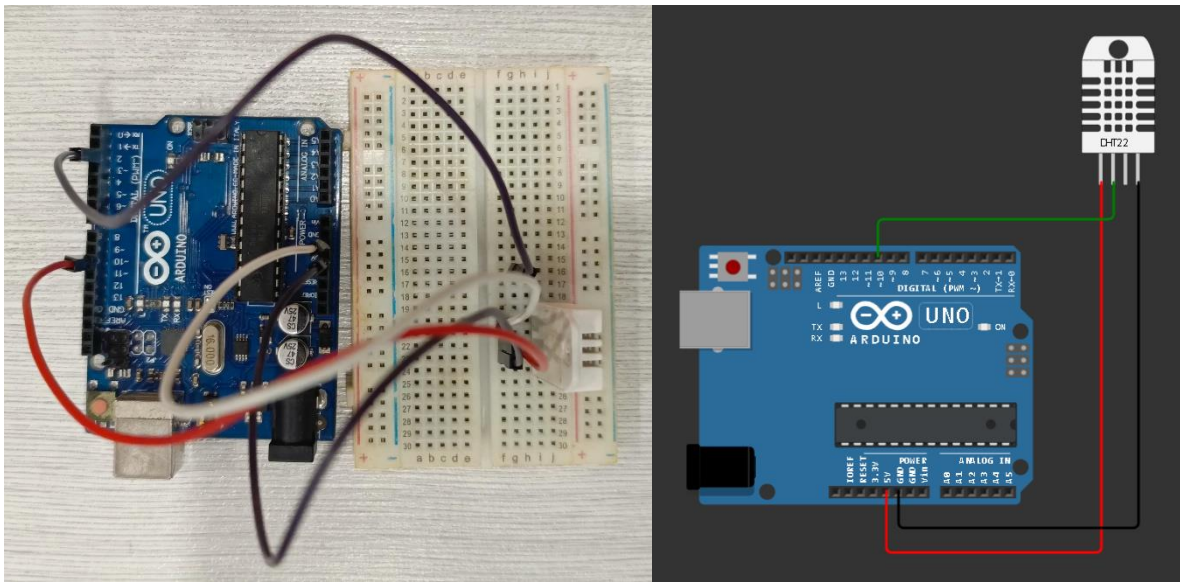
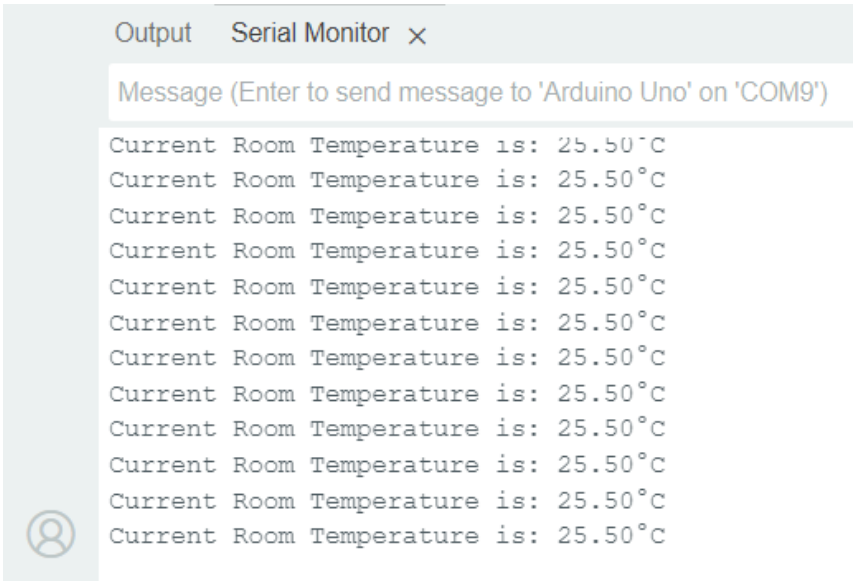
LAB TASKS

Q1: Develop the program to read the value of temperature in °C from DHT22 Temperature Sensor and display in on Serial Monitor. Capture it using snipping tool and add this image in your lab report.

