Assignment 9.1

In [25]:
```
import os
import sys
import shutil
import json
from pathlib import Path

import pandas as pd

from kafka import KafkaProducer, KafkaAdminClient
from kafka.admin.new_topic import NewTopic
from kafka.errors import TopicAlreadyExistsError

from pyspark.sql import SparkSession
from pyspark.streaming import StreamingContext
from pyspark import SparkConf
from pyspark.sql.functions import window, from_json, col
from pyspark.sql.types import StringType, TimestampType, DoubleType, StructField, Struct
from pyspark.sql.functions import udf

# Add in spark and driver
os.environ['PYSPARK_PYTHON'] = sys.executable
os.environ['PYSPARK_DRIVER_PYTHON'] = sys.executable

current_dir = Path(os.getcwd()).absolute()
checkpoint_dir = current_dir.joinpath('checkpoints')
locations_checkpoint_dir = checkpoint_dir.joinpath('locations')
accelerations_checkpoint_dir = checkpoint_dir.joinpath('accelerations')

if locations_checkpoint_dir.exists():
    shutil.rmtree(locations_checkpoint_dir)

if accelerations_checkpoint_dir.exists():
    shutil.rmtree(accelerations_checkpoint_dir)

locations_checkpoint_dir.mkdir(parents=True, exist_ok=True)
accelerations_checkpoint_dir.mkdir(parents=True, exist_ok=True)
```

Configuration Parameters

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

In [26]:
```
config = dict(
    bootstrap_servers=['kafka.kafka.svc.cluster.local:9092'],
    first_name='Josh',
    last_name='Greenert'
)

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

config['locations_topic'] = '{}-locations'.format(config['topic_prefix'])
config['accelerations_topic'] = '{}-accelerations'.format(config['topic_prefix'])
```

```
config['simple_topic'] = '{}-simple'.format(config['topic_prefix'])

config
```

Out[26]:
```
{'bootstrap_servers': ['kafka.kafka.svc.cluster.local:9092'],
 'first_name': 'Josh',
 'last_name': 'Greenert',
 'client_id': 'GreenertJosh',
 'topic_prefix': 'GreenertJosh',
 'locations_topic': 'GreenertJosh-locations',
 'accelerations_topic': 'GreenertJosh-accelerations',
 'simple_topic': 'GreenertJosh-simple'}
```

Create Topic Utility Function

The `create_kafka_topic` helps create a Kafka topic based on your configuration settings. For instance, if your first name is *John* and your last name is *Doe*, `create_kafka_topic('locations')` will create a topic with the name `DoeJohn-locations`. The function will not create the topic if it already exists.

In [27]:
```
def create_kafka_topic(topic_name, config=config, num_partitions=1, replication_factor=1
    bootstrap_servers = config['bootstrap_servers']
    client_id = config['client_id']
    topic_prefix = config['topic_prefix']
    name = '{}-{}'.format(topic_prefix, topic_name)

    admin_client = KafkaAdminClient(
        bootstrap_servers=bootstrap_servers,
        client_id=client_id
    )

    topic = NewTopic(
        name=name,
        num_partitions=num_partitions,
        replication_factor=replication_factor
    )

    topic_list = [topic]
    try:
        admin_client.create_topics(new_topics=topic_list)
        print('Created topic "{}"'.format(name))
    except TopicAlreadyExistsError as e:
        print('Topic "{}" already exists'.format(name))

create_kafka_topic('simple')
```

```
Topic "GreenertJosh-simple" already exists
```

In [28]:
```
spark = SparkSession\
    .builder\
    .appName("Assignment09")\
    .getOrCreate()

df_locations = spark \
  .readStream \
  .format("kafka") \
  .option("kafka.bootstrap.servers", "kafka.kafka.svc.cluster.local:9092") \
  .option("subscribe", config['locations_topic']) \
  .load()
```

```
23/05/13 23:53:25 WARN SparkSession: Using an existing Spark session; only runtime SQL c
onfigurations will take effect.
```

TODO: Create a data frame called `df_accelerations` that reads from the accelerations topic you

published to in assignment 8. In order to read data from this topic, make sure that you are running the notebook you created in assignment 8 that publishes acceleration and location data to the `LastnameFirstname-simple` topic.

