

Firestore와 NodeMCU

2023학년도 2학기

김진숙 교수

동의과학대학 컴퓨터정보과

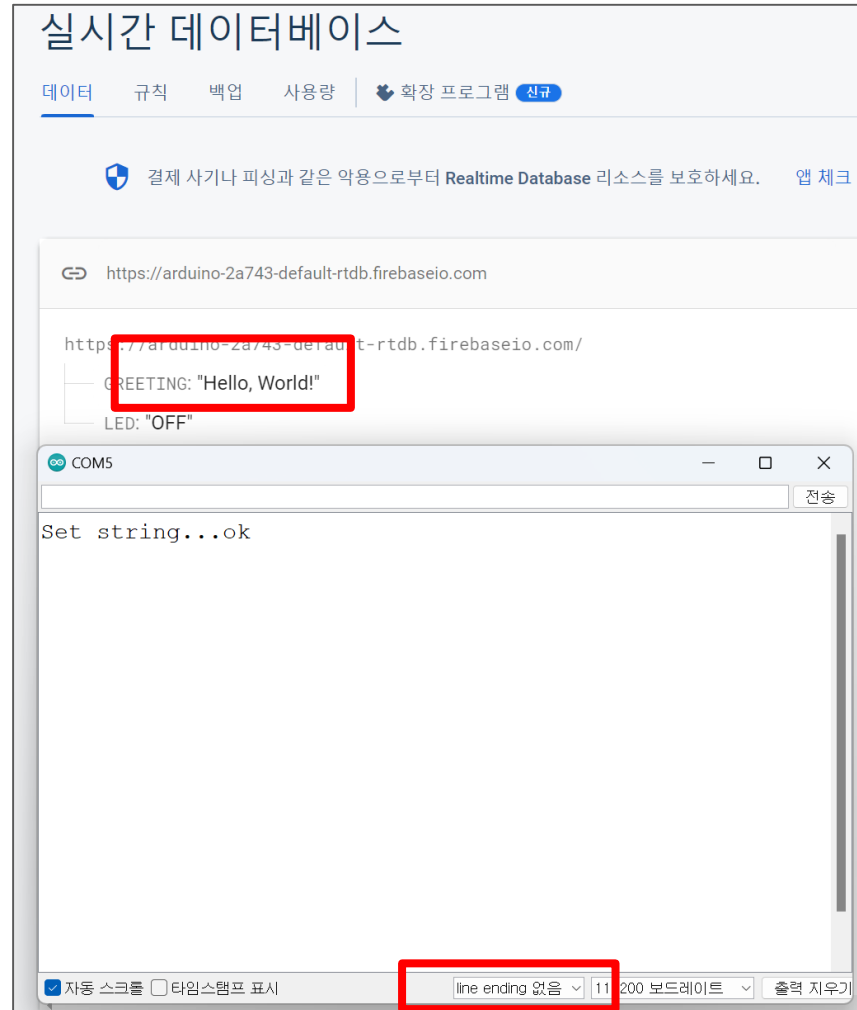
jinsook@dit.ac.kr

학습 내용

- 구글 파이어베이스 문자 데이터 웹 서비스 하기
- 구글 파이어베이스 정보로 웹에서 LED ON/OFF 하기
- 구글 파이어베이스 온습도 정보를 웹에 서비스 하기

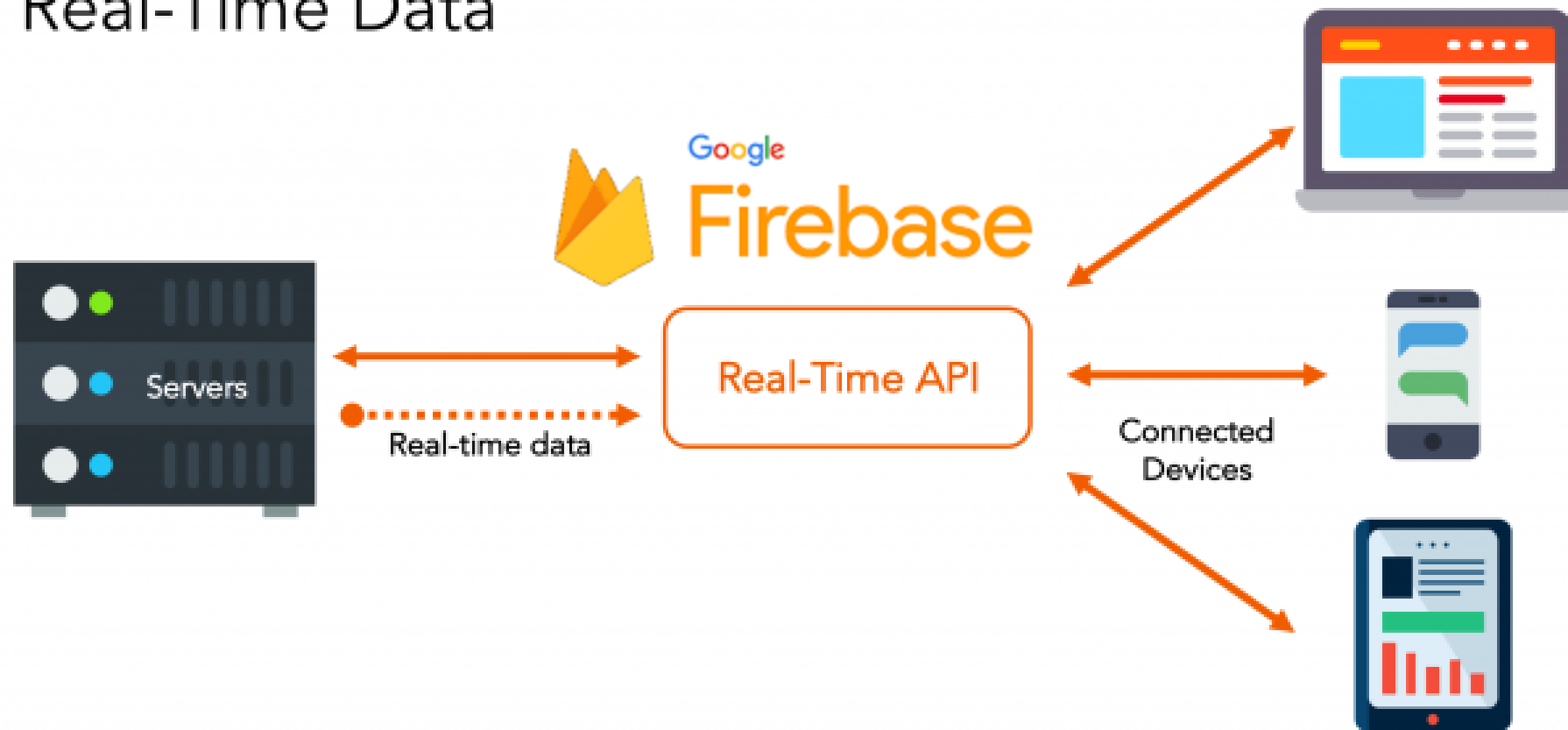
구글 파이어베이스에 문자보내기

- 주제 : 구글 파이어베이스 실시간 데이터베이스에 문자 보내기
 - 구글의 실시간 데이터베이스인 파이어베이스에 nodeMCU에서 "Hello, Worl!" 문자 보내기