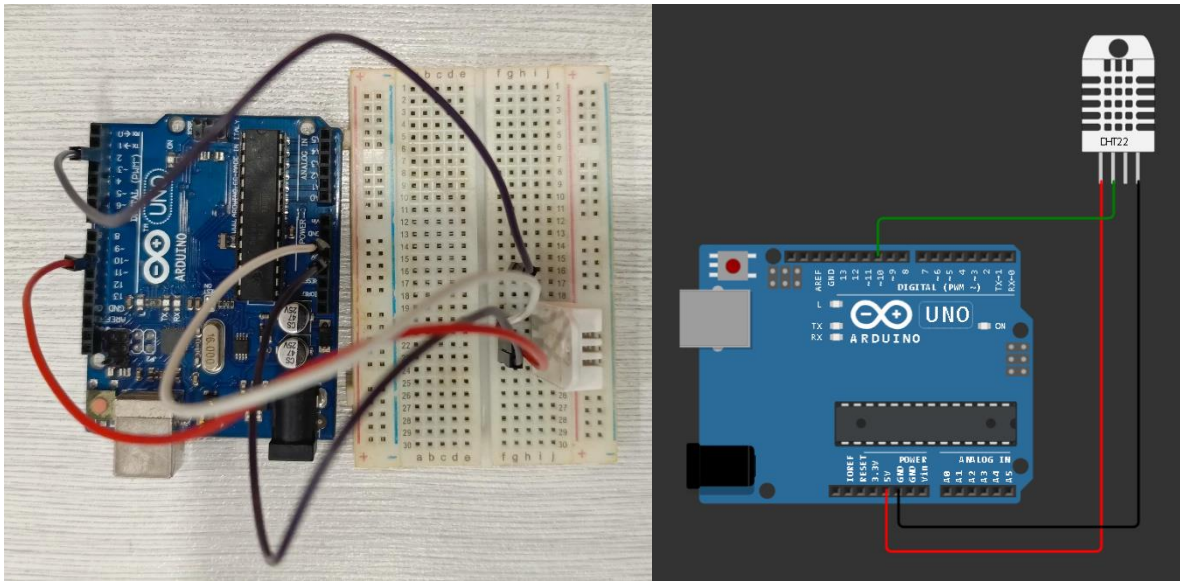
```
#include <DHT.h> //Including library to be used in the code

#define DHTPIN 10 // // Defining digital pin port for sensor
#define DHTTYPE DHT22 // DHT22 (AM2302)

DHT dht(DHTPIN, DHTTYPE); // Creating an object called dht while defining arguements (pin and type) and storing this into DHT

void setup() {
  Serial.begin(9600); // Calling serial monitor for data display
  dht.begin(); // Calling library for temperature input and calculations
}

void loop() {
  delay(2000); // Adding a time delay
  float temp = dht.readTemperature(); // Defining variable to store tempertaure value from library function in °C
  Serial.print("Current Room Temperature is: "); // Printing data to serial monitor
  Serial.print(temp); // Printing value of temperature to serial monitor
  Serial.println("°C"); // Printing temperature unit to serial monitor
}
```

