

Lehrziele:

- Methoden
- Strings

Aufgabenstellung

Gesucht sind **Neu**-Implementierungen von bereits bestehenden .net „String“-Methoden.

- **Contains**
Diese Methode bekommt als Parameter einen String (in dem gesucht werden soll) und ein Char (was gesucht werden soll). Der Rückgabewert ist „true“ wenn das Char gefundenen werden konnte, „false“ sonst.
- **IndexOf**
Diese Methode bekommt als Parameter einen String (in dem gesucht werden soll) und ein Char (was gesucht werden soll). Der Rückgabewert ist der **erste** Index des gefundenen Char oder -1 wenn nicht gefunden.
- **LastIndexOf**
Diese Methode bekommt als Parameter einen String (in dem gesucht werden soll) und ein Char (was gesucht werden soll). Der Rückgabewert ist der **letzte** Index des gefundenen Char oder -1 wenn nicht gefunden.
- **TrimStart, TrimEnd und Trim**
Diese Methoden bekommen als Parameter einen String und ein Char. TrimStart (und Trim) entfernt alle führenden Char, TrimEnd (und Trim) alle Char am Ende.
- **Substring**
Aus dem als Parameter übergebenen String wird ein Teilstring erstellt. Der zweite Parameter legt dabei fest, ab welchem Index (des Eingabestrings) der neue String beginnen soll. Der dritte Parameter bestimmt, wie groß der Teilstring **maximal** werden darf.
Beispiel: SubString(„Hallo“,2,2) hat als Ergebnis „ll“ – die beiden ‚l‘ ab Index 2.
Beispiel: SubString(„Hallo“,2,2000) hat als Ergebnis „llo“
- **Remove**
Die Methode entfernt von einem String ab einem übergebenen Index eine definierte Anzahl von Zeichen.
Beispiel: Remove(„Hallo“,2,1) = „Hallo“ – ein Zeichen ab der Position 2 wird entfernt.
Beispiel: Remove(„Hallo“,2,100) = „Ha“ – alle Zeichen ab der Position 2 wird entfernt.

Zum Testen der Methoden schreiben sie ein (Haupt-)Programm. Dabei werden zuerst zwei Char eingelesen (das erste Char für den Test der Methoden Contains, IndexOf und LastIndexOf, das zweite Char für die Trim Methoden). Falscheingaben sind zu prüfen und gegebenenfalls ist die Eingabe zu wiederholen.

Anschließend erfolgt eine String-Eingabe. Mit diesem String werden die Methoden aufgerufen und das Ergebnis ausgegeben. Die String-Eingabe wird so lange wiederholt, bis der Benutzer einen Leerstring eingibt.

Ausgabebeispiel:

```
String Methods
*****
Char for "IndexOf": 1
Char for "Trim":    $
String: $$Hallo Welt$$$
IndexOf("$$Hallo Welt$$$", 'l', 0) = 4
LastIndexOf("$$Hallo Welt$$$", 'l') = 10
Contains("$$Hallo Welt$$$", 'l') = True
TrimStart("$$Hallo Welt$$$", '$') = "Hallo Welt$$$"
TrimEnd("$$Hallo Welt$$$", '$') = "$$Hallo Welt"
Trim("$$Hallo Welt$$$", '$') = "Hallo Welt"
Remove("$$Hallo Welt$$$", 2, 2) = "$$llo Welt$$$"
String: Nur ein Text
IndexOf("Nur ein Text", 'l', 0) = -1
LastIndexOf("Nur ein Text", 'l') = -1
Contains("Nur ein Text", 'l') = False
TrimStart("Nur ein Text", '$') = "Nur ein Text"
TrimEnd("Nur ein Text", '$') = "Nur ein Text"
Trim("Nur ein Text", '$') = "Nur ein Text"
Remove("Nur ein Text", 2, 2) = "Nuein Text"
String:
```