

Schedule

Week 1

Mon • Goals:

- Basic comfort with Java
- Basic setup with IntelliJ
- (Java/OOP) Classes, fields, constructors, encapsulation
- (Java/OOP) methods, parameters, local variables, constants/final
- (Java/OOP) static methods
- Activity: 1 hour coding practice: grade reading and gpa computation
- Lesson Plan 1¹

Tue • Goals:

- Variable and function basic principles
- Basic refactorings: extract variable/field/method, rename, inline
- Function size principles (“extract till you drop”)
- (Coding) variable and method naming
 - Video: Clean Code (Remake)²
 - Video: Names++³
 - Reading: Clean Code chapters 1, 2
- (Coding) Extraction code into small functions
 - Video: Functions⁴
 - Reading: Clean Code chapter 3 ?????
- Lesson Plan 2⁵

Wed • Goals:

- Comment and Code formatting principles
- Function structure principles
- Prep:
 - Watch Function Structure video⁶
- (Coding) Function arguments, command-query separation
 - Video: Function Structure
 - Reading: Clean Code chapter 3 ?????
- (Coding) Comments, Formatting

¹[lessonPlans/lessonPlanDay1.html](#)

²[videos/01-clean_code.html](#)

³[videos/02-names.html](#)

⁴[videos/03-function_size.html](#)

⁵[lessonPlans/lessonPlanDay2.html](#)

⁶[videos/05-function_structure.html](#)

- Video: Clean Code, Form
- Reading: Clean Code, chapters 4, 5

- Thu**
- Goals:
 - Deep dive into extension mechanisms: inheritance and delegation
 - Discussion of the concept and value of polymorphism and information hiding
 - “Tell, don’t ask”
 - Prep:
 - Design Patterns book: 1.6 (maybe ?)
 - Clean Code, chapter 6
 - (Java/OOP) Inheritance vs Delegation
 - (Coding) Law of Demeter (Tell, don’t ask)

- Fri**
- Goals:
 - Code tracking with Version Control
 - Automated Testing and Test-driven development
 - (Practices) Version Control Basics
 - (Practices) Test-Driven Development
 - Prep:
 - Reading: Clean Code, chapter 9
 - Video: TDD Part 1 and 2

- For Later**
- Video: Advanced TDD Part 1?

Week 2

- Mon**
- Goals:
 - Exam 1
 - Introduction to Agile Methodologies
 - Introduction to the SOLID design principles and SRP
 - Introduce User Roles/Actors in context of SOLID principles
 - (Practices) Principles of Agile Development
 - Discussion of XP methodologies (pair programming, tests first etc)
 - (Principles) Single Responsibility Principle
 - Reading: ASD chapters 8
 - Video: In class
 - Prep:
 - Reading: ASD chapters 1-4

- Tue**
- Goals:

- Interfaces and dependency inversion (lamp example)
- Introduction to Open-Closed Principle
- (Principles) General values and principles for coding
- (Java/OOP) Interfaces
- (Principles) Open-Closed Principle
 - Reading: ASD chapter 9
 - Video: In class
- (Principles) Liskov Substitution Principle
 - Video: In class
 - Reading: ASD chapter 10
- (Practices) SOLID Principles Use Case (as in Clean Code video #14)
- Prep:
 - Reading: ASD chapters 7-8

- Wed**
- Goals:
 - Dependency Inversion Principle
 - Interface Segregation Principle
 - SOLID Principles Wrap-up Example
 - (Principles) Dependency Inversion Principle
 - Video: In class
 - Reading: ASD chapter 11
 - (Principles) Interface Segregation Principle
 - Video: In class
 - Reading: ASD chapter 12

- Thur**
- Goals:
 - Advanced TDD
 - Clean Tests
 - Test Design

- Fri**
- Goals:
 - Test Process
 - Mocking
 - Code Review

Week 3

Design Patterns in the morning; Work on project in the afternoon

- Not sure where these goes:**
- (Java/OOP) Inner classes
 - (Coding) Exception Handling

- Reading: Clean Code chapter 7
- Goals:
 - Planning: User Stories, CRC Cards, Acceptance Tests (Move to Later)
- (Practices) User Stories
 - Video: Clean Code, Architecture, Use Cases and High Level design
- (Practices) CRC Cards

- Mon**
- (Patterns) Command
 - Video: Clean Code: Design Patterns
 - Reading: ASD chapter 13
 - (Patterns) Template Method, Strategy
 - Video: Strategy and Template Method Patterns
 - Reading: ASD chapter 14
 - (Patterns) Facade, Mediator
 - Video: Clean Code: Pattern Roundup
 - Reading: ASD chapter 15
 - (Practices) UML Class Diagrams
 - (Patterns) Factory
 - Video: Clean Code: Factories
 - Reading: ASD chapter 21
 - TODO: Refactoring earlier
 - (Coding) Refactoring: Basic extractions, Inline
 - Video: Function Screencast, Prime Number Generation
 - Reading: ASD chapter 5
 - Work on Project

Tue - (Patterns) Null Object - Video: Clean Code: Pile O'Patterns - Reading: ASD chapter 16 ~ - (Patterns) Singleton, Monostate - Video: Clean Code: Pile O'Patterns - Reading: ASD chapter 17 - (Coding) Refactoring: Moving, signature change - Work on Project

- Wed**
- Reading: Clean Code, chapter 17 (smells/heuristics)
 - Work on Project

- Thu**
- (Patterns) Observer
 - Video: The Observer Pattern
 - Reading: ASD chapter 24
 - (Patterns) Adapter, Bridge, Proxy
 - Video: Clean Code: Pile O'Patterns
 - Reading: ASD chapters 25, 26

- Work on Project

- Fri**
- (Patterns) Composite
 - Reading: ASD chapter 23
 - (Practices) UML Sequence Diagrams
 - Work on Project

Week 4

- Mon**
- (Java/OOP) packages, modules (earlier?)
 - (Patterns) Visitor
 - Reading: ASD chapter 28
 - Work on Project
- Tue**
- (Patterns) State
 - Video: Finite State Machines and State Pattern
 - Reading: ASD chapter 29
 - Work on Project
- Wed**
- Work on Project
- Thu**
- Work on Project
- Fri**
- Work on Project