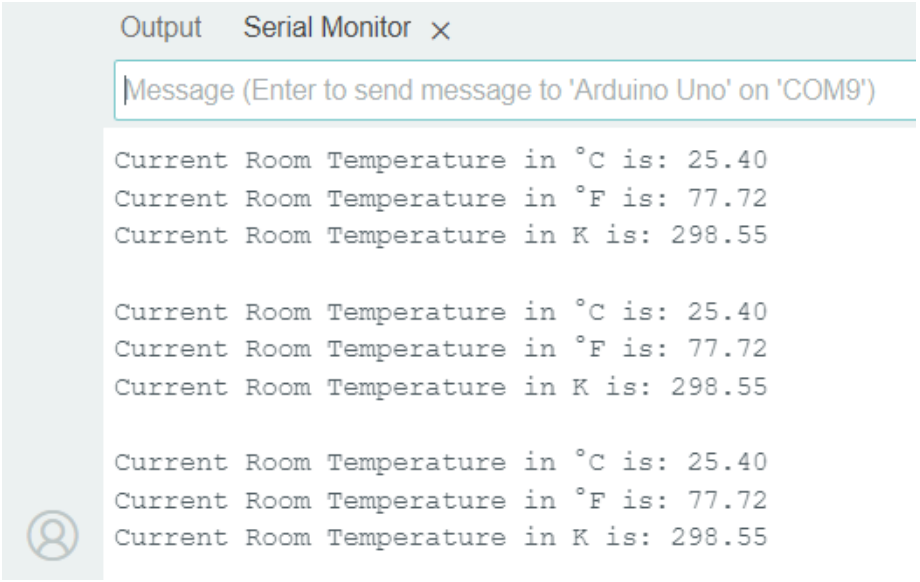




Q2: Develop the program to read the value of temperature in °F and K from DHT22 Temperature Sensor and display in on Serial Monitor. Capture it using snipping tool and add this image in your lab report.

```
#include <DHT.h> //Including library to be used in the code

#define DHTPIN 10 // // Defining digital pin port for sensor
#define DHTTYPE DHT22 // DHT22 (AM2302)

DHT dht(DHTPIN, DHTTYPE); // Creating an object called dht while defining arguements (pin and type) and storing this into DHT

void setup() {
  Serial.begin(9600); // Calling serial monitor for data display
  dht.begin(); // Calling library for temperature input and calculations
}

void loop() {
  delay(3000); // Adding a time delay
  float tempC = dht.readTemperature(); // Defining variable to store tempertaure value from library function in °C
  float tempF = ((tempC*1.8)+32); // Defining variable to store tempertaure value from formulae in °F
  float tempK = tempC + 273.15; // Defining variable to store tempertaure value from formulae in K
  Serial.println(""); // Adding this to add a new line for better visualization
  Serial.print("Current Room Temperature in °C is: "); // Printing data to serial monitor
  Serial.println(tempC); // Printing value of tempC to serial monitor
  Serial.print("Current Room Temperature in °F is: "); // Printing data to serial monitor
  Serial.println(tempF); // Printing value of tempF to serial monitor
  Serial.print("Current Room Temperature in K is: "); // Printing data to serial monitor
  Serial.println(tempK); // Printing value of tempK to serial monitor
}
```

