University of Abdelhamid Mehri-Constantine2
Faculty: NTIC
Department: IFA
Master's second year SDIA 2024/2025

Lab 5 BDPA
Spark Structured Streaming

Spark Structured Streaming is a scalable and fault-tolerant stream processing engine built on top of the Spark SQL API. In this lab, we illustrate this by using structured streaming.
We want to display the top scoring teams for each year in a streaming manner using **EPL-Historical-Data** dataset of lab03.

1. Create a directory `input_directory` that will contain all the necessary files for streaming;
2. We need to get the file schema in order to create the streaming version of the DataFrame. Read the `epl-training.csv` file and extract his schema.
3. Use this schema to create the streaming DataFrame as follows :

```
stream = spark.readStream.schema(<your_schema>)
      .option("maxFilesPerTrigger",1)
      .csv(input_directory)
```

4. Perform all the needed columns preparation (e.g., castings) ;
5. Create the aggregation on the streamed data that answers question 4 from Lab03
6. Create the streaming query :

```
query = <your_aggregation>.writeStream.queryName("outputs") \
      .outputMode("complete") \
      .format("memory") \
      .start()
```

   This instruction will create a table called "`outputs`" in which we can perform spark.sql queries. for this step, we need the presence of at least one file in the `input_directory` so we can test our streaming query;
7. In this step, we need to split the original csv file (`epl-training.csv`) into several files as follows : Each file must contain the samples recorded for one year (i.e., one file for each year);
8. Rerun the instruction of step 6, make sure to avoid conflicts, you can stop the spark session for every try.
9. Show the resulting table for each file using `spark.sql query` in "`outputs`" tables.