

**String**

**COLLABORATORS**

	<i>TITLE :</i> String		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 7, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>String</b>	<b>1</b>
1.1	String V1.50 . . . . .	1
1.2	nasc . . . . .	2
1.3	nlen . . . . .	2
1.4	nchr\$ . . . . .	2
1.5	nright\$ . . . . .	2
1.6	nleft\$ . . . . .	2
1.7	nucase\$ . . . . .	3
1.8	nlcase\$ . . . . .	3
1.9	nmki\$ . . . . .	3
1.10	nmkl\$ . . . . .	3
1.11	ncvl . . . . .	4
1.12	ncvi . . . . .	4

---

# Chapter 1

# String

## 1.1 String V1.50

String V1.50 General Information:

- \* Blitz Basic II library number : #166
- \* Library size when linked to executable: 150 bytes
- \* Number of commands : 11
- \* Resources automatically freed at end : Yes

Commands summary:

NAsc  
Function (Word)

NChr\$  
Function (String)

NCvi  
Function (String)

NCvl  
Function (String)

NLCase\$  
Function (String)

NLeft\$  
Function (String)

NLen  
Function (Long)

NMki\$  
Function (Word)

NMkl\$  
Function (Long)

```
NRight$  
Function (String)
```

```
NUCase$  
Function (String)
```

## 1.2 nasc

SYNTAX

```
Asc.w = NAsc(String$)
```

FUNCTION

Returns the ascii value of the first character in string\$.

## 1.3 nlen

SYNTAX

```
length.w = NLen(String$)
```

FUNCTION

Returns the character-length of the string.

## 1.4 nchr\$

SYNTAX

```
Text$ = NChr$(ASCII Value)
```

FUNCTION

Returns the letter associated with the given ASCII value.

## 1.5 nright\$

SYNTAX

```
Result$ = NRigth$(String$, Length)
```

FUNCTION

Returns the characters from the right of the string with the given length. This function won't crash if given an incorrect value for the length parameter, it will return the 'best' matching result.

## 1.6 nleft\$

---

## SYNTAX

```
Result$ = NLeft$(String$, Length)
```

## FUNCTION

Returns the characters from the left of the string with the given length. This function won't crash if given an incorrect value for the length parameter, it will return the 'best' matching result.

## 1.7 nucase\$

## SYNTAX

```
Result$ = NUCase$(String$)
```

## FUNCTION

Returns the original string converted to Upper-Case characters (if possible). This command also supports accented letters, so if an 'é' is found, it will be converted into Upper 'é'. Passing a null string causes the function to return a null string.

## 1.8 nlcase\$

## SYNTAX

```
Result$ = NLCase$(String$)
```

## FUNCTION

Returns the original string converted to Lower-Case characters (if possible). This command also supports accented letters, so if an Upper 'é' is found, it will be converted into 'é'. Passing a null string causes the function to return a null string.

## 1.9 nmki\$

## SYNTAX

```
Result$ = NMki$(Word number)
```

## FUNCTION

Converts a given word number into a 2-character string equivalent. It's often used to save disk space when saving numbers to disk. To convert the String back into a number, you can use the NCvi() function.

## 1.10 nmkl\$

## SYNTAX

```
Result$ = NMkl$(Long number)
```

## FUNCTION

Converts a given long number into a 4-character string equivalent. It's often used to save disk space when saving numbers to disk. To convert the String back into a number, you can use the `NCvl()` function.

## 1.11 `ncvl`

## SYNTAX

```
Result.l = NCvl(String$)
```

## FUNCTION

Converts a given 4-character string into a number equivalent. It's the complementary function to `NMkl()`.

## 1.12 `ncvi`

## SYNTAX

```
Result.w = NCvi(String$)
```

## FUNCTION

Converts a given 2-character string into a number equivalent. It's the complementary function to `NMki()`.

---