

Accessors

Access data.

BUTFIRST (BF)

Reports all but the first element of its input.

Syntax

```
BUTFIRST word  
BUTFIRST list
```

Description

BUTFIRST reports all but the first element of its input. If its input is a list, BUTFIRST reports a list containing all but the first element. If its input is a word or number, BUTFIRST reports all the characters of the word or number except the first character. See also [BUTMEMBER](#), [BUTLAST](#), [FIRST](#), and [LAST](#).

Example

```
BUTFIRST [MARY HAD A LITTLE LAMB]  
Result: [HAD A LITTLE LAMB]  
BUTFIRST "WHEAT"  
Result: HEAT  
BUTFIRST [WHEAT]  
Result: []  
BUTFIRST 2135  
Result: 135  
BUTFIRST [[JANUARY FEBRUARY][MARCH APRIL] [MAY JUNE]]  
Result: [[MARCH APRIL] [MAY JUNE]]
```

BUTLAST (BL)

Reports all but the last element of its input.

Syntax

```
BUTLAST word  
BUTLAST list
```

Description

BUTLAST reports all but the last element of its input. If the input is a list, BUTLAST reports a list containing all but the last element. If the input is a word or number, BUTLAST reports all the characters of the word or number except the last character. See also [BUTMEMBER](#), [BUTFIRST](#), [FIRST](#), and [LAST](#).

The procedure in the example makes a plural word or list of words into singular form (for plural words that end in the letter S).

Example

```

TO SINGULAR :WORD
  IF EMPTY? :WORD THEN STOP
  PRINT BUTLAST FIRST :WORD
  SINGULAR BUTFIRST :WORD
END
SINGULAR defined
SINGULAR [CATS]
CAT
SINGULAR [BOOKS TOOLS FLOWERS EYES RUNS]
BOOK
TOOL
FLOWER
EYE
RUN

```

BUTMEMBER (BM)

Removes an element from its input.

Syntax

```
BUTMEMBER word.or.list1 word.or.list2
```

Description

BUTMEMBER reports a word or list consisting of its second input with all occurrences of its first input removed. If the second input is a word, the first input must also be a word. If the second input is a list, the first input can be either a word or list. See also [BUTFIRST](#) and [BUTLAST](#).

Example

```

BUTMEMBER "AM [HI I AM FRED]
Result: [HI I FRED]
BM "D "ABCDABCDABCD
Result: ABCABCABC
BUTMEMBER 22 [11 22 33 44 55]
Result: [11 33 44 55]
BUTMEMBER [JANUARY 1] [[JANUARY 1][JULY 4][DECEMBER 25]]
Result: [[JULY 4][DECEMBER 25]]

```

FIRST

Reports the first element of its input.

Syntax

```

FIRST word
FIRST list

```

Description

FIRST reports the first element of its input. If the input is a word, FIRST reports the first character. If the input is a list, FIRST reports the first element of that list. If the word or list is empty, there is no first element, and FIRST throws an error. See also [BUTFIRST](#), [BUTLAST](#), and [LAST](#).

Example

```
FIRST [MARY HAD A LITTLE LAMB]
Result: MARY
FIRST "WHEAT
Result: W
FIRST 2135
Result: 2
```

FLATTEN

Reports a flat version of its list input.

Syntax

```
FLATTEN list
```

Description

FLATTEN takes a list as its input and converts it to a flat list. The elements of all sub-lists in the list are appended to that flat list, as are the elements of their sub-lists. If FLATTEN is called with an input that is not a list, FLATTEN reports its input.

Example

```
FLATTEN [HI [[LUCY ANN] SMITH]]
Result: [HI LUCY ANN SMITH]
FLATTEN 123
Result: 123
```

FROMMEMBER (FM)

Removes the first part of its input until a pattern is found.

Syntax

```
FROMMEMBER word/list word/list
```

Description

FROMMEMBER reports a word or list consisting of its second input with all elements removed up until the first occurrence of its first input. If the second input is a word, the first input must also be a word. If the second input is a list, the first input can be either a word or a list.