In [36]:
```
df_accelerations = spark.readStream.format("kafka").option("kafka.bootstrap.servers", "k
    .option("subscribe", config["accelerations_topic"]).load()
```

TODO: Create two streaming queries, `ds_locations` and `ds_accelerations` that publish to the `LastnameFirstname-simple` topic. See http://spark.apache.org/docs/latest/structured-streaming-programming-guide.html#starting-streaming-queries and http://spark.apache.org/docs/latest/structured-streaming-kafka-integration.html for more information.

In [43]:
```
ds_locations = df_locations \
  .writeStream \
  .format("kafka") \
  .option("kafka.bootstrap.servers", "kafka.kafka.svc.cluster.local:9092") \
  .option("topic", config['simple_topic']) \
  .option("checkpointLocation", str(locations_checkpoint_dir)) \
  .start()

ds_accelerations = df_accelerations \
  .writeStream \
  .format("kafka") \
  .option("kafka.bootstrap.servers", "kafka.kafka.svc.cluster.local:9092") \
  .option("topic", config['simple_topic']) \
  .option("checkpointLocation", str(accelerations_checkpoint_dir)) \
  .start()

try:
    ds_locations.awaitTermination()
    ds_accelerations.awaitTermination()
except KeyboardInterrupt:
    print("STOPPING STREAMING DATA")
```

```
23/05/14 01:58:58 WARN ResolveWriteToStream: spark.sql.adaptive.enabled is not supported
in streaming DataFrames/Datasets and will be disabled.
23/05/14 01:58:58 WARN ResolveWriteToStream: spark.sql.adaptive.enabled is not supported
in streaming DataFrames/Datasets and will be disabled.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'key.deserializer' was suppl
ied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'value.deserializer' was sup
plied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'enable.auto.commit' was sup
plied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'max.poll.records' was suppl
ied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'auto.offset.reset' was supp
lied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'key.deserializer' was suppl
ied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'value.deserializer' was sup
plied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'enable.auto.commit' was sup
plied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'max.poll.records' was suppl
ied but isn't a known config.
23/05/14 01:58:58 WARN AdminClientConfig: The configuration 'auto.offset.reset' was supp
lied but isn't a known config.
23/05/14 01:58:58 ERROR MicroBatchExecution: Query [id = 8f1f0776-a10c-4f43-8717-937494e
f4f1f, runId = 89eae48f-11a7-44ea-bab7-2431875e7281] terminated with error
java.lang.NoClassDefFoundError: org/apache/kafka/clients/admin/OffsetSpec
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$fetchLatestOffs
```

```
ets$2(KafkaOffsetReaderAdmin.scala:298)
        at scala.collection.TraversableLike.$anonfun$map$1(TraversableLike.scala:286)
        at scala.collection.Iterator.foreach(Iterator.scala:943)
        at scala.collection.Iterator.foreach$(Iterator.scala:943)
        at scala.collection.AbstractIterator.foreach(Iterator.scala:1431)
        at scala.collection.IterableLike.foreach(IterableLike.scala:74)
        at scala.collection.IterableLike.foreach$(IterableLike.scala:73)
        at scala.collection.AbstractIterable.foreach(Iterable.scala:56)
        at scala.collection.TraversableLike.map(TraversableLike.scala:286)
        at scala.collection.TraversableLike.map$(TraversableLike.scala:279)
        at scala.collection.mutable.AbstractSet.scala$collection$SetLike$$super$map(Set.
scala:50)
        at scala.collection.SetLike.map(SetLike.scala:105)
        at scala.collection.SetLike.map$(SetLike.scala:105)
        at scala.collection.mutable.AbstractSet.map(Set.scala:50)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$fetchLatestOffs
ets$1(KafkaOffsetReaderAdmin.scala:298)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$partitionsAssig
nedToAdmin$1(KafkaOffsetReaderAdmin.scala:501)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.withRetries(KafkaOffsetR
eaderAdmin.scala:518)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.partitionsAssignedToAdmi
n(KafkaOffsetReaderAdmin.scala:498)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.fetchLatestOffsets(Kafka
OffsetReaderAdmin.scala:297)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.$anonfun$getOrCreateIniti
alPartitionOffsets$1(KafkaMicroBatchStream.scala:251)
