In [1]:
```
%%configure
{ "conf": {
            "spark.jars":"hdfs:///apps/hudi/lib/hudi-spark-bundle.ja
            "spark.serializer":"org.apache.spark.serializer.KryoSeri
            "spark.sql.hive.convertMetastoreParquet":"false"
          }}
```

Current session configs: `{'conf': {'spark.jars': 'hdfs:///apps/hudi/lib/hudi-spark-bundle.jar,hdfs:///apps/hudi/lib/spark-avro.jar', 'spark.serializer': 'org.apache.spark.serializer.KryoSerializer', 'spark.sql.hive.convertMetastoreParquet': 'false'}, 'proxyUser': 'assumed-role_AdminNik_khokharn-Isengard', 'kind': 'pyspark'}`

No active sessions.

In [2]:
```
from pyspark.sql.functions import *
from pyspark.sql.types import *
```

Starting Spark application

ID	YARN Application ID	Kind	State	Spark UI	Driver log	Current session?
1	application_1628248144302_0003	pyspark	idle	Link	Link	✓

SparkSession available as 'spark'.

In [2]:
```
tableName = "hudi_cow_table_mobikwik"
tablePath = "s3://khokharn-hudi/MobiKwik/" + tableName
```

Starting Spark application

ID	YARN Application ID	Kind	State	Spark UI	Driver log	Current session?
3	application_1628248144302_0005	pyspark	idle	Link	Link	✓

SparkSession available as 'spark'.

In [27]:

```
hudiWriteConfig = {
    'className' : 'org.apache.hudi',
    'hoodie.table.name': tableName,
    'hoodie.datasource.write.operation': 'upsert',
    'hoodie.datasource.write.table.type': 'COPY_ON_WRITE',
    'hoodie.datasource.write.precombine.field': 'date',
    'hoodie.datasource.write.recordkey.field': 'name',
    'hoodie.datasource.write.partitionpath.field': 'name:SIMPLE,year
    'hoodie.datasource.write.keygenerator.class': 'org.apache.hudi.k
    'hoodie.deltastreamer.keygen.timebased.timestamp.type': 'MIXED',
    'hoodie.deltastreamer.keygen.timebased.input.dateformat': 'yyyy-
    'hoodie.deltastreamer.keygen.timebased.output.dateformat':'yyyy/
}

hudiGlueConfig = {
    'hoodie.datasource.hive_sync.enable': 'true',
    'hoodie.datasource.hive_sync.database': 'default',
    'hoodie.datasource.hive_sync.table': tableName,
    'hoodie.datasource.write.hive_style_partitioning' : 'true',
    'hoodie.datasource.hive_sync.partition_extractor_class': 'org.ap
    'hoodie.datasource.hive_sync.partition_fields': 'name,year,month
}

#'hoodie.datasource.hive_sync.jdbcurl': 'jdbc:hive2://localhost:1000
#'hoodie.datasource.write.partitionpath.field': 'name:SIMPLE,dt:TIME

combinedConf = {
    **hudiWriteConfig,
    **hudiGlueConfig
}
```

In [28]:

```
# use for first run
simpleData = [
    ("Person1","2021-07-22","1234","White"),
    ("Person2","2021-07-22","1234","White"),
    ("Person3","2021-07-22","1234","White"),
    ("Person4","2021-07-22","1234","White")
]

columns = ["name","date","col_to_update_integer","col_to_update_stri
df = spark.createDataFrame(data = simpleData, schema = columns)


df1 = df.select("name","date","col_to_update_integer","col_to_update
df1.printSchema()
df1.show(truncate=False)


df1.write.format("hudi") \
.options(**combinedConf) \
.mode("append") \
.save(tablePath)
```

```
root
 |-- name: string (nullable = true)
```

```
 |-- date: string (nullable = true)
 |-- col_to_update_integer: string (nullable = true)
 |-- col_to_update_string: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- day: integer (nullable = true)

+-------+----------+--------------------+--------------------+----+
-----+---+
|name   |date      |col_to_update_integer|col_to_update_string|year|
month|day|
+-------+----------+--------------------+--------------------+----+
-----+---+
|Person1|2021-07-22|1234                 |White               |2021|
7     |22 |
|Person2|2021-07-22|1234                 |White               |2021|
7     |22 |
|Person3|2021-07-22|1234                 |White               |2021|
7     |22 |
|Person4|2021-07-22|1234                 |White               |2021|
7     |22 |
+-------+----------+--------------------+--------------------+----+
-----+---+
```

In [29]:

```
## Check via PySpark
hudiDF = spark.read \
.format("hudi") \
.load(tablePath).show(truncate=False)
```

```
+------------------+--------------------+------------------+-------
-------------------------------+----------------------------------------
----------------------------------+-------+----------+------------
---------+--------------------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
|name   |date      |col_to_update_integer|col_to_update_string|year|
month|day|
+------------------+--------------------+------------------+-------
-------------------------------+----------------------------------------
----------------------------------+-------+----------+------------
---------+--------------------+----+-----+---+
|20210808155456     |20210808155456_2_3  |Person2           |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-228-72348_20210808155456.parquet|Person2|2021-07-22|1234
|White               |2021|7    |22 |
|20210808155456     |20210808155456_1_3  |Person1           |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-228-72347_20210808155456.parquet|Person1|2021-07-22|1234
|White               |2021|7    |22 |
|20210808155456     |20210808155456_0_4  |Person3           |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-228-72346_20210808155456.parquet|Person3|2021-07-22|1234
|White               |2021|7    |22 |
|20210808155456     |20210808155456_3_4  |Person4           |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-228-72349_20210808155456.parquet|Person4|2021-07-22|1234
|White               |2021|7    |22 |
+------------------+--------------------+------------------+-------
-------------------------------+----------------------------------------
----------------------------------+-------+----------+------------
---------+--------------------+----+-----+---+
```

In [30]:
```
# use for second run for in-place update
simpleData = [
    ("Person1","2021-07-22","4567","Yellow"),
    ("Person2","2021-07-22","4567","Yellow"),
    ("Person3","2021-07-22","4567","Yellow"),
    ("Person4","2021-07-22","4567","Yellow")
]

columns = ["name","date","col_to_update_integer","col_to_update_stri
df = spark.createDataFrame(data = simpleData, schema = columns)


df1 = df.select("name","date","col_to_update_integer","col_to_update
df1.printSchema()
df1.show(truncate=False)


df1.write.format("hudi") \
.options(**combinedConf) \
.mode("append") \
.save(tablePath)
```

```
root
 |-- name: string (nullable = true)
 |-- date: string (nullable = true)
 |-- col_to_update_integer: string (nullable = true)
 |-- col_to_update_string: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- day: integer (nullable = true)

+-------+----------+---------------------+--------------------+----+
-----+---+
|name   |date      |col_to_update_integer|col_to_update_string|year|
month|day|
+-------+----------+---------------------+--------------------+----+
-----+---+
|Person1|2021-07-22|4567                 |Yellow              |2021|
7     |22 |
|Person2|2021-07-22|4567                 |Yellow              |2021|
7     |22 |
|Person3|2021-07-22|4567                 |Yellow              |2021|
7     |22 |
|Person4|2021-07-22|4567                 |Yellow              |2021|
7     |22 |
+-------+----------+---------------------+--------------------+----+
-----+---+
```

In [31]:
```
## Check via PySpark
hudiDF = spark.read \
.format("hudi") \
.load(tablePath).show(truncate=False)
```

```
+-------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-----------------------------------+-------+----------+------------
---------+--------------------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
```

```
|name    |date      |col_to_update_integer|col_to_update_string|year|
month|day|
+--------------------+--------------------+------------------+-------
--------------------------------+--------------------------------------
--------------------------------------+-------+----------+------------
---------+--------------------+----+-----+---+
|20210808155922      |20210808155922_0_5  |Person3             |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-265-85904_20210808155922.parquet|Person3|2021-07-22|4567
|Yellow              |2021|7    |22 |
|20210808155922      |20210808155922_2_6  |Person2             |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-265-85906_20210808155922.parquet|Person2|2021-07-22|4567
|Yellow              |2021|7    |22 |
|20210808155922      |20210808155922_1_6  |Person1             |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-265-85905_20210808155922.parquet|Person1|2021-07-22|4567
|Yellow              |2021|7    |22 |
|20210808155922      |20210808155922_3_5  |Person4             |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-265-85907_20210808155922.parquet|Person4|2021-07-22|4567
|Yellow              |2021|7    |22 |
+--------------------+--------------------+------------------+-------
--------------------------------+--------------------------------------
--------------------------------------+-------+----------+------------
---------+--------------------+----+-----+---+
```

