

## WORD AND SENTENCE MANIPULATION PROCEDURES

The first chapter of the textbook deals exclusively with numeric data. To allow some variety, with interesting examples that aren't about calculus, we are going to use some additional Scheme procedures that manipulate linguistic data: words and sentences. A word can be considered as a string of characters, such as letters and digits. (Numbers can be treated as words.) A sentence is a string of words in parentheses.

## PROCEDURES TO TAKE APART WORDS AND SENTENCES:

FIRST	returns the first character of a word, or the first word of a sentence
BUTFIRST	returns all but the first character of a word, or all but the first word of a sentence
BF	same as BUTFIRST
LAST	returns the last character of a word, or the last word of a sentence
BUTLAST	returns all but the last character of a word, or all but the last word of a sentence
BL	same as BUTLAST

Examples:

```
> (first 'hello)
h

> (butlast '(symbolic data are fun))
(symbolic data are)
```

## PROCEDURES TO COMBINE WORDS AND SENTENCES

WORD	arguments must be words; returns the word with all the arguments strung together
SENTENCE	returns the sentence with all the arguments (words or sentences) strung together
SE	same as SENTENCE

Examples:

```
> (word 'now 'here)
nowhere

> (se 'lisp '(is cool))
(lisp is cool)
```

## PREDICATE PROCEDURES

**EQUAL?** returns true if its two arguments are the same word or the same sentence (a one-word sentence is not equal to the word inside it)

**MEMBER?** returns true if the first argument is a member of the second; the members of a word are its letters and the members of a sentence are its words

**EMPTY?** returns true if the argument is either the empty word [which can be represented as "" ] or the empty sentence [which can be represented as '() ]

## MISCELLANEOUS

**COUNT** returns the number of letters in the argument word, or the number of words in the argument sentence.

**ITEM** takes two arguments: a positive integer N, and a word or sentence; returns the Nth letter of the word, or the Nth word of the sentence (counting from 1).

Examples:

```
(define (buzz n)
  (cond ((member? 7 n) 'buzz)
        ((= (remainder n 7) 0) 'buzz)
        (else n) ))

(define (plural wd)
  (if (equal? (last wd) 'y)
      (word (bl wd) 'ies)
      (word wd 's) ))
```