1) Configure IOT sensors :

I) Magnetic Flow Sensor – For analysing water consumption.

II) pH Sensor – For analysing water contamination level and quality.

I) Configuring Magnetic Flow Sensor :

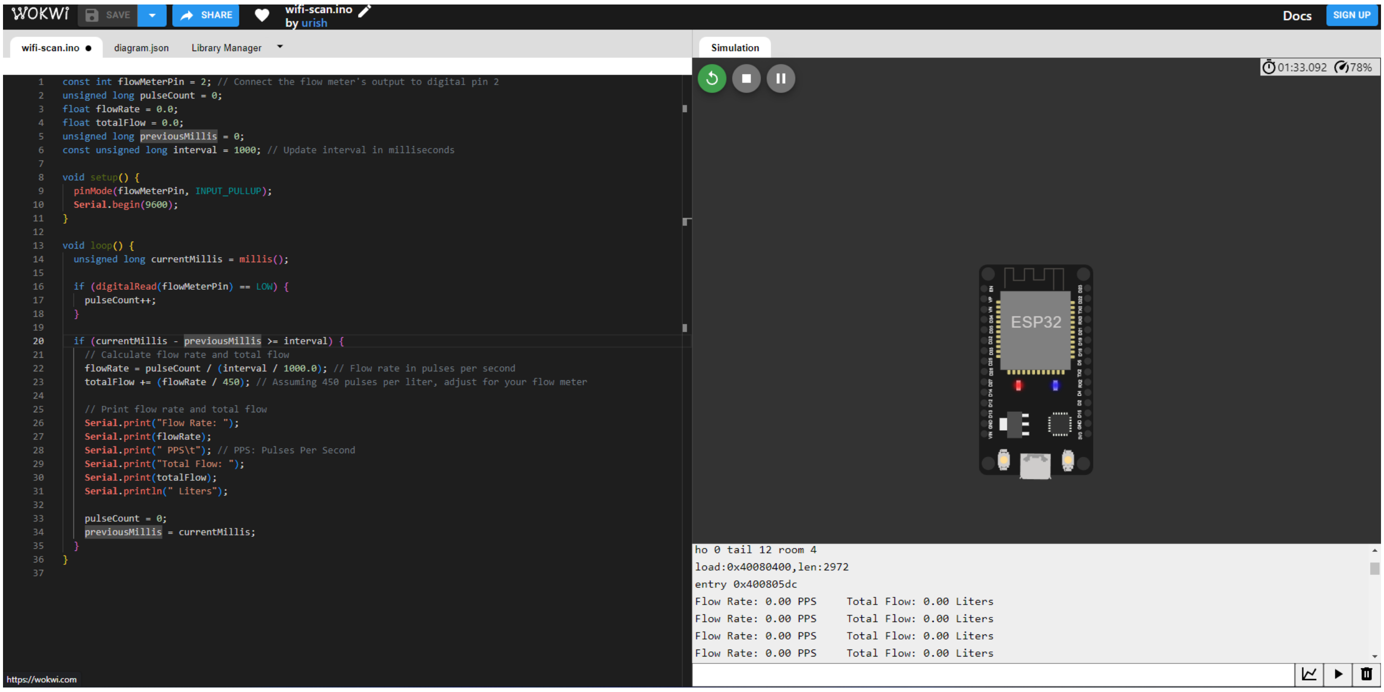
a) Get a Magnetic Flow Sensor.

b) Configure the flow sensor based on our need.

Input:

```
const int flowMeterPin = 2; // Connect the flow meter's output to digital pin
2
unsigned long pulseCount = 0;
float flowRate = 0.0;
float totalFlow = 0.0;
unsigned long previousMillis = 0;
const unsigned long interval = 1000; // Update interval in milliseconds

void setup() {
  pinMode(flowMeterPin, INPUT_PULLUP);
  Serial.begin(9600);
}

void loop() {
  unsigned long currentMillis = millis();

  if (digitalRead(flowMeterPin) == LOW) {
    pulseCount++;
  }

  if (currentMillis - previousMillis >= interval) {
    // Calculate flow rate and total flow
    flowRate = pulseCount / (interval / 1000.0); // Flow rate in pulses per second
    totalFlow += (flowRate / 450); // Assuming 450 pulses per liter, adjust for your flow meter

    // Print flow rate and total flow
    Serial.print("Flow Rate: ");
    Serial.print(flowRate);
    Serial.print(" PPS\t"); // PPS: Pulses Per Second
    Serial.print("Total Flow: ");
    Serial.print(totalFlow);
    Serial.println(" Liters");

    pulseCount = 0;
    previousMillis = currentMillis;
  }
}
```

