Challenge 3

Making it Battery Powered

Step 1: Let the D1 mini get some sleep

Currently your code is working nicely, but it is not very efficient! Your D1 mini is constantly doing something – even the 'delay' function is using a bunch of processing power. This means that your device would not currently last very long on battery power. We can fix this by restructuring our code and using the D1 mini's "deep sleep" functionality, which powers it right down when it's not actively reading data.

Firstly, move **all** of the code from inside your 'loop' to the 'setup' block. That's right – move absolutely everything! Just place it underneath the original code in the 'setup' block.

Why do we do this? Well, the 'loop' block runs forever without letting the D1 mini sleep. The 'setup' block runs once, every time the D1 mini is powered on or wakes up from a sleep. So we can use the 'setup' block to read data, then let the D1 mini sleep for a while.

Once everything is in the 'setup' block, add a short delay to the start of the 'setup' block. This will give the D1 mini time to gather its thoughts before trying to do things when it wakes up:

```
28 void setup() {
29   // Brief delay
30   delay(2000);
```

Now, just before the 'setup' block you will want to create a variable to specify how long you want your D1 mini to nap for. For now, set it to 10 seconds. This will make it easier to test for now, and you can increase the time later (soil doesn't change too fast, so you may want to only measure once every hour!).

```
25 // Set up the sleep time in seconds
26 const int sleepTimeS = 10;
```

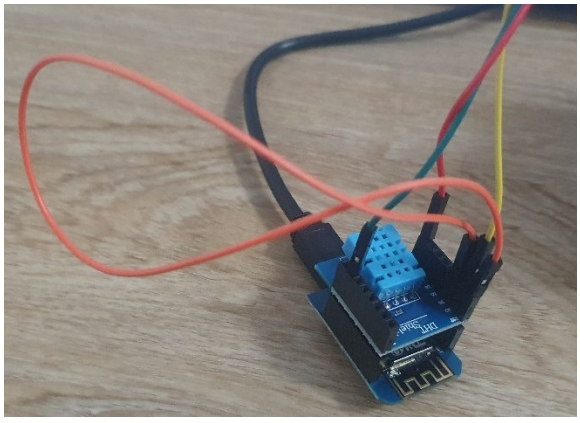
At the end of your 'setup' block, after you've read all of the data and sent it to ThingSpeak, you can let your D1 mini go to sleep. Tuck it into bed with the following code:

```
75   Serial.println("ESP8266 in sleep mode");
76   ESP.deepSleep(sleepTimeS * 1000000);
```

The Serial.println message is just for you to see that it has fallen asleep. The ESP.deepSleep line lets it know to sleep for 10 x 1000000 microseconds (in other words, 10 seconds).

Now you're ready to upload your code!

After you have uploaded your code, you will need to do is use a male-to-male jumper wire (the ones with pins on both ends) to connect the **RST** and **D0** pins on your board directly together. This connection is what the D1 mini will use to wake itself up from deep sleep. *You will need to remove this jumper whenever you upload new code, then plug it back in to enable the sleep/wake cycle to work.*



At this stage, check that you are receiving new data on ThingSpeak roughly every 10 seconds.

Step 2: Adding the batteries

You're now ready to connect the batteries to your D1 mini!

Note: it is important that you disconnect the D1 mini from your computer when it is connected to battery. This means that you should ***remove any batteries before connecting your D1 mini to your computer*** for future updates that you make.

Disconnect your D1 mini now and ask one of your instructors to assist you with connecting a battery holder to the D1 mini. You will need to solder the red wire from the battery holder to the "3V3" pin of the D1 mini, and the black wire from the battery holder to the "G" (or "GND") pin of your D1 mini.

Add some batteries and check to see if it's streaming to ThingSpeak!