

- How do you define an array of strings in TypeScript using two different syntaxes?
- • What is the purpose of the `tuple` type, and how does it differ from a standard array?
- • Explain the difference between `any` and `unknown`. When should you prefer one over the other?
- • How do you create an optional property in an `interface`?
- • What does the `readonly` modifier do when applied to a class property or interface member?
- • What is "Type Aliasing," and how do you define a union type using it?
- • Explain the difference between an `interface` and a `type` alias.
- • What is a "Type Guard" in TypeScript, and can you provide an example using the `typeof` operator?
- • How does the `never` type differ from the `void` type?
- • What is "Structural Typing" (also known as duck typing), and how does TypeScript implement it?
- • How do you use the `keyof` operator, and what kind of value does it produce?
- • What are "Generics," and why are they useful for creating reusable components?
- • How do you define a function type using an arrow function syntax in an `interface`?
- • What is the "Non-null assertion operator" (`!`), and what are the risks of using it?
- • How do you use "Intersection Types" to combine multiple interfaces into one?
-