        at scala.Option.getOrElse(Option.scala:189)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.getOrCreateInitialPartiti
onOffsets(KafkaMicroBatchStream.scala:246)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.initialOffset(KafkaMicroB
atchStream.scala:98)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$getStar
tOffset$2(MicroBatchExecution.scala:455)
        at scala.Option.getOrElse(Option.scala:189)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.getStartOffset(M
icroBatchExecution.scala:455)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$4(MicroBatchExecution.scala:489)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken(Pro
gressReporter.scala:411)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken$(Pr
ogressReporter.scala:409)
        at org.apache.spark.sql.execution.streaming.StreamExecution.reportTimeTaken(Stre
amExecution.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$2(MicroBatchExecution.scala:488)
        at scala.collection.TraversableLike.$anonfun$map$1(TraversableLike.scala:286)
        at scala.collection.Iterator.foreach(Iterator.scala:943)
        at scala.collection.Iterator.foreach$(Iterator.scala:943)
        at scala.collection.AbstractIterator.foreach(Iterator.scala:1431)
        at scala.collection.IterableLike.foreach(IterableLike.scala:74)
        at scala.collection.IterableLike.foreach$(IterableLike.scala:73)
        at scala.collection.AbstractIterable.foreach(Iterable.scala:56)
        at scala.collection.TraversableLike.map(TraversableLike.scala:286)
        at scala.collection.TraversableLike.map$(TraversableLike.scala:279)
        at scala.collection.AbstractTraversable.map(Traversable.scala:108)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$1(MicroBatchExecution.scala:477)
        at scala.runtime.java8.JFunction0$mcZ$sp.apply(JFunction0$mcZ$sp.java:23)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.withProgressLock
ed(MicroBatchExecution.scala:802)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.constructNextBat
ch(MicroBatchExecution.scala:473)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$runActi
vatedStream$2(MicroBatchExecution.scala:266)
```

```
        at scala.runtime.java8.JFunction0$mcV$sp.apply(JFunction0$mcV$sp.java:23)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken(Pro
gressReporter.scala:411)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken$(Pr
ogressReporter.scala:409)
        at org.apache.spark.sql.execution.streaming.StreamExecution.reportTimeTaken(Stre
amExecution.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$runActi
vatedStream$1(MicroBatchExecution.scala:247)
        at org.apache.spark.sql.execution.streaming.ProcessingTimeExecutor.execute(Trigg
erExecutor.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.runActivatedStre
am(MicroBatchExecution.scala:237)
        at org.apache.spark.sql.execution.streaming.StreamExecution.$anonfun$runStream$1
(StreamExecution.scala:306)
        at scala.runtime.java8.JFunction0$mcV$sp.apply(JFunction0$mcV$sp.java:23)
        at org.apache.spark.sql.SparkSession.withActive(SparkSession.scala:827)
        at org.apache.spark.sql.execution.streaming.StreamExecution.org$apache$spark$sql
$execution$streaming$StreamExecution$$runStream(StreamExecution.scala:284)
        at org.apache.spark.sql.execution.streaming.StreamExecution$$anon$1.run(StreamEx
ecution.scala:207)
Caused by: java.lang.ClassNotFoundException: org.apache.kafka.clients.admin.OffsetSpec
        ... 58 more
23/05/14 01:58:58 ERROR MicroBatchExecution: Query [id = 74122ee7-e384-4244-93ee-c31c623
994e0, runId = 6599b29e-8564-46a4-9517-5f6b765b0ec3] terminated with error
java.lang.NoClassDefFoundError: org/apache/kafka/clients/admin/OffsetSpec
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$fetchLatestOffs
ets$2(KafkaOffsetReaderAdmin.scala:298)
        at scala.collection.TraversableLike.$anonfun$map$1(TraversableLike.scala:286)
        at scala.collection.Iterator.foreach(Iterator.scala:943)
        at scala.collection.Iterator.foreach$(Iterator.scala:943)
        at scala.collection.AbstractIterator.foreach(Iterator.scala:1431)
        at scala.collection.IterableLike.foreach(IterableLike.scala:74)
        at scala.collection.IterableLike.foreach$(IterableLike.scala:73)
        at scala.collection.AbstractIterable.foreach(Iterable.scala:56)
