**Disnry plus Hotstar Data Engineer Interview Guide – Experienced 3+**

Hotstar is a leading streaming platform known for its large-scale data processing and real- time analytics needs. As a Data Engineer (SDE 2) at Hotstar, you will face challenging problems involving big data technologies, system design, and optimization strategies. Here is a detailed breakdown of interview experiences and sample questions to guide your preparation.

**Common Interview Rounds**

1. **Coding Round**: Focus on data structures, algorithms, and problem-solving.

2. **Low-Level Design (LLD)**: Designing reusable components and modular systems.

3. **High-Level Design (HLD)**: Architecting scalable, distributed data systems.

4. **Coding/Problem-Solving**: Additional complex coding problems.

5. **Hiring Manager (HM) Round**: Behavioral and scenario-based discussions.

**Detailed Interview Preparation**

**1. Coding Round**

**Key Focus Areas**: Array manipulation, dynamic programming, and tree traversal.

**Sample Questions:**

1. Solve [Longest Consecutive Sequence.](https://leetcode.com/problems/longest-consecutive-sequence/description/)

2. Given an n-ary tree, write code to flatten it and store the output in a list.

3. Implement a function to check if a given string has balanced parentheses using a stack.

4. Design an algorithm to merge k sorted lists of video streaming data.

5. Write a function to remove invalid parentheses from a string.

**2. Low-Level Design (LLD)**

**Key Focus Areas**: Object-oriented design, interface design, and modular architecture.

**Sample Scenario:**

1. Develop a **generic user profile system** for Hotstar that:

 Accepts inputs from various teams, such as a propensity score from personalization or a marketing channel.

 Consolidates inputs into a unified user profile.

 Supports daily updates with aggregation methods like min, max, last, and first.

2. Explain how you would implement a caching mechanism for frequently accessed video metadata.

**3. High-Level Design (HLD)**

**Key Focus Areas**: Scalability, fault tolerance, and distributed systems.

**Sample Scenario:**

1. Design a **threaded commenting system** similar to YouTube for Hotstar.

2. Propose an architecture to manage real-time analytics for user interactions during live sports events.

3. How would you design a system to support personalized recommendations at scale?

4. Describe a strategy for implementing a real-time content delivery monitoring system.

5. Architect a solution to handle notifications for millions of users with varying preferences.

**4. Coding/Problem-Solving Round**

**Key Focus Areas**: Advanced coding challenges and optimization.

**Sample Questions:**

1. Solve [Minimum Remove to Make Valid Parentheses.](https://leetcode.com/problems/minimum-remove-to-make-valid-parentheses/description/)

2. Write a function to detect anomalies in streaming data using a sliding window.

3. Implement an algorithm to find the longest common prefix among an array of strings.

4. Optimize a function to calculate moving averages of user engagement.

5. Write a solution to efficiently search a rotated sorted array.

**5. Hiring Manager (HM) Round**

**Key Focus Areas**: Behavioral insights, decision-making, and team dynamics.

**Sample Questions:**

1. Share a time when you explained a technical concept to a non-technical stakeholder.

2. Describe a challenging data-related issue you faced in production and how you resolved it.

3. Discuss a project where you significantly improved the performance of a data pipeline.

4. How do you prioritize tasks when handling multiple projects with tight deadlines?

5. Explain how you stay updated with evolving data engineering technologies.

**Additional Practice Questions**

1. How would you design a logging framework to track errors across multiple services?

2. Compare Kafka and RabbitMQ for real-time message processing in a streaming platform.

3. Explain the benefits of using columnar storage formats like Parquet or ORC.

4. What techniques would you use to ensure data consistency in a distributed database?

5. Describe how partitioning helps improve query performance in a large dataset.

6. How would you build a monitoring dashboard for ETL job failures?

7. Implement a rate-limiter to control API requests per user.

8. Discuss strategies for handling schema evolution in data warehouses.

**Preparation Areas to Focus On**

1. **Coding**: Leetcode medium to hard-level problems.

2. **Low-Level Design**: Reusable design patterns and clean coding principles.

3. **High-Level Design**: Scalable architectures and trade-offs.

4. **Behavioral Skills**: Communication, collaboration, and problem ownership.

By mastering these areas and understanding common patterns, you’ll be ready to impress during your Hotstar Data Engineer interview. Best of luck!