

TP : Logement Éco-responsable

Table des matières

Partie 1 : Base de données	2
I. Spécifications de la base de données	2
II. Remplissage de la base de données	2
 Partie 2 : Serveur RESTful	2
 Partie 3 : HTML/CSS/Javascript.....	3
 Ressources utilisées	4

Matériel utilisé :

- Module Wifi ESP 8266
- Capteur de température et d'humidité DHT 22

Partie 1 : Base de données

I. Spécifications de la base de données

Question 1

- Dossier partie 1

Question 2 à 8 :

Fichier *logement.sql* créé contenant :

- Un logement 4 pièces
- 4 types de capteurs
- 4 capteurs
- Minimum 2 mesures par capteur (capteur 2 et 4 rattachés à DHT 22 => plusieurs mesures)
- 4 factures

II. Remplissage de la base de données

Fichier *remplissage.py* créé contenant une fonction qui identifie les jours 3 derniers jours ouvrés du mois et sélectionne une date parmi ces 3 jours aléatoirement, garantissant une cohérence temporelle dans les données générées. Ensuite, j'insère ces données dans les tables mesure et facture (grâce à deux autres fonctions) avec des valeurs aléatoires réalistes pour les capteurs et les montants des factures, tout en m'assurant que les changements sont correctement sauvegardés dans la base de données.

Partie 2 : Serveur RESTful

Un serveur RESTful est un serveur web qui respecte les principes de l'architecture REST (Representational State Transfer). REST est un style d'architecture logicielle utilisé principalement pour concevoir des API web simples, légères et scalables, permettant la communication entre les systèmes distribués via le protocole HTTP.

Le framework Python que j'ai utilisé pour ce projet est Flask car j'avais déjà commencé par l'apprendre lors de la première partie d'IoT mais aussi parce qu'il bénéficie

d'une large communauté active, d'une documentation riche, et d'une vaste bibliothèque d'extensions.

Chaque question est traitée dans le dossier partie2 (toutes les questions ont été validé par M. Elyoth HARIAN)

Server.py => Question 1

Server2.py => Question 2 + preuve

Server3.py => Question 3 + preuve

Server4.py => Question 4 + code arduino pour esp8266

Partie 3 : HTML/CSS/Javascript

Lire README

Ressources utilisées :

- <https://flask.palletsprojects.com/>
- <https://flask-sqlalchemy.palletsprojects.com/>
- <https://flask-restful.readthedocs.io/>
- <https://flask-caching.readthedocs.io/>
- <https://www.freecodecamp.org/news/tag/flask/>
- <https://github.com/humiaozuzu/awesome-flask>
- <https://stackoverflow.com/questions/tagged/flask>
- <https://github.com/pallets/flask/tree/main/examples>
- <https://chatgpt.com/g/g-p-676022eaf4c08191b571f0e24175f643-site-web/c/6760239e-90bc-8009-a797-afb5a6fe81dc>

