



Arduino Lesson ()

Display DHT11 data with Arduino WeMos D1 ESP8266 & Blynk app

Name: _____ () Class: _____ Date: _____

Objectives: At the end of this lesson, you would be able to

1. Connect a Arduino **WeMos D1** Wifi UNO ESP8266 to receive readings from DHT11
2. Write sketches to use ESP8266 to display data with Blynk app on a smartphone

Apparatus:

- 01 Arduino **WeMos D1 Wifi UNO ESP8266**
- 01 **DHT11 temperature & humidity sensor module**
- 01 *USB micro B cable (used for typical powerbank)
- 01 mini-breadboard
- 03 male-male jumper wires
- smartphone (with **Blynk** app downloaded)

USB micro B cable



Note:

- WEMOS D1 is a [WIFI development board](#) based on ESP8266 12E, and is built resembling Arduino UNO.
- It turns the very popular ESP8266 wireless (WiFi) module into a [fully fledged development board](#).

Assignment 1: Install the Arduino WeMos on Arduino IDE

1. Go to **File** → **Preferences**

→ go to bottom of dialog box: **Additional Boards Manager URLs** text box

→ copy & paste http://arduino.esp8266.com/stable/package_esp8266com_index.json

→ click OK to download the package

2. Go to **Tools** → **Board & Boards Manager**: search and install **esp8266** by **ESP8266 Community** (e.g. *ESP8266 version 2.6.3*)

3. Connect the Arduino WeMos board to the laptop with USB micro B cable and check for connectivity:

- Select **Tools** → **Board**: scroll down to select "**WeMos D1 R1**"
- and select ***Port**: "COM#"

*Note:

- *Must use proper **USB micro B cable**. Those typically used for powerbank **often don't have data wires** (data lines) (**they are charge only**) such that the **Port** may not be detected.*
See <https://superuser.com/questions/1260407/are-all-usb-3-micro-b-cables-functionally-the-same>

Reference:

- <https://www.instructables.com/id/Send-Temperature-Humidity-to-Blynk-App-Wemos-D1-Mi/>
- <https://www.arduino.cc/en/Tutorial/ConnectWithWPA>
- <http://help.blynk.cc/en/articles/512105-how-to-install-blynk-library-for-arduino-ide>
- <http://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-an-arduino/> (using digital pin D7)
- <https://randomnerdtutorials.com/complete-guide-for-dht11-dht22-humidity-and-temperature-sensor-with-arduino/>

Assignment 2: Set up DHT11 circuit with Arduino WeMos

4. Insert the DHT11 sensor module onto the breadboard. Set up the DHT11 circuit using the diagram shown below.

With the sensor facing you, pins 1 to 3, from left to right:

- | | | |
|--------------|---|-----------------------------|
| DHT11 | → | Arduino WeMos board |
| Pin 1 (+) | → | 5 V |
| Pin 2 (out) | → | digital pin 4 (D4) |
| Pin 3 (-) | → | GND (ground) |

[Pin number may vary according to DHT11 model.]

5. Connect the Arduino WeMos board to the laptop and check for connectivity:

Select **Tools** → **Board:** “Arduino/Genuino Uno” & **Port:** “COM#”

6. **Install two libraries** into Arduino IDE (Integrated Development Environment):

Sketch → *Include Library* → *Manage Libraries*

- Type “**DHT**” in the search box and install the **DHT sensor** library from **Adafruit**.
- Type “**Adafruit Unified Sensors**” in the search box and install the **DHT sensor** library from **Adafruit** (or this may appear after installing DHT as “other dependencies to install”).

