Google Apprenticeship Prep Guide 2025 - Complete Playbook

# Core Preparation Principles

## 2️⃣ Core Coding Prep (Daily Practice)

🔹 Data Structures & Algorithms (must-know):  
- Arrays & Strings → sliding window, two pointers, prefix sums  
- Linked Lists → reverse, detect cycle, merge  
- Hash Tables / Sets → frequency maps, lookups  
- Trees & Graphs → BFS, DFS, shortest path (BFS/Dijkstra basics), recursion  
- Sorting & Searching → quicksort, mergesort, binary search  
- Dynamic Programming → subsequence problems (LCS, LIS, knapsack)  
  
📌 Action Plan:  
- Solve 1–2 LeetCode Easy/Medium daily  
- Mix topics: do 5 arrays, then 5 strings, then 5 trees, etc.  
- Keep a notes doc for patterns → this will be your revision sheet.  
  
Resources:  
- LeetCode Patterns  
- HackerRank Interview Prep Kit  
- Neetcode.io (excellent curated roadmap)

## 3️⃣ System Design Basics (for interviews)

Not full-blown design like senior SWE, but you should know:  
- Client-server model  
- REST APIs (how they work, status codes)  
- Database design basics (tables, primary key, indexing)  
- Caching (e.g., Redis basics)  
- Scalability → load balancer, replication basics  
  
📌 Action Plan:  
- Read System Design Primer (intro sections only)  
- Practice explaining how you’d design a simple URL shortener or ToDo app

## 4️⃣ Programming Language Focus

Pick one primary language (Python or Java recommended).  
  
Python Focus:  
- Practice writing clean, readable functions  
- Learn collections module (Counter, defaultdict, heapq)  
- OOP basics: classes, inheritance  
- File handling, exceptions  
  
Java Focus:  
- Learn Collections API (ArrayList, HashMap, HashSet)  
- OOP principles (encapsulation, inheritance, polymorphism, abstraction)  
- Exception handling & streams  
  
📌 Don’t try to master all 4 languages → focus on 1 solid + be aware of others.

## 5️⃣ Git & Collaboration Skills

Google values teamwork + collaboration.  
  
📌 Action Plan:  
- Learn basic GitHub workflows: clone, commit, branch, PR  
- Try one small open-source contribution (even documentation counts!)  
- Document your code (docstrings, README.md)

## 6️⃣ Communication & “Googliness”

During interviews:  
- Talk out loud while coding → show how you’re thinking  
- Ask clarifying questions → shows problem-solving maturity  
- Be positive, curious, collaborative  
  
📌 Practice mock interviews:  
- Pramp (free peer interviews)  
- Interviewing.io (sometimes free mock interviews with real engineers)

## 7️⃣ Resume & Application Review

Since this is an apprenticeship (not a senior SWE role):  
  
- Highlight projects (web apps, small games, automation scripts, open source)  
- Show learning ability → online courses, certifications, hackathons  
- Mention teamwork/collaboration (college projects, GitHub collabs)

# 8-Week Daily Checklist

# Week 1 – Arrays & Strings

## Day 1

☐ Read: Big-O Notation basics

☐ Solve: Two Sum (https://leetcode.com/problems/two-sum/)

☐ Solve: Best Time to Buy and Sell Stock (https://leetcode.com/problems/best-time-to-buy-and-sell-stock/)

## Day 2

☐ Solve: Valid Anagram (https://leetcode.com/problems/valid-anagram/)

☐ Solve: Group Anagrams (https://leetcode.com/problems/group-anagrams/)

☐ Write notes: Hashing patterns

## Day 3

☐ Solve: Longest Substring Without Repeating Characters (https://leetcode.com/problems/longest-substring-without-repeating-characters/)

☐ Solve: Valid Palindrome (https://leetcode.com/problems/valid-palindrome/)

☐ Practice: Explain thought process aloud

## Day 4

☐ Solve: Product of Array Except Self (https://leetcode.com/problems/product-of-array-except-self/)

☐ Solve: Maximum Subarray (https://leetcode.com/problems/maximum-subarray/)

☐ Review Big-O

## Day 5

☐ Solve: Container With Most Water (https://leetcode.com/problems/container-with-most-water/)

☐ Solve: 3Sum (https://leetcode.com/problems/3sum/)

☐ Write notes on Two-Pointer pattern

## Day 6

☐ Solve: 2 random Easy from Arrays/Strings (LeetCode)

☐ Build: Simple Python/Java program → reverse string

## Day 7 (Review)

☐ Revise all problems solved

☐ Mock explain 'Two Sum' aloud

☐ Rest & reflect

# Week 2 – Linked Lists, Hashing, Stacks, Queues

## Day 8

☐ Solve: Reverse Linked List (https://leetcode.com/problems/reverse-linked-list/)

☐ Solve: Merge Two Sorted Lists (https://leetcode.com/problems/merge-two-sorted-lists/)

## Day 9

☐ Solve: Linked List Cycle (https://leetcode.com/problems/linked-list-cycle/)

☐ Solve: Remove Nth Node From End (https://leetcode.com/problems/remove-nth-node-from-end-of-list/)

## Day 10

☐ Solve: LRU Cache (https://leetcode.com/problems/lru-cache/)

☐ Understand: Hashmap + Doubly Linked List

## Day 11

☐ Solve: Valid Parentheses (https://leetcode.com/problems/valid-parentheses/)

☐ Solve: Min Stack (https://leetcode.com/problems/min-stack/)

## Day 12

☐ Write notes on Stack/Queue operations

☐ System Design: What is an API? Example REST call

## Day 13

☐ Build: To-Do CLI app (Python/Java) with add/remove tasks

☐ Push to GitHub

## Day 14 (Review)

☐ Revise all Linked List/Stack/Queue problems

☐ Mock explain LRU Cache aloud