Firebase Real-Time Database : NoSQL

Real-Time Data



구글 파이어베이스에 문자보내기

2) 프로그래밍

```
1 #include <ESP8266WiFi.h>
2 #include <FirebaseESP8266.h>
3
4 // 와이파이 설정
5 #define WIFI_SSID "<ssid>"
6 #define WIFI_PASSWORD "<password>"
7
8 // 파이어베이스 API 키
9 #define FIREBASE_HOST "arduino-2a743-default-rtdb.firebaseio.com"
10 #define FIREBASE_AUTH "qv402KvQhYD29oVkegF2uY6xEVvn1XpBv1kAF4av"
11
12 // 파이어베이스 데이터 객체 생성
13 FirebaseData fbdo;
14
15 void setup() {
16     Serial.begin(115200);
17     WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
18
19     while (WiFi.status() != WL_CONNECTED) {
20         Serial.print(".");
21         delay(300);
22     }
23     Serial.println();
24     Serial.print("Connected with IP: ");
25     Serial.println(WiFi.localIP());
26     Serial.println();
27
28     //파이어베이스 연결
29     Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
30 }
```

구글 파이어베이스에 문자보내기

2) 프로그래밍

```
32 void loop() {  
33   if(Serial.available()){  
34     //시리얼모니터에 line ending 없음을 선택한 뒤 실행할 것  
35     String greeting = Serial.readStringUntil('\r');  
36  
37     //시리얼모니터에 입력한 내용을 파이어베이스에 입력하기  
38     bool res = Firebase.setString(fbdo, "/GREETING", greeting);  
39     if(res){  
40       Serial.println("Set string...ok");  
41     }else{  
42       Serial.println(fbdo.errorReason().c_str());  
43     }  
44  
45     //Serial.printf("Set string... %s\n", Firebase.setString(fbdo, F("/GREETING"), greeting) ? "ok" : fbdo.errorReason().c_str());  
46   }  
47 }
```

구글 파이어베이스에 문자보내기

3) 실행

The image shows two overlapping windows. The background window is the Google Firebase Realtime Database console for a database named '실시간 데이터베이스'. It displays the URL `https://arduino-2a743-default-rtdb.firebaseio.com/` and a JSON snapshot of the data: `{ "GREETING": "Hello, World!", "LED": "OFF" }`. The 'GREETING' value is highlighted with a red box. The foreground window is the Arduino IDE's serial monitor, connected to COM5. It shows the output `Set string...ok`. At the bottom of the serial monitor, the 'line ending' is set to '없음' (None) and the 'baud rate' is set to '115200', both of which are highlighted with a red box.

구글 파이어베이스에 문자보내기

The image shows the Firebase console interface. At the top, the 'Firebase' logo is visible. Below it, the text 'Firebase 프로젝트' (Firebase Project) is displayed. A large white card with a blue plus sign and the text '프로젝트 추가' (Add Project) is prominently shown. To the right, a card for the 'testweb' project (ID: testweb-7c77e) is visible. Below the 'Add Project' card, a sidebar menu is shown with options like '프로젝트 개요' (Project Overview), '제품 카테고리' (Product Category), '빌드' (Build), '출시 및 모니터링' (Release and Monitoring), '애널리틱스' (Analytics), and '참여' (Engagement). The main content area shows the 'testweb' project details, including a 'Spark 요금제' (Spark Plan) button and a red box highlighting the '</>' icon, which is used to add a new app. Below this, the text '앱을 추가하여 시작하기' (Add app to get started) is visible. At the bottom, the text '순식간에 앱 데이터 저장 및 동기화' (Save and sync app data in seconds) is displayed.

구글 파이어베이스 문자 웹서비스 하기

■ 프로젝트 설정 - 웹앱 추가

[참고]이미 프로젝트가 만들어진 경우

×

웹 앱에 Firebase 추가

1 앱 등록

앱 닉네임

myweb

☐ 또한 이 앱의 **Firebase** 호스팅을 설정하세요. [자세히 알아보기](#)

앱 등록

2 Firebase SDK 추가

Firebase

WebTest01

프로젝트 설정

프로젝트 개요

프로젝트 설정

사용자 및 권한

사용량 및 결제

제품 카테고리

빌드

출시 및 모니터링

애널리틱스

참여

모든 제품

이름

WebTest01

프로젝트 ID

webtest01-c32fc

프로젝트 번호

101028813653

기본 GCP 리소스 위치

아직 선택하지 않음

웹 API 키

이 프로젝트에 웹 API 키가 없습니다.

환경

이 설정은 앱 수명 주기의 여러 단계에 맞게 프로젝트를 맞춤설정합니다.

환경 유형

미지정

내 앱

프로젝트에 앱이 없습니다.

시작하려면 플랫폼을 선택하세요.

iOS+

Android

Web

Flutter

React Native

9

구글 파이어베이스 문자 웹서비스 하기

■ 프로젝트 설정 - 웹앱 추가

2 Firebase SDK 추가

☐ npm 사용 ☒ <script> 태그 사용

빌드 도구를 사용하지 않는 경우 이 옵션을 사용하여 Firebase JS SDK를 추가하고 사용하세요. 시작하는 데 이 옵션을 사용할 수 있지만 프로젝트 앱에는 권장되지 않습니다. [자세히 알아보기](#)

스크립트를 복사하여 <body> 태그 하단에 붙여넣으세요. Firebase 서비스를 사용하기 전에 진행해야 합니다.

```
<script type="module">
// Import the functions you need from the SDKs you need
import { initializeApp } from "https://www.gstatic.com/firebasejs/10.6.0/firebase-app.js";
import { getAnalytics } from "https://www.gstatic.com/firebasejs/10.6.0/firebase-analytics.js";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
// For Firebase JS SDK v7.20.0 and later, measurementId is optional
const firebaseConfig = {
  apiKey: "AIzaSyDAVrCrvrrABHAV-S5-wIshhojeFGBERC8",
  authDomain: "testweb-7c77e.firebaseio.com",
  projectId: "testweb-7c77e",
  storageBucket: "testweb-7c77e.appspot.com",
  messagingSenderId: "748891772285",
  appId: "1:748891772285:web:34d0e6bd651c958ca2a814",
  measurementId: "G-YX3E9ZQNVV"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);
</script>
```

npm 및 번들러(예: Webpack 또는 Rollup)를 사용 중이신가요? [모듈형 SDK](#)를 확인하세요.

