1. How do you create a DataFrame in PySpark?

Answer:

```
from pyspark.sql import SparkSession

spark = SparkSession.builder.appName("DataFrameExample").getOrCreate()

data = [("Alice", 1), ("Bob", 2), ("Cathy", 3)]

columns = ["Name", "Id"]

df = spark.createDataFrame(data, columns)

df.show()
```

- This creates a DataFrame with two columns: "Name" and "Id."

2. How do you filter rows in a DataFrame?

Answer:

```
filtered_df = df.filter(df.Id > 1)

filtered_df.show()
```

- This filters rows where the "Id" column is greater than 1.

3. How do you group and aggregate data in PySpark?

Answer:

```
df.groupBy("Name").count().show()
```

- This groups the DataFrame by the "Name" column and counts the occurrences of each name.

4. What is the difference between select and selectExpr in PySpark?

- select allows you to select columns or expressions directly, e.g., df.select("column1").
- selectExpr allows SQL expressions, e.g., df.selectExpr("column1 as c1", "column2 * 2 as double_col").

5. How can you add a new column to a DataFrame?

Answer:

```
df = df.withColumn("New_Column", df["Id"] * 2)

df.show()
```

- This adds a new column called "New_Column" with values as twice the "Id" values.

6. How do you remove duplicates in a DataFrame?

Answer:

df.dropDuplicates().show()

- This removes duplicate rows based on all columns.

7. How do you join two DataFrames?

Answer:

df1 = spark.createDataFrame([("Alice", 1), ("Bob", 2)], ["Name", "Id"])

df2 = spark.createDataFrame([("Alice", "F"), ("Bob", "M")], ["Name", "Gender"])

joined_df = df1.join(df2, on="Name", how="inner")

joined_df.show()

- This joins df1 and df2 on the "Name" column with an inner join.

8. How do you read and write a CSV file in PySpark?

Answer:

df = spark.read.csv("file_path.csv", header=True, inferSchema=True)

df.write.csv("output_path.csv", header=True)

- This reads a CSV file into a DataFrame and writes the DataFrame back to a CSV file.

9. Explain how to cache a DataFrame in PySpark.

Answer:

df.cache()

df.show()

- cache() persists the DataFrame in memory for faster access when reused multiple times.

10. How can you convert a DataFrame to an RDD and vice-versa?

Answer:

DataFrame to RDD

rdd = df.rdd

```
# RDD to DataFrame
new_df = rdd.toDF()
```

- This shows how to switch between DataFrame and RDD formats in PySpark.

PySpark DataFrame Operations

1. **How do you create a DataFrame from an RDD or a list of tuples in PySpark?**
2. **How do you add, rename, and drop columns in a PySpark DataFrame?**
3. **What is the difference between select(), filter(), and where() methods in PySpark?**
4. **How do you perform aggregations, such as sum(), avg(), and count() on a DataFrame?**
5. **Explain how to use window functions like row_number(), rank(), and dense_rank() in PySpark.**

Data Processing and Transformation

6. **How can you join two DataFrames in PySpark? What are the different types of joins available?**
7. **How do you handle missing or null values in a DataFrame?**
8. **How do you group and aggregate data using groupBy() and agg() in PySpark?**
9. **What is the difference between map() and flatMap() transformations when using RDDs?**
10. **Explain how to filter and sort data in a DataFrame.**

Performance Optimization

11. **What is the purpose of caching a DataFrame, and how do you use it?**
12. **How do you repartition and coalesce a DataFrame, and what's the difference between the two?**
13. **Explain the concept of broadcast join and when to use it in PySpark.**
14. **How do you monitor and optimize the performance of a PySpark application?**

15. **What are the advantages of using Spark SQL over RDD operations for processing structured data?**

File Formats and Storage

16. **How do you read and write data in different file formats (e.g., CSV, Parquet, JSON) using PySpark?**
17. **What is the difference between reading a file as a textFile vs. using a DataFrame API in PySpark?**
18. **Explain how you can read data from and write data to a Hive table using PySpark.**
19. **How do you handle schema inference while reading JSON and Parquet files in PySpark?**
20. **What are the best practices for handling large datasets in PySpark?**

