# **Tech Interview Prep Cheat Sheet**

## **Types of Tech Interviews**

1. **Phone/Video Screening** - Basic questions, culture fit
2. **Technical Screen** - Live coding, problem-solving
3. **Take-Home Project** - Build something in 2-5 days
4. **On-site/Final Round** - Mix of technical, behavioral, team fit

## **Common Technical Questions (Entry Level)**

**Programming Basics:**

* Explain the difference between == and === in JavaScript
* What is the difference between let, const, and var?
* How do you handle errors in your code?

**Problem-Solving:**

* "Walk me through how you would build a simple to-do app"
* "How would you debug a feature that's not working?"
* "Explain a challenging project you worked on"

**Web Development:**

* What happens when you type a URL into a browser?
* Explain the difference between frontend and backend
* How would you make a website mobile-friendly?

## **The STAR Method for Behavioral Questions**

**Situation** - Set the context **Task** - What needed to be done **Action** - What you specifically did **Result** - The outcome/impact

**Example:** "Tell me about a time you solved a difficult problem"

* **S:** While building my portfolio site, the contact form wasn't working
* **T:** I needed to debug and fix it before my job application deadline
* **A:** I used browser dev tools, checked console errors, and researched the API documentation
* **R:** Found the issue was a typo in my endpoint URL, fixed it, and the form worked perfectly

## **Questions to Ask Them**

**About the Role:**

* What does a typical day look like for this position?
* What technologies does the team currently use?
* What's the biggest challenge facing the team right now?

**About Growth:**

* What opportunities exist for learning and development?
* How do you support junior developers?
* What does success look like in this role after 6 months?

**About Culture:**

* How do you handle code reviews and feedback?
* What's your deployment process like?
* How does the team collaborate on projects?

## **Day-Before Checklist**

✓ Test your video/audio setup ✓ Review the job description and your application ✓ Practice explaining your projects out loud ✓ Prepare 3-5 questions to ask them ✓ Get a good night's sleep ✓ Set up a clean, quiet interview space

## **During the Interview**

**Do:** • Think out loud while problem-solving • Ask clarifying questions • Admit when you don't know something • Show enthusiasm for learning • Connect your past experience to the role

**Don't:** • Panic if you get stuck - talk through your thought process • Pretend to know technologies you don't • Speak negatively about previous employers • Forget to ask questions at the end

## **Live Coding Tips**

1. **Read the problem carefully** - ask questions if unclear
2. **Plan before coding** - outline your approach
3. **Start simple** - get basic functionality working first
4. **Test as you go** - use console.log to check your work
5. **Communicate constantly** - explain what you're thinking

## **Salary Discussion**

**Research first:** Use Glassdoor, PayScale, levels.fyi **For entry level:** Focus on learning opportunities over salary **If asked about expectations:** "I'm looking for fair compensation based on the role and my experience level. What's the typical range for this position?"

## **Follow-Up**

**Within 24 hours:** Send thank you email mentioning specific conversation points **Sample:** "Thank you for taking the time to discuss the junior developer role. I'm excited about the opportunity to contribute to [specific project discussed] and would love to join the team."