

HABIB UL REHMAN

01

C#

STRING INTERPOLATION

BEST PRACTICES



@habibdeveloper

FOLLOW



FINCHSHIP.COM

Direct Interpolation

Leverage the straightforward and concise syntax of direct interpolation for simple expressions within strings.

```
string name = "John";  
int age = 30;  
  
string info = $"Name: {name}, Age: {age}";  
  
//Output: Name: John, Age: 30
```



@habibdeveloper

FOLLOW

FINCHSHIP.COM

Format Strings

Enhance readability by combining string interpolation with format strings for more complex expressions.



```
decimal price = 99.99;  
  
string formattedPrice = $"Total Price: {price:C2}";  
  
//Output: Total Price: $99.99
```



@habibdeveloper

FOLLOW



FINCHSHIP.COM

Multiline Interpolation

Facilitate readability for multiline strings by utilizing the \$ prefix. This will preserve the formatting exactly as written in the source code.

```
string multiline = @"This is a
                    multiline
                    string.";

//Output: This is a
//          multiline
//          string.
```



@habibdeveloper

FOLLOW



FINCHSHIP.COM

Conditional Interpolation

Incorporate conditional statements directly into interpolations for dynamic content. The good approach is assign the result to a variable and then interpolate inside the string.

```
bool isPremium = true;

string message = $"User is {(isPremium ? "Premium" :
"Standard")} member.";

//Output: User is Premium member.
```



@habibdeveloper

FOLLOW



FINCHSHIP.COM

Function Calls

Invoke methods or functions inline within interpolations for dynamic text generation.

```
DateTime currentDate = DateTime.Now;  
  
string formattedDate = $"Current Date:  
{currentDate.ToShortDateString()}";  
  
//Output: Current Date: 7/1/2024
```



@habibdeveloper

FOLLOW



FINCHSHIP.COM

Verbatim Interpolation

For scenarios where escape characters need to be preserved, use verbatim interpolation.

```
string path = @"C:\MyDocuments";  
  
string fullPath = $"The full path is: {path}";  
  
//Output: The full path is: C:\MyDocuments
```



@habibdeveloper

FOLLOW



FINCHSHIP.COM

Culture-Specific Formatting

Ensure consistency across cultures by incorporating culture-specific formatting.

```
double numericValue = 1234.5678;  
  
string formattedValue = $"{numericValue:N2}";  
  
//Output: 1,234.57
```



@habibdeveloper

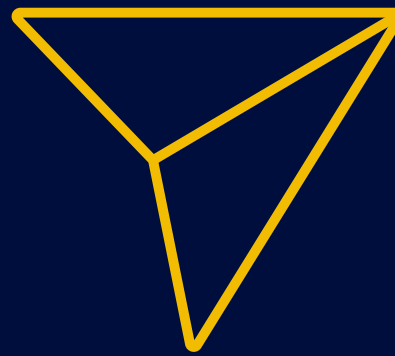
FOLLOW

FINCHSHIP.COM

HABIB UL REHMAN

09

USEFUL?



REPOST AND SHARE WITH FRIENDS TO ACCELERATE THEIR SKILLS

Follow for more updates



@habibdeveloper

FOLLOW



FINCHSHIP.COM