In [33]:

```
# use for forward schema evolution
# new columns are accept all good
simpleData = [
    ("Person1","2021-07-22","8910","Silver","abc"),
    ("Person2","2021-07-22","8910","Silver","abc"),
    ("Person3","2021-07-22","8910","Silver","abc"),
    ("Person4","2021-07-22","8910","Silver","abc")
]

columns = ["name","date","col_to_update_integer","col_to_update_stri
df = spark.createDataFrame(data = simpleData, schema = columns)


df1 = df.select("name","date","col_to_update_integer","col_to_update
df1.printSchema()
df1.show(truncate=False)


df1.write.format("hudi") \
.options(**combinedConf) \
.mode("append") \
.save(tablePath)
```

```
root
 |-- name: string (nullable = true)
 |-- date: string (nullable = true)
 |-- col_to_update_integer: string (nullable = true)
 |-- col_to_update_string: string (nullable = true)
 |-- new_col: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- day: integer (nullable = true)

+-------+----------+---------------------+--------------------+-----
--+----+-----+---+
```

```
|name   |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+-------+----------+---------------------+--------------------+-----
--+----+-----+---+
|Person1|2021-07-22|8910                 |Silver              |abc
|2021|7    |22 |
|Person2|2021-07-22|8910                 |Silver              |abc
|2021|7    |22 |
|Person3|2021-07-22|8910                 |Silver              |abc
|2021|7    |22 |
|Person4|2021-07-22|8910                 |Silver              |abc
|2021|7    |22 |
+-------+----------+---------------------+--------------------+-----
```

In [34]:

```
## Check via PySpark
hudiDF = spark.read \
.format("hudi") \
.load(tablePath).show(truncate=False)
```

```
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-----------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
|name   |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-----------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
|20210808160422     |20210808160422_0_10 |Person3           |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-336-113015_20210808160422.parquet|Person3|2021-07-22|8910
|Silver              |abc    |2021|7    |22 |
|20210808160422     |20210808160422_3_10 |Person4           |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-336-113018_20210808160422.parquet|Person4|2021-07-22|8910
|Silver              |abc    |2021|7    |22 |
|20210808160422     |20210808160422_2_9  |Person2           |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-336-113017_20210808160422.parquet|Person2|2021-07-22|8910
|Silver              |abc    |2021|7    |22 |
|20210808160422     |20210808160422_1_9  |Person1           |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-336-113016_20210808160422.parquet|Person1|2021-07-22|8910
|Silver              |abc    |2021|7    |22 |
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-----------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
```

