

How to Do Well in QC Sessions

Performing well in technical interviews, including QC, requires confidently answering both high-level conceptual questions on specific technologies as well as low-level details about how you've worked with them.

For example, as a developer you've most likely worked with git before. A common interview question to test your proficiency might be "Tell me about your experience working with git". How would you answer this? Take a minute to think about it and then consider the sample responses below. Which answer would impress you more as an interviewer? Which one demonstrates an understanding and proficiency with the tool better?

Response 1

*"Well, uh, for git uh I think I've used it on a few projects for pushing code... it's really **helpful**, **pretty easy** to work with, it lets me **keep track** of **stuff going on** in the project and **work with other people well**."*

Response 2

*"So I've used git for almost all of my projects. It's a really useful **distributed version control system** for managing **source code** - I've used it **on the command line** to **add, commit, and push** code to an **online repository**, usually **GitHub**, as well as switch between different **feature branches** that my team members and I work on. I like it because it **smooths the process of collaborating** on the same codebase and **merging or integrating changes**."*

Note a few problems in the first response: the use of filler words ("ummm", "ahh") - you should remove these as much as possible. Don't lower your voice and trail off at the end of sentences, it implies uncertainty. Also note the vague words that are **bolded** - it doesn't answer **how** git is helpful, how it is easy to work with, or how it lets you keep track of "stuff" (whatever that is). Finally, the only technical word mentioned is "pushing" code.

In contrast, the second answer is well-structured and smooth, and it drops in several technical keywords that demonstrates knowledge in the domain - it explains **what** ("distributed version control"), **how** ("add, commit, push code"), and **why** ("smoothing the process of collaborating"). Putting these together, it **sounds like the person giving the second answer has worked in-depth with git**.

Review one more high-level sample question and answer below.

Q: “Tell me about the Spring framework”

A: “The Spring framework is a **Java framework** for **production-ready enterprise applications** based on the concept of **inversion of control** where we hand off control over **instantiation of our objects** to an external entity. In Spring this entity is the **IOC container** which manages the **lifecycle of the application beans**, provided either through **XML, Java, or annotation configuration**. We can achieve **loose coupling** of our code by letting Spring **inject our bean dependencies** directly into our classes.”

Remember, there are 3 key questions you should ask for every technology you learn about - **what is it and what does it do, how does it work, and why is it useful?**

As you go through training, think about this approach for structuring your responses in QC, one-on-ones, and panels. To really improve at answering these conceptual questions, you **MUST** practice your own responses. **Also, it’s not enough to memorize the keywords to say - you must fully understand and internalize the knowledge, otherwise you will not be able to back yourself up when asked follow-up questions.**

Technical Exercise

The exercise below should help you improve your answers in QC, one-on-ones, panels, and interviews. **You should complete this exercise each week as you cover new technologies throughout training.**

1. Find a blank sheet of paper or open a text editor on your laptop
2. List the technologies, languages, libraries, and frameworks you have learned this week (e.g. HTML, CSS, JavaScript)
3. For each one, write a response to the question “Tell me about your experience with X” where X is the given technology.
4. Review your answers and practice saying them out loud. Note any verbal miscues or awkward phrasing
5. **Bold** the important keywords in your responses
6. Revise your answers. Look for ways to improve clarity, be specific, improve the structure, and make the answers more cohesive.
7. Find a partner (a roommate or batchmate or your trainer). Practice saying your answers out loud and gather feedback from your partner. Make final revisions to your answers.

If you do this every week, at the end of training you will have a complete set of answers to each language, framework, and major concept you’ve covered. You can use these to improve the bullet points and written sections in your portfolio as well as prepare for the final, cumulative QC session.