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

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)
  - Video: Names++
  - Reading: Clean Code chapters 1, 2
- (Coding) Function arguments, command-query separation
  - Video: Functions
  - Reading: Clean Code chapter 3

Wed

- Goals:
  - Comment and Code formatting principles
  - Deep dive into extension mechanisms: inheritance and delegation
  - Discussion of the concept and value of polymorphism and information hiding
  - "Tell, don't ask"
- (Java/OOP) Inheritance vs Delegation
- (Coding) Comments, Formatting
  - Video: Clean Code, Form
  - Reading: Clean Code, chapters 4, 5
- (Coding) Law of Demeter (Tell, don't ask)
  - Reading: Clean Code, chapter 6

Thu

- Goals:
  - Introduction to Agile Methodologies

  - Planning: User Stories, CRC Cards, Acceptance Tests
- (Practices) Principles of Agile Development
  - Reading: ASD chapters 1-3
  - Discussion of XP methodologies (pair programming, tests first etc)
- (Practices) User Stories
  - Video: Clean Code, Architecture, Use Cases and High Level design
- (Practices) CRC Cards

Fri
- Goals:
  - Project and code tracking: Version Control, issue management
  - Automated Testing and Test-driven development
- (Practices) Version Control Basics
- (Practices) Issue Management
- (Practices) Test-Driven Development
  - Reading: Clean Code, chapter 9
  - Reading: ASD, chapter 4
  - Video: TDD Part 1 and 2
  - Video: Advanced TDD Part 1?

Week 2

Mon
- Goals:
  - Interfaces and dependency inversion (lamp example)
  - Introduction to the SOLID design principles, SRP and Open-Closed Principle
- Exam 1?
- (Principles) General values and principles for coding
- (Java/OOP) Interfaces
- (Principles) Single Responsibility Principle
  - Reading: ASD chapter 8
  - Video: In class
- (Principles) Open-Closed Principle
  - Reading: ASD chapter 9
  - Video: In class

Tue
- (Principles) Liskov Substitution Principle
  - Video: In class
  - Reading: ASD chapter 10

- (Principles) Dependency Inversion Principle
  - Video: In class
  - Reading: ASD chapter 11
- (Principles) Interface Segregation Principle
  - Video: In class
  - Reading: ASD chapter 12 Wed
- (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
- Work on Project

Thu
- (Practices) Version Control Advanced
- Work on Project

Fri
- (Java/OOP) Inner classes
- (Coding) Exception Handling
  - Reading: Clean Code chapter 7

Week 3

Mon
- (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