Example Kafka Consumer SutowBrett

July 18, 2021

[1]:
```python
import json

from kafka import KafkaConsumer
```

0.0.1 Configuration Parameters

TODO: Change the configuration prameters to the appropriate values for your setup.

[2]:
```python
config = dict(
    bootstrap_servers=['kafka.kafka.svc.cluster.local:9092'],
    first_name='Brett',
    last_name='Sutow'
)

config['client_id'] = '{}{}'.format(
    config['last_name'],
    config['first_name']
)
config['topic_prefix'] = '{}{}'.format(
    config['last_name'],
    config['first_name']
)

config
```

[2]:
```
{'bootstrap_servers': ['kafka.kafka.svc.cluster.local:9092'],
 'first_name': 'Brett',
 'last_name': 'Sutow',
 'client_id': 'SutowBrett',
 'topic_prefix': 'SutowBrett'}
```

Create a consumer without subscribing to any particular topic

[3]:
```python
general_consumer = KafkaConsumer(
    bootstrap_servers=config['bootstrap_servers']
)
```

List all topics you are currently allowed to view

[4]:

```
general_consumer.topics()
```

[4]:

```
{'BankarDhiraj-accelerations',
 'BankarDhiraj-locations',
 'CatesEvelyn-accelerations',
 'CatesEvelyn-aceelerations',
 'CatesEvelyn-locations',
 'DoeJohn-accelerations',
 'DoeJohn-locations',
 'HarryTom-locations',
 'MestonRyan-accelerations',
 'MestonRyan-locations',
 'MisraAbhigyan-accelerations',
 'MisraAbhigyan-joined',
 'MisraAbhigyan-locations',
 'MisraAbhigyan-simple',
 'MisraAbhigyan-windowed',
 'MooreGloria-accelerations',
 'MooreGloria-locations',
 'NerallaRavindra-accelerations',
 'NerallaRavindra-locations',
 'OmprakashSantosh-accelerations',
 'OmprakashSantosh-locations',
 'SutowBrett-accelerations',
 'SutowBrett-locations'}
```

Close the consumer, waiting indefinitely for any needed cleanup.

[5]:

```
general_consumer.close()
```

[6]:

```
def create_kafka_consumer(topics, config=config):
    bootstrap_servers = config['bootstrap_servers']
    client_id = config['client_id']
    topic_prefix = config['topic_prefix']
    topic_list = ['{}-{}'.format(topic_prefix, topic) for topic in topics]

    return KafkaConsumer(
        *topic_list,
        client_id=client_id,
        bootstrap_servers=bootstrap_servers,
        auto_offset_reset='earliest',
        enable_auto_commit=False,
        value_deserializer=lambda x: json.loads(x)
    )

consumer = create_kafka_consumer(['locations', 'accelerations'])
```

Gets a list of this consumer's current subscriptions

[7]:
```
consumer.subscription()
```

[7]:
```
{'SutowBrett-accelerations', 'SutowBrett-locations'}
```

The following function prints messages from the current consumer subscriptions. It will continue until manually stopped.

[8]:
```
def print_messages(consumer=consumer):
    try:
        for message in consumer:
                msg_metadata = 'Message metadata: {}:{}:{}'.format(
                    message.topic, message.partition, message.offset
                )

                if message.key is not None:
                    msg_key = message.key.decode('utf-8')
                else:
                    msg_key = ''
                msg_value = json.dumps(message.value, indent=2)
                msg_value = '\n'.join(['  {}'.format(value) for value in␣
 ↪msg_value.split('\n')])

                print('Message metadata:')
                print('  Topic: {}'.format(message.topic))
                print('  Partition: {}'.format(message.partition))
                print('  Offset: {}'.format(message.offset))
                print('Message Key: {}'.format(msg_key))
                print('Message Value:')
                print(msg_value)
                print()
    except KeyboardInterrupt:
        print("STOPPING MESSAGE CONSUMER")

print_messages()
```

```
Message metadata:
  Topic: SutowBrett-locations
  Partition: 0
  Offset: 0
Message Key: 5a48ef4e7a404009898e0aa2da17b1b7
Message Value:
  {
    "key1": "value1",
    "key2": "value2"
  }

STOPPING MESSAGE CONSUMER
```

Close the consumer, waiting indefinitely for any needed cleanup.

[9]:
```
consumer.close()
```

[ ]:
```

```