
HOW TO APPROACH TECHNICAL INTERVIEWS

The objective of the interview is to try to understand your approach to problem solving. You could be asked to solve a problem in any way you choose, and then the interviewer could add further constraints or requirements. Consider the technical tools and skills required for the job. Identify if the role is looking for particular coding languages, data structures, system design, etc. An understanding of these helps identify what skills or tools you need to revisit as a refresher.

HERE ARE A LIST OF BEST PRACTICES TO CONFRONT A TECHNICAL INTERVIEW:

- Think out loud. Interviewers pay a lot of attention to the way you solve problems, which can be as important as having the right answer. Thinking out loud gives the interviewer insight into your thinking process and can also help them follow along with your solution.
- If you're interviewing remotely, locate a good interview spot. Choose a quiet place and ensure that you have a good internet connection and strong phone reception.
- Speak clearly. Ensure you are speaking clearly and likewise, if you can't hear the interviewer clearly, let them know so they can accommodate.
- Manage your time effectively. Spend some time figuring out the ideal solution to the question. Don't jump too quickly into brute forcing the first solution that comes to mind. If you can't find a better solution in a reasonable time, start writing a working solution, then iterate and improve it as you go. Some interviews end abruptly because the interviewee couldn't find the ideal solution. It's better to have a non-optimal but working solution than just an idea. Once you have a working solution, you can then try to improve its efficiency, design, or any other aspect of it.
- Share your reasoning: Make sure you can talk about your solution because you will probably be asked to explain it.
- Find and fix the errors by yourself. Don't wait for the interviewer to find them for you.
- If you are stuck, ask clarifying questions. Usually, the interviewer knows the question well enough to provide information that may help you move forward.
- Utilize your time to draw visuals that may assist with systems design questions.
- Recall experiences from past or current as the baseline for answering technical questions. What real-life examples can you use in solving technical questions?
- Be prepared for questions that are specific to when implementations went wrong. Sometimes a hiring manager wants to understand how you troubleshoot and recover from implementations that required a different approach once you started. Focus on how this experience made you better technically.
- Focus on who you're developing the solution for, who is the stakeholder or the expected outcome. Based on the customer, your solution may be designed or function differently.
- Be prepared to discuss other technical teams you've worked with to solve problems and how you've worked on technical projects. Explain processes followed to ensure delivery success.
- If the job is specific to coding, practice by doing an on-the-spot coding test.