A few (Questions) enquiries around Spark and ETL

1. What is the best way to store a large file for processing with Spark?
2. How can I copy a CSV file onto S3 in Parquet format?
3. How do you handle errors when copying to S3 in this process?
4. What are a couple of techniques for idempotency of S3 Parquet writes?
5. What is the difference between a file, object, and a row in S3?
6. Can you give a few examples of "files" and "folders" in S3 where folders are part of the key?
7. List my questions so far.
8. If processing a file in Spark requires an API call for each record, how do you go about doing it? Or if that is expensive, how do you avoid it?
9. What is partitioning the data in Spark? Can you give a few examples?
10. How are these partitions used during processing?
11. If I have only 1 partition, is it processed on only one node?
12. What is memoization?
13. How does Spark caching work?
14. What are the best practices for storing Spark results in an RDBMS?
15. When storing the results of Spark in a database, how do you handle commits and rollbacks?
16. How are commits and rollbacks handled when storing an RDD in a relational database, and each record needs an explicit commit?
17. What are the best practices for storing Spark results in an RDBMS?
18. Specifically, when you store an RDD in a relational database, what is the best approach if each record needs an explicit commit?
19. How do I handle storing errors during ETL operations using Spark, and how do I report and handle these errors afterward?
20. If you found that some records are in error while ETL via Spark, what is the best way to handle and report those errors and handling them afterward? Do you store them in a database? Can you use a UI to handle those records?
21. If processing a file in Spark requires an API call for each record, how do you go about doing it? Or if that is expensive, how do you avoid it?
22. What is partitioning the data in Spark? Can you give a few examples?
23. How are these partitions used during processing?
24. If I have only 1 partition, is it processed on only one node?
25. What is memoization?
26. How does Spark caching work?
27. Can you list my questions one more time please?
28. Can you list them without their categorization?
29. If you found that some records are in error while ETL via Spark, what is the best way to handle and report those errors and handling them afterward? Do you store them in a database? Can you use a UI to handle those records?
30. Does the "foreachPartition" commit or rollback the entire "partition"?
31. What if the entire RDD needs to be in the same file on S3?
32. When multiple partitions are used to write an RDD, is the eventual file has one name or multiple names?
33. How do I read a file that is in multiple partitions on S3 into a partitioned RDD in Spark?