

# Schedule

## Week 1

- Mon**
- Goals:
    - Basic comfort with Java
    - Basic setup with IntelliJ
  - Topics:
    - (Java/OOP) classes, fields, constructors, encapsulation
    - (Java/OOP) methods, parameters, local variables, constants/final
    - (Java/OOP) static methods
  - Activities for Day 1<sup>1</sup>
  - Lunchtime Video: Clean Code-Remake (54m)<sup>2</sup>
- Tue**
- Goals:
    - Variable and function basic principles
    - Basic refactorings: extract variable/field/method, rename, inline
    - Function size principles (“extract till you drop”)
  - Prep:
    - Read *Clean Code*<sup>3</sup>, chapters 1-3
    - Watch Names++<sup>4</sup>
  - Topics:
    - (Coding) naming variables and methods
    - (Coding) extracting code into small functions
  - Activities for Day 2<sup>5</sup>
  - Lunchtime Video: Refactoring screencast “2”: Prime generator<sup>6</sup>
- Wed**
- Goals:
    - Comment and Code formatting principles
    - Function structure principles
  - Prep:
    - Read *Clean Code*<sup>7</sup>, chapters 4, 5
    - Watch Function Structure video selected parts<sup>8</sup>
  - Topics:

---

<sup>1</sup>[lessonPlans/lessonPlanDay1.html](#)

<sup>2</sup>[videos/01-clean\\_code.html](#)

<sup>3</sup><https://learning.oreilly.com/library/view/clean-code/9780136083238/>

<sup>4</sup>[videos/02-names.html](#)

<sup>5</sup>[lessonPlans/lessonPlanDay2.html](#)

<sup>6</sup>[videos/04-prime\\_gen\\_screencast.html](#)

<sup>7</sup><https://learning.oreilly.com/library/view/clean-code/9780136083238/>

<sup>8</sup>[videos/05-function\\_structure.html](#)

- (Coding) function parameters, command-query separation
- (Coding) comments and formatting
- Activities for Day 3<sup>9</sup>
- Lunchtime Video: Form<sup>10</sup>

- 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:
    - Read *Design Patterns*, chapter 1.6
    - Read *Clean Code*<sup>11</sup>, chapter 6
  - Topics:
    - (Java/OOP) Inheritance vs Composition
    - (Coding) Law of Demeter (Tell, don’t ask)
  - Activities for Day 4<sup>12</sup>
  - Lunchtime Video: TDD Part 1<sup>13</sup>

- Fri**
- Goals:
    - Code tracking with Version Control
    - Automated Testing and Test-driven development
  - Prep:
    - Read *Clean Code*<sup>14</sup>, chapter 9
  - Topics:
    - (Practices) Version Control Basics
    - (Practices) Test-Driven Development
  - Activities for Day 5<sup>15</sup>
  - Lunchtime Video: TDD Part 2<sup>16</sup>

## Week 2

- Mon**
- Goals:
    - Exam 1

---

<sup>9</sup>[lessonPlans/lessonPlanDay3.html](#)

<sup>10</sup>[videos/06-form.html](#)

<sup>11</sup><https://learning.oreilly.com/library/view/clean-code/9780136083238/>

<sup>12</sup>[lessonPlans/lessonPlanDay4.html](#)

<sup>13</sup>[videos/07-tdd\\_part1.html](#)

<sup>14</sup><https://learning.oreilly.com/library/view/clean-code/9780136083238/>

<sup>15</sup>[lessonPlans/lessonPlanDay5.html](#)

<sup>16</sup>[videos/08-tdd\\_part2.html](#)

- 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 7, 8
  - Videos: In class: notes #1 and #2
- Activities for Day 6<sup>17</sup>
- 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)
- Activities for Day 7<sup>18</sup>
- 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

---

<sup>17</sup>[lessonPlans/lessonPlanDay6.html](https://lessonPlans/lessonPlanDay6.html)

<sup>18</sup>[lessonPlans/lessonPlanDay7.html](https://lessonPlans/lessonPlanDay7.html)

- Reading: ASD chapter 12

- Thur**
- Goals:
    - Advanced TDD
    - Clean Tests
    - Test Design
  - Video: Advanced TDD Part 1?

- 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 Screenshot, 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