EIOT LAB 7

Aim : To sense the data from sensors and send it to cloud system in simple text files, excel sheets or databases system (Tinkercad)

Objectives :
1.) To understand how sensor data is sent to the cloud.
2.) To learn various public cloud platforms.

FAQ's.

1.) List and state IoT cloud platforms.

i.) Amazon Web Services IoT Platform - It is an extremely scalable platform, claiming to be able to support billions of devices.

ii.) Mircosoft Azure IoT HUB - This platform includes services like machine learning data analytics on cloud platform.

iii.) Google cloud Platform - using this platform keeps us with the advantage of google's private global fibre network.

2.) What is IBM Katson?

IBM Katson is a fully-managed, cloud-hosted service with capabilities for device registration, connectivity, control, rapid visualization and data storage.

3.) What is Amazon Web Service?

Amazon Web Service (AWS) is a comprehensive, evolving cloud computing platform provided by amazon that includes a mixture of Infrastructure as a service (IaaS) Platform as a Service (PaaS), and Packaged Software as a Service (SaaS) offerings. Also provide organization tools such as compute power, database storage and content delivering services.

4.) What is role of cloud in IOT?

An IOT solution should connect and all communication between things, people and process and cloud computing plays a very important role in this collaboration to create a high visibility.

PB_05_Kushagra Suryawanshi

EIOT LAB 7
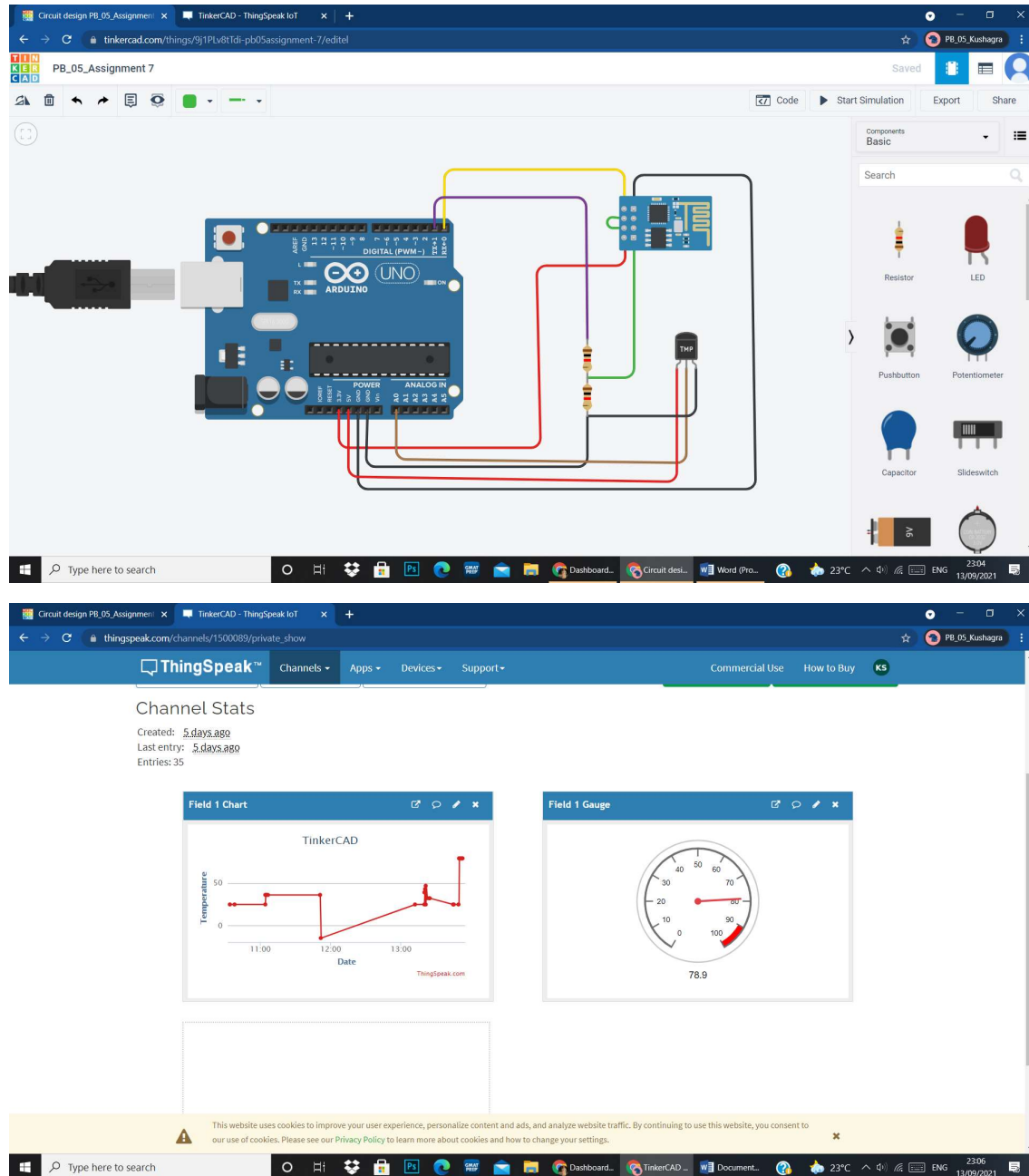
Code:

```
float val, voltage, temp;
String ssid     = "Simulator Wifi";  // SSID to connect to
String password = ""; //virtual wifi has no password
String host     = "api.thingspeak.com"; // Open Weather Map API
const int httpPort   = 80;
String url     = "/update?api_key=O3KUI70363Q4FU1E&field1=";
//Replace XXXXXXXXXXXXXXXX by your ThingSpeak Channel API Key

void setupESP8266(void) {
  // Start our ESP8266 Serial Communication
  Serial.begin(115200);   // Serial connection over USB to computer
  Serial.println("AT");   // Serial connection on Tx / Rx port to ESP8266
  delay(10);       // Wait a little for the ESP to respond
  if (Serial.find("OK"))
    Serial.println("ESP8266 OK!!!");

  // Connect to Simulator Wifi
  Serial.println("AT+CWJAP=\"" + ssid + "\",\"" + password + "\"");
  delay(10);       // Wait a little for the ESP to respond
  if (Serial.find("OK"))
    Serial.println("Connected to WiFi!!!");

  // Open TCP connection to the host:
  //ESP8266 connects to the server as a TCP client.

  Serial.println("AT+CIPSTART=\"TCP\",\"" + host + "\"," + httpPort);
  delay(50);       // Wait a little for the ESP to respond
  if (Serial.find("OK"))
   Serial.println("ESP8266 Connected to server!!!") ;

}

void anydata(void) {

  val=analogRead(A0);
  voltage=val*0.0048828125;
  temp = (voltage - 0.5) * 100.0;

  // Construct our HTTP call
  String httpPacket = "GET " + url + String(temp) + " HTTP/1.1\r\nHost: " + host + "\r\n\r\n";
  int length = httpPacket.length();
```

```
  // Send our message length
  Serial.print("AT+CIPSEND=");
  Serial.println(length);
  delay(10); // Wait a little for the ESP to respond if (!Serial.find(">")) return -1;

  // Send our http request
  Serial.print(httpPacket);
  delay(10); // Wait a little for the ESP to respond
  if (Serial.find("SEND OK\r\n"))
    Serial.println("ESP8266 sends data to the server");

}


void setup() {
  pinMode(A0, INPUT);
  setupESP8266();
}

void loop() {

 anydata();

  delay(1000);

}
```

Conclusion:

Thus, we have studied how to sense the data from sensors and send it to cloud system in simple text files, excel sheets and database system.