Amazon

Data Engineer Interview Questions (0-3 years) 25 LPA

DSA + SQL

1. Write a query to find the top 3 products with the highest revenue growth compared to the previous month.
2. Find the top 3 products with the steepest positive sales trend over the last 90 days.
3. Flag days where a product's revenue deviates by ±200% from its 7-day moving average.
4. Identify out-of-order events where a user's event arrives more than 1 hour late.
5. How do you optimize a query that's performing poorly?
6. When would you choose star schema vs snowflake schema in data modeling?
7. Optimise a query aggregating product revenue over 6 months on a billions-row Redshift table.
8. DSA: Given an array of non-negative integers representing money in houses, compute the maximum amount that can be robbed without robbing adjacent houses.

Python + DSA

1. **Design a Python service that performs real-time deduplication of clickstream events using a sliding time window.**

2. **Write a Python function to detect if a directed graph contains a cycle using DFS.**
3. **Implement a token bucket rate limiter for API calls**
4. **Explain time complexity of different Python data structures (dict, set, list).**
5. **How do you handle memory management in Python?**
6. **Question on Python context manager.**
7. **Some Leadership principles questions.**
8. **Dimensional modeling questions. Type of SCDs, when to use which.**

Big Data + Spark

1. How would you join two skewed datasets in Spark to avoid stragglers?
2. How would you optimize joins in PySpark (broadcast vs shuffle)?

3. What are the common causes of out-of-memory errors in Spark on EMR, and how would you fix them?
4. How would you use partitioning and bucketing in Spark to optimize joins of large tables?
5. How do you debug and optimize a slow PySpark job?
6. Explain how checkpointing and caching can help optimize iterative Spark workloads.
7. How to handle out of order events in streaming job.
8. Open table formats, when to use which
9. Spark fundamental questions
10. General discussion on projects and work exp

Data Pipeline Design

1. How would you design an event-driven order tracking system that can handle millions of users checking status simultaneously?
2. What are the trade-offs between using Kinesis vs Kafka for ingestion in this architecture?

3. What caching strategy would you use to handle millions of concurrent read requests efficiently?
4. What database design (SQL vs NoSQL) would you choose for storing order status history, and why?
5. How would you ensure exactly-once processing of status events in the pipeline?
6. How to scale each component individually.
7. How to implement data governance and quality.
8. Lot of design related trade offs
9. Some Leadership principles questions