        at scala.collection.TraversableLike.map(TraversableLike.scala:286)
        at scala.collection.TraversableLike.map$(TraversableLike.scala:279)
        at scala.collection.mutable.AbstractSet.scala$collection$SetLike$$super$map(Set.
scala:50)
        at scala.collection.SetLike.map(SetLike.scala:105)
        at scala.collection.SetLike.map$(SetLike.scala:105)
        at scala.collection.mutable.AbstractSet.map(Set.scala:50)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$fetchLatestOffs
ets$1(KafkaOffsetReaderAdmin.scala:298)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$partitionsAssig
nedToAdmin$1(KafkaOffsetReaderAdmin.scala:501)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.withRetries(KafkaOffsetR
eaderAdmin.scala:518)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.partitionsAssignedToAdmi
n(KafkaOffsetReaderAdmin.scala:498)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.fetchLatestOffsets(Kafka
OffsetReaderAdmin.scala:297)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.$anonfun$getOrCreateIniti
alPartitionOffsets$1(KafkaMicroBatchStream.scala:251)
        at scala.Option.getOrElse(Option.scala:189)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.getOrCreateInitialPartiti
onOffsets(KafkaMicroBatchStream.scala:246)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.initialOffset(KafkaMicroB
atchStream.scala:98)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$getStar
tOffset$2(MicroBatchExecution.scala:455)
        at scala.Option.getOrElse(Option.scala:189)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.getStartOffset(M
icroBatchExecution.scala:455)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$4(MicroBatchExecution.scala:489)
```

```
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken(Pro
gressReporter.scala:411)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken$(Pr
ogressReporter.scala:409)
        at org.apache.spark.sql.execution.streaming.StreamExecution.reportTimeTaken(Stre
amExecution.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$2(MicroBatchExecution.scala:488)
        at scala.collection.TraversableLike.$anonfun$map$1(TraversableLike.scala:286)
        at scala.collection.Iterator.foreach(Iterator.scala:943)
        at scala.collection.Iterator.foreach$(Iterator.scala:943)
        at scala.collection.AbstractIterator.foreach(Iterator.scala:1431)
        at scala.collection.IterableLike.foreach(IterableLike.scala:74)
        at scala.collection.IterableLike.foreach$(IterableLike.scala:73)
        at scala.collection.AbstractIterable.foreach(Iterable.scala:56)
        at scala.collection.TraversableLike.map(TraversableLike.scala:286)
        at scala.collection.TraversableLike.map$(TraversableLike.scala:279)
        at scala.collection.AbstractTraversable.map(Traversable.scala:108)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$1(MicroBatchExecution.scala:477)
        at scala.runtime.java8.JFunction0$mcZ$sp.apply(JFunction0$mcZ$sp.java:23)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.withProgressLock
ed(MicroBatchExecution.scala:802)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.constructNextBat
ch(MicroBatchExecution.scala:473)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$runActi
vatedStream$2(MicroBatchExecution.scala:266)
        at scala.runtime.java8.JFunction0$mcV$sp.apply(JFunction0$mcV$sp.java:23)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken(Pro
gressReporter.scala:411)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken$(Pr
ogressReporter.scala:409)
        at org.apache.spark.sql.execution.streaming.StreamExecution.reportTimeTaken(Stre
amExecution.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$runActi
vatedStream$1(MicroBatchExecution.scala:247)
        at org.apache.spark.sql.execution.streaming.ProcessingTimeExecutor.execute(Trigg
erExecutor.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.runActivatedStre
am(MicroBatchExecution.scala:237)
        at org.apache.spark.sql.execution.streaming.StreamExecution.$anonfun$runStream$1
(StreamExecution.scala:306)
        at scala.runtime.java8.JFunction0$mcV$sp.apply(JFunction0$mcV$sp.java:23)
        at org.apache.spark.sql.SparkSession.withActive(SparkSession.scala:827)
        at org.apache.spark.sql.execution.streaming.StreamExecution.org$apache$spark$sql
$execution$streaming$StreamExecution$$runStream(StreamExecution.scala:284)
