(define (make-table)

(cons '\*table\* '()))

(define (empty-table? t) (null? (cdr t)))

(define (lookup key table)

(let ((record (associ key (cdr table))))

(if record

(cdr record)

#f)))

(define (associ key records)

(cond ((null? records) #f)

((equal? key (caar records)) (car records))

(else (associ key (cdr records)))))

**(define (insert! key value table)**

**(let ((record (associ key (cdr table))))**

**(cond ((null? (cdr table))**

**(set-cdr! table (cons (cons key value) '())))**

**(record (set-cdr! record value))**

**(else (set-cdr! table**

**(cons (cons key value) (cdr table)))))))**

**(define (delete! key table)**

**(cond ((null? (cdr table)) #f)**

**((equal? (caadr table) key) (set-cdr! table (cddr table)))**

**(else (delete! key (cdr table)))))**