For this exercise, you have access to a dataset containing sales transactions from distributors to points of sales. The format of the dataset is the following:

Column	Type	Description	Example
`DISTRIBUTOR_ID`	String	The unique identifier of the distributor that is at the origin of the sales transaction	*dist_a*
`POS_ID`	String	The unique identifier of the point of sales that is the destination of the sales transaction	*pos_1*
`VALUE`	Number	The currency value of the transaction	*5*

We would like you to simulate live sales events by sending one line of the dataset, serialised with Avro, to a Kafka stream every *n* seconds, where *n* is a number randomly generated between 1 and 5. We then would like you to use the Java or Scala API (chose what suits you the best) of Spark for Structured Streaming to process the stream over a window of 3 seconds. At every (micro-)batch, we expect you to calculate and display the following information for each distributor who has been updated by the last batch:

- the total sales of the distributor
- the delta with his previous total value in percentage

Ideally, we also would like you to output the results into a csv file and to make the stream fault-tolerant (if you kill the spark process and restart it, it should pick up where it left off).