**Capgemini Data Engineer Interview Guide – Experienced 3+**

**Round 1: Technical Round**

1. **Introduction and Technology Overview**

 Tell me about yourself and your professional background.

 List all the technologies you have worked on in your project (e.g., Spark, Hadoop, Hive, Databricks).

2. **Spark Architecture**

Explain the architecture of Spark, including its components such as driver, executor, and cluster manager.

3. **Cluster Configuration**

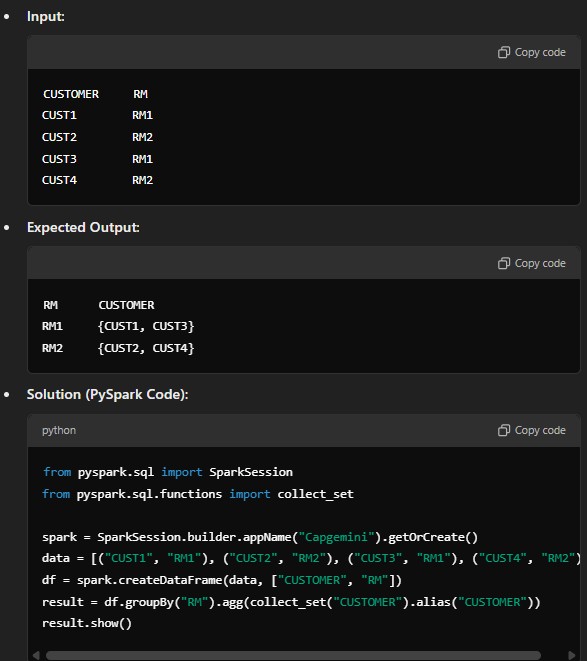
Describe the cluster configuration used in your project, including memory allocation, number of nodes, and executor/driver settings.

4. **Spark Version**

Which Spark version are you using in your project, and why did you choose it?

5. **PySpark Transformation**

Solve the following dataset transformation:



6. **Spark Optimization Techniques**

 Cache() vs Persist(): Explain the difference and use cases for caching and persisting data in Spark with memory levels.

 map() vs mapPartitions(): Highlight the difference between map (row-level transformation) and mapPartitions (partition-level transformation).

 Adaptive Query Execution (AQE): Discuss how AQE optimizes query execution in Spark dynamically based on runtime stats.

 repartition() vs coalesce(): Explain when to use repartition() (increases partitions) vs coalesce() (reduces partitions).

7. **SQL and Hive Questions**

 Window Function: Solve a problem using a window function in Spark or SQL.

 Indexing in SQL: How does indexing improve query performance?

 Managed vs External Tables in Hive: Explain the difference and when to use each.

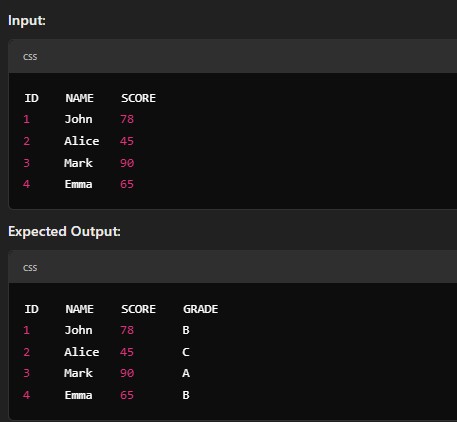
**Round 2: Technical + Managerial Round**

1. **UDF in PySpark**

 Define what a **User-Defined Function (UDF)** is and how to register it in

PySpark.

 Solve the following problem using a UDF:



**Solution (PySpark Code):**

from pyspark.sql.functions import udf

from pyspark.sql.types import StringType

def grade(score):

if score >= 80:

return "A"

elif 60 <= score < 80:

return "B" else:

return "C"

grade\_udf = udf(grade, StringType())

df = spark.createDataFrame([(1, "John", 78), (2, "Alice", 45), (3, "Mark", 90), (4, "Emma",

65)], ["ID", "NAME", "SCORE"])

result = df.withColumn("GRADE", grade\_udf(df["SCORE"]))

result.show()

2. **Experience-Based Questions**

 Explain the projects you have worked on, focusing on challenges and solutions you implemented.

 Discuss how you handled null values or unstructured data in your previous projects.

 Talk about your approach to deploying pipelines from development to

production.

3. **Delta Lake Concepts**

 Explain Delta Lakehouse architecture and its advantages.

 Discuss the Bronze, Silver, and Gold layers in the Delta Lake pipeline.

4. **Databricks Questions**

 How do you create a job cluster in Databricks?

 Explain the use of dbutils functions in Databricks.

 Discuss the process of moving files in Databricks File System (DBFS).

5. **Scala Traits and Azure Integration**

 Define traits in Scala and their applications in your project.

 Discuss how you integrated Azure services into your Spark application.

6. **Performance Optimization**

 What performance optimization techniques have you applied in Spark, Sqoop, or Databricks?

 Explain lazy evaluation in Spark and how it impacts performance.

7. **Project Management**

 How do you handle team coordination and deadlines in complex projects?

 Provide an example of a critical decision you made in a project and its impact.

**Summary**

 **Round 1** focused on core technical skills, including Spark architecture, transformations, optimizations, SQL, and Hive.

 **Round 2** tested advanced PySpark concepts, real-world problem-solving with UDFs,

Databricks, Delta Lake, and experience-based managerial scenarios.