In [35]:

```
# use for backward schema evolution
# fails if newly added column is no longer found
# Caused by: org.apache.parquet.io.InvalidRecordException: Parquet/A
simpleData = [
    ("Person1","2021-07-22","11121314","Purple"),
    ("Person2","2021-07-22","11121314","Purple"),
    ("Person3","2021-07-22","11121314","Purple"),
    ("Person4","2021-07-22","11121314","Purple")
]

columns = ["name","date","col_to_update_integer","col_to_update_stri
df = spark.createDataFrame(data = simpleData, schema = columns)


df1 = df.select("name","date","col_to_update_integer","col_to_update
df1.printSchema()
df1.show(truncate=False)


df1.write.format("hudi") \
.options(**combinedConf) \
.mode("append") \
.save(tablePath)
```

```
An error was encountered:
An error occurred while calling o843.save.
: org.apache.spark.SparkException: Job aborted due to stage failure:
Task 0 in stage 373.0 failed 4 times, most recent failure: Lost task
0.3 in stage 373.0 (TID 126585) (ip-172-31-58-135.ap-south-1.comput
e.internal executor 10): org.apache.hudi.exception.HoodieUpsertExcep
tion: Error upserting bucketType UPDATE for partition :0
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpsertPartition(BaseSparkCommitActionExecutor.java:27
9)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.lambda$execute$ecf5068c$1(BaseSparkCommitActionExecutor.jav
a:135)
        at org.apache.spark.api.java.JavaRDDLike.$anonfun$mapPartiti
onsWithIndex$1(JavaRDDLike.scala:102)
        at org.apache.spark.api.java.JavaRDDLike.$anonfun$mapPartiti
onsWithIndex$1$adapted(JavaRDDLike.scala:102)
        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsWithInde
x$2(RDD.scala:915)
        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsWithInde
x$2$adapted(RDD.scala:915)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.$anonfun$getOrCompute$1(RDD.scal
a:386)
        at org.apache.spark.storage.BlockManager.$anonfun$doPutItera
tor$1(BlockManager.scala:1440)
        at org.apache.spark.storage.BlockManager.org$apache$spark$st
orage$BlockManager$$doPut(BlockManager.scala:1350)
        at org.apache.spark.storage.BlockManager.doPutIterator(Block
```

```
Manager.scala:1414)
        at org.apache.spark.storage.BlockManager.getOrElseUpdate(Blo
ckManager.scala:1237)
        at org.apache.spark.rdd.RDD.getOrCompute(RDD.scala:384)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:335)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.
scala:90)
        at org.apache.spark.scheduler.Task.run(Task.scala:131)
        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$ru
n$3(Executor.scala:497)
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.sca
la:1439)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executo
r.scala:500)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadP
oolExecutor.java:1149)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Thread
PoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:748)
Caused by: org.apache.hudi.exception.HoodieException: org.apache.hud
i.exception.HoodieException: java.util.concurrent.ExecutionExceptio
n: org.apache.hudi.exception.HoodieException: operation has failed
        at org.apache.hudi.table.action.commit.SparkMergeHelper.runM
erge(SparkMergeHelper.java:102)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpdateInternal(BaseSparkCommitActionExecutor.java:30
8)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpdate(BaseSparkCommitActionExecutor.java:299)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpsertPartition(BaseSparkCommitActionExecutor.java:27
2)
        ... 28 more
Caused by: org.apache.hudi.exception.HoodieException: java.util.conc
urrent.ExecutionException: org.apache.hudi.exception.HoodieExceptio
n: operation has failed
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.execute(BoundedInMemoryExecutor.java:143)
        at org.apache.hudi.table.action.commit.SparkMergeHelper.runM
erge(SparkMergeHelper.java:100)
        ... 31 more
Caused by: java.util.concurrent.ExecutionException: org.apache.hudi.
exception.HoodieException: operation has failed
        at java.util.concurrent.FutureTask.report(FutureTask.java:12
2)
        at java.util.concurrent.FutureTask.get(FutureTask.java:192)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.execute(BoundedInMemoryExecutor.java:141)
        ... 32 more
Caused by: org.apache.hudi.exception.HoodieException: operation has
failed
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.th
rowExceptionIfFailed(BoundedInMemoryQueue.java:247)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.re
adNextRecord(BoundedInMemoryQueue.java:226)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.ac
cess$100(BoundedInMemoryQueue.java:52)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue$Qu
eueIterator.hasNext(BoundedInMemoryQueue.java:277)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueueCon
sumer.consume(BoundedInMemoryQueueConsumer.java:36)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
```

```
r.lambda$null$2(BoundedInMemoryExecutor.java:121)
        at java.util.concurrent.FutureTask.run(FutureTask.java:266)
        ... 3 more
Caused by: org.apache.parquet.io.InvalidRecordException: Parquet/Avr
o schema mismatch: Avro field 'new_col' not found
        at org.apache.parquet.avro.AvroRecordConverter.getAvroField
(AvroRecordConverter.java:225)
        at org.apache.parquet.avro.AvroRecordConverter.<init>(AvroRe
cordConverter.java:130)
        at org.apache.parquet.avro.AvroRecordConverter.<init>(AvroRe
cordConverter.java:95)
        at org.apache.parquet.avro.AvroRecordMaterializer.<init>(Avr
oRecordMaterializer.java:33)
        at org.apache.parquet.avro.AvroReadSupport.prepareForRead(Av
roReadSupport.java:138)
        at org.apache.parquet.hadoop.InternalParquetRecordReader.ini
tialize(InternalParquetRecordReader.java:183)
        at org.apache.parquet.hadoop.ParquetReader.initReader(Parque
tReader.java:156)
        at org.apache.parquet.hadoop.ParquetReader.read(ParquetReade
r.java:135)
        at org.apache.hudi.common.util.ParquetReaderIterator.hasNext
(ParquetReaderIterator.java:49)
        at org.apache.hudi.common.util.queue.IteratorBasedQueueProdu
cer.produce(IteratorBasedQueueProducer.java:45)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.lambda$null$0(BoundedInMemoryExecutor.java:92)
        at java.util.concurrent.FutureTask.run(FutureTask.java:266)
        at java.util.concurrent.Executors$RunnableAdapter.call(Execu
tors.java:511)
        ... 4 more

Driver stacktrace:
        at org.apache.spark.scheduler.DAGScheduler.failJobAndIndepen
dentStages(DAGScheduler.scala:2465)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$abortSta
ge$2(DAGScheduler.scala:2414)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$abortSta
ge$2$adapted(DAGScheduler.scala:2413)
        at scala.collection.mutable.ResizableArray.foreach(Resizable
Array.scala:62)
        at scala.collection.mutable.ResizableArray.foreach$(Resizabl
eArray.scala:55)
        at scala.collection.mutable.ArrayBuffer.foreach(ArrayBuffer.
scala:49)
        at org.apache.spark.scheduler.DAGScheduler.abortStage(DAGSch
eduler.scala:2413)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$handleTa
skSetFailed$1(DAGScheduler.scala:1124)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$handleTa
skSetFailed$1$adapted(DAGScheduler.scala:1124)
        at scala.Option.foreach(Option.scala:407)
        at org.apache.spark.scheduler.DAGScheduler.handleTaskSetFail
ed(DAGScheduler.scala:1124)
        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.d
oOnReceive(DAGScheduler.scala:2679)
        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.o
nReceive(DAGScheduler.scala:2621)
        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.o
nReceive(DAGScheduler.scala:2610)
        at org.apache.spark.util.EventLoop$$anon$1.run(EventLoop.sca
la:49)
        at org.apache.spark.scheduler.DAGScheduler.runJob(DAGSchedul
er.scala:914)
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2
238)
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2
```

```
259)
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2
278)
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2
303)
        at org.apache.spark.rdd.RDD.count(RDD.scala:1253)
        at org.apache.hudi.HoodieSparkSqlWriter$.commitAndPerformPos
tOperations(HoodieSparkSqlWriter.scala:433)
        at org.apache.hudi.HoodieSparkSqlWriter$.write(HoodieSparkSq
lWriter.scala:218)
        at org.apache.hudi.DefaultSource.createRelation(DefaultSourc
e.scala:134)
        at org.apache.spark.sql.execution.datasources.SaveIntoDataSo
urceCommand.run(SaveIntoDataSourceCommand.scala:46)
        at org.apache.spark.sql.execution.command.ExecutedCommandExe
c.sideEffectResult$lzycompute(commands.scala:70)
        at org.apache.spark.sql.execution.command.ExecutedCommandExe
c.sideEffectResult(commands.scala:68)
        at org.apache.spark.sql.execution.command.ExecutedCommandExe
c.doExecute(commands.scala:90)
        at org.apache.spark.sql.execution.SparkPlan.$anonfun$execut
e$1(SparkPlan.scala:185)
        at org.apache.spark.sql.execution.SparkPlan.$anonfun$execute
Query$1(SparkPlan.scala:223)
        at org.apache.spark.rdd.RDDOperationScope$.withScope(RDDOper
ationScope.scala:151)
        at org.apache.spark.sql.execution.SparkPlan.executeQuery(Spa
rkPlan.scala:220)
        at org.apache.spark.sql.execution.SparkPlan.execute(SparkPla
n.scala:181)
        at org.apache.spark.sql.execution.QueryExecution.toRdd$lzyco
mpute(QueryExecution.scala:134)
        at org.apache.spark.sql.execution.QueryExecution.toRdd(Query
Execution.scala:133)
        at org.apache.spark.sql.DataFrameWriter.$anonfun$runComman
d$1(DataFrameWriter.scala:989)
        at org.apache.spark.sql.catalyst.QueryPlanningTracker$.withT
racker(QueryPlanningTracker.scala:107)
        at org.apache.spark.sql.execution.SQLExecution$.withTracker
(SQLExecution.scala:232)
        at org.apache.spark.sql.execution.SQLExecution$.executeQuer
y$1(SQLExecution.scala:110)
        at org.apache.spark.sql.execution.SQLExecution$.$anonfun$wit
hNewExecutionId$6(SQLExecution.scala:135)
        at org.apache.spark.sql.catalyst.QueryPlanningTracker$.withT
racker(QueryPlanningTracker.scala:107)
        at org.apache.spark.sql.execution.SQLExecution$.withTracker
(SQLExecution.scala:232)
        at org.apache.spark.sql.execution.SQLExecution$.$anonfun$wit
hNewExecutionId$5(SQLExecution.scala:135)
        at org.apache.spark.sql.execution.SQLExecution$.withSQLConfP
ropagated(SQLExecution.scala:253)
        at org.apache.spark.sql.execution.SQLExecution$.$anonfun$wit
hNewExecutionId$1(SQLExecution.scala:134)
        at org.apache.spark.sql.SparkSession.withActive(SparkSessio
n.scala:772)
        at org.apache.spark.sql.execution.SQLExecution$.withNewExecu
tionId(SQLExecution.scala:68)
        at org.apache.spark.sql.DataFrameWriter.runCommand(DataFrame
Writer.scala:989)
        at org.apache.spark.sql.DataFrameWriter.saveToV1Source(DataF
rameWriter.scala:438)
        at org.apache.spark.sql.DataFrameWriter.saveInternal(DataFra
meWriter.scala:415)
        at org.apache.spark.sql.DataFrameWriter.save(DataFrameWrite
r.scala:293)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Metho
```

```
d)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodA
ccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(Delegatin
gMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:2
44)
        at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.
java:357)
        at py4j.Gateway.invoke(Gateway.java:282)
        at py4j.commands.AbstractCommand.invokeMethod(AbstractComman
d.java:132)
        at py4j.commands.CallCommand.execute(CallCommand.java:79)
        at py4j.GatewayConnection.run(GatewayConnection.java:238)
        at java.lang.Thread.run(Thread.java:748)
Caused by: org.apache.hudi.exception.HoodieUpsertException: Error up
serting bucketType UPDATE for partition :0
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpsertPartition(BaseSparkCommitActionExecutor.java:27
9)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.lambda$execute$ecf5068c$1(BaseSparkCommitActionExecutor.jav
a:135)
        at org.apache.spark.api.java.JavaRDDLike.$anonfun$mapPartiti
onsWithIndex$1(JavaRDDLike.scala:102)
        at org.apache.spark.api.java.JavaRDDLike.$anonfun$mapPartiti
onsWithIndex$1$adapted(JavaRDDLike.scala:102)
        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsWithInde
x$2(RDD.scala:915)
        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsWithInde
x$2$adapted(RDD.scala:915)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.$anonfun$getOrCompute$1(RDD.scal
a:386)
        at org.apache.spark.storage.BlockManager.$anonfun$doPutItera
tor$1(BlockManager.scala:1440)
        at org.apache.spark.storage.BlockManager.org$apache$spark$st
orage$BlockManager$$doPut(BlockManager.scala:1350)
        at org.apache.spark.storage.BlockManager.doPutIterator(Block
Manager.scala:1414)
        at org.apache.spark.storage.BlockManager.getOrElseUpdate(Blo
ckManager.scala:1237)
        at org.apache.spark.rdd.RDD.getOrCompute(RDD.scala:384)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:335)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.
scala:90)
        at org.apache.spark.scheduler.Task.run(Task.scala:131)
        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$ru
n$3(Executor.scala:497)
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.sca
la:1439)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executo
r.scala:500)
```

```
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadP
oolExecutor.java:1149)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Thread
PoolExecutor.java:624)
        ... 1 more
Caused by: org.apache.hudi.exception.HoodieException: org.apache.hud
i.exception.HoodieException: java.util.concurrent.ExecutionExceptio
n: org.apache.hudi.exception.HoodieException: operation has failed
        at org.apache.hudi.table.action.commit.SparkMergeHelper.runM
erge(SparkMergeHelper.java:102)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpdateInternal(BaseSparkCommitActionExecutor.java:30
8)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpdate(BaseSparkCommitActionExecutor.java:299)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpsertPartition(BaseSparkCommitActionExecutor.java:27
2)
        ... 28 more
Caused by: org.apache.hudi.exception.HoodieException: java.util.conc
urrent.ExecutionException: org.apache.hudi.exception.HoodieExceptio
n: operation has failed
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.execute(BoundedInMemoryExecutor.java:143)
        at org.apache.hudi.table.action.commit.SparkMergeHelper.runM
erge(SparkMergeHelper.java:100)
        ... 31 more
Caused by: java.util.concurrent.ExecutionException: org.apache.hudi.
exception.HoodieException: operation has failed
        at java.util.concurrent.FutureTask.report(FutureTask.java:12
2)
        at java.util.concurrent.FutureTask.get(FutureTask.java:192)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.execute(BoundedInMemoryExecutor.java:141)
        ... 32 more
Caused by: org.apache.hudi.exception.HoodieException: operation has
failed
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.th
rowExceptionIfFailed(BoundedInMemoryQueue.java:247)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.re
adNextRecord(BoundedInMemoryQueue.java:226)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.ac
cess$100(BoundedInMemoryQueue.java:52)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue$Qu
eueIterator.hasNext(BoundedInMemoryQueue.java:277)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueueCon
sumer.consume(BoundedInMemoryQueueConsumer.java:36)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.lambda$null$2(BoundedInMemoryExecutor.java:121)
        at java.util.concurrent.FutureTask.run(FutureTask.java:266)
        ... 3 more
Caused by: org.apache.parquet.io.InvalidRecordException: Parquet/Avr
o schema mismatch: Avro field 'new_col' not found
        at org.apache.parquet.avro.AvroRecordConverter.getAvroField
(AvroRecordConverter.java:225)
        at org.apache.parquet.avro.AvroRecordConverter.<init>(AvroRe
cordConverter.java:130)
        at org.apache.parquet.avro.AvroRecordConverter.<init>(AvroRe
cordConverter.java:95)
        at org.apache.parquet.avro.AvroRecordMaterializer.<init>(Avr
oRecordMaterializer.java:33)
        at org.apache.parquet.avro.AvroReadSupport.prepareForRead(Av
roReadSupport.java:138)
        at org.apache.parquet.hadoop.InternalParquetRecordReader.ini
tialize(InternalParquetRecordReader.java:183)
        at org.apache.parquet.hadoop.ParquetReader.initReader(Parque
tReader.java:156)
```

```
        at org.apache.parquet.hadoop.ParquetReader.read(ParquetReade
r.java:135)
        at org.apache.hudi.common.util.ParquetReaderIterator.hasNext
(ParquetReaderIterator.java:49)
        at org.apache.hudi.common.util.queue.IteratorBasedQueueProdu
cer.produce(IteratorBasedQueueProducer.java:45)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.lambda$null$0(BoundedInMemoryExecutor.java:92)
        at java.util.concurrent.FutureTask.run(FutureTask.java:266)
        at java.util.concurrent.Executors$RunnableAdapter.call(Execu