        at org.apache.spark.sql.execution.streaming.StreamExecution$$anon$1.run(StreamEx
ecution.scala:207)
Caused by: java.lang.ClassNotFoundException: org.apache.kafka.clients.admin.OffsetSpec
        ... 58 more
Exception in thread "stream execution thread for [id = 8f1f0776-a10c-4f43-8717-937494ef4
f1f, runId = 89eae48f-11a7-44ea-bab7-2431875e7281]" java.lang.NoClassDefFoundError: org/
apache/kafka/clients/admin/OffsetSpec
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$fetchLatestOffs
ets$2(KafkaOffsetReaderAdmin.scala:298)
        at scala.collection.TraversableLike.$anonfun$map$1(TraversableLike.scala:286)
        at scala.collection.Iterator.foreach(Iterator.scala:943)
        at scala.collection.Iterator.foreach$(Iterator.scala:943)
        at scala.collection.AbstractIterator.foreach(Iterator.scala:1431)
        at scala.collection.IterableLike.foreach(IterableLike.scala:74)
        at scala.collection.IterableLike.foreach$(IterableLike.scala:73)
        at scala.collection.AbstractIterable.foreach(Iterable.scala:56)
        at scala.collection.TraversableLike.map(TraversableLike.scala:286)
        at scala.collection.TraversableLike.map$(TraversableLike.scala:279)
        at scala.collection.mutable.AbstractSet.scala$collection$SetLike$$super$map(Set.
scala:50)
```

```
        at scala.collection.SetLike.map(SetLike.scala:105)
        at scala.collection.SetLike.map$(SetLike.scala:105)
        at scala.collection.mutable.AbstractSet.map(Set.scala:50)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$fetchLatestOffs
ets$1(KafkaOffsetReaderAdmin.scala:298)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$partitionsAssig
nedToAdmin$1(KafkaOffsetReaderAdmin.scala:501)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.withRetries(KafkaOffsetR
eaderAdmin.scala:518)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.partitionsAssignedToAdmi
n(KafkaOffsetReaderAdmin.scala:498)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.fetchLatestOffsets(Kafka
OffsetReaderAdmin.scala:297)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.$anonfun$getOrCreateIniti
alPartitionOffsets$1(KafkaMicroBatchStream.scala:251)
        at scala.Option.getOrElse(Option.scala:189)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.getOrCreateInitialPartiti
onOffsets(KafkaMicroBatchStream.scala:246)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.initialOffset(KafkaMicroB
atchStream.scala:98)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$getStar
tOffset$2(MicroBatchExecution.scala:455)
        at scala.Option.getOrElse(Option.scala:189)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.getStartOffset(M
icroBatchExecution.scala:455)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$4(MicroBatchExecution.scala:489)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken(Pro
gressReporter.scala:411)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken$(Pr
ogressReporter.scala:409)
        at org.apache.spark.sql.execution.streaming.StreamExecution.reportTimeTaken(Stre
amExecution.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$2(MicroBatchExecution.scala:488)
        at scala.collection.TraversableLike.$anonfun$map$1(TraversableLike.scala:286)
        at scala.collection.Iterator.foreach(Iterator.scala:943)
        at scala.collection.Iterator.foreach$(Iterator.scala:943)
        at scala.collection.AbstractIterator.foreach(Iterator.scala:1431)
        at scala.collection.IterableLike.foreach(IterableLike.scala:74)
        at scala.collection.IterableLike.foreach$(IterableLike.scala:73)
        at scala.collection.AbstractIterable.foreach(Iterable.scala:56)
        at scala.collection.TraversableLike.map(TraversableLike.scala:286)
        at scala.collection.TraversableLike.map$(TraversableLike.scala:279)
        at scala.collection.AbstractTraversable.map(Traversable.scala:108)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$1(MicroBatchExecution.scala:477)
        at scala.runtime.java8.JFunction0$mcZ$sp.apply(JFunction0$mcZ$sp.java:23)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.withProgressLock
ed(MicroBatchExecution.scala:802)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.constructNextBat
ch(MicroBatchExecution.scala:473)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$runActi
vatedStream$2(MicroBatchExecution.scala:266)
