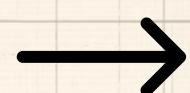




TypeScript vs JavaScript: What Every Web Developer Should Know Swipe to learn the key differences and when to choose each.

Chandrika Ajmera





What is JavaScript?

- JavaScript is a lightweight, dynamic scripting language for web interactivity.
- Runs natively in browsers and on servers (Node.js).
- Dynamic typing means variables can change type at runtime.
- Supported by vast frameworks like React, Angular, and Vue.



What is TypeScript?

- Superset of JavaScript adding static typing and advanced features.
- Developed by Microsoft to enhance error detection before runtime.
- Requires compilation to JavaScript to run in browsers.
- Ideal for large-scale projects and collaborative teams.



Key Differences

Feature	TypeScript	JavaScript
Typing	Static (explicit types)	Dynamic (runtime types)
Error Checking	Compile-time catch	Runtime errors
Tooling	Rich IDE support & compile checks	Limited tooling
Syntax	JS + types/interfaces /enums	Standard JavaScript
Learning Curve	Steeper due to types/interfaces	Easier, quick start
Compatibility	Compiles to JS, runs anywhere	Runs natively in browsers/Node





When to Use TypeScript?

- Large apps needing maintainability and scalability
- Teams requiring consistent, error-reduced code
- Projects benefiting from tooling and IDE productivity boosts
- When using modern OOP features: interfaces, enums, generics



When JavaScript Stands Out?

- Quick prototypes or small projects
- Dynamic, flexible scripting needs
- Avoiding compilation step
- Leveraging huge ecosystem and community knowledge



Final Thoughts

- **TypeScript enhances JavaScript – it's not a replacement.**
- **Mastering both expands your versatility as a web developer.**
- **Both have vital roles depending on project scale and needs.**