

IUT DE COLMAR

SAE 24

PROJET INTÉGRATIF

Partie Collecte

RT11

Martin BAUMGAERTNER

RT12

Mehdi REHM

RT11

Sâji DOGHMANE

23 juin 2022

Table des matières

1	Introduction	4
2	Récupération de données	4
2.1	Configuration du script	4
2.2	Exécution du script	5
3	Sauvegarde des valeurs dans la base de données	5
3.1	Explications	5
3.2	Configuration du script	5

Table des codes

1	Configuration des IDs de connexion	4
2	Installation des paquets nécessaire au script MQTT	4
3	Connexion à la base de données	5
4	Création des tables	6

Table des figures

1	Récupération des données	5
---	------------------------------------	---

1 Introduction

Au courant de l'année nous avons pu voir différents mode de collecte de données, notamment la récupération via MQTT. D'abord, qu'est-ce que MQTT ? MQTT, pour "Message Queuing Telemetry Transport", est un protocole open source de messagerie qui assure des communications non permanentes entre des appareils par le transport de leurs messages.

Le but de cette partie étant de récupérer des données. Nous devons réceptionner des valeurs de température sur une pièce. Nous devons être capable de les afficher selon les critères définis, et les intégrer dans une base de données qui nous servira plus tard pour la partie Web.

2 Récupération de données

2.1 Configuration du script

Pour pouvoir récupérer les données depuis le MQTT, j'ai donc dû adapter le script python que nous a été donné dans le diaporama et j'ai dû l'adapter pour qu'il récupère les bonnes données.

J'ai modifié les lignes suivantes, en y ajoutant les bonnes valeurs de connexion :

```
broker = 'test.mosquitto.org'
topic = "IUT/Colmar/SAE24/Maison1"
```

Code 1 – Configuration des IDs de connexion

Par la suite j'ai dû installer un paquet qui était prérequis pour que le script puisse s'exécuter correctement à savoir :

```
pip3 install paho-mqtt python-etcd
```

Code 2 – Installation des paquets nécessaire au script MQTT

2.2 Exécution du script

Au moment de l'exécution du programme, j'obtiens bien les valeurs que nous voulions recevoir comme nous pouvons le constater ci dessous :

```
martin@SRV-UBUNTU:~/SAE-24/SAE-24/Collecte$ python3 mqtt.py
Connected to MQTT Broker!
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:45:41,temp=23.46' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:45:46,temp=10.92' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:45:51,temp=16.41' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:45:56,temp=5.22' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:01,temp=28.23' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:06,temp=17.01' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:11,temp=17.59' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:16,temp=10.97' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:21,temp=22.23' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:26,temp=26.55' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:31,temp=20.13' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:36,temp=8.38' from 'IUT/Colmar/SAE24/Maison1' topic
Received 'Id=B8A5F3569EFF,piece=sejour,date=21/06/2022,time=13:46:41,temp=13.44' from 'IUT/Colmar/SAE24/Maison1' topic
```

FIGURE 1 – Récupération des données

Le problème est qu'il nous était demandé de récupérer un ID différent à chaque fois qu'une valeur de température est reçue.

3 Sauvegarde des valeurs dans la base de données

3.1 Explications

L'objectif finale de cet exercice d'ensuite pouvoir sauvegarder les données dans une base de données pour ensuite permettre l'affichage de ces données dans une page web. Pour ce faire, j'ai donc écrit un script qui permet premièrement la réception des données MQTT, puis qui les stock dans une base de données.

3.2 Configuration du script

Le script est composé de plusieurs parties. Il commence par se connecter à la base de données :

```
try:
    db=_mysql.connect("10.37.129.3","martin","martin", "temp")
except OperationalError:
    db=_mysql.connect("10.37.129.3","martin","martin")
    db.query("CREATE DATABASE martin")
    db.query("USE martin")
```

Code 3 – Connexion à la base de données

L'adresse IP renseigné co Puis, on défini les tables que l'on créer et les valeurs que l'on ajoute etc. :

```
db.ping(True)
db.query("""
CREATE TABLE IF NOT EXISTS martin.sensors (
    id INT NOT NULL AUTO_INCREMENT,
    macaddr VARCHAR(12) NOT NULL,
    piece VARCHAR(50) NOT NULL,
    emplacement VARCHAR(50),
    nom VARCHAR(50),
    UNIQUE (macaddr),
    PRIMARY KEY (id))
""")

db.query("""
CREATE TABLE IF NOT EXISTS martin.sensors_data (
    id INT NOT NULL AUTO_INCREMENT,
    sensor_id INT NOT NULL,
    CONSTRAINT sensorFK
        FOREIGN KEY (sensor_id)
        REFERENCES martin.sensors(id),
    datetime DATETIME NOT NULL,
    temp FLOAT NOT NULL,
    PRIMARY KEY (id))
""")
```

Code 4 – Création des tables