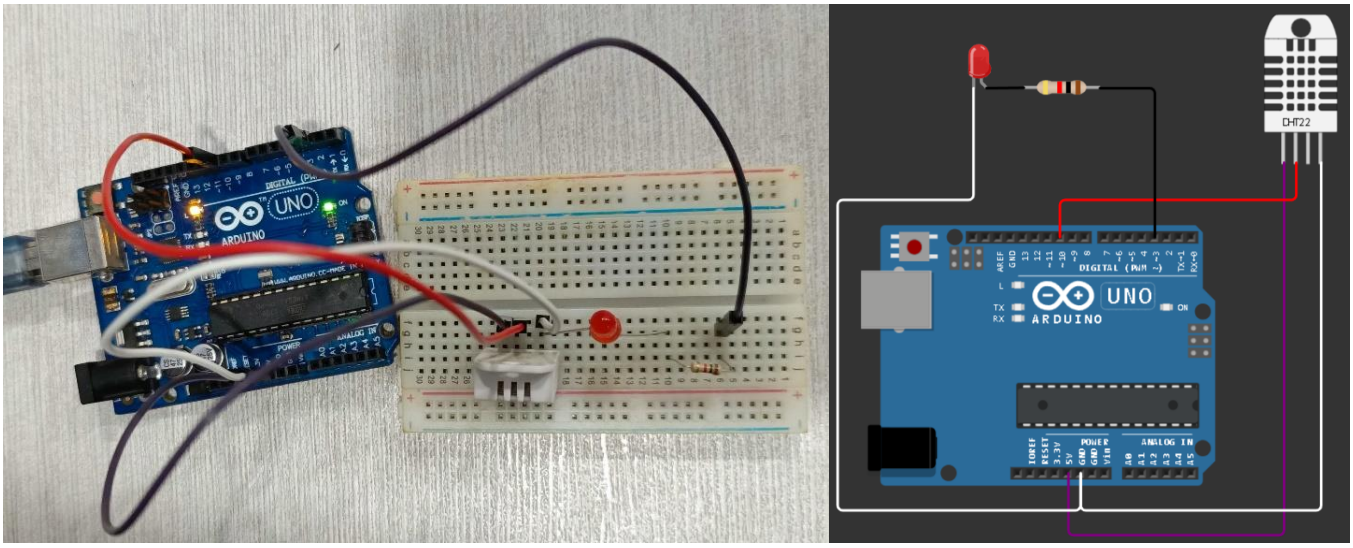


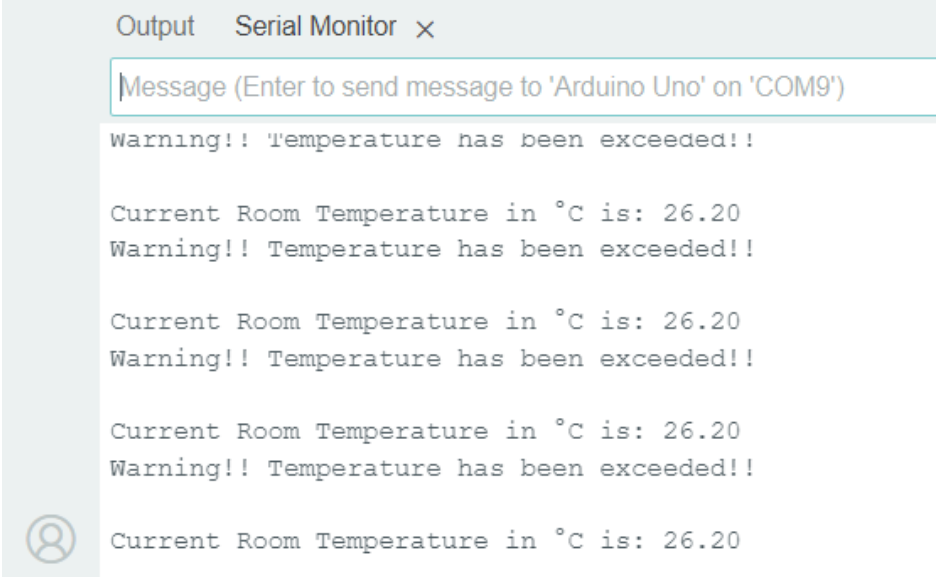


Q3: Design a temperature alarm system which alerts you on a particular temperature. The purpose is to simplify a real-life problem of fire protection which gives you warning in the form of siren or light. Indicate the alarm with help of flashing LED as well as display on serial monitor. (Consider the warning temp = 15°C).

```
#include <DHT.h> //Including library to be used in the code

#define DHTPIN 10 // Defining digital pin port for sensor
#define DHTTYPE DHT22 // DHT22 (AM2302)
#define LEDPIN 2 // Defining digital pin port for LED

DHT dht(DHTPIN, DHTTYPE); // Creating an object called dht while defining arguements (pin and type) and storing this into DHT
float tempC; // Defining variable to start temperature

void setup() {
  Serial.begin(9600); // Calling serial monitor for data display
  dht.begin(); // Calling library for temperature input and calculations
}
```

