

Objektorientierte Softwareentwicklung: Spezielle Programmierprinzipien mit C++ - der Weg zum Clean Code

Ziele - Ihr Nutzen

Sie lernen die wichtigsten Regeln der Softwareentwicklung und ihre Bedeutung sowie Muster kennen, nach denen Sie Ihre Codestruktur verbessern können.

Teilnehmer

Software-Entwickler, Software-Architekten

Voraussetzungen

Kenntnisse einer objektorientierten Programmiersprache (Übungen werden wahlweise in C++ oder C# durchgeführt)

Objektorientierte Softwareentwicklung: Spezielle Programmierprinzipien mit C++ - der Weg zum Clean Code

Inhalt

Softwareentwicklung als Handwerkskunst - 'Software Craftmanship' - Der Weg zum Clean Code

Warum ist guter Code wichtig?

- Was zeichnet guten Code aus?
- Welche Probleme verursacht schlechter Code?
- Was sind die Ursachen für schlechten Code?
- Warum ist es sinnvoll, auf guten Code Wert zu legen?
- Wie entsteht guter Code?

Grundregeln zur Erstellung guten Codes

- DRY - Don't Repeat Yourself
- KISS - Keep it simple, stupid
- Geheimnisprinzip
- Programming to an Interface
- Modularisierung
- Prinzip der losen Kopplung
- Prinzip der hohen Kohäsion
- Vorsicht vor Optimierungen
- POLS - Principle of Least Surprise
- Übungen zum besseren Verständnis der Prinzipien

Die SOLID-Prinzipien

- Single-Responsibility-Prinzip
- Open-Closed-Prinzip
- Liskovsches Substitutionsprinzip
- Interface-Segregation-Prinzip
- Dependency-Inversion-Prinzip
- Übungen zum besseren Verständnis der Prinzipien

Weitere Prinzipien

- FCol - Favour Composition over Inheritance
- SLA - Single Level of Abstraction
- Tell don't ask
- Law of Demeter
- YAGNI - You Ain't Gonna Need It

- Nutze Source Code Konventionen
- Übungen zum besseren Verständnis der Prinzipien

Refaktorisierung von Code

- Was ist Refaktorisierung?
- Welche Arten gibt es?
- Wie wird eine Refaktorisierung durchgeführt?
- 'Smells', die auf die Notwendigkeit einer Refaktorisierung hinweisen
- Refaktorisierungspatterns
- Übung: Finden von 'Smells' im Code
- Übungen zum Refactoring: Einsatz ausgewählter Patterns

Hinweise zur Verbesserung der Codequalität im Projekt

- Wie lässt sich Bewusstsein für guten Code schaffen?
- Wie lässt sich Code kontinuierlich verbessern?

Praktische Übungen

- Übungen zum besseren Verständnis der Programmierprinzipien
- Finden von 'Code-Smells'
- Übungen zum Einsatz von Refactoring-Patterns

MicroConsult PLUS

- Als Teilnehmer haben Sie die folgenden Möglichkeiten, um Ihre Übungsverzeichnisse und Lösungsbeispiele für alle Übungsaufgaben mitzunehmen:
- Sie nehmen die Dateien direkt auf einem von MicroConsult bereitgestellten kostenfreien USB-Stick mit, oder
- Sie senden sich die Dateien selbst per E-Mail zu, oder
- Sie erhalten auf Anfrage Zugriff auf die Dateien per Download.

Offenes Training

★ Mit Durchführungsgarantie

Termin	Preis *	Dauer
18.11.2019 – 20.11.2019	1.650,00 €	3 Tage ★
09.03.2020 – 11.03.2020	1.650,00 €	3 Tage
25.05.2020 – 27.05.2020	1.650,00 €	3 Tage
14.09.2020 – 16.09.2020	1.650,00 €	3 Tage
14.12.2020 – 16.12.2020	1.650,00 €	3 Tage
22.03.2021 – 24.03.2021	1.650,00 €	3 Tage

* Preis je Teilnehmer, in Euro zzgl. USt.

Anmeldecode: OOP-FORCPP

Coaching

Unsere Coaching-Angebote bieten den großen Vorteil, dass unsere Experten ihr Wissen und ihre Erfahrungen direkt in Ihren Lösungsprozess einbringen und damit unmittelbar zu Ihrem Projekterfolg beitragen.

Für Ihre Anfrage oder weiterführende Informationen stehen wir Ihnen gern zur Verfügung.