

## NagarajVinay - Consumer 8.2

May 9, 2021

```
[1]: import json

from kafka import KafkaConsumer
```

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

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': 'Vinay',
      'last_name': 'Nagaraj',
      'client_id': 'NagarajVinay',
      'topic_prefix': 'NagarajVinay'}
```

Create a consumer without subscribing to any particular topic

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

List all topics you are currently allowed to view

```
[4]: general_consumer.topics()
```

```
[4]: {'AdamCurry-accelerations',
      'AdamCurry-locations',
      'AtmadjaRio-accelerations',
      'AtmadjaRio-joined',
      'AtmadjaRio-locations',
      'AtmadjaRio-simple',
      'AtmadjaRio-windowed',
      'BranchJolene-accelerations',
      'BranchJolene-locations',
      'ByardMegan-accelerations',
      'ByardMegan-locations',
      'CorpuzCaleb-accelarations',
      'CorpuzCaleb-accelerations',
      'CorpuzCaleb-locations',
      'CurryAdam-accelerations',
      'CurryAdam-locations',
      'DandaRama-accelerations',
      'DandaRama-locations',
      'DoeJohn-accelerations',
      'DoeJohn-locations',
      'DoeJohn-simple',
      'DoeagerJohn-accelerations',
      'DoeagerJohn-locations',
      'EagerBrian-accelerations',
      'EagerBrian-locations',
      'EdmundsEvan-accelerations',
      'EdmundsEvan-locations',
      'FoxAndrea-accelerations',
      'FoxAndrea-locations',
      'IsmailJason-accelerations',
      'IsmailJason-locations',
      'LoydSam-accelerations',
      'LoydSam-locations',
      'MalenchikAli-locations',
      'MedeirosCorinne-accelerations',
      'MedeirosCorinne-locations',
      'NagarajVinay-locations',
      'OkeOsmond-accelerations',
      'OkeOsmond-locations',
      'OmprakashSantosh-accelerations',
      'OmprakashSantosh-locations',
      'OpitzNeil-accelerations',
      'OpitzNeil-df_to_json',
      'OpitzNeil-locations',
      'OpitzNeil-locations_data_json',
      'OpitzNeil-locdata',
      'PayneBarbara-accelerations',
```

```

'PayneBarbara-locations',
'RoyAvishek-accelerations',
'RoyAvishek-locations',
'SahaDisha-accelerations',
'SahaDisha-locations',
'SamLoyd-simple',
'SearsSamuel-accelerations',
'SearsSamuel-locations',
'SuryaBhushan-accelerations',
'SuryaBhushan-locations',
'test_this',
'testthis'}

```

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]: {'NagarajVinay-accelerations', 'NagarajVinay-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
            )
            print(msg_metadata)
    except KeyboardInterrupt:
        pass

```

```

    )

    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: NagarajVinay-locations
  Partition: 0
  Offset: 0
Message Key: 24d4ed9681a5473e9e936f9e3822b325
Message Value:
{
  "key1": "value1",
  "key2": "value2"
}

```

```

Message metadata:
  Topic: NagarajVinay-locations
  Partition: 0
  Offset: 1
Message Key: 243a2faff14c446da47d5ed2a164f817
Message Value:
{
  "key1": "value1",
  "key2": "value2"
}

```

```

Message metadata:
  Topic: NagarajVinay-locations

```

```
Partition: 0
Offset: 0
Message Key: 24d4ed9681a5473e9e936f9e3822b325
Message Value:
{
  "key1": "value1",
  "key2": "value2"
}
```

```
Message metadata:
Topic: NagarajVinay-locations
Partition: 0
Offset: 1
Message Key: 243a2faff14c446da47d5ed2a164f817
Message Value:
{
  "key1": "value1",
  "key2": "value2"
}
```

STOPPING MESSAGE CONSUMER

Close the consumer, waiting indefinitely for any needed cleanup.

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

```
[ ]:
```