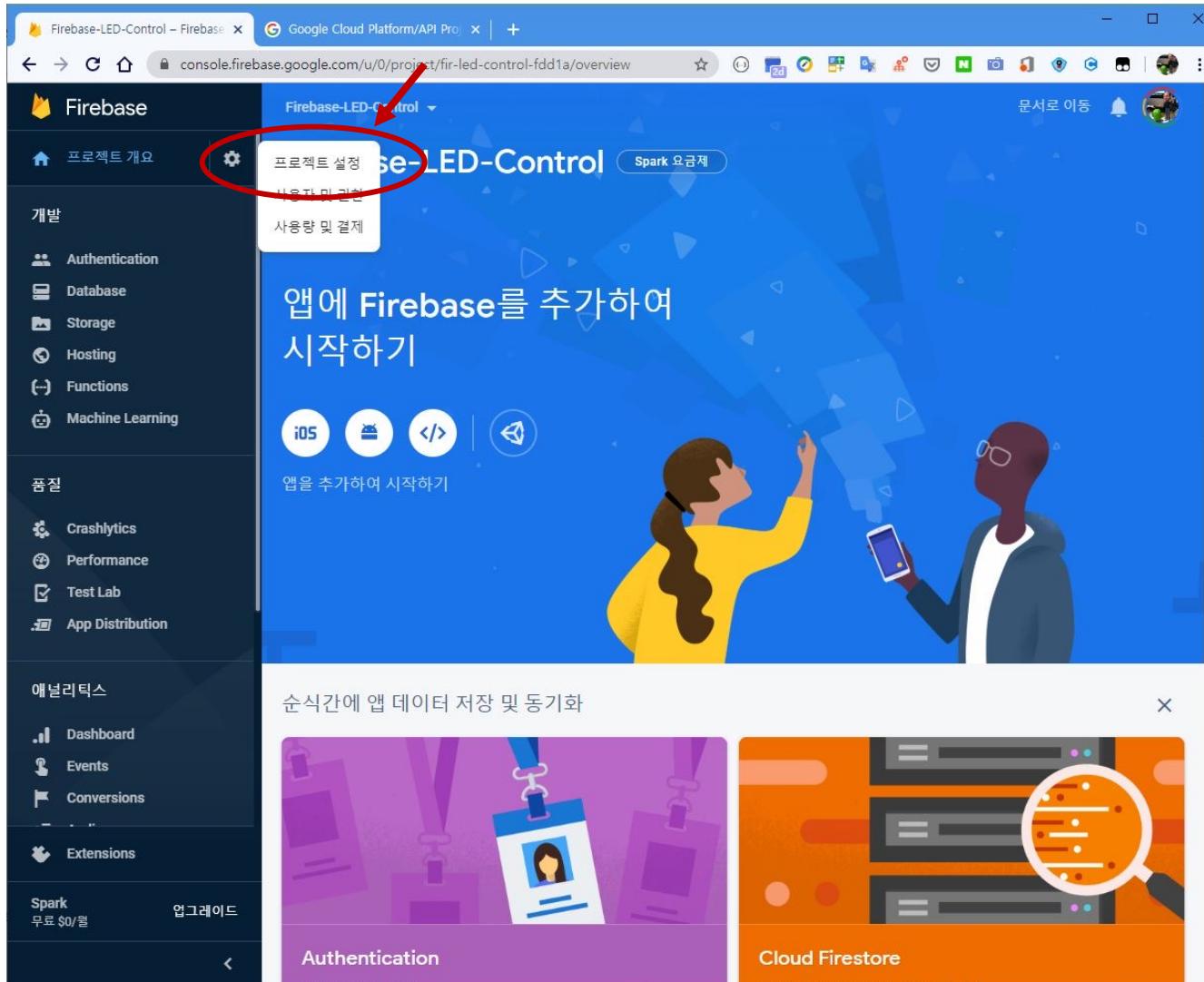
이전

프로젝트 만들기

프로젝트 만들기



프로젝트 설정



데이터베이스 호스트 이름, 비밀번호

Firebase

Firebase-LED-Control

문서로 이동

프로젝트 개요

개발

- Authentication
- Database
- Storage
- Hosting
- Functions
- Machine Learning

서비스 계정

일반

클라우드 메시징

통합

데이터 개인정보 보호

사용자 및 권한

서비스 계정 권한 관리

Firebase Admin SDK

데이터베이스 비밀번호

이전 사용자 인증 정보

데이터베이스 비밀번호

주의: 데이터베이스 비밀번호는 현재 지원이 종단되었으며 기존 Firebase 토큰 생성기를 사용합니다. 소스 코드를 Firebase Admin SDK로 업데이트하세요.

자세히 알아보기

다른 서비스 계정

Google Cloud Platform의 서비스 계정 5개

데이터베이스

fir-led-control-fdd1a

비밀번호

30c4f3hwAZr0WpqW2qX4etDSapBqvbcBu5UcMWyo

비밀번호 추가

Dashboard

데이터 베이스(Realtime Database) 만들기

Firebase

프로젝트 개요

개발

- Database** (highlighted with a red circle)
- Storage
- Hosting
- Functions
- Machine Learning

품질

- Crashlytics
- Performance
- Test Lab
- App Distribution

애널리틱스

- Dashboard
- Events
- Conversions

Extensons

Spark 무료 \$0/월 업그레이드

Realtime Database

Firebase의 기준 데이터베이스입니다. Cloud Firestore와 마찬가지로 실시간 데이터 동기화를 지원합니다.

문서 보기 자세히 알아보기

데이터베이스 만들기

개발자를 위한 추가 기능

Functions

Firebase 기능 확장 및 연결

Storage

사용자 제작 콘텐츠 저장 및 검색

Firebase

프로젝트 개요

개발

- Authentication
- Database**
- Storage
- Hosting
- Functions
- Machine Learning

품질

- Crashlytics
- Performance
- Test Lab
- App Distribution

애널리틱스

- Dashboard
- Events
- Conversions

Extensons

Spark 무료 \$0/월 업그레이드

실시간 데이터베이스 보안 규칙

데이터 구조를 정의한 후 규칙을 작성해 데이터를 보호해야 합니다.