tors.java:511)
        ... 4 more

Traceback (most recent call last):
  File "/usr/lib/spark/python/lib/pyspark.zip/pyspark/sql/readwrite
r.py", line 1109, in save
    self._jwrite.save(path)
  File "/usr/lib/spark/python/lib/py4j-0.10.9-src.zip/py4j/java_gate
way.py", line 1305, in __call__
    answer, self.gateway_client, self.target_id, self.name)
  File "/usr/lib/spark/python/lib/pyspark.zip/pyspark/sql/utils.py",
line 111, in deco
    return f(*a, **kw)
  File "/usr/lib/spark/python/lib/py4j-0.10.9-src.zip/py4j/protocol.
py", line 328, in get_return_value
    format(target_id, ".", name), value)
py4j.protocol.Py4JJavaError: An error occurred while calling o843.sa
ve.
: org.apache.spark.SparkException: Job aborted due to stage failure:
Task 0 in stage 373.0 failed 4 times, most recent failure: Lost task
0.3 in stage 373.0 (TID 126585) (ip-172-31-58-135.ap-south-1.comput
e.internal executor 10): org.apache.hudi.exception.HoodieUpsertExcep
tion: Error upserting bucketType UPDATE for partition :0
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpsertPartition(BaseSparkCommitActionExecutor.java:27
9)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.lambda$execute$ecf5068c$1(BaseSparkCommitActionExecutor.jav
a:135)
        at org.apache.spark.api.java.JavaRDDLike.$anonfun$mapPartiti
onsWithIndex$1(JavaRDDLike.scala:102)
        at org.apache.spark.api.java.JavaRDDLike.$anonfun$mapPartiti
onsWithIndex$1$adapted(JavaRDDLike.scala:102)
        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsWithInde
x$2(RDD.scala:915)
        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsWithInde
x$2$adapted(RDD.scala:915)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.$anonfun$getOrCompute$1(RDD.scal
a:386)
        at org.apache.spark.storage.BlockManager.$anonfun$doPutItera
tor$1(BlockManager.scala:1440)
        at org.apache.spark.storage.BlockManager.org$apache$spark$st
orage$BlockManager$$doPut(BlockManager.scala:1350)
        at org.apache.spark.storage.BlockManager.doPutIterator(Block
Manager.scala:1414)
        at org.apache.spark.storage.BlockManager.getOrElseUpdate(Blo
ckManager.scala:1237)
        at org.apache.spark.rdd.RDD.getOrCompute(RDD.scala:384)
```

```
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:335)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.
scala:90)
        at org.apache.spark.scheduler.Task.run(Task.scala:131)
        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$ru
n$3(Executor.scala:497)
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.sca
la:1439)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executo
r.scala:500)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadP
oolExecutor.java:1149)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Thread
PoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:748)
Caused by: org.apache.hudi.exception.HoodieException: org.apache.hud
i.exception.HoodieException: java.util.concurrent.ExecutionExceptio
n: org.apache.hudi.exception.HoodieException: operation has failed
        at org.apache.hudi.table.action.commit.SparkMergeHelper.runM
erge(SparkMergeHelper.java:102)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpdateInternal(BaseSparkCommitActionExecutor.java:30
8)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpdate(BaseSparkCommitActionExecutor.java:299)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpsertPartition(BaseSparkCommitActionExecutor.java:27
2)
        ... 28 more
Caused by: org.apache.hudi.exception.HoodieException: java.util.conc
urrent.ExecutionException: org.apache.hudi.exception.HoodieExceptio
n: operation has failed
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.execute(BoundedInMemoryExecutor.java:143)
        at org.apache.hudi.table.action.commit.SparkMergeHelper.runM
erge(SparkMergeHelper.java:100)
        ... 31 more
Caused by: java.util.concurrent.ExecutionException: org.apache.hudi.
exception.HoodieException: operation has failed
        at java.util.concurrent.FutureTask.report(FutureTask.java:12
2)
        at java.util.concurrent.FutureTask.get(FutureTask.java:192)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.execute(BoundedInMemoryExecutor.java:141)
        ... 32 more
Caused by: org.apache.hudi.exception.HoodieException: operation has
failed
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.th
rowExceptionIfFailed(BoundedInMemoryQueue.java:247)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.re
adNextRecord(BoundedInMemoryQueue.java:226)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.ac
cess$100(BoundedInMemoryQueue.java:52)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue$Qu
eueIterator.hasNext(BoundedInMemoryQueue.java:277)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueueCon
sumer.consume(BoundedInMemoryQueueConsumer.java:36)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.lambda$null$2(BoundedInMemoryExecutor.java:121)
        at java.util.concurrent.FutureTask.run(FutureTask.java:266)
        ... 3 more
Caused by: org.apache.parquet.io.InvalidRecordException: Parquet/Avr
```

```
o schema mismatch: Avro field 'new_col' not found
        at org.apache.parquet.avro.AvroRecordConverter.getAvroField
(AvroRecordConverter.java:225)
        at org.apache.parquet.avro.AvroRecordConverter.<init>(AvroRe
cordConverter.java:130)
        at org.apache.parquet.avro.AvroRecordConverter.<init>(AvroRe
cordConverter.java:95)
        at org.apache.parquet.avro.AvroRecordMaterializer.<init>(Avr
oRecordMaterializer.java:33)
        at org.apache.parquet.avro.AvroReadSupport.prepareForRead(Av
roReadSupport.java:138)
        at org.apache.parquet.hadoop.InternalParquetRecordReader.ini
tialize(InternalParquetRecordReader.java:183)
        at org.apache.parquet.hadoop.ParquetReader.initReader(Parque
tReader.java:156)
        at org.apache.parquet.hadoop.ParquetReader.read(ParquetReade
r.java:135)
        at org.apache.hudi.common.util.ParquetReaderIterator.hasNext
(ParquetReaderIterator.java:49)
        at org.apache.hudi.common.util.queue.IteratorBasedQueueProdu
cer.produce(IteratorBasedQueueProducer.java:45)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.lambda$null$0(BoundedInMemoryExecutor.java:92)
        at java.util.concurrent.FutureTask.run(FutureTask.java:266)
        at java.util.concurrent.Executors$RunnableAdapter.call(Execu
tors.java:511)
        ... 4 more

Driver stacktrace:
        at org.apache.spark.scheduler.DAGScheduler.failJobAndIndepen
dentStages(DAGScheduler.scala:2465)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$abortSta
ge$2(DAGScheduler.scala:2414)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$abortSta
ge$2$adapted(DAGScheduler.scala:2413)
        at scala.collection.mutable.ResizableArray.foreach(Resizable
Array.scala:62)
        at scala.collection.mutable.ResizableArray.foreach$(Resizabl
eArray.scala:55)
        at scala.collection.mutable.ArrayBuffer.foreach(ArrayBuffer.
scala:49)
        at org.apache.spark.scheduler.DAGScheduler.abortStage(DAGSch
eduler.scala:2413)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$handleTa
skSetFailed$1(DAGScheduler.scala:1124)
        at org.apache.spark.scheduler.DAGScheduler.$anonfun$handleTa
skSetFailed$1$adapted(DAGScheduler.scala:1124)
        at scala.Option.foreach(Option.scala:407)
        at org.apache.spark.scheduler.DAGScheduler.handleTaskSetFail
ed(DAGScheduler.scala:1124)
        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.d
oOnReceive(DAGScheduler.scala:2679)
        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.o
nReceive(DAGScheduler.scala:2621)
        at org.apache.spark.scheduler.DAGSchedulerEventProcessLoop.o
nReceive(DAGScheduler.scala:2610)
        at org.apache.spark.util.EventLoop$$anon$1.run(EventLoop.sca
la:49)
        at org.apache.spark.scheduler.DAGScheduler.runJob(DAGSchedul
er.scala:914)
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2
238)
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2
259)
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2
278)
        at org.apache.spark.SparkContext.runJob(SparkContext.scala:2
```

```
303)
        at org.apache.spark.rdd.RDD.count(RDD.scala:1253)
        at org.apache.hudi.HoodieSparkSqlWriter$.commitAndPerformPos
tOperations(HoodieSparkSqlWriter.scala:433)
        at org.apache.hudi.HoodieSparkSqlWriter$.write(HoodieSparkSq
lWriter.scala:218)
        at org.apache.hudi.DefaultSource.createRelation(DefaultSourc
e.scala:134)
        at org.apache.spark.sql.execution.datasources.SaveIntoDataSo
urceCommand.run(SaveIntoDataSourceCommand.scala:46)
        at org.apache.spark.sql.execution.command.ExecutedCommandExe
c.sideEffectResult$lzycompute(commands.scala:70)
        at org.apache.spark.sql.execution.command.ExecutedCommandExe
c.sideEffectResult(commands.scala:68)
        at org.apache.spark.sql.execution.command.ExecutedCommandExe
c.doExecute(commands.scala:90)
        at org.apache.spark.sql.execution.SparkPlan.$anonfun$execut
e$1(SparkPlan.scala:185)
        at org.apache.spark.sql.execution.SparkPlan.$anonfun$execute
Query$1(SparkPlan.scala:223)
        at org.apache.spark.rdd.RDDOperationScope$.withScope(RDDOper
ationScope.scala:151)
        at org.apache.spark.sql.execution.SparkPlan.executeQuery(Spa
rkPlan.scala:220)
        at org.apache.spark.sql.execution.SparkPlan.execute(SparkPla
n.scala:181)
        at org.apache.spark.sql.execution.QueryExecution.toRdd$lzyco
mpute(QueryExecution.scala:134)
        at org.apache.spark.sql.execution.QueryExecution.toRdd(Query
Execution.scala:133)
        at org.apache.spark.sql.DataFrameWriter.$anonfun$runComman
d$1(DataFrameWriter.scala:989)
        at org.apache.spark.sql.catalyst.QueryPlanningTracker$.withT
racker(QueryPlanningTracker.scala:107)
        at org.apache.spark.sql.execution.SQLExecution$.withTracker
(SQLExecution.scala:232)
        at org.apache.spark.sql.execution.SQLExecution$.executeQuer
y$1(SQLExecution.scala:110)
        at org.apache.spark.sql.execution.SQLExecution$.$anonfun$wit
hNewExecutionId$6(SQLExecution.scala:135)
        at org.apache.spark.sql.catalyst.QueryPlanningTracker$.withT
racker(QueryPlanningTracker.scala:107)
        at org.apache.spark.sql.execution.SQLExecution$.withTracker
(SQLExecution.scala:232)
        at org.apache.spark.sql.execution.SQLExecution$.$anonfun$wit
hNewExecutionId$5(SQLExecution.scala:135)
        at org.apache.spark.sql.execution.SQLExecution$.withSQLConfP
ropagated(SQLExecution.scala:253)
        at org.apache.spark.sql.execution.SQLExecution$.$anonfun$wit
hNewExecutionId$1(SQLExecution.scala:134)
        at org.apache.spark.sql.SparkSession.withActive(SparkSessio
n.scala:772)
        at org.apache.spark.sql.execution.SQLExecution$.withNewExecu
tionId(SQLExecution.scala:68)
        at org.apache.spark.sql.DataFrameWriter.runCommand(DataFrame
Writer.scala:989)
        at org.apache.spark.sql.DataFrameWriter.saveToV1Source(DataF
rameWriter.scala:438)
        at org.apache.spark.sql.DataFrameWriter.saveInternal(DataFra
meWriter.scala:415)
        at org.apache.spark.sql.DataFrameWriter.save(DataFrameWrite
r.scala:293)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Metho
d)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodA
ccessorImpl.java:62)
        at sun.reflect.DelegatingMethodAccessorImpl.invoke(Delegatin
```

```
gMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at py4j.reflection.MethodInvoker.invoke(MethodInvoker.java:2
44)
        at py4j.reflection.ReflectionEngine.invoke(ReflectionEngine.
java:357)
        at py4j.Gateway.invoke(Gateway.java:282)
        at py4j.commands.AbstractCommand.invokeMethod(AbstractComman
d.java:132)
        at py4j.commands.CallCommand.execute(CallCommand.java:79)
        at py4j.GatewayConnection.run(GatewayConnection.java:238)
        at java.lang.Thread.run(Thread.java:748)
Caused by: org.apache.hudi.exception.HoodieUpsertException: Error up
serting bucketType UPDATE for partition :0
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpsertPartition(BaseSparkCommitActionExecutor.java:27
9)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.lambda$execute$ecf5068c$1(BaseSparkCommitActionExecutor.jav
a:135)
        at org.apache.spark.api.java.JavaRDDLike.$anonfun$mapPartiti
onsWithIndex$1(JavaRDDLike.scala:102)
        at org.apache.spark.api.java.JavaRDDLike.$anonfun$mapPartiti
onsWithIndex$1$adapted(JavaRDDLike.scala:102)
        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsWithInde
x$2(RDD.scala:915)
        at org.apache.spark.rdd.RDD.$anonfun$mapPartitionsWithInde
x$2$adapted(RDD.scala:915)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.$anonfun$getOrCompute$1(RDD.scal
a:386)
        at org.apache.spark.storage.BlockManager.$anonfun$doPutItera
tor$1(BlockManager.scala:1440)
        at org.apache.spark.storage.BlockManager.org$apache$spark$st
orage$BlockManager$$doPut(BlockManager.scala:1350)
        at org.apache.spark.storage.BlockManager.doPutIterator(Block
Manager.scala:1414)
        at org.apache.spark.storage.BlockManager.getOrElseUpdate(Blo
ckManager.scala:1237)
        at org.apache.spark.rdd.RDD.getOrCompute(RDD.scala:384)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:335)
        at org.apache.spark.rdd.MapPartitionsRDD.compute(MapPartitio
nsRDD.scala:52)
        at org.apache.spark.rdd.RDD.computeOrReadCheckpoint(RDD.scal
a:373)
        at org.apache.spark.rdd.RDD.iterator(RDD.scala:337)
        at org.apache.spark.scheduler.ResultTask.runTask(ResultTask.
scala:90)
        at org.apache.spark.scheduler.Task.run(Task.scala:131)
        at org.apache.spark.executor.Executor$TaskRunner.$anonfun$ru
n$3(Executor.scala:497)
        at org.apache.spark.util.Utils$.tryWithSafeFinally(Utils.sca
la:1439)
        at org.apache.spark.executor.Executor$TaskRunner.run(Executo
r.scala:500)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadP
oolExecutor.java:1149)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(Thread
PoolExecutor.java:624)
```

```
        ... 1 more
Caused by: org.apache.hudi.exception.HoodieException: org.apache.hud
i.exception.HoodieException: java.util.concurrent.ExecutionExceptio
n: org.apache.hudi.exception.HoodieException: operation has failed
        at org.apache.hudi.table.action.commit.SparkMergeHelper.runM
erge(SparkMergeHelper.java:102)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpdateInternal(BaseSparkCommitActionExecutor.java:30
8)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpdate(BaseSparkCommitActionExecutor.java:299)
        at org.apache.hudi.table.action.commit.BaseSparkCommitAction
Executor.handleUpsertPartition(BaseSparkCommitActionExecutor.java:27
2)
        ... 28 more
Caused by: org.apache.hudi.exception.HoodieException: java.util.conc
urrent.ExecutionException: org.apache.hudi.exception.HoodieExceptio
n: operation has failed
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.execute(BoundedInMemoryExecutor.java:143)
        at org.apache.hudi.table.action.commit.SparkMergeHelper.runM
erge(SparkMergeHelper.java:100)
        ... 31 more
Caused by: java.util.concurrent.ExecutionException: org.apache.hudi.
exception.HoodieException: operation has failed
        at java.util.concurrent.FutureTask.report(FutureTask.java:12
2)
        at java.util.concurrent.FutureTask.get(FutureTask.java:192)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.execute(BoundedInMemoryExecutor.java:141)
        ... 32 more
Caused by: org.apache.hudi.exception.HoodieException: operation has
failed
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.th
rowExceptionIfFailed(BoundedInMemoryQueue.java:247)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.re
adNextRecord(BoundedInMemoryQueue.java:226)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue.ac
cess$100(BoundedInMemoryQueue.java:52)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueue$Qu
eueIterator.hasNext(BoundedInMemoryQueue.java:277)
        at org.apache.hudi.common.util.queue.BoundedInMemoryQueueCon
sumer.consume(BoundedInMemoryQueueConsumer.java:36)
        at org.apache.hudi.common.util.queue.BoundedInMemoryExecuto
r.lambda$null$2(BoundedInMemoryExecutor.java:121)
        at java.util.concurrent.FutureTask.run(FutureTask.java:266)
        ... 3 more
Caused by: org.apache.parquet.io.InvalidRecordException: Parquet/Avr
o schema mismatch: Avro field 'new_col' not found
        at org.apache.parquet.avro.AvroRecordConverter.getAvroField
(AvroRecordConverter.java:225)
        at org.apache.parquet.avro.AvroRecordConverter.<init>(AvroRe
cordConverter.java:130)
        at org.apache.parquet.avro.AvroRecordConverter.<init>(AvroRe
cordConverter.java:95)
        at org.apache.parquet.avro.AvroRecordMaterializer.<init>(Avr
oRecordMaterializer.java:33)
        at org.apache.parquet.avro.AvroReadSupport.prepareForRead(Av
roReadSupport.java:138)
        at org.apache.parquet.hadoop.InternalParquetRecordReader.ini
tialize(InternalParquetRecordReader.java:183)
        at org.apache.parquet.hadoop.ParquetReader.initReader(Parque
tReader.java:156)
        at org.apache.parquet.hadoop.ParquetReader.read(ParquetReade
r.java:135)
        at org.apache.hudi.common.util.ParquetReaderIterator.hasNext
(ParquetReaderIterator.java:49)
```

In [109…

```
# ensure the incomong record has the correct current schema, new fre
def evolveSchema(df,table,forcecast=False):
    try:
        #get existing table's schema
        print("\nexisting schema in hive is :")
        original_df = spark.sql("SELECT * FROM "+table+" LIMIT 0")
        original_df.printSchema()

        #sanitize for hudi specific system columns
        print("\nexisting schema in hive (sanitized for hudi columns
        columns_to_drop = ['_hoodie_commit_time', '_hoodie_commit_se
        odf = original_df.drop(*columns_to_drop)
        odf.printSchema()

        if (df.schema != odf.schema):
            merged_df = df.unionByName(odf, allowMissingColumns=True

        return (merged_df)
    except Exception as e:
        print (e)
        return (df)
```

In [111…

```
# use for backward schema evolution
# manually add default values for columns which exist in schema but

simpleData = [
    ("Person1","2021-07-22","11121314","Purple"),
    ("Person2","2021-07-22","11121314","Purple"),
    ("Person3","2021-07-22","11121314","Purple"),
    ("Person4","2021-07-22","11121314","Purple")
]

columns = ["name","date","col_to_update_integer","col_to_update_stri
df = spark.createDataFrame(data = simpleData, schema = columns)


df1 = df.select("name","date","col_to_update_integer","col_to_update
df1.printSchema()
df1.show(truncate=False)

df2 = evolveSchema(df1,tableName,False)

df2.printSchema()
df2.show(truncate=False)

df2.write.format("hudi") \
.options(**combinedConf) \
.mode("append") \
.save(tablePath)
```

```
root
 |-- name: string (nullable = true)
 |-- date: string (nullable = true)
 |-- col_to_update_integer: string (nullable = true)
```

```
 |-- col_to_update_string: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- day: integer (nullable = true)

+-------+----------+--------------------+--------------------+----+
-----+---+
|name   |date      |col_to_update_integer|col_to_update_string|year|
month|day|
+-------+----------+--------------------+--------------------+----+
-----+---+
|Person1|2021-07-22|11121314            |Purple              |2021|
7    |22 |
|Person2|2021-07-22|11121314            |Purple              |2021|
7    |22 |
|Person3|2021-07-22|11121314            |Purple              |2021|
7    |22 |
|Person4|2021-07-22|11121314            |Purple              |2021|
7    |22 |
+-------+----------+--------------------+--------------------+----+
-----+---+


existing schema in hive is :
root
 |-- _hoodie_commit_time: string (nullable = true)
 |-- _hoodie_commit_seqno: string (nullable = true)
 |-- _hoodie_record_key: string (nullable = true)
 |-- _hoodie_partition_path: string (nullable = true)
 |-- _hoodie_file_name: string (nullable = true)
 |-- date: string (nullable = true)
 |-- col_to_update_integer: string (nullable = true)
 |-- col_to_update_string: string (nullable = true)
 |-- new_col: string (nullable = true)
 |-- name: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- day: integer (nullable = true)


existing schema in hive (sanitized for hudi columns) is :
root
 |-- date: string (nullable = true)
 |-- col_to_update_integer: string (nullable = true)
 |-- col_to_update_string: string (nullable = true)
 |-- new_col: string (nullable = true)
 |-- name: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- day: integer (nullable = true)

root
 |-- name: string (nullable = true)
 |-- date: string (nullable = true)
 |-- col_to_update_integer: string (nullable = true)
 |-- col_to_update_string: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- day: integer (nullable = true)
 |-- new_col: string (nullable = true)

+-------+----------+--------------------+--------------------+----+
-----+---+-------+
|name   |date      |col_to_update_integer|col_to_update_string|year|
month|day|new_col|
+-------+----------+--------------------+--------------------+----+
-----+---+-------+
```

```
|Person1|2021-07-22|11121314            |Purple               |2021|
7    |22 |null   |
|Person2|2021-07-22|11121314            |Purple               |2021|
7    |22 |null   |
|Person3|2021-07-22|11121314            |Purple               |2021|
7    |22 |null   |
|Person4|2021-07-22|11121314            |Purple               |2021|
7    |22 |null   |
```

In [113…

```
## Check via PySpark
hudiDF = spark.read \
.format("hudi") \
.load(tablePath).show(truncate=False)
```

```
+-------------------+--------------------+------------------+-------
------------------------------+--------------------------------------
------------------------------------+-------+----------+-----------
----------+--------------------+----+-----+---+-------+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
|name   |date      |col_to_update_integer|col_to_update_string|year|
month|day|new_col|
+-------------------+--------------------+------------------+-------
------------------------------+--------------------------------------
------------------------------------+-------+----------+-----------
----------+--------------------+----+-----+---+-------+
|20210808171322     |20210808171322_3_2  |Person4           |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-510-167394_20210808171322.parquet|Person4|2021-07-22|11121314
|Purple               |2021|7    |22 |null   |
|20210808171322     |20210808171322_0_2  |Person3           |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-510-167391_20210808171322.parquet|Person3|2021-07-22|11121314
|Purple               |2021|7    |22 |null   |
|20210808171322     |20210808171322_2_1  |Person2           |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-510-167393_20210808171322.parquet|Person2|2021-07-22|11121314
|Purple               |2021|7    |22 |null   |
|20210808171322     |20210808171322_1_1  |Person1           |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-510-167392_20210808171322.parquet|Person1|2021-07-22|11121314
|Purple               |2021|7    |22 |null   |
+-------------------+--------------------+------------------+-------
------------------------------+--------------------------------------
------------------------------------+-------+----------+-----------
----------+--------------------+----+-----+---+-------+
```

