



## 🌟 Developer Helpers – 19 VBA String Functions That Every Developer Will Love!

If you work with VBA, you know the drill: strings are everywhere — during import, reporting, data preparation and text processing. And every time it's: substring here, trim there, replace over there... 😊

### That ends now!

With my new Developer Helpers you get a toolbox of 19 string functions that support you in every text task — **fast, elegant and error-free.**

### 🌿 What makes them special?

Every function is directly testable via an integrated form interface.

You see the result immediately, can check your own inputs and combine functions as you like.

That gives you maximum flexibility — and you work more productively, cleaner and with less debugging.

### 🛒 All 19 functions at a glance

#### ✅ StringFixLength

Sets a fixed length for a text.

If the string is too long it will be truncated — if it's too short it will be padded.

`StringFixLength("ABC", 6, "_")` → `"ABC___"`

Ideal for fixed-width fields in reports or exports.

## ✓ StringCombine

Joins two or more strings with a freely selectable separator.

`StringCombine("FirstName", "LastName", " ")` → `"FirstName LastName"`

Perfect for names, paths or CSV formats.

---

## ✓ StringRemoveTrailingChars

Removes specific characters at the end of a text (e.g. trailing commas or spaces).

`StringRemoveTrailingChars("Test, ", ", ")` → `"Test"`

---

## ✓ StringRemoveChars

Deletes specific characters throughout the entire string.

`StringRemoveChars("12-34-56", "-")` → `"123456"`

---

## ✓ StringRemoveWords

Removes specified words from a text.

`StringRemoveWords("This is a test text", "test text")` → `"This is a"`

---

## ✓ StringExtractBetween

Extracts text between two markers — perfect for parsers or log analysis.

`StringExtractBetween("Hello [World]!", "[", "]")` → `"World"`

---

## ✓ StringCount

Counts letters, numbers, special characters or words in the text.

`StringCount("ABC 123", "letters")` → `3`

---

## ✓ StringCapitalizeWords

Capitalizes the first letter of each word.

`StringCapitalizeWords("max mustermann")` → `"Max Mustermann"`

---

## ✓ StringCountSubstring

Counts how often a given substring occurs.

`StringCountSubstring("banana", "na")` → `2`

---

## ✓ StringIsNumeric

Checks whether a string represents a valid number (including minus or decimal point).

`StringIsNumeric("-12.5")` → `True`

---

## ✓ StringContainsAny

Searches for any of a list of terms within a string.

`StringContainsAny("This is great", Array("great", "awesome"))` → `True`

---

## ✓ StringIsEmpty

Detects empty, null or whitespace-only strings.

`StringIsEmpty(" ")` → `True`

---

## ✓ StringExtractEmails

Finds all email addresses inside a text.

`StringExtractEmails("Contact: test@mail.com or info@company.de")`  
→ `Array("test@mail.com", "info@company.de")`

---

## ✓ StringClean

Cleans up excessive spaces and control characters.

`StringClean(" Hello World ")` → `"Hello World"`

---

## ✓ StringRepeat

Repeats a string a specified number of times.

`StringRepeat("-", 10)` → `"-----"`

---

## ✓ StringPadNumber

Pads numbers with leading characters (e.g. zeros).

`StringPadNumber(42, 5, "0")` → `"00042"`

---

## ✓ StringWrap

Wraps long text to a specified line length.

`StringWrap("This is a very long text", 10)`

---

## ✓ StringToInitials

Generates initials from names or phrases.

`StringToInitials("Max Mustermann")` → `"MM"`

---

## ✓ StringMaskSensitive

Masks sensitive parts of text — shows only selected segments.

`StringMaskSensitive("CreditCard1234", 4)` → `"*****1234"`

---

## ⚙ Deployment & Integration

All functions are commented, validate inputs and can be inserted directly into existing projects. With the

test form you can try every function immediately — no setup, no add-in required. You only need the module **modHelper\_String**. Everything else is just for testing and presentation.


---

### **Conclusion: More power for your VBA workflow**

With these 19 Developer Helpers your string handling becomes faster, cleaner and much more efficient.

- ◆ Less code
- ◆ Fewer errors
- ◆ More productivity

### **Try it out**

Using my small testing form you can try the functions directly. This makes it easy to learn the functions step by step. Do you develop a lot with Access? Then give it a try — I'm looking forward to your feedback or ideas for additional functions! 



## 🌟 Developer Helpers – 19 VBA String-Funktionen, die jeden Entwickler begeistern!

Wer mit VBA arbeitet, kennt es: Zeichenketten sind überall – beim Import, beim Reporting, bei Datenaufbereitung und Textverarbeitung.

Und jedes Mal heißt es: Substring hier, Trim da, Replace dort... 🤖

### Damit ist jetzt Schluss!

Mit meinen neuen Developer Helpers bekommst du einen Werkzeugkasten mit 19 String-Funktionen, die dich bei jeder Arbeit mit Texten unterstützen – schnell, elegant und fehlerfrei.

### 🌟 Was ist besonders?

Jede Funktion ist direkt testbar über eine integrierte Formularoberfläche.