자세히 알아보기

잠금 모드로 시작 모든 일기 및 사용자를 차단하여 데이터베이스를 비공개로 설정하세요.

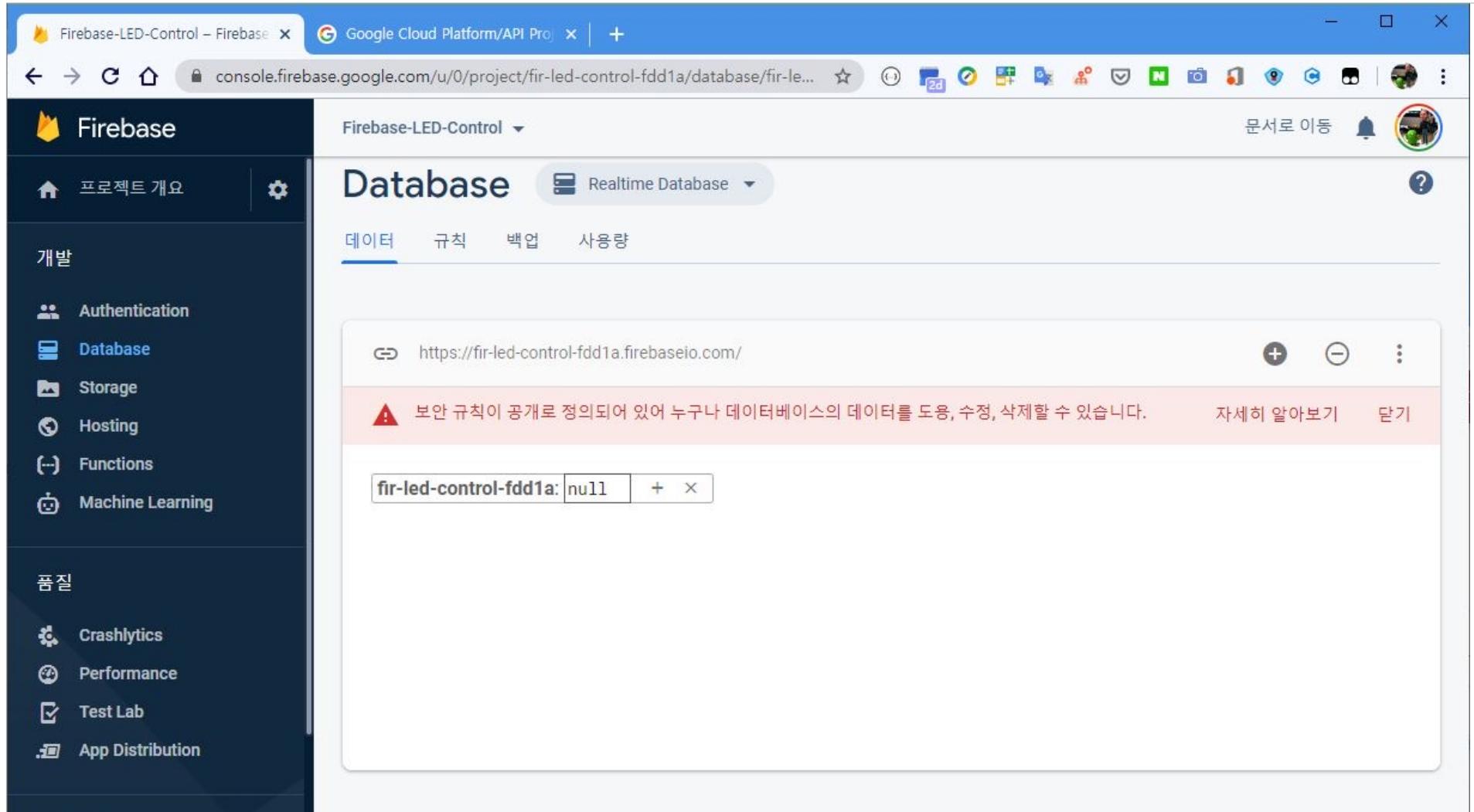
테스트 모드로 시작 데이터베이스에 대한 모든 읽기 및 쓰기를 허용하여 빠르게 설정하세요.

```
{ "rules": { ".read": true, ".write": true } }
```

! 데이터베이스 참조를 사용하는 누구나 데이터베이스를 읽고 쓸 수 있습니다.

취소 사용 설정

데이터베이스 생성



데이터 베이스 값 수정

The screenshot shows the Firebase Realtime Database console in a web browser. The URL in the address bar is `console.firebaseio.google.com/u/0/project/fir-led-control-fdd1a/database.firebaseio.com/`. The left sidebar contains navigation links for various Firebase services: Authentication, Database, Storage, Hosting, Functions, Machine Learning, Crashlytics, Performance, Test Lab, and App Distribution. The main area is titled "Database" and "Realtime Database". It displays a single data entry under the path "fir-led-control-fdd1a":

- Value: null
- Name: LED_STATUS
- Value: OFF

A red oval highlights the "Name" and "Value" fields for the "LED_STATUS" entry. Below the table are buttons for "최소" (Minimum) and "추가" (Add).

