

02

## O tipo implícito any

### Transcrição

Nossa instância de `Negociacao` está recebendo como valores das propriedades `_data`, `_quantidade` e `_valor` valor em string. Lembre-se que isso acontece porque os elementos do DOM que guardam os valores digitados pelo usuário sempre os guardam como string.

Como o nome da linguagem já diz, TypeScript é baseado em tipos. Mas se precisamos tipar nossas propriedades, como nosso código funcionou sem definirmos um tipo. Até porque, linguagens como Java ou C# não deixariam passar declarações sem a informação do tipo.

Isso acontece porque o padrão do TypeScript é adotar o tipo `any` implícito. O tipo `any` indica que a variável, propriedade, parâmetro de método ou seu retorno podem ser de qualquer tipo. Esse comportamento ajuda na migração de um código legado em JavaScript para esta linguagem. No entanto, como estamos começando um projeto do zero, podemos impedir que o TypeScript assuma implicitamente o tipo `any`, forçando-nos a explicitar a tipagem.

Vamos alterar `alurabak/tsconfig.json` para adicionarmos no em seguida a configuração `"noImplicitAny": true`:

```
{  
  "compilerOptions": {  
    "target": "es6",  
    "outDir": "app/js",  
    "noEmitOnError": true,  
    "noImplicitAny": true  
  },  
  "include": [  
    "app/ts/**/*"  
  ]  
}
```

Feche e abra novamente o VCode para que ele entenda a nova configuração. Agora, nosso arquivo `Negociacao.ts` e `NegociacaoController.ts` apresentarão erros de compilação, pois não tipamos as propriedades da classe. Vamos começar por `Negociacao.ts`.