

Type alias

Vimos um requinte da linguagem TypeScript é que é o uso de Union Types e Type Guards.

Vejamos um exemplo:

```
function processaToken(token: string | number) {  
  
    if(typeof(token) === 'string') {  
  
        return token.replace(/2/g, 'X');  
    } else {  
        return token.toFixed().replace(/2/g, 'X');  
    }  
  
}
```

Contudo, é possível criar um alias para todos os tipos envolvidos no union type. Vejamos:

```
// criando o alias!  
type MeuToken = string | number;  
  
function processaToken(token: MeuToken) {  
  
    if(typeof(token) === 'string') {  
  
        return token.replace(/2/g, 'X');  
    } else {  
  
        return token.toFixed().replace(/2/g, 'X');  
    }  
  
}
```

O código continua compilando, mas em vez de indicarmos no parâmetro da função os tipos `string` e `number`, passamos o alias que criamos, no caso `MeuToken`. O Type alias é interessante quando trabalhamos com muitos union types e queremos padronizar em um único local esses conjunto de tipos.

É um recurso que não utilizamos em nossa aplicação, mas que o aluno precisa conhecer para poder analisar código de terceiros com maior segurança.