A **string** in TypeScript is a sequence of characters used to represent text. It can be declared using:

1. **Single quotes** 'like this'

2. **Double quotes** "like this"

3. **Backticks (Template literals)** `like this` – useful when embedding variables inside strings.

```
let str1: string = 'Single quote';

let str2: string = "Double quote";

let str3: string = `Backtick`;
```

## Why use Backticks?

Backticks allow **template literals**, which support **string interpolation** using ${}:

```
let num: number = 10;

console.log(`Number is: ${num}`); // ✅ Correct way
```

❌ 'Number is: ${num}' or "Number is: ${num}" won't work as expected.

## Common String Methods

```
let greeting: string = "Hello, TypeScript!";
```

1. **.length** – Get the number of characters
   greeting.length → 18

2. **.toUpperCase() / .toLowerCase()** – Change case
   greeting.toUpperCase() → HELLO, TYPESCRIPT!
   greeting.toLowerCase() → hello, typescript!

3. **.charAt(index) / .indexOf(substring)** – Get a character or find a substring
   greeting.charAt(4) → o
   greeting.indexOf("Type") → 7

4. **.substring(start, end)** – Extract a part of the string
   greeting.substring(7, 11) → Type

5. **.includes(substring)** – Check if a substring exists
   greeting.includes("Script") → true

6.  **.startsWith() / .endsWith()** – Check start or end of string
    greeting.startsWith("Hello") → true
    greeting.endsWith("!") → true

7.  **.replace(old, new)** – Replace part of a string
    greeting.replace("TypeScript", "World") → Hello, World!

8.  **.split(delimiter)** – Split into array
    greeting.split(" ") → [ 'Hello,', 'TypeScript!' ]

9.  **.trim() / .trimStart() / .trimEnd()** – Remove extra spaces
    Removes spaces from start, end, or both.

## String Immutability

Strings are **immutable** – once created, they can't be changed. Methods return **new strings**:

```
let original = "Hello";

let modified = original.concat(", World!");

// original is still "Hello"
```

## Multi-line Strings

Using backticks allows strings to span multiple lines:

```
let multiLine = `Line one

Line two

Line three`;
```