        at scala.runtime.java8.JFunction0$mcV$sp.apply(JFunction0$mcV$sp.java:23)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken(Pro
gressReporter.scala:411)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken$(Pr
ogressReporter.scala:409)
        at org.apache.spark.sql.execution.streaming.StreamExecution.reportTimeTaken(Stre
amExecution.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$runActi
vatedStream$1(MicroBatchExecution.scala:247)
        at org.apache.spark.sql.execution.streaming.ProcessingTimeExecutor.execute(Trigg
erExecutor.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.runActivatedStre
```

```
am(MicroBatchExecution.scala:237)
        at org.apache.spark.sql.execution.streaming.StreamExecution.$anonfun$runStream$1
(StreamExecution.scala:306)
        at scala.runtime.java8.JFunction0$mcV$sp.apply(JFunction0$mcV$sp.java:23)
        at org.apache.spark.sql.SparkSession.withActive(SparkSession.scala:827)
        at org.apache.spark.sql.execution.streaming.StreamExecution.org$apache$spark$sql
$execution$streaming$StreamExecution$$runStream(StreamExecution.scala:284)
        at org.apache.spark.sql.execution.streaming.StreamExecution$$anon$1.run(StreamEx
ecution.scala:207)
Caused by: java.lang.ClassNotFoundException: org.apache.kafka.clients.admin.OffsetSpec
        ... 58 more
Exception in thread "stream execution thread for [id = 74122ee7-e384-4244-93ee-c31c62399
4e0, runId = 6599b29e-8564-46a4-9517-5f6b765b0ec3]" java.lang.NoClassDefFoundError: org/
apache/kafka/clients/admin/OffsetSpec
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$fetchLatestOffs
ets$2(KafkaOffsetReaderAdmin.scala:298)
        at scala.collection.TraversableLike.$anonfun$map$1(TraversableLike.scala:286)
        at scala.collection.Iterator.foreach(Iterator.scala:943)
        at scala.collection.Iterator.foreach$(Iterator.scala:943)
        at scala.collection.AbstractIterator.foreach(Iterator.scala:1431)
        at scala.collection.IterableLike.foreach(IterableLike.scala:74)
        at scala.collection.IterableLike.foreach$(IterableLike.scala:73)
        at scala.collection.AbstractIterable.foreach(Iterable.scala:56)
        at scala.collection.TraversableLike.map(TraversableLike.scala:286)
        at scala.collection.TraversableLike.map$(TraversableLike.scala:279)
        at scala.collection.mutable.AbstractSet.scala$collection$SetLike$$super$map(Set.
scala:50)
        at scala.collection.SetLike.map(SetLike.scala:105)
        at scala.collection.SetLike.map$(SetLike.scala:105)
        at scala.collection.mutable.AbstractSet.map(Set.scala:50)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$fetchLatestOffs
ets$1(KafkaOffsetReaderAdmin.scala:298)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.$anonfun$partitionsAssig
nedToAdmin$1(KafkaOffsetReaderAdmin.scala:501)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.withRetries(KafkaOffsetR
eaderAdmin.scala:518)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.partitionsAssignedToAdmi
n(KafkaOffsetReaderAdmin.scala:498)
        at org.apache.spark.sql.kafka010.KafkaOffsetReaderAdmin.fetchLatestOffsets(Kafka
OffsetReaderAdmin.scala:297)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.$anonfun$getOrCreateIniti
alPartitionOffsets$1(KafkaMicroBatchStream.scala:251)
        at scala.Option.getOrElse(Option.scala:189)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.getOrCreateInitialPartiti
onOffsets(KafkaMicroBatchStream.scala:246)
        at org.apache.spark.sql.kafka010.KafkaMicroBatchStream.initialOffset(KafkaMicroB
atchStream.scala:98)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$getStar
tOffset$2(MicroBatchExecution.scala:455)
        at scala.Option.getOrElse(Option.scala:189)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.getStartOffset(M
icroBatchExecution.scala:455)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$4(MicroBatchExecution.scala:489)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken(Pro
gressReporter.scala:411)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken$(Pr
ogressReporter.scala:409)
        at org.apache.spark.sql.execution.streaming.StreamExecution.reportTimeTaken(Stre
amExecution.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$constru
ctNextBatch$2(MicroBatchExecution.scala:488)
        at scala.collection.TraversableLike.$anonfun$map$1(TraversableLike.scala:286)
        at scala.collection.Iterator.foreach(Iterator.scala:943)
