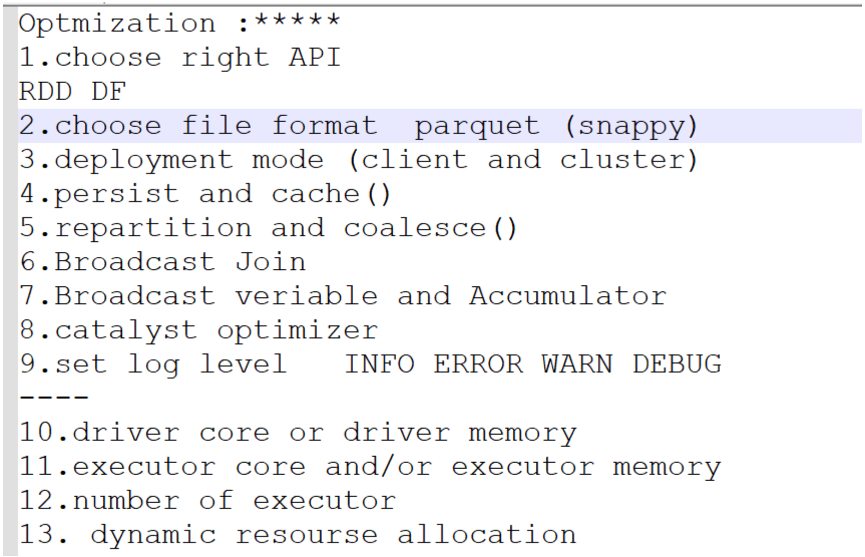
# Spark | Day - 21 | 01/12/23

## Important for the Interview

===============================================================

**How to Optimize work in Spark and what are the deciding factors ↓**



```
Optmization :*****
1.choose right API
RDD DF
2.choose file format  parquet (snappy)
3.deployment mode (client and cluster)
4.persist and cache()
5.repartition and coalesce()
6.Broadcast Join
7.Broadcast veriable and Accumulator
8.catalyst optimizer
9.set log level   INFO ERROR WARN DEBUG
----
10.driver core or driver memory
11.executor core and/or executor memory
12.number of executor
13. dynamic resourse allocation
```

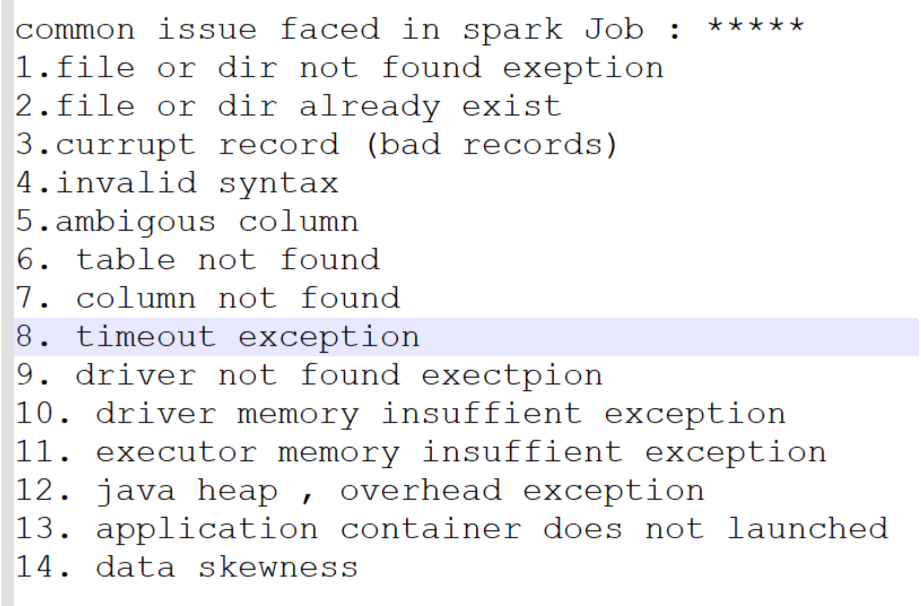
If the data is Structured or Semi Structured then we can go with DF and if the data is in Unstructured then – RDD

Parquet File format has default file compression technique as snappy

From point 10) here using some techniques we can control the Spark Job performance at run time.