아두이노 소스 코드

```
// 아두이노 Firebase 라이브러리
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>

// 데이터 베이스 호스트 이름
#define FIREBASE_HOST "fir-led-controlxxx.firebaseio.com"

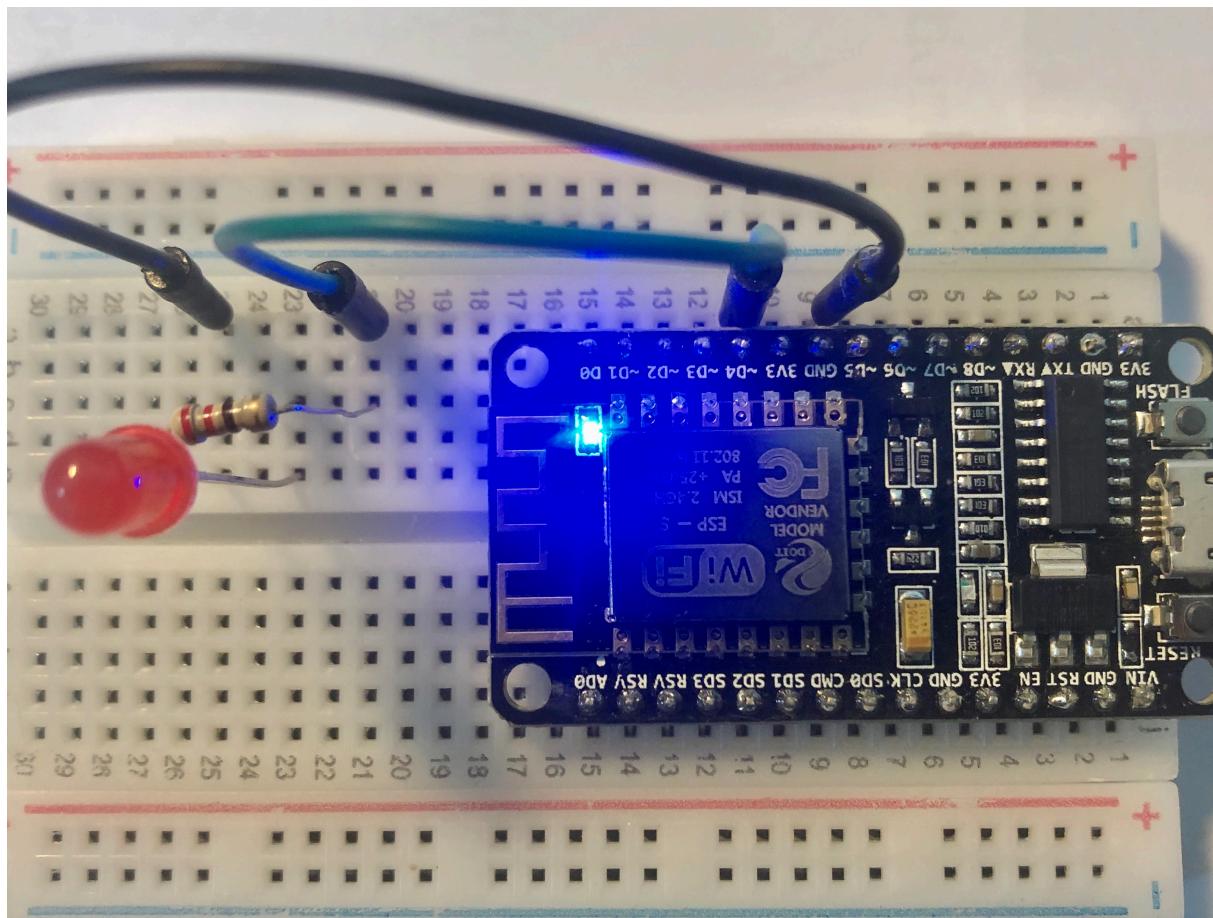
// 데이터 베이스 비밀번호
#define FIREBASE_AUTH "KeiqJV41s*****LdNXL"

// Wi-Fi 설정
#define WIFI_SSID "melon"
#define WIFI_PASSWORD "12345678
```

```
String fireStatus = ""; // firebase DB로 부터 받은 값 저장
int led = D4; // NodeMCU LED pin
void setup()
{
    Serial.begin(9600);
    delay(1000);
    pinMode(led, OUTPUT);
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    Serial.print("Connecting to ");
    Serial.print(WIFI_SSID);
    while (WiFi.status() != WL_CONNECTED) {
        Serial.print(".");
        delay(500);
    }
    Serial.println();
    Serial.print("Connected to ");
    Serial.println(WIFI_SSID);
    Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH); // firebase에 연결
    Firebase.setString("LED_STATUS", "OFF"); //초기 LED 상태를 OFF로 설정
}
```

```
void loop()
{
    fireStatus = Firebase.getString("LED_STATUS"); // Firebase 데이터베이스로 부터 LED 상태 값 읽음
    if (fireStatus == "ON") { // Firebase 데이터베이스로 부터 받은 LED 상태 값이 "ON"이면 LED 켜기
        Serial.println( " Led Turned ON " ); // Serial Monitor에 "Led Turned ON" 문자열 출력
        digitalWrite(led, HIGH); // LED OFF
    } else if (fireStatus == "OFF") { // Firebase 데이터베이스로 부터 LED 상태 값 읽음
        Serial.println("Led Turned OFF"); // Serial Monitor에 "Led Turned OFF" 문자열 출력
        digitalWrite(led, LOW); // LED OFF
    } else {
        Serial.println("Command Error! Please send ON/OFF");
    }
}
```

실행 : 초기 상태



← → C ⌂ https://console.firebaseio.google.com/u/0/project/led-on-off-2020-b7a1e/database/l...

Firebase LED-ON-OFF-2020 Database Realtime Database 문서로 이동 ?

프로젝트 개요 | 설정

개발

- Authentication
- Database
- Storage
- Hosting
- Functions
- Machine Learning

품질

- Crashlytics
- Performance
- Test Lab
- App Distribution

Extensions

Spark 무료 \$0/월 업그레이드

데이터 규칙 백업 사용량

https://led-on-off-2020-b7a1e.firebaseio.com/ + - :

