One of my colleagues asked me to substitute for her in a systems engineering course that I used to teach until a few years ago. The assigned topic was how students could best work effectively in software development groups. With the instructor’s agreement, I decided to discuss some beliefs, attitudes and behavior that can help students entering the workforce for the first time as interns or new employees make the best of their opportunities. Some of these topics may be helpful to a wider audience.

From this point on, I’ll simply address readers rather than using the past tense.

1) How to enter a new work group

The worst approach a new employee can take – especially a recent graduate or an intern – is to swagger into a new workplace and start comparing the way things are done to the style at a previous workplace – or worse, to what some professor told the student in a course.

The first steps to success in any new job are to observe and learn: listen, watch, think! Keep your opinions to yourself until you have learned more than superficial impressions about your new work environment. Watch, listen, learn, and think. If you don’t mind the idea, keep a notebook (file) about what you learn.

2) Making suggestions

Keep your opinions to yourself until you have earned credibility by doing your job well and being thoughtful, courteous and helpful. If you do see opportunities for improvement, find out who is likely to be responsible for making the changes you think might be useful. Don’t launch into a diatribe about how rotten the current situation is: ask the person if she can discuss the specific issue you are concerned about. Describe your impression of the current situation, define the problem neutrally (avoid emotional language), and ask your contact about what she thinks about it. Then offer your suggestion respectfully (not arrogantly, not from a position of assumed superiority, not rudely) and be prepared to listen to a different perspective. Don’t assume that just because you think something ought to be changed that it will be.

3) Work (and life) is not a zero-sum game

Helping someone to do better (at work, in your family, in your marriage, in your life) does not subtract from your success. Take every opportunity to share knowledge, lend a hand, prevent an accident – you will win as a worker and as a human being. Don’t believe the cynics who tell you that the individual is all that counts, that there is no valid social group beyond the family, and that everyone should maximize their gain at the expense of all comers. Life does not have to consist of a battle with every person competing against every other.

4) Egoless work
Suggestions for improvement to code or writing are not attacks on you. When we write or code, we sometimes see in our work what we want to communicate rather than what a reader or a computer will see and execute: our assumptions are often implicit and invisible to ourselves. Thus an editor or a code reviewer may challenge a passage of an article or a section of code and point to improvements. Be grateful, not resentful. For more on this topic, see an article I wrote in this column in 2006<http://www.networkworld.com/newsletters/sec/2006/0130sec2.html> and the essay “On Writing.”<http://www.mekabay.com/methodology/writing.pdf>

5) Don’t blindside your boss

Always work to make your boss look good. In particular, keep your boss informed of anything out of the ordinary: the last thing you want is to have your supervisor challenged by a superior officer demanding to know what you have been doing – and have the supervisor unable to answer.

6) Be honest

Do what you say you will do: don’t pretend. For example, if you are seeking someone’s opinion, listen honestly and openly – don’t go through the motions. In Eric Berne’s famous book, Games People Play: The Basic Handbook of Transactional Analysis,<http://www.amazon.com/exec/obidos/ASIN/0345410033/qid=1099166101> the author describes the why-don’t-you-yes-but game, in which someone asks for advice and then proceeds to show why every possible solution is wrong or impossible to implement. A typical application of this game is for a dishonest manager to ask for employee opinions about a planned or existing policy and then to ignore or discount every comment as meaningless or wrong. The manager is being dishonest. Don’t do that.

So endeth today’s lesson. Go forth and be good human beings.

* * *

M. E. Kabay, PhD, CISSP-ISSMP <mailto:mekabay@gmail.com> specializes in security and operations management consulting services. CV online.<http://www.mekabay.com/cv/>

Copyright © 2009 M. E. Kabay. All rights reserved.

Permission is hereby granted to Network World to distribute this article at will, to post it without limit on any Web site, and to republish it in any way they see fit.