

```
1 #lang racket
2 (define numbers (list 7 6 10 1))
3
4 (filter (lambda (x) (= (modulo x 7) 0)) numbers)
5
6 (define (my-filter f lst)
7   (if (empty? lst)
8       lst
9       (if (f (first lst))
10           (cons (first lst) (my-filter f (rest
10 lst))))
11           (my-filter f (rest lst))))))
12
13 (my-filter (lambda (x) (= (modulo x 7) 0))
13 numbers)
14
15 (define (all-titlecase lst)
16   (filter (lambda (x) (equal? (string-titlecase
16 x) x)) lst))
17
18 (all-titlecase (list "cat" "Cat" "CAT" "All
18 Cats"))
```