

SOLID principles

- <http://butunclebob.com/ArticleS.UncleBob.PrinciplesOfOod>
 - Articles also available in 'References' section of: <https://en.wikipedia.org/wiki/SOLID>

Tests – TDD

- <http://misko.hevery.com/code-reviewers-guide/>
- <http://misko.hevery.com/attachments/Guide-Writing%20Testable%20Code.pdf>
- [YouTube's search:](#)
 - misko hevery - How to Write Clean, Testable Code
 - https://www.youtube.com/watch?v=XcT4yYu_TTs
 - misko hevery - The Clean Code Talks -- Inheritance, Polymorphism, & Testing
 - <https://www.youtube.com/watch?v=4F72VULWFvc&list=PLED6CA927B41FF5BD>
- <https://martinfowler.com/articles/mocksArentStubs.html>
- <https://blog.cleancoder.com/uncle-bob/2014/05/10/WhenToMock.html>
- <https://blog.cleancoder.com/uncle-bob/2014/05/14/TheLittleMocker.html>
- Extreme Programming Explained, Embrace Change – Kent Beck
- xUnit Test Patterns, Refactoring Test Code – Gerard Meszaros
- https://www.verifysoft.com/fr_code_coverage_levels.html
- [Google search:](#) Mock Brittle tests

Patterns and anti patterns

- Design Patterns: Elements of Reusable Object-Oriented Software - GOF - Erich Gamma, Richard Helm, Ralph Johnson et John Vlissides
- Design Patterns For Dummies - Steve Holzner

Getter-setter

- <http://typicalprogrammer.com/doing-it-wrong-getters-and-setters>
- <https://www.javaworld.com/article/2073723/why-getter-and-setter-methods-are-evil.html>
- <https://martinfowler.com/bliki/GetterEradicator.html>

Singleton

- [Google search:](#) singleton consider evil, harmful
<https://blog.cleancoder.com/uncle-bob/2014/05/10/WhenToMock.html>

Immutable

Object
<https://blog.cleancoder.com/uncle-bob/2014/05/10/WhenToMock.html>

- [Google search:](#) immutable object

Refactoring

- Refactoring: Improving the Design of Existing Code - Martin Fowler, Kent Beck (Contributor), John Brant (Contributor), William Opdyke, don Roberts

Java

- Effective Java: Programming Language Guide - Joshua Bloch

Misc

- Working Effectively with Legacy Code - Michael C. Feathers
- The Pragmatic Programmer - Andrew Hunt, David Thomas
- Code Complete – Steve McConnell
- The Mythical Man-Month - Brooks Jr., Frederick P.
- Out of the Tar Pit - Ben Moseley, Peter Marks
- The art of computer Programming – Donald Knuth

Domain Driven Design

- Domain-Driven Design Vite fait - Abel Avram & Floyd Marinescu
- Domain-Driven Design: Tackling Complexity in the Heart of Software - Eric Evans

Functional programming

- Quick articles:

<http://blog.jenkster.com/2015/12/what-is-functional-programming.html>

<https://maryrosecook.com/blog/post/a-practical-introduction-to-functional-programming>

<https://www.smashingmagazine.com/2014/07/dont-be-scared-of-functional-programming/>

<https://blog.cleancoder.com/uncle-bob/2012/12/22/FPBE1-Whats-it-all-about.html>

https://wiki.haskell.org/Functional_programming

- Structure and Interpretation of Computer Programs - Harold Abelson and Gerald Jay Sussman with Julie Sussman
 - <https://mitpress.mit.edu/sites/default/files/sicp/index.html>
 - <https://www.youtube.com/playlist?list=PLE18841CABEA24090>
 - <https://berkeley-cs61as.github.io/index.html>

- [YouTube's search:](#)

Anjana Vakil

Anjana Vakil: Immutable data structures for functional JS

<https://www.youtube.com/watch?v=Wo0qiGPSV-s>

More OOP

- Object-Oriented Analysis and Design with Applications - Graddy boch, Robert Maksimchuk, Michael Engle
- Object-Oriented Software Engineering: A Use Case Driven Approach - Ivar Jacobson
- Information hiding (vs encapsulation):
 - Protected Variation: The Importance of Being Closed - Craig Larman
<https://martinfowler.com/ieeeSoftware/protectedVariation.pdf>
 - 1971 - On the criteria to be used in decomposing systems into modules - David Lorge Parnas
<https://prl.ccs.neu.edu/img/p-tr-1971.pdf>
- [Google search:](#) Traits/mixin

Architectures

Clean architecture

- <https://blog.cleancoder.com/uncle-bob/2012/08/13/the-clean-architecture.html>

MVC

- <http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html>

DCI

- <https://klevas.mif.vu.lt/~donatas/Vadovavimas/Temos/DCI/2009%20The%20DCI%20Architecture%20-%20A%20New%20Vision%20of%20OOP.pdf>
- <https://folk.uio.no/trygver/2008/commonsense.pdf>
- <http://fulloo.info/Documents/CoplienReenskaugASA2012.pdf>

GUI architectures

- http://griffon-framework.org/tutorials/5_mvc_patterns.html

Fowler on GUI architecture

- <https://martinfowler.com/eaDev/uiArchs.html>
- <https://martinfowler.com/eaDev/OrganizingPresentations.html>

MVP

- MVP: Model-View-Presenter, The Taligent Programming Model for C++ and Java - Mike Potel
<https://www.wildcrest.com/Potel/Portfolio/mvp.pdf>
- TWISTING THE TRIAD, The evolution of the Dolphin Smalltalk MVP application framework
- Andy Bower, Blair McGlashan
<http://www.object-arts.com/downloads/papers/TwistingTheTriad.PDF>

MVVM

- <https://blogs.msdn.microsoft.com/johngossman/2005/10/08/introduction-to-modelviewviewmodel-pattern-for-building-wpf-apps/>

Boundary – Control – Entity

- Search it with google, or read the Ivar Jacobson book in “More OOP” section

Uncle Bob (Robert C. Martin)

- Agile Principles, Patterns, and Practices in C# - Martin C. Robert, Martin Micah
- The Clean Coder, A Code Of Conduct For Professional Programmers - Robert C. Martin
- Clean Code, A Handbook of Agile Software Craftsmanship - Robert C. Martin

- Clean Architecture, a craftsman's guide to software structure and design - Robert C. Martin
- <http://blog.cleancoder.com/>
- <https://sites.google.com/site/unclebobconsultingllc/>
- Old blog: <http://butunclebob.com/ArticleS.UncleBob>
- [YouTube's search:](#) Uncle bob

Martin Fowler

- <https://martinfowler.com/>

Rich Hickey

- [YouTube's search:](#)
- Simple Made Easy 2012 - Rich Hickey
<https://www.youtube.com/watch?v=oytL881p-nQ>
- The Value of Values with Rich Hickey
<https://www.youtube.com/watch?v=-6BsiVyC1kM>
- Many other YouTube conferences.