## INTERVALS

```
;; SeatNum is Integer[1,32]
;;          Number[1,32] includes decimals
;;          Natural[1,32) not including 32

;; interp. seat numbers in a row, 1 and 32 are aisle seats

(define SN1 1)  ; aisle
(define SN2 12) ; middle
(define SN3 32) ; aisle

#;
(define (fn-for-seat-num sn)
  (...sn)
  )

;; Template rules used:
;; -atomic, Non-distinct: Integar[1,32]
```

## ENUMERATION

```
;; LetterGrade is one of:
;; - "A"
;; - "B"
;; - "C"

;;interp. The letter grade in a course.
;; examples are redundant for enumerations

(define (fn-for-letter-grade lg)
  (cond [(string=? lg "A") (...)]
        [(string=? lg "B") (...)]
        [(string=? lg "C") (...)]

;; Template rules used:
;; - One of: 3 cases
;;       Atomic distinct value: "A"
;;       Atomic distinct value: "B"
;;       Atomic distinct value: "C"
```

# ITEMIZATION

```
;; CountDown is one of:
;;  - false
;;  - Natural[1,10]
;;  - "complete"

;; interp.
;; false         means countdown has not started
;; Natural[1,10] means countdown is running and how many seconds left
;; "complete"    means countdown is over

(define CD1 false)
(define CD2 10)  ; just started
(define CD3 1)   ; almost over
(define CD4 "complete")

#;
(define (fn-for-countdown c)
  (cond [(false? c) (...)]
        [(and (number? c) (<= 1 c) (<= c 10)) (... c)]
        [else (...)]))


;; Template rules used:
;; One of: 3 cases
;;     Atomic distinct: false
;;     Atomic non-distinct: Natural [1,10]
;;     Atomic distinct: "complete"
```