Member-only story

Answering the Unanswerable

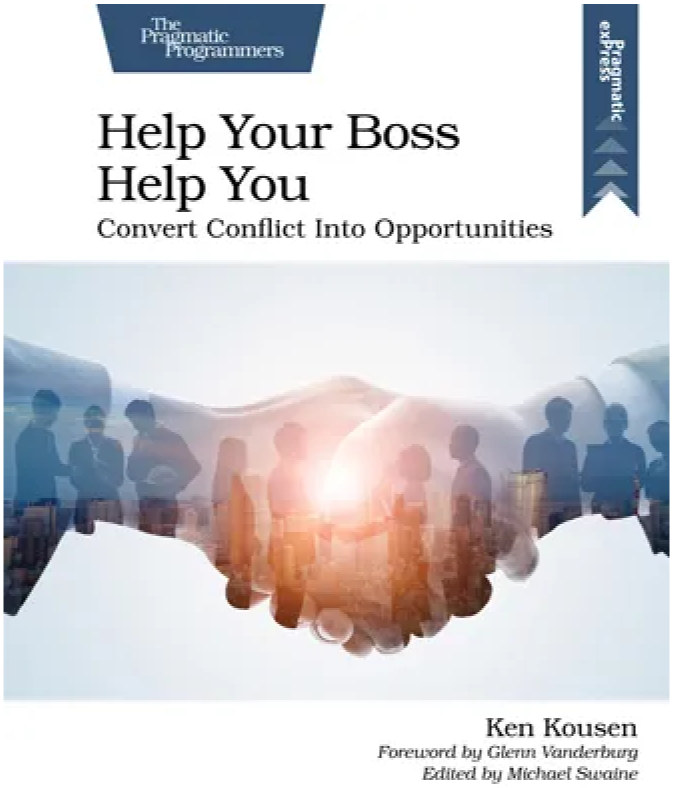
Giving Good Enough Answers

Kenneth Kousen

Published in The Pragmatic Programmers · 9 min read · Jul 1, 2021

113 3

In this article, you'll get a template for replying to open-ended questions from your manager with a minimum of effort.


The Pragmatic Programmers
Pragmatic exPress
Help Your Boss Help You
Convert Conflict Into Opportunities
Ken Kousen
Foreword by Glenn Vanderburg
Edited by Michael Swaine

Help Your Boss Help You by Ken Kousen

This article is one of a series about how to build a relationship with your manager that gets you what you want on the job when you want it. Look for the other articles in the series at the bottom.

The content is based on Help Your Boss Help You by Ken Kousen, published by The Pragmatic Bookshelf. Hint: read to the end of the article for a promo code.

The Disruption of Irrelevant Questions

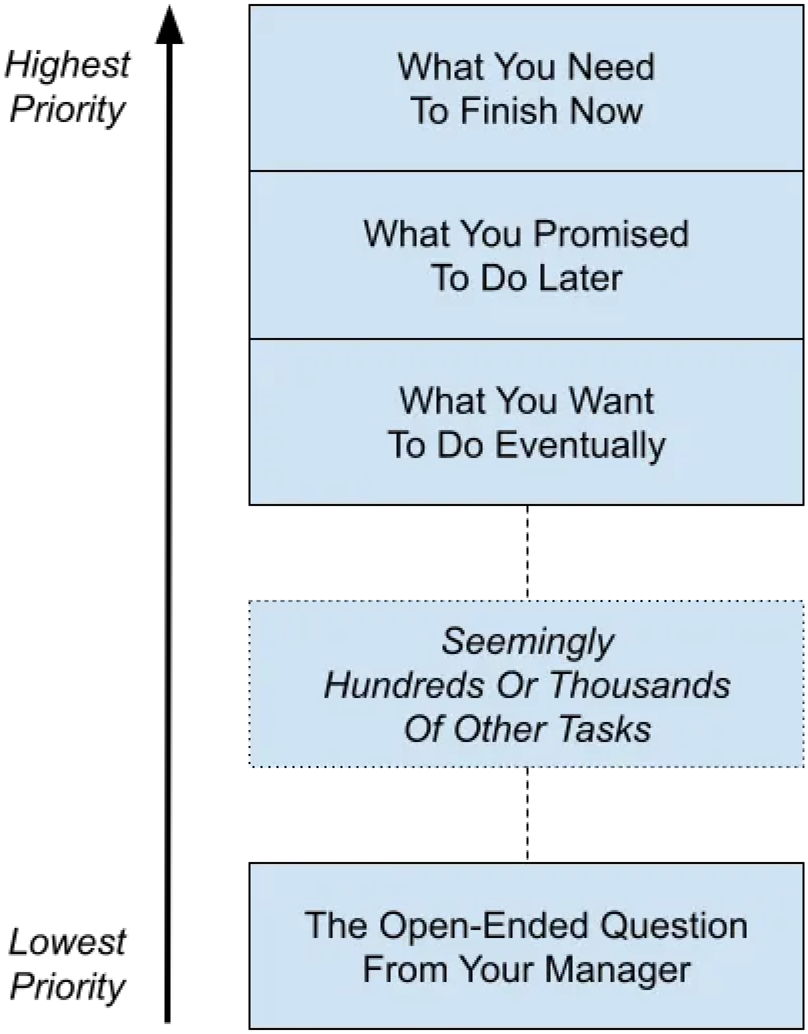
Imagine that one day you're working at your job, trying to complete an urgent task and hoping to get to tasks you'd actually *like to do* when an email comes from your boss:

