thoughtworks

Data Engineering Interview Questions

Ankita Gulati

Shubh Goyal

Job Details

- **Position:** Data Engineer II
- **Experience:** 4+ years
- **Location:** Bangalore
- **Work mode:** Office
- **Compensation:** ₹25+ LPA
- **Total Rounds:** 4
- **Top Required Skills:**

1. SQL
2. PySpark / Python
3. Cloud Data Engineering
4. ETL / Data Modeling
5. Big Data & Streaming
6. System Design

Ankita Gulati

Shubh Goyal

Round 1
Online Test

1. **Maximum Subarray Sum**

• Problem: Given an integer array, find the contiguous subarray with the maximum sum.

• Expected: Explain and implement Kadane's Algorithm.

• Discuss time complexity O(n) and why it's optimal.

2. **Reverse Words in a String Without Extra Space**

• Problem: Reverse the order of words in a string in-place without using additional memory.

• Example: "I love coding" → "coding love I".

• Edge Cases: Multiple spaces, leading/trailing spaces.

3. **Balanced Parentheses Using Stack**

• Problem: Check if a string containing (), {}, and [] is balanced.

• Example: "({[]})" → Balanced, "({[})" → Not Balanced.

• Expected: Implement using a stack (push/pop), handle edge cases.

Ankita Gulati Shubh Goyal

Round 2
Online Test (DSA & Algorithms)

1. **Longest Consecutive Subsequence**
• Problem: Given an unsorted array, find the length of the longest sequence of consecutive numbers.
• Example: [100, 4, 200, 1, 3, 2] → longest sequence [1, 2, 3, 4] → length = 4.
• Expected: Compare HashSet approach (O(n)) vs brute-force ($O(n^2)$).

2. **Minimum Cost Problem** (Greedy Approach)
• Problem: Minimize total cost when combining items (e.g., ropes problem).
• Expected: Show why greedy algorithm (using min-heap / priority queue) works.
• Discuss time complexity O(n log n).

Ankita Gulati

Shubh Goyal

3. **String Compression** (Run-Length Encoding)

• Problem: Compress a string by replacing consecutive repeating characters with the character and count.

• Example: "aaabbc" → "a3b2c1".

• Edge Cases: Single characters, empty string.

Ankita Gulati

Shubh Goyal

Round 3
Coding & Refactoring

1. **Implement a Discount Rule in Shopping Cart**
• Example rules:
• “10% off if total items > 5.”
• “Buy 2 get 1 free for a specific product.”
• Expected: Design, implement, and test this in a modular and scalable way.

2. **Add Coupon Validation Feature**
• Implement coupon validation: check expiry date, validity, and apply discounts.
• Ensure code is extensible and maintainable.

3. **Identify and Refactor Code Smells**
• Given a legacy code snippet, identify issues such as:
• Duplicate code, long methods, poor variable naming.
• Suggest and implement refactoring improvements.

Ankita Gulati

Shubh Goyal

4. **Interactive Problem Solving**

• Work with interviewers in a collaborative environment.

• Explain thought process clearly while coding.

Ankita Gulati

Shubh Goyal

Round 4
Technical Interview

React & Project Discussion

• Explain React Hooks you've used (useState, useEffect, useContext) and why.
• Difference between functional components vs class components.
• Discuss code flow and architecture of your favorite project.
• Challenges faced in implementing features → how you solved them.

System Design Problem

• Book Recommendation System
→ Design a recommendation system based on user purchase history.
→ Discuss storage of user data, book data, recommendation logic.
→ Ensure scalability and OOP principles are considered.

DSA & OOP Questions

1. Stack and Queue Using Arrays

→ Implement stack (LIFO) and queue (FIFO) using fixed-size arrays.

→ Handle overflow/underflow cases.

2. Singly Linked List Problems

→Reverse a linked list, detect a cycle, or find the middle node.

→Explain time & space complexity.

3. Diamond Problem in Multiple Inheritance

→Explain the diamond inheritance problem in Python.

→Show how MRO (Method Resolution Order) resolves it.

4. Polymorphism Example

→Demonstrate method overriding or operator overloading in Python.

Ankita Gulati Shubh Goyal

5. Difference Between __init__ and __new__

• Explain object creation (__new__) vs initialization (__init__).

6. Execution and Destruction Flow in Inheritance

• Analyze multi-level inheritance → explain constructor & destructor order.

Ankita Gulati

Shubh Goyal

Thank You

Best of luck with your upcoming interviews — you've got this!

Ankita Gulati

Shubh Goyal