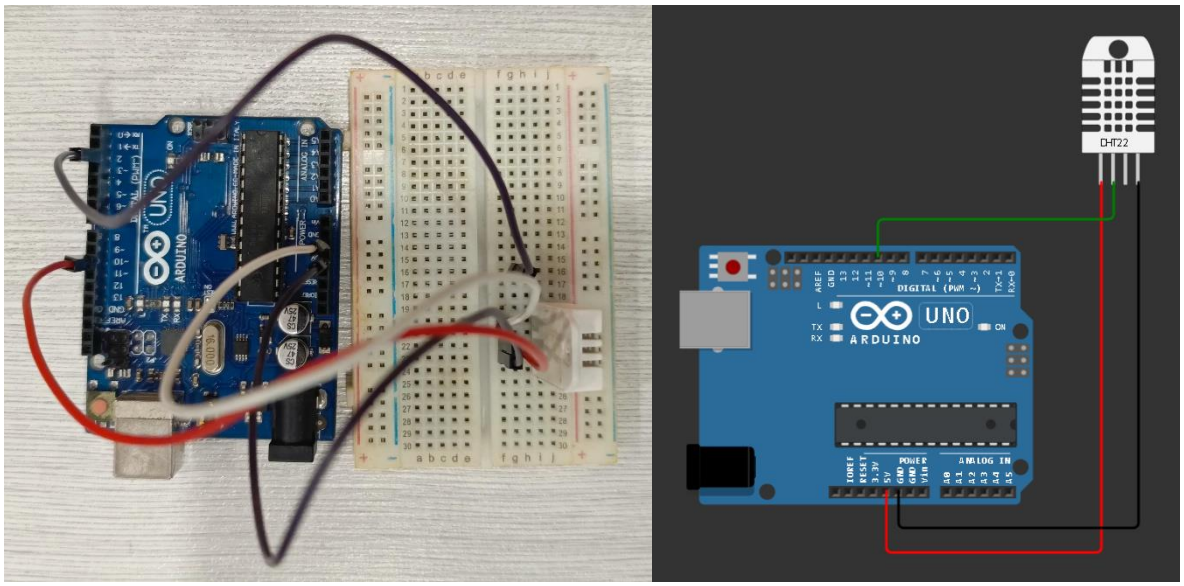
```
void loop() {
  tempC = dht.readTemperature(); // Storing tempertaure value from library function in °C

  if (tempC > 15) { // Defining if condition to check temerature limit
  Serial.println("Warning!! Temperature has been exceeded!!"); // Printing warning statement to serial monitor when if condition verifies
  Serial.println(""); // Adding this to add a new line for better visualization
  Serial.print("Current Room Temperature in °C is: "); // Printing data to serial monitor
  Serial.println(tempC); // Printing value of tempC to serial monitor

  digitalWrite(LEDPIN, HIGH); // Assignning LED with 1 (High)
  delay(100); // Adding a time delay
  digitalWrite(LEDPIN, LOW); // Assignning LED with  0 (Low)
  delay(100); // Adding a time delay
  }
  else { // Condition is true when if does not verify
    Serial.println("Temperature is normal!"); // Printing statement to serial monitor when else condition verifies
  }
}
```

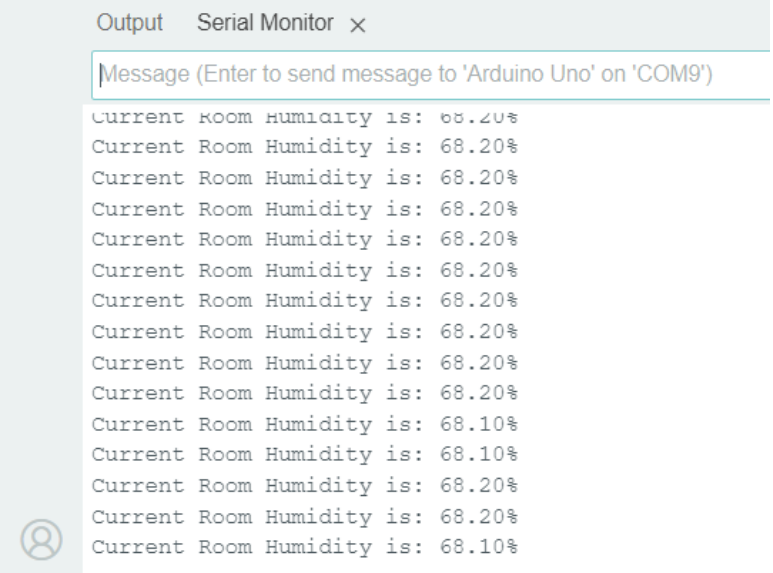




Q4: Develop the program to read the value of relative humidity in percentage from DHT22 Humidity Sensor and display in on Serial Monitor.

```
#include <DHT.h> //Including library to be used in the code

#define DHTPIN 10 // // Defining digital pin port for sensor
#define DHTTYPE DHT22 // DHT22 (AM2302)

DHT dht(DHTPIN, DHTTYPE); // Creating an object called dht while defining arguements (pin and type) and storing this into DHT

void setup() {
  Serial.begin(9600); // Calling serial monitor for data display
  dht.begin(); // Calling library for temperature input and calculations
}

void loop() {
  delay(1000); // Adding a time delay
  float humidity = dht.readHumidity(); // Defining variable to store humidity value from library function
  Serial.print("Current Room Humidity is: "); // Printing data to serial monitor
  Serial.print(humidity); // Printing value of humidity to serial monitor
  Serial.println("%"); // Printing percentage unit to serial monitor
}
```