> *Hey, <your_name>, I have a question. How do you think the recent advances in A.I. and Machine Learning are going to affect our primary product over the next few years?*

You now have a dilemma. Your honest answer is, "I have no idea," followed closely by, "why are you asking me? Don't you realize I have work to do? Work that you gave me? You realize this is a major distraction, right?"

Of course, you don't really want to say that, especially if it's true.

We can summarize your present situation with the following figure:


Highest Priority
What You Need To Finish Now
What You Promised To Do Later
What You Want To Do Eventually
Seemingly Hundreds Or Thousands Of Other Tasks
Lowest Priority
The Open-Ended Question From Your Manager

Therefore, you feel you have two options:

1. Ignore the question until you have time to dig into it.
2. Put aside all your work and spend the next few days figuring out the right answer.

Unfortunately, neither option will get you what you want, which is to get rid of this question while still appearing responsive. The best approach to the problem is to formulate a *Good Enough Answer*. That's an answer that isn't necessarily right or complete but satisfies both you and your manager. You'll find a template for Good Enough Answers below, but first, let's talk about why creating them is so challenging.

The Game of School

Photo by Felicia Buitenwerf on Unsplash

Part of the reason you're stuck is that most of us feel we have to respond to every question with the *right* answer. This pattern is drilled into us in school. In academic environments, when a teacher asks you a question, they already have an answer in mind. Even more, they want you to demonstrate that you can solve the problem in the way they want to see, known as showing your work. You're being evaluated both on your ability to produce correct answers on demand and to generate them the right way. Therefore, to be successful in school, you have to not only know the answers, you have to know how the teacher wants to see them produced.

I call this skill winning *The Game of School,* as opposed to actually learning the material. Being good at The Game of School comes from a combination of empathy (in order to figure out what the teacher wants) and good written and oral communication skills.

The other problem with academic environments is that there is a harsh penalty for not knowing the correct answer, and even incomplete answers are downgraded, costing you points. The result is that years of school train you to value complete, right answers, given in the right way, and to avoid anything short of that.

Unfortunately, the business world doesn’t work that way.

The Game of Business

Photo by Avel Chuklanov on Unsplash

The biggest difference between The Game of School and The Game of Business is that when somebody on the job asks you a question, they don't know the right answer. They're stuck, or they wouldn't be asking you in the first place.

As an example, let me pick an issue that arose for me in software development. One of the technologies I deal with on a regular basis is the Git source code control system. Git is very powerful and very popular, but it has a well-deserved reputation for weird, counter-intuitive syntax. The Git-related tasks I do all the time I know how to do, but for almost anything else, I'm lost. To handle that, I have a colleague that I email with questions. If I encounter a Git issue, I'll send him a message.

"I accidentally committed my changes to the wrong branch," I'll say. "How do I undo them and add them to the right one?"

Then I wait, and I wait, and I wait, and during that time I get more and more frustrated because I can't make progress until I hear back from him. Eventually, I'll run into him in a meeting or on a Zoom call.

"Why didn't you answer my email?" I'll ask.

"Sorry, I've got 6000 unread emails," he'll reply.

"So? Why didn't you answer mine?"

I might not say that, but I'll certainly be thinking it. Now imagine how much worse that situation is when the questioner is your boss. If your boss is stuck and asks you a question, they expect a timely reply and any delay is going to be a problem.

Say your manager asked you a distracting question involving topics you don't really know. If you postpone answering for either of those perfectly legitimate reasons (you don't know or you're too busy right now to figure it out), your manager will be aware only of the delay and will feel like they're being ignored. If you put aside your current project work to figure out an answer, you run the risk of not delivering on your existing commitments. Both of those approaches get you into trouble.

Answering the Unanswerable

Photo by Mark König on Unsplash

Fortunately, there is an easy way to handle open-ended questions that are unrelated to your current job. In order to see the pattern, here's a hypothetical answer to the AI/Machine Learning question posed above. I should admit upfront that I'm not an expert in either subject, but that's the point.

