

inferencia literal typescript, oque é e como usar?



- > Inferência literal em TypeScript é um recurso que permite ao compilador deduzir tipos específicos com base em valores literais específicos que aparecem no código. Isso significa que o TypeScript pode inferir o tipo exato de uma variável ou propriedade com base em seu valor literal.

Por exemplo, se você definir uma variável com um valor literal de string, o TypeScript irá inferir automaticamente que o tipo da variável é uma string. Isso pode ser útil para garantir que o tipo de uma variável seja consistente com seu valor literal, sem precisar declarar explicitamente o tipo.

Aqui está um exemplo:

ruby

Copy code

```
const animal = "cachorro"; // o tipo de 'animal' é inferido como 'string'
const idade = 3; // o tipo de 'idade' é inferido como 'number'
const ehMacho = true; // o tipo de 'ehMacho' é inferido como 'boolean'
```

Note que em todos os casos acima, o tipo da variável é inferido automaticamente com base no valor literal.

A inferência literal também pode ser usada em conjunto com tipos específicos, como enumerações, para criar tipos mais precisos. Por exemplo:

perl

Copy code

```
enum Tamanho { Pequeno = "P", Médio = "M", Grande = "G" }
const tamanho: Tamanho = Tamanho.Pequeno; // 'tamanho' é inferido como 'Tamanho.Pequeno', do tipo 'Tamanho'
```

Nesse exemplo, o tipo da variável `tamanho` é inferido como `Tamanho.Pequeno`, que é uma das opções da enumeração `Tamanho`. Isso ajuda a garantir que o tipo da variável seja consistente com a enumeração.