

ONE PAGE SUMMARY

IT & Software > Frontend Development > TypeScript

# Introduction to TypeScript

TypeScript: Elevate Your JavaScript with Strong Typing!

DEFINITION

TypeScript is a superset of JavaScript that adds static types, enhancing code quality and maintainability. It helps developers catch errors early through type checking.

KEY CONCEPTS

Static Typing

Static typing allows you to define variable types, reducing runtime errors.

Interfaces

Interfaces define the structure of objects, ensuring consistent data shapes across your application.

Generics

Generics enable writing flexible, reusable functions and classes that work with various data types.

Type Inference

TypeScript can automatically infer types based on the assigned values, simplifying code.

Decorators

Decorators are special annotations that can modify classes and methods, enhancing functionality.

EXAMPLES

→ let user: { name: string; age: number } = { name: 'Alice', age: 30 };  
→ function identity<T>(arg: T): T { return arg; }  
→ interface Car { make: string; model: string; year: number; }

MEMORY TIPS

★ Think of TypeScript as JavaScript with a safety net for catching errors.  
★ Remember 'I.G.S.D' for Interfaces, Generics, Static Typing, and Decorators.  
★ Visualize types as labels on boxes, ensuring you know what's inside before opening.

COMMON MISTAKES

✗ Forgetting to define types, leading to implicit 'any' types.  
✗ Misusing interfaces and types, causing type mismatches.  
✗ Neglecting to enable strict mode, missing out on type safety benefits.

QUICK RECAP

TypeScript enhances JavaScript by adding static types, improving code reliability. Key features include interfaces, generics, and type inference. Remember to define types to avoid common pitfalls.