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

콘솔로 이동

 **Firebase**

프로젝트 개요

제품 카테고리

빌드

출시 및 모니터링

애널리틱스

참여

모든 제품

testweb

앱 1개

+ 앱 추가

구글 파이어베이스 문자 웹서비스 하기

■ 프로젝트 설정 - 웹앱 추가

The screenshot shows the Firebase console interface. On the left sidebar, the '프로젝트 설정' (Project Settings) tab is selected and highlighted with a red box. The main content area displays the 'testweb' project settings, including the project name, ID, and version. Below this, the '환경' (Environment) section shows the project's configuration. At the bottom, the '웹 앱' (Web App) section is visible, showing a list of web apps with a red box around the 'myweb' entry. To the right of the 'myweb' entry, the '연 ID' (App ID) is displayed, and a button labeled 'Firebase 호스팅 사이트에 연결' (Connect to Firebase Hosting site) is visible. Below this, the 'SDK 설정 및 구성' (SDK Setup and Configuration) section shows the 'CDN' option selected, and a code snippet for initializing the Firebase SDK is provided.

구글 파이어베이스 문자 웹서비스 하기

■ 프로젝트 설정 - 웹앱 추가

The screenshot shows the Firebase console interface. On the left, the 'testweb' project is selected, and the '프로젝트 설정' (Project Settings) tab is active, highlighted with a red box. The main area displays project details: Project Name (testweb), Project ID (testweb-7c77e), Project Number (748891772285), and the Web API Key (not set). Below this, the '환경' (Environment) section shows a warning that the project is not yet configured for production. At the bottom, the '내 앱' (My Apps) section is visible, showing a list of apps with a red box around the 'myweb' app. To the right of the 'myweb' app, the '연 추가' (Add App) button is visible. The '연 추가' button is also highlighted with a red box. Below the '연 추가' button, the 'SDK 설정 및 구성' (SDK Setup and Configuration) section is shown, with the 'CDN' option selected. The 'CDN' option is highlighted with a red box. The 'CDN' section includes instructions on how to use the Firebase JavaScript SDK and a code snippet for the SDK setup.

Firebase console interface showing project settings and adding a new web app.

Project Settings (testweb):

- 프로젝트 이름: testweb
- 프로젝트 ID: testweb-7c77e
- 프로젝트 번호: 748891772285
- 기본 GCP 리소스 위치: 아직 선택하지 않음
- 웹 API 키: 이 프로젝트에 웹 API 키가 없습니다.

환경 (Environment):

이 설정을 사용하여 수백 개의 다른 장치에 맞게 프로젝트를 맞춤 설정합니다.

환경 유형: 미지정

내 앱 (My Apps):

- myweb (선택)

연 추가 (Add App):

연 ID: 1:748891772285:web:34d0e6bd651c958ca2a814

[Firebase 호스팅 사이트에 연결](#)

SDK 설정 및 구성:

- ☐ npm
- ☒ CDN
- ☐ 구성

CDN(콘텐츠 전송 네트워크)에서 Firebase JavaScript SDK 라이브러리를 로드하세요.
[자세히 알아보기](#)

스크립트를 복사하여 <body> 태그 하단에 붙여넣으세요. Firebase 서비스를 사용하기 전에 진행해야 합니다.

```
<script type="module">
// Import the functions you need from the SDKs you need
import { initializeApp } from "https://www.gstatic.com/firebasejs/
import { getAnalytics } from "https://www.gstatic.com/firebasejs/1
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries
```

구글 파이어베이스 문자 웹서비스 하기

■ 웹페이지 작성

```
firebase-greeting.html x
firebase-greeting.html > {} "firebase-greeting.html" > html > head
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <title>nodeMCU - Firebase</title>
6   </head>
7   <body>
8     <h2>Message Control Wep App</h2>
9     <h1 id="OUT_TEXT"></h1>
10
11     <script type="module">
12       // Import the functions you need from the SDKs you need
13       import { initializeApp } from "https://www.gstatic.com/firebasejs/10.5.0/firebase-app.js";
14       import { getDatabase, ref, set, get, onValue, child } from "https://www.gstatic.com/firebasejs/10.5.0/firebase-database.js";
15
16       // TODO: Add SDKs for Firebase products that you want to use
17       // https://firebase.google.com/docs/web/setup#available-libraries
18
19       // Your web app's Firebase configuration
20       // For Firebase JS SDK v7.20.0 and later, measurementId is optional
21       const firebaseConfig = {
22         apiKey: "AIzaSyDbYTsiuUI7IIQ12zefAzBxWn6KuKOAR1Y",
23         authDomain: "fir-led-control-1ddb6.firebaseio.com",
24         databaseURL: "https://fir-led-control-1ddb6-default-rtdb.firebaseio.com",
25         projectId: "fir-led-control-1ddb6",
26         storageBucket: "fir-led-control-1ddb6.appspot.com",
27         messagingSenderId: "1057751920144",
28         appId: "1:1057751920144:web:68840f2d9fa14d3dba6495",
29         measurementId: "G-XJBVSEB9D"
30       };
31     </script>
```

구글 파이어베이스 문자 웹서비스 하기

- 웹페이지 작성

```
32
33      // Initialize Firebase
34      const app = initializeApp(firebaseConfig);
35      const db = getDatabase(app);
36      console.log(db);
37
38      const dbRef = ref(db, 'GREETING');
39      onValue(dbRef, (snapshot) => {
40          console.log(snapshot.val());
41          OUT_TEXT.innerText = 'Greeting : ' + snapshot.val();
42      })
43
44      </script>
45
46      </body>
47  </html>
48
```

Firebase

프로젝트 개요

프로젝트 바로가기

Authentication

Realtime Database

Test Lab

App Check

제품 카테고리

빌드

출시 및 모니터링

애널리틱스

Firebase-LED-Control

실시간 데이터베이스

데이터
 규칙
 백업
 사용량
 확장 프로그램

보안 규칙이 공개로 정의되어 있어 누구나

https://fir-led-control-1ddb6-default-rtadb.firebaseio.com/

GREETING: "hello, world!"

LED_STATUS: "OFF"

nodeMCU - Firebase
 Firebase-LED-Control - 실시간

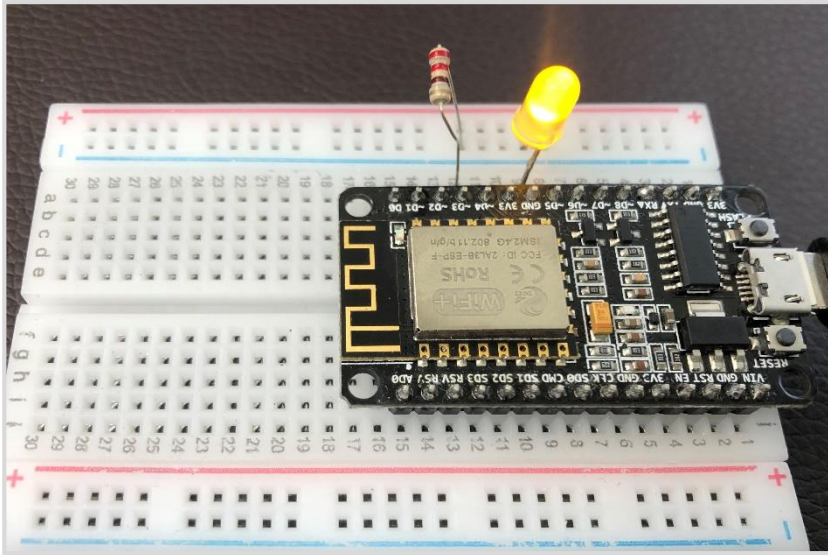
주의 요함 | 203.237.244.21:5501/firebase-greeting.html

Message Control Wep App

Greeting : hello, world!

구글 Firebase의 정보로 LED ON/OFF 하기

- 주제 : 구글 파이어베이스에 제어문자 읽어와 실행하기
 - 구글의 실시간 데이터베이스인 파이어베이스에서 LED 제어하는 문자 on/off를 읽어와 led 켜고 끄기
- 준비사항
 - 파이어베이스 설정
 - 라이브러리 : <FirebaseESP8266.h>, <ESP8266WiFi.h>



< NodeMCU IoT 디바이스 >



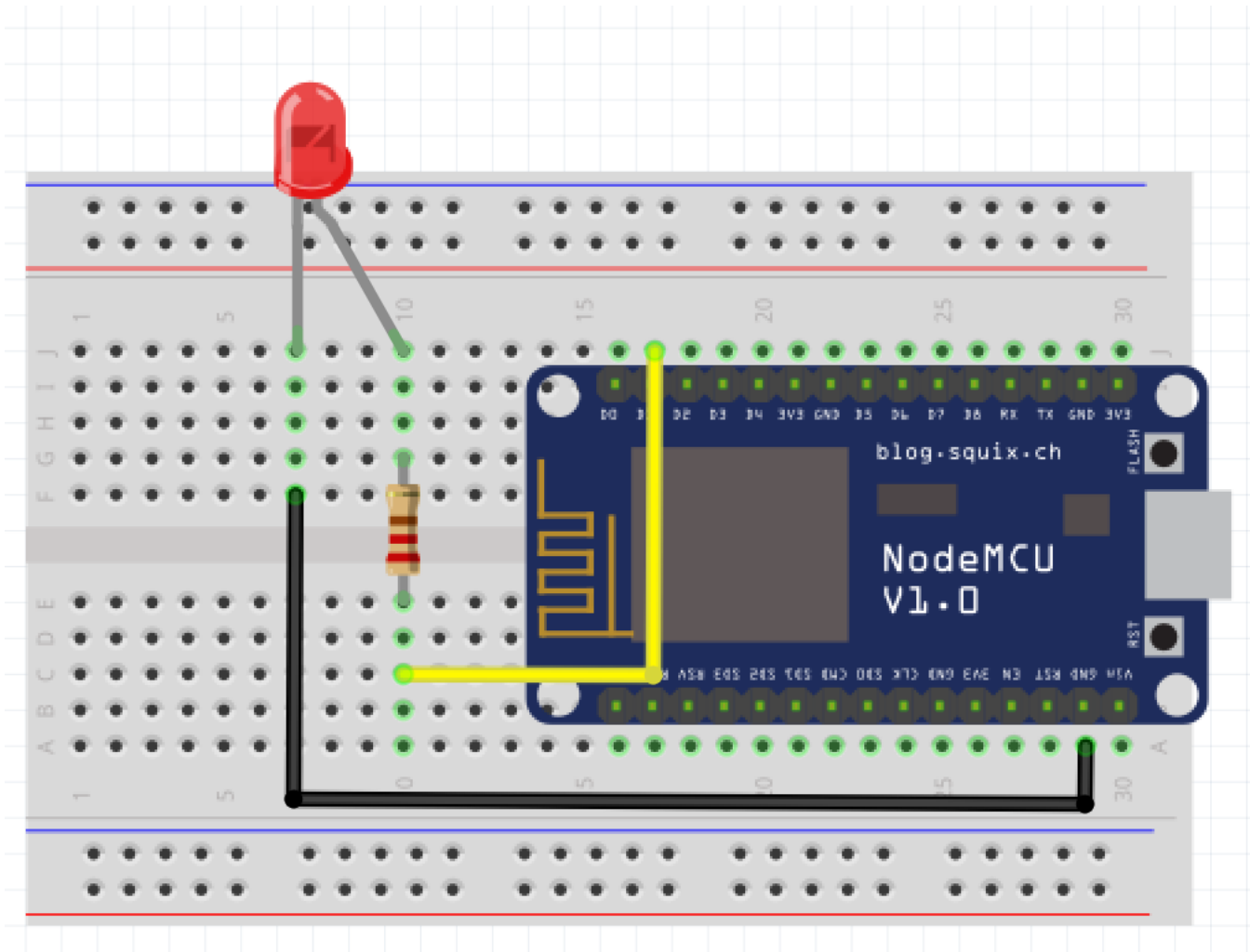
< Firebse IoT 웹 앱 >



< Firebase Realtime Database >

구글 파이어베이스 정보로 LED ON/OFF 하기

1) 회로 구성



구글 파이어베이스 정보로 LED ON/OFF 하기

2) 프로그래밍

```
1 #include <ESP8266WiFi.h>
2 #include <FirebaseESP8266.h>
3
4 #define FIREBASE_HOST "fir-led-control-1ddb6-default-rtdb.firebaseio.com"
5 #define FIREBASE_AUTH "AIzaSyDbYTsiuUI7IIQ12zefAzBxWn6KuKOAR1Y"
6
7 #define WIFI_SSID "DIT_CS_703"
8 #define WIFI_PASSWORD ""
9 int led = D2;
10
11 FirebaseData firebaseData;
12
13 String send_data = "OFF";
14 String read_data = "";
15
16 void setup() {
17     Serial.begin(115200);
18     pinMode(led, OUTPUT);
19
20     WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
21     Serial.print("Connecting to ");
22     Serial.print(WIFI_SSID);
23
24     while (WiFi.status() != WL_CONNECTED) {
25         Serial.print(".");
26         delay(500);
27     }
28     Serial.print("Connected to "); Serial.println(WIFI_SSID);
29     Serial.print("IP Address is : "); Serial.println(WiFi.localIP());
30
31     Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
32     Firebase.reconnectWiFi(true);
33     Serial.println();
34     delay(1000);
35 }
```

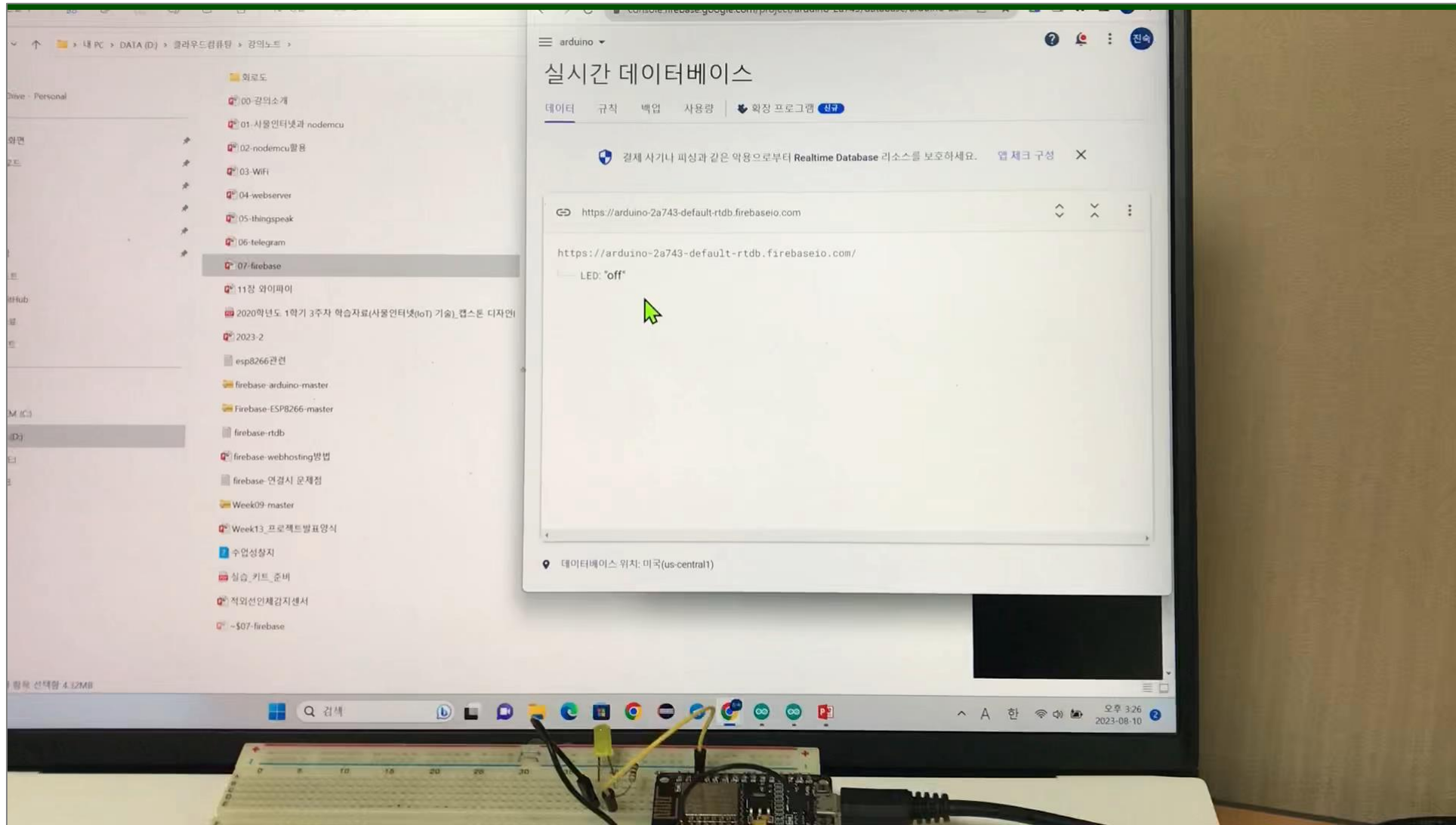
구글 파이어베이스 정보로 LED ON/OFF 하기

2) 프로그래밍

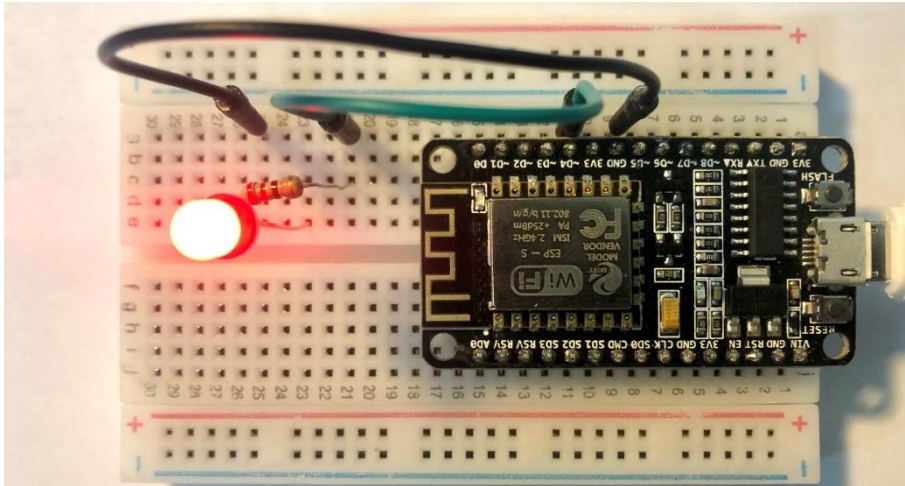
```
37 void loop() {  
38  
39   if(Firebase.getString(firebaseData, "LED_STATUS")) {  
40     read_data = firebaseData.stringData();  
41     Serial.print("Read_data = ");  
42     Serial.println(read_data);  
43  
44     if(read_data == "ON"){  
45       digitalWrite(led, HIGH);    // LED active low  
46       Serial.println("LED turned ON");  
47     }else if(read_data == "OFF") {  
48       digitalWrite(led, LOW);  
49       Serial.println("LED turned OFF");  
50     }  
51     Serial.println();  
52   }else{  
53     Serial.println(firebaseData.errorReason());  
54   }  
55   delay(3000);  
56 }
```

구글 파이어베이스 정보로 LED ON/OFF 하기

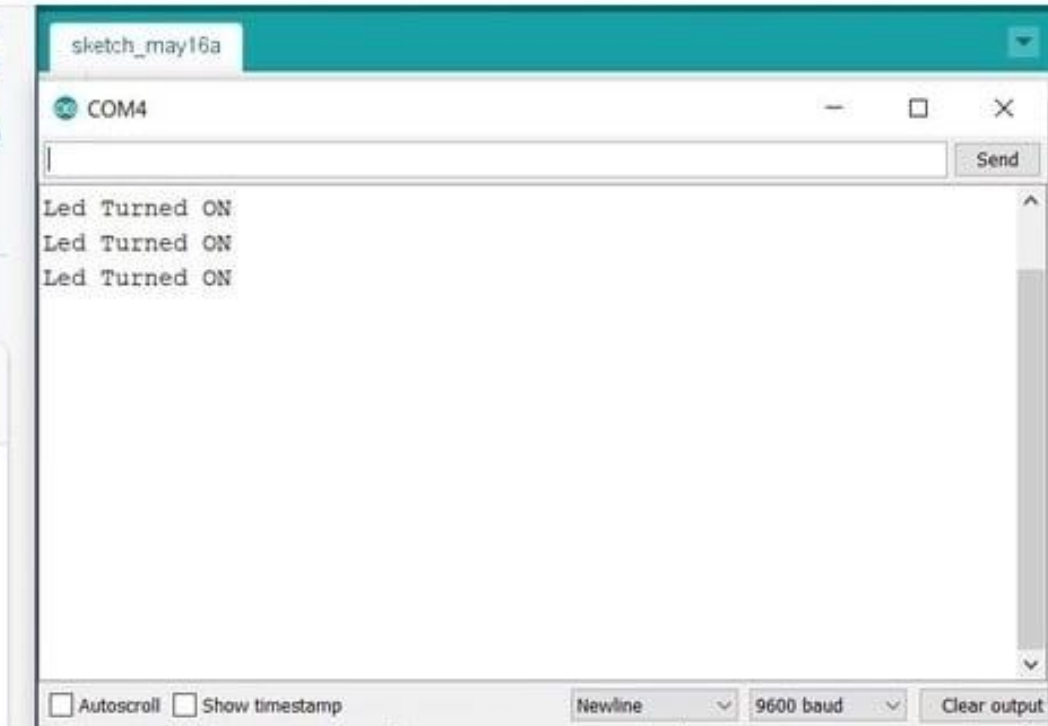
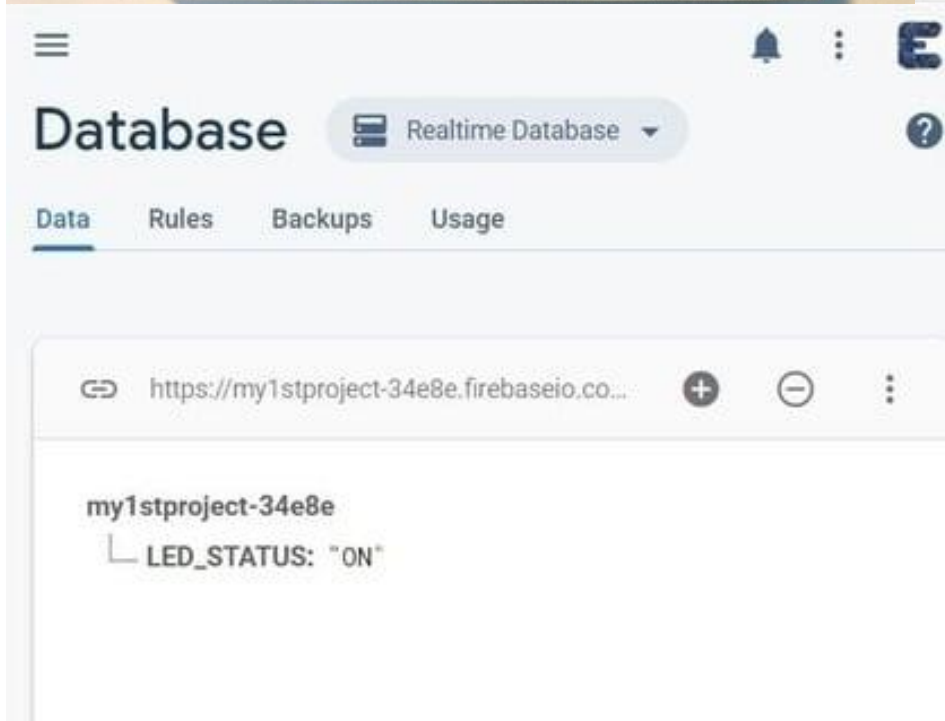
3) 실행



구글 파이어베이스 정보로 LED ON/OFF 하기



Firebase



웹에서 LED 제어하기

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>nodeMCU - Firebase</title>
    <style>
      .on-button {
        background-color: red;
        border: none;
        color: white;
        padding: 15px 30px;
        text-align: center;
        text-decoration: none;
        display: inline-block;
        font-size: 30px;
        margin: 4px 2px;
        cursor: pointer;
      }
      .off-button {
        background-color: blue;
        border: none;
        color: white;
        padding: 15px 30px;
        text-align: center;
        text-decoration: none;
        display: inline-block;
        font-size: 30px;
        margin: 4px 2px;
        cursor: pointer;
      }
    </style>
  </head>
```

```
<body>
  <h2>IoT Cloud Remote LED Control Wep App</h2>
  <h1 id="OUT_TEXT"></h1>

  <button class="on-button" id = "onBtn" >LED ON</button>
  <button class="off-button" id = "offBtn" >LED OFF</button>

  <script type="module">
    // Import the functions you need from the SDKs you need
    import { initializeApp } from
    "https://www.gstatic.com/firebasejs/10.5.0/firebase-app.js";
    import { getDatabase, ref, set, get, onValue, child } from
    "https://www.gstatic.com/firebasejs/10.5.0/firebase-database.js";

    const firebaseConfig = {
      apiKey: "AIzaSyDbYTsiuUI7IIQ12zefAzBxWn6KuKOAR1Y",
      authDomain: "fir-led-control-1ddb6.firebaseio.com",
      databaseURL: "https://fir-led-control-1ddb6-default-rtdb.firebaseio.com",
      projectId: "fir-led-control-1ddb6",
      storageBucket: "fir-led-control-1ddb6.appspot.com",
      messagingSenderId: "1057751920144",
      appId: "1:1057751920144:web:68840f2d9fa14d3dba6495",
      measurementId: "G-XJBVSEB9D"
    };
  </script>
```


웹에서 LED 제어하기

```
// Initialize Firebase
const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);

const db = getDatabase();
console.log(db);

var OnButton = document.getElementById("onBtn");
OnButton.addEventListener("click", onBtnClick, false);

var OffButton = document.getElementById("offBtn");
OffButton.addEventListener("click", offBtnClick, false);

function onBtnClick(event) {
  set(ref(db, 'LED_STATUS'), 'ON');
  console.log("LED ON");
}

function offBtnClick(event) {
  set(ref(db, 'LED_STATUS'), 'OFF');
  console.log("LED OFF");
}

const dbRef = ref(db, 'LED_STATUS');
onValue(dbRef, (snapshot) => {
  console.log(snapshot.val());
  OUT_TEXT.innerText = 'LED is ' + snapshot.val();
})
</script>

</body>
</html>
```

웹에서 LED 제어하기

- 실행

ToT Cloud Remote LED Control Wep App

LED is OFF

LED ON

LED OFF

도전하기

ToT Cloud Remote LED Control Wep App

LED is OFF

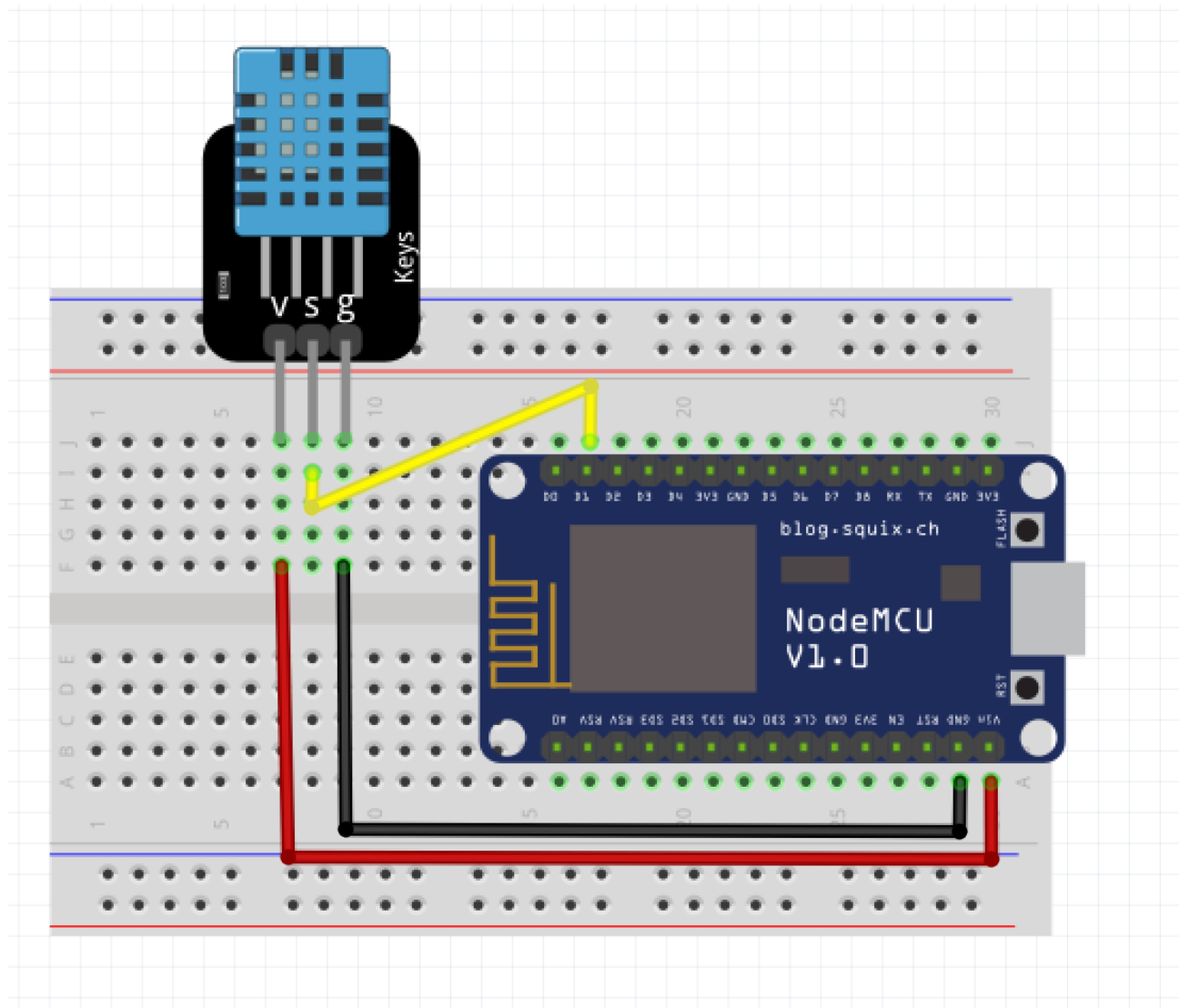


구글 파이어베이스 온습도 정보를 웹에 서비스 하기

1. nodeMCU에서 온습도 데이터 firebase에 전송 코드 작성
2. firebase에 온습도 데이터가 저장되어 있는지 확인
3. javascript 용 firebase sdk 사용 확인 및 복사
4. 웹페이지 코드 작성

구글 파이어베이스에 온습도 데이터 보내기

1) 회로 구성



구글 파이어베이스에 온습도 데이터 보내기

2) 프로그래밍

```
1 #include <ESP8266WiFi.h> //wifi보드 사용을 위해
2 #include <FirebaseESP8266.h> // 파이어베이스 연동을 위해
3 #include <DHT.h>
4
5 #define WIFI_SSID "<ssid>" //와이파이 이름
6 #define WIFI_PASSWORD "<password>" //와이파이 비밀번호
7
8 // 파이어베이스 API 키
9 #define FIREBASE_HOST "arduino-2a743-default-rtdb.firebaseio.com"
10 #define FIREBASE_AUTH "qv402KvQhYD29oVkegF2uY6xEVvn1XpBv1kAF4av"
11
12 // Firebase Data Object to hold data and instances.
13 FirebaseData fbdo;
14
15 // 온습도 센서
16 DHT dht(D1, DHT11);
17
18 void setup() {
19     Serial.begin(115200);
20
21     WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
22     Serial.print("connecting");
23     while (WiFi.status() != WL_CONNECTED) {
24         Serial.print(".");
25         delay(500);
26     }
27     Serial.println();
28     Serial.print("connected: ");
29     Serial.println(WiFi.localIP());
30
31     // 파이어베이스 시작
32     Firebase.reconnectWiFi(true); //Firebase에 재접속 허용
33     Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
34
35     // 온습도 센서 시작
36     Serial.println("DHT11 test!");
37     dht.begin();
38 }
```

구글 파이어베이스에 온습도 데이터 보내기

2) 프로그래밍

```
40 int n = 0; //파이어베이스와 연동이 잘 되는지 확인을 위해 계속 1씩 오르는 변수 하나 설정해둠
41
42 void loop() {
43     //온습도 데이터를 DHT11에서 측정하기
44     float humid = dht.readHumidity();
45     float temp = dht.readTemperature();
46     // 둘 다 측정된 데이터가 없을 경우 에러 출력
47     if (isnan(humid) || isnan(temp)) {
48         Serial.print("Error");
49         return;
50     }
51     n++; //입력되는 데이터의 갯수를 파악하기 위한 카운터
52     Firebase.pushInt(fbdo, "/nodeMCU/num", n);
53     Firebase.pushFloat(fbdo, "/nodeMCU/Humidity", humid);
54     Firebase.pushFloat(fbdo, "/nodeMCU/Temperature", temp);
55
56     delay(5000); //5초에 한 번씩 측정 데이터 전달
57 }
```

구글 파이어베이스에 온습도 데이터 보내기

3) 실행

실시간 데이터베이스

데이터 규칙 백업 사용량 확장 프로그램 신규

결제 사기나 피싱과 같은 악용으로부터 Realtime Database 리소스를 보호하세요. 앱 체크 구성

https://arduino-2a743-default-rtdb.firebaseio.com

https://arduino-2a743-default-rtdb.firebaseio.com/

- LED: "off"
- nodeMCU
 - Humidity
 - NbTVrKMG7HLOmvA0Y-y: 95
 - NbTVsmPGQC19r5a9o4y: 95
 - Temperature
 - NbTVrPG8XMQRiEnIOiY: 28.3
 - NbTVsrGxmqiWvVlr8Jt: 28.3
 - num
 - NbTVrFYf8o8640vwky2: 1
 - NbTVshelBdkIiqW6BIN: 2

firebase realtime database 확인

Firestore-LED-Control - 실시간 | x

console.firebase.google.com/project/fir-led-control-1ddb6/database/fir-led-control-1ddb6-default-rtdb...

Firestore-LED-Control

실시간 데이터베이스

데이터 규칙 백업 사용량 확장 프로그램

https://fir-led-control-1ddb6-default-rtdb.firebaseio.com

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

```
https://fir-led-control-1ddb6-default-rtdb.firebaseio.com/  
├── GREETING: "안녕하세요!!"  
├── LED_STATUS: "ON"  
└── nodeMCU  
    ├── humidity  
    ├── num  
    └── temperature
```

데이터베이스 위치: 미국(us-central1)

참고자료

[firebase realtime database 시작하기 매뉴얼]

<https://firebase.google.com/docs/database/web/start>

[리스트 데이터 이벤트 처리하기]

<https://firebase.google.com/docs/database/web/lists-of-data>

웹페이지 코드 작성

```
1 <html>
2   <head>
3     <meta charset="utf-8">
4     <title>firebase weather station</title>
5   </head>
6   <body>
7     <h1>Weather Station</h1>
8     nodeMCU + Firebase + Web
9     <hr>
10    <h2 id="temp"></h2>
11    <h2 id="humid"></h2>
12
13    <button type="button" onclick="window.location.reload()">데이터 새로 고침</button>
14
15    <script type="module">
16      import { initializeApp } from "https://www.gstatic.com/firebasejs/9.4.1/firebase-app.js";
17      import { getDatabase, ref, onValue, onChildAdded } from "https://www.gstatic.com/firebasejs/9.4.1/firebase-database.js";
18
19      const firebaseConfig = {
20        apiKey: "AIzaSyDbYTsiuUI7IIQ12zefAzBxWn6KuKOAR1Y",
21        authDomain: "fir-led-control-1ddb6.firebaseio.com",
22        databaseURL: "https://fir-led-control-1ddb6-default-rtdb.firebaseio.com",
23        projectId: "fir-led-control-1ddb6",
24        storageBucket: "fir-led-control-1ddb6.appspot.com",
25        messagingSenderId: "1057751920144",
26        appId: "1:1057751920144:web:68840f2d9fa14d3dba6495",
27        measurementId: "G-XJBVSEB9D"
28      };
```

웹페이지 코드 작성

```
30 const app = initializeApp(firebaseConfig);
31 const db = getDatabase(app);
32
33 const dbRefTemp = ref(db, '/nodeMCU/temperature');
34 const dbRefHumi = ref(db, '/nodeMCU/humidity');
35
36 onChildAdded(dbRefTemp, (snapshot)=>{
37   const data = snapshot.val();
38   console.log(data);
39   temp.innerText = "Temperature : " + data + " °C";
40 });
41
42 onChildAdded(dbRefHumi, (snapshot)=>{
43   const data = snapshot.val();
44   humid.innerText = "Humidity : " + data + " %";
45 });
46
47 </script>
48 </body>
49 </html>
```

Weather Station

nodeMCU + Firebase + Web

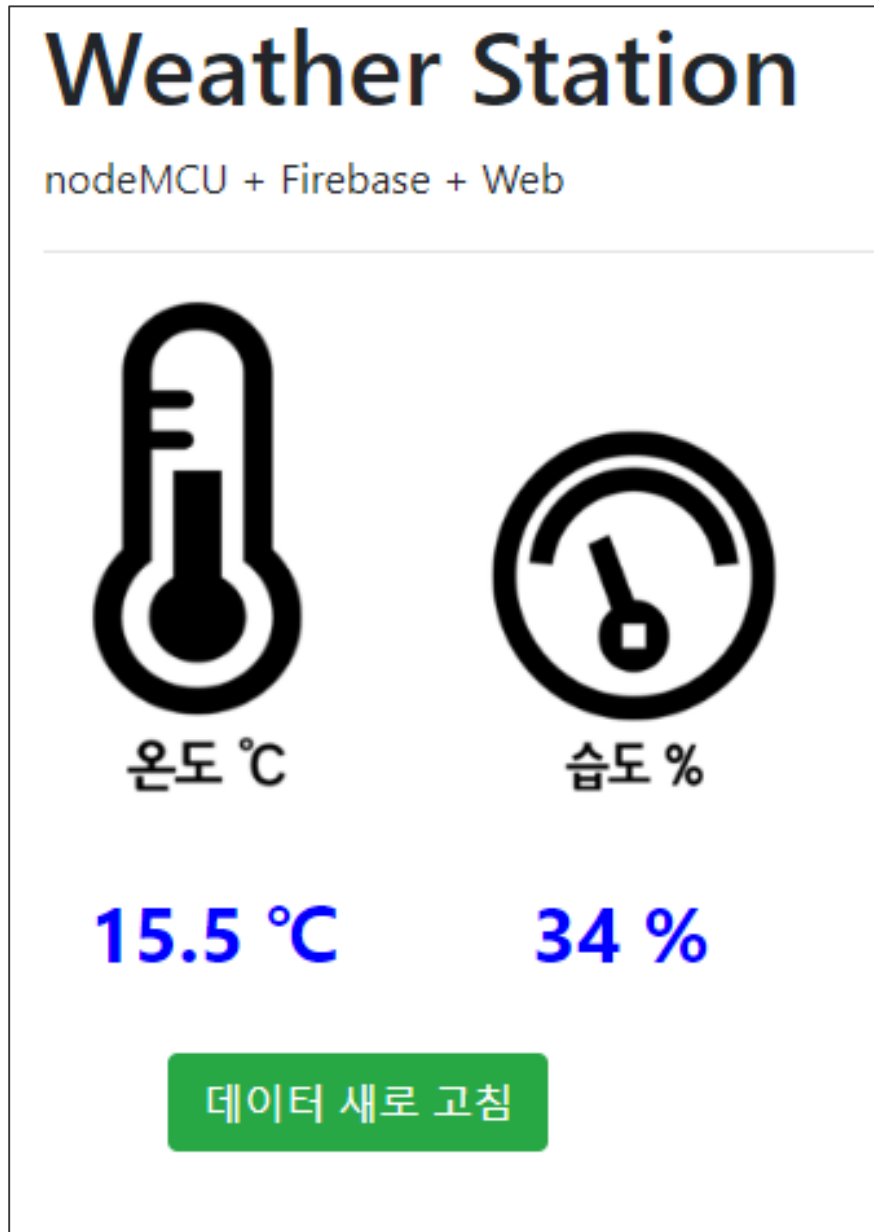
Temperature : 15 °C

Humidity : 33 %

데이터 새로 고침

실행화면

도전 과제 1



도전 과제 2

- 화면 합치기

ToT Cloud Remote LED Control Wep App

LED is OFF



Weather Station

nodeMCU + Firebase + Web



온도 °C

15.5 °C



습도 %

34 %

데이터 새로 고침