50

Strings Practice Questions

By Tech Jashwanth

🟢 EASY (10 Questions)

(Beginner-friendly, direct string inputs)

1. Reverse the string `"Amazon"`.
2. Count the number of vowels in `"JavaScript"`.
3. Check if `"Madam"` is a palindrome (ignore case).
4. Return the first and last characters of `"Hello World"`.
5. Trim the extra spaces from `"  Welcome to DSA  "`.
6. Convert `"hello-world"` to camelCase.
7. Convert `"apple,banana,grape"` into an array, then join it back with `" | "`.
8. Replace the first occurrence of `"cat"` with `"dog"` in `"The cat sat on the cat."`.
9. Find the index of `"abc"` in `"xyzabc123"`.
10. Check if `"https://google.com"` starts with `"https"` and ends with `".com"`.

🟡 MEDIUM (15 Questions)

(More logic, multiple operations, combinations)

11. Reverse the words in `"I am a tech Jashwanth Subscriber"`.
12. Count occurrences of each character in `"banana"`.
13. Check if `"listen"` and `"silent"` are anagrams (ignore case & spaces).
14. Remove duplicate characters from `"programming"` (keep order).
15. Find the longest word in `"Hard work beats talent"`.
16. Capitalize each word in `"welcome to javascript"`.
17. Check if `"erbottlewat"` is a rotation of `"waterbottle"`.
18. Find all starting indices of `"aba"` in `"abaabaabab"` (return an array of indices).
19. Truncate `"JavaScriptDeveloper"` to **10 characters** and add `"..."`.
20. Check if `"ace"` is a subsequence of `"abcde"`.
21. Remove all vowels from `"Beautiful Day"`.
22. Find the longest common prefix between `"flower"`, `"flow"`, `"flight"`.
23. Convert `"I love coding"` into `"I love coding"` (remove extra spaces).
24. Find the most frequent word in `"This is a test. This test is simple."`
25. Pad `"42"` to become `"000042"` using padStart (total length = 6).

🔴 DIFFICULT (25 Questions)

(Advanced string logic, DP, sliding window, pattern matching)

26. Find the length of the longest substring without repeating characters in `"abcabcbb"`.

27. Find the longest palindromic substring in `"babad"`.

28. Count all **distinct substrings** of `"aaa"`.

29. Compress `"aaabbccccd"` → return `"a3b2c4d1"` (or original if not shorter).

30. Implement substring search and find the first index of `"needle"` in `"findtheneedleinahaystack"`.

31. Find the shortest substring of `"aabcbcdbca"` that contains all distinct characters of the string.

32. Given string = `"ADOBECODEBANC"` and pattern = `"ABC"`, find the minimum window containing all characters.

33. Remove the minimum number of parentheses to make `"a)b(c)d"` valid; return a valid result.

34. Check if `"great"` and `"rgeat"` are scramble strings.

35. Return all palindromic partitions of `"aab"` (e.g., `["a","a","b"]`, `["aa","b"]`).

36. Match text `"adceb"` with pattern `"*a*b"` where `*` = any sequence and `?` = any single char.

37. Restore all valid sentences from `"catsanddog"` using dictionary `["cat","cats","and","sand","dog"]`.

38. Convert `"MCMXCIV"` to integer (Roman to number).

39. Given `"abcd"`, find its **next lexicographical permutation**.

40. Given string `"AABABBA"` and k = 1, find the longest substring where you can replace at most 1 character to make all characters same.

41. Check if `"abca"` can become a palindrome by removing at most one character.

42. Count how many anagrams of `"abc"` exist inside `"cbadbac"` (sliding window).

43. Stream of characters: `"a","a","b","c","a","b","d"` — return the first non-repeating character at each step.

44. Compute the edit distance between `"horse"` and `"ros"`.

45. Validate if `" -42.50 "` is a valid number (integers & decimals allowed).

46. Normalize email addresses in `["test.email+alex@gmail.com","test.e.mail+bob@gmail.com"]` and return count of unique ones.

47. Form the shortest palindrome by adding characters **in front** of `"abcd"`.

48. Find the **longest common substring** between `"abcdef"` and `"zcdemf"`.

49. Given `"applepenapple"` and dictionary `["apple","pen"]`, check if it can be segmented (word break).

50. Evaluate the expression `"3+5*2-9/3"` (build an expression evaluator).