IOT102 Progress Test 01

Bài 1.

1. void setup() {
2. pinMode(13,OUTPUT);
3. }
4. void loop() {
5. for(int i=0;i<5;i++){
6. digitalWrite(LED\_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
7. delay(1000); // wait for a second
8. digitalWrite(LED\_BUILTIN, LOW); // turn the LED off by making the voltage LOW
9. delay(1000);}
10. for(int x=0;x<5;x++){
11. digitalWrite(LED\_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
12. delay(2000); // wait for a second
13. digitalWrite(LED\_BUILTIN, LOW); // turn the LED off by making the voltage LOW
14. delay(2000);}
15. }

Bài 2.

1. const int buttonPin = 2; // the number of the pushbutton pin
2. // variables will change:
3. int buttonState = 0; // variable for reading the pushbutton status
4. void setup() {
5. // initialize the LED pin as an output:
6. pinMode(13, OUTPUT);
7. pinMode(12, OUTPUT);
8. pinMode(11, OUTPUT);
9. // initialize the pushbutton pin as an input:
10. pinMode(buttonPin, INPUT);
11. }
12. void loop() {
13. // read the state of the pushbutton value:
14. buttonState = digitalRead(buttonPin);
15. // check if the pushbutton is pressed. If it is, the buttonState is HIGH:
16. if (buttonState == HIGH) {
17. // turn LED on:
18. digitalWrite(13, HIGH);
19. digitalWrite(12, HIGH);
20. digitalWrite(11, HIGH);
21. } else {
22. // turn LED off:
23. digitalWrite(13, LOW);
24. digitalWrite(12, LOW);
25. digitalWrite(11, LOW);
26. }
27. }

Bài 3.

1. // These constants won't change:
2. const int sensorPin = A0; // pin that the sensor is attached to
3. const int ledPin = 9; // pin that the LED is attached to
4. // variables:
5. int sensorValue = 0; // the sensor value
6. int sensorMin = 1023; // minimum sensor value
7. int sensorMax = 0; // maximum sensor value
8. void setup() {
9. // turn on LED to signal the start of the calibration period:
10. pinMode(13, OUTPUT);
11. digitalWrite(13, HIGH);
12. // calibrate during the first five seconds
13. while (millis() < 5000) {
14. sensorValue = analogRead(sensorPin);
15. // record the maximum sensor value
16. if (sensorValue > sensorMax) {
17. sensorMax = sensorValue;
18. }
19. // record the minimum sensor value
20. if (sensorValue < sensorMin) {
21. sensorMin = sensorValue;
22. }
23. }
24. // signal the end of the calibration period
25. digitalWrite(13, LOW);
26. }
27. void loop() {
28. // read the sensor:
29. sensorValue = analogRead(sensorPin);
30. // apply the calibration to the sensor reading
31. sensorValue = map(sensorValue, sensorMin, sensorMax, 0, 255);
32. // in case the sensor value is outside the range seen during calibration
33. sensorValue = constrain(sensorValue, 0, 255);
34. // fade the LED using the calibrated value:
35. analogWrite(ledPin, sensorValue);
36. }