        at scala.collection.Iterator.foreach$(Iterator.scala:943)
        at scala.collection.AbstractIterator.foreach(Iterator.scala:1431)
```

```
        at scala.collection.IterableLike.foreach(IterableLike.scala:74)
        at scala.collection.IterableLike.foreach$(IterableLike.scala:73)
        at scala.collection.AbstractIterable.foreach(Iterable.scala:56)
        at scala.collection.TraversableLike.map(TraversableLike.scala:286)
        at scala.collection.TraversableLike.map$(TraversableLike.scala:279)
        at scala.collection.AbstractTraversable.map(Traversable.scala:108)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$construct
NextBatch$1(MicroBatchExecution.scala:477)
        at scala.runtime.java8.JFunction0$mcZ$sp.apply(JFunction0$mcZ$sp.java:23)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.withProgressLock
ed(MicroBatchExecution.scala:802)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.constructNextBat
ch(MicroBatchExecution.scala:473)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$runActi
vatedStream$2(MicroBatchExecution.scala:266)
        at scala.runtime.java8.JFunction0$mcV$sp.apply(JFunction0$mcV$sp.java:23)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken(Pro
gressReporter.scala:411)
        at org.apache.spark.sql.execution.streaming.ProgressReporter.reportTimeTaken$(Pr
ogressReporter.scala:409)
        at org.apache.spark.sql.execution.streaming.StreamExecution.reportTimeTaken(Stre
amExecution.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.$anonfun$runActi
vatedStream$1(MicroBatchExecution.scala:247)
        at org.apache.spark.sql.execution.streaming.ProcessingTimeExecutor.execute(Trigg
erExecutor.scala:67)
        at org.apache.spark.sql.execution.streaming.MicroBatchExecution.runActivatedStre
am(MicroBatchExecution.scala:237)
        at org.apache.spark.sql.execution.streaming.StreamExecution.$anonfun$runStream$1
(StreamExecution.scala:306)
        at scala.runtime.java8.JFunction0$mcV$sp.apply(JFunction0$mcV$sp.java:23)
        at org.apache.spark.sql.SparkSession.withActive(SparkSession.scala:827)
        at org.apache.spark.sql.execution.streaming.StreamExecution.org$apache$spark$sql
$execution$streaming$StreamExecution$$runStream(StreamExecution.scala:284)
        at org.apache.spark.sql.execution.streaming.StreamExecution$$anon$1.run(StreamEx
ecution.scala:207)
Caused by: java.lang.ClassNotFoundException: org.apache.kafka.clients.admin.OffsetSpec
        ... 58 more
---------------------------------------------------------------------------
StreamingQueryException                   Traceback (most recent call last)
Cell In[43], line 18
      9 ds_accelerations = df_accelerations \
     10   .writeStream \
     11   .format("kafka") \
   (...)
     14   .option("checkpointLocation", str(accelerations_checkpoint_dir)) \
     15   .start()
     17 try:
---> 18     ds_locations.awaitTermination()
     19     ds_accelerations.awaitTermination()
     20 except KeyboardInterrupt:

File /opt/conda/lib/python3.10/site-packages/pyspark/sql/streaming/query.py:201, in Stre
amingQuery.awaitTermination(self, timeout)
    199     return self._jsq.awaitTermination(int(timeout * 1000))
    200 else:
--> 201     return self._jsq.awaitTermination()

File /opt/conda/lib/python3.10/site-packages/py4j/java_gateway.py:1322, in JavaMember.__
call__(self, *args)
   1316 command = proto.CALL_COMMAND_NAME +\
   1317     self.command_header +\
   1318     args_command +\
   1319     proto.END_COMMAND_PART
   1321 answer = self.gateway_client.send_command(command)
```

```
-> 1322 return value = get return value(
   1323     answer, self.gateway_client, self.target_id, self.name)
   1325 for temp_arg in temp_args:
   1326     if hasattr(temp_arg, "_detach"):

File /opt/conda/lib/python3.10/site-packages/pyspark/errors/exceptions/captured.py:175,
 in capture_sql_exception.<locals>.deco(*a, **kw)
    171 converted = convert_exception(e.java_exception)
    172 if not isinstance(converted, UnknownException):
    173     # Hide where the exception came from that shows a non-Pythonic
    174     # JVM exception message.
--> 175     raise converted from None
    176 else:
    177     raise

StreamingQueryException: [STREAM_FAILED] Query [id = 8f1f0776-a10c-4f43-8717-937494ef4f1
f, runId = 89eae48f-11a7-44ea-bab7-2431875e7281] terminated with exception: org/apache/k
afka/clients/admin/OffsetSpec
```

In [ ]: