

```
ssh -i ~/.ssh/id_rsa_student898_2 student898_2@37.139.41.176
hdfs dfs -ls
hdfs dfs -mkdir input_csv_for_stream
```

```
student898_2@bigdataanalytics-worker-3: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
drwxr-xr-x - student898_2 student898_2      0 2022-01-04 14:47 shadrin_iris_console_checkpoint
drwxr-xr-x - student898_2 student898_2      0 2022-01-12 19:42 shadrin_iris_file_checkpoint
drwxr-xr-x - student898_2 student898_2      0 2022-01-13 19:03 shadrin_iris_kafka_checkpoint
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -mkdir input_csv_for_stream
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -ls
Found 8 items
drwx----- - student898_2 student898_2      0 2022-01-11 06:00 .Trash
drwxr-xr-x - student898_2 student898_2      0 2022-01-10 18:26 .sparkStaging
drwxr-xr-x - student898_2 student898_2      0 2021-12-15 22:13 for_stream
drwxr-xr-x - student898_2 student898_2      0 2022-01-15 19:33 input_csv_for_stream
drwxr-xr-x - student898_2 student898_2      0 2022-01-12 19:44 my_parquet_sink
drwxr-xr-x - student898_2 student898_2      0 2022-01-04 14:47 shadrin_iris_console_checkpoint
drwxr-xr-x - student898_2 student898_2      0 2022-01-12 19:42 shadrin_iris_file_checkpoint
drwxr-xr-x - student898_2 student898_2      0 2022-01-13 19:03 shadrin_iris_kafka_checkpoint
[student898_2@bigdataanalytics-worker-3 ~]$
```

Скопируем подготовленные файлы на удаленный сервер с помощью команды `scp`. Эта команда запускается в другом терминале на локальном компьютере, а не на удалённом сервере

```
scp -i ~/.ssh/id_rsa_student898_2 -r data.csv student898_2@37.139.41.176:~/for_stream
scp -i ~/.ssh/id_rsa_student898_2 -r data.json student898_2@37.139.41.176:~/for_stream
```

```
igor@igor-MS-7808: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3: ~ x  igor@igor-MS-7808: ~ x
igor@igor-MS-7808:~$ scp -i ~/.ssh/id_rsa_student898_2 -r data.csv student898_2@37.139.41.176:~/for_stream
data.csv 100% 5200KB 15.4MB/s 00:00
igor@igor-MS-7808:~$ scp -i ~/.ssh/id_rsa_student898_2 -r data.json student898_2@37.139.41.176:~/for_stream
data.json 100% 15MB 18.5MB/s 00:00
igor@igor-MS-7808:~$
```

ls for_stream/

```
student898_2@bigdataanalytics-worker-3: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3: ~ x  igor@igor-MS-7808: ~ x
[student898_2@bigdataanalytics-worker-3 ~]$ ls for_stream/
archive.csv dataset.csv iris.json product_list1.csv product_list2.csv product_list3.csv product_list4.csv product_list.csv
[student898_2@bigdataanalytics-worker-3 ~]$ ls for_stream/
archive.csv data.json iris.json product_list2.csv product_list4.csv
data.csv dataset.csv product_list1.csv product_list3.csv product_list.csv
[student898_2@bigdataanalytics-worker-3 ~]$
```

cat for_stream/data.csv

```
student898_2@bigdataanalytics-worker-3: ~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3: ~ x  igor@igor-MS-7808: ~ x
"2022-01-10 13:08:01","2000","27","28"
"2022-01-10 13:09:02","2000","27","28"
"2022-01-10 13:10:02","2000","27","28"
"2022-01-10 13:11:01","2000","27","28"
"2022-01-10 13:12:02","2000","27","28"
"2022-01-10 13:13:02","2000","27","28"
"2022-01-10 13:14:01","2000","27","28"
"2022-01-10 13:31:13","14.4","23","42"
"2022-01-10 13:32:02","15.53","23","38"
[student898_2@bigdataanalytics-worker-3 ~]$
```

Запускаем `pyspark`

Инициализация стрима

В командной строке `pyspark` импортируем нужные методы и определяем функцию `console_output` для вывода стрима в консоль.

```
from pyspark.sql import functions as F
from pyspark.sql.types import StructType, StringType
def console_output(df, freq):
```

```

return df.writeStream \
    .format("console") \
    .trigger(processingTime='%s seconds' % freq) \
    .options(truncate=False) \
    .start()

```

```

student898_2@bigdataanalytics-worker-3:~
version 2.3.2.3.1.4.0-315
Using Python version 2.7.5 (default, Nov 16 2020 22:23:17)
SparkSession available as 'spark'.
>>> from pyspark.sql import functions as F
>>> from pyspark.sql.types import StructType, StringType
>>> def console_output(df, freq):
...     return df.writeStream \
...         .format("console") \
...         .trigger(processingTime='%s seconds' % freq) \
...         .options(truncate=False) \
...         .start()
...
>>>

```

Определяем схему наших файлов

```

schema = StructType().add("time_id", StringType()).add("ping_ms", StringType()).add("temperature_c",
StringType()).add("humidity_p", StringType())

```

Создаём стрим чтения из файла (с параметром `.format("csv")`). В `options`` указываем папку на HDFS, из которой будут читаться файлы

```

raw_files = spark \
    .readStream \
    .format("csv") \
    .schema(schema) \
    .options(path="input_csv_for_stream", header=True) \
    .load()

```

Запускаем стрим

```

out = console_output(raw_files, 5)

```

```

student898_2@bigdataanalytics-worker-3:~
>>> schema = StructType().add("time_id", StringType()).add("ping_ms", StringType()).add("temperature_c", StringType()).add("humidity_p", StringType())
>>> raw_files = spark \
...     .readStream \
...     .format("csv") \
...     .schema(schema) \
...     .options(path="input_csv_for_stream", header=True) \
...     .load()
>>> out = console_output(raw_files, 5)
>>>

```

В соседнем терминале подключаемся к удалённому серверу `worker-3` и переходим в каталог с загруженными файлами

```

ssh -i ~/.ssh/id_rsa_student898_2 student898_2@37.139.41.176

```

```

ls

```

```

ll for_stream

```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

student898_2@bigdataanalytics-worker-3:~
student898_2@bigdataanalytics-worker-3:~

[student898_2@bigdataanalytics-worker-3 ~]$ ll for_stream
итого 20172
-rw-rw-r-- 1 student898_2 student898_2 11686 янв 3 18:18 archive.csv
-rw-rw-r-- 1 student898_2 student898_2 5325242 янв 15 19:45 data.csv
-rw-rw-r-- 1 student898_2 student898_2 15230928 янв 15 19:45 data.json
-rwxr-xr-x 1 student898_2 student898_2 43320 дек 29 20:29 dataset.csv
-rw-rw-r-- 1 student898_2 student898_2 15802 янв 1 10:37 iris.json
-rw-rw-r-- 1 student898_2 student898_2 98 дек 16 18:20 product_list1.csv
-rw-rw-r-- 1 student898_2 student898_2 126 дек 16 19:21 product_list2.csv
-rw-rw-r-- 1 student898_2 student898_2 128 дек 16 19:17 product_list3.csv
-rw-rw-r-- 1 student898_2 student898_2 128 дек 16 19:19 product_list4.csv
-rw-rw-r-- 1 student898_2 student898_2 125 дек 16 18:10 product_list.csv
[student898_2@bigdataanalytics-worker-3 ~]$
```

Копируем файл на HDFS

```
hdfs dfs -put for_stream/data.csv input_csv_for_stream
```

Завершаем стрим командой `out.stop()`

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

student898_2@bigdataanalytics-worker-3:~
student898_2@bigdataanalytics-worker-3:~

>>> out = console_output(raw_files, 5)
-----
Batch: 0
-----
+-----+
|time_id|ping_ms|temperature_c|humidity_p|
+-----+
|2021-09-30 21:08:02|17.28|25|35|
|2021-09-30 21:09:02|17.73|23|40|
|2021-09-30 21:10:01|18.59|22|41|
|2021-09-30 21:12:02|16.73|22|42|
|2021-09-30 21:13:02|18.12|22|42|
|2021-09-30 21:14:01|18.21|22|43|
|2021-09-30 21:15:01|17.92|22|43|
|2021-09-30 21:16:02|17.2|22|43|
|2021-09-30 21:17:02|18.16|22|43|
|2021-09-30 21:18:02|21.35|22|42|
|2021-09-30 21:19:01|17.31|22|43|
|2021-09-30 21:20:01|17.78|22|42|
|2021-09-30 21:21:02|16.9|22|42|
|2021-09-30 21:22:02|16.82|22|42|
|2021-09-30 21:23:02|17.35|22|42|
|2021-09-30 21:24:01|18|22|43|
|2021-09-30 21:25:01|16.35|22|43|
|2021-09-30 21:26:02|18.63|22|42|
|2021-09-30 21:28:01|18.52|22|43|
|2021-09-30 21:29:01|18.17|22|42|
+-----+
only showing top 20 rows

22/01/15 20:09:32 WARN ProcessingTimeExecutor: Current batch is falling behind. The trigger interval is 5000 milliseconds, but spent 12313 milliseconds
out.stop()
>>> out.stop()
```

Попробуем запустить стрим с другими опциями.

Параметр `maxFilesPerTrigger` определяет сколько файлов будет прочитано в одном батче. При этом, если необработанных файлов меньше чем `maxFilesPerTrigger`, то они не будут прочитаны и батч не появится

```
raw_files = spark \
    .readStream \
    .format("csv") \
    .schema(schema) \
```

```
.options(path="input_csv_for_stream",
        header=True,
        maxFilesPerTrigger=1) \
.load()
```

Так же добавим свою колонку `test`

```
extra_files = raw_files \
    .withColumn("test", F.col("humidity_p") / F.col("temperature_c"))
```

Запускаем стрим

```
out = console_output(extra_files, 5)
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~
...      .withColumn("test", F.col("humidity_p") / F.col("temperature_c"))
>>> out = console_output(extra_files, 5)
-----
Batch: 0
-----
+-----+-----+-----+-----+-----+
|time_id|ping_ms|temperature_c|humidity_p|test|
+-----+-----+-----+-----+-----+
|2021-09-30 21:08:02|17.28|25|35|1.4|
|2021-09-30 21:09:02|17.73|23|40|1.7391304347826086|
|2021-09-30 21:10:01|18.59|22|41|1.8636363636363635|
|2021-09-30 21:12:02|16.73|22|42|1.9090909090909092|
|2021-09-30 21:13:02|18.12|22|42|1.9090909090909092|
|2021-09-30 21:14:01|18.21|22|43|1.9545454545454546|
|2021-09-30 21:15:01|17.92|22|43|1.9545454545454546|
|2021-09-30 21:16:02|17.2|22|43|1.9545454545454546|
|2021-09-30 21:17:02|18.16|22|43|1.9545454545454546|
|2021-09-30 21:18:02|21.35|22|42|1.9090909090909092|
|2021-09-30 21:19:01|17.31|22|43|1.9545454545454546|
|2021-09-30 21:20:01|17.78|22|42|1.9090909090909092|
|2021-09-30 21:21:02|16.9|22|42|1.9090909090909092|
|2021-09-30 21:22:02|16.82|22|42|1.9090909090909092|
|2021-09-30 21:23:02|17.35|22|42|1.9090909090909092|
|2021-09-30 21:24:01|18|22|43|1.9545454545454546|
|2021-09-30 21:25:01|16.35|22|43|1.9545454545454546|
|2021-09-30 21:26:02|18.63|22|42|1.9090909090909092|
|2021-09-30 21:28:01|18.52|22|43|1.9545454545454546|
|2021-09-30 21:29:01|18.17|22|42|1.9090909090909092|
+-----+-----+-----+-----+-----+
only showing top 20 rows

out.stop()
>>> out.stop()
>>>
```

Закрываем стрим и выходим из консоли `r pyspark`

```
out.stop()
```

```
exit()
```

Удаляем файлы из HDFS. Локально пока оставим, может пригодятся

```
hdfs dfs -rm -r input_csv_for_stream
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~ x student898_2@bigdataanalytics-worker-3:~ x
+-----+
only showing top 20 rows
out.stop()
>>> exit()
[student898_2@bigdataanalytics-worker-3 ~]$ hdfs dfs -rm -r input_csv_for_stream
22/01/15 21:01:06 INFO fs.TrashPolicyDefault: Moved: 'hdfs://bigdataanalytics-head-0.mcs.local:8020/user/student898_2/input_csv_for_stream' to trash at: hdfs://bigdataanalytics-head-0.mcs.local:8020/user/student898_2/.Trash/Current/user/student898_2/
input_csv_for_stream
[student898_2@bigdataanalytics-worker-3 ~]$
```

Задание 2. Создать свой топик/топики, загрузить туда через консоль осмысленные данные с kaggle. Лучше в формате json. Много сообщений не нужно, достаточно штук 10-100. Прочитать свой топик так же, как на уроке.

`/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --list --zookeeper bigdataanalytics-worker-3:2181`

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~ x student898_2@bigdataanalytics-worker-3:~ x
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --list --zookeeper bigdataanalytics-worker-3:2181
898_1
__consumer_offsets
cherneev-test
cherneev_test
incident_event_json
orders_json
shadrin_iris
shadrin_iris_sink
test_lesson2_1
[student898_2@bigdataanalytics-worker-3 ~]$
```

Аналогично второму уроку создадим топик ``shadrin_data_test``

`/usr/hdp/current/kafka-broker/bin/kafka-topics.sh --create --topic shadrin_data_test --zookeeper bigdataanalytics-worker-3:2181 --partitions 1 --replication-factor 1`

В одном терминале запустим ``console-consumer`` чтобы прочитать из kafka (проконтролировать)

`/usr/hdp/current/kafka-broker/bin/kafka-console-consumer.sh --topic shadrin_data_test --bootstrap-server bigdataanalytics-worker-3:6667`

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~ x student898_2@bigdataanalytics-worker-3:~ x
WARNING: Due to limitations in metric names, topics with a period ('.') or underscore ('_') could collide. To avoid issues it
is best to use either, but not both.
Created topic "shadrin_data test".
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-topics.sh --list --zookeeper bigdataanalytics-worker-3:2181
898_1
__consumer_offsets
cherneev-test
cherneev_test
incident_event_json
orders_json
shadrin_data_test
shadrin_iris
shadrin_iris_sink
test_lesson2_1
[student898_2@bigdataanalytics-worker-3 ~]$ /usr/hdp/current/kafka-broker/bin/kafka-console-consumer.sh --topic shadrin_data_test --bootstrap-server bigdataanalytics-worker-3:6667

```


В другом терминале
less for_stream/data.json

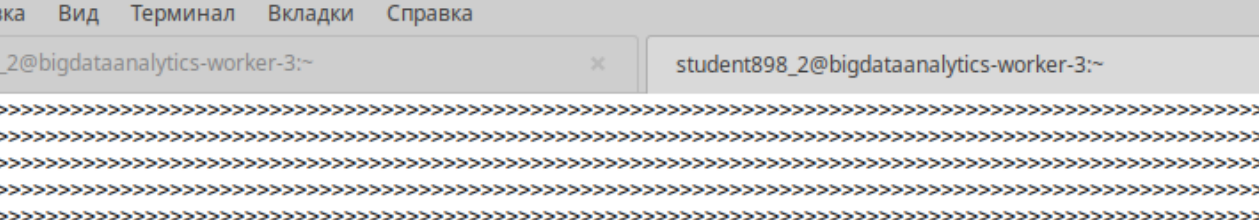
```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~ student898_2@bigdataanalytics-worker-3:~
/**
Export to JSON plugin for PHPMyAdmin
@version 4.6.6deb5
*/

// Database 'cs_project'

// cs_project.data

[{"time_id": "2021-09-30 21:08:02",
  "ping_ms": "17.28",
  "temperature_c": "25",
  "humidity_p": "35"},
 {"time_id": "2021-09-30 21:09:02",
  "ping_ms": "17.73",
  "temperature_c": "25",
  "humidity_p": "35"}]
for stream/data.json
```

```
/usr/hdp/current/kafka-broker/bin/kafka-console-producer.sh --topic shadrin_data --broker-list bigdataanalytics-worker-3:6667 < for stream/data.json
```



The screenshot shows a terminal window titled "student898_2@bigdataanalytics-worker-3:~ - Терминал". The window has a menu bar with "Файл", "Правка", "Вид", "Терминал", "Вкладки", and "Справка". Below the menu bar, there are two tabs: "student898_2@bigdataanalytics-worker-3:~" and "student898_2@bigdataanalytics-worker-3:~". The main content of the terminal is a large number of "======" characters, which appear to be a separator or a large output. The prompt "[student898_2@bigdataanalytics-worker-3 ~]\$" is visible at the bottom right.

```
/usr/hdp/current/kafka-broker/bin/kafka-console-consumer.sh --topic shadrin_data_test --bootstrap-server bigdataanalytics-worker-3:6667 --from-beginning
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~
student898_2@bigdataanalytics-worker-3:~
}, {
  "time_id": "2022-01-10 13:14:01",
  "ping_ms": "2000",
  "temperature_c": "27",
  "humidity_p": "28"
}, {
  "time_id": "2022-01-10 13:31:13",
  "ping_ms": "14.4",
  "temperature_c": "23",
  "humidity_p": "42"
}, {
  "time_id": "2022-01-10 13:32:02",
  "ping_ms": "15.53",
  "temperature_c": "23",
  "humidity_p": "38"
}]
^CProcessed a total of 651695 messages
[student898_2@bigdataanalytics-worker-3 ~]$
```

Посмотрим пред загруженное shadrin_data

```
/usr/hdp/current/kafka-broker/bin/kafka-console-consumer.sh --topic shadrin_data --bootstrap-server
bigdataanalytics-worker-3:6667 --from-beginning
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~
student898_2@bigdataanalytics-worker-3:~
}, {
  "time_id": "2022-01-10 13:14:01",
  "ping_ms": "2000",
  "temperature_c": "27",
  "humidity_p": "28"
}, {
  "time_id": "2022-01-10 13:31:13",
  "ping_ms": "14.4",
  "temperature_c": "23",
  "humidity_p": "42"
}, {
  "time_id": "2022-01-10 13:32:02",
  "ping_ms": "15.53",
  "temperature_c": "23",
  "humidity_p": "38"
}]
^CProcessed a total of 651695 messages
[student898_2@bigdataanalytics-worker-3 ~]$
```

Переходим в консоль pyspark.

```
export SPARK_KAFKA_VERSION=0.10
```

```
pyspark --master local[1] --packages org.apache.spark:spark-sql-kafka-0-10_2.11:2.3.2
```



```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~
... option("kafka.bootstrap.servers", kafka_brokers). \
... option("subscribe", "shadrin_data"). \
... option("startingOffsets", "earliest"). \
... option("endingOffsets", ""{"shadrin_data":{"0":20}}"""). \
... load()
>>> out = console_output(raw_data, 10)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "<stdin>", line 2, in console_output
  File "/usr/hdp/current/spark2-client/python/pyspark/sql/dataframe.py", line 238, in writeStream
    return DataStreamWriter(self)
  File "/usr/hdp/current/spark2-client/python/pyspark/sql/streaming.py", line 684, in __init__
    self._jwrite = df._jdf.writeStream()
  File "/usr/hdp/current/spark2-client/python/lib/py4j-0.10.7-src.zip/py4j/java_gateway.py", line 1257, in __call__
  File "/usr/hdp/current/spark2-client/python/pyspark/sql/utils.py", line 69, in deco
    raise AnalysisException(s.split(':', 1)[1], stackTrace)
pyspark.sql.utils.AnalysisException: u"'writeStream' can be called only on streaming Dataset/DataFrame;"
>>>
```

```
raw_data = spark.readStream. \
    format("kafka"). \
    option("kafka.bootstrap.servers", kafka_brokers). \
    option("subscribe", "shadrin_data"). \
    option("startingOffsets", "earliest"). \
    option("maxOffsetsPerTrigger", "5"). \
    load()
```

```
out = console_output(raw_data, 10)
out.stop()
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~
-----
Batch: 1
-----
+-----+-----+-----+-----+-----+-----+-----+
| key|          value|      topic|partition|offset|          timestamp|timestampType|
+-----+-----+-----+-----+-----+-----+-----+
|null|[2F 2F 20 44 61 7...|shadrin_data|      0|    5|2022-01-15 21:14:...|      0|
|null|[]|shadrin_data|      0|    6|2022-01-15 21:14:...|      0|
|null|[2F 2F 20 63 73 5...|shadrin_data|      0|    7|2022-01-15 21:14:...|      0|
|null|[]|shadrin_data|      0|    8|2022-01-15 21:14:...|      0|
|null|[5B 7B]|shadrin_data|      0|    9|2022-01-15 21:14:...|      0|
+-----+-----+-----+-----+-----+-----+-----+
out.stop()
>>>
```

Парсинг сообщений.
Посмотрим в каком формате в Кафке хранятся сообщения.
raw_data.printSchema()

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

student898_2@bigdataanalytics-worker-3:~
student898_2@bigdataanalytics-worker-3:~
[null]          []|shadrin_data|    0|    8|2022-01-15 21:14:...|    0|
[null]          [5B 7B]|shadrin_data|  0|    9|2022-01-15 21:14:...|    0|
+-----+-----+-----+-----+-----+-----+-----+-----+
out.stop()
>>> raw_data.printSchema()
root
|-- key: binary (nullable = true)
|-- value: binary (nullable = true)
|-- topic: string (nullable = true)
|-- partition: integer (nullable = true)
|-- offset: long (nullable = true)
|-- timestamp: timestamp (nullable = true)
|-- timestampType: integer (nullable = true)
>>>
```

`value` это всегда либо бинарный код, либо строка.
Определяем структуру данных нашего исходного датасета.
schema = StructType() \
 .add("time_id", StringType()) \
 .add("ping_ms", StringType()) \
 .add("temperature_c", StringType()) \
 .add("humidity_p", StringType())

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

student898_2@bigdataanalytics-worker-3:~
student898_2@bigdataanalytics-worker-3:~
>>> schema = StructType() \
...     .add("time_id", StringType()) \
...     .add("ping_ms", StringType()) \
...     .add("temperature_c", StringType()) \
...     .add("humidity_p", StringType())
>>>
```

value_data = raw_data \
 .select(F.from_json(F.col("value").cast("String"), schema).alias("value"), "offset")

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

student898_2@bigdataanalytics-worker-3:~
student898_2@bigdataanalytics-worker-3:~
...     .add("ping_ms", StringType()) \
...     .add("temperature_c", StringType()) \
...     .add("humidity_p", StringType())
>>> value_data = raw_data \
...     .select(F.from_json(F.col("value").cast("String"), schema).alias("value"), "offset")
>>>
```

value_data.printSchema()

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~
>>> value_data.printSchema()
root
|-- value: struct (nullable = true)
|   |-- time_id: string (nullable = true)
|   |-- ping_ms: string (nullable = true)
|   |-- temperature_c: string (nullable = true)
|   |-- humidity_p: string (nullable = true)
|-- offset: long (nullable = true)
>>>
```

```
parsed_data = value_data.select("value.*", "offset")
parsed_data.printSchema()
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~
>>> parsed_data = value_data.select("value.*", "offset")
>>> parsed_data.printSchema()
root
|-- time_id: string (nullable = true)
|-- ping_ms: string (nullable = true)
|-- temperature_c: string (nullable = true)
|-- humidity_p: string (nullable = true)
|-- offset: long (nullable = true)
>>>
```

```
extended_data = parsed_data.withColumn('test', F.col("humidity_p") / F.col("temperature_c"))
extended_data.printSchema()
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка
student898_2@bigdataanalytics-worker-3:~
>>> extended_data = parsed_data.withColumn('test', F.col("humidity_p") / F.col("temperature_c"))
>>> extended_data.printSchema()
root
|-- time_id: string (nullable = true)
|-- ping_ms: string (nullable = true)
|-- temperature_c: string (nullable = true)
|-- humidity_p: string (nullable = true)
|-- offset: long (nullable = true)
|-- test: double (nullable = true)
>>>
```

```
out = console_output(extended_data, 30)
out.stop()
```

```
student898_2@bigdataanalytics-worker-3:~ - Терминал
Файл  Правка  Вид  Терминал  Вкладки  Справка

student898_2@bigdataanalytics-worker-3:~
student898_2@bigdataanalytics-worker-3:~

+-----+-----+-----+-----+-----+
|time_id|ping_ms|temperature_c|humidity_p|offset|test|
+-----+-----+-----+-----+-----+
|  null|  null|      null|    null|    0|null|
|  null|  null|      null|    null|    1|null|
|  null|  null|      null|    null|    2|null|
|  null|  null|      null|    null|    3|null|
|  null|  null|      null|    null|    4|null|
+-----+-----+-----+-----+-----+

out.stop()-----
Batch: 1
-----
+-----+-----+-----+-----+-----+
|time_id|ping_ms|temperature_c|humidity_p|offset|test|
+-----+-----+-----+-----+-----+
|  null|  null|      null|    null|    5|null|
|  null|  null|      null|    null|    6|null|
|  null|  null|      null|    null|    7|null|
|  null|  null|      null|    null|    8|null|
|  null|  null|      null|    null|    9|null|
+-----+-----+-----+-----+-----+

>>> █
```

2.file source.py + 3. kafka source.py запускаются одной командой sparksubmit
применить, выполнить она повторяет эти команды не из консоли, а просто одной командой sparksubmit что
то, что то, что то файл который скину, т. е. Запустить на прямую
Команда для запуска файлов Spark-Submit
spark-submit —master spark://VirtualBox:7077 ~/3.kafka_source.py ~/2.file_source.py`