



CODING AKADEMIE
MÜNCHEN

Java Clean Code

Programmieren nach dem „Reinheitsgebot“

Das Clean Code Seminar ist das Seminar für Einsteiger und Fortgeschrittene, die den eigenen oder andere Codes in einen deutlich besseren Code verwandeln möchten. Ein besserer Code bedeutet, klar strukturiert, sehr gute Lesbarkeit, kompakt und effizient. Ein solcher Code weist hohe Qualität auf und zeichnet sich dadurch aus, dass er leicht zu verstehen und zu erweitern ist. Damit ist dieser sogenannte Clean Code sehr flexibel, was auch große Entwicklerteams schnell auf neue Anforderungen oder Änderungen reagieren lässt

Das Java Clean Code Seminar der Coding Akademie München setzt – wie der Name bereits sagt – voll auf klare Programmierung. Denn auch die komplexesten Anwendungen lassen sich mit einem Clean Code schneller, sicherer und vor allem für andere nachvollziehbar programmieren. Das ist gut für Ihre Mitarbeiter, Ihren Arbeitgeber oder Ihre Kunden, die auf Ihrem Code aufbauen möchten. Ein echter Java Clean Code ist aber in allererster Linie wichtig für Sie selbst. Denn wenn Sie sich nach Monaten oder Jahren nicht erst wieder einarbeiten müssen, gewinnen Sie dadurch wertvolle Zeit. Die können Sie künftig beispielsweise auch entspannt im Urlaub verbringen. Schließlich kann jeder Programmierer auf Ihren sauber vorbereiteten Code aufsetzen und generiert Ihnen damit Ihre wohlverdiente Freizeit.

Dass ein Java Clean Code nicht schwer zu lernen ist, sehen Sie spätestens beim Blick auf die Semindauer. An nur drei Tagen erfahren Sie alle goldenen Regeln für einen hochwertigen Code. Unter Anleitung von Java Experten und ausgewiesenen Profis Ihres Fachs lernen Sie einen kompakten und klar strukturierten Code zu produzieren. Damit eliminieren Sie den sogenannten Spagetti-Code, der schwer lesbar oder erweiterbar ist.

Genauso strukturiert, wie Ihr späterer Java Clean Code wird auch Ihr Seminar in München sein. Nach einer kurzen theoretischen Einleitung starten wir direkt in die Praxis. Anhand realer Fallbeispiele setzen wir den Code strukturiert auf, beleuchten, wie existierender Code verbessert und folgender vereinfacht werden kann. Das alles in angenehmer Atmosphäre in unseren modernen Schulungsräumen. Und an besten Rechnern. Sie sehen: Das Java Clean Code Seminar der Coding Akademie München passt zu Ihnen, wie guter Code zu erfolgreichen Projekten.

Inhalte

Clean Code Seminar Inhalt

Der Clean Code Kurs ist ein praxisorientierter Kurs, basierend auf dem Standardwerk „Clean Code“ von Robert Martin. Kombiniert mit unseren jahrelangen Erfahrungen aus Industrie- und Akademie.

- Code Rote
- Clean Code
- Names Rules
- Function Rules
- Exception Rules
- Good Comments
- Bad Comments
- Object-Oriented Programming
- SOLID Principle
- Boundaris
- Unit Tests
- Test Driven Development TDD
- Important Refactoring Patterns
- Code Smells: Names
- Code Smells: Function
- Code Smells: Comments
- Code Smells: Test
- Über 15 kleine Workshop
- Abschluss Projekt

Agenda

Der Java Grundkurs wurde für 3 Tage konzipiert. Innerhalb dieser drei intensiven Tage vermitteln wir Ihnen ausnahmslos alles, was Sie als Java Einsteiger benötigen. Das spart Zeit und damit bares Geld. Für jedes Thema gibt es Beispiele und Programmieraufgaben. Die Theorie Anteil beträgt 20% und die Praxisanteil von 80%. Es gibt über 14 kleine Workshops und am dritten Tag ein Abschluss-Projekt
 Folgende Themen lernen Sie in unserem Java Grundkurs für Anfänger

Clean Code Tag 1

Code Rote

- The Company killed by Code
- Productivity Problem
- Manager Reaction
- Code Rote: Rigidity
- Code Rote: Fragility
- Code Rote: Inseparability
- Code Rote: Opacity

Clean Code

- Clean Code after Grady Boosch
- Clean Code after Bjarne Stroustrup
- Clean Code after Ward Cunningham
- Clean Code after Michael Feathers

Name Rules

- What is a good name?
- What is a bad name?
- Ottinger's Rules for good names
- Intention-Revealing Names
- Pronounceable Names
- Searchable Names
- Describe the Problem not the Implementation
- Avoid Disinformation
- Make Meaningful Distinctions
- Avoid Encodings like Hungarian Notation
- Choose Parts of Speech Well
- Use Scope Length Rules