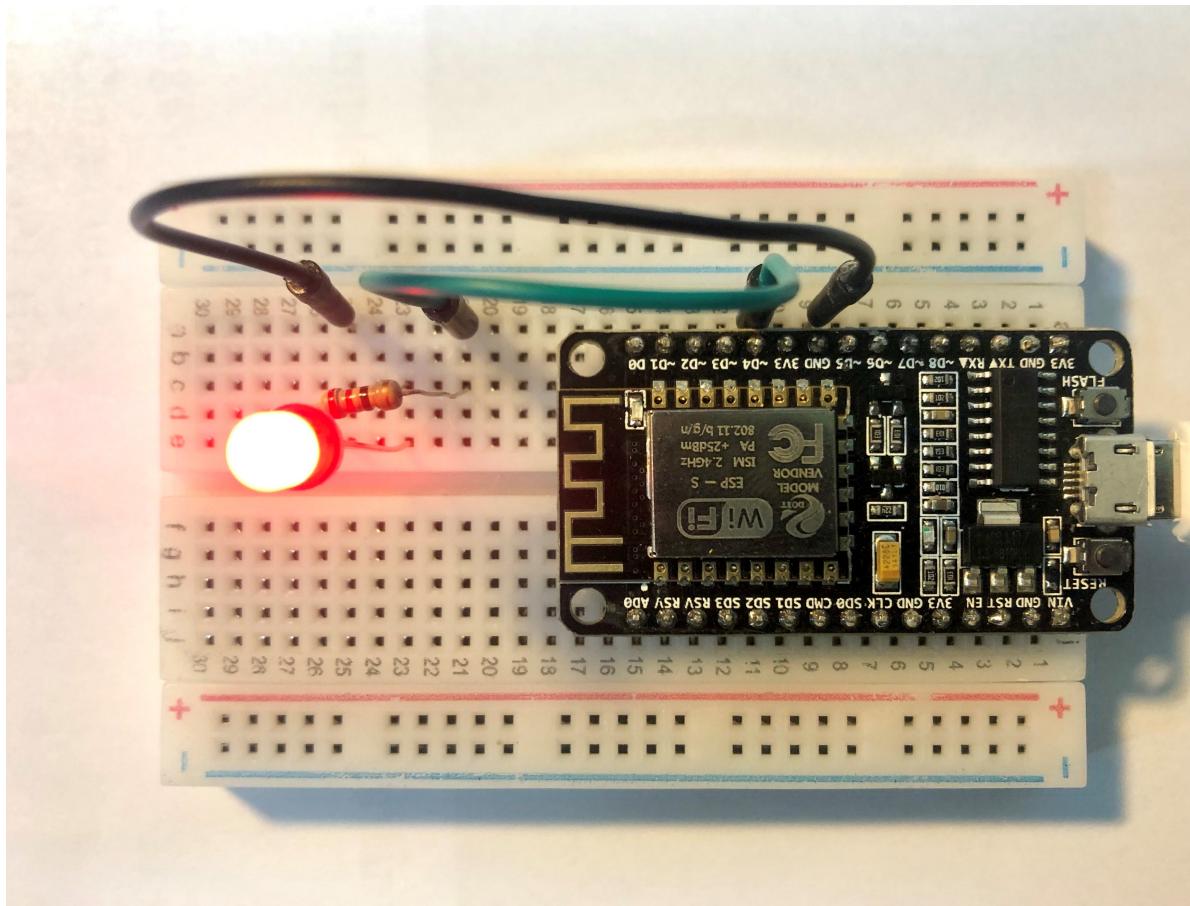
⚠ 보안 규칙이 공개로 정의되어 있어 누구나 데이터베이스의 데이터를 도용, 수정, 삭제할 수 있습니다. 자세히 알아보기 닫기

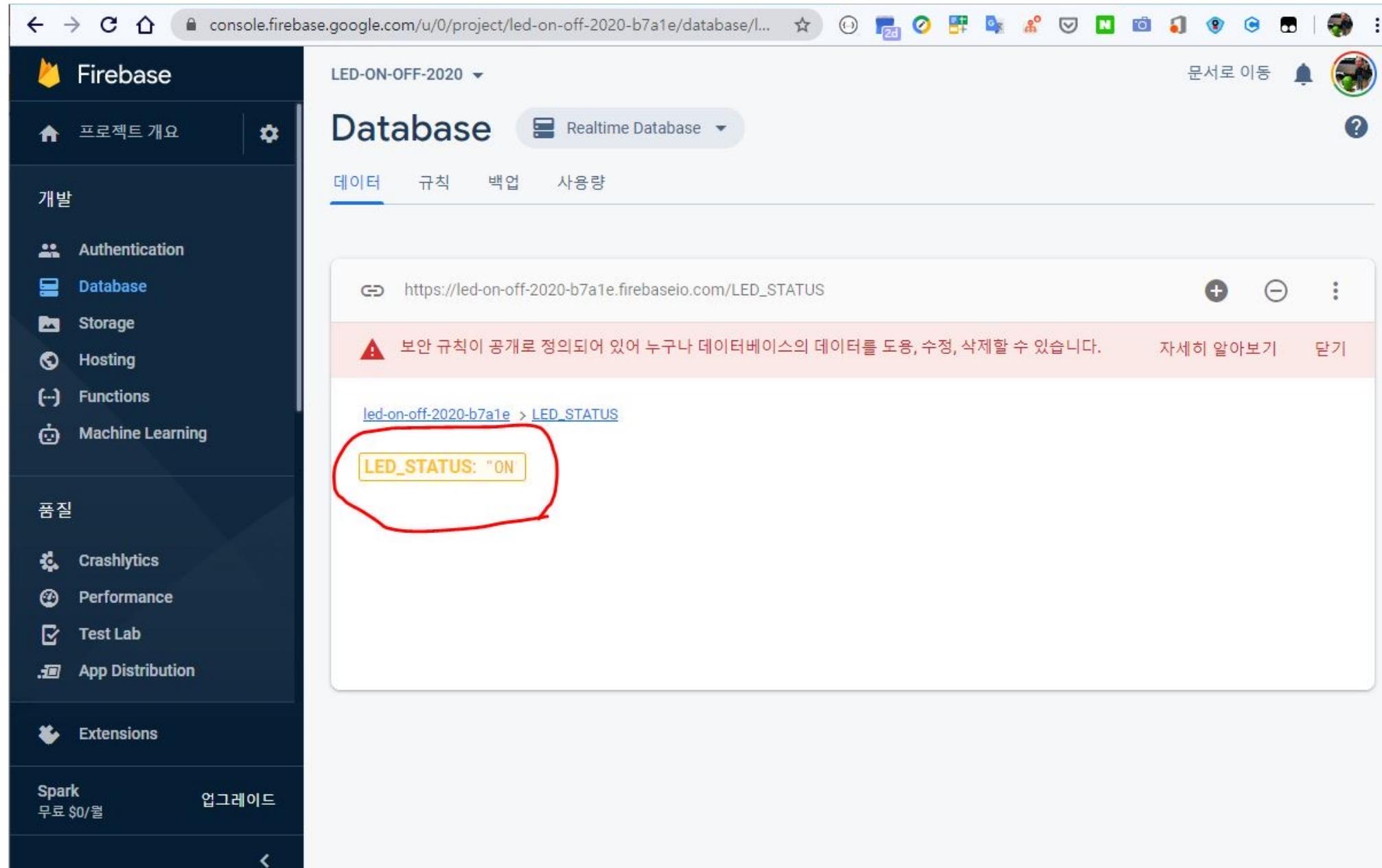
led-on-off-2020-b7a1e

LED_STATUS: "OFF"