Du siehst sofort das Ergebnis, kannst eigene Eingaben prüfen und jede Funktion beliebig kombinieren.

So entsteht maximale Flexibilität – und du arbeitest produktiver, sauberer und mit weniger Debugging.

### 📁 Alle 19 Funktionen im Überblick

#### ✅ StringFixLength

Setzt eine feste Länge für einen Text.

Ist der String zu lang, wird er abgeschnitten – ist er zu kurz, wird aufgefüllt.

`StringFixLength("ABC", 6, "_")` → `"ABC_ _"`

Ideal für fixe Feldlängen in Reports oder Exporten.

#### ✅ StringCombine



Verbindet zwei oder mehr Strings mit einem frei wählbaren Trennzeichen.

`StringCombine("Vorname", "Nachname", " ")` → `"Vorname Nachname"`

Perfekt für Namen, Pfade oder CSV-Formate.

---

## ✓ StringRemoveTrailingChars

Entfernt gezielt Zeichen am Ende eines Textes (z. B. Kommas oder Leerzeichen).

`StringRemoveTrailingChars("Test, ", ", ")` → `"Test"`

---

## ✓ StringRemoveChars

Löscht bestimmte Zeichen im gesamten String.

`StringRemoveChars("12-34-56", "-")` → `"123456"`

---

## ✓ StringRemoveWords

Entfernt gezielt Wörter aus einem Text.

`StringRemoveWords("Das ist ein Testtext", "Testtext")` → `"Das ist ein"`

---

## ✓ StringExtractBetween

Extrahiert Text zwischen zwei Markern – perfekt für Parser oder Logauswertungen.

`StringExtractBetween("Hallo [Welt]!", "[", "]")` → `"Welt"`

---

## ✓ StringCount

Zählt Buchstaben, Zahlen, Sonderzeichen oder Wörter im Text.

`StringCount("ABC 123", "letters")` → `3`

---

## ✓ StringCapitalizeWords

Wandelt jeden Wortanfang in Großbuchstaben um.

`StringCapitalizeWords("max mustermann")` → `"Max Mustermann"`

---

## ✓ StringCountSubstring

Zählt, wie oft ein bestimmter Teilstring vorkommt.

`StringCountSubstring("banana", "na")` → `2`

---

## ✓ StringIsNumeric

Prüft, ob ein String eine gültige Zahl darstellt (inkl. Minus oder Komma).

`StringIsNumeric("-12,5")` → `True`

---

## ✓ StringContainsAny

Sucht nach beliebigen Begriffen in einem String.

`StringContainsAny("Das ist toll", Array("toll", "super"))` → `True`

---

## ✓ StringIsEmpty

Erkennt leere, null oder nur mit Leerzeichen gefüllte Strings.

`StringIsEmpty(" ")` → `True`

---

## ✓ StringExtractEmails

Findet alle E-Mail-Adressen im Text.

`StringExtractEmails("Kontakt: test@mail.com oder info@firma.de")`  
→ `Array("test@mail.com", "info@firma.de")`

---

## ✓ StringClean

Bereinigt überflüssige Leerzeichen und Steuerzeichen.

`StringClean(" Hallo Welt ")` → `"Hallo Welt"`

## ✓ StringRepeat

Wiederholt einen Text beliebig oft hintereinander.

`StringRepeat("-", 10)` → "-----"

---

## ✓ StringPadNumber

Füllt Zahlen mit führenden Zeichen (z. B. Nullen) auf.

`StringPadNumber(42, 5, "0")` → "00042"

---

## ✓ StringWrap

Bricht lange Texte auf eine bestimmte Zeilenlänge um.

`StringWrap("Dies ist ein sehr langer Text", 10)`

---

## ✓ StringToInitials

Erzeugt Initialen aus Namen oder Begriffen.

`StringToInitials("Max Mustermann")` → "MM"

---

## ✓ StringMaskSensitive

Verdeckt sensible Textteile – zeigt nur gezielte Abschnitte an.

`StringMaskSensitive("Kreditkarte1234", 4)` → "\*\*\*\*\*1234"

---

## ⚙ Einsatz & Integration

Alle Funktionen sind kommentiert, validieren Eingaben und lassen sich direkt in bestehende Projekte einfügen. Mit dem Testformular kannst du jede Funktion sofort ausprobieren – ohne Setup, ohne Add-In. Du brauchst nur das Modul **modHelper\_String**. Alles andere ist nur zum testen und präsentieren.

---

## 🚀 Fazit: Mehr Power für deinen VBA-Alltag

Mit diesen 19 Developer Helpers wird dein Umgang mit Strings **schneller, sauberer und deutlich effizienter**.

- ◆ Weniger Code
- ◆ Weniger Fehler
- ◆ Mehr Produktivität

## 🧪 Einfach ausprobieren

Anhand meiner kleinen **Testing-Form** könnt ihr die Funktionen direkt ausprobieren. Das macht es einfach, die Funktionen Schritt für Schritt kennenzulernen. Du entwickelst viel mit Access? **Dann probier's aus** – ich freu mich auf dein Feedback oder Ideen für weitere Funktionen! 🚀