DataFrame Basics

1. **How do you create a DataFrame from an RDD or a list of tuples in PySpark?**
2. **How do you display the schema and the first few rows of a DataFrame?**
    - Code: df.printSchema() and df.show()
3. **What is the difference between select(), selectExpr(), and withColumn() in PySpark?**
4. **How do you rename columns in a DataFrame?**
    - Example: df.withColumnRenamed("old_name", "new_name")

Data Transformation and Filtering

5. **How do you filter rows in a DataFrame based on multiple conditions?**
6. **How can you create a new column based on the transformation of an existing column?**
7. **How do you drop a column or multiple columns from a DataFrame?**
    - Code: df.drop("column_name")
8. **How do you sort a DataFrame based on a column in ascending and descending order?**
    - Code: df.orderBy("column_name", ascending=False)
9. **Explain the difference between distinct() and dropDuplicates().**

Grouping, Aggregation, and Joins

10. **How do you group data and calculate aggregate statistics using groupBy()?**
11. **What is the purpose of agg() in PySpark? Give an example of using it with multiple aggregation functions.**
12. **How do you perform an inner, left, right, and full outer join in PySpark?**
13. **How can you join two DataFrames on multiple columns?**
14. **What is the difference between groupBy() and rollup() in PySpark?**

Window Functions

15. **How do you use window functions like row_number(), rank(), and dense_rank() in PySpark?**
16. **Explain the purpose of lead() and lag() functions. How do you use them in a DataFrame?**
17. **How can you calculate running totals using window functions in PySpark?**

Data Handling and Cleaning

18. **How do you handle missing or null values in a DataFrame?**

- Examples: dropna(), fillna(), na.replace()

19. **What is the difference between filter() and where() methods in PySpark?**
20. **How do you handle columns with complex data types like arrays, maps, or structs in PySpark?**
21. **Explain how to use the explode() function for flattening arrays in a DataFrame.**

Performance Optimization

22. **What is caching, and how do you cache a DataFrame in PySpark?**
23. **How do you repartition a DataFrame? Explain the difference between repartition() and coalesce().**
24. **What is a broadcast join, and when should you use it?**
25. **How do you use the persist() method, and what are its different storage levels?**

File Handling and Storage Formats

26. **How do you read and write CSV files in PySpark? Explain the parameters like header, inferSchema, and delimiter.**
27. **What is the difference between reading a file as textFile() and using the DataFrame API in PySpark?**
28. **How do you read and write Parquet files in PySpark? Why is Parquet often preferred for large datasets?**
29. **Explain how you can read from and write to a Hive table using PySpark.**
30. **How do you read data from and write data to an S3 bucket using PySpark?**

Advanced Transformations

31. **How do you pivot a DataFrame using the pivot() function in PySpark?**
32. **What is the purpose of unpivot() or using melt() in PySpark?**
33. **How do you union two DataFrames, and what considerations should you make when using union() or unionByName()?**
34. **How can you convert a PySpark DataFrame to a Pandas DataFrame, and what are the limitations?**

UDFs (User-Defined Functions)

35. **What is a UDF, and how do you define and register one in PySpark?**
36. **How can you use pandas_udf() for vectorized operations in PySpark?**
37. **What are the performance implications of using UDFs, and how can you optimize them?**

Data Conversion and Interoperability

38. **How do you convert an RDD to a DataFrame and vice versa?**
39. **How do you convert a DataFrame to an RDD and perform operations on it using lambda functions?**
40. **How can you save a DataFrame as a global temporary view and query it using Spark SQL?**

- Example: df.createOrReplaceGlobalTempView("view_name")

File Formats and Schema Inference

41. **How does schema inference work in PySpark when reading JSON or Parquet files?**
42. **How do you define a schema manually when reading a file in PySpark?**

43. **What are the differences between reading a file as a DataFrame and as an RDD, and when would you choose each approach?**

Advanced Performance Techniques

44. **What is a Tungsten engine, and how does it optimize Spark performance?**
45. **How do you manage skewed data in Spark, and what techniques can you use to optimize joins involving skewed data?**
46. **Explain the purpose of Catalyst Optimizer and how it helps improve query performance.**

Miscellaneous PySpark Operations

47. **How do you save a DataFrame to different formats like ORC, Avro, or Delta Lake?**
48. **How do you handle nested columns (structs) when working with JSON files?**
49. **Explain how to use crossJoin() and when you would use it.**
50. **How do you configure and manage Spark sessions and application parameters in PySpark?**