실행 : Firebase DB “ON” update

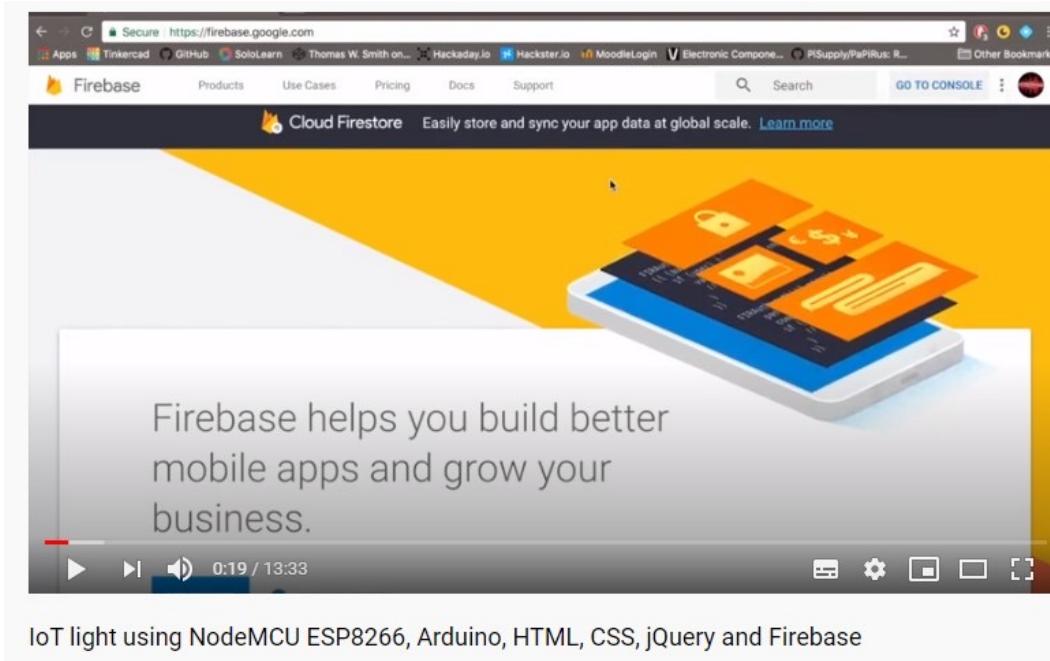




학습 참고

- 학습 YouTube 동영상

<https://youtu.be/usa7lcZeUMc>



- 소스 코드 다운로드

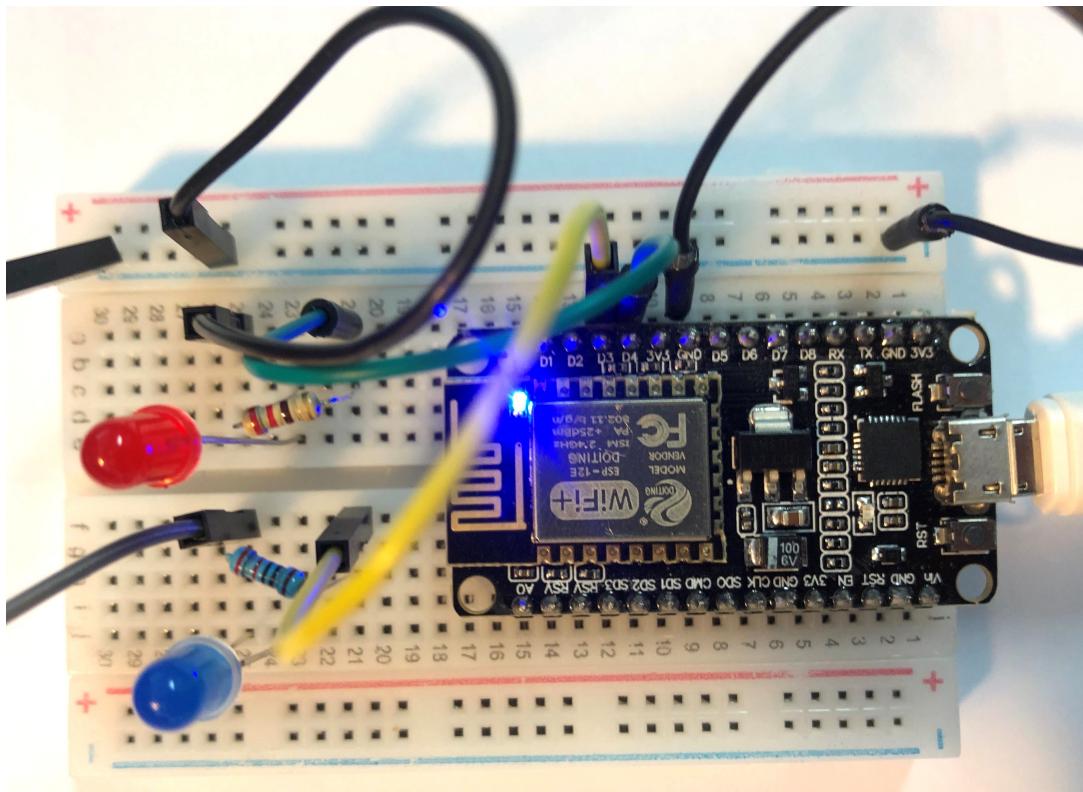
<https://github.com/IoT-Lab-02/Week09-Lab>