Output:



```
const int flowMeterPin = 2; // Connect the flow meter's output to digital pin 2
unsigned long pulseCount = 0;
float flowRate = 0.0;
float totalFlow = 0.0;
unsigned long previousMillis = 0;
const unsigned long interval = 1000; // Update interval in milliseconds

void setup() {
  pinMode(flowMeterPin, INPUT_PULLUP);
  Serial.begin(9600);
}

void loop() {
  unsigned long currentMillis = millis();

  if (digitalRead(flowMeterPin) == LOW) {
    pulseCount++;
  }

  if (currentMillis - previousMillis >= interval) {
    // Calculate flow rate and total flow
    flowRate = pulseCount / (interval / 1000.0); // Flow rate in pulses per second
    totalFlow += (flowRate / 450); // Assuming 450 pulses per liter, adjust for your flow meter

    // Print flow rate and total flow
    Serial.print("Flow Rate: ");
    Serial.print(flowRate);
    Serial.print(" PPS\t"); // PPS: Pulses Per Second
    Serial.print("Total Flow: ");
    Serial.print(totalFlow);
    Serial.println(" Liters");

    pulseCount = 0;
    previousMillis = currentMillis;
  }
}

```

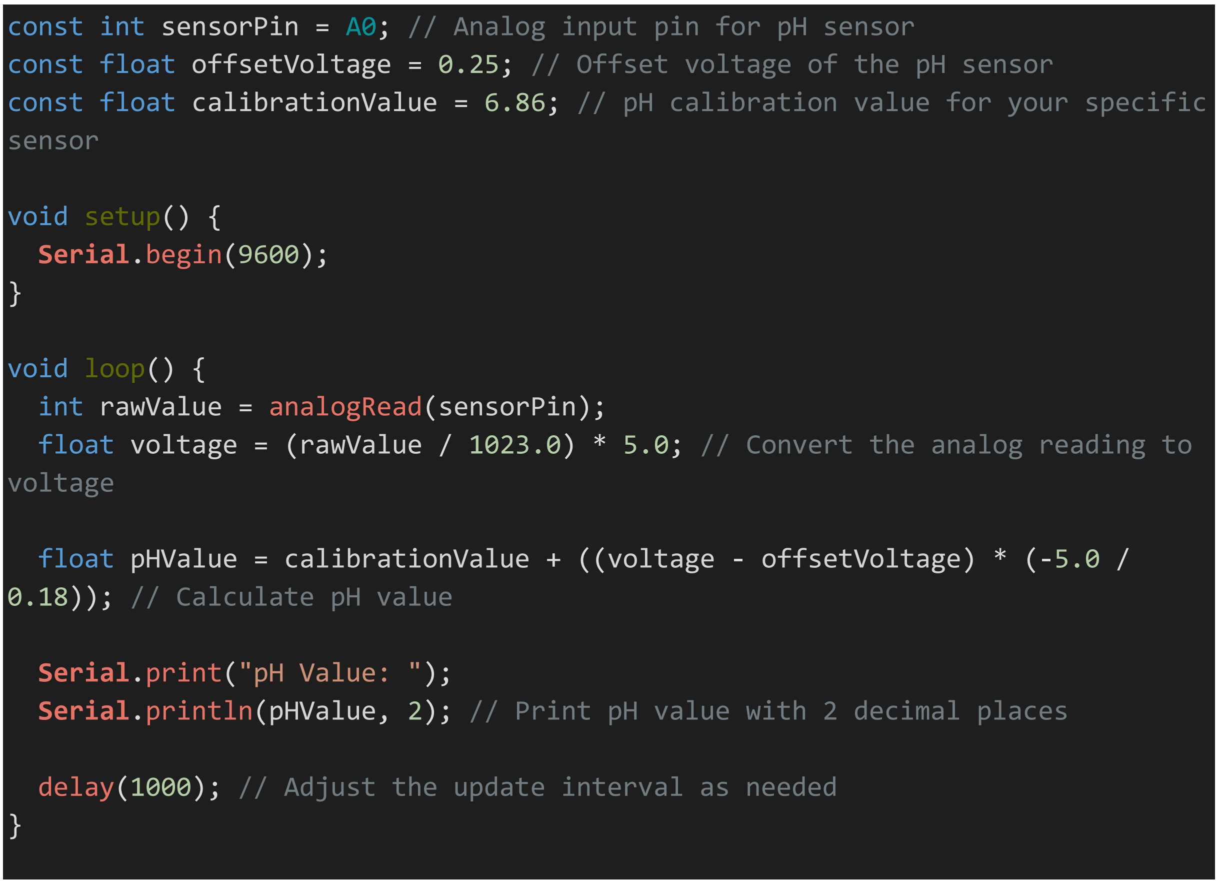
```
ho 0 tail 12 room 4
load:0x40080400,len:2972
entry 0x400805dc
Flow Rate: 0.00 PPS     Total Flow: 0.00 Liters
Flow Rate: 0.00 PPS     Total Flow: 0.00 Liters
Flow Rate: 0.00 PPS     Total Flow: 0.00 Liters
Flow Rate: 0.00 PPS     Total Flow: 0.00 Liters
```

