**TRAFFIC MANAGEMENT**

**Introduction:**

* IOT Based Traffic Management System
* An Internet of Things (IoT)-enabled intelligent traffic management system can solve pertinent issues by leveraging technologies like wireless connectivity & intelligent sensors.
* Considered a cornerstone of a smart city, they help improve the comfort and safety of drivers, passengers & pedestrians.

**Code:**

Import paho.mqtt.client as mqtt

Import time

# Define MQTT broker settings

Broker\_address = “your\_broker\_address”

Port = 1883

Topic = “traffic/management”

# Define a callback for when a message is received

Def on\_message(client, userdata, message):

Message\_payload = str(message.payload.decode(“utf-8”)

Print(f”Received message: {message\_payload}”)

# Add traffic management logic here

# For example, control traffic lights or send instructions to IoT devices

# Create an MQTT client

Client = mqtt.Client(“TrafficManagementClient”)

Client.connect(broker\_address, port)

# Subscribe to the topic

Client.subscribe(topic)

Client.on\_message = on\_message

# Main loop

While True:

# Implement your traffic management logic here

# For example, analyze sensor data and publish instructions to IoT devices

# Simulate sending traffic updates

Traffic\_data = “Traffic is congested on Main Street”

Client.publish(topic, traffic\_data)

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

# Start the MQTT client loop

Client.loop\_forever()

**Code Explanation:**

* This code is a basic example of how you might set up an IoT traffic management system that communicates with IoT devices via MQTT.
* In practice, you’d replace “your\_broker\_address” with the address of your MQT
* T broker and implement more complex logic for traffic management, such as analyzing sensor data and controlling traffic lights or other devices.
* Please note that a complete IoT traffic management system would require extensive development and integration of various hardware components and sensors.
* The above code serves as a starting point for building such a system.