

```

## Importation des bibliothèques
import serial
from kafka import KafkaProducer
import time
import pickle

## Création d'un fichier pour sauvegarder les données issues du capteur (optionel)
listeDonnee = []
pickle.dump( listeDonnee, open( "saveLISTEDONNEE.txt", "wb" ) )

## Paramètre de la carte Arduino et de l'ip du serveur Kafka
portArduino = "COM3"
adresseKafka = 'localhost:9092'

##Fonction pour permettre et récupérer le nom du topic ainsi que la valeur (!dépend )
def separateur_serial(chaine, separateur = "!"):
    indice = -1

    for i in range (len(chaine) -1, -1 , -1):
        if chaine[i] == separateur:
            indice = i
            break
    topic = chaine[:indice]
    valeur = chaine[indice +1:]
    return topic,valeur
    return valeur

## Initialisation de la connection avec la carte Arduino
serial_port = serial.Serial(port = portArduino, baudrate = 9600)

## Initialisation de la connection avec le sreveur Kafka
producer = KafkaProducer(bootstrap_servers=adresseKafka)

## On boucle à l'infini pour transmettre continuellement des données
while True:
    # Récupération de la donnée issue de la carte Arduino
    msg = serial_port.readline()

    # convertit le msg de l'ARDUINO en chaine de caractères
    msg = msg.decode('utf-8')[:-2] # enlève les \r\n
    # Récupération du temps pour la sauvegarde dans un fichier externe (optionel)
    temp = time.time()
    donnee = (msg, temp)
    print(msg)

    # sauvegarde des données dans un fichier de sauvegarde !consomme du temps si le fichier est trop gr
    and
    listeDonnee = pickle.load( open( "saveLISTEDONNEE.txt", "rb" ) )
    listeDonnee.append(donnee)
    pickle.dump( listeDonnee, open( "saveLISTEDONNEE.txt", "wb" ) )

    topic,message = separateur_serial(msg)
    # print(topic,message)

```

```
# convertit le message en bytes  
message_bytes = bytes(message, 'utf-8')
```

```
#envoi de la donnée sur le serveur Kafka  
producer.send(topic, message_bytes)
```

```
## code pour fermeture du producer  
## producer.close()
```