===========================

**common issue faced in spark Job →**



```
common issue faced in spark Job : *****
1.file or dir not found exeption
2.file or dir already exist
3.currupt record (bad records)
4.invalid syntax
5.ambigous column
6. table not found
7. column not found
8. timeout exception
9. driver not found exectpion
10. driver memory insuffient exception
11. executor memory insuffient exception
12. java heap , overhead exception
13. application container does not launched
14. data skewness
```

**When we get Timeout exception → When we are trying to write the data in Hive and if Hive is down there then we can get this error (Google for more)**

There is one more approach where can exactly estimate how much memory is required to run the Spark Job →

- This can be achieved by the "**Dry Run Method**". For this we need to submit our Spark job with Memory 1 GB and Drive memory as 1 GB.
- By this the job will fail but it will show how much memory it is expecting to complete the job

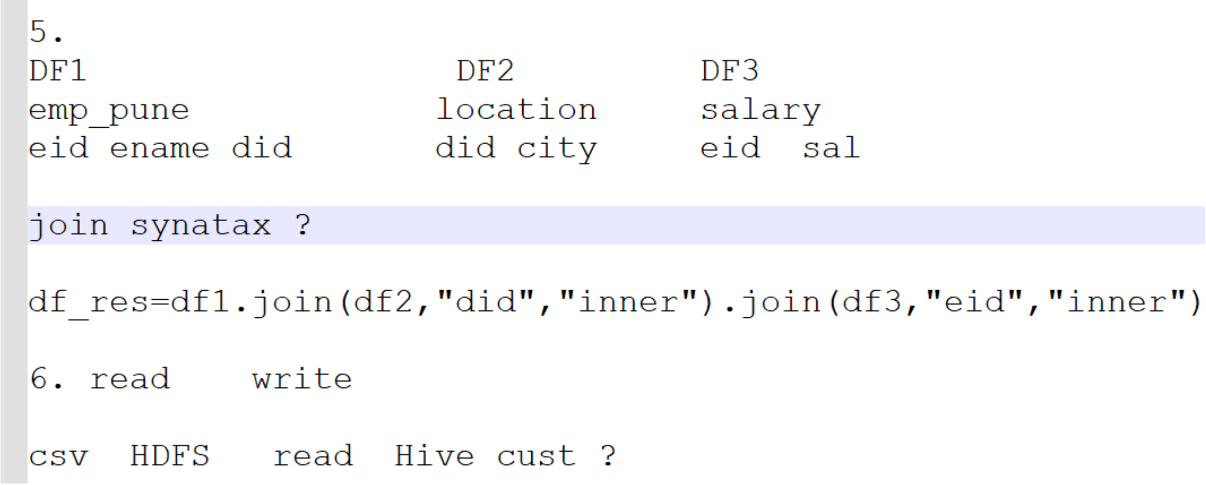
Executor memory is divided into few parts like execution, user and free memory. So, if the user memory is insufficient that time we can get the Java Heap or Java Overhead error.

**Application container does not launched →** Whenever there is no resources available on the cluster and the job is in waiting state since long then it will fail with this error.

==========================================

**Scenarios based theoretical questions in Spark →**

1) When can we use withColumn function?
2) Supppose we have 2 data frames and we need to understand what is the data present in 1st data Frame that is not present in 2nd DF ?
   → Here we need to use exceptAll function → **df1.exceptAll(df2)**
3) If we want to drop the null value then → **use dropna**
4) To fill the null values → **then fillna**
5)



```
5.
DF1                  DF2        DF3
emp_pune            location    salary
eid ename did       did city    eid  sal

join synatax ?

df_res=df1.join(df2,"did","inner").join(df3,"eid","inner")

6. read    write

csv  HDFS   read  Hive cust ?
```

6) Read the file in one format from HDFS → Process it using the Spark → store it in Hive (So they will ask to write the program)

Go Google for more questions?