수업 소스 코드 : 2개의 LED 제어

<https://github.com/IoT-Lab-02/Week09-Lab>

기말 과제 01

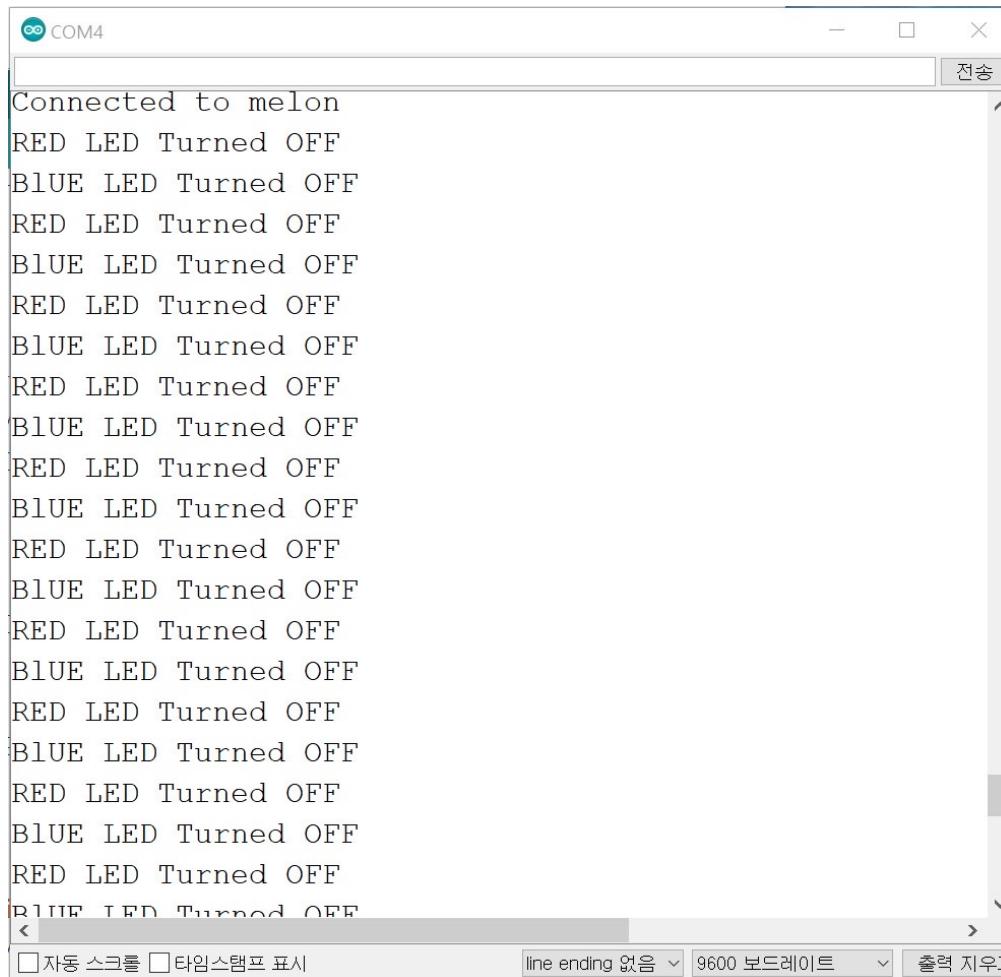
- Firebase DB로 RED LED, BLUE LED ON/OFF를 개별적으로 제어하기
- 2개의 LED 모두 OFF : 초기 상태



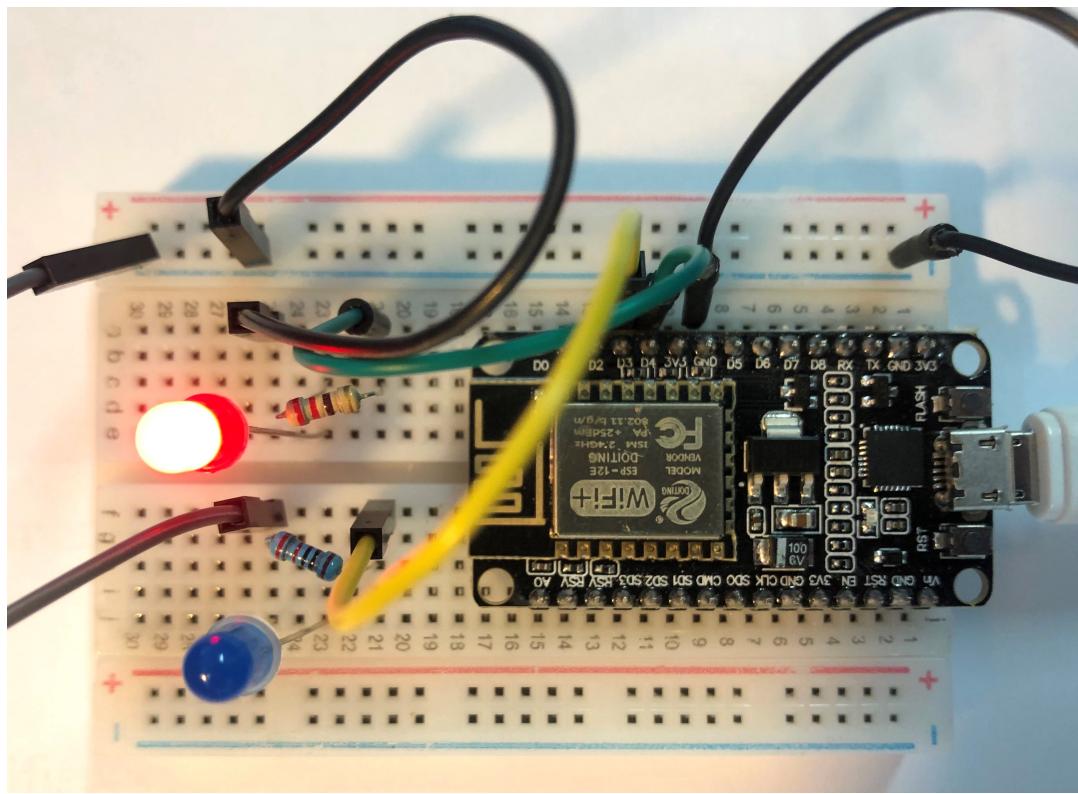
The screenshot displays the Firebase Realtime Database interface. On the left, the database structure is shown with a single node named "led-on-off-2020-c1453". Under this node, there are two child nodes: "BLUE_LED_STATUS" and "RED_LED_STATUS", both of which have the value "OFF". The right side of the screen shows the URL <https://led-on-off-2020-c1453.firebaseio.com/>.

```
led-on-off-2020-c1453
  |- BLUE_LED_STATUS: "OFF"
  |- RED_LED_STATUS: "OFF"
```