> *I don't know, but here's what I can tell you now. Both fields are active and producing results on a regular basis, and they seem to be most effective in areas related to pattern matching and natural language processing. It also appears that Google has taken a lead in the field, especially with their open source TensorFlow tool.*
>
> *I therefore think that for our product, we may be able to incorporate some kind of expert system that understands what the customer actually wants when*

> *they ask questions. We also might be able to use TensorFlow to dig into our own data to find patterns we might not otherwise detect.*
>
> *To do all that would take a lot of expertise both in the software itself (which likely means learning more Python and probably some advanced statistics), and understanding what those tools produce and how to interpret the results. Since this is an active area of research, that may take a fair amount of effort. On the other hand, those tools are likely to stabilize and become commodities in the future, so if we wait we may eventually be able to just buy one and customize it with our own data.*
>
> *Do you want me to look into this in more detail?*

See how the answer breaks down into a few key sections:

- You acknowledge right away that you don’t know the *right* answer, but you’ll try to respond anyway. That’s important because it covers your response while still being responsive. Add any additional caveats you need in order to feel comfortable giving an answer.
- Add a few sentences explaining what you know now. It’s likely you’ll know something about the topic, or your manager probably wouldn’t have asked you in the first place.
- Then include what you believe to be true or is likely to happen in the future. That is the part most likely to be wrong, but you’ve already covered yourself. In this section, try to apply what you know to the specifics asked by your boss.
- The next section is very important: summarize the steps you believe would be necessary to produce a correct answer. This part is really

valuable to your boss because it's a rough time estimate (and therefore a rough cost estimate) of what it would take to solve the problem.

- Last comes the *magic question*: Ask your boss whether or not you should spend that extra time.

There are several advantages to this approach. You're *adding caveats* that explain that you are aware this isn't a complete answer. You then show that *you do have some thoughts* about what comes next. Then you're *giving a rough cost estimate* of what it would take to actually solve the problem. That's critical, because managers are always keenly aware of costs, and any estimate you give them will be appreciated and taken into account.

Finally and most importantly, you're *letting your manager make the decision* about whether you should put actual work on hold to dig into the details. In the business world, professionals advise, while managers decide. If you either ignore the question or postpone project work to deal with it, you're making the decision about what to do next yourself, without knowing why the question was asked in the first place. Let your manager decide that. I promise you that in the vast majority of cases, the response will be:

"That's fine. That's all I needed. Thanks for responding so quickly."

To summarize the pattern, this is how you handle open-ended questions from your boss:

1. Start with "*I don't know, but ….*" and add any necessary caveats.
2. Add the parts you're sure about: "*Here's what I know…*".
3. Add what you believe: "*Here's what I think…*".

4. Give a rough cost estimate by showing what to do next: *"Here's what I would do to solve the problem…"*.
5. Then ask the magic question: *"Do you want me to look into it?"*

Again, the response from your manager will almost certainly be, *"Thanks, that's good enough."*

Why did your manager ask you this question in the first place, when the current focus of your job is unrelated to AI and machine learning? Maybe they were in a meeting where somebody brought up the subject. Or maybe they were contacted by a vendor who wanted to sell them a related product. Or maybe their manager asked them, and they're collecting information from all their direct reports. It could be anything, but most reasons will not require you to change your career specialization in order to respond. In the unlikely event that your manager does ask you to go further, then they're the person telling you to add this task to your schedule, so they're responsible for helping you plan and dealing with any consequences.

The idea is to produce a good enough answer in a timely fashion. If you follow the Pomodoro technique, you can produce one in a single time block. Just set a timer for 20 minutes and do your best. With practice, you can produce good enough answers that sound thorough and informative but still cover yourself when the details are uncertain.

Best of all, by replying quickly (1) you were responsive, which is extremely valuable to your manager (and a real problem if you're not), and (2) you got this annoying question out of your inbox with a minimum of effort.

The basic principle can be expressed this way:

> *A good enough answer today is way, way better than a great answer next week.*

This is a difficult lesson for most professionals to adopt— we hate being wrong and we tend to be perfectionists. Most of us spent years in school learning that incomplete answers are penalized. But if you adopt good enough answers, you can save yourself an enormous amount of stress. Give a good enough answer right away, and you’ll make your manager happy and build a reputation for responsiveness that will help that relationship in the future.

📝 In case your manager does want you to spend time on those subjects, *The Pragmatic Bookshelf* has books on Python and AI and Machine Learning, as well as the Pomodoro technique.

More Wisdom From Help Your Boss Help You

If you enjoyed this article, you may also enjoy the following articles, which are based on the concepts from *Help Your Boss Help You* by Ken Kousen:

Hacks: Your Boss Is Not Your Friend

What Do Popular TV Shows Have to Do with Your Work Relationships?

medium.com