In [114…

```
# use for backward schema evolution
# manually add default values for columns which exist in schema but

simpleData = [
    ("Person1","2021-07-22","15161718","Orange","again"),
    ("Person2","2021-07-22","15161718","Orange","again"),
    ("Person3","2021-07-22","15161718","Orange","again"),
    ("Person4","2021-07-22","15161718","Orange","again")
]

columns = ["name","date","col_to_update_integer","col_to_update_stri
df = spark.createDataFrame(data = simpleData, schema = columns)


df1 = df.select("name","date","col_to_update_integer","col_to_update
df1.printSchema()
df1.show(truncate=False)

#df2 = evolveSchema(df1,tableName,False)
#df2.printSchema()
#df2.show(truncate=False)

df1.write.format("hudi") \
.options(**combinedConf) \
.mode("append") \
.save(tablePath)
```

```
root
 |-- name: string (nullable = true)
 |-- date: string (nullable = true)
 |-- col_to_update_integer: string (nullable = true)
 |-- col_to_update_string: string (nullable = true)
 |-- new_col: string (nullable = true)
 |-- year: integer (nullable = true)
 |-- month: integer (nullable = true)
 |-- day: integer (nullable = true)

+-------+----------+---------------------+--------------------+-----
--+----+-----+---+
|name   |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+-------+----------+---------------------+--------------------+-----
--+----+-----+---+
|Person1|2021-07-22|15161718             |Orange              |again
|2021|7    |22 |
|Person2|2021-07-22|15161718             |Orange              |again
|2021|7    |22 |
|Person3|2021-07-22|15161718             |Orange              |again
|2021|7    |22 |
|Person4|2021-07-22|15161718             |Orange              |again
|2021|7    |22 |
+-------+----------+---------------------+--------------------+-----
--+----+-----+---+
```

In [115…

```
## Check via PySpark
hudiDF = spark.read \
.format("hudi") \
.load(tablePath).show(truncate=False)
```

```
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
----------+--------------------+-------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
|name    |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
----------+--------------------+-------+----+-----+---+
|20210808171924     |20210808171924_0_12 |Person3           |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-581-194531_20210808171924.parquet|Person3|2021-07-22|15161718
|Orange              |again  |2021|7    |22 |
|20210808171924     |20210808171924_2_11 |Person2           |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-581-194533_20210808171924.parquet|Person2|2021-07-22|15161718
|Orange              |again  |2021|7    |22 |
|20210808171924     |20210808171924_1_12 |Person1           |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-581-194532_20210808171924.parquet|Person1|2021-07-22|15161718
|Orange              |again  |2021|7    |22 |
|20210808171924     |20210808171924_3_11 |Person4           |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-581-194534_20210808171924.parquet|Person4|2021-07-22|15161718
|Orange              |again  |2021|7    |22 |
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
----------+--------------------+-------+----+-----+---+
```

In [3]:

```
#TimeTravel
# available commit timestamps in .hoodie folder of S3 :
# 20210808155456
# 20210808155922
# 20210808160202
# 20210808160422
# 20210808171322
# 20210808171924

starttime = "0000"
endtime = "20210808155456"

hudiDF = spark.read \
.format("hudi") \
.option("hoodie.datasource.query.type", "incremental") \
.option("hoodie.datasource.read.begin.instanttime", starttime) \
.option("hoodie.datasource.read.end.instanttime", endtime) \
.load(tablePath).show(truncate=False)
```

```
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
---------+--------------------+-------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
|name    |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
---------+--------------------+-------+----+-----+---+
```

```
+----------------------------------+-------+----------+------------
---------+--------------------+-------+----+-----+---+
|20210808155456     |20210808155456_2_3  |Person2           |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-228-72348_20210808155456.parquet|Person2|2021-07-22|1234
|White               |null   |2021|7    |22 |
|20210808155456     |20210808155456_1_3  |Person1           |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-228-72347_20210808155456.parquet|Person1|2021-07-22|1234
|White               |null   |2021|7    |22 |
|20210808155456     |20210808155456_0_4  |Person3           |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-228-72346_20210808155456.parquet|Person3|2021-07-22|1234
|White               |null   |2021|7    |22 |
|20210808155456     |20210808155456_3_4  |Person4           |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-228-72349_20210808155456.parquet|Person4|2021-07-22|1234
|White               |null   |2021|7    |22 |
+------------------+--------------------+------------------+-------
-----------------------------+-----------------------------------
----------------------------------+-------+----------+------------
```

