

# Firestore 기반 IoT 온습도 모니터링 웹앱 만들기



Temperature



Humidity



# Ardunio Code

```
void loop() {  
  // Read temp & Humidity for DHT22  
  float h = dht.readHumidity();  
  float t = dht.readTemperature();  
  
  // 객체(Object) 단위로 처리  
  StaticJsonBuffer<200> jsonBuffer;  
  JsonObject& root = jsonBuffer.createObject();  
  
  root["temperature"] = t; // root[key] = value  
  root["humidity"] = h;  
  
  // append a new value to /logDHT  
  String name = Firestore.push("logDHT", root);  
  delay(5000);  
}
```

- 전체 소스 코드 : [https://github.com/DIT-IoT-Cloud-2021-2/Source/tree/main/Firebase\\_RTDB\\_DHT11\\_WebApp](https://github.com/DIT-IoT-Cloud-2021-2/Source/tree/main/Firebase_RTDB_DHT11_WebApp)

## Realtime Database

데이터 규칙 백업 사용량

결제 사기나 피싱과 같은 악용으로부터 Realtime Database 리소스를 보호하세요.

<https://dht11-firebase-1be7c.firebaseio.com/logDHT>

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

[dht11-firebase-1be7c](#) > [logDHT](#)

logDHT

```
-MowNVi8a0J1FQBk5YdN  
  humidity: 44  
  temperature: 28.6  
-MowNYCp0QZVh2kWvhfD  
  humidity: 52  
  temperature: 22.4  
-MowN_iy5a1GA9-HB5r_  
  humidity: 51  
  temperature: 22.6  
-MowNcDybTmLc0GtiPp0  
  humidity: 52  
  temperature: 22.8  
+MowNeiwRtPgxAbyE6co  
+MowNhDyETF3-y3qOuXt  
+MowNjpXdZSI74643gMN  
+MowNmKMBjwBxgQvBc2  
+MowNopMsVEWMdlerdhh  
+MowNrKLZu55ALxqliwB  
+MowNtpNYI3esHnWXJ8B
```

- onChildAdded() 메소드는 child 추가 이벤트가 발생하면 데이터베이스의 스냅샷(snapshot)을 가져 온다.
- data.val().humidity/ temperature 메소드로 스냅샷의 값을 뽑아 웹페이지(Javascript)에 데이터를 출력한다.

```
<body>
  <h2> Firebase IoT Weather Monitor using Object Data
  <h2 id="temp"> </h2>
  <h2 id="humi"> </h2>
  <script type="module">
    ➡ insert web app's Firebase configuration
    // Initialize Firebase
    const app = initializeApp(firebaseConfig);
    const analytics = getAnalytics(app);
    // Get a reference to the database service
    const db = getDatabase(app);
    const dbRef = ref(db, 'logDHT/');

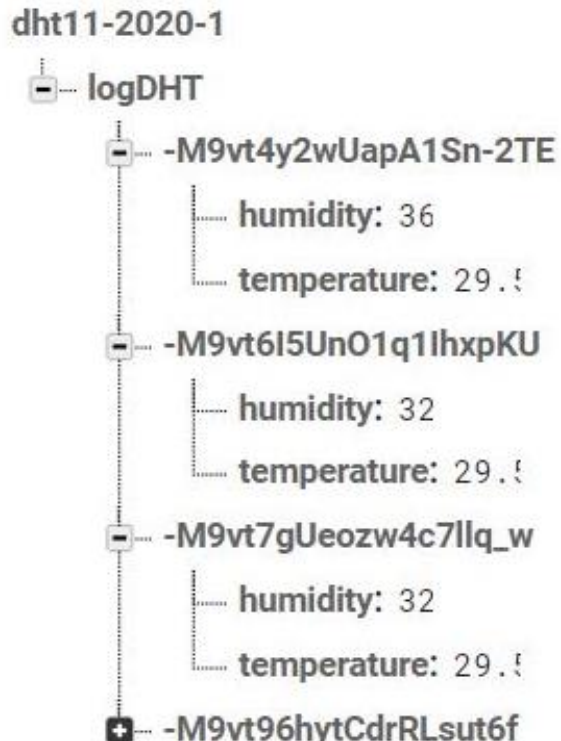
    onChildAdded(dbRef, (data) => {
      var newHumi = data.val().humidity;
      var newTemp = data.val().temperature;
      console.log(newTemp);
      console.log(newHumi)
      humi.innerText = "Humidity : " + newHumi + "%";
      temp.innerText = "Temperature : " + newTemp + "°C";
    });
  </script>
</body>
```