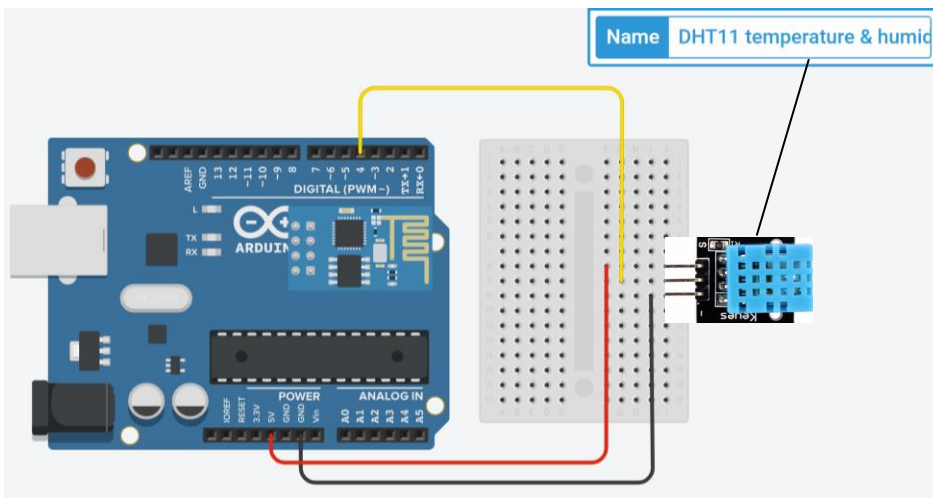
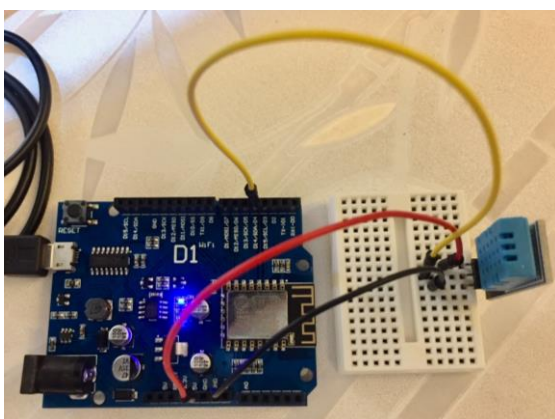


Diagram drawn in tinkercad.com (circuits) (DHT11 module added separately)

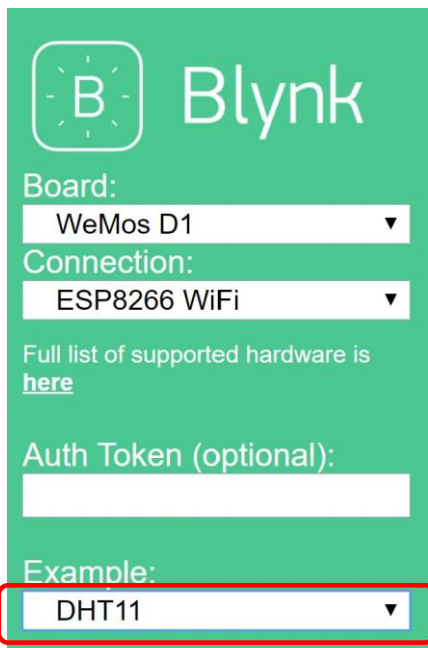


Arduino WeMos D1 board with DHT11



Assignment 3: Install Blynk library and sketch on Arduino IDE

7. Go to **Sketch** → **Include Library** → scroll to the top **Manage Libraries ...**
8. In **Library manager**, search and install latest version of **Blynk**
 - **Blynk** by Volodymyr Shymanskyy version 0.6.1 installed (1 Jan 2020)
9. Go to **Sketch** → **Include Library** menu → scroll down to check **Blynk** is present.
10. Go to **File** > **Examples** > **Blynk** > **More** > open the sketch DHT11.
 - **OR** Go to website <http://help.blynk.cc/en/articles/512062-how-to-find-code-for-my-hardware>
→ **Open Blynk Examples Builder** to find sketch for DHT11 to copy into Arduino IDE.
11. Once **Auth Token** from Blynk app (**Assignment 4**) is available, update the sketch and upload into Arduino WeMos microcontroller.
12. Activate the **serial monitor** to show temperature and humidity readings via Blynk.



