

## introduction

### tips

- care about your craft
  - \* no point in developing software unless you care about doing it well
- think about your work
  - \* never run on auto-pilot
- *kaizen* - continuously making many small improvements
- when you do *accept the responsibility* for an outcome
  - \* you should expect to be held accountable for it
  - \* when *you make a mistake* or an *error in judgement*, **admit it honestly** and **try to offer options**
- instead of excuses, *provide options*
  - \* don't say it can't be done, *explain what can be done* to salvage the situation
- all the facts programmers know about computing, the application domains they work in, and all their experience can be said to be their **knowledge portfolios**
  - \* managing them is very similar to managing a financial portfolio
    1. serious investors *invest regularly* - as a habit
    2. diversification is the key to *long term success*
    3. smart investors balance their portfolios between conservative and high-risk, high-reward investments
    4. investors try to *buy low* and *sell high* for maximum return
    5. portfolios should be *reviewed* and *rebalanced periodically*
- **invest regularly** in your **knowledge portfolio**
- it is important to think **critically** about what you read and hear
  - \* you need to ensure that the knowledge in your portfolio is *accurate* and *unswayed* by either vendor or media hype

### care and cultivation of gurus

- know exactly what you want to ask, and be *as specific as you can be*
- *frame your question carefully and politely*

- \* remember, you're *asking for favour*, don't seem to be *demanding an answer*
- once you've framed your question, *stop and look again for the answer*
  - \* pick out some *keywords* and *search the web*
  - \* look for *appropriate FAQs*
- decide if you want to ask *publicly* or *privately*
- *sit back and be patient*
  - \* *people are busy* and it may take days to get a specific answer
- be sure to *thank anyone* who responds you

## communication

- **know what you want to say**
  - \* *plan* what you want to say
    - \*\* write an *outline*
    - \*\* then ask yourself '*does this get across whoever I'm trying to say?*'
      - \*\* **refine it until it does**