JavaScript  
WebApp

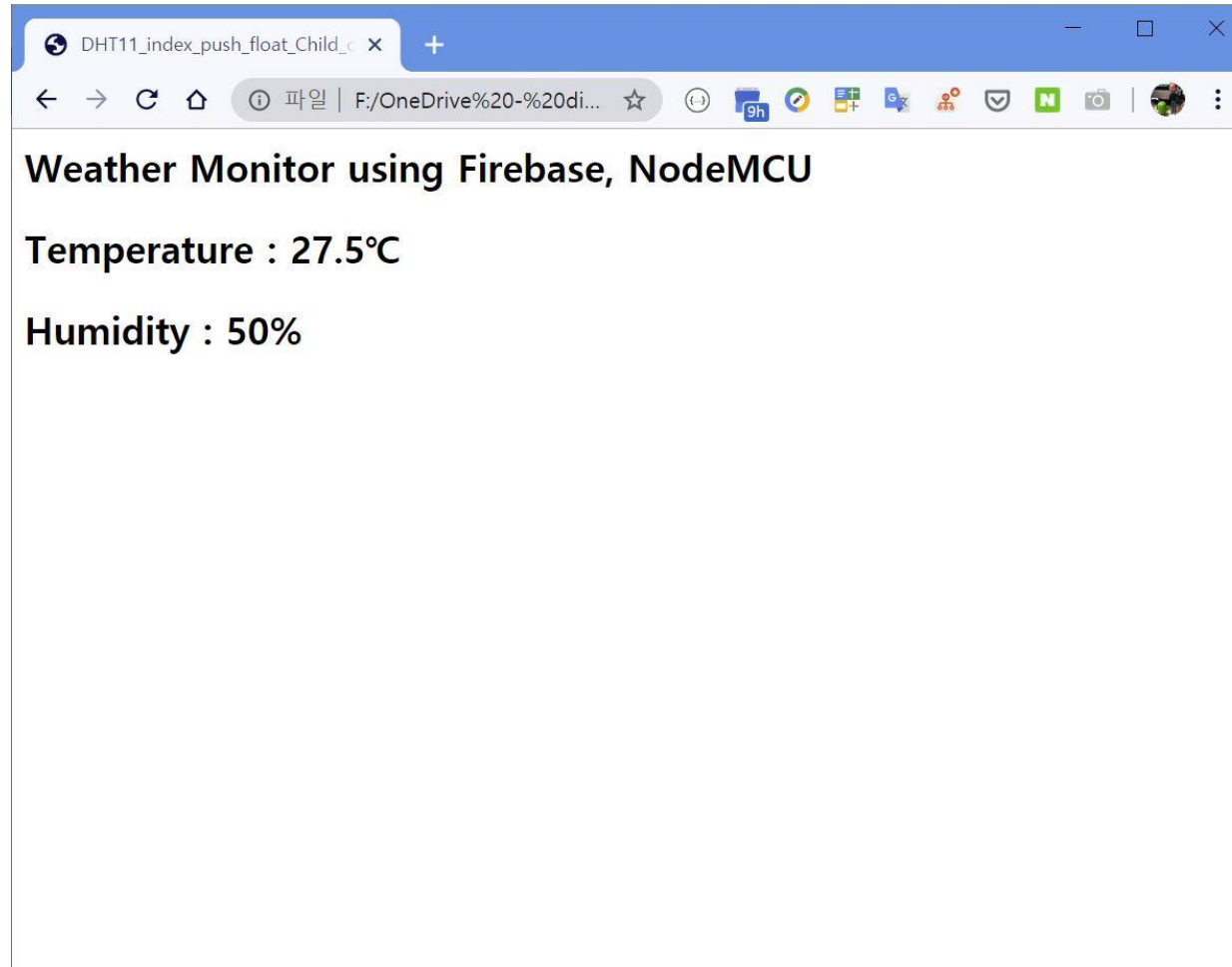
- 전체 소스 코드 : [https://github.com/DIT-IoT-Cloud-2021-2/Source/tree/main/Firebase\\_RTDB\\_DHT11\\_WebApp](https://github.com/DIT-IoT-Cloud-2021-2/Source/tree/main/Firebase_RTDB_DHT11_WebApp)

# Firestore-Event Types

- onDataChange
  - 데이터베이스의 데이터가 변화가 있을 때 이벤트 발생
  - 자식(children)을 포함하는 모든 데이터를 읽어온다.
- onChildAdded
  - 데이터베이스에 새로운 하위 객체(child)가 추가될 때 이벤트 발생
  - 새로 추가된 객체(Object)를 읽어 온다.



# 웹 앱 결과 : temperature, Humidity / pushFloat()



# Firestore 웹에서 데이터 읽기 쓰기 API

- <https://firebase.google.com/docs/database/web/read-and-write?authuser=0>

# 과제

- CSS, 이미지 등을 사용하여 웹앱의 UI를 보기 좋게 만드시오.
- Bootstrap 활용 참고 :

[https://www.w3schools.com/bootstrap/bootstrap\\_ver.asp](https://www.w3schools.com/bootstrap/bootstrap_ver.asp)



### Weather Monitor using Firebase & NodeMCU

Temperature  
27.2 °C

Humidity  
72 %

Temperature

Humidity



```
pushed: /logDHT/-M9kxtY3IJExghg0y72t
Temperature = 27.10
Humidity = 72.00
pushed: /logDHT/-M9kxv4CkygVSkNhtlm1
Temperature = 27.10
Humidity = 72.00
pushing /logs failed:
Temperature = 27.10
```

```
DHT11_Firebase_DB_NodeMCU
1 #include <ESP8266WiFi.h>
2 #include <FirebaseArduino.h>
3
4 #define DHTPIN D4
5 #define DHTTYPE DHT11
6
7 #include <DHT.h>
8 DHT dht(DHTPIN, DHTTYPE);
9
10 // Set these to run example
11 #define FIREBASE_HOST "dht1
12 #define FIREBASE_AUTH "f6cH
13 #define WIFI_SSID "Amadeus"
14 #define WIFI_PASSWORD "deit
15
16 void setup() {
17   Serial.begin(9600);
18   // connect to wifi.
19   WiFi.begin(WIFI_SSID, WIF
20   dht.begin(9600);
21
22   Serial.print("connecting"
23   while (WiFi.status() != W
24     Serial.print(".");
25     delay(500);
```