II) Configuring pH Sensor :

a) Get a pH Sensor.

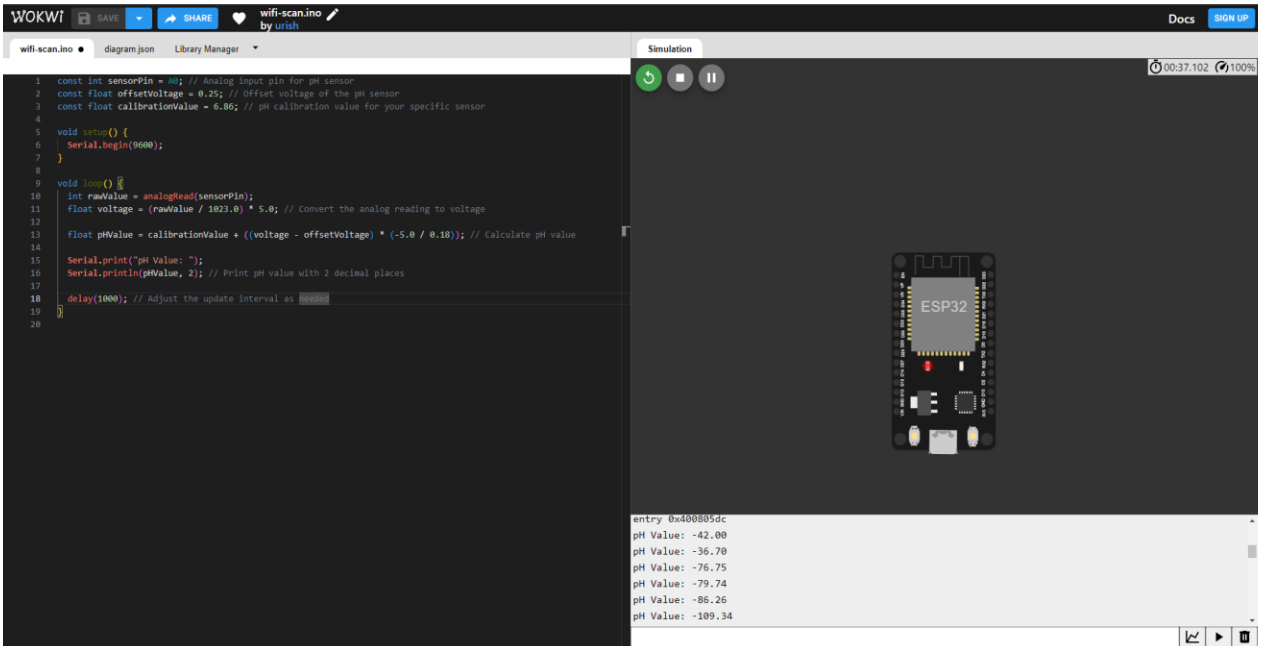
b) Configure the pH sensor based on our need.