Step 10

Assignment 3: Sample sketch

```
/*  
Download latest Blynk library here: .....  
Downloads, docs, tutorials: http://www.blynk.cc  
Sketch generator: http://examples.blynk.cc  
Blynk community: http://community.blynk.cc  
*****  
This example shows how value can be pushed from Arduino to the Blynk App.  
WARNING :  
For this example you'll need Adafruit DHT sensor libraries:  
https://github.com/adafruit/Adafruit\_Sensor  
https://github.com/adafruit/DHT-sensor-library  
  
App project setup:  
Value Display widget attached to V5  
Value Display widget attached to V6
```

```

*****/

/* Comment this out to disable prints and save space */
#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <DHT.h>

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "YourAuthToken"; // replace YourAuthToken with Auth Token

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "YourNetworkName"; // replace YourNetworkName with Network SSID
char pass[] = "YourPassword"; // replace YourPassword with Network key

#define DHTPIN D4 // connect to digital pin 4

#define DHTTYPE DHT11 // DHT 11

DHT dht(DHTPIN, DHTTYPE);
BlynkTimer timer;

// This function sends Arduino's up time every second to Virtual Pin (5).
// In the app, Widget's reading frequency should be set to PUSH. This means
// that you define how often to send data to Blynk App.

void sendSensor()
{
  float h = dht.readHumidity();
  float t = dht.readTemperature(); // or dht.readTemperature(true) for Fahrenheit

  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }
  // You can send any value at any time.
  // Please don't send more that 10 values per second.
  Blynk.virtualWrite(V5, h);
  Blynk.virtualWrite(V6, t);
}

void setup()
{
  // Debug console
  Serial.begin(9600);

  Blynk.begin(auth, ssid, pass);

  dht.begin();

  // Setup a function to be called every second
  timer.setInterval(1000L, sendSensor);
}

void loop()
{
  Blynk.run();
  timer.run();
}

```


Note:

- E.g. **Auth Token:** 275UBpwLoy3I5U1FdhvH0ZR6bsG4aVRx
 Network SSID: SINGTEL-1234
 Network Key: abcdefghij
- **Auth Token:** is obtained from **Blynk email** or **Project Settings** in **Blynk** app on smartphone
- **Network SSID** (Service Set ID): **network name** that displays when you refresh your wireless network list.

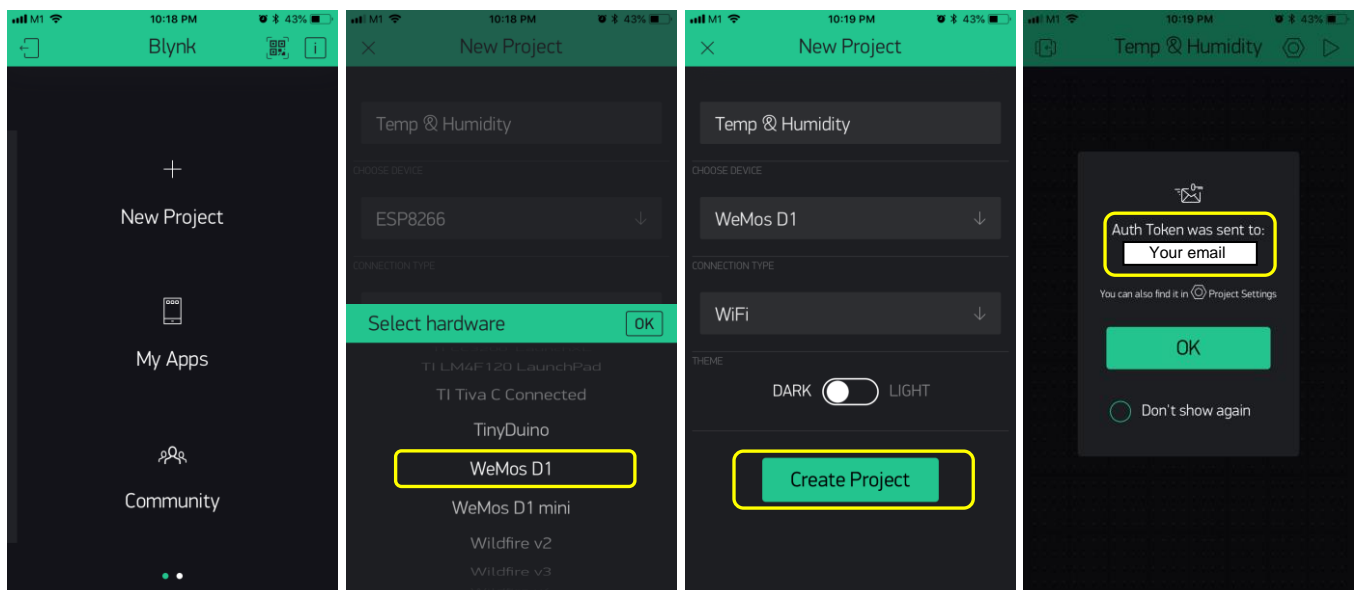
Assignment 4: Set up Blynk app on smartphone & temperature & humidity gauges