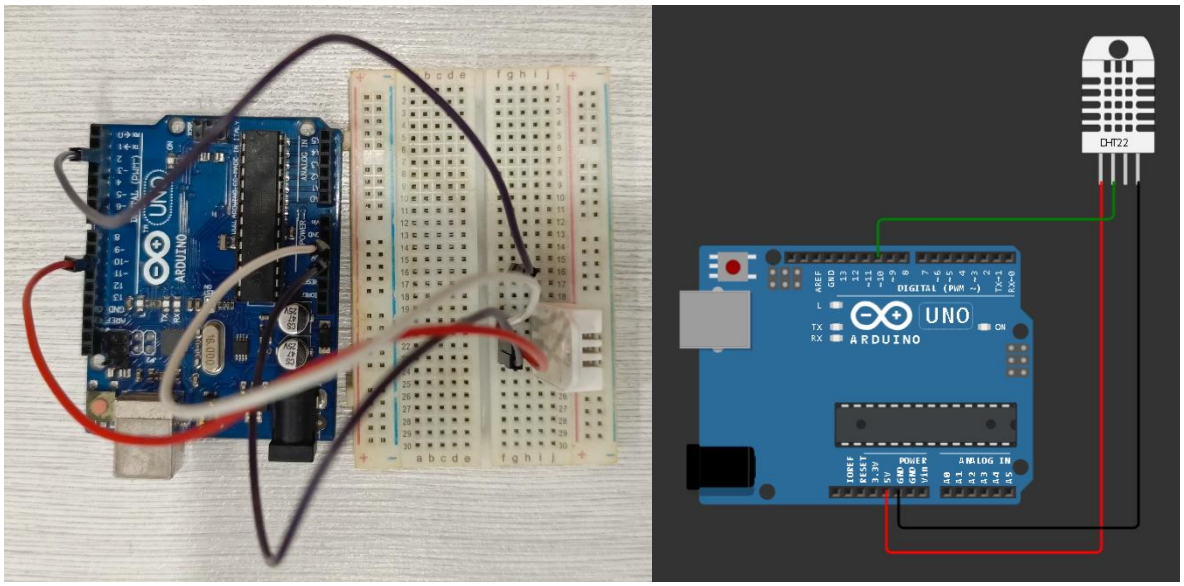


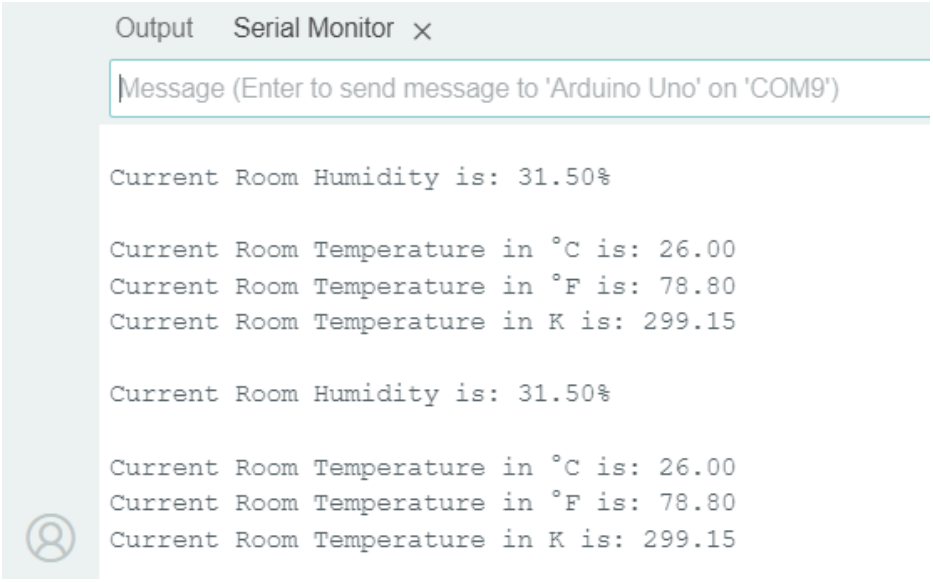


Q5: Write a program to read the value of Humidity in percentage along with Temperature in °C, °F and K from DHT22 Temperature & Humidity Sensor and display in on Serial Monitor. Capture it using snipping tool and add this image in your lab report.

```
#include <DHT.h> //Including library to be used in the code

#define DHTPIN 10 // // Defining digital pin port for sensor
#define DHTTYPE DHT22 // DHT22 (AM2302)

DHT dht(DHTPIN, DHTTYPE); // Creating an object called dht while defining arguements (pin and type) and storing this into DHT

void setup() {
  Serial.begin(9600); // Calling serial monitor for data display
  dht.begin(); // Calling library for temperature input and calculations
}

void loop() {
  delay(1000); // Adding a time delay

  float humidity = dht.readHumidity(); // Defining variable to store humidity value from library function

  Serial.println(""); // Adding this to add a new line for better visualization
  Serial.print("Current Room Humidity is: "); // Printing data to serial monitor
  Serial.print(humidity); // Printing value of humidity to serial monitor
  Serial.println("%"); // Printing percentage unit to serial monitor

  float tempC = dht.readTemperature(); // Defining variable to store tempertaure value from library function in °C
  float tempF = ((tempC*1.8)+32); // Defining variable to store tempertaure value from formulae in °F
  float tempK = tempC + 273.15; // Defining variable to store tempertaure value from formulae in K
```

```
  Serial.println(""); // Adding this to add a new line for better visualization
  Serial.print("Current Room Temperature in °C is: "); // Printing data to serial monitor
  Serial.println(tempC); // Printing value of tempC to serial monitor
  Serial.print("Current Room Temperature in °F is: "); // Printing data to serial monitor
  Serial.println(tempF); // Printing value of tempF to serial monitor
  Serial.print("Current Room Temperature in K is: "); // Printing data to serial monitor
  Serial.println(tempK); // Printing value of tempK to serial monitor
}
```