Function Rules

- What is a function?
- Function Grammars
- Big Function
- Extract till you drop
- Step-Down Rule
- Function Type Rules
- Command-Query Separation Rule
- Command-Query Separation Rule in Threads
- Tell, Don't Ask Principle
- Train Wrecks
- The Law of Demeter
- Parameter List Rules
- Use Maximal three Arguments
- No Output Argument Rule
- No Boolean Argument Rule
- No null Argument Rule
- Statement Block Rules
- Do One Thing Rule
- One Level of Abstraction Rule
- Small Function Rule
- Big Functions hide Classes Rule
- Minimize Side-Effects
- Avoid and Minimize Switth Statement

Error Handling Rules

- What is Exception and Error Handing?
- Error Handling First Rule
- Prefer Exception over special Values
- Use the most suitable Exception Classes
- Use Unchecked Exceptions (RuntimeException

Good Comments

- Good Comments
- Legal Comments
- Informative Comments
- Explanation of Intent
- Clarification
- Warning of Consequences
- TODO Comments
- Amplification
- Javadocs in Public APIs

Bad Comments

- Mumbling
- Redundant Comments
- Misleading Comments
- Mandated Comments
- Journal Comments
- Noise Comments
- Scary Noise

Clean Code Tag 2

Object-Oriented Programming

- What is Object-Oriented Programming?
- Dependency Management in OOP
- Dependency Inversion
- Switch Statement and OOP
- Switch Statement and the Fan-Out Problem

Solid Principles

- Single Responsibility Principle
- Open/Closed Principle
- Liskov Substitution Principle
- Interface Segregation Principle
- Dependency Inversion Principle

Boundries

- The Main Area
- Learning Boundries
- Clean Boundaries

Unit Test

- Introduction to JUnit 5
- Test Life Cycle
- Junit 5 Annotations
- Junit 5 Assertion Methods

Test-Driven Development TDD

- What is Test-Driven Development TDD?
- RED Phase
- GREEN Phase
- REFACTORING Phase

Imortant Refactoring Patterns

- Rename Pattern
- Move-to Pattern
- Extract Method Pattern
- Extract Class from a Big Method Pattern
- Extract an Interface Pattern

Clean Code Tag 3

Code Smells: Names

- Choose Descriptive Names
- Choose Names at the Appropriate Level of Abstraction
- Use Standard Nomenclature Where Possible
- Unambiguous Names
- Use Long Names for Long Scope
- Avoid Encodings
- Name Should Describe Side-Effects

Code Smells: Functions

- Too Many Arguments
- Output Arguments
- Flag Arguments
- Dead Function

Code Smells: Comments

- Inappropriate Information
- Obsolete Comment
- Redundant Comment
- Poorly Written Comment
- Commented-Out Code

Code Smells: Test

- Insufficient Tests
- Use a Coverage Tool
- Don't Skip Trivial Tests
- Test Ambiguity
- Test Boundary Conditions
- Exhaustively Test Near Bugs
- Patterns of Failure Are Revealing
- Test Coverage Patterns Can Be Revealing
- Tests Should Be Fast

Nutzen

Nutzen & Vorteile des Java Clean Code Kurses

- Deutliche bessere Qualität Ihrer Codes
- Geringere Einarbeitungszeit für neue Mitarbeiter
- Schnellere Programmierungs-Erfolge dank leicht adaptierbarem Clean Code
- Die starke Abhängigkeit von Top-Programmierern reduzieren
- Verständlicher Code macht Übergaben bspw. zum anstehenden Urlaub einfacher

Ziele

Nach dem Java Clean Code Seminar in München werden Sie

- Die goldenen Regeln zum Produzieren von hochqualitativem Java Code meistern
- Einen deklarativen und qualitativ hochwertigen "Java Clean Code" schreiben
- Schlechten Code anhand von „Code Smells“ Regel erkennen und eliminieren
- Existierenden Java Code optimieren und besser strukturieren
- Debugging und Refactoring Aufwand minimieren
- Produktivität, Leistung und Qualität deutlich maximieren

Zielgruppe

Software-Entwickler und Programmierer

Der Java Clean Code Kurs wurde konzipiert für Software-Entwickler und Programmierer mit soliden und guten Java Kenntnissen. Zuvor haben Sie vielleicht unseren Java Grundkurs, das Java Aufbau-seminar oder ein anderes Clean Code Seminar besucht. Allerdings ist das keine Voraussetzung. Lediglich Ihr Umgang mit den theoretischen Java Strukturen sollte Ihnen geläufig sein.

Zertifikat

Krönender Abschluss Ihres Java Clean Code Seminars ist die feierliche Übergabe Ihres persönlichen Zertifikates. Das werden Sie sich nach den drei Seminartagen leicht verdient haben. Schließlich gelingt Ihnen die Abschluss-Arbeit garantiert im Handumdrehen. Als Coding Zertifikat Inhaber sind Sie auf dem Markt mehr gefragt denn je. Denn gute Programmierer gibt es viele – aber nur wenige Programmierer mit gutem Clean Code. Darum ist das Zertifikat bei Arbeitgebern und Seminarteilnehmern gleichermaßen begehrt.

Inklusive

Seminarunterlagen und Verpflegung

Kursdauer

3 Tage, je 8 Stunden pro Tag

Maximale Teilnehmeranzahl:

Max. Teilnehmer

8

Die nächsten Termine

Mo., 13.01. - Mi., 15.01.2020

[Seminar buchen](#)