Question 2: Tell me about a book, blog, article, or GitHub repo you read or liked recently, and why you liked it and why you should recommend I do the same.

A book I have read recently that I feel is important is Adaptive Code by Gary McLean Hall. It is a review of agile coding with design patterns and SOLID principles.

Most coders can probably tell you what SOLID principles are, and yet I have recently encountered code that broke those principles in a way that prevented me from completing a DEVOPS project. This book goes through some re-factoring examples and tries to illustrate why loose decoupling is important to create maintainable, extendable code, breathing life into SOLID principles. It also heavily stresses the importance of TDD ( test-driven design), TFD (test-first development), and unit tests in terms of the design/development process.

It covers some of the Gang Of Four design patterns, but not in an exhaustive manner. There are probably better books for that. Still, it is important to know why a developer would choose one of these patterns, and I feel this book does a good job in describing that aspect of coding. It also stresses that re-factoring is sometimes an art, and you need to recognize when you have gone far enough.