

Design Strategies

-----Karunam Goyal-----

#138935

1. Loosely Coupled ask questions. Single Responsibility.
2. Highly Cohesive and loosely coupled == Single Responsibility.
3. Wait for more information. Do not take action prematurely asses future cost
4. DRY - Don't Repeat Yourself
5. Depend on Behaviour not Data.
6. Encourage Reuse
7. Use dependency injection
8. Isolate dependencies
9. Asking for what instead of telling hows
10. Put best interface
11. Talk to your immediate neighbours
12. MetaProgramming is useful but to be used carefully
13. Use dynamic typing it has a little tradeoff
14. Promotes Abstraction
15. Use composition
16. Open for extension closed for modification promoted by inheritance
17. Refactoring - Does not alter its Behaviour
18. Tests Provide only reliable documentation of design
19. Use Test To Document Roles.
20. Use Mocks Mocks are a test of Behaviour.
21. Liskov Sub P subtype should be substitutable for their super type.
22. The best tests are loosely coupled to the underlying code and test everything once and in the proper place.