In [4]:

```
starttime = "20210808155456"
endtime = "20210808155922"

hudiDF = spark.read \
.format("hudi") \
.option("hoodie.datasource.query.type", "incremental") \
.option("hoodie.datasource.read.begin.instanttime", starttime) \
.option("hoodie.datasource.read.end.instanttime", endtime) \
.load(tablePath).show(truncate=False)
```

```
+------------------+--------------------+------------------+-------
-----------------------------+-----------------------------------
----------------------------------+-------+----------+------------
---------+--------------------+-------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
|name    |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+------------------+--------------------+------------------+-------
-----------------------------+-----------------------------------
----------------------------------+-------+----------+------------
---------+--------------------+-------+----+-----+---+
|20210808155922     |20210808155922_2_6  |Person2           |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-265-85906_20210808155922.parquet|Person2|2021-07-22|4567
|Yellow              |null   |2021|7    |22 |
|20210808155922     |20210808155922_0_5  |Person3           |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-265-85904_20210808155922.parquet|Person3|2021-07-22|4567
|Yellow              |null   |2021|7    |22 |
|20210808155922     |20210808155922_1_6  |Person1           |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-265-85905_20210808155922.parquet|Person1|2021-07-22|4567
|Yellow              |null   |2021|7    |22 |
|20210808155922     |20210808155922_3_5  |Person4           |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-265-85907_20210808155922.parquet|Person4|2021-07-22|4567
|Yellow              |null   |2021|7    |22 |
+------------------+--------------------+------------------+-------
-----------------------------+-----------------------------------
----------------------------------+-------+----------+------------
```

In [5]:

```
starttime = "20210808155922"
endtime = "20210808160202"

hudiDF = spark.read \
.format("hudi") \
.option("hoodie.datasource.query.type", "incremental") \
.option("hoodie.datasource.read.begin.instanttime", starttime) \
.option("hoodie.datasource.read.end.instanttime", endtime) \
.load(tablePath).show(truncate=False)
```

```
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
---------+--------------------+-------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
|name   |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
---------+--------------------+-------+----+-----+---+
|20210808160202     |20210808160202_1_8  |Person1           |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-302-99463_20210808160202.parquet|Person1|2021-07-22|8910
|Silver               |null   |2021|7    |22 |
|20210808160202     |20210808160202_0_8  |Person3           |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-302-99462_20210808160202.parquet|Person3|2021-07-22|8910
|Silver               |null   |2021|7    |22 |
|20210808160202     |20210808160202_2_7  |Person2           |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-302-99464_20210808160202.parquet|Person2|2021-07-22|8910
|Silver               |null   |2021|7    |22 |
|20210808160202     |20210808160202_3_7  |Person4           |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-302-99465_20210808160202.parquet|Person4|2021-07-22|8910
|Silver               |null   |2021|7    |22 |
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
---------+--------------------+-------+----+-----+---+
```

In [6]:

```
starttime = "20210808160202"
endtime = "20210808160422"

hudiDF = spark.read \
.format("hudi") \
.option("hoodie.datasource.query.type", "incremental") \
.option("hoodie.datasource.read.begin.instanttime", starttime) \
.option("hoodie.datasource.read.end.instanttime", endtime) \
.load(tablePath).show(truncate=False)
```

```
+------------------+--------------------+------------------+-------
------------------------------+-------------------------------------
-------------------------------------+-------+----------+----------
---------+--------------------+-------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
```

```
|name    |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+-------------------+--------------------+-----------------+-------
-------------------------------+-------------------------------------
-------------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
|20210808160422     |20210808160422_0_10 |Person3          |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-336-113015_20210808160422.parquet|Person3|2021-07-22|8910
|Silver             |abc    |2021|7    |22 |
|20210808160422     |20210808160422_3_10 |Person4          |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-336-113018_20210808160422.parquet|Person4|2021-07-22|8910
|Silver             |abc    |2021|7    |22 |
|20210808160422     |20210808160422_2_9  |Person2          |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-336-113017_20210808160422.parquet|Person2|2021-07-22|8910
|Silver             |abc    |2021|7    |22 |
|20210808160422     |20210808160422_1_9  |Person1          |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-336-113016_20210808160422.parquet|Person1|2021-07-22|8910
|Silver             |abc    |2021|7    |22 |
+-------------------+--------------------+-----------------+-------
-------------------------------+-------------------------------------
-------------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
```

In [7]:

```
starttime = "20210808160422"
endtime = "20210808171322"

hudiDF = spark.read \
.format("hudi") \
.option("hoodie.datasource.query.type", "incremental") \
.option("hoodie.datasource.read.begin.instanttime", starttime) \
.option("hoodie.datasource.read.end.instanttime", endtime) \
.load(tablePath).show(truncate=False)
```

```
+-------------------+--------------------+-----------------+-------
-------------------------------+-------------------------------------
-------------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path               |_hoodie_file_name
|name    |date      |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+-------------------+--------------------+-----------------+-------
-------------------------------+-------------------------------------
-------------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
|20210808171322     |20210808171322_2_1  |Person2          |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-510-167393_20210808171322.parquet|Person2|2021-07-22|11121314
|Purple             |null   |2021|7    |22 |
|20210808171322     |20210808171322_0_2  |Person3          |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-510-167391_20210808171322.parquet|Person3|2021-07-22|11121314
|Purple             |null   |2021|7    |22 |
|20210808171322     |20210808171322_3_2  |Person4          |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-510-167394_20210808171322.parquet|Person4|2021-07-22|11121314
|Purple             |null   |2021|7    |22 |
|20210808171322     |20210808171322_1_1  |Person1          |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-510-167392_20210808171322.parquet|Person1|2021-07-22|11121314
```

```
|Purple                |null   |2021|7    |22 |
+--------------------+--------------------+------------------+-------
--------------------------------+------------------------------------
--------------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
```

In [8]:

```
starttime = "20210808171322"
endtime = "20210808171924"

hudiDF = spark.read \
.format("hudi") \
.option("hoodie.datasource.query.type", "incremental") \
.option("hoodie.datasource.read.begin.instanttime", starttime) \
.option("hoodie.datasource.read.end.instanttime", endtime) \
.load(tablePath).show(truncate=False)
```

```
+--------------------+--------------------+------------------+-------
--------------------------------+------------------------------------
--------------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
|_hoodie_commit_time|_hoodie_commit_seqno|_hoodie_record_key|_hoodie
_partition_path                |_hoodie_file_name
|name    |date       |col_to_update_integer|col_to_update_string|new_c
ol|year|month|day|
+--------------------+--------------------+------------------+-------
--------------------------------+------------------------------------
--------------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
|20210808171924      |20210808171924_2_11 |Person2           |name=Pe
rson2/year=2021/month=7/day=22|c2f7bac6-24ab-4661-8791-4e299781dd92-
0_2-581-194533_20210808171924.parquet|Person2|2021-07-22|15161718
|Orange                |again  |2021|7    |22 |
|20210808171924      |20210808171924_0_12 |Person3           |name=Pe
rson3/year=2021/month=7/day=22|f265b2d9-0b54-4c0d-bdf8-25326d6454e9-
0_0-581-194531_20210808171924.parquet|Person3|2021-07-22|15161718
|Orange                |again  |2021|7    |22 |
|20210808171924      |20210808171924_1_12 |Person1           |name=Pe
rson1/year=2021/month=7/day=22|7e28e1f2-9cee-4b6c-ada0-3bf587f8aa7c-
0_1-581-194532_20210808171924.parquet|Person1|2021-07-22|15161718
|Orange                |again  |2021|7    |22 |
|20210808171924      |20210808171924_3_11 |Person4           |name=Pe
rson4/year=2021/month=7/day=22|a1544097-1394-4214-8de5-43aa8f27f273-
0_3-581-194534_20210808171924.parquet|Person4|2021-07-22|15161718
|Orange                |again  |2021|7    |22 |
+--------------------+--------------------+------------------+-------
--------------------------------+------------------------------------
--------------------------------------+-------+----------+-----------
----------+--------------------+-------+----+-----+---+
```

In [ ]: