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** - there is an **IOC container** which manages the **lifecycle of your 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.

Pre-Cumulative QC Exercise

Pick any 4 technologies you've worked with in training. Use the spaces below to write out your responses to the question "Tell me about your experience with X" where X is the technology. **Bold** the important keywords in your responses.

1. _____
   _____
   _____
   _____

2. _____
   _____
   _____
   _____

3. _____
   _____
   _____
   _____

4. _____
   _____
   _____
   _____

Now, review this, revise it, ask your teammates to review it and give their feedback, and then finalize your answers. You should practice saying these answers so that it comes naturally to you when you are asked these questions in interviews.