13. Download "**Blynk**" app from IOS Appstore.
 14. Sign up using your email (e.g. gmail – can reset password).
 15. Swipe sideways to "**New Project**". Enter your project name (if needed).
- **Choose Device:** Scroll to select hardware "WeMos D1". Click OK.
 - **Connection Type:** Scroll to select connection type "WIFI".
 - Click **Create Project**. Click OK.

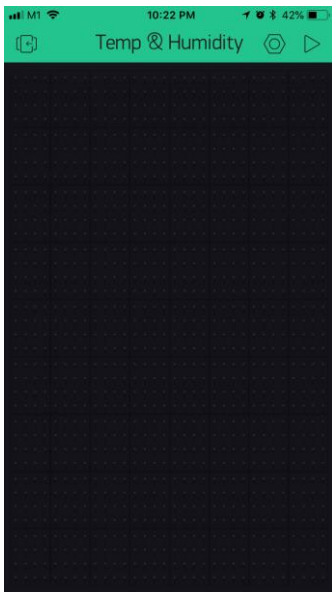
Note:

- Check **email** for **Auth Token**. (Also found in top right "nut icon"  **Project Settings**).
 - Insert **Auth Token** into the **Blynk sketch** in Assignment 3.
16. Tap on the empty canvas to open "**Widget Box**".
 17. Scroll up/down, touch and hold "Gauge" to add gauge. Add another gauge.
 18. Tap on first rectangle representing "Gauge".
 19. Key in "Temperature". Under **INPUT**, tap on **PIN** → select **Virtual** → select **V6** to choose pin connection for temperature. Select "**1 sec**". Click OK.
 20. Under **Label**, indicate **/pin/°C**. Click OK (top right corner).
 21. Repeat for 2nd rectangle representing "Gauge": Key in "Humidity". Select **Virtual** → select **V5**. Click OK. Under **Label**, indicate **/pin/%**.
 22. Under the project where you can see both gauges, click the **play button** or "**start**" **triangle** (top right corner) to activate both gauges. If wifi is connected, you would see the readings on Blynk app.

