TypeScript Advanced

Generics, Mapped Types, Template Literals und Branded Types – TypeScript, das Bugs findet, bevor der Code läuft.

1 ODER 2 TAGE | EXPERTEN | REMOTE ODER INHOUSE | DEUTSCH & ENGLISCH | BIS ZU 12 TEILNEHMENDE (MEHR AUF ANFRAGE)

FÜR WEN

- Entwickler, die täglich TypeScript nutzen und über die Grundlagen hinaus wollen
- Teams, die gemeinsame Libraries oder komplexe Domänenmodelle pflegen

DANACH KANN EUER TEAM

- Fachregeln im Typsystem statt in Laufzeitprüfungen abbilden
- Generische, wiederverwendbare Utilities mit präzisen Typen schreiben
- Fortgeschrittene Typen in Drittbibliotheken lesen und erweitern

VORAUSSETZUNGEN

- Praktische TypeScript-Erfahrung
- Sicher im Umgang mit Interfaces, Unions und Generics-Grundlagen

Bringt eure TypeScript-Kenntnisse mit diesem TypeScript Advanced Training auf das nächste Level. Versteht tiefer, wie TypeScript funktioniert und wie ihr fortgeschrittene Konzepte und Typen in euren Softwareprojekten einsetzt. Robusterer, ausdrucksstärkerer Code und bewährte Best Practices verbessern eure Developer Experience, machen euch und euer Team effizienter und reduzieren den Wartungsaufwand.

AGENDA

Reihenfolge und Tiefe der Blöcke passen wir an euer Team an – was ihr schon beherrscht, lassen wir weg, eure Fragen bekommen die Zeit.

01 **Grundlagen**
Eine kurze Zusammenfassung der grundlegenden Möglichkeiten und Grenzen von TypeScript.

02 **Typsystem**
Typsysteme verschiedener Sprachen kennenlernen und verstehen, wie das Typsystem von TypeScript funktioniert.

03 **Advanced Types**
Flexiblere und ausdrucksstärkere Typen mit Union und Intersection Types erstellen.

04 **Utility Types**
Die Utility Types von TypeScript nutzen, die einige der häufigsten Typ-Transformationen abdecken.

05 **Never**
Das Konzept des Never Type verstehen und TypeScript damit mitteilen, welche Situationen niemals eintreten dürfen.

06 **Generics**
Generics meistern: ein Deep Dive in Generic Constraints, das Ableiten von Typen aus anderen Typen und Typ-Rekursion.

07 **Literals & Template Literals**
Spezifischere Typen als nur string und number einsetzen, sie kombinieren und TypeScript gezielt anweisen, Objekte und Arrays selbstständig auf ihren spezifischsten Typ einzugrenzen.

08 **Mapped Types**
Typen aus anderen Typen ableiten, um Redundanz zu vermeiden und Änderungen über zusammenhängende Typen hinweg durchzusetzen.

09 Type Guards

Spezifische Typen sicher erkennen und eingrenzen.

10 Moderne TypeScript-Features

Der satisfies-Operator, const-Typparameter, Standard-Decorators und using-Deklarationen – neuere Features, die Typen präziser und Code sicherer machen.

11 Branded Types

Ein fortgeschrittenes Konzept, um Typen für komplexere Muster wie E-Mail-Adressen oder Sozialversicherungsnummern zu erstellen.

EUER TRAINER



Tim Honermann

SENIOR SOFTWARE ENGINEER

Senior Software Engineer für Angular und Java, baut Full-Stack-Anwendungen für Kunden und schreibt darüber in seinem Blog.

TRAINING ANFRAGEN

- Nur für euer Team – Remote oder Inhouse
- bis zu 12 Teilnehmende (mehr auf Anfrage)
- ab CHF 2’400 pro Tag für das ganze Team
- Vor Ort: zuzüglich Reisekosten
- Fixes Angebot innerhalb eines Werktags
- Das Erstgespräch ist kostenlos und unverbindlich
- Deutsch oder Englisch
- Sitz bei Zürich – vor Ort in CH, DE, AT; remote überall

Training anfragen: **offering.solutions/de/training-typescript-advanced/** · mail@offering.solutions