

## TypeScript - Ein praktischer Einstieg

Typsicheres JavaScript für skalierbare Webanwendungen

# DAS INHALTS- VERZEICHNIS

» Hier geht's  
direkt  
zum Buch

<b>Vorwort Josh Goldberg</b> .....	<b>19</b>
<b>Vorwort zur deutschen Ausgabe</b> .....	<b>25</b>

---

## Teil I Konzepte

<b>1 Von JavaScript zu TypeScript</b> .....	<b>29</b>
Die Geschichte von JavaScript .....	29
Die Tücken von purem JavaScript .....	30
Kostspielige Freiheit .....	30
Schwache Dokumentation .....	31
Schwächere Entwicklertools .....	31
TypeScript! .....	32
Erste Schritte auf dem TypeScript-Spielplatz .....	32
TypeScript in Aktion .....	33
Freiheit durch Restriktion .....	33
Präzise Dokumentation .....	34
Bessere Entwicklertools .....	35
Kompilieren .....	36
Erste lokale Schritte .....	36
TypeScript lokal ausführen .....	37
Editor-Funktionen .....	38
Was TypeScript nicht ist .....	38
Ein Heilmittel für schlechten Code .....	39
Eine Erweiterung für JavaScript (größtenteils) .....	39
Langsamer als JavaScript .....	40
Ergebnis einer abgeschlossenen Entwicklung .....	40
Zusammenfassung .....	40

<b>2</b>	<b>Das Typsystem</b> .....	<b>43</b>
	Was ist ein Typ? .....	43
	Typsysteme .....	45
	Fehlerarten .....	46
	Zuweisbarkeit .....	47
	Zuweisbarkeitsfehler verstehen .....	47
	Typanmerkungen .....	48
	Unnötige Typanmerkungen .....	49
	Typformen .....	50
	Module .....	51
	Zusammenfassung .....	53
<b>3</b>	<b>Vereinigungs- und Literaltypen</b> .....	<b>55</b>
	Vereinigungstypen .....	55
	Deklaration von Vereinigungstypen .....	56
	Eigenschaften von Vereinigungstypen .....	56
	Type Narrowing .....	57
	Type Narrowing durch Zuweisung .....	57
	Bedingungsabhängige Prüfungen .....	58
	typeof-Prüfungen .....	59
	Literaltypen .....	59
	Zuweisbarkeit von Literaltypen .....	61
	Strikte Nullprüfung .....	61
	Der Milliarden-Dollar-Fehler .....	62
	Type Narrowing durch Wahrheitsprüfung .....	63
	Variablen ohne Anfangswerte .....	64
	Typalias .....	64
	Typalias sind kein JavaScript .....	65
	Typalias kombinieren .....	65
	Zusammenfassung .....	66
<b>4</b>	<b>Objekte</b> .....	<b>67</b>
	Objekttypen .....	67
	Deklaration von Objekttypen .....	68
	Aliasing von Objekttypen .....	68
	Strukturelle Typisierung .....	69
	Verwendungsprüfung .....	70
	Prüfung auf überzählige Eigenschaften .....	71
	Verschachtelte Objekttypen .....	72
	Optionale Eigenschaften .....	73
	Vereinigungen von Objekttypen .....	74
	Abgeleitete Vereinigungen von Objekttypen .....	74
	Explizite Vereinigungen von Objekttypen .....	75

Type Narrowing von Objekttypen .....	76
Diskriminierte Vereinigungstypen .....	77
Schnittmengentypen .....	78
Gefahren von Schnittmengentypen .....	79
Zusammenfassung .....	80

---

## Teil II Features

<b>5 Funktionen .....</b>	<b>83</b>
Funktionsparameter .....	83
Erforderliche Parameter .....	84
Optionale Parameter .....	85
Standard-Parameter .....	86
Restparameter .....	86
Rückgabetypen .....	87
Explizite Rückgabetypen .....	87
Funktionstypen .....	88
Klammersetzung bei Funktionstypen .....	90
Ableitung von Parametertypen .....	90
Typalias für Funktionen .....	91
Weitere Rückgabetypen .....	91
void zurückgeben .....	91
never zurückgeben .....	93
Überladung von Funktionen .....	93
Kompatibilität der Aufrufsignaturen .....	94
Zusammenfassung .....	95
<b>6 Arrays .....</b>	<b>97</b>
Array-Typen .....	98
Array- und Funktionstypen .....	98
Arrays mit Vereinigungstyp .....	98
Evolving-any-Arrays .....	99
Mehrdimensionale Arrays .....	99
Array-Elemente .....	100
Fallstrick: Unzuverlässige Ableitungen .....	100
Spreads und Restparameter .....	101
Spreads .....	101
Spreading von Restparametern .....	102
Tupel .....	102
Zuweisbarkeit von Tupeln .....	103
Typinferenz bei Tupeln .....	105
Zusammenfassung .....	107

