

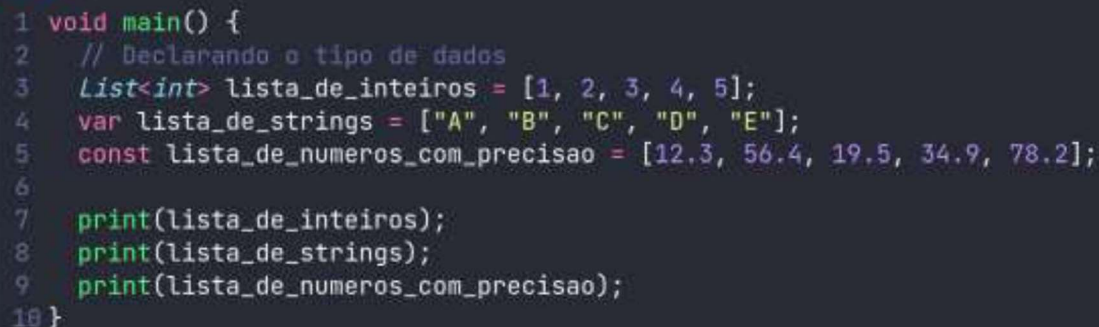
Declaração de listas

A declaração de uma lista é feita através de atribuição de um objeto List a uma variável ou constante.

Ao atribuírmos uma lista a uma variável ou constante, podemos fazê-lo de duas formas:

- Declarando o tipo de dados presente na lista.
- Declarando a lista sem um tipo de dados e inferindo o tipo de acordo com os valores nela inseridos.

Talvez você ainda não tenha compreendido muito bem o que quero dizer com isso. Mas tudo bem, passemos a um exemplo prático.



```
1 void main() {
2     // Declarando o tipo de dados
3     List<int> lista_de_inteiros = [1, 2, 3, 4, 5];
4     var lista_de_strings = ["A", "B", "C", "D", "E"];
5     const lista_de_numeros_com_precisao = [12.3, 56.4, 19.5, 34.9, 78.2];
6
7     print(lista_de_inteiros);
8     print(lista_de_strings);
9     print(lista_de_numeros_com_precisao);
10 }
```

Propriedades

Como bem sabemos, todos os tipos de dados do Dart são objetos. Dessa forma possuem propriedades, e com a lista não seria diferente.

Neste tópico falaremos das principais propriedades do objeto List.

Propriedade isEmpty

A propriedade isEmpty, como o próprio nome já sugere, verifica se a lista está vazia, ou seja, se ela não possui nenhum elemento. Caso a lista esteja vazia, retornará true, caso contrário retornará false.

```
1 void main() {  
2     List<String> vingadores = [  
3         "Homem de ferro",  
4         "Capitão América",  
5         "Thor",  
6         "Hulk"  
7     ];  
8  
9     // Verificamos se a lista está vazia  
10    print(vingadores.isEmpty);  
11}
```

Propriedade isEmpty

A propriedade isEmpty, como o próprio nome já sugere, verifica se a lista não está vazia, ou seja, se ela possui nenhum elemento. Caso a lista não esteja vazia, retornará true, caso contrário retornará false.

```
1 void main() {  
2     List<String> vingadores = [  
3         "Homem de ferro",  
4         "Capitão América",  
5         "Thor",  
6         "Hulk"  
7     ];  
8  
9     // Verificamos se a lista NÃO está vazia  
10    print(vingadores.isNotEmpty);  
11}
```

Propriedade Length

A propriedade length, como o próprio nome já sugere, retorna o número de elementos armazenados na lista.

Esse valor é muito utilizado para verificações relacionadas a impressão de elementos na tela, sendo utilizada em conjuntos com estruturas de repetição, como o while e o for.

```
1 void main() {  
2     List<String> vingadores = [  
3         "Homem de ferro",  
4         "Capitão América",  
5         "Thor",  
6         "Hulk"  
7     ];  
8  
9     // Imprimiremos o número de elementos presentes na lista  
10    print("A lista de vingadores possui ${vingadores.length} vingadores!");  
11}  
12
```