- Serial Monitor(PC) : RED LED OFF/ BLUE LED OFF

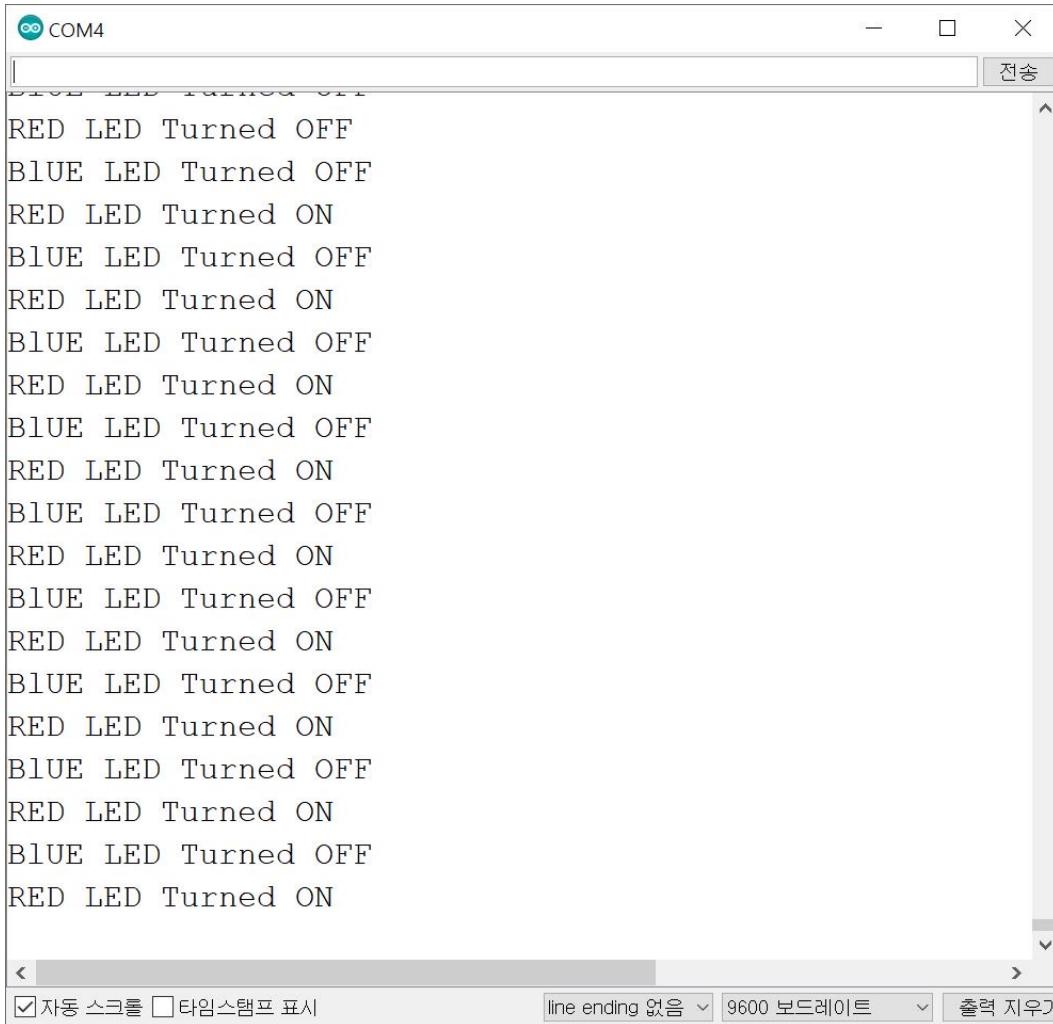


- RED LED ON/ BLUE LED OFF

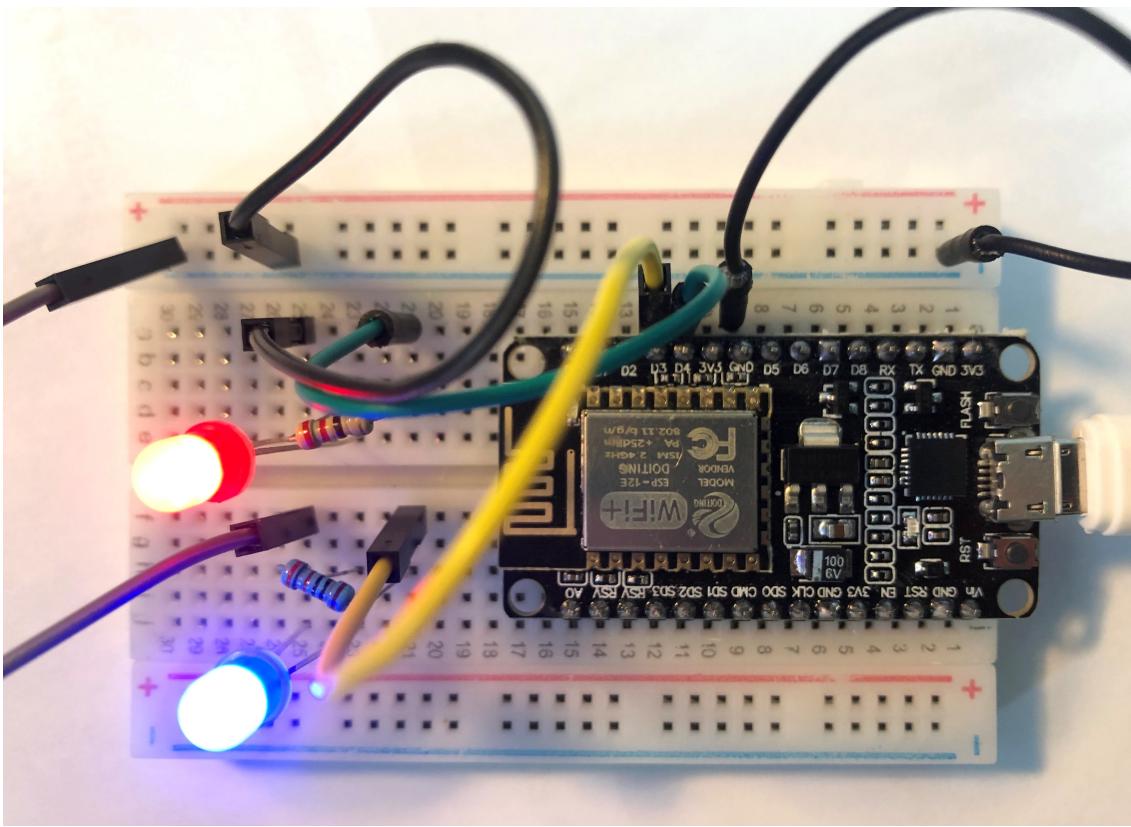


```
led-on-off-2020-c1453
  +-- BLUE_LED_STATUS: "ON"
  +-- RED_LED_STATUS: "ON"
```

