PySpark Q&A

1. How to read avro files

df = spark.read.format("avro").load("examples/src/main/resources/users.avro")

df.select("name", "favorite\_color").write.format("avro").save("namesAndFavColors.avro")

1. How to use parquet file in pyspark?

peopleDF = spark.read.json("examples/src/main/resources/people.json")

# DataFrames can be saved as Parquet files, maintaining the schema information.

peopleDF.write.parquet("people.parquet")

# Read in the Parquet file created above.

# Parquet files are self-describing so the schema is preserved.

# The result of loading a parquet file is also a DataFrame.

parquetFile = spark.read.parquet("people.parquet")

# Parquet files can also be used to create a temporary view and then used in SQL statements.

parquetFile.createOrReplaceTempView("parquetFile")

teenagers = spark.sql("SELECT name FROM parquetFile WHERE age >= 13 AND age <= 19")

teenagers.show()

1. How do you read json files?

Spark.read.json() however, each line must be a valid json object

1. When to use parquet files over avro?

Parquet is a columnar format which means columns are stored in sequential blocks rather than a record, hence retrieving columns at once becomes faster than any row based file format in which all records will be loaded, parsed for columns and then columns will be returned.

1. How do you write spark dataframe to redshift database ?

Df.write.

format(“com.databricks.spark.redshift”)

.option(“url”,”redshift\_url”)

.option(“dbtable”,”table\_name”)

.option(“driver”, “com.amazon.redshift.jdbc42.Driver”)

.option(“tempdir”,”temp\_out\_dir”)

.option(“aws\_iam\_role”,”role”)

.option(“tempformat”, “CSV GZIP”)

.option(“csv\_separator”, “,”)

.option(“preactions”,”actions to be performed before writin to the table”)

.option(“postactions”, ”actions to be performed after writin to the table”)

.mode(“append”)

.save()

1. What is the default format in which data is written in temp directory while writing to redhisft?

Default is AVRO, but CSV is faster.

1. How have you used spark dataframes in your project?

We had stored sql queries and config details in the database as key and value columns, to read that we used pandas read\_sql function and stored results in a var lets call it sqldetails.

Post that we used sd=spark.createDataFrame(sqldetails) and then iterated(row) over it to set those values in the environment variables using sd.rdd.toLocalIterator() and os.environ[row.key]=row.value

1. Which property would you use to avoid using show on each dataframe in jupyter?

Set the param spark.sql.repl.eagerEval.enabled to True and set the value of spark.sql.repl.eagerEval.maxNumRows to control the number of rows.

1. How do you fetch rows based on some conditions?

Df.filter(df.a==’abc’)

1. Have you used pandas UDFs?

No, but to use those we need to use the decorator @pandas\_udf or to apply pandas function defined in your program, you can use mapInPandas or applyInPandas on your spark dataframe.

1. Difference between mapInPandas and applyInPandas.

Apply works on grouped data, map yields generators .

1. Difference between merge and merge\_asof

Merge is used to perform inner join merge\_asof for left join and match is done based on nearest keys rather than exact keys

1. Can you use pandas udf in spark sql

Yes, for that you need to register the udf with

Spark.udf.register

1. DataFrame and Spark SQL share the same execution engine so they can be interchangeably used seamlessly. Give an example of this

Df.createOrReplaceTempView(“tablename”)

Now you can run spark.sql on this tablename.

1. Difference between coalesce and repartition

Repartition can be used to increase or decrease the number of partitions, whereas coalesce can only be used to decrease.

1. How can you sort on a dataframe?

Use df.sort or df.orderby, asc for ascending and desc otherwise. Default is asc.

1. How can you use pyspark sql functions?

By importing **pyspark.sql.functions**

Spark Q&A

1. What all is required to get spark up and running?

Python – used anaconda distribution for that.

JDK 8.0 – spark not available for 9+

Apache spark – from Apache.com

1. Sparkcontext object is automatically created in the pyspark or scala shell.
2. What is Maven?

Maven is a popular package management tool for Java-based languages that lets you link to libraries in public repositories.

1. How do you run python standalone applications for spark?

By using spark-submit script which includes spark dependencies in Python.

1. How do you create a spark session?

**from** **pyspark.sql** **import** SparkSession

spark = SparkSession.builder.getOrCreate()

1. How can you control the number of rows that you can show on a jupyter notebook with pyspark dataframe?

By using variables like :

spark.conf.set('spark.sql.repl.eagerEval.enabled', True)

This will set the eager evaluation to true and when the following value is set to a specific number only those number of lines will be shown.

spark.sql.repl.eagerEval.maxNumRows

1. How to collect the dataframe data from all executors at the driver?

Use df.collect()

1. Convert a spark df to pandas df

Df.toPandas()

1. How can you create a new column in existing df?

Use the withColumn function as below:

df.withColumn('upper\_c', upper(df.c)).show()

1. How to apply conditions to filter data from a dataframe?

df.fliter(df.a == 1).show()

1. How can you run pandas functions directly on your spark dataframes?

Use mapInPandas() function

def pandas\_filter\_func(iterator):

for pandas\_df in iterator:

yield pandas\_df[pandas\_df.a == 1]

df.mapInPandas(pandas\_filter\_func, schema=df.schema).show()

1. kjads