Example

```
FROMMEMBER "B "ABC
Result: BC
FROMMEMBER 3 [1 2 3 4 5]
Result: [3 4 5]
FROMMEMBER "HAT "MANHATTAN
Result: HATTAN
FM "CHARLIE [ARCHIE BETSY CHARLIE DINAH EDWARD FRANCIS]
Result: [CHARLIE DINAH EDWARD FRANCIS]
```

ITEM

Returns a specific element of its input.

Syntax

```
ITEM number word
ITEM number list
```

Description

ITEM reports the nth element from the second input where n is the first input, an integer, and the second input is a number, word, or list. See also [MEMBER?](#).

Example

```
ITEM 3 "CAT
Result: T
ITEM 2 753
Result: 5
ITEM 3 [IN AT ON]
Result: ON
```

JOIN

Converts a list to a word.

Syntax

```
JOIN list separator
```

Description

JOIN concatenates each element of the list it receives as its first input and insert the separator given as the second input between each list element. It reports the resulting word.

Use [SPLIT](#) to convert a JOINed word into a list.

Example

```
JOIN [A B C D E] "| is |  
Result: A is B is C is D is E  
SPLIT "A,B,C,D,E "  
Result: [A B C D E]
```

LAST

Reports the last element of its input.

Syntax

```
LAST word  
LAST list
```

Description

LAST reports the last element of its input. If the input is a word, LAST reports the last character. If the input is a list, LAST reports the last element of that list. If the word or list is empty, there is no last element, and LAST throws an error. See also [BUTFIRST](#), [BUTLAST](#), and [FIRST](#).

Example

```
LAST [MARY HAD A LITTLE LAMB]  
Result: LAMB  
LAST "WHEAT  
Result: T  
LAST 2135  
Result: 5
```

LOWERCASE

Converts its argument to lower case.

Syntax

```
LOWERCASE word
```

Description

LOWERCASE converts its input to lower case. See also [MIXEDCASE](#) and [UPPERCASE](#).

Example

```
LOWERCASE "HELLO  
Result: hello  
LOWERCASE "|Hello|  
Result: hello
```

MIXEDCASE

Converts its argument to mixed case.

Syntax

```
MIXEDCASE word
```

Description

MIXEDCASE converts its input to mixed case. Every word of its input starts with an upper case letter, while the remainder of the word is lower case. See also [LOWERCASE](#) and [UPPERCASE](#).

Example

```
MIXEDCASE "HELLO  
Result: Hello  
MIXEDCASE "|hello world|  
Result: Hello World
```

PICK

Randomly picks an element.

Syntax

```
PICK object
```

Description

PICK picks a randomly selected element from its input, a word or list.

Example

```
MAKE "MUSIC [JAZZ POP ROCK CLASSIC]  
PICK :MUSIC  
Result: POP  
PICK :MUSIC  
Result: CLASSIC
```

REMOVE

Removes elements from a word or a list.

Syntax

```
REMOVE thing wordOrList
```

Description

REMOVE outputs a copy of “wordOrList” with every member equal to “thing” removed. If “wordOrList” is a word, the “thing” cannot be a list.

Example

```
REMOVE "A "ABCABC  
Result: BCBC  
REMOVE [A] [[A] B C A B C]  
Result: [B C A B C]
```

SPLIT

Splits a word into a list using a separator.

Syntax

```
SPLIT word separator
```

Description

SPLIT uses the separator character(s) given as its second input to split a word into a list of words. USE [JOIN](#) to convert such a list back to a word.

Example

```
SPLIT "A,B,C,D,E "  
Result: [A B C D E]  
JOIN [A B C D E] "| is |"  
Result: A is B is C is D is E
```

UPPERCASE

Converts its argument to upper case.

Syntax

```
UPPERCASE word
```

Description

UPPERCASE converts its input to upper case. See also [LOWERCASE](#) and [MIXEDCASE](#).

Example

```
UPPERCASE "HELLO
```

```
Result: HELLO
```

```
UPPERCASE "|hello world|
```

```
Result: HELLO WORLD
```