- Serial Monitor(PC) : RED LED ON/ BLUE LED OFF



- RED LED ON/ BLUE LED ON

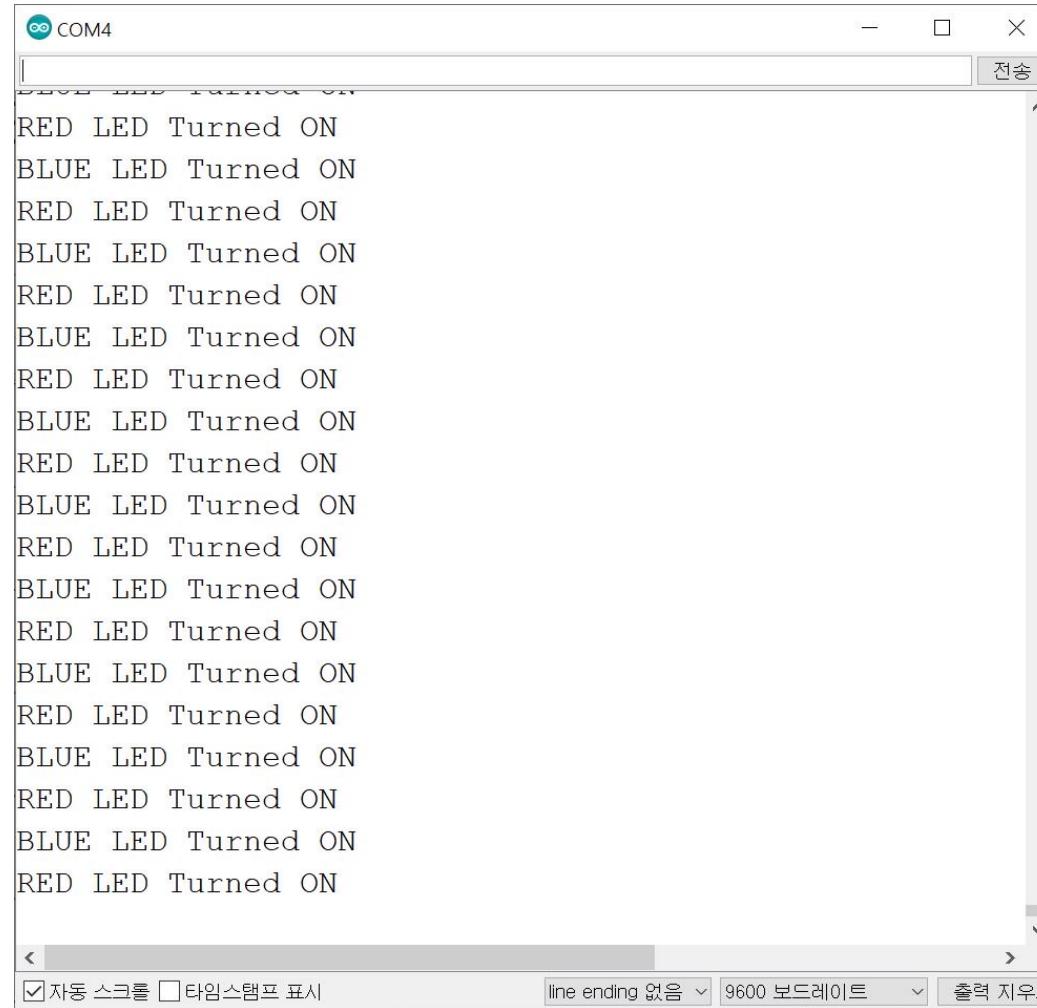


The screenshot shows the Firebase Realtime Database interface. The project name is "2-LED-ON-OFF-2020". The database structure is as follows:

```
led-on-off-2020-c1453
  |- BLUE_LED_STATUS: "ON"
  |- RED_LED_STATUS: "ON"
```

A warning message in Korean is displayed: "보안 규칙이 공개로 정의되어 있어 누구나 데이터베이스의 데이터를 도용, 수정, 삭제할 수 있습니다." (Security rules are publicly defined, so anyone can abuse, modify, or delete the data in the database).

- Serial Monitor(PC) : RED LED ON/ BLUE LED ON



기말과제 01 제출

- 향후 기말과제가 약 4~5개 출제 예정입니다.
- 기말과제는 중요하므로 일반과제 보다 배점이 높고 성적에 영향을 많이 끼칩니다..
 - 예 : 일반과제 5점(1개), 기말과제 20점(1개)
- 제출기한
 - 6월 15일(월) 까지
- 제출물
 - 소스 코드, 동작 촬영 동영상