

TypeScript Union Types

The TypeScript union has the ability to combine one or two different types of data (i.e., number, string, float, double, etc). It is the most powerful way to express a variable with multiple types. Use pipe ('|') symbol to combine two or more data types to achieve Union type.

Syntax:

(type1|type2|type3|...|type-n)

Union types are used when a value can be more than a single type. Such as when a property would be **string** or **number**.

Union | (OR)

Using the | we are saying our parameter is a **string** or **number**:

Example

```
function printStatusCode(code: string | number) {  
  console.log(`My status code is ${code}.`)  
}  
printStatusCode(404);  
printStatusCode('404');
```

Union Type Errors

Note: you need to know what your type is when union types are being used to avoid type errors:

Example

```
function printStatusCode(code: string | number) {  
  console.log(`My status code is ${code.toUpperCase()}.`) // error:  
  Property 'toUpperCase' does not exist on type 'string | number'.  
  Property 'toUpperCase' does not exist on type 'number'  
}
```

In our example we are having an issue invoking **toUpperCase()** as its a **string** method and **number** doesn't have access to it.