

Documentation Globale Python

Groupe : 1B05

GUILLEVIC Yann

THOMPSON Victor

CABRILLAC Anthony

DOUVILLE Léo

HU Shiyu

- 1) Expliquer la structure du code
- 2) Décrire les étapes d'installation
- 3) Décrire les étapes de lancement du code

1) Expliquer la structure du code :

Le programme python est divisé en 2 parties distinctes :

- Le fichier de configuration
- Le programme principal

Le fichier `config_yaml.yaml` est le fichier de configuration accessible par l'utilisateur. C'est ici que le client peut décider quels sont les paramètres d'exécution du code ainsi que la fréquence d'émission des données et des seuils maximaux de celles-ci. Ainsi, il est séparé en trois parties :

- En haut la partie "config" dans laquelle l'utilisateur peut modifier les paramètres des serveurs d'accès, des ports et des appareils.
- La partie "fréquence" fait référence au délai d'attente que va respecter le main pour écrire les données dans le fichier. Ce délai est en seconde.
- Enfin, la partie "seuilMax" qui définit les valeurs maximales que ne doivent pas dépasser les données.

Au niveau du fichier python, celui-ci s'articule autour d'une méthode principale : la méthode `get_data`.

Celle-ci va d'abord récupérer les données émises par les capteurs et les affecter à des variables. Exemple : `donneeCo2 = jsonMsg["object"]["co2"]`. Ici, la variable `donneeCo2` récupère la valeur de CO2 émise par le capteur. Ces données sont également affichées dans le terminal pour les tests.

Ensuite, arrive la partie de gestion des seuils maximaux. Celle-ci utilise simplement une vérification par une boucle `if` simple. Si la valeur sortie par le capteur est supérieure à la valeur maximum, il y a erreur et on affiche une alerte dans le terminal.

Enfin, la gestion de la fréquence d'émission est un peu plus compliquée. Nous utilisons `import signal` et notamment la méthode `signal.alarm` qui va envoyer un signal dès que le temps en seconde passé en paramètre est écoulé (correspondant à la fréquence voulue par le client). Le signal envoyé sera réceptionné plus tard dans le programme pour exécuter une méthode.

Nous arrivons à la méthode "écriture". Celle-ci démarre à la réception du signal émis par l'instruction `signal.alarm`. Cette méthode va ouvrir un fichier texte en écriture, en le créant si il n'existe pas, et va écrire les données reçues dedans. Cela permettra de respecter la fréquence d'écriture des données choisie par le client.

Au niveau de l'exécution du programme, les paramètres du fichier de configuration sont chargés dans une variable "param" et ont l'architecture d'un dictionnaire. C'est pour cela que pour la gestion du maximum de l'humidité ,par exemple, la syntaxe est la suivante : "param["seuilMax"]["Hum"]". On récupère la valeur de la variable param correspondant à la clé "seuilMax" et "Hum".

Pour finir, le client va se connecter sur le serveur de son choix et s'abonner au flux se son choix également.

L'exécution tourne en boucle pour récupérer les données continuellement via l'instruction "client.loop_forever()".

2) Décrire les étapes d'installation :

Si vous souhaitez lancer le programme python sur Windows via PyCharm, pour pouvez vous renseigner sur notre documentation technique JAVA dans la partie Procédure d'installation.

Tout d'abord, vous devez posséder un logiciel capable d'interpréter du code python, comme par exemple Visual Studio Code. Lorsque votre logiciel est ouvert, vous pouvez glisser-déposer les fichiers que vous souhaitez exécuter dans votre projet dans le logiciel.

Si vous êtes sur VS Code, vous avez 2 méthodes pour lancer le code :

- Premièrement, vous pouvez cliquer en haut à gauche sur l'onglet "File" puis "new text file" et choisir le langage python. Vous pourrez ainsi y copier les codes python que vous aurez récupéré. Pour l'exécuter vous pouvez ouvrir un nouveau terminal et y copier la commande suivante pour exécuter le main.py
`: python3 main.py`

N'oubliez pas d'importer également le fichier de configuration config_yaml.yaml.

- Deuxièmement, vous pouvez installer l'extension python dans le menu des extensions à gauche et lancer directement le code avec l'onglet "exécuter".

Pour ces deux manières, si vous êtes sur Ubuntu par exemple, il vous faudra peut-être installer l'extension paho avec la commande suivante : `sudo pip3 install paho-mqtt`.

3) Décrire les étapes de lancement du code :

Si vous souhaitez lancer le programme python sur Windows via PyCharm, pour pouvez vous renseigner sur notre documentation technique JAVA dans la partie Procédure d'installation.

Si vous souhaitez lancer le main.py via des terminaux, vous pouvez utiliser la commande : `python3 main.py`.

Attention toutefois à bien placer le fichier de configuration dans le fichier contenant votre main. Le programme ira de lui-même récupérer les données dans le fichier de configuration.