TypeScript Function Overloading

Function overloading is a mechanism or ability to create multiple methods with the **same name** but different parameter types and **return type**. However, it can have the same number of parameters. Function overloading is also known as method overloading.

The Function/Method overloading is allowed when:

* The function name is the same
* The number of parameters is different in each overloaded function.
* The number of parameters is the same, and their type is different.
* The all overloads function must have the same return type.

Suppose we have to perform **multiplication** of the numbers, which has a different number of parameters. We write the **two** methods such as mul\_a(number, number) for **two parameters**, and mul\_b(number, number, number) for **three parameters**. Now, it may be difficult for us as well as other programmers to understand the behavior of the method because its name **differs**. In that case, we need to use function overloading, which increases the readability of the program.

Advantage of function overloading

* It saves the memory space so that program execution becomes fast.
* It provides code reusability, which saves time and efforts.
* It increases the readability of the program.
* Code maintenance is easy.
* /\*1.Using overloading technique, write methods to:
* • accept two int type data as parameters and return their sum. \*/
* //Function with string type parameter
* function add(a:string, b:string): string;
* //Function with number type parameter
* function add(a:number, b:number): number;
* //Function Definition
* function add(a: any, b:any): any {
* return a + b;
* }
* //Result
* console.log("Addition: " +add("Hello ", "JavaTpoint"));
* console.log("Addition: "+add(30, 20));
* /\*Function overloading in a class\*/
* class A
* {
* public foo(s: string): number;
* public foo(n: number): string;
* public foo(arg: any): any
* {
* if (typeof(arg) === 'number')
* return arg.toString();
* if (typeof(arg) === 'string')
* return arg.length;
* }
* }
* let obj = new A();
* console.log("Result: " +obj.foo(101));
* console.log("Length of String: " +obj.foo("JavaTpoint"));
* /\*2:-accept three int type data as parameters and return their sum.\*/

