Acing the Technical Interview

Technical interviews allow you to showcase your knowledge and skills to prospective employers.

They can be exciting, knowing that you're that much closer to a job offer if you do well. They can also be intimidating — especially if you've never done one before or struggled with them in the past. But don't worry. We're here to help.

Below, we've outlined everything you'll need to know to ace your next **technical interview**, including preparation, practice, and tips for impressing your interviewer. Read on for the full guide or jump to a specific section using the table of contents below.

What is a technical interview?

Technical interviews are a part of the hiring process for tech professionals. This process also includes phone screenings and in-person interviews, but the assignments you complete during technical interviews help interviewers evaluate your skills.

Assignments vary by company, but common tasks include:

- Debugging existing code.
- Building a small project.
- Adding a feature to an existing code base.
- Explaining the design and architecture of a technical problem.

While you'll likely be asked to solve a problem, don't worry too much about getting it right. Interviewers are typically more interested in your thought process than your solution.

Preparing for a technical interview

Preparation is the key to a successful technical interview. You might not know what your assignment will be, but there are several steps you can take to ensure that you’re equipped for anything. Plus, your interviewer is bound to be impressed by your readiness.

To help you hit the ground running, our developers put together a list of tips to help you prepare for a technical interview.

Ask questions leading up to your interview

By the time you have your technical interview, you’ve probably already spoken with the company’s recruiter or hiring manager. Feel free to ask them questions about what you should expect or the technologies used by the team you’ll be joining.

Master your programming languages

You can usually use your preferred programming language for the technical interview, and knowing its ins and outs will help you make a good impression.

Mastering your programming language will allow you to give the assignment your full attention instead of wasting time figuring out syntax. Plus, you might get the chance to wow your interviewer with a lesser-known feature or method.

It’s also worth taking the time to learn data structures and algorithms. Not only will this make your coding more efficient, but it’ll also deepen your understanding of the concepts employed in your technical interview.

Practice with projects and coding challenges

Theoretical knowledge will only take you so far. Completing coding challenges and projects will help instill the skills you’ll use in your new career. Plus, the more you practice, the faster you’ll become. Imagine how impressed your interviewer will be when you’ve efficiently solved the problem in half of the allotted time.

As you complete your projects, take note of how you approached the problems you encountered. Taking the time to analyze your approach helps develop your problem-solving ability and improve your capacity for technical presentations.

You could also practice with technical interview courses. Our Technical Interview Practice with Python, Java, and JavaScript courses cover advanced knowledge and skills, with practice problems to help you nail your next interview.

Memorize a story that illustrates your skill

At some point during your technical interview, you may be asked about your prior experience. Having a story readily available will help illustrate your familiarity with the knowledge and skills required for the position.

Use the notes you took while completing a project to create an outline and commit it to memory. Then, test it out on people. Share the story with your friends until you're able to tell it without any stammering or pauses. This will help you sound confident during your technical interview, and being able to succinctly describe the problems you encountered and your solutions will demonstrate your expertise.

Other preparation tips

The tips listed above will help ensure that you're fully equipped for your next technical interview. Still, sometimes it's not what you say but how you say it. Check out these tips from our Curriculum Developers for more advice on how to prepare for a technical interview.

How to pass a technical interview

After taking all the steps to ensure that you're ready to put your best foot forward, it's time to take your technical interview. Technical interviews can occur both in-person or remotely. If it's in-person, you may be asked to write your code on a whiteboard. If it's remote, you might use a code editor.

Regardless of where it's held, there are several steps you can take to make sure your technical interview is a success.

Understand the problem

Taking the time to understand the problem will make or break your interview. Too often, candidates rush in and start coding. Don't be afraid to talk through it aloud.

Also, feel free to ask questions. It's better to clarify any confusion at the beginning than when you're halfway through your solution. Inquire about edge cases, intended program behaviors, error handling — anything that might occur to you. Not only will this show your critical thinking skills, but it'll also illustrate your openness to feedback.

Explain your approach

Again, interviewers are primarily concerned with your thought process. Here's your chance to give them a look.

Back in December, we spoke with Daniel Chae, a programmer and data analyst from Los Angeles. While discussing the hiring process, Daniel stressed the importance of explaining your approach in technical interviews. He breaks it down into two steps:

- Identifying your core goal.
- Generating a step by step process you can explain aloud.

Explaining your approach allows interviewers to better understand how you go about tackling a problem. Daniel describes how, despite his solution not being perfect, explaining his approach gave him a leg up in a technical interview with Google.

Don't stop explaining, even when you run into a problem. Talking through it will help you organize your thoughts and find a solution. Plus, your interviewer is much more likely to give you a hand if you're already engaged in conversation.

Optimize your solution

Going back to optimize your solution is a great way to display your attention to detail. Tidy up your code. Make sure you're not using complicated functions for a problem that can be solved with your language's basic features. If you're familiar with Big-O notation, discuss it with your interviewer.

Technical interview questions and answers

Certain positions require specific languages, and some technical interviews will test your capacity with them. Below, we've compiled a list of interview questions covering the basic and advanced features of some of the most popular programming languages:

- 7 HTML interview questions to practice
- 7 top CSS interview questions to practice

- 7 JavaScript interview questions to practice

We wish you the best of luck in your interview, and your new career!