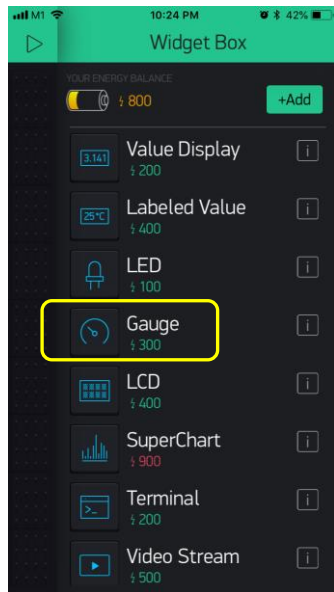
Screen shots of Blynk on smartphone



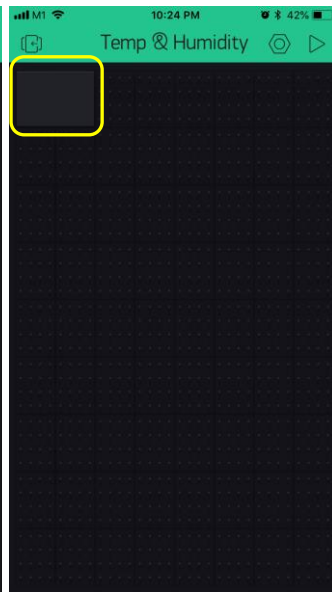
Step 15



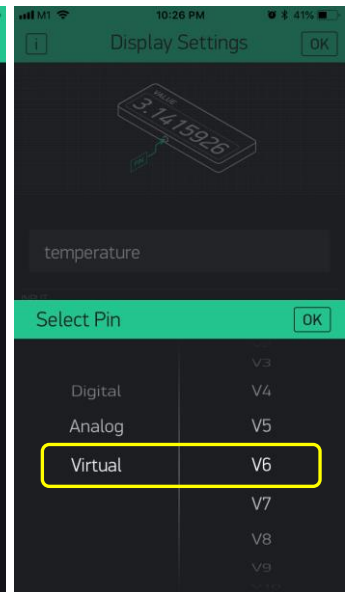
Step 16



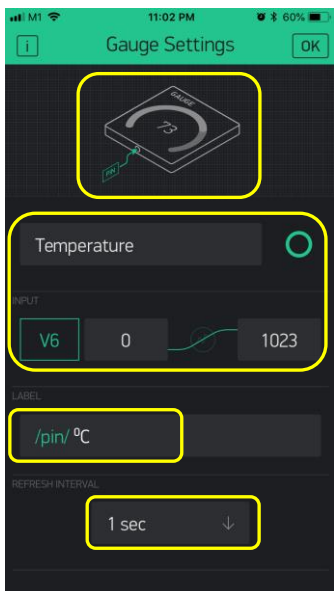
Step 17



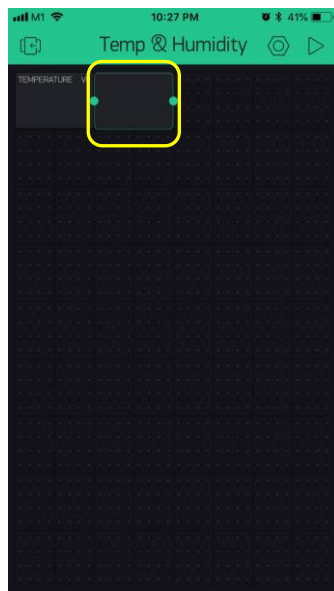
Step 18



Step 19



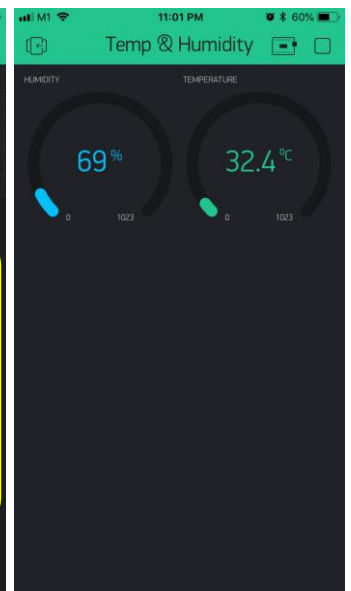
Step 20



Step 21

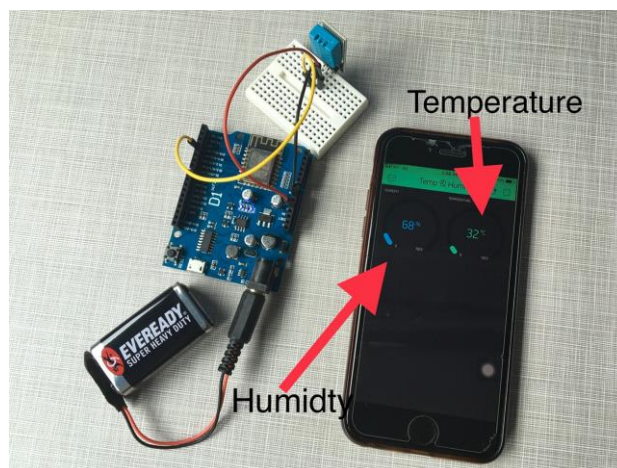


Step 21



Step 22

Reference: <https://docs.blynk.cc/>



Final setup with display of data on smartphone

Test if DHT11 sensor module (3 pins) is in working order

Connect DHT11 to an **Arduino Uno** microcontroller

With the sensor facing you, pins 1 to 3, from left to right:

