```
;; equality
(eq    a b) ;reference
(eql   a b) ;atom val
(equal a b) ;structural
(equalp a b) ;semantic


;; numbers
2  3.5  1/3  6.02e23
+ - * / 1+ 1-
min max evenp oddp zerop
= /=  >  < >= <=
(incf x) ;increment
(decf x) ;decrement


;; logic
T  NIL
(not a)
(and a b..) ; short
circuit
(or a b..)  ; short
circuit
```

```
;; sequence functions
;; (list, vector, string)
(length s)
(reverse s)
(nreverse s)
(subseq s start [end])
(copy-seq s)
(some pred s)
(reduce fn seq)
(map type fn seq)
;every, notany notevery


;; lists
() (1 2 3) (1 . 2)
(cons h t)
(car p)     (cdr p)
(rplaca p h) (rplacd p t)
(null e)
(listp lst)
(length lst)
(list args..)
```

```
;; vector
#() #(0 1 2)
(make-array dim)
(vector args..)
(length v)
(aref v ix)
(vector-push v e)
(vector-pop v)


;; I/O
(read strm?)     ;s-expr
(print val strm?) ;s-expr
(print val strm?) ;value
(open filename)
; :direction :output
(close strm)
(with-open-file (var
args..)
  body..)
```

```
;; characters
#\a #\1 #\Newline #\Space
(char= c1 c2) ; /=, <, etc
(char-code c)
(make-char i)


;; strings
"hello"
(format strm s args..)
;nil->return, t->console
; ~A  ascii   ~S  s-exp
; ~D  dec     ~B  binary
; ~O  octal   ~X  hexadec
; ~%  newline ~~ tilde
; ~nR radix (2-36)
; ~F ~E ~G  floats


;; TODO:
; -- paths
; -- clos
```

Common Lisp

Programming Language
Quick Reference Booklet

draft 1

```
;; self documentation
(apropos "name") ;search
(describe 'name) ;docs

;; loading code
(load "filename")

;; packages: hold symbols
(import 'qual:symbol..)
(defpackage :name
  (:use packagename)
  (:export symbol..)..)
(in-package :name)

;; variables
(let ((x val) y ..)
  body..)
(let* ..) ; sequential
(defvar *x* val?)
(setf x val)
```

```
;; functions
(defun name (args..)
  body..)
(lambda (args..) body..)
(labels ((fn (args..))..)
  body..)

;; control flow
(if expr
  truebody
  falsebody?)
(when expr body..)
(unless expr body..)
(cond
  (expr body)..
  (t elsebody))
(do ((var init next?)..)
    (endtest result?)
  loopbody..)
;do* initializes serially
```

Community References:
http://www.cliki.net/index

Hyperspec:
http://www.lispworks.com/documentation/HyperSpec/Front/

Books (online):
http://www.cs.cmu.edu/Groups/AI/html/cltl/clm/clm.html
http://www.gigamonkeys.com/book/

Other quick-references:
http://clqr.berlios.de/
http://hyperpolyglot.org/lisp

© 2011 Aaron Lahman
http://ra3s.com/