1. Python Mastery:

0. Why Learn Python? Real-World Use Cases

0.1. Python in AI, Web, Automation & DSA

0.2. Installing Python, VS Code, Jupyter, PyCharm

0.3. Creating and Running .py Files

0.4. Python REPL, IDLE & Virtual Environments (venv, virtualenv)

1. Variables & Data Types

1.1. Variables, Memory Allocation, id()

1.2. Constants & Naming Conventions

1.3. Data Types: int, float, bool, str, NoneType, complex

1.4. Type Checking and Conversion

1.5. Number Systems: int, float, complex, decimal, fractions

1.6. Floating-Point Precision Issues

1.7. Deep Dive: math, decimal, fractions

2. Strings in Python

2.1. Creating & Accessing Strings

2.2. Indexing, Slicing & Extended Slicing

2.3. Immutability & Interning

2.4. String Methods (Part 1 & 2)

2.5. Escape Sequences & Raw Strings

2.6. Unicode & Multiline Strings

2.7. String Formatting: f-strings, .format(), % formatting

2.8. Template Strings & Use Cases

3. Booleans & Operators

3.1. Boolean Values & Comparisons

3.2. Logical Operators: and, or, not

3.3. Operator Precedence

11. Exception Handling & Debugging

11.1. try-except, else, finally

11.2. Custom Exceptions

11.3. raise, assert

11.4. Debugging: pdb, breakpoint(), traceback

11.5. Logging

12. Iterators & Generators

12.1. Iterator Protocol

12.2. Creating Custom Iterators

12.3. Generator Functions, yield

12.4. Generator Expressions

13. Functional Programming

13.1. map(), filter(), reduce()

13.2. zip(), sorted()

13.3. any(), all()

14. MODULE: WEB & API PROJECTS IN PYTHON | Requests Module

14.1. requests.get, post, headers, params

14.2. Status Codes, Timeout, Retry

14.3. Downloading Files

15. Mini Project

15.1. CLI Weather App using OpenWeatherMap API

16. MODULE: DATA STRUCTURES & ALGORITHMS (DSA) | Foundations

16.1. What are DS & Algorithms?

16.2. Time & Space Complexity

16.3. Big-O, Omega, Theta

20.5. Min/Max Stack

21. Queues

21.1. Normal Queue

21.2. Circular Queue

21.3. Deque (Double Ended Queue)

21.4. Monotonic Queue

21.5. Sliding Window Maximum

21.6. NGE (Next Greater Element)

22. Recursion

22.1. Base & Recursive Case

22.2. Tail Recursion

22.3. Recursion Tree

23. Backtracking

23.1. N-Queens

23.2. Subsets

23.3. Permutations

23.4. Combinations

23.5. Sudoku Solver

23.6. Word Search

23.7. Rat in Maze

24. Linear Search

24.1. Concept & Use

25. Binary Search

25.1. Basic Dry Run

25.2. First/Last Occurrence

25.3. Rotated Sorted Array

29.2. Merge Sort

29.3. Quick Sort

29.4. Majority Element

29.5. Closest Pair of Points

30. Bit Manipulation

30.1. AND, OR, XOR, NOT

30.2. Shifts: <<, >>

30.3. Count Set Bits

30.4. Power of 2

30.5. XOR Tricks

30.6. Bit Masking

31. Dynamic Programming

31.1. Recursion → Memo → Tabulation

31.2. Fibonacci Variations

31.3. Climbing Stairs

31.4. House Robber

31.5. Min Cost Climbing Stairs

31.6. 0/1 Knapsack

31.7. Subset Sum

31.8. Target Sum

31.9. Longest Common Subsequence

31.10. Longest Palindromic Subsequence

31.11. Matrix Chain Multiplication

31.12. DP on Trees

31.13. DP on Grids

31.14. DP with Bitmask

31.15. DP with Multiple States

32. Graph Algorithms

33.15. Trie (Prefix Tree)

34. Heaps & Priority Queues

34.1. Heap Basics: Max/Min

34.2. Heapify Logic

34.3. Python PriorityQueue

34.4. Kth Largest/Smallest Element

34.5. Median in Stream

34.6. Merge K Sorted Arrays

34.7. Top K Frequent Elements

35. Sliding Window Techniques

35.1. Max/Min Sum in Window

35.2. Longest Substring without Repeat

35.3. Max Sliding Window

36. Two Pointer Techniques

36.1. Pair Sum

36.2. Container With Most Water

36.3. 3Sum, 4Sum

36.4. Remove Duplicates

37. MODULE: PRACTICE + CHALLENGES

37.1. 3 problems per concept (LeetCode, GfG, Codeforces)

37.2. Practice Sheets

37.3. Daily Coding Challenge Plan

37.4. Mock Tests

38. MODULE: CAPSTONE PROJECTS | Project 1: Build LRU Cache

38.1. DLL + HashMap Design

38.2. Insert/Delete/Access with O(1)

38.3. Unit Testing

39. Project 2: Resume Filter System

39.1. Parse Resumes from CSV/PDF

39.2. JD-Resume Match Score using Keywords

39.3. Hashing + Search + Ranking Logic

39.4. GUI/CLI Options

40. Project 3: Visual Pathfinding App

40.1. BFS, DFS, Dijkstra on Grid

40.2. GUI with Tkinter or Streamlit

40.3. Realtime Node Visualization

40.4. Export Solution Path