Telling Better Stories For Behavioral Programming Interview Questions

"*Show*, don't tell"

You've probably heard this advice before. Maybe it was your 10th grade English teacher. Maybe it was career services in college. "Remember: *show*, don't tell."

And it's good advice. When it comes to answering behavioral questions (like "Tell me about yourself") in coding interviews, the difference between a good answer and a *great* answer comes down to *showing* rather than telling.

The problem is, people who give you the advice of "Show, don't tell"... are *themselves* failing to follow it. They're *telling* you to show, but they should be *showing* you *how to show*. That's the hardest part!

So here are three specific tips for showing more and telling less.

1. Sprinkle in specific details

Imagine two responses to the stock interview question "Tell me about yourself."

First:

> *I started programming about two years ago with some personal projects. I eventually got a job at a small tech company in my home town, and I've been working there about a year and a half. I like my job, but I'm looking for a new challenge, which I think your company could provide.*

Then:

> *I got started programming because I wanted to build a social network for cats. That didn't take off, but the prototype helped me get a job at a small tech company in my home town.*
>
> *Last month, I read an awesome article on Hacker News about the social network your company is building. The scaling challenges you face seem like they'll help me grow faster and stronger than my current role will.*

The second response says *a lot* more about the candidate.

Why? Because of the *specific details*. An interviewer won't remember the tenth person to say "I'm looking for a new challenge." They *will* remember the person who tried to build a social network for cats and read about their company on Hacker News.

So don't skimp on the details. **Look out for opportunities to use specifics**, especially if they're at all quirky, funny, surprising, or otherwise *memorable*.

2. Tell a story from your life

Take another common question: "Why do you want to work here?"

People tend to just cross-reference their values with those of the company or team they're interviewing with:

> *I'm really interested in technical blogging and open source. So I like that your company has some open-source work and contributes back to the community.*

That's a fine response. But to really wow your interviewer, try adding a specific story around those values:

> *A couple years ago, when I was still new to programming, I was working on this tricky bug. I found a post on a company blog where an engineer explained how her team solved the issue. She included a code snippet she'd open-sourced. I appreciated that she took the time to write about her team's experience and share their solution. It helped me!*
>
> *That's how I first started getting into open source. I really wanna work with more engineers like that—who write about their work and try to help others in the community. So I was excited to see all the stuff your team shares on your blog and on the company's Github profile.*

The second response just sounds more genuine. It *shows* a personal connection to open source and technical blogging, instead of just *telling* it.

Anyone can look up a company's core values and repeat them during an interview. It's more meaningful to **tell a story from your life** that shows how those values benefited you or taught you something.

3. Use someone else's voice

This one's a neat trick. Consider one more standard behavioral question: "What's your biggest strength?"

You might tell the interviewer:

> *I work well with others. Even under tough circumstances, I make sure my coworkers feel supported.*

But a lightly detailed story is better suited to show this strength:

> *I have a coworker, Ana, who's been an engineer for almost a decade. We worked together on this really tough, messy project.*
>
> *Towards the end, she told me, "For such a hellish project, you really made things feel sane." I think this is my biggest strength—I work well with others, even under tough circumstances.*

When you respond with a story, you can **refer to what *other* people have said about your best qualities**. In this case, a ten-year tech veteran said you made a project feel less awful. That kind of praise is a lot more credible when it comes from someone else.

Practice, practice, practice

Remember these specific tricks for showing rather than telling:

1. **Use specific, memorable details**. "Social network for cats" instead of "a personal project."
2. **Tell a story from your life**. "I was trying to solve a tricky bug…" instead of "I value open source contributions."
3. **Use someone else's voice**. "'You really made things feel sane'" instead of "I work well with others."

Try these tactics out on the questions below. Keep in mind, sometimes it's easiest to start with a "tell" response, then spruce it up to "show."

- Tell me your biggest weakness as an engineer.
- Describe a tricky bug you've encountered.

- What's the biggest project you've shipped?
- What's your favorite programming language? Why?
- How do you overcome interpersonal conflicts with coworkers?

Want more coding interview help?

Check out **interviewcake.com** for more advice, guides, and practice questions.