

# Einführung in TypeScript

## Vorteile von TypeScript

TypeScript erweitert JavaScript um statische Typisierung und bringt Vorteile wie Fehlererkennung, Code-Completion und bessere Wartbarkeit.

- **Statische Typisierung:** TypeScript ermöglicht die Definition von Typen für Variablen, Funktionen und Objekte.
- **Bessere Entwicklererfahrung:** TypeScript bietet IntelliSense und Code-Completion.
- **Klare Typisierung & Wartbarkeit:** TypeScript macht den Code verständlicher und sicherer.

## Hello World

Node.js und tsc

1. Installiere TypeScript: `npm install -g typescript`
2. Erstelle eine `hello.ts` Datei:

```
const message: string = "Hello, TypeScript!";
console.log(message);
```

3. Kompiliere die Datei: `tsc hello.ts`
4. Führe die JavaScript-Datei aus: `node hello.js`
5. Konfiguriere TypeScript mit `tsconfig.json`: `tsc --init`

## Deno

Deno unterstützt TypeScript direkt, ohne zusätzliche Konfiguration.

1. Erstelle eine `hello.ts` Datei:

```
const message: string = "Hello, Deno!";
console.log(message);
```

2. Führe die Datei aus: `deno run hello.ts`

## Beispielsetup für ein Webprojekt (React mit Vite)

Vite ist ein schnellerer und effizienterer Build-Tool-Ersatz für Create React App.

1. Erstelle ein neues React-Projekt mit Vite:

```
npm create vite@latest my-react-app --template react-ts
cd my-react-app
npm install
```

2. Starte das Projekt im Entwicklungsmodus:

```
npm run dev
```