20 DSA Prompts for PDF

Prompt 1: Learn AnyDSAConcept "You areatop-tier DSA tutor. Explain [topic] like I’m 15 using simple intuition, visuals, real-life analogies, and step-by-step breakdowns. Then give 3 coding examples (easy → medium → hard)."

Prompt 2: Fix My Code "You are a FAANG-level engineer. Debug my code, explain the exact issue in detail, outline what caused it, and rewrite an optimized version with clear comments for learning. Code: [PASTE CODE]"

Prompt 3: Break Down the Problem "You are a DSA breakdown expert. Convert this problem into easy step-by-step logic, explain your thought process, analyze edge cases deeply, and provide multiple test cases for clarity. Problem: [PASTE PROBLEM]"

Prompt 4: Give Me Practice Questions "You are a coding mentor. Give me 10 curated DSA questions for my level with patterns, difficulty tags, expected outcomes, and what skill each question strengthens. Level: [BEGINNER/INTERMEDIATE/ADVANCED]"

Prompt 5: Teach Me the Pattern "You are a DSA pattern coach. Explain the underlying pattern, show why it works mathematically, give practical use cases, and share 3 similar problems to reinforce my understanding. Question: [PASTE QUESTION]"

Prompt 6: Simplify Complexity "You are a complexity analyst. Break down the time and space complexity in simple terms, compare it with alternative approaches, identify possible optimizations, and rewrite a more efficient version. Code: [PASTE CODE]"

Prompt 7: Create My DSA Plan "You are my personal coding coach. Create a balanced 30-day DSA plan with daily tasks, topic order, pattern focus, problem sets, checkpoints, and review cycles for consistent progress."

Prompt 8: Explain With Visuals "Use diagrams, flowcharts, and simple illustrations to explain [DSA topic]. Include intuition, cases, and sample problems for clarity."

Prompt 9: Convert Brute Force to Optimized "Take my brute-force solution, explain why it's inefficient, and transform it into an optimized solution using a proper data structure or technique. Code: [PASTE CODE]"

Prompt 10: Compare Similar Concepts "Compare these two DSA concepts: [X] vs [Y]. Explain when to use each, their pros/cons, and show 2 examples for both."

Prompt 11: Teach Me Recursion Better "Explain recursion like I'm a beginner. Show intuition, stack visualization, common patterns, and two problems to practice."

Prompt 12: Build My Intuition "Build intuition for solving [problem type] by breaking down patterns, edge cases, and thought flow before coding."

Prompt 13: Evaluate My DSA Progress "Assess my current DSA level based on this: [describe your skills]. Then give a custom improvement plan."

Prompt 14: Generate Daily Problems "Create a daily set of 3 practice problems (easy, medium, hard) for 14 days focused on [topic/pattern]."

Prompt 15: Mock DSA Interview "Conduct a mock DSA interview. Ask me questions, evaluate my answers, correct mistakes, and guide me toward optimal solutions."

Prompt 16: Simplify Graph Theory "Explain graph basics, BFS, DFS, adjacency lists, and real use cases in simple words with examples."

Prompt 17: Improve My Code Quality "Review my DSA solution for readability, structure, naming, and efficiency. Suggest improvements with explanations. Code: [PASTE CODE]"

Prompt 18: Spot Patterns in Problems "Given this problem set: [LIST], tell me the patterns they fall under and how to approach each group."

Prompt 19: Create DSA Flashcards "Create simple flashcards for important DSA concepts with definitions, examples, and quick memory hacks."

Prompt 20: Build Confidence Before Interviews "Act as my mentor. Help me overcome fear of DSA interviews by giving mindset tips, warm-up problems, and confidence-boosting exercises."