**Smart water fountains**

**Set Up Hardware:**

Connect IoT sensors (e.g., water level, temperature) to your Raspberry Pi.

Ensure you have the necessary libraries to interact with the sensors (e.g., RPi.GPIO for GPIO pins).

**Install Required Libraries:**

pip install paho-mqtt

**Create a Python Script:**

import time

import json

import RPi.GPIO as GPIO

import paho.mqtt.client as mqtt

# Define your MQTT broker's details

MQTT\_BROKER = "mqtt.example.com"

MQTT\_PORT = 1883

MQTT\_TOPIC = "water\_fountain/status"

# Initialize GPIO pins for sensors

WATER\_LEVEL\_PIN = 17

TEMPERATURE\_PIN = 18

GPIO.setmode(GPIO.BCM)

GPIO.setup(WATER\_LEVEL\_PIN, GPIO.IN)

GPIO.setup(TEMPERATURE\_PIN, GPIO.IN)

def on\_connect(client, userdata, flags, rc):

print("Connected to MQTT broker with code " + str(rc))

client = mqtt.Client()

client.on\_connect = on\_connect

client.connect(MQTT\_BROKER, MQTT\_PORT, 60)

while True:

water\_level = GPIO.input(WATER\_LEVEL\_PIN)

temperature = GPIO.input(TEMPERATURE\_PIN)

# Prepare the data as a JSON payload

data = {

"water\_level": water\_level,

"temperature": temperature

}

# Publish the data to the MQTT topic

client.publish(MQTT\_TOPIC, json.dumps(data))

time.sleep(5) # Adjust the interval as needed

client.loop\_forever()

**Run the Script:**

Run the Python script on your Raspberry Pi. It will read data from the sensors and publish it to the MQTT topic.

**Set Up the Platform:**

You'll need to set up an MQTT broker on your platform to receive the data. You can use popular cloud platforms like AWS IoT, Google Cloud IoT, or a self-hosted MQTT broker.