Input:



```
const int sensorPin = A0; // Analog input pin for pH sensor
const float offsetVoltage = 0.25; // Offset voltage of the pH sensor
const float calibrationValue = 6.86; // pH calibration value for your specific sensor

void setup() {
  Serial.begin(9600);
}

void loop() {
  int rawValue = analogRead(sensorPin);
  float voltage = (rawValue / 1023.0) * 5.0; // Convert the analog reading to voltage

  float pHValue = calibrationValue + ((voltage - offsetVoltage) * (-5.0 / 0.18)); // Calculate pH value

  Serial.print("pH Value: ");
  Serial.println(pHValue, 2); // Print pH value with 2 decimal places

  delay(1000); // Adjust the update interval as needed
}
```

Output:



```
const int sensorPin = A0; // Analog input pin for pH sensor
const float offsetVoltage = 0.25; // Offset voltage of the pH sensor
const float calibrationValue = 6.86; // pH calibration value for your specific sensor

void setup() {
  Serial.begin(9600);
}

void loop() {
  int rawValue = analogRead(sensorPin);
  float voltage = (rawValue / 1023.0) * 5.0; // Convert the analog reading to voltage

  float pHValue = calibrationValue + ((voltage - offsetVoltage) * (-5.0 / 0.18)); // Calculate pH value

  Serial.print("pH Value: ");
  Serial.println(pHValue, 2); // Print pH value with 2 decimal places

  delay(1000); // Adjust the update interval as needed
}
```

```
entry 0x400805dc
pH Value: -42.00
pH Value: -36.70
pH Value: -76.75
pH Value: -79.74
pH Value: -86.26
pH Value: -109.34
```

2) Python script on the IOT sensors to send real-time water consumption data to the data-sharing platform:

Steps:

