

Pragmatic Programmer

David Janzen

Preface

- Tip #1: Care About Your Craft
 - Why do it unless you intend to do it well
 - How is your work similar/different to that of a cabinet maker, a painter, a stonemason, ...
- Tip #2: Think! About Your Work
 - Everyday
- Kaizen
 - Make small improvements every day

Chapter 1

- The Cat Ate My Source Code
 - Who is responsible for you and your actions?
 - You
 - Not your parents, your professors, your boss
 - Not your friends, your spouse, your kids
 - Life will always be unfair and uncontrollable
 - You cannot choose what happens to you
 - You can choose **your** actions and how you respond
- Tip #3: Provide Options, Don't Make Lame Excuses

Tip 4

- Don't Live with Broken Windows
- What are Broken Windows in Software?
 - Bad designs, wrong decisions, poor code, no tests; beginning of *software entropy*
- If you don't have time to fix it what do you do?
 - Board it up, i.e. remove it or declare it as “not implemented”

Stone Soup and Frog Soup

- Summarize the Stone Soup story
- How does it relate to software?
 - Tip 5: Be a catalyst for change
 - Start something good, show people and say, “of course it would be better if we added...”
- Summarize the Boiling Frog story
- How does it relate to software?
 - Tip 6: Remember the Big Picture

Your Knowledge Portfolio

- How is your software knowledge portfolio similar to a stock portfolio?
 - Invest regularly
 - Diversify
 - Manage risk
 - Buy low, sell high
 - Review and rebalance

Tip 10

- It's both what you say and the way you say it