Citi Bank Big Data Developer Interview Guide – Experienced 3+

Technical Round 1

1. Introduction to the project and tech stacks used

Introduce your recent project, explaining its goal, architecture, tools, and technologies. Mention tools like Hadoop, Spark, Hive, Oozie, AWS, or Azure, and describe how these were used for data processing, storage, and analytics.

2. Teradata to Hadoop migration and handling data with SCD Type 2

Teradata to Hadoop migration involves exporting data from Teradata using tools like Sqoop or custom ETL processes and loading it into HDFS or Hive.
SCD Type 2 (Slowly Changing Dimension) is handled using versioning or effective date columns. Insert a new row for each change while retaining historical records.

3. Oozie workflow files (how many used)

Workflow files define actions like Hive, Spark, or shell scripts. Depending on project complexity, multiple workflows can be linked. Mention how many were used in your project for different ETL jobs.

4. Oozie join condition

Oozie **joins** synchronize multiple control flows in a workflow using <join> and <fork> tags to manage parallel task execution.

5. How is Oozie called?

Oozie can be called using:

- Command-line interface (CLI): oozie job -oozie <URL> -config job.properties -run
- REST API: Use HTTP calls to submit jobs.

6. How to view Oozie jobs?

Use:

- Oozie Web Console to view job status and logs.
- Command-line: oozie jobs or oozie job -info <job_id>.

7. Shell commands for renaming a file

mv old_filename new_filename

8. Shell: change permissions

chmod 755 filename – Sets read, write, and execute permissions.

9. Shell: how to run jobs/scripts in the background

Add & at the end:

bash

Copy code

bash script.sh &

10. Shell: command to check processes running in the background

jobs – Lists background jobs.
ps -aux – Shows all processes.

11. Using shell, how to find the difference between two files

diff file1 file2 – Shows line-by-line differences.

12. What type of wrapper is used, or which language is used?

Most workflows use shell scripts or Python wrappers to automate Oozie and Spark jobs.

13. Amazon Deequ usage and what sort of quality checks are done using it

Amazon Deequ performs data quality checks like:

- Uniqueness of values in a column.
- Completeness (no nulls).
- Freshness of data.

14. Spark repartition vs. coalesce

- repartition() increases or decreases partitions, causing a shuffle.
- coalesce() reduces partitions without a shuffle, suitable for optimization when decreasing partition count.
- Use repartition for better data distribution and coalesce for reducing overhead.

15. Bucketing vs. partitioning

- Partitioning splits data by columns into directories.
- Bucketing divides data into fixed-size buckets for better join performance.

16. How to handle data skewness

- Use salting by adding a random key.
- Repartition with balanced keys.
- Apply broadcast joins for small reference tables.

17. An existing job running longer suddenly: how to analyze the issue

- Check data size increase.
- Look for skewed data or straggler tasks.
- Analyze shuffle operations or GC overhead.

18. Given 1TB of a file, how to check word count

Use Spark:

python

Copy code

```
spark.read.text("file.txt").flatMap(lambda line: line.split()).count()
```

19. Usage of UDFs

UDFs (User-Defined Functions) allow custom transformations in Spark or Hive when built-in functions are insufficient.

20. Have you worked on any window functions?

Explain usage for running totals, ranks, or lead-lag functions in Spark.

21. Methods to avoid duplicates in PySpark/Scala

- Use .distinct() or .dropDuplicates().
- Implement **deduplication logic** using primary keys.

22. Partitioning a table with card details and transactions

Partition by date or region; for age, create buckets for ranges like 18-25, 26-35.

23. Agile methodologies used

Discuss sprint planning, daily stand-ups, and retrospectives.

24. What is a broadcast join?

A broadcast join replicates a small dataset to all nodes, avoiding shuffle.

25. CI/CD-related questions

Explain how Git, Jenkins, and Docker automate deployment.

26. Code automation

Discuss using Airflow or Oozie for scheduling and managing workflows.

27. Sprint duration and casual discussion

Typical sprints last 2 weeks.

28. Is GitHub cloud-based in the project?

Most projects use cloud-hosted Git platforms like GitHub or Bitbucket.

29. Cloud-based tools used

Examples: AWS S3, Azure Data Factory, Google Cloud Storage.

30. File formats used

- Parquet/ORC for analytics.
- CSV/JSON for raw data.

31. File format for fetching all data

Parquet or ORC – Columnar and highly compressed.

32. File format for fetching a few columns

Parquet/ORC allow efficient column pruning.

Technical Round 2

PySpark

1. Explain the difference between repartition() and coalesce() in PySpark.
2. What are broadcast variables, and when would you use them in Spark?
3. Explain the concept of RDD, DataFrame, and Dataset in PySpark.
4. How do you handle data skewness in a PySpark job?
5. What is the difference between SparkSession and SparkContext?
6. Describe the significance of lazy evaluation in Spark.
7. How do you optimize a join operation in Spark for large datasets?
8. Explain the concept of checkpointing in Spark and why it is important.

Kafka

1. What is the role of a partition in Kafka, and how does it impact scalability?
2. Explain the concept of consumer groups in Kafka. How do they affect message processing?
3. Describe how Kafka ensures data durability and fault tolerance.
4. What is a Kafka topic, and how do you choose the number of partitions for it?
5. How would you design a Kafka-based pipeline for processing streaming data in real-time?

Airflow

1. What is a DAG in Apache Airflow, and how is it used for scheduling workflows?
2. Explain the difference between TriggerDagRunOperator and ExternalTaskSensor in Airflow.
3. How do you handle failures in Airflow tasks, and what retry strategies can you use?
4. Describe how to pass data between tasks in Airflow using XComs.

Projects and Architecture

1. Describe an end-to-end data pipeline project you worked on, highlighting your role and the technologies used.
2. What challenges did you face when scaling your data processing pipeline, and how did you overcome them?
3. How do you ensure data quality and consistency across different stages of a data pipeline?

Glassdoor Citi Bank Review –

https://www.glassdoor.co.in/Reviews/Citi-Reviews-E8843.htm

Citi Bank Careers –

https://jobs.citi.com/

Subscribe to my YouTube Channel for Free Data Engineering Content –

https://www.youtube.com/@shubhamwadekar27

Connect with me here –

https://bento.me/shubhamwadekar

Checkout more Interview Preparation Material on –

https://topmate.io/shubham_wadekar