DHT11 →	Arduino WeMos microcontroller
Pin 1 (+)	→ 5 V
Pin 2 (out)	→ digital pin 2 (D2)
Pin 3 (-)→	GND (ground)

Use the usual libraries and uploading.

Activate the **serial monitor** to show temperature and humidity readings.

Sketch for testing DHT11

```
#include "DHT.h"
#define DHTPIN 2 // what pin we're connected to
#define DHTTYPE DHT11 // DHT 11

// Initialize DHT sensor for normal 16mhz Arduino
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println("DHT11 test!");
  dht.begin();
}

void loop() {

  // Wait a few seconds between measurements.
  delay(2000);
  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius
  float t = dht.readTemperature();

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }
  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.println(" %\t");
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.println(" *C ");
}
```

Trouble-shooting

1. WeMos board Pin 2 in original sketch from Blynk website

- Pin 2 is **not** D2 - <https://community.blynk.cc/t/solved-wemos-d1-mini-dht11-shield-example-failed-to-read-from-dht-sensor/11256>
- Pin 2 & types of WeMos boards (D1, D1R2) - <https://community.blynk.cc/t/help-with-pin-settings/8768/27>
- The 2 that is referred to in the sketch is GPIO 2 for the ESP and **NOT** D2 on your board.
- This is the pinout for the R2 and you should try to find a pinout for the R1. Based on the R2 **GPIO 2** is **D4** on the WeMos. So the data pin of the DHT11 would be connected to **D4**.



- <https://community.blynk.cc/t/wemos-d1-cannot-read-dht-11-help/19168/12>

A Comparison of Wemos D1 "R2" vs. "R1" Pinouts

WeMos D1 R2 WiFi Board Pinouts:

R1 Pin	Pin	Function	ESP-8266 Pin
D0	TX	TXD	TXD
D1	RX	RXD	RXD
D2	D0	IO	GPIO16
D3 (D15)	D1	IO, SCL	GPIO5
D4 (D14)	D2	IO, SDA	GPIO4
D5 (D13)	D3	IO, 10k Pull-up	GPIO0
D6 (D12)	D4	IO, 10k Pull-up, BUILTIN_LED	GPIO2
D7 (D11)	D5	IO, SCK	GPIO14
D8	D6	IO, MISO	GPIO12
D9	D7	IO, MOSI	GPIO13
D10	D8	IO, 10k Pull-down, SS	GPIO15
A0		Analog input, max 3.3V input	A0
G		Ground	GND
5V		5V	-
3V3		3.3V	3.3V
RST		Reset	RST

All of the IO pins have interrupt/pwm/I2C/one-wire support except D0

WeMos D1 "R1" WiFi Board Pinouts:

R1 Pin	Function	ESP-8266 Pin
D0	RX	GPIO3
D1	TX	GPIO1
D2	IO	GPIO16
D3 (D15)	IO, SCL	GPIO5
D4 (D14)	IO, SDA	GPIO4
D5 (D13)	IO, SCK	GPIO14
D6 (D12)	IO, MISO	GPIO12
D7 (D11)	IO, MOSI	GPIO13
D8	IO, Pull-up	GPIO0
D9	IO, Pull-up, BUILTIN_LED	GPIO2
D10	IO, Pull-down, SS	GPIO15
A0	Analog Input	A0
G	Ground	GND
5V	5V	-
3V3	3.3V	3.3V
RST	Reset	RST

*All IO have interrupt/pwm/I2C/one-wire supported(except D2)