



escola  
britânica de  
artes criativas  
& tecnologia

# Profissão: Engenheiro Front-End



# Conheça o TypeScript



## GUIA DA AULA 2



# Tipos básicos

● Sobre os tipos

● Tupla

● Union Type

● Any



Acompanhe aqui  
os temas que  
serão tratados  
na videoaula



## Sobre o Typescript

No TypeScript possuímos alguns tipos básicos, primitivos, são eles: boolean, number, string, array, readonlyArray, tuplas, union e any.

Os arrays podem ser declarados através dos colchetes ou das palavras reservadas Array e readonlyArray:

```
const nomes: string[] = []
```

```
const nomes2: Array<string> = []
```

```
const nomes3: readonlyArray<string> = []
```



A única diferença que temos é: ao utilizar o `readonlyArray`, onde os métodos para manipulação do array serão removidos, por exemplo, não poderemos adicionar um novo item com *push*.

No JavaScript podemos ter num mesmo vários tipos de dados. Isso não é possível no TypeScript, onde temos que definir do que o array será composto, porém temos um tipo de dado que permite um comportamento parecido, a **tupla**.



## Tupla

Para criar uma tupla, utilizamos os colchetes informando os tipos que serão aceitos:

```
const lista: [string, boolean, number] = ["ana", true, 22];
```

Assim temos uma estrutura similar ao de um array> Também podemos informar um nome para cada índice, como se fossem campos:

```
const lista: [nome: string, estaEstudando: boolean, idade: number] = ["ana", true, 22];
```



## Union Type

Apesar da forte tipagem que o TypeScript nos fornece, podemos aceitar mais um tipo em uma variável, utilizando o Union Type:

```
let idade: number | string;
```

```
idade = 20
```

```
idade = "20 anos"
```



## Any

O TypeScript fornece um tipo fraco, chamado *any* (do inglês: qualquer), onde podemos atribuir qualquer tipo de dado. Mas este recurso deve ser usado em último caso:

```

let minhaVariavel: any;
minhaVariavel = [1, 2, 3, 4]
minhaVariavel = "gian"
minhaVariavel = {}
minhaVariavel = true
  
```