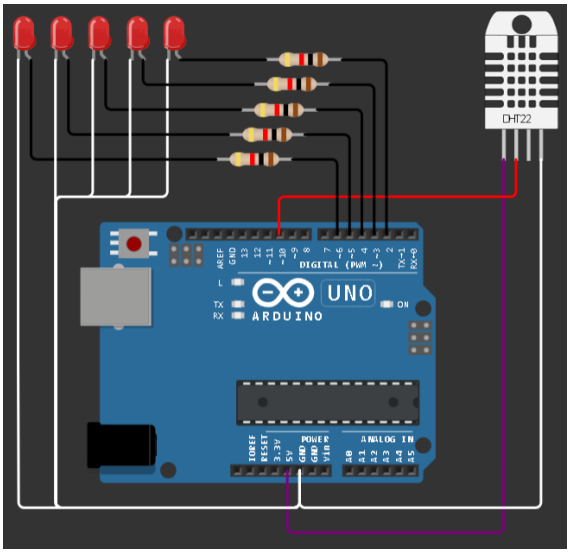




Q6: Design a humidity level meter using DHT22 sensor which indicates you the range of humidity (0-100%) with the help of 5 LEDs. (Code is required with comment on each line along with the circuit diagram. Picture of the circuit and output on the serial monitor are not applicable in this task.)

```
#include <DHT.h> //Including library to be used in the code

#define DHTPIN 10 // // Defining global pin port for sensor
#define DHTTYPE DHT22 // DHT22 (AM2302)
#define LEDPIN1 2 // Defining global pin port for LED
#define LEDPIN2 3 // Defining global pin port for LED
#define LEDPIN3 4 // Defining global pin port for LED
```

```
#define LEDPIN4 5 // Defining global pin port for LED
#define LEDPIN5 6 // Defining global pin port for LED

DHT dht(DHTPIN, DHTTYPE); // Creating an object called dht while defining arguements (pin and type) and storing this into DHT

void setup() {
  Serial.begin(9600); // Calling serial monitor for data display
  dht.begin(); // Calling library for humidity input and calculations
  pinMode(LEDPIN1, OUTPUT); // Port location for Red LED
  pinMode(LEDPIN2, OUTPUT); // Port location for Red LED
  pinMode(LEDPIN3, OUTPUT); // Port location for Red LED
  pinMode(LEDPIN4, OUTPUT); // Port location for Red LED
  pinMode(LEDPIN5, OUTPUT); // Port location for Red LED
}

void loop() {
  delay(1000); // Adding a time delay
  float humidity = dht.readHumidity(); // Defining variable to store humidity value from library function

  if (humidity >= 0) { //Checks the range of humidity and executes the following command if True
    digitalWrite(LEDPIN1, HIGH); //Output defined for respective LED
  }
  if (humidity >= 40) { //Checks the range of humidity and executes the following command if True
    digitalWrite(LEDPIN2, HIGH); //Output defined for respective LED
  }
  if (humidity >= 60) { //Checks the range of humidity and executes the following command if True
    digitalWrite(LEDPIN3, HIGH); //Output defined for respective LED
  }
  if (humidity >= 80) { //Checks the range of humidity and executes the following command if True
    digitalWrite(LEDPIN4, HIGH); //Output defined for respective LED
  }
  if (humidity >= 100) { //Checks the range of humidity and executes the following command if True
    digitalWrite(LEDPIN5, HIGH); //Output defined for respective LED
  }

  Serial.print("Current Room Humidity is: "); // Printing data to serial monitor
  Serial.print(humidity); // Printing value of humidity to serial monitor
  Serial.println("%"); // Printing percentage unit to serial monitor
}
```



POST LAB TASKS

Q1: Can we use DHT11 for this application? What is the temperature and humidity range of DHT11 and which one is better, DHT11 or DHT22?

The DHT11 sensor measures humidity and temperature, but it may not be ideal for our application since we want to work with ranges between 0-100% humidity due to its limitations. Typically, it operates within a temperature range of 0°C to 50°C (32°F to 122°F) and a humidity range of 20% to 80%. In contrast, the DHT22 has a wider temperature range, from -40°C to 80°C (-40°F to 176°F), and can measure humidity within the full 0% to 100% range. Additionally, the DHT22 boasts higher accuracy and resolution compared to the DHT11. The choice between these sensors depends on specific requirements; if a wider range and higher accuracy are needed, the DHT22 would be more suitable. However, if the application operates within a narrower range where slightly lower accuracy is acceptable, the DHT11 presents a cost-effective option.

Q2: What is datasheet? What meaningful information can you extract from the DHT22 datasheet that has been shared with you on LMS? Also mention who creates/publishes datasheets.

A datasheet is a document provided by manufacturers that contains detailed technical information about a specific component or product. It typically includes details necessary for understanding and utilizing the component effectively.

Regarding the DHT22 datasheet, some meaningful information that can be extracted includes:

1. Electrical Characteristics: Voltage supply range, current consumption, and communication interface details.
2. Operating Conditions: Temperature and humidity ranges, accuracy, and resolution specifications.
3. Physical Dimensions: Package outline, pin configurations, and layout information.

Datasheets are created and published by the manufacturers of the components. They aim to provide users with precise technical details required for utilizing the component effectively in their projects or systems.

Q3: Define any practical application (excluding above) of Temperature Sensors where they can be used.

Electric heaters can be used in combination with the Temperature Sensor. Electric heaters usually have a manual temperature control mechanism. By incorporating this sensor, users won't have to manually modify the heaters' settings when the temperature in the room changes—whether it grows warmer or colder. Instead, the heaters' operation may be automated.

Q4: Define any practical application (excluding above) of Humidity Sensors where they can be used.

Humidity sensors serve an essential part in preserving a greenhouse's moisture balance, thereby encouraging plant development. They control humidity to protect crop quality and productivity by avoiding situations that might result in mold, fungal development, or excessive evaporation.