<b>7</b>	<b>Interfaces</b> .....	<b>109</b>
	Typalias im Vergleich zu Interfaces .....	109
	Eigenschaftstypen .....	110
	Optionale Eigenschaften .....	111
	Schreibgeschützte Eigenschaften .....	111
	Funktionen und Methoden .....	113
	Aufrufsignaturen .....	114
	Indexsignaturen .....	115
	Verschachtelte Interfaces .....	118
	Erweiterung von Interfaces .....	119
	Überschreiben von Eigenschaften .....	120
	Gleichzeitige Erweiterung mehrerer Interfaces .....	120
	Verschmelzung von Interfaces .....	121
	Namenskonflikte bei Mitgliedern .....	122
	Zusammenfassung .....	123
<b>8</b>	<b>Klassen</b> .....	<b>125</b>
	Klassenmethoden .....	125
	Klasseneigenschaften .....	126
	Funktionseigenschaften .....	127
	Initialisierungsprüfung .....	128
	Optionale Eigenschaften .....	130
	Schreibgeschützte Eigenschaften .....	130
	Klassen als Typen .....	132
	Klassen und Interfaces .....	133
	Gleichzeitige Implementierung mehrerer Interfaces .....	134
	Eine Klasse erweitern .....	136
	Zuweisbarkeit erweiterter Klassen .....	136
	Überschreiben von Konstruktoren .....	137
	Überschreiben von Methoden .....	139
	Überschreiben von Eigenschaften .....	139
	Abstrakte Klassen .....	140
	Sichtbarkeit von Mitgliedern .....	141
	Statische Feld-Modifier .....	144
	Zusammenfassung .....	144
<b>9</b>	<b>Typ-Modifier</b> .....	<b>147</b>
	Top Types .....	147
	any, schon wieder .....	147
	unknown .....	148
	Typprädikate .....	149
	Typoperatoren .....	151
	keyof .....	151
	typeof .....	153

Typzusicherungen . . . . .	154
Zusicherung des Fehlertyps . . . . .	155
Nicht-null-Zusicherung . . . . .	156
Fallstricke bei Typzusicherungen . . . . .	157
Const-Zusicherungen . . . . .	159
Literale statt primitiver Datentypen . . . . .	160
Schreibgeschützte Objekte . . . . .	161
Zusammenfassung . . . . .	162
<b>10 Generics . . . . .</b>	<b>163</b>
Generische Funktionen . . . . .	164
Explizite generische Aufruftypen . . . . .	165
Multiple Typparameter für Funktionen . . . . .	166
Generische Interfaces . . . . .	167
Abgeleitete generische Interface-Typen . . . . .	168
Generische Klassen . . . . .	169
Explizite generische Klassentypen . . . . .	170
Erweiterung generischer Klassen . . . . .	171
Implementierung von generischen Interfaces . . . . .	172
Methoden-Generics . . . . .	173
Generics von statischen Klassenmitgliedern . . . . .	173
Generische Typaliase . . . . .	174
Generische diskriminierte Vereinigungstypen . . . . .	174
Generische Modifier . . . . .	175
Generische Standardwerte . . . . .	175
Eingeschränkte generische Typen . . . . .	177
keyof und eingeschränkte Typparameter . . . . .	178
Promises . . . . .	179
Promises erstellen . . . . .	179
Asynchrone Funktionen . . . . .	180
Generics richtig verwenden . . . . .	181
Die goldene Regel der Generics . . . . .	181
Namenskonventionen für Generics . . . . .	182
Zusammenfassung . . . . .	182

---

## Teil III Verwendung

<b>11 Deklarationsdateien . . . . .</b>	<b>187</b>
Deklarationsdateien . . . . .	187
Laufzeitwerte deklarieren . . . . .	188
Globale Werte . . . . .	190
Verschmelzung globaler Interfaces . . . . .	190
Globale Augmentationen . . . . .	191