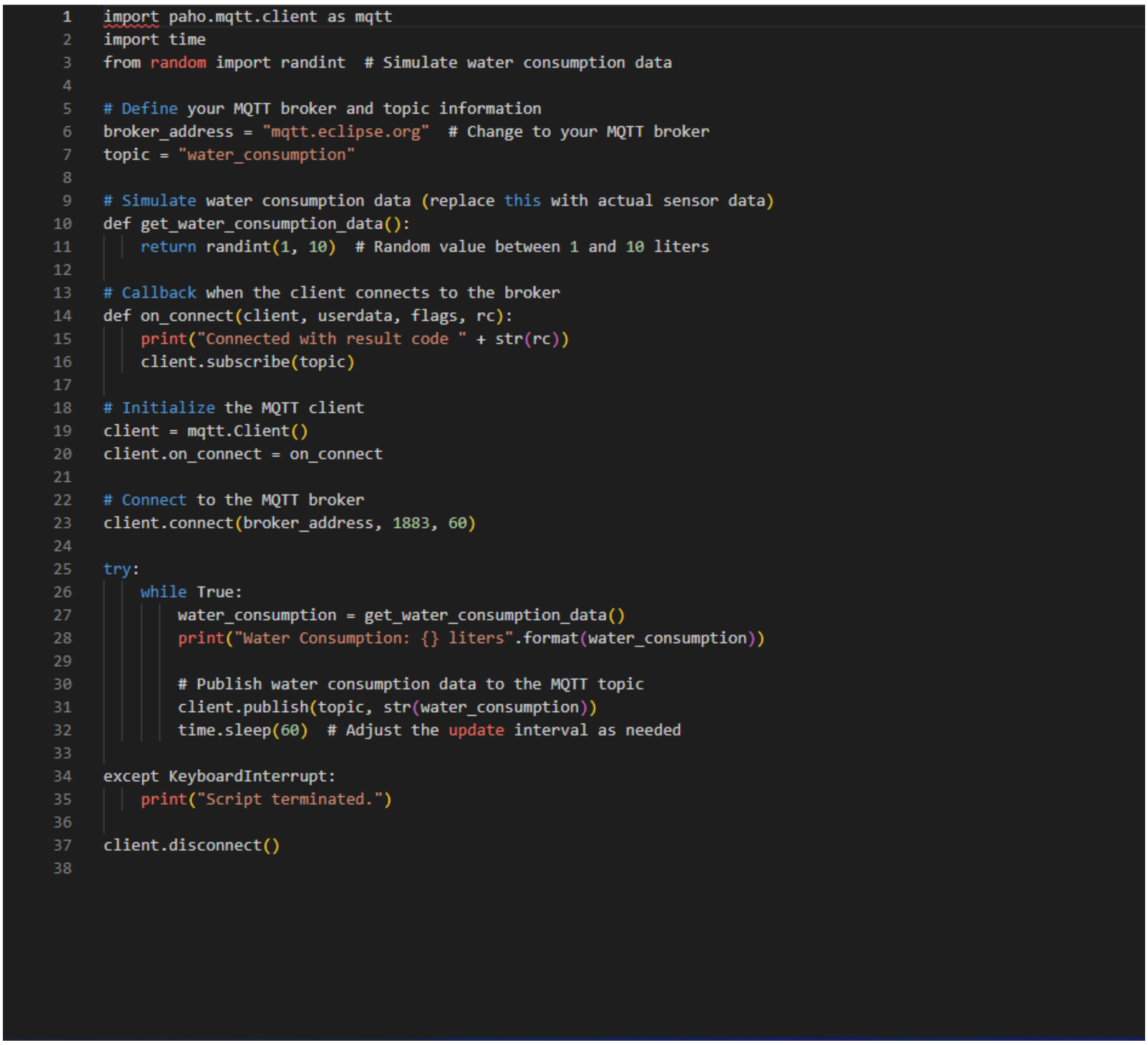
1. IOT Device: This could be a microcontroller (e.g., Raspberry Pi, Arduino) with a compatible water consumption sensor (e.g., flow sensor).

2. Internet Connectivity: Ensure your IOT device can connect to the internet, either through Wi-Fi, Ethernet, or a cellular module.

3. Data-Sharing Platform: Choose a platform where you want to send the data. Common choices include cloud-based services like AWS, Azure, Google Cloud, or dedicated IOT platforms.

4. Libraries: You may need specific libraries or SDKs to communicate with your chosen data-sharing platform and sensor.

Code:



```python
import paho.mqtt.client as mqtt
import time
from random import randint  # Simulate water consumption data

# Define your MQTT broker and topic information
broker_address = "mqtt.eclipse.org"  # Change to your MQTT broker
topic = "water_consumption"

# Simulate water consumption data (replace this with actual sensor data)
def get_water_consumption_data():
    return randint(1, 10)  # Random value between 1 and 10 liters

# Callback when the client connects to the broker
def on_connect(client, userdata, flags, rc):
    print("Connected with result code " + str(rc))
    client.subscribe(topic)

# Initialize the MQTT client
client = mqtt.Client()
client.on_connect = on_connect

# Connect to the MQTT broker
client.connect(broker_address, 1883, 60)

try:
    while True:
        water_consumption = get_water_consumption_data()
        print("Water Consumption: {} liters".format(water_consumption))

        # Publish water consumption data to the MQTT topic
        client.publish(topic, str(water_consumption))
        time.sleep(60)  # Adjust the update interval as needed

except KeyboardInterrupt:
    print("Script terminated.")

client.disconnect()
```

Conclusion:

*In this we have developed a program for configuring the IOT sensors like Magnetic Flow Sensor and pH Sensor for checking water consumption and checking the contamination in the water by analysing the pH value.

*Then we have developed a program for sending real-time water consumption data to the data-sharing platform using MQTT services.