

Day 52/60 - Understanding the Difference Between String Interpolation and String.Format in C#

Formatting strings efficiently is crucial in C#. Two common approaches are String Interpolation (\$) and `String.Format()`. Let's compare them.

String Interpolation (\$) - Modern and Readable

Introduced in C# 6, allowing direct variable embedding into strings.
Improves readability and eliminates concatenation errors.

Example:

```
string name = "Abhinn";  
int age = 25;  
string message = $"My name is {name} and I am {age} years old.";  
Console.WriteLine(message);
```

Best for: Readable, modern string formatting.

String.Format() - Legacy Formatting

Used before C# 6 for formatted strings.
Uses index-based placeholders instead of direct variable embedding.

Example:

```
string message = String.Format("My name is {0} and I am {1} years old.", "Abhinn", 25);  
Console.WriteLine(message);
```

Downside: Less readable and harder to maintain than interpolation.

When to Use What?

Use String Interpolation (\$) for clean, modern string formatting.
Use `String.Format()` only when working with older C# versions (before C# 6).

String interpolation makes C# cleaner and more readable, so use it whenever possible!

`#dotnet` `#csharp` `#stringinterpolation` `#stringformat` `#bestpractices`