Integrierte Deklarationen . . . . .	192
Bibliotheks-Deklarationen . . . . .	192
DOM-Deklarationen . . . . .	194
Moduldeklarationen . . . . .	195
Moduldeklarationen mit Platzhaltern . . . . .	195
Typen von Paketen . . . . .	196
declaration . . . . .	196
Typen von Dependency-Paketen . . . . .	197
Paket-Typen bereitstellen . . . . .	198
DefinitelyTyped . . . . .	199
Verfügbarkeit von Typinformationen . . . . .	200
Zusammenfassung . . . . .	200
<b>12 IDE-Funktionen verwenden . . . . .</b>	<b>203</b>
Im Code navigieren . . . . .	204
Definitionen suchen . . . . .	205
Verwendungen suchen . . . . .	206
Implementierungen suchen . . . . .	207
Code schreiben . . . . .	208
Namen vervollständigen . . . . .	208
Automatische Import-Updates . . . . .	209
Code-Aktionen . . . . .	210
Effektiv mit Fehlern umgehen . . . . .	213
Fehler des Sprachdiensts . . . . .	213
Zusammenfassung . . . . .	217
<b>13 Konfigurationsoptionen . . . . .</b>	<b>219</b>
tsc-Optionen . . . . .	219
Pretty-Mode . . . . .	220
Watch-Mode . . . . .	220
TSCConfig-Dateien . . . . .	221
tsc --init . . . . .	222
Kommandozeile oder Konfigurationsdatei? . . . . .	222
Dateien einschließen . . . . .	223
include . . . . .	223
exclude . . . . .	224
Alternative Dateiendungen . . . . .	224
JSX-Syntax . . . . .	224
resolveJsonModule . . . . .	226
Ausgabe von JavaScript . . . . .	227
outDir . . . . .	227
target . . . . .	228
Ausgabe von Deklarationsdateien . . . . .	229

Sourcemaps . . . . .	231
noEmit . . . . .	232
Typechecking . . . . .	232
lib . . . . .	232
skipLibCheck . . . . .	233
strict-Mode . . . . .	233
Module . . . . .	238
module . . . . .	239
moduleResolution . . . . .	239
Interoperabilität mit CommonJS . . . . .	240
isolatedModules . . . . .	241
JavaScript . . . . .	242
allowJs . . . . .	242
checkJs . . . . .	243
JSDoc-Unterstützung . . . . .	244
Konfigurationserweiterungen . . . . .	245
extends . . . . .	245
Basiskonfigurationsdateien . . . . .	246
Projektreferenzen . . . . .	247
composite . . . . .	248
references . . . . .	248
Build-Mode . . . . .	249
Zusammenfassung . . . . .	250

---

## Teil IV Zugaben

<b>14 Syntaxerweiterungen . . . . .</b>	<b>255</b>
Parametereigenschaften in Klassen . . . . .	256
Experimentelle Dekoratoren . . . . .	258
Enums . . . . .	259
Automatische Zuordnung numerischer Werte . . . . .	261
Enums mit String-Werten . . . . .	262
Const-Enums . . . . .	263
Namensräume . . . . .	264
Namespace-Exporte . . . . .	265
Verschachtelte Namensräume . . . . .	267
Namensräume in Typdefinitionen . . . . .	267
Geben Sie Modulen den Vorzug vor Namensräumen . . . . .	268
Type-Only-Importe und -Exporte . . . . .	269
Zusammenfassung . . . . .	270



<b>15 Typoperationen</b> .....	<b>271</b>
Abgebildete Typen .....	271
Abgebildete Typen auf Basis von Objekttypen .....	272
Änderung von Modifiern .....	274
Generische abgebildete Typen .....	275
Bedingte Typen .....	277
Generische bedingte Typen .....	277
Distributive Typen .....	279
Abgeleitete Typen .....	279
Abgebildete bedingte Typen .....	280
never .....	281
never und Schnittmengen- und Vereinigungstypen .....	281
never und bedingte Typen .....	281
never und abgebildete Typen .....	282
Template-Literaltypen .....	283
Intrinsische String-Manipulationstypen .....	284
Template-Literalschlüssel .....	285
Remapping von Schlüsseln .....	286
Typoperationen und Komplexität .....	288
Zusammenfassung .....	288
<b>Glossar</b> .....	<b>289</b>
<b>Index</b> .....	<b>299</b>