

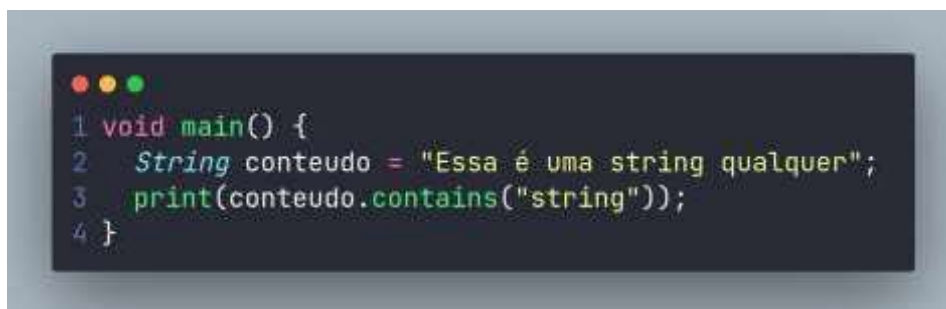
Métodos

A classe String possui alguns métodos, os quais podemos utilizar para manipular o valor do objeto. Vejamos quais são eles e como podemos utilizá-los.

Método contains

O método Contains, como o próprio nome já sugere, tem como objetivo descobrir se uma string possui ou não uma outra string como parte de seu conteúdo. Caso encontre a string ele retornará true, caso contrário retornará false.

Parâmetro	Descrição
string	String a qual deseja localizar


Um terminal de código com fundo escuro e texto colorido. No topo, há três pontos de status (vermelho, amarelo, verde). O código é o seguinte:

```
1 void main() {  
2     String conteudo = "Essa é uma string qualquer";  
3     print(conteudo.contains("string"));  
4 }
```

Método startsWith

O método startsWith, como o próprio nome já sugere, tem como objetivo descobrir se uma string começa ou não uma outra string. Caso encontre a string ele retornará true, caso contrário retornará false.

Parâmetro	Descrição
string	String a qual deseja localizar no início da string alvo

Um terminal de código com fundo escuro e texto colorido. No topo, há três pontos de status (vermelho, amarelo, verde). O código é o seguinte:

```
1 void main() {  
2     String conteudo = "Google é o motor de buscas mais famoso do mundo";  
3     print(conteudo.startsWith("Google"));  
4 }
```

Método endsWith

O método `endsWith`, é muito parecido com o método `startsWith`, se diferenciando pelo fato de descobrir se uma string termina ou não uma outra string. Caso encontre a string ele retornará `true`, caso contrário retornará `false`.

Parâmetro	Descrição
string	String a qual deseja localizar no final da string alvo

```
1 void main() {  
2     String conteudo = "Google é o motor de buscas mais famoso do mundo";  
3     print(conteudo.endsWith("Google"));  
4 }
```

Método substring

O método `substring` retorna parte do conteúdo de acordo com as configurações de ponto de início até o ponto de término.

Parâmetro	Descrição
início	Caractere inicial a ser lido
término	Caractere final a ser lido

```
1 void main() {  
2     String conteudo = "Google é o motor de buscas mais famoso do mundo";  
3     String prefixo = "O Bing é um";  
4     // Retornará → motor de buscas  
5     String sufixo = conteudo.substring(11,26);  
6     print("$prefixo $sufixo");  
7 }
```

Método toUpperCase

O método `toUpperCase` transforma o conteúdo (números, letras e caracteres especiais) de uma string para maiúsculo.

```

1 void main() {
2   String nome_do_usuario = "pedro Henrique";
3   print(nome_do_usuario.toUpperCase());
4 }
5

```

Método toLowerCase

O método toLowerCase transforma o conteúdo (números, letras e caracteres especiais) de uma string para minúsculo.

```

1 void main() {
2   String nome_do_usuario = "pedro Henrique";
3   print(nome_do_usuario.toLowerCase());
4 }
5

```

Método trim

O método trim retorna uma string com os espaços retirados do início e do final de string.

```

1 void main() {
2   String nome_do_usuario = "  Pedro Henrique  ";
3   print(nome_do_usuario.trim());
4 }
5

```

É importante ressaltar que o Dart possui ainda as funções trimRight e trimLeft, que removem, respectivamente, apenas os espaços em branco da direita e esquerda de uma string.