

Clean Code



Black grade – Shows interest in CCD principles

PRINCIPLES

Don't repeat yourself
Keep it simple, stupid
Beware of optimization!
Composition over inheritance

Red Grade

Boy scout rule
Root cause analysis
Version control system
Simple refactorings
Reflect daily

PRACTICES

PRINCIPLES

Single level of abstraction
Single responsibility principle
Separation of concerns
Source code conventions

Orange Grade

Issue tracking
Automated int.-tests
Read, Read, Read
Reviews

PRACTICES

PRINCIPLES

Interface segregation principle
Dependency inversion principle
Liskov substitution principle
Principle of least astonishment
Information hiding principle

Yellow Grade

Automated unit tests
Mockups
Code coverage analyse
Conferences
Complex refactoring

PRACTICES

PRINCIPLES

Open closed principle
Tell, don't ask
Law of Demeter

Green Grade

Continuous Integration
Static code analysis
Inversion of control container
Pass experiences
Measure bugs

PRACTICES

PRINCIPLES

Design and implementation
do not overlap
Implementation reflects design
You ain't gonna need it (YAGNI)

Blue Grade

Continuous Deployment
Iterative development
Component oriented
Tests first

PRACTICES