

```
1 #lang racket
2
3 (define x 2)
4
5 (match x
6   (5 "five")
7   (10 "ten")
8   (else "other"))
9
10 (cond ((equal? x 5) "five")
11       ((equal? x 10) "ten")
12       (else "other"))
13
14 ;; using (? pattern exp) for type-checking
15
16 (define (check-type y)
17   (match y
18     (_ "string")
19     ((? number? x) "number")
20     ((? boolean? x) "bool")
21     (else "I don't know")))
22
23 (check-type 10)
24
25 (define (get-last lst)
26   (match lst
27     ((list a) a)
28     ((list _ ... a) a)))
29
30 (get-last (list 1 2 4 5))
```