

<b>Travaux pratique</b>
<b>La programmation en Python</b>
<b>Introduction à MQTT avec Python</b>

<b>BTS SN-IR</b>
<b>2<sup>ème</sup> année</b>
Page 1 sur 3

MQTT (Message Queuing Telemetry Transport) est un protocole de messagerie léger conçu pour les situations où la bande passante est limitée ou la connexion réseau est peu fiable. Il est souvent utilisé dans l'Internet des objets (IoT) pour la communication entre des appareils distants. En Python, il existe une bibliothèque populaire appelée paho-mqtt qui facilite l'utilisation de MQTT.

### Installation de la bibliothèque paho-mqtt

```
pip install paho-mqtt
```

### Concepts de base

1. **Broker MQTT** : Un serveur qui gère la distribution des messages entre les clients. Vous pouvez utiliser un broker public comme celui fourni par HiveMQ, ou installer votre propre broker (par exemple, Mosquitto).
2. **Client MQTT** : Une application qui se connecte au broker pour publier ou recevoir des messages.

### Exemple de code MQTT Publisher (Éditeur)

```
import paho.mqtt.publish as publish

# Définir le broker
broker_address = "broker.hivemq.com"

# Définir le sujet (topic) sur lequel publier
topic = "snir/bts/2023/msg"

# Définir le message à publier
message = "Bonjour, les BTS SNIR2"

# Publier le message
publish.single(topic, message, hostname=broker_address)
```

## Exemple de code MQTT Subscriber (Abonné)

```
import paho.mqtt.client as mqtt

# Définir le broker
broker_address = "broker.hivemq.com"

# Définir le sujet (topic) auquel s'abonner
topic = "snir/bts/2023/msg"

# Fonction appelée lorsqu'un message est reçu
def on_message(client, userdata, msg):
    print(f"Message reçu sur le sujet {msg.topic}: {msg.payload.decode()}")

# Configurer le client MQTT
client = mqtt.Client()
client.on_message = on_message

# Se connecter au broker
client.connect(broker_address)
client.subscribe(topic)

# Boucle pour écouter les messages
client.loop_forever()
```

## Explications

- Le code Publisher utilise la fonction `publish.single()` pour envoyer un message sur un sujet donné à un broker MQTT.
- Le code Subscriber utilise la bibliothèque `paho-mqtt` pour créer un client MQTT, se connecter au broker, s'abonner à un sujet, et enfin, écouter les messages en utilisant une boucle infinie (`client.loop_forever()`).

Exécutez le code Publisher pour publier un message.

Exécutez le code Subscriber pour écouter les messages sur le même sujet.

Vous devriez voir le message publié par le Publisher s'afficher dans la console du Subscriber.