100 XP

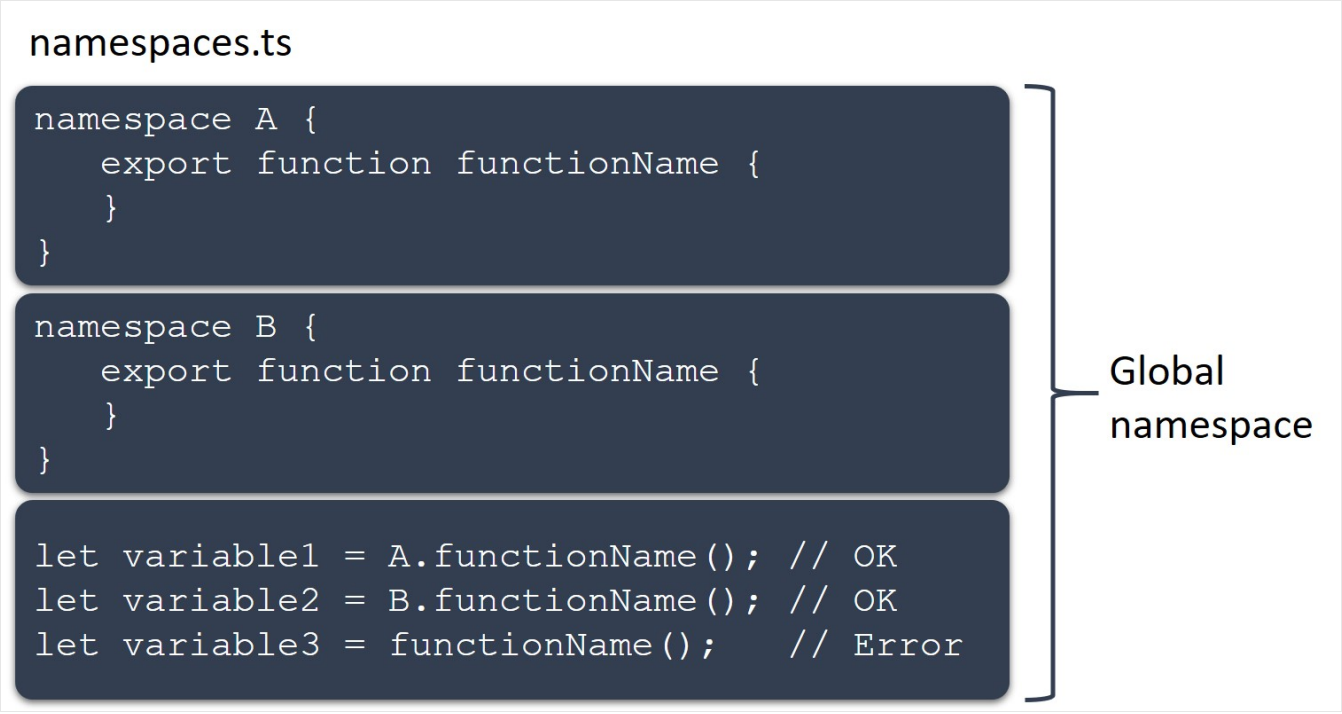
Introduction to TypeScript namespaces

5 minutes

Namespaces (referred to as "internal modules" in earlier versions of TypeScript) are a TypeScript-specific way to organize and categorize your code, enabling you to group related code together. Namespaces allow you to group variables, functions, interfaces, or classes related to business rules in one namespace and security in another.

Code inside a namespace is pulled from the global scope and into the scope of the namespace. This placement can help you avoid naming conflicts between components in the global namespace and can be beneficial when working with distributed development teams that may use similar component names.

For example, `namespace A` and `namespace B` both share a function called `functionName`. Any attempt to access the function without referencing the containing namespace results in an error because the variable declarations are in the global namespace, while the two functions are contained within the scope of their respective namespaces.



You can use namespaces to:

- Reduce the amount of code in the global scope, limiting "global scope pollution."
- Provide a context for names, which helps reduce naming collisions.
- Enhance reusability.

Next unit: Exercise - Organize code by using single file namespaces

Continue >

How are we doing? ☆ ☆ ☆ ☆ ☆