

```
1 #lang racket
2
3 (struct Bool (val) #:transparent)
4
5 (Bool 1)
6
7 (define true (Bool 1))
8 true
9
10 (Bool 1)
11 (Bool 0)
12 (Bool #\ $)
13
14 (define (make-Bool v)
15   (match v
16     (1 (Bool v))
17     (0 (Bool v))
18     (_ (printf "Error: not a valid value for a
18 Boolean!\n"))))
19
20 (make-Bool 1)
21 (make-Bool 0)
22 (make-Bool #\ $)
23
24 (define (our-and v1 v2)
25   (match v1
26     ((Bool 1)(if (= 1 (Bool-val v2)) (Bool 1) (Bool 0)))
27     ((Bool 0)(Bool 0))))
28
29 (our-and true true)
30 (our-and true (Bool 0))
31
```