🎉 Introduction to TypeScript

A friendly guide for JavaScript developers

🧐 What is TypeScript?

- A **superset of JavaScript**
- Adds **static types** ✅
- Developed by **Microsoft**
- Compiles to **plain JavaScript**
- Catches **errors early**

💡 Why Use TypeScript?

- **Type safety**: Find bugs before runtime
- **Better IDE support**: Autocomplete & refactoring
- **Readable & maintainable code**
- Popular in **large projects**

🧰 Basic Types

```typescript
let isDone: boolean = false;
let age: number = 25;
let name: string = "Alice";
let numbers: number[] = [1, 2, 3];
```

- `boolean` → true/false
- `number` → integers & floats
- `string` → text
- `array` → list of values

⚙ Functions with Types

```
function greet(name: string): string {
  return `Hello, ${name}!`;
}
console.log(greet("Bob"));
```

- Parameters have **types**
- Functions can have **return types**

📦 Interfaces

```
interface Person {
  name: string;
  age: number;
}

let user: Person = { name: "Alice", age: 30 };
```

- Define **object shapes**
- Catch mistakes like missing properties

✨ Optional & Default Parameters

```
function greet(name: string, age?: number) {
  console.log(`Hello, ${name}, age ${age ?? "unknown"}`);
}
```

- `?` → optional parameter
- `=` → default value

🔍 Type Inference

```
let message = "Hello TypeScript"; // inferred as string
```

- TypeScript can **guess types**
- Explicit typing is optional

⚔️ TypeScript vs JavaScript

Feature	JavaScript	TypeScript
Static Types	❌	✅
Compile-time checks	❌	✅
Object-oriented features	❌	✅
IDE Autocomplete	Limited	Excellent

🏁 Getting Started

1. Install Node.js (includes npm)
2. Install TypeScript:

```
npm install -g typescript
```

3. Compile a file:

```
tsc app.ts
```

4. Run JavaScript:

```
node app.js
```

📚 Resources

- TypeScript Docs
- TypeScript Handbook
- TypeScript Playground

🎉 Summary

- TypeScript = JavaScript + Types
- Catch errors early ✅
- Interfaces & type annotations improve readability
- Compile TypeScript → JavaScript